



HAL
open science

Reconnaissance de partitions musicales par modélisation floue des informations extraites et des règles de notation

Florence Rossant

► **To cite this version:**

Florence Rossant. Reconnaissance de partitions musicales par modélisation floue des informations extraites et des règles de notation. domain_other. Télécom ParisTech, 2006. English. NNT: . pastel-00002037

HAL Id: pastel-00002037

<https://pastel.hal.science/pastel-00002037>

Submitted on 29 Jan 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



École Doctorale
d'Informatique,
Télécommunications
et Électronique de Paris

Thèse

présentée pour obtenir le grade de Docteur
de l'École Nationale Supérieure des Télécommunications

Spécialité : **Signal et Images**

Florence ROSSANT

Reconnaissance de partitions musicales par
modélisation floue des informations extraites
et des règles de notation

Soutenue le 06 Octobre 2006 devant le jury composé de :

| | |
|-------------------|--------------------|
| Jaime Lopez-Krahe | Président |
| Karl Tombre | Rapporteurs |
| Jean Camillerapp | |
| Amara Amara | Examineurs |
| Michel Ciazynski | |
| Isabelle Bloch | Directeur de thèse |

REMERCIEMENTS

Je tiens à exprimer tous mes remerciements à Isabelle Bloch, Professeur à l'ENST, qui m'a suivie et conseillée tout au long de cette thèse. Je ne pouvais avoir meilleur guide pour mes débuts dans la recherche. Je lui suis particulièrement reconnaissante pour sa disponibilité, sa gentillesse, et pour le soutien et la confiance qu'elle m'a toujours accordés. Cette collaboration a été pour moi une source d'enrichissements sur bien des plans.

Mes remerciements vont également à Michel Ciazynski, Directeur de l'ISEP, Amara Amara, Directeur de Recherche à l'ISEP, qui m'ont donné l'opportunité de me consacrer à ce thème de recherche. Je les remercie vivement pour leur confiance, leurs encouragements et leur participation à mon jury. Bien entendu, je n'oublie pas Michel Terré, Maître de Conférences au CNAM, anciennement responsable du département Télécoms de l'ISEP, qui m'a, à l'origine, incitée à me lancer dans cette voie.

Je remercie vivement Jaime Lopez-Krahe, Professeur à l'Université Paris 8, qui a accepté de présider le jury, ainsi que les rapporteurs, Karl Tombre, Professeur à l'École des Mines de Nancy et Jean Camillerapp, Professeur à l'INSA de Rennes, pour tout le temps qu'ils ont consacré à l'étude approfondie de ce manuscrit. Leurs commentaires et leurs conseils m'ont permis de l'améliorer, et me seront profitables dans mes futures activités de recherche.

Cette thèse n'aurait pas eu lieu sans les « coups de pouce » décisifs de Bernard Robinet, Directeur de l'EDITE de Paris. Bernard Robinet est le premier maillon qui m'a conduit à rencontrer Isabelle Bloch, et l'ultime maillon qui m'a permis d'aller plus loin dans ce projet de recherche et de réaliser cette thèse. Je tiens à lui adresser toute ma reconnaissance, et je le remercie très vivement pour l'intérêt qu'il a porté à mon travail, la confiance qu'il m'a manifestée.

Un grand merci à tous mes collègues de l'ISEP, qui m'ont témoigné beaucoup de gentillesse lors des pics de stress... et qui m'ont apporté leur aide au quotidien. Un merci spécial à Béata Mikovicova, dont l'amitié m'a été précieuse.

Un dernier merci, et non des moindres, à mes principaux « supporters » : mon mari, Philippe, mes enfants, Clarence et Maxence, et mes parents. C'est grâce à leur indispensable soutien, leurs encouragements et leur compréhension que j'ai pu réaliser cette thèse.

RESUME

Nous présentons dans cette thèse une méthode complète de reconnaissance de partitions musicales imprimées, dans le cas monodique. Le système procède en deux phases distinctes :

- La segmentation et l'analyse des symboles (essentiellement par corrélation), conçues pour surmonter les difficultés liées aux interconnexions et aux défauts d'impression, aboutissant à des hypothèses de reconnaissance.
- L'interprétation de haut niveau, fondée sur une modélisation floue des informations extraites de l'image et des règles de notation, menant à la décision.

Dans cette approche, la décision est reportée tant que le contexte n'est pas entièrement connu. Toutes les configurations d'hypothèses sont successivement évaluées, et la plus cohérente est retenue, par optimisation de tous les critères. Le formalisme utilisé, fondé sur la théorie des ensembles flous et des possibilités, permet de prendre en compte les différentes sources d'imprécision et d'incertitude, ainsi que la souplesse et la flexibilité de l'écriture musicale. Afin de gagner en fiabilité, nous proposons également des méthodes d'indication automatique des erreurs potentielles de reconnaissance, ainsi qu'une procédure d'apprentissage, optimisant les paramètres du système pour le traitement d'une partition particulière. Les performances obtenues sur une large base de données ont permis de montrer l'intérêt de la méthode proposée.

ABSTRACT

This thesis deals with Optical Music Recognition (OMR), in case of monophonic typeset music. The proposed method relies on two separated stages:

- The symbol segmentation and analysis step, designed in order to deal with common printing defects and numerous symbol interconnexions. A set of recognition hypotheses is generated, based on correlation scores with class reference models.
- A high-level interpretation step, based on the fuzzy modeling of the extracted information and of musical rules, leading to the decision.

In this approach, the decision is delayed until the entirely context can be evaluated. All the hypothesis configurations are considered, and the decision is taken through a global consistency evaluation. This high-level interpretation step relies on the fuzzy sets and possibility framework, since it allows dealing with symbol variability, the flexibility and the imprecision of music rules, and merging all these heterogeneous pieces of information. Other innovative features are the indication of potential errors, and the possibility of applying learning procedures, in order to gain in robustness. Experiments conducted on a large data base show that the proposed method constitutes an interesting contribution to OMR.

TABLE DES MATIERES

| | |
|---|-----------|
| Table des Matières..... | 5 |
| Introduction | 9 |
| Chapitre 1..... | 13 |
| Principales méthodes de lecture automatique de partitions musicales..... | 13 |
| 1.1. Quelques rappels sur la notation musicale classique | 13 |
| 1.2. Difficultés propres à l'écriture et à l'édition musicale..... | 19 |
| 1.3. Méthodes existantes | 22 |
| 1.3.1. Stratégies générales..... | 22 |
| 1.3.2. Détection des portées | 24 |
| 1.3.3. Segmentation..... | 29 |
| 1.3.4. Méthodes d'analyse des symboles | 33 |
| 1.3.5. Modélisations structurelles et syntaxiques..... | 40 |
| 1.3.6. Prise en compte de l'incertitude..... | 43 |
| 1.3.7. Principaux systèmes et évaluation | 46 |
| 1.4. Conclusion | 47 |
| Chapitre 2 | 51 |
| Structure du système de reconnaissance proposé | 51 |
| 2.1. Type de partitions traitées et objectifs | 51 |
| 2.2. Acquisition et format des images..... | 53 |
| 2.3. Présentation générale du système | 54 |
| 2.4. Discussion..... | 55 |
| Chapitre 3..... | 59 |
| Prétraitements et segmentation..... | 59 |
| 3.1. Prétraitements..... | 59 |
| 3.1.1. Redressement de l'image | 62 |
| 3.1.2. Détection et caractérisation des portées | 64 |
| 3.1.3. Poursuite des portées..... | 71 |
| 3.1.4. Conclusion | 75 |
| 3.2. Segmentation..... | 76 |
| 3.2.1. Effacement des lignes de portée..... | 77 |
| 3.2.2. Détection des symboles caractérisés par un segment vertical..... | 81 |
| 3.2.3. Images des silences | 91 |
| 3.2.4. Résultats et conclusion..... | 92 |

| | |
|---|------------|
| Chapitre 4..... | 95 |
| Analyse individuelle des symboles | 95 |
| 4.1. Mise en correspondance avec des modèles..... | 95 |
| 4.2. Analyse des symboles caractérisés par un segment vertical..... | 98 |
| 4.2.1. Préclassification | 98 |
| 4.2.2. Zones de calcul de la corrélation..... | 101 |
| 4.2.3. Cas des barres de mesure | 103 |
| 4.2.4. Génération d'hypothèses | 104 |
| 4.2.5. Analyse de la hauteur des notes et altérations..... | 107 |
| 4.2.6. Durée des notes : résultats préliminaires..... | 107 |
| 4.2.7. Conclusion | 111 |
| 4.3. Analyse des autres symboles..... | 112 |
| 4.3.1. Zones de corrélation pour les silences situés sur la troisième ligne de portée et les rondes..... | 112 |
| 4.3.2. Zones de corrélation pour les silences inclus dans des groupes de notes | 113 |
| 4.3.3. Génération d'hypothèses de reconnaissance (silences et rondes) | 114 |
| 4.3.4. Points allongeant la durée des silences | 117 |
| 4.3.5. Conclusion | 118 |
| 4.4. Choix du modèle de classe en fonction de la partition | 119 |
| 4.5. Exemples et conclusion..... | 119 |
| Chapitre 5..... | 123 |
| Modélisation floue | 123 |
| 5.1. Objectifs | 123 |
| 5.2. Modélisation des classes de symboles | 127 |
| 5.3. Cohérence graphique..... | 129 |
| 5.3.1. Compatibilité graphique entre une altération accidentelle et une note | 130 |
| 5.3.2. Compatibilité graphique entre une appoggiature et une note | 132 |
| 5.3.3. Compatibilité graphique entre une note et un point de durée | 133 |
| 5.3.4. Compatibilité graphique entre un point et une note de son voisinage | 134 |
| 5.3.5. Compatibilité graphique entre deux symboles quelconques..... | 135 |
| 5.3.6. Modification des hypothèses de reconnaissance..... | 136 |
| 5.3.7. Fusion : compatibilité graphique d'un symbole avec tous ses voisins | 137 |
| 5.4. Cohérence syntaxique | 138 |
| 5.4.1. Armure | 138 |
| 5.4.2. Altérations accidentelles | 138 |
| 5.4.3. Métrique..... | 140 |
| 5.5. Fusion des informations et décision..... | 145 |
| 5.5.1. Fusion..... | 146 |
| 5.5.2. Décision | 148 |
| 5.6. Exemples..... | 149 |
| 5.6.1. Exemple 1 | 149 |
| 5.6.2. Exemple 2 | 151 |
| 5.6.3. Exemple 3 | 157 |
| 5.7. Conclusion | 161 |
| Chapitre 6..... | 163 |
| Améliorations de la robustesse..... | 163 |
| 6.1. Détection automatique d'erreurs | 163 |
| 6.1.1. Indication des ajouts et des confusions potentiels | 164 |
| 6.1.2. Détection des symboles manquants | 166 |

| | |
|--|------------|
| 6.1.3. Analyse de la rythmique | 167 |
| 6.1.4. Conclusion | 169 |
| 6.2. Adaptation à la partition analysée | 170 |
| 6.2.1. Apprentissage des modèles de classe | 170 |
| 6.2.2. Apprentissage des paramètres | 171 |
| 6.2.3. Conclusion | 175 |
| 6.3. Conclusion | 176 |
| Chapitre 7..... | 177 |
| Résultats | 177 |
| 7.1. Conditions d'expérimentation et données en sortie du système | 177 |
| 7.1.1. Conditions d'expérimentation | 177 |
| 7.1.2. Données en sortie du programme..... | 178 |
| 7.1.3. Méthode d'évaluation de la précision et de la fiabilité du système..... | 178 |
| 7.2. Résultats sur l'analyse individuelle des symboles..... | 179 |
| 7.2.1. Résultats et analyse | 179 |
| 7.2.2. Conclusion | 184 |
| 7.3. Taux de reconnaissance | 184 |
| 7.3.1. Evaluation du système et analyse des résultats | 185 |
| 7.3.2. Hauteur et durée des notes | 188 |
| 7.3.3. Apport de la modélisation floue..... | 189 |
| 7.3.4. Robustesse aux paramètres | 190 |
| 7.4. Temps de calcul | 191 |
| 7.5. Comparaison avec un logiciel du commerce | 192 |
| 7.6. Résultats sur l'indication des erreurs potentielles | 194 |
| 7.7. Evaluation de la méthode d'apprentissage supervisé | 195 |
| 7.8. Conclusion | 197 |
| Chapitre 8..... | 199 |
| Conclusion | 199 |
| 8.1. Méthode proposée et caractéristiques | 199 |
| 8.2. Compléments | 201 |
| 8.2.1. Améliorations diverses..... | 201 |
| 8.2.2. Compléments dans l'analyse des symboles..... | 202 |
| 8.2.3. Reconnaissance automatique des informations globales | 202 |
| 8.3. Perspectives..... | 203 |
| 8.3.1. Reconnaissance à partir d'images dégradées..... | 203 |
| 8.3.2. Intégration d'informations structurelles | 204 |
| 8.3.3. Structure du système de reconnaissance | 205 |
| 8.4. Extension à la musique polyphonique | 206 |
| Bibliographie..... | 209 |
| Publications | 215 |
| Publications relatives à la thèse | 215 |
| Autres publications | 215 |
| Annexe | 217 |

INTRODUCTION

La reconnaissance de partitions musicales s'inscrit dans le domaine plus vaste de la reconnaissance de documents numérisés. On la désigne généralement par l'acronyme OMR pour Optical Music Recognition. Elle est souvent comparée à l'OCR (Optical Character Recognition), en ce sens qu'elle permet de passer d'une image à la description symbolique puis sémantique de son contenu, par des méthodes de traitement et d'analyse d'images numériques. Dans le cas de la reconnaissance de la musique, un tel procédé permet de rééditer la partition sur ordinateur, ou de la convertir en un format électronique tel que le Midi, permettant de jouer la musique.

Les utilisations possibles d'un logiciel d'OMR sont extrêmement nombreuses, liées, comme nous l'avons évoqué, à l'édition ou à la restitution de la musique, et, de plus en plus, à la constitution de bases de données. Une fois les symboles musicaux reconnus, il est possible de rééditer la partition et de la modifier à loisir : transcription, transposition de tonalité, arrangements, etc. Un gain de temps appréciable est obtenu grâce à la reconnaissance automatique, puisque la tâche de saisie manuelle, particulièrement longue en musique, est évitée. La conversion en un format audio ou Midi permet au musicien d'écouter la partition, de s'accompagner des autres parties musicales, celles-ci pouvant être jouées par un instrument électronique ou un ordinateur. Cette application nécessite de déduire des symboles reconnus l'interprétation finale du morceau (hauteur et durée des notes, durée des silences, phrasé, etc), par une analyse sémantique fondée sur les règles de la notation musicale. Enfin, la représentation symbolique et sémantique d'une partition enrichit considérablement les bases de données, puisque des caractéristiques liées au contenu musical lui-même peuvent être extraites et servir de critères d'indexation et de recherche.

La reconnaissance de partitions musicales est facilitée par les nombreuses informations qui sont disponibles :

- le nombre de symboles est assez restreint, du moins si on ne considère que les symboles nécessaires à la restitution de la mélodie (clés, notes, altérations, points, silences), dans la notation classique ;
- de nombreuses règles codifient les relations structurelles (organisation des groupes de notes par exemple), graphiques (comme la position des symboles sur la portée), et syntaxiques (métrique, tonalité, etc.) entre ces symboles. Ces règles apportent des informations a priori, exploitables pour la reconnaissance.

Introduction

Les problèmes à résoudre sont néanmoins très nombreux et interviennent aux différents stades de l'analyse. On peut citer les difficultés de segmentation dues au fort degré d'interconnexion entre les symboles (par la présence des lignes de portée notamment), toutes les difficultés liées aux défauts d'impression ou à une mise en page souvent approximative (symboles fractionnés, mal positionnés, connexions parasites, etc.), à la variabilité des polices de symboles, la construction complexe des groupes de notes à partir de primitives, la flexibilité des règles musicales. Bien que les partitions musicales soient des documents fortement structurés, suivant des règles apparemment bien définies, on remarque qu'en pratique ces règles sont très souples, soit dans leur paramétrage (déterminant par exemple la position d'une altération par rapport à une note), soit dans leur mode d'application (rappel non obligatoire d'altération, différents groupements de notes pour un même rythme, etc.). Toutes ces spécificités de la notation musicale font que le domaine de l'OMR est finalement fort différent des autres domaines relatifs à l'analyse de documents, en particulier de l'OCR, et qu'il soulève des problèmes techniques particuliers, nécessitant des solutions innovantes.

La recherche dans le domaine de l'OMR a débuté dans les années 70. Les difficultés, liées à la qualité de la numérisation et à la puissance de calcul nécessaire, semblaient cependant insurmontables avec les moyens de l'époque. Depuis, de nombreuses méthodologies ont été proposées, en lien avec les progrès technologiques, et le premier logiciel commercial est apparu sur le marché dès le début des années 90. Néanmoins, comme nous le verrons dans la bibliographie, les solutions proposées jusqu'à présent, complémentaires ou concurrentes, ne sont pas encore totalement satisfaisantes. L'utilisation de produits commerciaux, comme SmartScore [Musitek], conforte cette idée, la reconnaissance échouant dans de nombreuses configurations, probablement parce que les difficultés énumérées ci-dessus ne sont pas encore complètement résolues. On remarque en particulier que ces logiciels sont très sensibles à la qualité d'impression de la partition, et que les règles musicales ne semblent pas suffisamment intégrées dans le processus de décision (erreurs de métrique, altérations mal positionnées ou incohérentes, etc.).

Les axes de recherche portent à la fois sur les méthodes de bas niveau, pour la segmentation et la reconnaissance des symboles, et sur les méthodes de plus haut niveau, en particulier la modélisation et l'intégration des règles musicales. Ce point est particulièrement important, car les travaux menés jusqu'à présent sont généralement limités aux aspects structurels et graphiques, et laissent de côté les aspects syntaxiques, comme les règles relatives à la tonalité et aux altérations. Cela est probablement dû aux difficultés liées à la modélisation de telles règles, à la fusion d'informations aussi hétérogènes. Il faut également remarquer que les méthodes de haut niveau doivent prendre en compte toutes les sources d'imprécision, liées aux informations extraites de l'image, ou à la notation musicale elle-même (souplesse, flexibilité des règles). Cet aspect a été insuffisamment traité jusqu'à présent et, le cas échéant, les solutions proposées restent partielles. En résumé, on peut dire que l'OMR est techniquement un domaine très intéressant, de nombreux problèmes restant ouverts. Il touche au traitement et à l'analyse d'image bien sûr, mais aussi à d'autres domaines connexes, relatifs à la modélisation de contraintes souples, à la fusion d'informations (génériques et provenant de l'image, souvent hétérogènes), à la modélisation de l'imprécision et de l'incertitude, aux méthodes de décision. C'est en particulier à ce niveau que se situent nos contributions.

Notre ambition est de concevoir un système complet d'OMR, pour la notation musicale classique, apportant des réponses aux problèmes précédemment évoqués. Afin que cet objectif soit réalisable dans le cadre d'une thèse, nous nous limiterons au traitement des partitions imprimées monodiques, moins complexes que les partitions polyphoniques. Le système sera testé sur une large base de données, et les performances comparées, sur quelques exemples au moins, avec un logiciel du commerce, afin de valider la méthodologie proposée et d'évaluer sa contribution au domaine de l'OMR. Afin de gagner en fiabilité, nous proposerons également des méthodes permettant d'indiquer automatiquement les erreurs potentielles de reconnaissance, de manière à faciliter la correction du résultat. C'est une voie très novatrice, très peu évoquée dans la bibliographie jusqu'à présent. Elle est pourtant essentielle, car la vérification et la correction entièrement manuelles de partitions complètes est une tâche extrêmement longue et fastidieuse, qui limite l'intérêt des logiciels d'OMR. Enfin, des procédures d'apprentissage supervisé seront proposées, permettant d'améliorer la fiabilité de la reconnaissance de partitions particulières. Ce dernier point, également innovant, peut s'avérer particulièrement intéressant lorsque de grands volumes sont à traiter.

La suite de ce mémoire s'organise en 8 chapitres.

Le premier introduit quelques rappels sur la notation musicale, et met en évidence les difficultés spécifiques au domaine de l'OMR. L'étude bibliographique permettra de résumer l'état de l'art, d'analyser dans quelle mesure les problèmes mentionnés sont résolus, et de dégager les axes de recherche primordiaux.

Le second chapitre présente le système proposé : hypothèses de travail, objectifs, architecture générale. Les différentes étapes de la reconnaissance sont décrites dans les trois chapitres suivants. Le chapitre 3 est consacré aux prétraitements et à la segmentation de l'image, permettant de détecter les portées, de localiser les différentes primitives ou symboles à reconnaître. Ceux-ci sont ensuite analysés, par comparaison avec des modèles de référence. Cette analyse, qui est décrite dans le chapitre 4, aboutit à un ensemble d'hypothèses de reconnaissance : plusieurs classes sont sélectionnées par objet, mais aucune classification définitive n'est effectuée. Les règles structurelles, graphiques et syntaxiques de la notation musicale sont ensuite modélisées et intégrées, afin de lever les ambiguïtés de classification et de prendre une décision globale, cohérente par rapport à la notation musicale, par fusion de tous les critères. La modélisation, la fusion et la décision, fondées sur la théorie des ensembles flous et des possibilités, sont décrites au chapitre 5.

Le chapitre 6 traite des procédures qui permettent de gagner en robustesse : indication des erreurs potentielles et apprentissage supervisé d'une partition spécifique.

Tous les résultats, obtenus sur une large base de données, sont présentés dans le chapitre 7.

Enfin, le dernier chapitre conclut sur l'ensemble de la méthode proposée, en dégageant les points forts, les axes d'améliorations, et les perspectives.

CHAPITRE 1

Principales méthodes de lecture automatique de partitions musicales

Dans ce chapitre, nous présentons des rappels succincts sur l'écriture musicale (notation classique). Nous analyserons ensuite les difficultés propres à la reconnaissance automatique de partitions musicales, certaines résultant directement de la notation elle-même, d'autres de la qualité du document original. Nous terminerons par une étude critique des méthodes déjà proposées, qui nous permettra dans les chapitres suivants de situer notre méthode, de dégager les aspects novateurs contribuant à résoudre certaines difficultés.

1.1. Quelques rappels sur la notation musicale classique

Nous présentons dans cette section quelques rappels sur l'écriture musicale, afin de définir les symboles musicaux que nous cherchons à reconnaître, et de préciser les principes fondamentaux de la théorie musicale, qui permettent d'interpréter les symboles reconnus et de restituer la mélodie. Pour davantage de précisions, on pourra se rapporter à un ouvrage sur la théorie musicale, par exemple [Danhauser 96].

Portées, mesures et symboles

La Figure 1.1a montre un extrait d'une partition musicale :

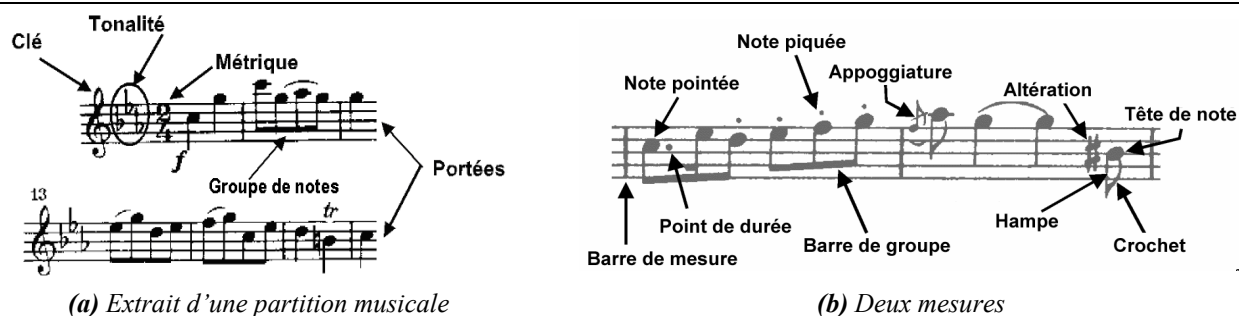


Figure 1.1 : Terminologie musicale

Une partition musicale est composée de portées, une portée étant formée de cinq lignes parallèles

régulièrement espacées. Les symboles musicaux, dont les principaux sont présentés en figure 1.2, sont positionnés relativement à la portée. La plupart se situent sur la portée elle-même, mais certains sont placés au-dessus ou au-dessous. Les portées sont divisées en mesures, une mesure étant constituée de l'ensemble des symboles entre deux barres verticales, appelées barres de mesure (Figure 1.1b). Les portées sont lues de gauche à droite, si bien que l'axe horizontal de l'image représente le temps.

Les notes codent à la fois une hauteur (fréquence de la note) et une durée (pendant laquelle la note est maintenue).

La hauteur est donnée par la position de la tête de note (Figure 1.1b), qui est placée sur les lignes ou entre les lignes de la portée (Figure 1.3). Cette hauteur peut être modifiée par une altération (dièse, bémol ou bécarre) placée devant la note. Le dièse augmente la hauteur de la note d'un demi-ton, le bémol la descend d'un demi-ton, le bécarre annule l'effet d'un précédent dièse ou bémol, et restitue à la note sa hauteur naturelle.

Il y a sept figures de notes : la ronde, la blanche, la noire, la croche, la double croche, la triple croche, et la quadruple croche (Figure 1.2). Chacune de ces notes a une durée spécifique : la durée de référence est celle de la ronde, et toutes les autres s'obtiennent par division par deux. Ainsi la blanche équivaut à une demi-ronde, la noire à une demi-blanche, la croche à une demi-noire, etc. La durée d'une note sera donc déduite de la tête de note, ronde, blanche ou noire, mais aussi, dans ce dernier cas, du nombre de crochets à l'extrémité de la hampe (Figures 1.1b et 1.2). Cependant, des points de durée peuvent être placés après la tête de note ; on multiplie alors sa durée primitive par 1,5. Par exemple, une croche pointée vaut trois doubles croches.

Pour améliorer la lisibilité, les notes peuvent être reliées par des barres, elles forment alors des groupes de notes (Figure 1.1). Le nombre de barres doit être égal au nombre de crochets qu'elles remplacent. En conséquence, la durée de chaque note est indiquée par le nombre maximal de barres à l'extrémité de la hampe. Ainsi, la deuxième note de la figure 1.1b est une double croche.


















| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | 2 | 1 | 1/2 | 1/4 | 4 | 2 | 1 | 1/2 | 1/4 | 1/8 | | | | | | |
| ronde | blanche | noire | croche | double croche | pause | ½ pause | soupirs | 1/2, 1/4, 1/8 de soupir | | | dièse | bémol | bécarre | appoggiature | barre de mesure | point |
| Notes | | | | | Silences | | | | | Altérations | | | Autres | | | |

Figure 1.2 : Principaux symboles musicaux, avec leur durée relative (1 pour une noire)

En plus des barres de mesure, des notes et de leurs modificateurs (altérations, points), il existe des silences (Figure 1.2), qui indiquent une interruption du son. Il y a également sept figures différentes, dont les durées suivent la même logique de division binaire par rapport à la pause qui sert de référence. Les silences peuvent également être allongés par un point. Ils peuvent aussi faire partie de groupes de notes.

Remarquons que nous n'avons indiqué pour l'instant que des durées relatives, en prenant la noire comme unité dans la figure 1.2. Pour connaître la durée absolue de chaque note, il faut connaître la métrique de la partition, notion que nous explicitons ci-dessous.

Informations globales : clé, métrique et tonalité

Des informations globales sont indiquées en début de portée : la clé, la tonalité, et, sur la première portée, la métrique (Figure 1.1a).

La clé définit la référence utilisée pour déduire la hauteur d'une note de sa position sur la portée. Une clé de sol par exemple implique qu'une note placée sur la deuxième ligne de portée est un sol, la première ligne de portée étant la ligne inférieure. Les notes suivent graphiquement sur la portée la progression de la gamme (do, ré, mi, fa, sol, la, si, do).



Figure 1.3 : La gamme (clé de sol)

La tonalité est indiquée par une succession de dièses ou de bémols, juste après la clé, ou après une barre de mesure. Ces altérations, formant l'armure de la clé, suivent un ordre bien défini par la théorie musicale. Elles sont alors implicitement appliquées à toutes les notes du même nom (c'est-à-dire de la même hauteur, à l'octave près), et évitent de surcharger l'écriture musicale.

La métrique indique le nombre de temps par mesure (chiffre supérieur), la référence de temps étant codée par le nombre inférieur. Ces deux chiffres sont disposés sous la forme d'une fraction dont la ronde est l'unité, juste après l'armure.

Avant d'explicitier cette notation, il faut revenir à la notion de mesure. Toutes les mesures d'une partition ont la même durée. La mesure se subdivise en deux, trois ou quatre parties que l'on nomme temps. Il existe deux types de mesures : les mesures simples, dont les temps sont binaires, c'est-à-dire qu'ils sont divisibles par deux, et les mesures composées, dont les temps sont ternaires, c'est-à-dire divisibles par trois.

Dans la mesure simple, le chiffre inférieur précise la durée qu'occupe un temps, 1 représentant la ronde, 2 la blanche, 4 la noire, 8 la croche. Le chiffre supérieur donne le nombre de temps. Par exemple, 2/4 sur la figure 1.1a, indique qu'il y a deux temps par mesure, la durée d'une noire représentant le temps ; 2/8 indiquerait qu'il y a deux temps par mesure, le temps étant cette fois la croche.

Dans la mesure composée (i.e. ternaire), un temps équivaut toujours à un signe pointé, soit, une ronde pointée, une blanche pointée, une noire pointée, ou une croche pointée. Le chiffre inférieur (2, 4, 8, ou 16) précise cette fois la durée qu'occupe un tiers de temps, et le chiffre supérieur indique la quantité de ces valeurs. Par exemple, 6/8 signifie qu'il y a 6 croches par mesure, le temps étant constitué de 3 croches (noire pointée).

Parfois, le découpage temporel d'une mesure simple ne suit plus une logique binaire. Prenons le cas le plus fréquent d'une mesure simple dont le temps est la noire (2/4, 3/4, ou 4/4). Lorsque n croches sont regroupées, avec le chiffre n indiqué au centre du groupe, la durée de chacune n'est plus 1/2 temps mais 1/n temps. Le cas le plus courant est une division ternaire, avec $n=3$ (groupe de trois croches formant un triolet). On trouve également des groupements de doubles croches qui forment 1

temps ou 1/2 temps. La figure 1.4 montre quelques exemples.

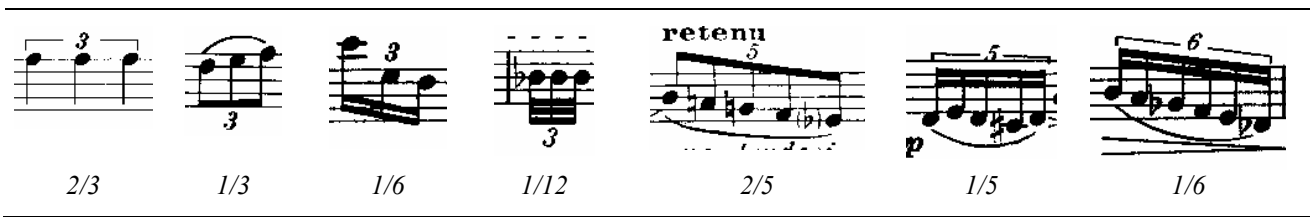


Figure 1.4 :Exemples de n-plets, avec la durée de chaque note (1 noire = 1 temps)

A l'inverse, on peut trouver dans des mesures ternaires des divisions binaires de notes pointées. Par exemple, un groupe de deux croches, avec un 2 au-dessus du groupe, est appelé duolet, et a la même durée totale qu'un groupe de trois croches.

Autres symboles

Jusqu'à présent, nous avons présenté les signes nécessaires à la production de la musique, c'est-à-dire ce que l'on joue. Il existe également des signes qui permettent d'indiquer le phrasé, c'est-à-dire comment la musique doit être jouée. Ce sont les ornements, les signes de nuance, etc. Nous ne les détaillons pas ici, car nous n'avons pas pour objectif de les reconnaître.

Musique monodique et musique polyphonique

On appelle « voix » une ligne mélodique qui correspond à un seul instrument. Dans le cas de la musique monodique, il n'y a qu'une seule voix par portée, sans aucun accord (notes jouées simultanément). Une partition d'orchestre est au contraire polyphonique puisqu'elle est constituée de plusieurs voix. Certaines d'entre elles sont strictement monodiques (les instruments à vent, Figure 1.5a), d'autres peuvent contenir des accords (les instruments à cordes, Figure 1.5b). Les partitions les plus complexes sont certainement les partitions qui présentent des portées doubles (orgue, piano, etc.), une pour chaque main, avec des accords. Dans ce cas, il est possible qu'une ligne mélodique passe d'une portée à l'autre (Figure 1.5c).



Figure 1.5 : Musique monodique ou polyphonique

Règles de notation

La théorie musicale codifie l'écriture de la musique. Les règles sont d'ordre graphique ou syntaxique. Nous indiquons ici les plus importantes.

Les règles graphiques sont relatives à la position des symboles :

1. Une altération doit être placée devant la note qu'elle altère, et à la même position verticale.
2. Un point de durée doit être placé après la tête de note.
3. Le point d'une note piquée est placé au-dessus de la tête de note.

Les règles syntaxiques sont relatives à la tonalité de la partition ainsi qu'à la métrique :

4. Le nombre de temps par mesure doit toujours correspondre à la métrique indiquée au début de la première portée.
5. Les notes sont généralement groupées en temps, en multiple de temps, ou en fraction de temps, pour faciliter la lecture rythmique. Prenons l'exemple d'une mesure simple, dont la référence de temps est la noire (2/4, 3/4, 4/4). On trouve alors des groupes de croches dont la durée totale vaut une ou plusieurs noires, ou une croche, ou même parfois une double croche (Figure 1.6). Pour un rythme ternaire, par exemple 3/8, 6/8, 12/8, les groupes forment le plus souvent un temps, donc une durée équivalente à une noire pointée (trois croches, Figure 1.7a), mais parfois des durées équivalentes à une ou deux croches (Figure 1.7b). Remarquons que des silences peuvent remplacer des notes dans le groupe (Figure 1.7c), quelle que soit la métrique.
6. Les altérations à la clé (armure) suivent un ordre prédéfini et indiquent la tonalité du morceau. Elles sont implicitement appliquées à toutes les notes du même nom (même hauteur à l'octave près) : par exemple, à tous les fa de la partition.
7. Une altération est appliquée à la note suivante, mais aussi, implicitement, à toutes les notes du même nom présentes dans le reste de la mesure.



Figure 1.6 : Exemples de découpages rythmiques usuels pour des mesures binaires



Figure 1.7 : Exemples de découpages rythmiques usuels pour des mesures ternaires

Ces règles sont néanmoins appliquées avec des degrés variables de souplesse, ou à quelques exceptions près.

Typiquement, une règle stricte est la règle 4 concernant la métrique. Elle est toujours respectée, à l'exception des anacrouses (notes qui précèdent la première mesure d'un morceau), ou des reprises. Dans ce dernier cas, c'est la somme de la mesure précédant la barre de reprise, et de la

mesure à laquelle on est renvoyé, qui doit satisfaire à la métrique.

La plupart des autres règles sont des règles souples, c'est-à-dire qu'elles sont généralement respectées, mais qu'elles peuvent aussi être relâchées, ou encore, qu'elles peuvent être appliquées de différentes façons.

Par exemple, la règle 7 indique qu'il n'est théoriquement pas utile de rappeler une altération dans une même mesure. Néanmoins, on peut trouver des altérations redondantes, qui n'apportent aucune information supplémentaire et qui donc ne devraient théoriquement pas être présentes, mais qui permettent de faciliter la lecture. La figure 1.8 donne quelques exemples.



(a) Les bémols 1 et 2 annulent les altérations à la clé, le bémol 5 annule le dièse 4 qui le précède dans la même mesure. En revanche, le dièse 6 est redondant car il est présent implicitement (à la clé) ; il facilite la lecture, car le ré était bémol dans la mesure précédente.

(b) Le bémol 2 est redondant : il succède à un si bémol, mais dans une nouvelle mesure, et il n'est pas altéré à la clé ; là encore, il permet de faciliter la lecture.

Figure 1.8 : Exemples d'altérations dont certaines sont redondantes

La règle 5, concernant les arrangements rythmiques de croches, est également une règle souple, parce que les notes peuvent être regroupées de manières différentes, chaque arrangement étant conforme à la règle (Figures 1.9a et 1.9b), et que cette règle peut être relâchée pour des questions de phrasé (Figure 1.9c).

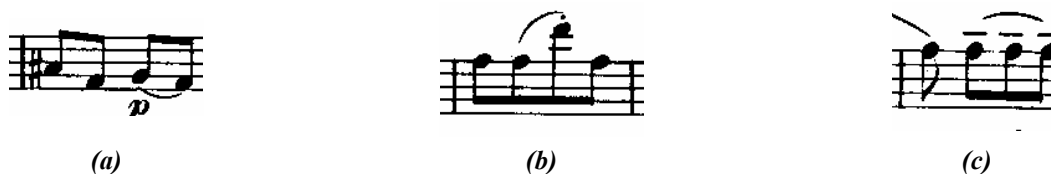


Figure 1.9 : Exemples d'arrangements rythmiquement équivalents (métrique binaire)

Enfin, les règles graphiques indiquent approximativement la position des symboles les uns par rapport aux autres. Néanmoins, on peut trouver des décalages variant d'une édition à l'autre, ou même à l'intérieur d'une même partition, suivant la densité des symboles.

Concluons ce paragraphe en remarquant que toutes ces règles qui codifient l'écriture musicale agissent à plusieurs niveaux. Certaines sont relatives à la structure des symboles (groupements de croches) ou à leurs positions relatives, d'autres sont purement syntaxiques. Certaines sont locales, d'autres mettent en cause des symboles très distants. Enfin, certaines règles expriment des contraintes binaires (entre deux symboles), d'autres des contraintes d'ordre supérieur. Toutes participent conjointement à l'interprétation. C'est dans la modélisation de ces règles et leur introduction dans le processus de reconnaissance que réside une des originalités de l'approche que nous proposons dans les chapitres suivants.

1.2. Difficultés propres à l'écriture et à l'édition musicale

La reconnaissance optique de partitions musicales est un domaine très spécifique, bien différent par exemple de la reconnaissance de caractères. On peut trouver dans [Blostein, Baird 92], ainsi que dans un grand nombre d'articles (par exemple [Ng, Boyle 96], [Bainbridge, Bell 97]) une analyse très intéressante des difficultés rencontrées.

Au premier abord, le problème semble relativement simple. En effet, l'écriture musicale met en jeu un nombre assez faible de symboles, et elle est assez bien codifiée par des règles de notation. Celles-ci peuvent être utilisées à plusieurs niveaux, pour simplement vérifier la cohérence du résultat de reconnaissance [Coüasnon, Rétif 95], pour conduire le processus de reconnaissance [Coüasnon 96b] [Stückelberg et al. 97], pour restituer le contenu sémantique à partir des primitives reconnues [Fahmy, Blostein 91], pour extraire la solution correcte parmi un ensemble de possibilités de reconnaissance [Fahmy, Blostein 98].

Les difficultés rencontrées sont en fait très importantes. La première se situe dès l'étape de segmentation. Cette étape préliminaire doit permettre de localiser et d'isoler les symboles musicaux, avant d'appliquer l'algorithme qui permettra de les reconnaître. Dans le cas de la musique, les symboles sont largement connectés entre eux par les lignes de portée, les barres de groupe de notes. Les lignes de portée interfèrent de trois manières [Prérau 70] : elles connectent des symboles qui devraient être séparés, elles camouflent le contour des symboles, elles remplissent les symboles creux. Ainsi, il est difficile de savoir si, sans portée, certains pixels seraient blancs ou noirs [Bainbridge, Bell 97].

A cette difficulté structurelle s'ajoute les difficultés liées à la qualité de l'édition originale, souvent médiocre. On remarque notamment sur un grand nombre de partitions des segments coupés, scindant ainsi certains symboles en deux, ou au contraire, des connexions parasites (Figures 1.10 et 1.11), défauts d'impression mais aussi conséquence de la densité souvent élevée des symboles [Coüasnon 96b].

Ainsi, la segmentation de l'image en entités musicales cohérentes est une étape très délicate. Généralement, elle commence par l'effacement des lignes de portée, prétraitement qui détériore les symboles. En effet, on se trouve face au paradoxe classique suivant : pour segmenter correctement les objets, il faudrait les avoir identifiés, mais pour les identifier, il faut les avoir préalablement segmentés. Cela est particulièrement vrai pour les symboles creux, comme les blanches, les bémols, tangents aux lignes de portée (voir par exemple [Martin, Bellissant 91], [Carter, Bacon 92]). L'imperfection de la segmentation a pour conséquence de générer de l'ambiguïté, c'est-à-dire que l'analyse individuelle d'un objet pourra conduire à plusieurs interprétations possibles.

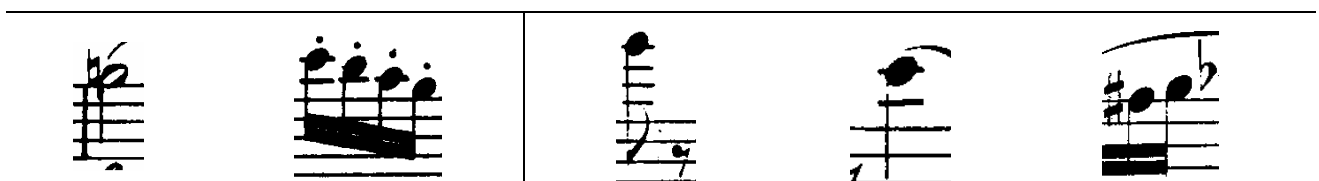


Figure 1.10 : Exemples de défauts d'impression : à gauche, connexions parasites, à droite fragmentations

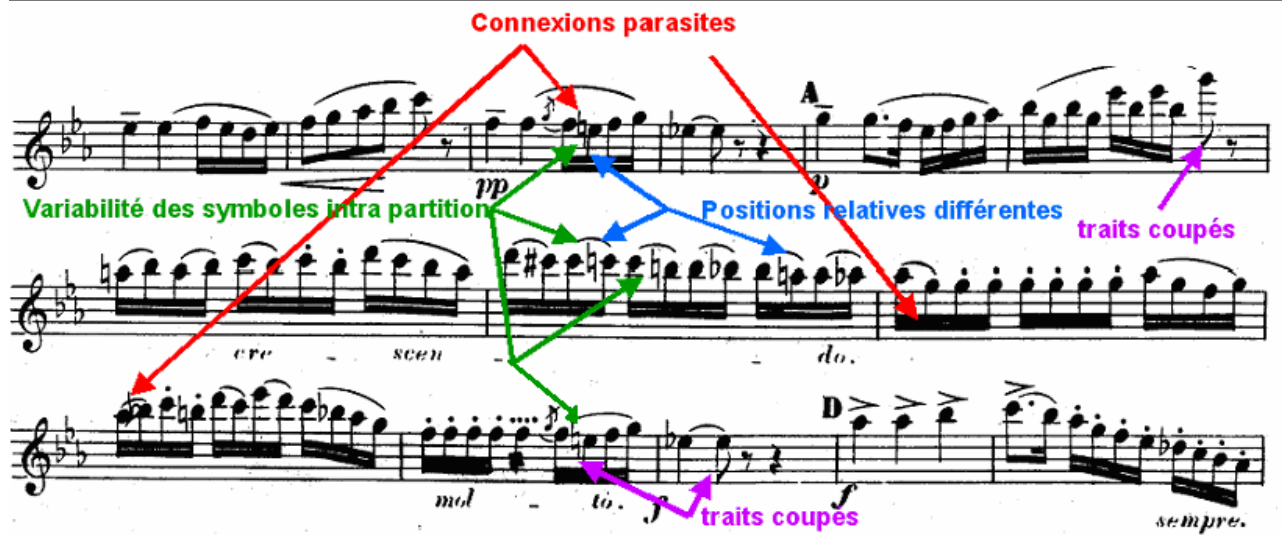


Figure 1.11 : Exemples d'imperfections dans l'édition originale

Une deuxième difficulté est due à la variabilité des formes [Fujinaga 88]. On peut trouver dans des éditions différentes des formes variées d'un même symbole (polices différentes). Les symboles peuvent même varier de manière significative à l'intérieur d'une même partition (Figures 1.11 et 1.12), notamment à cause de l'imperfection de l'impression. Il en résulte de nouveau un risque d'ambiguïté, si les modèles de classe utilisés en reconnaissance ne correspondent pas tout à fait aux symboles de la partition traitée.

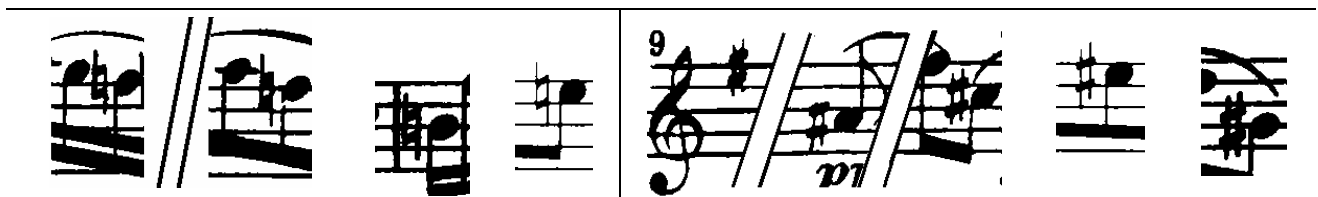


Figure 1.12 : Variabilité des symboles, inter et intra partition

La notion de symbole musical est importante et doit être précisée. Dans le précédent paragraphe, nous avons défini les symboles (Figure 1.2) comme les entités que le musicien perçoit et interprète : une blanche, une croche, une noire, une altération, un silence. Néanmoins, on trouve des définitions différentes dans la littérature relative à la reconnaissance optique de la musique. La plus simple consiste à définir un symbole comme un ensemble de pixels connexes après effacement des lignes de portée [Prerau 70]. Ainsi, un groupe de notes est un symbole. Cependant, beaucoup d'auteurs, par exemple [Mahoney 82] [Ng, Boyle 96], distinguent deux catégories de symboles : les symboles isolés et les symboles composés, appelés respectivement signes symboliques et signes iconiques ou construits [Martin 92][Coüasnon 96b]. Les signes symboliques, comme les altérations ou les silences, sont, tels les caractères, à peu près invariants en forme et taille. Les signes iconiques sont en fait constitués d'un arrangement spatial de différentes primitives. Ainsi, un groupe de 4 doubles croches est constitué de 4 têtes de note noires, 4 hampes, 2 barres de groupe. Les têtes de note sont relativement invariantes ; en revanche, les hampes et les barres de groupe sont paramétrées en taille et en orientation. Pour que le symbole composé soit bien reconnu, il faut que toutes les primitives le soient. Il est à noter que, si l'ensemble des primitives est très restreint, le

nombre d'arrangements possibles est au contraire quasi illimité.

Les systèmes de reconnaissance décrits dans la littérature n'ont pas tous défini le même ensemble de primitives. Celui-ci est lié aux objectifs fixés et à la méthode proposée. Cependant, la plupart poussent le niveau de décomposition très loin. Par exemple, [Ng, Boyle 96] [Bellini et al. 01] réalisent une segmentation récursive des objets jusqu'aux composantes les plus élémentaires : segments, arcs, têtes de note. La reconstruction est réalisée par introduction de la connaissance syntaxique et structurelle. Les difficultés rencontrées se situent au niveau de la sous-segmentation et de la prise en compte de l'ambiguïté sur la classe des primitives segmentées.

Nous avons évoqué des règles régissant la construction des symboles composés à partir de primitives. Il existe par ailleurs un certain nombre de règles de notation musicale qui expriment les interactions entre les symboles musicaux. Nous avons indiqué les principales dans la section précédente.

Nous avons montré que beaucoup de ces règles expriment des contraintes souples entre deux ou davantage de symboles, graphiquement proches ou très distants. Ainsi, une altération doit être cohérente avec les altérations à la clé, avec les autres altérations dans la mesure et éventuellement dans les mesures précédentes. En cas d'ambiguïté sur la classe des altérations (dièse, bémol ou bécarre), il faut vérifier leur cohérence mutuelle, sachant qu'il n'existe généralement pas une unique combinaison possible, à cause de la souplesse des règles musicales.

D'autre part, plusieurs règles sont généralement mises simultanément en jeu pour reconnaître et interpréter un symbole musical. Par exemple, pour retrouver l'interprétation complète d'une croche pointée, faisant partie d'un groupe de croches, il faut trouver sa hauteur et sa durée : la quasi totalité des règles mentionnées est donc susceptible de s'appliquer ! En particulier, pour valider la présence du point de durée, il faut non seulement que ce point se trouve correctement positionné près de la tête de note, sans être situé au-dessus de la tête de note suivante car ce serait plutôt un point de staccato, mais aussi que le groupe de notes auquel il appartient corresponde à un groupement usuel, et que le nombre de temps dans la mesure soit cohérent avec la métrique. On doit donc non seulement tester des règles graphiques entre primitives voisines, mais aussi prendre en considération des groupes de symboles (ici le groupe de notes reconstitué et la mesure) pour tester des règles syntaxiques. On voit donc, grâce à cet exemple, que les règles de notation se situent à des niveaux d'interprétation différents : les règles graphiques sont directement appliquées sur les primitives, alors que les règles syntaxiques se trouvent à un niveau d'interprétation supérieur. Cependant, toutes ces règles sont complémentaires et l'application d'une seule d'entre elles ne suffit pas à valider de façon certaine la présence du point de durée. Par conséquent, la fusion de toutes les informations, provenant de l'application des différentes règles, est un point crucial, mais difficile, car les informations sont de natures très différentes et se situent à des niveaux d'interprétation différents [Fahmy, Blostein 98].

Ainsi, les règles de la notation musicale sont un atout pour fiabiliser les résultats, et la nécessité de les modéliser et de les intégrer dans l'algorithme de reconnaissance est depuis longtemps reconnue [Blostein, Baird 92]. De toute évidence, cette tâche est très délicate, car il faut aller bien au-delà de l'application séquentielle de règles locales ou de contraintes binaires strictes.

Pour conclure, on peut donc dire que l'enjeu est, d'une part de détecter, segmenter et reconnaître de la manière la plus fiable possible les primitives de base, qui doivent être bien

définies, d'autre part, de modéliser au mieux la connaissance a priori, graphique et syntaxique, pour lever l'ambiguïté sur les primitives extraites et restituer l'interprétation de haut niveau. La complexité de cette tâche se situe à plusieurs niveaux :

1. L'ambiguïté est importante, à cause des défauts d'impression, de la difficulté de segmenter la partition en entités cohérentes, et de la variabilité des primitives.
2. Cette ambiguïté est difficile à lever, car, si le nombre de primitives est restreint, le nombre d'arrangements de primitives, lui, est infini.
3. Les règles de notation expriment pour la plupart des contraintes souples et non strictes, ou sont valables avec des degrés de précision variables.
4. Les règles de notation peuvent mettre en jeu un grand nombre de symboles graphiquement éloignés les uns des autres.
5. Les règles de notation sont de natures très différentes, elles se situent à tous les niveaux d'interprétation, et cependant sont interdépendantes.

1.3. Méthodes existantes

Dans cette section, nous analyserons les systèmes présentés dans la littérature. Notons que ces projets traitent la notation musicale classique (CMN : Common Music Notation). Nous dégagerons tout d'abord les différentes étapes généralement mises en œuvre, puis nous discuterons plus précisément chacune d'entre elles.

1.3.1. Stratégies générales

La plupart des systèmes sont constitués d'un ensemble de traitements séquentiels, allant des traitements de bas niveau vers l'interprétation de haut niveau. Les différentes phases sont typiquement les suivantes :

1. Détection des lignes de portée.
2. Segmentation, généralement après suppression des lignes de portée.
3. Reconnaissance des primitives segmentées, et réassemblage des symboles composés.
4. Analyse syntaxique et sémantique.

Les tâches de bas niveau sont la détection des lignes de portée et la segmentation, qui permettent de localiser les symboles musicaux à reconnaître. Certaines connaissances a priori sur l'écriture musicale sont d'ores et déjà intégrées dans les algorithmes, comme le parallélisme et l'équidistance des lignes de portée. La phase de reconnaissance permet de classer chacun des objets localisés, par exemple attribuer à un objet la classe dièse. Beaucoup de systèmes reconnaissent les groupes de notes en extrayant et en réassemblant les primitives qui les composent (têtes de note, segments), grâce à des règles structurelles portant sur la forme et la position relative des primitives (e.g. [Baumann 95] [Couasnon 96b], [Ng, Boyle 96], [Bellini et al. 01]). L'analyse syntaxique et sémantique est réalisée à un niveau d'abstraction plus élevé. Il s'agit de vérifier la cohérence entre les symboles reconnus : s'assurer par exemple que le nombre de temps dans la mesure est correct

par rapport à la métrique [Coüasnon, Rétif 95]. Il s'agit également de restituer l'interprétation de haut niveau, telle que la hauteur réelle d'une note en fonction de l'altération placée devant la tête de note ou des altérations précédentes dans la mesure [Fahmy, Blostein 91].

Toutes ces étapes sont liées entre elles et le résultat final est conditionné par la qualité du résultat produit par chacune. Ainsi, une mauvaise détection des lignes de portée conduira à une mauvaise segmentation, donc une mauvaise reconnaissance, et donc une mauvaise interprétation finale. C'est pourquoi la tendance générale a été d'introduire au maximum toute l'information a priori disponible, dans toutes les phases du processus de reconnaissance. D'après [Coüasnon 96a], le résultat est que cette information est souvent injectée de manière ponctuelle, sans formalisation précise, et qu'elle est généralement incomplète. Ce point de vue est peut-être discutable. Nous aurions plutôt tendance à penser que certaines connaissances peuvent être utilisées ponctuellement, avec une formalisation adaptée à chacune d'elles, si elles sont décorréliées des autres. Par exemple, il ne paraît pas dommageable de rechercher des têtes de note sur ou entre des lignes de portée uniquement, puisque l'on sait qu'elles ne peuvent être ailleurs. En revanche, il faut une unité de formalisation pour toutes les autres connaissances qui sont interdépendantes, typiquement les règles graphiques et syntaxiques qui mettent en jeu plusieurs symboles simultanément.

Une critique, formulée à propos de cette architecture standard, est qu'elle est unidirectionnelle : dans la plupart des méthodes proposées, les différentes étapes sont exécutées les unes après les autres, sans remise en cause des résultats obtenus. Dans ce contexte, une erreur de reconnaissance due à une mauvaise segmentation ne peut être corrigée. Seuls quelques auteurs ont tenté de faire coopérer les différentes étapes dans les deux directions : [Kato, Inokuchi 90] ont effectivement réalisé ce type d'architecture pour la reconnaissance de partitions de piano, quatre modules de traitement communiquant dans les deux sens via une mémoire à cinq couches ; [Stückelberg et al. 97] ont annoncé, de manière très prospective, une architecture en trois couches coopérantes permettant une interaction bidirectionnelle et continue entre la connaissance de haut niveau et les données de bas niveau ; McPherson et Bainbridge tentent d'améliorer les performances du système Cantor en introduisant des méthodes rétroactives, séquencées par un module spécifique [McPherson, Bainbridge 01] [McPherson 02]. La bidirectionnalité permet d'utiliser les conclusions des couches supérieures pour diriger ou revoir les tâches de bas niveau, jusqu'à l'obtention d'un résultat cohérent. D'autres méthodes n'adoptent pas de manière explicite ce type d'architecture, mais mettent en œuvre des mécanismes de remise en question, permettant notamment d'adapter la segmentation en fonction du contexte [Coüasnon 96a], ou utilisent la connaissance syntaxique et sémantique pour confirmer ou corriger les résultats de reconnaissance [Ng, Boyle 96].

Nous allons donc présenter dans la suite de ce chapitre les principales méthodes qui ont été proposées pour mettre en œuvre les différentes étapes du processus de reconnaissance. A noter qu'une revue détaillée des publications antérieures à 1991 est disponible dans [Blostein, Baird 92]. Nous consacrerons un paragraphe aux méthodologies prenant en compte l'incertitude, qui, comme nous l'avons indiqué dans la section 1.2, est importante, à cause des défauts de segmentation, de la variabilité des symboles intra et inter partitions, de l'imprécision et de la souplesse des règles d'écriture musicale. Nous terminerons par un résumé très succinct des principaux systèmes actuellement à l'étude, et par quelques remarques sur l'évaluation des résultats.

1.3.2. Détection des portées

Tous les systèmes de reconnaissance commencent par localiser les portées. En effet, elles constituent le support graphique sur lequel sont positionnés les différents symboles musicaux. Elles jouent donc un rôle central dans la lecture de la partition musicale :

- Elles définissent l'horizontalité de la partition.
- L'interligne, c'est-à-dire la distance entre deux lignes de portée, exprimée en pixels, indique l'échelle de la partition. Ce paramètre peut servir de facteur de normalisation pour les mesures de longueurs et de distances [Fujinaga 88].
- Les symboles musicaux doivent être recherchés sur les portées ou légèrement au-dessus ou au-dessous. Certains, comme les clés, ont une localisation très précise. On peut également citer les silences qui, dans le cas de la musique monodique, se situent autour de la troisième ligne de portée, et les têtes de note qui sont sur les lignes de portée ou dans les interlignes. Par conséquent, et bien que cette remarque apparaisse rarement dans la littérature, les algorithmes de segmentation et d'analyse des symboles peuvent tirer profit de ces informations a priori, en recherchant et en analysant les symboles musicaux aux endroits où ils peuvent être sur la portée, compte tenu de leur classe.
- L'interprétation sémantique des symboles musicaux tient compte de leur position sur les lignes de portée : typiquement, le nom d'une note est déduit de la position de la tête de note sur la portée. Celle-ci doit donc être connue précisément, pour toute coordonnée horizontale.
- Enfin, beaucoup de systèmes commencent la segmentation par un effacement des lignes de portée, ce qui suppose une grande précision sur la localisation et la caractérisation de celle-ci.

La détection des lignes de portée et des portées n'est pas immédiate. En effet, les lignes de portée des partitions imprimées ne sont pas parfaites : d'après [Prerau 70], elles ne sont pas exactement parallèles, horizontales, équidistantes, d'épaisseur constante, ni même droites. Par ailleurs, la principale difficulté rencontrée pour leur localisation précise est due à la présence des symboles musicaux qui interfèrent avec elles, surtout ceux qui ont une orientation horizontale, comme certaines barres de groupe [Carter 89]. Nous résumons donc dans cette section les principales méthodes proposées, en indiquant dans quelle mesure elles permettent de surmonter ces difficultés.

Trois grandes catégories de méthodes peuvent être distinguées dans la littérature : les méthodes qui permettent de calculer les paramètres des portées (interligne et épaisseur des lignes) avant même leur localisation, les méthodes plus ou moins sophistiquées utilisant les projections, les méthodes qui modélisent les lignes horizontales par une agglutination de colonnes de pixels noirs, appelés empan.

Détection préalable de l'interligne et de l'épaisseur des lignes

La distance séparant deux lignes de portée peut être déterminée avant la localisation même des portées. Ainsi, on peut calculer l'histogramme de la longueur des segments verticaux blancs et

des segments verticaux noirs, de largeur 1 pixel, appelés aussi « empan » en français, et « run-lengths » en anglais. Le maximum du premier histogramme donne l'interligne, alors que le maximum du second indique l'épaisseur des lignes de portée. Kato et Inokuchi ([Kato, Inokuchi 90], [Kato, Inokuchi 92]) réalisent cet histogramme sur 10 colonnes régulièrement espacées sur la largeur de l'image. Cette méthode a été largement reprise, notamment par [Bellini et al 01] et [Miyao 02]. [Bellini et al 01] calculent cependant les histogrammes sur toute l'image, et déterminent de plus les intervalles de variation d'après l'épaisseur des pics des histogrammes. Ces résultats préliminaires sont par la suite utilisés pour analyser des projections horizontales de l'image et pour paramétrer le processus de segmentation.

Méthodes fondées sur les projections horizontales

Si l'on considère que les lignes de portée sont rectilignes, et à peu près horizontales, alors cette méthode est extrêmement simple. Elle consiste à calculer la somme des pixels de chaque ligne. Le tableau obtenu, appelé profil vertical, met en évidence des groupes de cinq pics équidistants correspondant aux lignes de portée. La méthode est appliquée telle quelle par [Fujinaga 88] et [Sicard 92] sur toute l'image. Cependant, Carter [Carter, Bacon 92] et Blostein [Blostein, Baird 92] soulignent qu'une faible inclinaison, d'un demi-interligne, la rend inefficace, car les pics fusionnent.

Pour pallier le problème du biais, plusieurs stratégies ont été proposées. Nous exposons les principales.

[Baumann, Dengel 92] réalisent les projections en découpant l'image en zones de faible largeur, sur lesquelles les lignes peuvent être considérées comme pratiquement horizontales. La méthode paraît cependant sensible aux interférences entre portée et symboles, surtout dans des zones denses comprenant beaucoup d'objets superposés ou tangents aux lignes de portée, telles les barres de groupe. Elle n'est en outre pas très précise.

[Kato, Inokuchi 92] commencent par localiser grossièrement les portées, en utilisant les paramètres (interligne et épaisseur des lignes) déduits des histogrammes des empan noirs et blancs. Ils effacent ensuite les petits segments horizontaux, sur de petites sections proches des bords droit et gauche de la portée, avant de projeter localement dans les deux directions : ils trouvent ainsi la position précise des lignes de portée, au début et à la fin de celle-ci. Les lignes de portée sont approximativement les droites passant par ces points extrêmes. [Ramel et al. 94] projettent également aux extrémités droite et gauche de l'image (zones de largeur égale à 1/5 de la largeur totale) pour localiser les lignes de portée et les portées. Dans leur méthode, l'interligne et l'épaisseur des lignes n'ont pas été préalablement calculés, et ils sont estimés par l'espacement et l'épaisseur des pics de l'histogramme. Enfin, [Martin 89] teste différentes rotations de l'image, avant de réaliser la projection. Le biais de l'image correspond à la rotation qui maximise les pics de la projection. Cette méthode est très lourde en calculs, puisqu'il faut effectuer un grand nombre de rotations, alors que le plus judicieux serait de trouver l'angle de rotation avant de réaliser la rotation appropriée. C'est pourquoi l'auteur [Martin 92] retient finalement une méthode fondée sur la maximisation des longueurs des cordes. Une corde est définie ainsi : un segment passant par un point P d'une composante 8-connexe C , de pente θ , inclus dans C . La longueur de la corde, c'est-à-dire la distance entre les deux points extrêmes situés sur la frontière de C , est maximale lorsque celle-ci se trouve sur une ligne de portée. Ainsi, en testant plusieurs angles θ , on peut détecter le biais, le corriger, et ensuite seulement

projeter pour localiser les portées. Le procédé est plus rapide. Notons par ailleurs qu'une méthode comparable avait été proposée par [Roach, Tatem 88], non seulement pour calculer l'angle d'inclinaison, mais aussi pour identifier les portées.

Cependant, toutes ces propositions supposent toujours que les lignes de portée sont parfaitement rectilignes, et elles ne permettent pas de gérer des courbures éventuelles, ni de connaître précisément la position des lignes de portée en chaque coordonnée horizontale. Deux stratégies ont été proposées pour résoudre le problème : appliquer un algorithme de détection très local [Bellini et al. 01] [Bainbridge, Bell 97], ou rechercher préalablement les portions de portée sans symboles avant de reconstituer les lignes complètes [Randiamahefa et al. 93].

[Bellini et al. 01] réalisent des projections sur des fenêtres très étroites (quelques pixels de large), parcourant toute la hauteur de l'image. Les pics correspondant aux lignes de portée sont validés en vérifiant que leur espacement et leur épaisseur sont cohérents avec les paramètres trouvés par l'analyse préalable des histogrammes des longueurs des emfans [Kato, Inokuchi 90]. La localisation est donc précise, au moins sur les portions sans symboles. Le problème de l'occultation locale, partielle ou totale, par des symboles n'est pas évoqué, bien que cette méthode semble très sensible au bruit interférent, à cause de son caractère très local.

[Bainbridge, Bell 97] mettent en œuvre un algorithme permettant d'affiner la localisation des lignes de portée, obtenue par projection. Les emfans, appelés "slithers", sont recherchés dans une zone contrainte par la position du "slither" précédent. La présence d'un objet sur la ligne de portée est détectée sur un critère de longueur ("slither" plus long), et dans ce cas la position du "slither" précédent est conservée. La méthode ne semble pas non plus très robuste en cas de forte densité de symboles, surtout si ceux-ci sont tangents aux lignes de portée, parce que la fenêtre est réduite à un unique pixel et que chaque ligne semble être poursuivie indépendamment des autres.

Une solution différente est proposée par [Randiamahefa et al. 93]. Cette fois, les auteurs détectent préalablement les portions de portée sans symboles, par projection verticale et recherche des minima locaux. Ensuite ils projettent ces régions horizontalement, et trouvent ainsi un ensemble de pics qui peuvent correspondre aux lignes de portée. Pour valider les points obtenus et les relier entre eux, ils recherchent la droite qui passe au plus près, avec un seuil d'acceptation peu sévère pour tolérer la courbure. Mais on peut se demander si l'algorithme est robuste en cas de forte densité des symboles, car alors, le nombre de points révélant la portée est très réduit et peut-être insuffisant.

Analyse des emfans noirs pour la détection et le suivi des lignes de portée

Trois méthodes ont traité ce problème sans faire appel aux projections, essayant de faire face à tous les défauts possibles, de manière à obtenir une détection robuste et une localisation précise des lignes de portée. Dans la première [Miyao 02] [Reed, Parker 96], la présence de points régulièrement espacés dans la direction verticale révèle les points de passage des lignes de portée, qui sont ensuite approchées par les segments reliant horizontalement ces points. Les deux autres méthodes, bien que très différentes, procèdent toutes deux par agglutination d'emfans noirs pour former des segments. Il s'agit de la méthode proposée par [Carter 89], présentée également dans [Carter, Bacon 92] et [Blostein, Baird 92], fondée sur le graphe des lignes adjacentes, et du détecteur de segments par filtre de Kalman, proposé par [Poulain d'Andecy et al. 94]. Nous présentons chacun de ces axes dans la suite de ce paragraphe.

Miyao [Miyao 02] commence par calculer l'épaisseur des lignes de portée et l'interligne suivant [Kato, Inokuchi 90]. La partition est ensuite divisée en zones de largeurs égales par 35 lignes verticales. Sur chacune des lignes, les emfans de longueur comparable à l'épaisseur des lignes de portée sont détectés. On obtient ainsi sur chaque ligne verticale une série de points, dits candidats, situés au centre des emfans retenus, correspondant potentiellement à l'intersection avec une ligne de portée. La mise en correspondance de deux séries consécutives de points candidats permet de tracer des segments horizontaux les reliant, qui sont donc potentiellement sur une ligne de portée. Les critères utilisés portent sur l'inclinaison tolérée du segment, et la proportion de pixels noirs sur le segment. Les lignes de portée sont déduites des segments obtenus, sachant qu'ils doivent être séparés verticalement d'un interligne. Les points candidats erronés (par exemple dus aux petites lignes au-dessus de la portée) sont supprimés, et les points manquants (occultés par des symboles) sont obtenus par interpolation. Le résultat produit consiste en cinq lignes dites "polygonales" définies par des points régulièrement espacés. Les seuils utilisés dans l'algorithme permettent de faire face à des inclinaisons (jusqu'à 5°), de faibles courbures, et des discontinuités. Dans la méthode [Reed, Parker 96], l'épaisseur des lignes de portée n'est pas préalablement déterminée, et le critère de sélection des emfans candidats porte sur la présence de cinq emfans consécutifs de longueur comparable et régulièrement espacés, appelés échantillons. Les portées sont extraites grâce à des critères de similarité (espacement et longueur des emfans) et d'inclinaison des segments reliant les échantillons voisins. Ces méthodes s'apparentent à celle de [Randiamahefa et al. 93], dans le sens où elles extraient des points potentiels de passage des lignes de portée, sur des plages non occultées, et valident ceux qui satisfont à des critères d'alignement, avec des paramètres autorisant une courbure. Miyao note qu'il peut y avoir échec en cas de forte densité de notes, ou de présence de barres de groupe sur la portée, parce que le nombre insuffisant de points candidats trouvés conduit à une prédiction erronée des points manquants.

L'objectif de Carter [Carter 89] est de trouver un moyen de détecter les lignes de portée, tolérant de petites rotations (jusqu'à 10°), de faibles courbures et des variations locales d'épaisseur des lignes de portée. Il s'agit également de traiter correctement la segmentation des symboles tangents aux lignes de portée, mais nous reviendrons sur ce point dans le paragraphe concerné. Carter construit un graphe des lignes adjacentes (LAG), de la manière suivante : les emfans verticaux (appelés segments) sont détectés lors d'une première passe. Dans une seconde passe, les emfans connexes d'épaisseur comparable sont agglomérés pour former des sections. Ces sections se terminent par des jonctions : une jonction se produit lorsqu'un emfan est connexe à plusieurs autres emfans de la colonne voisine, ou qu'il y a une forte variation d'épaisseur entre l'emfan et son voisin. Les sections constituent les nœuds du graphe des lignes adjacentes, les jonctions sont les liens. Grâce au critère d'épaisseur, les portions de portée sans symboles et les symboles eux-mêmes forment des sections différentes dans le graphe. Des critères structurels (rapport épaisseur/longueur, courbure) permettent de chercher les sections qui peuvent correspondre à des portions de lignes de portée. Celles-ci sont appelées filaments. Les filaments colinéaires sont concaténés pour former des chaînes et, finalement, une portée est détectée lorsque cinq chaînes de filaments sont à peu près équidistantes et se chevauchent. Les lignes de portée sont donc trouvées, dans un premier temps, comme une liste de fragments de portée, précisément localisés malgré les défauts potentiels. Les fragments manquants peuvent être déduits par interpolation. Par ailleurs, on voit que cette méthode amorce la segmentation, puisque les sections peuvent être étiquetées ligne de portée ou non ligne de

portée, et que les jonctions entre lignes de portée et symboles sont bien identifiées. Le procédé a été inclus dans le système proposé par [Ferrand et al. 99]

L'objectif de la méthode développée par [Poulain d'Andecy et al. 94] est également plus vaste que la simple détection des portées. Partant de la constatation que beaucoup de symboles ont une structure linéaire, les auteurs ont réalisé un détecteur robuste de segments, pouvant tolérer des courbures, des variations d'épaisseur, de brèves ruptures et la superposition de symboles interférents. A partir d'un empan initial, ils appliquent un filtre de Kalman pour tenter de suivre son évolution. Pour détecter une ligne globalement horizontale, on part donc d'un empan vertical et on le poursuit de colonne en colonne, par prédiction de la position suivante (coordonnée verticale du point central de l'empan) en fonction des positions précédentes, puis appariement de l'empan prédit à l'empan réel. Le modèle théorique sous-jacent pour la détection des lignes de portée est celui d'une droite horizontale. L'intérêt du filtre de Kalman est qu'il permet de tolérer des erreurs par rapport au modèle théorique, parce qu'il fournit des indications pour choisir l'observation à associer à la prédiction, et qu'il s'adapte en fonction de l'erreur commise entre la prédiction et l'observation. Ainsi les segments peuvent être détectés en dépit des défauts (épaisseur non constante et variations de position). D'autre part, lorsque l'appariement ne peut être réalisé, typiquement si l'on rencontre un symbole sur la portée, alors l'algorithme passe à la prédiction suivante, sans réajuster le filtre, jusqu'à ce qu'on retrouve une observation compatible avec la prédiction. Ainsi, la méthode peut également faire face aux interférences dues aux symboles musicaux. Une fois les segments horizontaux trouvés, les auteurs utilisent des critères structurels pour leur classification. Les lignes de portée sont les segments qui satisfont à des critères de longueur, d'épaisseur et d'équidistance. Le taux de réussite est parfait sur la douzaine de partitions testées.

Conclusion

La localisation des lignes de portée et le calcul de ses paramètres (épaisseur des lignes et interligne) a donc fait l'objet de nombreuses recherches. L'histogramme de la longueur des empan blancs et noirs semble donner de très bons résultats pour le calcul de l'interligne et de l'épaisseur des lignes [Kato, Inokuchi 90]. L'approximation des lignes de portée par des droites semble permettre de les détecter par des projections [Fujinaga 88] [Sicard 92], avec quelques adaptations pour prendre en compte le biais [Bauman, Dengel 92] [Kato Inokuchi 92] [Martin 92]. Mais le résultat est trop approximatif pour la suite de l'analyse si bien qu'il faut mettre en œuvre des algorithmes de suivi de portée [Bainbridge, Bell 97], ou réaliser des projections très locales [Randiamahefa et al. 93] [Bellini et al. 01], pour obtenir une localisation précise, prenant en compte les courbures. Cependant, les solutions proposées jusqu'à présent ne semblent pas très robustes aux symboles interférents. Cette remarque peut être également formulée pour toutes les méthodes qui, avec ou sans projections, s'appuient sur la détection préalable de portions sans symboles [Randiamahefa et al. 93] [Reed, Parker 96] [Miyao 02]. Les méthodes fondées sur le graphe des lignes adjacentes [Carter 89] ou le filtrage de Kalman [Poulain d'Andecy et al. 94] s'affranchissent de toute projection et semblent donner de bons résultats, quels que soient les défauts des lignes de portée. Le filtrage de Kalman semble en outre approprié pour faire face aux interférences dues aux symboles superposés sur la portée.

1.3.3. Segmentation

La segmentation de l'image en entités musicales est une étape primordiale et déterminante de la qualité de la reconnaissance. Malheureusement, elle est très difficile à réaliser avec précision dans le cas des partitions, à cause de trois particularités de la notation et de l'édition musicale :

- Les symboles sont tous interconnectés par les lignes de portée, voire d'autres inscriptions comme les signes de phrasé, les liaisons.
- Les groupes de notes sont composés de primitives, qui sont par construction interconnectées. Ils se présentent sous des formes, des dimensions, des orientations très variables suivant l'arrangement réalisé, la densité de la partition, le type d'édition, etc.
- La qualité du document original est imparfaite, et on peut constater, même dans des éditions récentes, de nombreuses connexions parasites entre symboles voisins syntaxiquement séparés ou au contraire des fragmentations. De telles imperfections ont été illustrées dans les figures 1.10 et 1.11.

Une première idée, communément adoptée dans la littérature à quelques rares exceptions près [Matsushima et al. 85] [Bellini et al. 01], est d'effacer les lignes de portée. Ce premier traitement force la déconnexion de nombreux symboles musicaux, sans cependant résoudre les cas de connexions parasites. Les groupes de notes sont isolés, mais leurs composantes ne sont toujours pas localisées. Nous allons donc dans la suite de ce paragraphe décrire les processus d'effacement explicites dans la littérature. Puis nous détaillerons les méthodes qui complètent la segmentation en localisant les différentes primitives formant les symboles composés.

Suppression des lignes de portée

Les premiers systèmes [Pruslin 66][Prerau 70] (revus dans [Kassler 72]) réalisent la suppression des lignes de portée. Pruslin élimine tous les segments fins horizontaux ou verticaux, par érosion systématique, ce qui a pour conséquence de séparer les primitives (par exemple les têtes de note des hampes), mais détériore les objets musicaux, rendant leur classification difficile voire infaisable. Prerau efface les lignes de portée sur toute leur longueur, sur une épaisseur constante, fragmentant de nombreux symboles ; ceux-ci sont ensuite réassemblés, si la distance qui les sépare est égale à l'épaisseur d'une ligne de portée et s'ils se chevauchent horizontalement. Certains symboles restent néanmoins déconnectés, comme les clés de fa. Quelques années plus tard, Mahoney ([Mahoney 82] revu dans [Blostein, Baird 92]) introduit l'effacement de portions de portée sans symboles : les lignes fines sont extraites, et des descripteurs sont utilisés pour vérifier qu'il s'agit de lignes de portée pouvant être supprimées. Le procédé est également appliqué dans la direction verticale sur les fins segments verticaux. Ainsi les têtes de note sont séparées des hampes. Mahoney note par ailleurs que la suppression complète des lignes de portée permet de déconnecter des primitives adjacentes, typiquement les têtes de note d'accords. Il s'oriente donc vers un système qui considère les groupes de notes comme des arrangements de primitives, dont certaines sont paramétrées en taille et orientation (les lignes formant les hampes et les barres de groupe), et d'autres invariantes (les têtes de note). Cette vision est d'ailleurs commune à tous les systèmes d'OMR, de manière plus ou moins explicite. Dans le système de Mahoney, c'est l'effacement des lignes détectées (hampes, lignes de portée, barres de mesure) qui conduit pas à pas à la

segmentation. Cette méthodologie sera également largement appliquée par d'autres auteurs.

Depuis, la tendance est effectivement de supprimer les portions de portée sans symboles. Les lignes de portée sont poursuivies, et les empan verticaux sont effacés si leur longueur est inférieure à un seuil déduit de l'épaisseur des lignes de portée (par exemple [Reed, Parker 96] [Randriamahefa et al. 93] [Kato, Inokuchi 92] [Ng, Boyle 92]). Clarke, quant à lui, considère les configurations de pixels au voisinage immédiat de chaque empan, pour décider de son effacement [Clarke et al. 88] [Bainbridge, Bell 96]. Dans les deux cas cependant, les symboles creux sont dégradés car les fines portions tangentes aux lignes de portée sont effacées, conduisant dans le pire des cas à des fragmentations. Certains auteurs ont tenté de limiter ces distorsions en évaluant un voisinage plus large de manière à mieux différencier les symboles des lignes de portée [Martin, Bellissant 91] [Bainbridge, Bell 97]. Ces procédures sont néanmoins complexes et introduisent un surcoût de calcul important, dont on peut douter finalement de la pertinence. En effet, les auteurs soulignent que tous les problèmes ne sont pas résolus. Par ailleurs, puisque les fragmentations et autres défauts peuvent être présents dans les documents originaux, ils doivent de toute manière être pris en compte dans les étapes ultérieures de reconnaissance.

Extraction des primitives par reconnaissance puis effacement

Un grand nombre d'auteurs ne vont pas au-delà de l'effacement des lignes de portée, et finalement appliquent directement des algorithmes de reconnaissance pour extraire les objets. [Sayeed Choudhury et al. 01], par exemple, considèrent que barres de mesure, têtes de note noires et hampes peuvent être directement identifiées par des techniques simples (Run Length Encoding et analyse de connexité [Fujinaga 97]). Les primitives reconnues sont ensuite progressivement effacées dans l'image, ce qui provoque la segmentation des primitives restantes, et facilite leur reconnaissance. L'ordre d'extraction diffère d'un auteur à l'autre. [Ramel et al. 94] commencent par les têtes de notes, poursuivent par les hampes et les barres de groupe, et terminent par tous les autres symboles (de type caractère). [Sicard 92] suit un ordre différent : les barres de groupe, les hampes et les barres de mesure, puis les têtes de notes. [Poulain d'Andecy et al. 95] utilisent leur détecteur robuste de segments, et extraient tout d'abord les segments de tendance horizontale (dont les barres de groupe), puis les segments verticaux (dont les hampes et les barres de mesure), puis les têtes de notes, vues comme des segments courts très épais. Là encore, les segments qui peuvent être étiquetés sur la base de critères structurels sont effacés.

L'inconvénient majeur de ces méthodes est que la reconnaissance dépend progressivement des primitives déjà extraites : à cause de l'effacement dans l'image, ou parce que la recherche de nouvelles primitives est guidée par les objets déjà étiquetés (par exemple recherche d'une tête de note au voisinage d'un segment vertical). Une erreur sur une primitive peut donc provoquer une cascade de nouvelles erreurs.

D'autres auteurs ont essayé de mieux gérer l'incertitude. Dans le système proposé par [Kato, Inokuchi 92], les primitives, extraites par le module de plus bas niveau ("Primitive Extraction Module"), sont effacées dans l'image, mais elles devront être confirmées par les modules de plus haut niveau qui vérifient leur cohérence mutuelle. Si cette cohérence n'est pas acquise, les traitements de bas niveau sont revus. Dans le système Cantor [Bainbridge, Bell 96], les primitives

sont décrites dans un langage de haut niveau (Primela), avec les procédures qui permettent leur extraction ; mais elles ne sont effacées dans l'image que si le degré de confiance fourni par l'algorithme de reconnaissance est supérieur à un seuil. Sinon, de nouveaux tests peuvent être pratiqués, pouvant conduire à des primitives de degré de confiance supérieur.

Sous-segmentation en primitives par projections

Les projections sont largement utilisées, pour séparer les symboles (projection verticale, [Marinai, Nesi 99]), entourer les symboles d'une boîte englobante (projections dans les deux directions, [Fujinaga 88]). [Bellini et al. 01] [Fujinaga et al. 92] [Ng, Boyle 96] proposent d'aller plus loin dans leur utilisation, en calculant des profils locaux qui permettent de déterminer les points de séparation des primitives composant les groupes de notes.

Bellini commence par distinguer les groupes de notes des symboles isolés. Des projections locales, dans les deux directions, permettent d'isoler les notes puis de localiser les têtes et les barres de groupe. Des projections sur l'axe horizontal sont également appliquées sur des objets larges, composés en fait de symboles syntaxiquement isolés mais improprement connectés dans l'image (par exemple, dièses de l'armure qui se touchent). Les auteurs appliquent sur les profils des traitements (suppression de la contribution des lignes de portée, filtrages passe-haut ou passe-bas), qui permettent de les rendre plus lisibles et plus facilement interprétables [Bellini et al. 01].

Ng effectue une segmentation récursive des objets jusqu'aux primitives élémentaires, par une succession de cycles réalisant subdivision puis reconnaissance [Ng, Boyle 96]. Les points de séparation sont déduits de la dérivée seconde du profil [Ng, Boyle 92]. Les critères d'arrêt de la sous-segmentation sont : la reconnaissance de l'objet, des dimensions inférieures à un seuil, une densité de pixels noirs dans le cadre englobant trop importante [Ng, Boyle 96]. Dans tous les autres cas, une nouvelle subdivision est opérée.

Les projections ont aussi été largement utilisées par Fujinaga pour la décomposition récursive des objets en primitives [Fujinaga et al. 92]. Quelques années plus tard, Fujinaga explique que la segmentation des groupes de notes, des accords, ou des objets connectés, ne peut être réalisée avant la classification. Les projections sont donc appliquées sur des objets qui ont été identifiés comme objets composés par le module de reconnaissance, et dont on doit poursuivre la segmentation [Fujinaga 97]. La segmentation initiale est fondée sur une analyse de connexité, optimisée par un codage RLE (Run Length Encoding) de l'image.

Une première remarque peut être formulée à propos de ces méthodes : on constate de nouveau que segmentation et reconnaissance sont imbriquées, ce qui complique la gestion de l'ambiguïté. Des corrections sont opérées par des méthodes ad hoc, lorsqu'une incohérence est détectée [Ng, Boyle 96], ce qui conduit à une méthodologie où la connaissance a priori est injectée ponctuellement un peu partout dans le programme, et à des frontières mal définies entre les différentes phases de reconnaissance, des traitements bas niveau à l'interprétation sémantique. La seconde remarque est que les projections sur l'axe horizontal semblent facilement interprétables, à la différence des projections dans l'autre direction. Enfin, on peut se demander comment se comportent ces méthodes sur des images très denses ou bruitées, et si elles pourraient être étendues à des partitions très complexes.

Méthode complète de segmentation [Carter, Bacon 92]

La méthodologie de Carter, fondée sur le graphe des lignes adjacentes (LAG), aboutit à une segmentation complète qui semble limiter la fragmentation des symboles. Nous avons explicité dans le paragraphe 1.3.2 comment la construction du graphe et l'introduction de critères structurels conduisent à la détection des segments correspondant aux portions de portée sans symboles. Celles-ci sont clairement différenciées des autres objets (sections étiquetées "portée" dans les nœuds du graphe), et les connexions avec ces autres objets sont bien déterminées (arcs du graphe). Les nœuds restants sont donc des sections correspondant à des objets ou à des portions d'objets. Les sections connexes sont recombinaées pour former des symboles complets ou des composantes de symboles, et les points de jonctions sont marqués dans le graphe. On obtient donc une segmentation complète de la partition.

Problèmes des objets qui se touchent et des fragmentations

Nous avons précédemment indiqué que certains auteurs limitent les fragmentations dues à l'effacement des lignes de portée [Carter, Bacon 92] [Martin, Bellissant 91] [Bainbridge, Bell 97]. Cependant, ces défauts, ainsi que les connexions parasites, sont présents dans les documents originaux et posent des problèmes majeurs. Voyons maintenant comment ils ont été abordés dans la littérature.

On peut citer quatre auteurs. Tout d'abord, [Bellini et al. 01] et [Ng, Boyle 92], qui, grâce aux projections, peuvent séparer des symboles distincts improprement connectés. Dans [Ng, Boyle 92], une méthode d'extraction des longues primitives fines et horizontales (barres de groupe, liaisons de phrasé) est aussi proposée, en complément des projections : l'effacement "intelligent" de ces objets permet de séparer les notes groupées, mais aussi les objets connectés par des liaisons.

Dans le système Cantor décrit par Bainbridge, ce sont les informations a priori sur la forme de la primitive recherchée (modélisées dans la procédure d'identification), conjointement avec les informations image (formes isolées), qui conduisent à définir les régions où l'on doit rechercher la primitive [Bainbridge, Carter 97]. Cette zone est définie par rapport à la boîte englobante de l'objet, mais étendue en fonction de la classe testée [Bainbridge, Bell 97]. Les auteurs font ainsi face au problème de fragmentation. L'analyse de l'image source est jugée plus appropriée pour les symboles souvent endommagés par l'effacement des lignes de portée. Enfin, ils autorisent plusieurs résultats de reconnaissance sur chaque zone analysée, ce qui cette fois pallie le problème des objets connectés.

Coüasnon propose un système d'OMR entièrement contrôlé par une grammaire [Coüasnon, Camillerapp 94] [Coüasnon 96a] [Coüasnon 96b]. Le contexte, modélisé par cette grammaire, est utilisé pour tenter une segmentation adaptée du symbole recherché, c'est-à-dire tenant compte de ses caractéristiques géométriques : par exemple, pour tenter d'extraire une altération devant une tête de note déjà reconnue, si le pattern correspondant n'a pas pu être identifié jusque-là. Cela permet de régler des cas de sur ou sous-segmentation d'un signe symbolique. D'autre part, l'ordre d'évaluation des règles force la reconnaissance préliminaire des segments et des têtes de note, avant de progresser plus avant. Comme ces primitives sont effacées, des connexions parasites sont

supprimées. Cela suppose néanmoins des détecteurs de segments [Poulain d'Andecy et al. 94] et de têtes de note très robustes.

Conclusion

Il ressort de cette étude bibliographique que la segmentation de l'image en entités musicales est effectivement une difficulté majeure. La suppression des lignes de portée est communément adoptée (sauf [Bellini et al. 01] dans les articles récents), mais elle ne suffit pas, puisque les primitives composant les groupes de notes ne sont pas extraites, et que les problèmes de fragmentation ou de connexions parasites ne sont pas résolus.

Après cette étape, rares sont les systèmes qui, à l'instar de [Carter, Bacon 92], réalisent une segmentation complète, sans impliquer des procédures de classification : dans certains cas la segmentation est forcée par l'effacement de primitives reconnues (typiquement [Ramel et al. 94], [Sicard 92]), dans d'autres cas les objets sont subdivisés jusqu'à identification (typiquement [Ng, Boyle 96], ou encore [Armand 93]). En conséquence, la reconnaissance de certaines primitives dépend des extractions déjà réalisées, et l'ambiguïté, liée à la segmentation et à la classification, ne peut être vraiment prise en compte. Beaucoup de systèmes introduisent donc ponctuellement de la connaissance a priori pour vérifier, voire réviser certains résultats, mais on voit bien que toute l'information contextuelle ne peut être utilisée. L'un des objectifs de Coüasnon est de remédier à cet absence de formalisme, en proposant un système fondé sur une grammaire qui contrôle toutes les étapes, dont la segmentation [Coüasnon 96a], mais les critères utilisés restent locaux. Une autre démarche intéressante est celle de [Kato, Inokuchi 92], qui permet de remettre en cause des résultats de segmentation et de classification sur la base des interprétations de plus haut niveau. Notons enfin la grande part faite aux projections dans la segmentation [Ng, Boyle 92] [Bellini et al. 01] [Fujinaga 88] [Fujinaga et al. 92].

On constate finalement que la localisation précise de toutes les primitives, préalablement à la phase de reconnaissance, semble relever de la gageure, compte tenu de la nature de la notation et des imperfections de l'édition et de l'impression. Il semble néanmoins souhaitable d'éviter d'imbriquer segmentation et reconnaissance, afin de mieux gérer l'ambiguïté, ou d'éviter de mettre en place des architectures lourdes et complexes. Dans tous les cas, il apparaît que l'étape de reconnaissance devra au mieux gérer l'incertitude liée aux défauts de l'image et aux imprécisions de la segmentation.

1.3.4. Méthodes d'analyse des symboles

Les méthodes de classification des symboles, ou des primitives les composant, sont extrêmement variées, et un bon nombre de techniques courantes en analyse d'image sont représentées dans la littérature relative à l'OMR. Beaucoup de systèmes adoptent d'ailleurs plusieurs d'entre elles, le choix dépendant de la classe à reconnaître. Deux aspects semblent importants : tout d'abord les méthodes d'analyse et les règles de décision, mais aussi la manière dont elles sont intégrées dans le système de reconnaissance. Ce second point se rapporte entre autres à l'ordonnancement des tâches, leur imbrication avec la segmentation, l'incorporation de

connaissances a priori pour diriger la reconnaissance. Nous commenterons ces différents aspects dans la suite de ce paragraphe.

1.3.4.1. Méthodes d'analyse

Nous allons tenter de recenser les principales méthodes, en essayant de dégager leurs points forts et leurs points faibles dans le contexte de l'OMR.

Extraction des primitives linéaires

Les structures linéaires, très présentes dans la notation musicale, font souvent l'objet d'une analyse spécifique :

- Projection dans la direction verticale, pour la détection des barres de mesure et des hampes [Kato, Inokuchi 92] [Baumann, Dengel 92] [Sicard 92] [Miyao, Nakano 95].
- Filtre de Kalman pour la détection de tous les segments [Poulain d'Andecy et al. 95] [Coüasnon 96b].
- Analyse de connexité : extraire des pixels connexes qui forment des segments fins verticaux [Hori et al. 99] [Sayeed Choudhury et al. 01] [Genfang, Shunren 03] pour la détection des barres de mesure et des hampes ; extraire les pixels connexes qui forment des structures horizontales, comme les barres de groupe ou les liaisons [Sicard 92] [Ng, Boyle 92].
- LAG (Graphe des Lignes Adjacentes) : cette représentation se prête naturellement à la reconnaissance des segments. [Carter, Bacon 92] extraient les hampes des groupes de notes dans un graphe réalisé à partir des empan horizontaux, en considérant les sections correspondant à des segments fins verticaux colinéaires. [Reed, Parker 96] reprennent l'idée en construisant cette fois deux graphes par symbole (au sens objets obtenus après effacement des lignes de portée), l'un suivant les lignes, l'autre suivant les colonnes. Ils obtiennent de bons résultats pour la détection des barres de mesure, des hampes et des barres de groupe.
- Détection des segments verticaux ou horizontaux par morphologie mathématique [Armand 93].

Des critères de taille, de position et d'agencement conduisent à l'étiquetage final des segments.

Extraction de caractéristiques géométriques

Les premiers systèmes ont tenté d'extraire des caractéristiques géométriques ou structurelles simples pour la reconnaissance des objets segmentés. Derrière cette approche, il y avait naturellement l'obligation de minimiser la complexité des algorithmes, afin qu'ils soient compatibles avec la puissance de calcul des ordinateurs de l'époque. Il a été très vite remarqué que les dimensions des objets, représentées par la hauteur et la largeur de la boîte englobante, sont des paramètres discriminants. Le premier système [Prerau 70] les utilise en préclassification. Des tests heuristiques, portant sur la position du symbole, la syntaxe et d'autres caractéristiques spécifiques, permettent de compléter la classification en départageant les classes sélectionnées (3 à 5 par objet typiquement).

De nombreux systèmes fondent leur méthode de reconnaissance sur la notion de boîte englobante. D'autres attributs discriminants complètent les informations de dimension, afin de

mieux caractériser la géométrie et la structure des symboles :

- [Coüasnon 96b] : les critères de classification, jugés insuffisants par l'auteur, sont la taille des boîtes englobantes, la densité des pixels noirs, la répartition des pixels noirs dans certaines zones de la boîte englobante.
- [Ng, Boyle 96] définissent un ensemble de règles pour tenter d'identifier certains symboles isolés et dits "simples". Les critères portent sur des mesure faites sur la boîte englobante : rapport largeur/hauteur, densité des pixels (aire de l'objet divisée par celle du rectangle englobant), comparaison de la densité des pixels dans des sous-sections de la boîte englobante. La procédure récursive de sous-segmentation est ensuite lancée. La classification des primitives est réalisée par la méthode des k-plus-proches-voisins sur l'espace hauteur/largeur de la boîte englobante. D'autres critères sont ensuite ajoutés pour terminer l'identification de certains symboles : par exemple les symboles étiquetés "altération" seront classés en comparant la position de 4 pixels extrêmes (pixels d'abscisses minimale et maximale dans deux sous-rectangles de la boîte englobante).
- [Carter, Bacon 92] stockent dans le graphe des lignes adjacentes les dimensions des boîtes englobantes et la liste des sections qui forment le symbole, et ces critères servent à la classification.
- [Fujinaga 97] (voir aussi [Sayeed Choudhury et al. 01] [Droettboom et al. 02]), crée pour chaque objet un vecteur de caractéristiques comprenant la hauteur, la largeur, le rapport largeur/hauteur, l'aire de la boîte englobante, l'aire de l'objet, le rapport des aires précédentes, des moments centrés normalisés, le nombre moyen de trous par coupes horizontale et verticale. La décision est prise par les k-plus-proches-voisins, avec une mesure de distance pondérée (poids appris au moyen d'un algorithme génétique).
[Armand 93] expose une méthodologie identique, avec une distance euclidienne simple. Les attributs utilisés sont géométriques (masse, surface du rectangle circonscrit, compacité, inertie) et topographiques.
[Baumann, Dengel 92] ont une démarche également similaire, mais la classification est réalisée par un arbre de décision.
- [Hori et al. 99] extraient les hampes sur des critères de longueur et de densité des pixels autour du segment vertical. Un cadre englobant comprenant 8 sous-régions est alors défini à partir de la hampe, et la densité des pixels dans chacune des sous-régions sert de vecteur de caractéristiques pour la classification des notes par un réseau de neurones.

L'inconvénient de ces méthodes est leur sensibilité aux erreurs de segmentation. En particulier, en cas de sur ou de sous-segmentation, on peut supposer que les caractéristiques extraites sur la boîte englobante conduisent à des interprétations erronées.

Projections

Nous avons déjà évoqué les projections verticales utilisées pour la détection des barres de mesure et des hampes. Les projections ont également été très utilisées pour reconnaître les autres primitives ou symboles. Fujinaga a initialement proposé un système fondé sur les projections, pour la segmentation, mais aussi pour la classification [Fujinaga 88]. Le nombre de pics du profil

horizontal local est utilisé en paramètre discriminant, complétant les informations de dimension.

Depuis, on peut citer l'usage des profils dans certains cas particuliers, pour la détection des têtes de note par exemple [Bellini et al. 01]. Bellini effectue les projections verticalement et horizontalement, tandis que [Ramel et al. 94] analysent les profils horizontaux et obliques. Bainbridge identifie aussi les projections comme méthode d'analyse pertinente de certains symboles (comme la clé [Bainbridge, Bell 96] [Fotinea et al. 00]), ainsi que le "slicing" (dénombrer le nombre de transitions sur une coupe de l'objet) [Bainbridge, Carter 97] [Clarke et al. 88].

Des variantes ont également été proposées. Citons [Reed, Parker 96], qui calculent des profils correspondant à des distances entre les points de contour de l'objet analysé et les côtés de sa boîte englobante. La position et l'amplitude des maxima et des minima servent à la classification. La méthode est appliquée sur les objets de type caractère (clés, silences, altérations).

De nouveau, on peut constater que les méthodes proposées sont très pertinentes si les objets peuvent être bien isolés, mais, qu'elles ne sont pas suffisamment robustes dans le cas contraire, comme le font remarquer [Bainbridge, Carter 97] et [Reed, Parker 96].

Squelette

Les squelettes ont été utilisés, aussi bien pour la reconnaissance des symboles de type caractère que pour décomposer les groupes de notes.

Martin applique un algorithme de squelettisation des objets segmentés (par effacement des lignes de portée), et construit un graphe dont les sommets sont les extrémités et les jonctions du squelette [Martin 92]. Après approximation linéaire, il obtient un découpage de l'objet en segments, qui lui permet notamment de détecter les hampes, à partir desquelles il reconstitue les groupes de notes. Pour les autres symboles (altérations, silences, etc.), le rectangle d'encadrement du graphe est partitionné en 9 fenêtres. Un vecteur de 72 bits code la présence ou l'absence de chaque type de sommet dans chaque fenêtre, la présence ou l'absence de segments dont les extrémités sont dans deux zones différentes. La classification est faite selon la distance de Hamming, par réseaux de neurones, ou par arbre de décision. Les meilleurs taux de reconnaissance sont réalisés avec la première méthode, mais des performances comparables sont obtenus par réseaux de neurones, en 6 fois moins de temps.

[Randriamahefa et al. 93] décrivent également la construction d'un graphe attribué à partir du squelette, préalablement simplifié par polygonalisation. Les nœuds du graphe sont les segments, les liens représentent les jonctions. Des paramètres sont calculés pour reconnaître les différentes primitives d'un groupe de notes : pente, distance au contour, connexions entre segments. Les auteurs indiquent que les têtes de note ne peuvent être reconnues de cette manière dans le cas des accords.

Le principal inconvénient de ces méthodes est la sensibilité du squelette au bruit. D'autre part, il y a toujours la notion de rectangle englobant l'objet dont on calcule le squelette, qui ne permet pas de surmonter les problèmes de sur ou sous-segmentation.

Morphologie mathématique

Outre l'utilisation de la squelettisation, quelques projets font un usage intensif des outils de morphologie mathématique.

Modayur définit des éléments structurants pour la détection des primitives de symboles [Modayur 91]. La localisation des barres de mesure et des hampes est réalisée grâce à une ouverture par des segments verticaux. Des régions d'intérêt sont définies par rapport aux segments détectés, pour la recherche des têtes de note (ouverture par un disque). Il s'agit également d'isoler des symboles plus complexes, en définissant des éléments structurants adaptés. Modayur décrit une séquence d'opérations de morphologie mathématique qui permet de détecter les clés de fa (F), en s'appuyant sur le corps ("tête" et "arc" (G)), et sur les deux points de part et d'autre de la quatrième ligne de portée [Modayur 96]. L'idée est d'extraire des caractéristiques assez simples et générales, qui permettent une certaine tolérance par rapport aux défauts de l'image (effacements de pixels, segments coupés, etc.), et sans doute une certaine variabilité des typographies : ainsi, pour la détection du corps de la clé, l'élément est défini par un disque (tête) et deux portions de la ligne médiane de l'arc. Les taux de reconnaissance évoluent entre 89.39% (sopirs) et 100% (dièses).

[Genfang, Shunren 03] décrivent un système fondé sur des transformations en tout ou rien pour la reconnaissance des symboles : têtes de note, altérations, silences, etc. Les pixels du fond et de la forme sont donc pris en compte. Les auteurs semblent considérer que la forme des objets est fixée, et exempte de défauts. Ils obtiennent un taux de reconnaissance global de 94% sur leur base d'images.

L'utilisation de la morphologie mathématique pour la reconnaissance des symboles musicaux est donc assez rare. Les difficultés majeures sont la prise en compte de la variabilité des fontes, la résistance au bruit, la forte ressemblance de certains symboles. Alors que le système [Modayur 96] se focalise sur ces problèmes, celui proposé par [Genfang, Shunren 03] paraît être rigide et inapplicable pour l'analyse d'une grande variété de partitions.

Appariement de formes

Cette méthode a été très utilisée, notamment pour la détection des têtes de note noires. Dans le projet Wabot-2, elle est exécutée par un circuit spécialisé effectuant un ET entre l'image (en intégralité) et huit modèles différents [Matsushima et al. 85]. Depuis, la mise en correspondance est réalisée sur des zones restreintes [Randriamahefa et al. 93] [Bainbridge 96] [Fotinea et al. 00], afin de diminuer le coût de calcul. Par exemple, les têtes de notes sont recherchées le long de la hampe [Martin 92][Miyao, Nakano 95]. Quelques variantes peuvent être mentionnées :

Bainbridge souligne l'intérêt du template matching pour la reconnaissance des symboles sujets aux fragmentations et aux connexions parasites [Bainbridge, Bell 97]. Il indique que les méthodes graphiques de reconnaissance (dans le sens : analyse directe des pixels, par opposition à l'extraction de caractéristiques sur la boîte englobante) ne nécessitent pas la localisation précise des symboles. Les boîtes englobantes servent à définir les zones de corrélation, mais elles peuvent être étendues pour pallier les problèmes de fragmentation, en fonction de la classe testée. Plusieurs

résultats peuvent également être admis dans une même zone, cette fois pour prendre en compte les connexions entre symboles différents. Une version modifiée ("weighted template matching") a été implémentée pour améliorer les résultats de reconnaissance [Bainbridge, Wijaya 99] : le poids attribué à un pixel qui ne correspond pas dépend du résultat sur ses voisins. Des optimisations sont proposées pour réduire le surcoût de calcul. Néanmoins, le système Cantor [Bainbridge, Bell 96] met en œuvre d'autres méthodes de classification (transformée de Hough, projections, slicing), le choix dépendant de la primitive recherchée.

[Reed, Parker 96] reconnaissent les têtes de note par template matching, appliqué sur l'image avec portées. Différents modèles sont donc définis pour prendre en considération la position de la note, sur une ligne ou dans un interligne. Trois types de pixels sont distingués : les pixels du fond (blancs), les pixels objet noirs, les pixels blancs à l'intérieur de l'objet. Seuls les deux derniers types participent au score de corrélation.

Des réseaux de neurones ont également été directement appliqués aux pixels image [Martin 92] [Su et al. 01]. Martin propose de rééchantillonner l'imagette correspondant au symbole analysé en une matrice de 10x10 pixels en niveaux de gris. Couäsnon, qui a mis en œuvre une méthodologie comparable [Coüasnon 91], souligne dans son mémoire de thèse [Coüasnon 96b] que l'information sur les dimensions de l'objet est perdue, et que cette perte génère des erreurs. D'où la nécessité de valider le résultat du classifieur par les dimensions réelles de l'objet.

Transformée de Hough

La transformée de Hough est particulièrement intéressante pour rechercher des formes paramétrées : droites, disques, etc. Les symboles musicaux, qui contiennent de telles primitives (segments, ellipses), dont certaines varient en taille et/ou orientation (typiquement les barres de groupe), peuvent donc être analysés de cette manière. [Stückelberg, Doerman 99] appliquent la méthode pour les segments et les têtes de note, [Wong, Choi 94] pour la détection des notes. Bainbridge classe cette méthodologie dans la catégorie des outils graphiques qui peuvent surmonter des imprécisions de segmentation et des fragmentations [Bainbridge, Bell 97].

Wong et Choi exploitent les relations graphiques entre les primitives composant les notes : en fusionnant les accumulateurs de Hough obtenus, d'une part sur un objet principal (la tête de note noire modélisable par une ellipse), et un objet secondaire (la hampe, assimilable à un segment), on augmente la fiabilité de la détection de l'objet principal. En effet, la détection de l'objet secondaire sert à confirmer la présence de l'objet principal. Les objets secondaires ("supporting objects") utilisés pour la reconnaissance des têtes de note peuvent être la hampe, mais aussi un crochet (pour les croches isolées) ou un point de durée.

Notons enfin que Martin a également testé la transformée de Hough pour la détection des têtes de note noires [Martin 89], mais que, face aux difficultés rencontrées (temps de calcul, interprétation de l'espace des votes), il a finalement choisi le template matching [Martin 92].

1.3.4.2. Prise en compte de la variabilité

L'édition musicale présente deux caractéristiques : une forte variabilité des typographies, une certaine évolutivité des signes et de la manière de transcrire la musique. La plupart des projets présentés traitent la notation musicale classique (CMN : Common Music Notation), pour laquelle seule la variabilité des polices pose réellement problème, les signes essentiels à la restitution de la musique étant quant à eux bien connus.

Fujinaga présente un système fondé sur l'extraction de caractéristiques et la classification par les k-plus-proches-voisins, qui permet d'inclure de nouvelles classes (extensibilité), ou de nouveaux prototypes de symboles pour affiner leur reconnaissance (prise en compte de la variabilité des polices). Un algorithme génétique permet de réaliser l'apprentissage hors ligne des poids de la distance euclidienne utilisée pour la décision [Fujinaga 95] [Fujinaga et al. 98] [Sayeed Choudhury et al. 01].

Bainbrige traite en revanche davantage de l'extensibilité du système de reconnaissance à d'autres types de notations musicales, grâce à un formalisme et une structuration du programme permettant une certaine généralité. Les primitives à reconnaître sont décrites dans un langage spécifique, Primela [Bainbridge 96]. Le système Cantor permet de reconnaître la notation musicale classique, mais aussi d'autres notations comme le chant grégorien. En revanche, le problème de la variabilité des formes est moins bien géré : la description des primitives semble assez rigide, conduisant à de nombreuses erreurs de reconnaissance lorsqu'une description est mal adaptée à la partition traitée, et nécessitant un ajustage manuel des paramètres [Bainbridge, Bell 96].

1.3.4.3. Quelle méthode ?

La question qui se pose, face à cette diversité des méthodologies, est naturellement de savoir quelle est la plus appropriée pour la reconnaissance des primitives musicales. Il serait très présomptueux de donner une réponse à cette question, d'autant que les taux de reconnaissance des systèmes proposés (s'ils sont indiqués) sont obtenus avec des bases d'images et des définitions de primitives très différentes, et ils sont donc difficilement comparables.

On peut néanmoins supposer que les méthodes qui réalisent des mesures (paramètres géométriques, moments, projections, etc.) sur des rectangles englobants pourront difficilement faire face aux problèmes de segmentation. En revanche, celles qui analysent directement les pixels, sur des zones image qui n'ont pas à être déterminées précisément (transformée de Hough, template matching, réseaux de neurones directement appliqués aux pixels), permettront sans doute de mieux résoudre ces difficultés. Leur inconvénient est généralement une plus grande sensibilité à la variabilité des formes, comparativement aux techniques du premier type, qui portent davantage sur la structure des objets. Cette remarque ne vaut néanmoins que si les objets sont bien localisés. Les méthodes fondées sur la morphologie mathématique semblent a priori très sensibles aux défauts de segmentation et au bruit (extraction du squelette), et difficiles à mettre en œuvre dans le cas de systèmes omni fontes (problème de définition de l'élément structurant).

C'est pourquoi certains auteurs préconisent de mettre en œuvre, non pas une unique

technique, mais plusieurs techniques, suivant la primitive recherchée. Cette démarche peut néanmoins poser un problème au niveau de l'homogénéité des résultats pour la prise de décision : par exemple, comment comparer deux hypothèses obtenues pour un même objet, l'une par extraction de caractéristiques et k-ppv, et l'autre par template matching? Cette comparaison n'est d'ailleurs généralement pas faite, car les différentes classes de symboles sont extraites de manière séquentielle (e.g. [Randriamahefa et al. 93] [Ramel et al. 94]). Seuls Bainbridge et Stückelberg mentionnent la possibilité d'évaluer des hypothèses concurrentes qui ne sont pas nécessairement reconnues par le même procédé, mais qui sont affectées d'un degré de confiance [Bainbridge, Bell 03] ou d'une probabilité [Stückelberg et al. 97] [Stückelberg, Doerman 99].

On remarque par ailleurs l'introduction de connaissances a priori dans les méthodes de reconnaissance.

D'une manière évidente, les algorithmes de reconnaissance encodent des informations concernant la forme et/ou la position des symboles. C'est en effet la description des primitives musicales, au niveau pixel ou dans l'espace des caractéristiques, qui permet de les reconnaître. Dans certains systèmes cependant, on remarque que cette information est introduite de manière ponctuelle, sans unité de formalisation. Des tests isolés, fondés sur un ou quelques critères stricts, sont utilisés pour prendre des décisions définitives : hypothèse sur l'épaisseur maximale d'un segment vertical [Sicard 92], longueur minimale d'une hampe [Ramel et al. 94] [Genfang, Shunren 03], etc. Bainbridge note que des descriptions trop strictes et figées des primitives musicales ne permettent pas de prendre en compte la diversité des éditions musicales, la variabilité des primitives [Bainbridge, Bell 96]. Si les résultats sont bons sur les partitions conformes à la modélisation, de nombreuses erreurs sont en revanche faites sur les autres, et les paramètres internes du système doivent alors être ajustés.

On constate également l'introduction d'informations de plus haut niveau dans le processus d'étiquetage des primitives. Des critères structurels et syntaxiques sont utilisés pour définir les zones de recherche de certaines primitives en fonction de primitives déjà reconnues, typiquement pour la reconnaissance des groupes de notes (e.g. [Ramel et al. 94] [Sicard 92]), mais aussi pour la reconnaissance des altérations ou des points de durée, dont la localisation est contrainte par la position des têtes de note [Matsushima et al. 85] [Fujinaga 88] [Kato, Inokuchi 90]. De nouveau, le problème de la gestion de l'ambiguïté se pose : que se passe-t-il si une primitive, mal étiquetée, à cause de critères trop stricts par exemple, conditionne la reconnaissance d'une autre primitive ? [Kato, Inokuchi 90] prévoient de revenir aux modules de bas niveau lorsque des incohérences sont détectées dans les modules de plus haut niveau ; mais de nombreux auteurs n'envisagent pas de remise en question, ou corrigent, par des méthodes ad hoc, des erreurs de reconnaissance détectées grâce à la vérification ponctuelle de règles syntaxiques [Ng, Boyle 96]. On constate de nouveau un manque d'unité dans la formalisation. De plus, les critères utilisés, en détection et en correction, sont trop locaux et ne prennent pas en compte l'ensemble du contexte.

1.3.5. Modélisations structurelles et syntaxiques

Afin de formaliser de manière rigoureuse la notation musicale, certains auteurs ont proposé des méthodologies fondées sur des grammaires. Cette approche se justifie amplement, car beaucoup

d'informations sont véhiculées par les relations spatiales entre les objets musicaux : notes constituées de primitives qui satisfont à des règles d'assemblage, interactions graphiques contraignant les positions relatives de certaines classes de symboles, comme la position d'une altération ou d'un point par rapport à la note.

Andronico a proposé le premier une formalisation grammaticale, celle-ci étant composée de deux niveaux [Andronico et al. 82]. Le haut niveau concerne l'organisation générale de la notation musicale (séquences de portées, position de la clé, de l'armure, de la signature temporelle). La grammaire de bas niveau décrit la structure des symboles, et elle est utilisée pour leur reconnaissance. Cinq opérateurs de position (au-dessus, au-dessous, à droite, au-dessus à droite, au-dessus à gauche) permettent d'établir les liens entre les terminaux et les non-terminaux. Les terminaux sont les têtes de note, noires et blanches, et les segments orientés.

Fahmy propose une grammaire de graphes attribués [Fahmy, Blostein 91], les graphes se prêtant bien à la manipulation d'informations 2D. L'objectif n'est pas de reconnaître les symboles, mais de reconstituer l'interprétation de haut niveau à partir des primitives reconnues. Un nœud du graphe représente une primitive ou un symbole, et un arc un lien sémantique entre deux primitives. Un mécanisme de réécriture permet de passer du graphe initial composé de nœuds isolés (les primitives reconnues), au graphe final, représentatif du contenu sémantique de la partition (par exemple une note, avec sa hauteur et sa durée). Ce passage d'un graphe à un nouveau graphe, reflétant un plus haut niveau de compréhension, se fait par des règles de production, modélisant la connaissance a priori. Elles sont formées de :

- un prédicat P définissant les conditions d'application de la règle,
- une transformation T sur le graphe permettant de générer le nouveau graphe,
- une fonction F de calcul des nouveaux attributs.

Le système est séquencé en trois passes : la première établit les liens potentiellement intéressants, la seconde supprime des associations inintéressantes ou conflictuelles, la troisième incorpore la sémantique dans les attributs des associations restantes, et supprime les nœuds inutilisés. Le nombre de nœuds diminue donc progressivement, et le contenu informatif est mémorisé dans les attributs. L'interprétation finale est ainsi obtenue. Il faut néanmoins remarquer que la méthode suppose que toutes les primitives sont bien segmentées et parfaitement reconnues.

Baumann a également proposé ce type de grammaire [Baumann, 95], en sortie d'un module de classification qui produit trois hypothèses de reconnaissance par objet segmenté. Il y a donc une certaine prise en compte de l'incertitude sur la classe des primitives. L'auteur avance en conclusion deux améliorations : la possibilité de traiter plusieurs configurations en parallèle, avec des coefficients de confiance qui permettront de sélectionner la meilleure, et l'incorporation de critères sémantiques (probablement ce que nous appelons dans ce mémoire critère syntaxique par opposition à critère graphique).

Constatant également la nécessité de gérer l'ambiguïté de classification, Fahmy et Blostein ont étendu leur méthodologie [Fahmy, Blostein 98]. Le classifieur peut cette fois produire plusieurs hypothèses de reconnaissance. Celles-ci sont toutes représentées dans les nœuds initiaux du graphe, mais reliées par un lien d'exclusion. La grammaire de graphes est cette fois utilisée, non seulement

pour restituer la sémantique, mais aussi pour réduire l'incertitude. Le résultat du graphe consiste en une ou plusieurs interprétations possibles. Deux aspects sont très intéressants dans la méthodologie présentée : d'une part l'ambiguïté est préservée de bout en bout, d'autre part le modèle est hiérarchique : les deux premiers niveaux de la grammaire sont relatifs à des relations binaires (associer un objet à une mesure, associer un crochet à une hampe, etc.), le dernier exprime une relation d'ordre supérieur (le nombre de temps totalisés par tous les symboles de la mesure). Certaines limitations sont cependant à noter : les contraintes formalisées sont toutes strictes et sont surtout limitées aux aspects graphiques (ainsi, la cohérence syntaxique des altérations n'est pas modélisée), la segmentation est toujours supposée parfaite. Les auteurs soulignent en conclusion l'intérêt qu'il y aurait à introduire un paramètre de confiance sur l'identité des primitives, et certaines contraintes souples de la notation musicale.

Coüasnon propose une grammaire qui cette fois contrôle tout le processus de reconnaissance, y compris la segmentation, jugée impossible à réaliser parfaitement en l'absence d'informations contextuelles [Coüasnon 96b]. La grammaire est composée d'une partie graphique qui permet de reconnaître les notes par la description de leur structure et du positionnement relatif de leurs attributs (altérations, points, signes d'attaque), et d'une partie syntaxique, pour la reconnaissance de tous les symboles qui se rattachent à une voix. Une grande différence par rapport aux approches classiques est que l'analyseur associé à la grammaire permet de modifier la structure de données analysée, pour introduire le contexte dans la phase de segmentation [Coüasnon, Camillerapp 94]. Supposons par exemple le cas de deux altérations qui se touchent : elles correspondent alors à une composante connexe non reconnue. En prenant en compte le contexte (la présence d'une tête de note), il est possible de deviner la présence d'une altération au niveau de la composante non reconnue, et, sous cette hypothèse, de tenter une segmentation adaptée. Les nouvelles matrices de pixels sont alors reproposées au classifieur, et, en cas de succès, elles remplacent la matrice de pixels initiale dans la structure de données [Coüasnon 96a]. Les erreurs de segmentation peuvent donc être corrigées. La partie syntaxique de la grammaire permet en outre de détecter des erreurs de reconnaissance, en vérifiant la cohérence des durées par rapport à l'alignement vertical dans l'image (partitions d'orchestre) [Coüasnon, Rétif 95]. Coüasnon note l'importance du classifieur, qui doit être performant. Le classifieur utilisé étant insuffisant, la méthodologie n'a pas pu être complètement testée [Coüasnon 96b]. Cette démarche est très intéressante pour la résolution des problèmes de segmentation, mais elle ne semble pas gérer d'autres sources d'ambiguïté, comme la variabilité des symboles. D'autre part, les critères utilisés pour la reconnaissance restent locaux et restreints à l'aspect graphique.

Bainbridge et Bell proposent également la définition d'une grammaire, pour la reconstruction des notes et la restitution du contenu sémantique [Bainbridge, Bell 96] [Bainbridge, Bell 03]. La volonté affichée est de réduire la complexité des systèmes présentés, en limitant la grammaire à l'assemblage des primitives, et en adaptant une grammaire DCG (Definite Clause Grammar) au traitement d'informations 2D. Le résultat de l'analyse est un ensemble de graphes, qui décrivent les symboles reconstitués, et qui sont transmis au module d'analyse sémantique. Cette seconde étape vise à établir des liens entre les objets, d'après leurs relations spatiales, et à restituer le contenu informatif de l'image par des procédures adaptées : hauteur de la note en appliquant la clé, l'armure et les altérations accidentelles, durée des symboles, synchronisation des voix, etc.

Conclusion

Toutes ces approches, fondées sur des grammaires, modélisent essentiellement les règles graphiques décrivant la structure des notes, le positionnement de leurs attributs. Elles permettent également de reconstituer le contenu sémantique de la partition. Les limitations suivantes ont été constatées : d'une part, les décisions prises durant les étapes premières de l'application d'une grammaire reposent sur des informations très locales, limitant l'intelligence qui peut être appliquée [Watkins 96] ; d'autre part, les règles syntaxiques impliquant de nombreux symboles ne sont pas modélisées (la vérification ultime de la métrique exceptée), probablement à cause de leur caractère plus global et de leur plus grande flexibilité.

1.3.6. Prise en compte de l'incertitude

Comme nous l'avons souligné au paragraphe 1.2, les sources d'ambiguïté sont très nombreuses dans le domaine de la reconnaissance des partitions : difficultés de segmentation de l'image en entités cohérentes, variabilité des symboles, imprécision et flexibilité des règles musicales, etc. Voyons maintenant plus en détail les idées directrices qui ont été proposées jusqu'à présent pour prendre en compte l'incertitude qui en résulte.

Génération d'hypothèses – degrés de confiance

Une première approche, déjà évoquée, est de procéder par génération d'hypothèses de reconnaissance. Dans [Fahmy, Blostein 98], plusieurs primitives peuvent être présentes, par objet, dans les nœuds initiaux du graphe, et c'est l'application des règles de production qui permet de ne retenir que les hypothèses cohérentes par rapport à la théorie musicale. Les auteurs suggèrent, à l'instar de [Baumann, 95], d'intégrer des coefficients de confiance, idée qui a été mise en application par [Bainbridge, Bell 03] : les décisions de classification sont assorties d'un score compris entre 0 et 1, qui permet de retenir un assemblage de primitives parmi un ensemble de possibilités, par maximisation d'un score.

On note cependant que les décisions prises restent très locales, et cela restreint considérablement l'efficacité de ces méthodes [Watkins 96]. Watkins indique que les choix doivent au contraire être différés jusqu'à ce que toute l'information, locale et globale, soit disponible. Pour cela, il propose une grammaire floue, dans laquelle le prédicat d'applicabilité binaire est remplacé par une fonction de certitude continue. Ainsi, un prédicat du type "tête de note proche de la hampe" prend des valeurs comprises entre 0 et 1, indiquant dans quelle mesure cette relation est vraie. Notons en outre que cette modélisation correspond aux situations réelles, puisque ce type de relation n'est en réalité pas défini précisément par la théorie musicale. La méthode est cependant restreinte à la construction des notes, sans introduction de relations syntaxiques.

Méthodes bidirectionnelles

La majorité des systèmes présentés passe séquentiellement des traitements de bas niveau aux modules d'interprétation de haut niveau, sans remise en cause des résultats antérieurs. Un système

bidirectionnel donne la possibilité de revoir des décisions après l'introduction d'informations contextuelles obtenues dans les modules de haut niveau. Cette nouvelle architecture offre donc la possibilité de gérer l'ambiguïté.

Kato et Inokuchi ont, les premiers, proposé ce type d'approche, pour la reconnaissance de partitions de piano [Kato, Inokuchi 90] [Kato, Inokuchi 92]. La structure du système est composée d'une mémoire de 5 couches, correspondant à 5 niveaux d'abstraction, et permettant à 4 modules de traitement de communiquer dans les deux sens :

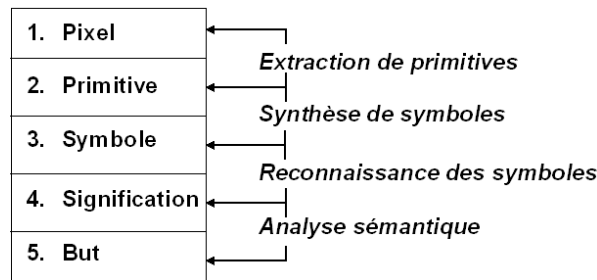


Figure 1.13 : Architecture bidirectionnelle proposée par Kato et Inokuchi [Kato, Inokuchi 90]

Les unités de reconnaissance sont gouvernées par un seuil variable qui contrôle le degré d'appariement requis pour l'extraction d'une primitive. Les primitives reconnues sont effacées dans la couche image. Dans les couches hautes, les règles sont appliquées sur les objets, suivant un ordre de priorité prédéfini, jusqu'à ce qu'un résultat soit produit dans la couche but. Si ce résultat n'est pas généré, alors les hypothèses inacceptables sont rejetées aux niveaux hauts, et les traitements des modules de bas niveau sont repris : les primitives sont restaurées et de nouveau analysées, avec cependant un seuil d'appariement moins sévère. [Ferrand et al. 99] présentent une méthodologie qui s'inspire beaucoup de celle proposée par Kato et Inokuchi. Les erreurs de métrique révélant la présence de symboles qui n'ont pas été classifiés, des hypothèses sur ces symboles manquants sont faites, et les traitements de bas niveau sont revus, avec des paramètres plus lâches, afin de corriger ces erreurs.

D'autres auteurs se sont ensuite tournés vers des architectures bidirectionnelles. Stückelberg suggère un système en trois couches ("Metaprocessor", "Conceptual system", "Features detector") qui communiquent dans les deux sens [Stückelberg et al. 97]. L'incertitude est gérée dans le cadre de la théorie des probabilités. Les hypothèses sont développées, par déductions successives, sous le contrôle du module de plus haut niveau ("Metaprocessor") qui analyse les probabilités obtenues. L'article est cependant très prospectif, et la méthodologie ne semble pas avoir été implémentée dans un système qui fonctionne.

McPherson [McPherson, Bainbridge 01] reprend les travaux de Bainbridge, en proposant de transformer l'architecture du système Cantor [Bainbridge 97]. Les différents modules de traitement (détection des portées, extraction des primitives, reconnaissance des primitives, assemblage des primitives, etc.) sont désormais contrôlés par une structure ("Co-ordinator"), qui dirige l'exécution du flux, les modules ne communiquant jamais directement entre eux. Cette architecture permet de corriger des erreurs, par détection d'incohérence et ré-exécution de certaines tâches : par exemple, si un bémol ne correspond à aucune note dans la phase d'assemblage, le module transmet cette

information au coordinateur qui relance la reconnaissance de la primitive, avec des paramètres différents. Cette architecture permet donc de compléter ou de revoir certaines décisions, en introduisant du contexte. Mais les exemples présentés montrent qu'une information partielle et non globale est utilisée. Dans [McPherson 02], l'auteur combine génération d'hypothèses et architecture bidirectionnelle pour gérer l'incertitude : le module de reconnaissance de primitives peut fournir, pour un objet donné, une liste de classes qui peuvent avoir un sens avec les objets voisins. L'implémentation ne semble cependant pas finalisée, et l'apport de la méthodologie par rapport au système initial n'est pas encore évalué.

Détection, correction des erreurs

Dans le paragraphe précédent, nous avons mentionné les systèmes fondés sur une architecture volontairement bidirectionnelle. Certains systèmes, qui demeurent essentiellement unidirectionnels, intègrent néanmoins des procédures rétroactives : la cohérence du résultat final de classification est vérifiée, et des corrections sont effectuées. Il est à noter que les tests de cohérence sont toujours fondés sur la métrique : vérification du nombre de temps par mesure, vérification de l'alignement temporel dans le cas de la musique polyphonique. Les corrections sont réalisées avec plus ou moins de rigueur. Dans [Coüasnon, Rétif 95], il s'agit de modifier la durée de notes ; dans [Droettboom et al. 02], sept procédures ad hoc de correction sont successivement testées, tant que la correction proposée ne permet pas de restituer la cohérence recherchée. [Blostein, Haken 99] présentent une démarche originale qui consiste à re-générer une image à partir des résultats de classification, en utilisant un éditeur. Les défauts d'alignement de deux voix synchrones révèlent de nouveau les erreurs de classification, et différentes corrections possibles sont énumérées. La correction retenue est celle qui restaure l'alignement et qui se rapproche au mieux du format de mise en page usuel, en respectant notamment l'espacement des notes en fonction de leur durée.

Conclusion

Deux voies principales ont donc été explorées pour traiter l'ambiguïté : générer des hypothèses concurrentes et extraire la combinaison qui satisfait aux règles musicales, ou adopter des architectures qui autorisent une rétroaction, de manière à revoir des décisions en fonction du contexte. Les procédures de correction s'apparentent à ces architectures bidirectionnelles.

Quelle que soit l'orientation choisie, on peut remarquer que les règles intégrées dans ces systèmes sont limitées aux règles graphiques locales (assemblage des primitives, position des attributs de notes), et à la vérification du nombre de temps par mesure. L'alignement de voix synchrones est, quant à lui, utilisé en détection/correction d'erreurs exclusivement. Toutes les connaissances du domaine ne participent donc pas à la décision : en particulier les règles souples concernant les altérations ou les regroupements de notes semblent n'avoir jamais été modélisées.

L'incertitude relative à la classification d'une primitive ou d'un symbole musical n'est généralement pas formalisée, sauf dans [Stückelberg et al. 97] [Stückelberg, Doermann 99] dans le cadre probabiliste, mais de manière très prospective. Certains auteurs mentionnent la possibilité de calculer des scores de confiance qui quantifieraient l'incertitude attachée à une hypothèse de classification, sans néanmoins expliciter ce calcul, ni intégrer ces scores dans une évaluation

globale de la validité d'une combinaison d'hypothèses. Au contraire, l'incertitude devrait plutôt être propagée de bout en bout, jusqu'à la décision finale.

La souplesse de la notation musicale n'est pas non plus formalisée : l'imprécision des règles graphiques n'est pas prise en compte, sauf dans [Watkins 96], et les règles syntaxiques flexibles (sur les altérations et la tonalité, l'organisation rythmique des groupes de notes) ne sont pas intégrées.

Il semblerait donc que l'imprécision et l'incertitude, relatives aux informations extraites de l'image ou aux connaissances génériques, soient encore assez peu traitées, et que les règles musicales intégrées dans les systèmes, pour la réduction de l'ambiguïté, soient encore incomplètes.

1.3.7. Principaux systèmes et évaluation

La recherche dans le domaine de la reconnaissance de partitions musicales a commencé avec les travaux de Pruslin [Pruslin 66], et n'a cessé d'être active depuis. C'est néanmoins à la fin des années 80 qu'elle a pris davantage d'ampleur, en relation avec l'émergence des ordinateurs personnels et l'accroissement considérable de la puissance de calcul.

Les systèmes qui ont été proposés jusqu'au début des années 1992 sont analysés dans [Blostein, Baird 92]. La conclusion de cet article mettait en évidence les difficultés relatives à la segmentation de l'image, notamment le problème des objets fragmentés ou qui se touchent, et le manque de généralité des méthodologies proposées, limitées à un sous-ensemble de la notation musicale. Beaucoup de recherches avaient été menées pour la reconnaissance des primitives, mais très peu au niveau de l'intégration des règles musicales, et les auteurs constataient effectivement la difficulté réelle que constitue la formalisation de l'écriture musicale, compte tenu de sa complexité. L'un des projets les plus aboutis semblait être celui proposé par Kato et Inokuchi, pour la reconnaissance des partitions de piano, avec des taux de reconnaissance allant de 83.3% à 95.6% sur les quatre partitions évaluées [Kato, Inokuchi 92].

L'évaluation et la comparaison rigoureuse des systèmes proposés sont impossibles à réaliser : les partitions traitées sont très différentes (musique monodique ou polyphonique, niveau de difficulté variable, etc.) et les objectifs, en termes de primitives devant être reconnues, sont également très différents [Blostein, Baird 92] [Coüasnon 96b] [Bainbridge, Carter 97]. Ce problème n'est d'ailleurs pas encore résolu : il n'existe toujours pas de base d'images de référence, ni de méthodologie pour l'estimation de la fiabilité et de la précision des systèmes, bien que des propositions aient été faites [Baumann, Tombre 95] ou soient en cours d'élaboration [Interactive Music Network]. Des taux de reconnaissance sont néanmoins parfois calculés, mais sur une base de données trop limitée (e.g. [Kato, Inokuchi 92]) ou manquant de généralité (e.g. [Bainbridge, Wijaya 99]).

Actuellement, trois groupes de recherche au moins sont très présents dans le domaine, et proposent des systèmes assez complets et déjà fonctionnels.

Le premier s'est développé sur la base des travaux de Fujinaga [Fujinaga 97], dont l'originalité repose essentiellement sur la capacité du système à apprendre de nouveaux prototypes

(système adaptable). Les taux de reconnaissance atteignent 99% [Fujinaga et al. 98] ; mais aucune indication sur la base de test n'est donnée, l'idée étant surtout de comparer des résultats pour démontrer l'apport de l'algorithme génétique. Le système s'est ensuite enrichi d'une analyse syntaxique et sémantique [Droettboom et al. 02]. Il doit permettre la reconnaissance d'une large collection de partitions (musiques populaires américaines), et la constitution d'une base de données incluant des fichiers Midi [Sayeed Choudhury et al. 00].

Le deuxième système, nommé Cantor et développé par Bainbridge, est extensible, dans le sens où le programme peut être adapté pour la reconnaissance de notations musicales autres que la notation classique (CMN) [Bainbridge 97]. Les recherches se sont ensuite orientées vers les formalisations grammaticales [Bainbridge, Bell 03]. Les taux de reconnaissance, indiqués dans [Bainbridge, Wijaya 99], sont obtenus sur deux recueils de partitions, et sont respectivement égaux à 94.6% et 93.7%. Les primitives creuses (comme les blanches et les rondes), particulièrement sensibles à l'effacement des lignes de portée, ont un taux de reconnaissance de 77%. Les points de durée sont également mal reconnus. Actuellement, le système évolue vers une architecture bidirectionnelle [McPherson, Bainbridge 01] [McPherson 02] qui pourrait permettre de mieux gérer l'ambiguïté.

Le troisième projet, O³MR, mené par Nesi [Marinai, Nesi 99] [Bellini et al. 01], est proposé par l'université de Florence [O³MR]. Des évaluations ont été réalisées et comparées avec deux logiciels commerciaux, dont SmartScore [Musitek], sur sept images de test [Interactive Music Network]. Les résultats produits par SmartScore et O³MR sont comparables. Le nombre d'images traitées est cependant trop faible pour que l'évaluation soit vraiment significative.

Ces trois systèmes reconnaissent les partitions monodiques, éventuellement les accords, d'après les exemples présentés.

Le premier logiciel commercial, MidiScan, a été lancé il y a une quinzaine d'années. Il produisait des résultats assez décevants. Depuis, on a pu constater de nets progrès, et les systèmes deviennent de plus en plus efficaces en termes de rapidité, de fiabilité, et d'ergonomie. Actuellement, le plus avancé semble être SmartScore, annonçant un taux de reconnaissance dépassant 99% sur des partitions bien imprimées [SmartScore 06]. Le test de la version d'évaluation montre que les taux de reconnaissance chutent néanmoins considérablement dans des conditions moins idéales. L'analyse des résultats produits indique que les échecs sont dus à des erreurs d'extraction ou de reconnaissance des primitives, mais aussi à une intégration insuffisante des règles musicales dans la méthode de reconnaissance (voir les exemples présentés au chapitre 7, paragraphe 7.5).

1.4. Conclusion

De nombreux travaux ont donc déjà été menés dans le domaine de l'OMR, conduisant à des méthodologies très différentes, concurrentes ou complémentaires, mais ne résolvant pas encore tous les problèmes.

Une caractéristique commune aux systèmes présentés est leur architecture, constituée globalement des étapes suivantes : prétraitements, détection des lignes de portée, segmentation,

reconnaissance des primitives, analyse syntaxique et interprétation sémantique.

De nombreux articles ont été consacrés à l'extraction des lignes de portée. Leur localisation précise, faisant face aux défauts courants (biais, courbure), s'est avérée indispensable. On peut retenir comme méthodes simples qui semblent bien fonctionner le calcul de l'interligne et de l'épaisseur des lignes de portée par l'analyse des histogrammes des empan noirs et blancs [Kato, Inokuchi 90]. Réaliser une projection horizontale de l'image redressée semble également pertinent, si le résultat est complété par une analyse fine de la position des lignes de portée, plus robuste aux symboles interférents que ce qui a été proposé jusqu'à présent.

Les méthodes pour la segmentation et la reconnaissance des symboles sont ensuite extrêmement variées, bien que la segmentation soit presque toujours amorcée par l'effacement des lignes de portée. On peut remarquer que segmentation et reconnaissance sont souvent imbriquées, sans unité de formalisation. Cela est dû à la difficulté de segmenter l'image en entités musicales, sans aucune connaissance préalable sur son contenu. Cette méthodologie est très certainement source d'erreurs. Elle ne permet pas, en tout cas, de gérer toute l'ambiguïté, car des décisions sont prises sur la base de connaissances incomplètes, voire erronées. Un axe d'investigation important sera donc de trouver des procédures qui permettent de bien séparer la segmentation de l'analyse des primitives, et de définir des méthodes de reconnaissance qui permettent de faire face à l'ambiguïté provenant des imprécisions de segmentation.

Concernant l'analyse proprement dite, beaucoup de propositions ont également été faites. La plupart d'entre elles sont fondées sur les rectangles englobants et l'extraction de caractéristiques géométriques et structurelles. Beaucoup moins d'auteurs ont exploré les méthodes qui analysent directement les pixels de l'image, comme le template matching. Cette voie semble cependant très intéressante, car elle ne nécessite pas une segmentation précise des formes. Elle semble en outre plus robuste aux défauts d'impression (symboles fractionnés ou qui se touchent). Les ambiguïtés de classification sont généralement traitées par des procédures ad hoc, utilisant des connaissances sur la notation musicale. De nouveau, il faut souligner le manque d'unité de formalisation, qui ne permet probablement pas de résoudre correctement l'ambiguïté. Un objectif, qui doit par conséquent être fixé, est de n'introduire dans la phase de reconnaissance que les connaissances relatives aux symboles eux-mêmes, indépendamment des autres, les règles musicales définissant les relations entre les symboles ne devant intervenir que dans les phases d'interprétation de haut niveau.

La dernière étape, l'analyse syntaxique, a été beaucoup moins développée dans la littérature que les précédentes. Il s'agit surtout de reconstruire les symboles à partir des primitives, et de restituer l'interprétation de haut niveau, par des méthodologies fondées sur des grammaires. Celles-ci formalisent les contraintes sur la position relative des primitives ou des symboles. Les règles syntaxiques, portant sur les altérations, les groupements de notes, ne sont pas introduites, probablement à cause des difficultés suivantes : leur flexibilité, et le fait qu'elles concernent un nombre quelconque de symboles pouvant être distants dans la partition. Les imprécisions sur la position des objets, dues aux erreurs de segmentation et d'analyse, la souplesse de la notation musicale elle-même, ne sont pas non plus modélisées. De nombreux problèmes restent donc ouverts au niveau de la modélisation de la notation musicale et de son intégration dans la méthode de

reconnaissance.

Enfin, on peut remarquer que les systèmes ne gèrent généralement pas l'ambiguïté de classification, ou alors insuffisamment. Beaucoup d'auteurs soulignent la nécessité de remédier à cette lacune (e.g. [McPherson 02]). La méthodologie proposée dans [Fahmy, Blostein 98], procédant par génération d'hypothèses et décision, semble la plus appropriée pour prendre en compte simultanément toutes les sources d'ambiguïté.

En résumé, cette étude montre que certaines voies ont encore été insuffisamment explorées :

- modéliser l'imprécision et l'incertitude liées aux informations extraites de l'image.
- modéliser et intégrer dans le système de reconnaissance l'ensemble des règles musicales régissant les relations entre les symboles, en allant au-delà des règles graphiques locales. En particulier, les règles syntaxiques relatives aux altérations, à la tonalité, à la métrique, doivent être considérées, au même titre que les règles graphiques.
- modéliser l'imprécision et la flexibilité de ces règles.
- fusionner toutes ces informations de manière à prendre une décision par optimisation globale (par opposition à des décisions locales successives).

Ces différents points devraient permettre de mieux gérer l'ambiguïté et d'accéder à une plus grande fiabilité. Nous essaierons donc de proposer des solutions à ces problèmes qui restent très ouverts.

Les autres objectifs importants que nous nous fixons, et qui sont encore insuffisamment atteints dans la littérature, sont les suivants :

- proposer une méthodologie qui sépare bien les différentes étapes du système de reconnaissance, et qui formalise de manière rigoureuse les connaissances a priori pouvant être utilisées pour rendre le système plus robuste.
- proposer des méthodes de segmentation qui surmontent au mieux les défauts d'impression, en particulier les problèmes de fragmentation et de connexion entre symboles.
- proposer des méthodes d'analyse des symboles, capables de faire face à ces défauts, ainsi qu'aux imprécisions de segmentation. L'analyse par template matching, très peu appliquée en OMR, semble une voie intéressante.

Un autre axe de recherche innovant concerne les différentes procédures qui permettraient de gagner en robustesse et en souplesse d'utilisation, comme l'indication automatique d'erreurs potentielles, ou l'apprentissage supervisé d'une partition spécifique. Nous tenterons également d'apporter des éléments de réponse à ces questions, très peu étudiées jusqu'à présent.

CHAPITRE 2

Structure du système de reconnaissance proposé

Dans ce chapitre, nous rappelons les objectifs de notre étude, et nous en précisons le cadre : type de partitions analysées et constitution de la base d'images. Ensuite, nous présenterons la structure générale de notre système de reconnaissance, et nous discuterons de l'intérêt de cette architecture par rapport à celles qui ont déjà été proposées.

2.1. Type de partitions traitées et objectifs

Notre objectif est de proposer de nouvelles méthodes, qui permettent de contribuer à la résolution de difficultés encore insuffisamment surmontées en reconnaissance de partitions musicales imprimées, et donc d'obtenir une plus grande fiabilité des résultats. Les difficultés que nous souhaitons traiter en priorité sont les suivantes :

- prise en compte de l'incertitude, due à la variabilité des polices de symboles, aux défauts d'impression du document original, et à l'imperfection de la segmentation.
- modélisation et intégration dans le processus de reconnaissance des règles de la notation musicale, qu'elles soient strictes ou souples.

Deux autres points seront également étudiés, de manière à améliorer la robustesse du système :

- indication automatique des erreurs de reconnaissance potentielles.
- adaptation du système de reconnaissance à la partition traitée.

L'indication automatique d'erreurs potentielles est un aspect très important, bien qu'il n'ait été abordé, à notre connaissance, que par [Coüasnon, Rétif 95]. En effet, la vérification de la partition reconstruite, symbole par symbole, est une tâche longue et fastidieuse, même si les taux de reconnaissance sont globalement satisfaisants. Au final, le gain de temps obtenu par rapport à une édition manuelle complète est fortement réduit, et, par conséquent, l'intérêt de la reconnaissance automatique n'est plus évident.

Le dernier point concerne l'adaptation à la partition traitée. L'idée est de réaliser un apprentissage, à partir d'un extrait de la partition, reconnu par le système et corrigé par l'utilisateur. Cet apprentissage supervisé peut concerner les modèles de classe ou d'autres spécificités, et permet

d'aller plus loin dans la résolution des problèmes liés à la diversité de l'édition musicale. On autorise donc une certaine interactivité entre le système et l'utilisateur, qui peut s'avérer particulièrement intéressante lorsque le volume à traiter est important : le temps consacré au cycle de reconnaissance/correction d'un extrait est largement compensé par l'amélioration des taux de reconnaissance sur le reste de la partition, d'autant plus que l'indication des erreurs potentielles facilite l'intervention de l'utilisateur.

Nous avons décidé de traiter dans un premier temps les partitions monodiques (Figure. 2.1a) uniquement. En d'autres termes, nous ne traitons pas les partitions qui présentent des accords, plusieurs voix écrites sur une même portée, ou des voix écrites sur plusieurs portées simultanément. Cela exclut typiquement les partitions de piano (Figure 2.1c). En revanche, les partitions de musique de chambre ou d'orchestre peuvent être analysées, si chaque voix est monodique et inscrite sur une portée indépendante (Figure 2.1b). Chaque voix est alors analysée indépendamment des autres, sans aucune vérification de leur cohérence mutuelle.

Nous traitons donc un sous-ensemble de la notation musicale, qui va nous permettre d'évaluer l'apport de notre méthodologie, sans avoir à faire face simultanément à toutes les difficultés des partitions les plus complexes. Nous validerons un processus de reconnaissance complet, c'est-à-dire réalisant tous les traitements à partir de l'image scannée, et nous effectuerons une analyse réaliste des résultats obtenus sur une large base de partitions. Néanmoins, l'extension de notre méthode à la musique polyphonique sera discutée dans la conclusion.



Figure 2.1 : Exemples de partitions

Les musiques analysées peuvent être de n'importe quel genre (musique classique, jazz, etc.), si elles utilisent la notation classique.

Enfin, nous réalisons la reconnaissance de tous les symboles musicaux qui sont essentiels à la restitution de la mélodie :

| | | | | | | | | | | | | | | | |
|-------|---------------|-------|-------|-----------------------------|-------|---------|--------------|-----------------|----------|-----------|--------|--------|-------------|-------------|-------------|
| | | | | | | | | | | | | | | | |
| ronde | blanche | noire | point | dièse | bémol | bécarré | appoggiature | barre de mesure | pause | 1/2 pause | sourir | sourir | 1/2 sourire | 1/4 sourire | 1/8 sourire |
| Note | Têtes de note | | | Altérations et appoggiature | | | | | Silences | | | | | | |

Figure 2.2 : Symboles analysés

Tous les autres signes, tels que les signes d'ornement, d'attaque, de phrasé, texte, doivent être

ignorés. Exception a été faite pour les appoggiatures, car leur intégration dans l'analyse a permis de réduire les confusions faites sur ces symboles. La clé est pour l'instant donnée en paramètre d'entrée du programme.

Le programme de reconnaissance produit deux fichiers : un fichier MIDI (Musical Instrument Digital Interface) qui permet de jouer automatiquement la musique reconnue, par l'ordinateur ou par un instrument MIDI, et un fichier texte, qui contient la liste des symboles reconnus, avec leur position dans l'image. Ce dernier nous permet de vérifier les résultats de reconnaissance et de calculer des taux d'erreurs. Le fichier MIDI permet la réédition de la partition, via un éditeur du commerce qui accepte ce format en entrée, mais il faut souligner que beaucoup d'informations ont été perdues. En effet, la norme MIDI est fondée sur l'encodage des événements élémentaires (appui d'une touche pour produire le son ou relâche pour le stopper) et des intervalles de temps séparant deux événements. Beaucoup d'informations, en particulier la mise en page, doivent donc être devinées par l'éditeur. Pour une réédition de la partition proche de l'original, il faudrait sauvegarder la partition reconnue dans un format tel que le NIFF, qui encode explicitement tous les symboles musicaux et optionnellement des informations image. Cet outil n'a pas été développé, car nous n'en avons pas utilisé dans notre étude, mais cela pourrait être réalisé sans difficulté.

2.2. Acquisition et format des images

Les partitions sources sont imprimées (non manuscrites) et de format standard A4. L'acquisition des images est faite au moyen d'un scanner, avec une résolution de 300 dpi, communément adoptée dans la littérature. Cette qualité est suffisante pour la reconnaissance des symboles, même des plus petits comme les points de durée. Une définition supérieure ne conduirait qu'à augmenter la taille des images en mémoire, sans pour autant aider à résoudre les problèmes majeurs auxquels nous sommes confrontés : les défauts d'impression de la partition originale, la variabilité des fontes de symboles.

Pour constituer la base de données, trois scanners différents ont été utilisés. L'image obtenue $I_0(x,y)$ est binaire, les pixels noirs valant 1, les pixels blancs valant 0. La numérisation a été faite avec soin (document original correctement placé sur la vitre du scanner, options d'acquisition appropriées). Aucun prétraitement n'a ensuite été appliqué pour améliorer la qualité de l'image.

La base de données comprend plus de 100 pages de musique, extraites d'une soixantaine de morceaux, de genres, compositeurs, et surtout éditeurs différents (plus de 25). Elle inclut des partitions de difficultés variables. Les documents originaux sont globalement de bonne qualité, sans dégradations majeures. Ils peuvent cependant présenter les défauts d'impression ou de mise en page caractéristiques de l'édition musicale (symboles connectés, ruptures de segment, bruit, espacements entre symboles inhabituels, etc.). Au total, la base comprend 1191 portées, soit plus de 48000 symboles (Figure 2.2) à reconnaître.

Nous avons donc pris soin d'être le plus général possible, tant au niveau des sources que des

moyens d'acquisition. Ainsi, cette base de données importante et variée nous permettra d'évaluer des taux de reconnaissance significatifs.

2.3. Présentation générale du système

Le système de reconnaissance prend en paramètres d'entrée le fichier graphique contenant l'image (format GIF), ainsi que des informations globales : la clé, la métrique et la tonalité. Il est divisé en trois parties :

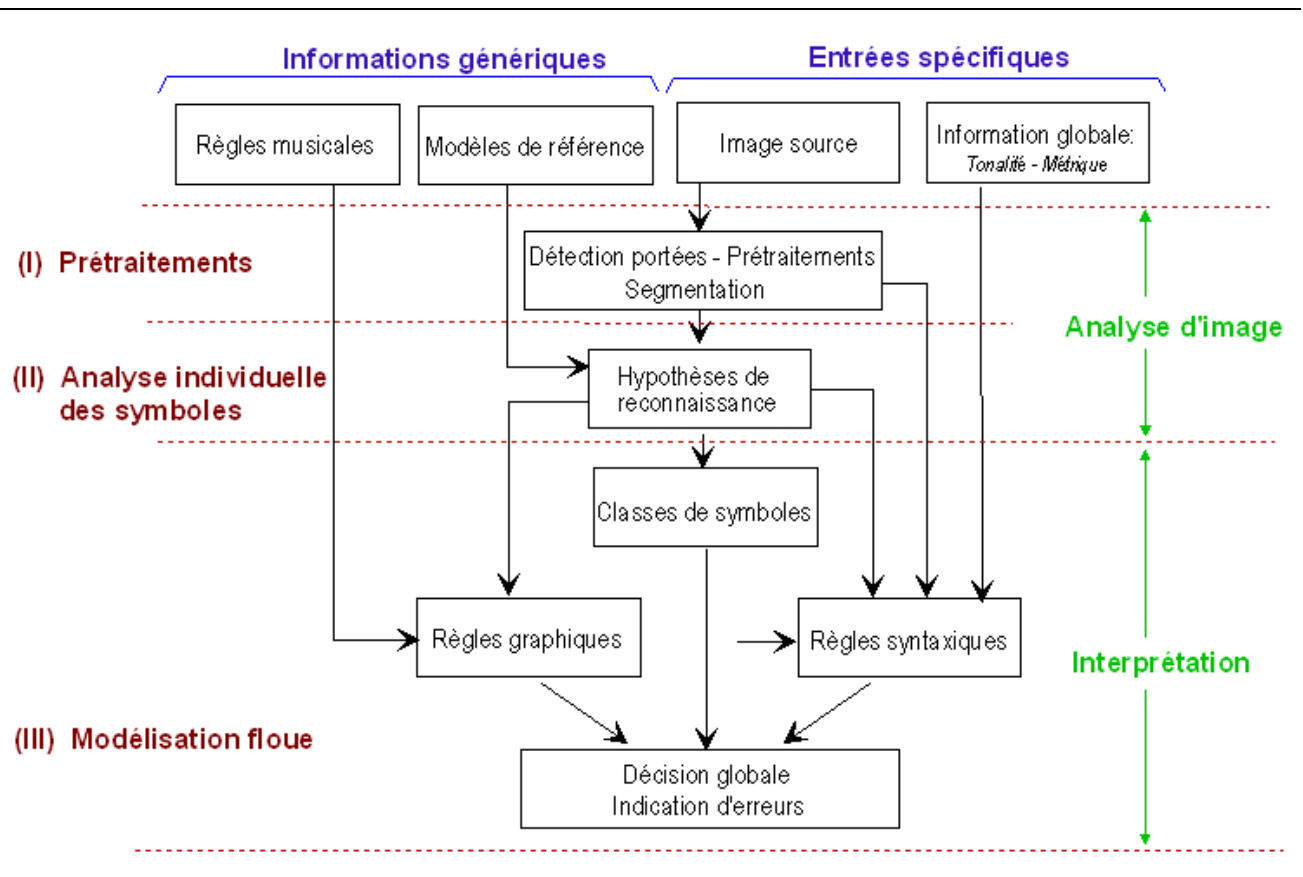


Figure 2.3 : Structure générale du système de reconnaissance

La première étape (prétraitements) réalise la mesure et la correction du biais de l'image, la détection précise des portées et des lignes de portée. L'image est ensuite segmentée.

Les objets segmentés sont analysés par corrélation avec des modèles de référence (Figure 2.2). Cette deuxième étape, dite d'analyse individuelle des symboles, aboutit à des hypothèses de reconnaissance, trois au plus par objet détecté. Une hypothèse de reconnaissance est l'attribution d'une classe (note, soupir, etc.) à l'objet. Dans certains cas, on laisse également la possibilité que cet objet ne soit pas un symbole musical : le nombre total d'hypothèses peut alors atteindre quatre.

La théorie des ensembles flous et des possibilités [Dubois, Prade 80] permet, dans une troisième étape, de combiner les informations de position et de corrélation fournies par l'étape précédente, de modéliser et d'intégrer les règles d'écriture de la musique. On définit ainsi pour

chaque hypothèse de reconnaissance un degré de possibilité d'appartenance à la classe, et des degrés de compatibilité graphique et syntaxique entre les objets. La modélisation des classes par des sous-ensembles flous permet de prendre en compte la variabilité des fontes de symboles, ainsi que les défauts d'impression et de segmentation. Les coefficients de compatibilité expriment les principales règles de la notation musicale, en modélisant les relations graphiques, structurelles et syntaxiques qui existent entre les différents symboles : par exemple, la position d'une altération par rapport à la note qu'elle altère, les méthodes de regroupement de croches par des barres de groupe, la cohérence des altérations par rapport à la tonalité du morceau. La modélisation de ces règles, sous la forme de relations floues entre symboles, permet de prendre en compte l'imprécision et l'incertitude qui existent au niveau de la position des objets (règles graphiques), ainsi que la souplesse de l'écriture musicale (règles syntaxiques).

La décision est ensuite prise. Elle doit être exprimée comme une optimisation globale de tous les critères. Plutôt que de réaliser cette optimisation sur toute la partition, nous préférons procéder mesure par mesure. Cette division en sous-problèmes est naturelle, car elle correspond parfaitement à la structure de l'écriture musicale, tant au niveau de la décomposition de la mélodie que de l'application des règles de musique. Elle nous permet de réduire la complexité de l'algorithme. Celui-ci teste donc mesure par mesure toutes les configurations d'hypothèses, en fusionnant tous les degrés de possibilité et coefficients de compatibilité, et retient la plus cohérente par maximisation. Enfin, les résultats de la modélisation floue, qui avaient été obtenus pour cette configuration, sont utilisés de manière à indiquer les erreurs potentielles. Cela permet de faciliter la vérification et la correction manuelle du résultat de reconnaissance.

Le dernier point, qui n'est pas représenté dans la figure 2.3 pour plus de lisibilité, est optionnel. Il concerne les procédures d'adaptation du système de reconnaissance à la partition traitée, grâce à un cycle de reconnaissance/correction effectué par l'utilisateur. Cet apprentissage supervisé permet d'affiner les modèles de référence et certains paramètres liés à ces modèles. L'objectif est d'améliorer la robustesse sur le reste de la partition.

Les prétraitements et la segmentation de l'image sont exposés au chapitre 3 ; ils conduisent à l'étape d'analyse individuelle des symboles, décrite au chapitre 4 ; la modélisation floue et la décision sont présentées au chapitre 5. Le chapitre 6 traite des améliorations permettant d'accroître la robustesse : l'indication des erreurs potentielles et les méthodes d'adaptation du système à la partition.

2.4. Discussion

Sans entrer dans les détails de chaque étape du système de reconnaissance, nous pouvons d'ores et déjà indiquer les aspects novateurs de cette architecture.

Notre approche suit une logique comparable à celle de nombreux auteurs [Blostein, Baird 92], et classique en analyse d'image, puisqu'il s'agit d'un processus séquentiel réalisant prétraitements, segmentation, et analyse. Néanmoins, elle nous permet de prendre en considération et de traiter les difficultés mentionnées au chapitre précédent.

L'ambiguïté, qui est due à la variabilité des polices de symboles, aux défauts d'impression, aux défauts de segmentation, est prise en compte, car l'étape d'analyse individuelle des symboles n'aboutit pas à une décision mais à un ensemble d'hypothèses. Ce n'est qu'après introduction du contexte, formalisé sous la forme de relations floues entre symboles, que la décision sera prise. En ce sens, la méthode est similaire à celle de [Fahmy, Blostein 98], puisqu'elle prend une décision sur des hypothèses précédemment générées, les deux phases, génération d'hypothèses et décision, étant bien distinctes et réalisées l'une après l'autre.

Il est intéressant de discuter ce type de modèle par rapport à d'autres architectures permettant de gérer l'ambiguïté. Les approches de type [Kato, Inokuchi, 92] ou encore [Stückelberg et al. 97] [McPherson, Bainbridge 01] [McPherson 02], sont, au contraire de la nôtre, bidirectionnelles. C'est-à-dire que les différentes étapes du processus de reconnaissance, du plus bas niveau (extraction des primitives) au plus haut niveau (analyse contextuelle), communiquent également dans le sens descendant, pour orienter ou contraindre les tâches de bas niveau en fonction de l'information recherchée. L'inconvénient de ces méthodes est qu'elles doivent mettre en œuvre un processus complexe d'ordonnement des tâches à effectuer. De plus, il n'est pas certain que ce processus puisse prendre en compte toute l'information contextuelle, si celle-ci n'est pas encore disponible, et qu'il n'y ait pas un risque de propagation d'erreurs, si l'on oriente le processus d'analyse en fonction de résultats (hypothèses ou décisions) erronés. Au contraire, notre méthode permet de prendre une décision globale, avec une méthodologie simple : parcourir toutes les configurations d'hypothèses. Si la solution est dans cet espace, alors elle peut être trouvée par optimisation simultanée de tous les critères. Lors de l'analyse individuelle, nous choisissons des seuils de corrélation bas pour accepter une hypothèse, et nous autorisons en cas de forte ambiguïté jusqu'à quatre hypothèses simultanées pour chaque objet, de telle sorte qu'il est très rare que la bonne solution soit absente de l'ensemble des hypothèses. Il y a certes un risque d'explosion combinatoire, à cause de ces seuils bas, et parce que toutes les hypothèses sont générées en aveugle, c'est-à-dire de manière complètement indépendante du contexte. Néanmoins, en divisant le problème en sous-problèmes (la mesure), l'expérience montre que l'on reste dans des limites possibles. Par ailleurs, on peut trouver des heuristiques qui permettent de réduire le coût de calcul, notamment en évitant de tester des configurations que l'on sait, grâce aux précédents tests, impossibles. De plus, on peut s'appuyer sur la notion de mesure, car la détection des barres de mesure est très fiable.

La décomposition du processus de reconnaissance en trois étapes distinctes, analyse individuelle des symboles, modélisation floue et décision, présente deux autres avantages.

Le premier est qu'elle permet d'adapter le processus de reconnaissance à la partition traitée. En effet, les paramètres qui définissent les sous-ensembles flous modélisant les classes de symboles sont appris à partir des résultats de corrélation, qui ont été obtenus sur toute la partition durant la phase d'analyse individuelle, de sorte que le modèle s'adapte. Le problème de la variabilité des polices peut être ainsi traité.

Le second est qu'elle permet de structurer la modélisation des règles d'écriture musicale de manière rigoureuse, évitant de les disséminer un peu partout dans la méthode, contrairement à ce qui a souvent été fait dans les systèmes présentés dans la littérature. La connaissance a priori concernant les symboles, chacun indépendamment des autres, est intégrée dans la phase d'analyse

individuelle : par exemple, le fait qu'une barre de mesure est nécessairement entre la première et la cinquième ligne de portée, que les notes sont sur les lignes de portée ou dans les interlignes. En revanche, toutes les règles qui expriment des interactions entre symboles sont introduites dans la deuxième phase, un module gérant les règles graphiques, un autre gérant les règles syntaxiques. La formalisation, fondée sur la théorie des ensembles flous et des possibilités, permet de modéliser et de fusionner ces informations très hétérogènes [Dubois et al. 99], par conséquent de prendre une décision globale, et c'est aussi l'un des aspects novateurs de notre méthodologie.

La méthode de reconnaissance proposée est unidirectionnelle, comme nous venons de le préciser. Néanmoins, les procédures proposées pour gagner en robustesse introduisent dans une certaine mesure une rétroaction : tout d'abord, au niveau de l'indication automatique des erreurs potentielles, puisque les résultats obtenus sur les symboles finalement retenus sont réexaminés dans ce but, mais sans remise en cause de la décision ; ensuite, de manière plus évidente, dans la méthode (optionnelle) d'apprentissage d'une partition : des modèles sont appris sur un extrait puis introduits dans le programme pour la reconnaissance du reste de la partition. Néanmoins, il ne s'agit que d'ajustements de paramètres internes, la méthodologie de reconnaissance restant identique et fondamentalement unidirectionnelle. Ces deux points, qui n'ont à notre connaissance par encore été abordés dans la littérature, constituent des idées innovantes permettant d'améliorer considérablement les performances d'un système d'OMR.

CHAPITRE 3

Prétraitements et segmentation

L'image en entrée, notée I_0 , est binaire, $I_0(x,y)$ au point de coordonnées (x,y) prenant les valeurs 0 (pixel blanc) ou 1 (pixel noir correspondant à l'impression). On considère un système de coordonnées dont l'origine est le coin en haut à gauche de l'image, l'axe des x vertical et orienté vers le bas, l'axe des y horizontal et orienté vers la droite. L'image a une largeur de W pixels, et une hauteur de H pixels (typiquement $W = 2400$ pixels et $H = 3400$ pixels pour une partition de format A4). Ainsi :

$$I_0(x,y) \in \{0,1\}, 0 \leq x < H, 0 \leq y < W \quad (\text{Eq 3.1})$$

Les prétraitements permettent de corriger l'inclinaison de l'image et de déterminer la position des lignes de portée. L'image est ensuite segmentée de sorte que les symboles de la partition puissent être analysés (chapitre 4).

3.1. Prétraitements

Comme nous l'avons détaillé dans la section 1.3.2, la détection des lignes de portée est une étape fondamentale, car elle permet de déduire des paramètres essentiels :

- Les lignes de portée devant être horizontales, leur détection permet de calculer l'inclinaison de l'image et de la redresser.
- L'espace entre les lignes de portée, appelé interligne, calculé en nombre de pixels, indique l'échelle de l'image et permet de normaliser les distances et les longueurs.
- Les symboles musicaux sont positionnés relativement aux lignes de portée. La détection des portées est donc une étape préliminaire à leur localisation et elle est nécessaire à leur interprétation.

Nous procédons en trois étapes : la première consiste à détecter le biais de l'image et à le corriger. Les portées sont ensuite globalement localisées. Enfin, un algorithme de poursuite de portée permet de connaître précisément la position de chaque portée en chaque ordonnée.

Deux images de test, données en figures 3.1 et 3.2, nous permettront d'illustrer les différentes étapes de la méthode. Toutes deux sont assez denses, en particulier l'image test 2 d'une partition polyphonique, avec de nombreuses barres de groupe horizontales qui se superposent aux

lignes de portée et les occultent donc partiellement. Elles présentent en outre un biais important et des courbures locales.

48 *mf* *cresc.* - - - - - *f*

55 *Tutti*

66 *Solo* 3 *mp*

77 *p*₃ *mp*₃

83 *mf*₃ *p*

89 *Tutti* *f* *f*

98 *Solo* *mf*

110 *p* *cresc.* - -

118 *f*

125 *p*

131 *cresc.* - - - - - *mf*

137 *f*

264

Figure 3.1 : Image test 1 (monodique)

220

Fl. 1 3 5

Ob.

Cl.

Fag.

Hbl.

Str. *pp*

*pp*₂

cresc.

cresc.

Tutti

222 II

f

f *Tutti*

225

f *Tutti*

229

233

cresc.

ff

Edition Breitkopf 27534

Figure 3.2 : Image test 2 (polyphonique)

3.1.1. Redressement de l'image

Nous proposons dans un premier temps de calculer l'angle d'inclinaison de la page de musique. Pour cela, nous pouvons exploiter la similitude qui existe au niveau des lignes de portée, supposées rectilignes, entre la moitié gauche et la moitié droite de l'image. Les lignes de portée des deux sous-images sont superposées pour un décalage h_{max} de la moitié droite par rapport à la moitié gauche (Figure 3.3). La valeur de h_{max} , qui peut être obtenue par simple calcul de corrélation, nous permet de déduire l'angle d'inclinaison :

$$\theta = \arctan\left(\frac{2h_{max}}{W}\right) \quad (\text{Eq. 3.2})$$

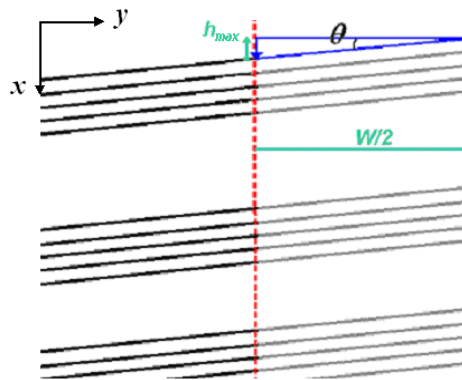


Figure 3.3 : Principe de la détection de l'inclinaison de l'image

Dans cette étape, et pour toute la suite, nous définissons la corrélation entre deux images I_1 et I_2 , de taille $W*H$, par :

$$C = \frac{1}{W.H} \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} I'_1(x,y).I'_2(x,y) \quad (\text{Eq. 3.3})$$

avec :

$$I'_i(x,y) = \begin{cases} -1 & \text{si } I_i(x,y) = 0 \\ 1 & \text{si } I_i(x,y) = 1 \end{cases}, \quad 0 \leq x < H, \quad 0 \leq y < W, \quad i = 1,2 \quad (\text{Eq. 3.4})$$

Il s'agit donc d'une corrélation normalisée entre -1 et 1, la valeur maximale étant obtenue pour deux images identiques, la valeur minimale lorsque I_2 est le négatif de I_1 . Cette définition donne le même poids aux pixels du fond qu'aux pixels objet. Appliquons cette définition pour calculer la corrélation entre les deux moitiés de l'image :

$$C(h) = \frac{2}{W.H} \sum_{x=0}^{H-1} \sum_{y=0}^{W/2-1} I'_0(x,y).I'_0(x+h,y+W/2) \quad (\text{Eq. 3.5})$$

La figure 3.4 présente les résultats de corrélation obtenus sur nos deux images de test. Le pic principal de corrélation correspond effectivement au décalage h_{max} recherché. On remarque en

outre la présence d'un deuxième maximum local, qui est dû à la périodicité des lignes de portée : l'écart entre le pic principal et le pic secondaire correspond à l'interligne.

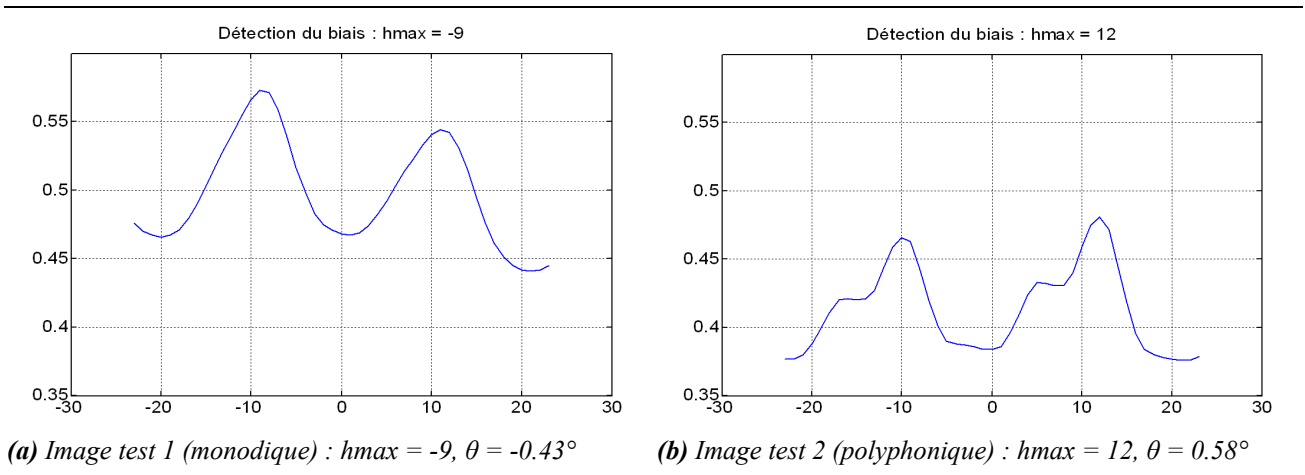


Figure 3.4 : Exemples de calcul de l'inclinaison

Cette méthode, qui n'avait à notre connaissance pas encore été mise en œuvre, suppose que les lignes de portée sont globalement rectilignes, ce qui est toujours le cas, en pratique, sur un document original. Les distorsions les plus marquées sont généralement introduites lors de la numérisation, lorsque le document n'est pas correctement aplati sur la vitre, au niveau de la reliure par exemple. Les deux images de test, qui ont été volontairement scannées avec négligence, présentent de tels défauts. Nous verrons que la méthode de détection des portées donne cependant de bons résultats. En particulier, le calcul préalable de l'inclinaison globale de la page (Figure 3.4) est exact.

Lorsque l'angle obtenu est différent de zéro, l'étape suivante consiste à corriger l'inclinaison de l'image. Une méthode rigoureuse consisterait à réaliser une rotation d'un angle $-\theta$. Nous supposons ici que l'image a été scannée avec une inclinaison inférieure à 1° . Typiquement, cela représente un décalage $2h_{max} < 40$ pixels entre le bord gauche et le bord droit d'une partition A4, ce qui est en pratique tout à fait réalisable et non contraignant. Sous cette hypothèse, on peut se contenter de corriger l'inclinaison de la page par simple décalage vertical des pixels d'une quantité proportionnelle à leur coordonnée horizontale y . Formellement, la transformation utilisée est la suivante :

$$I(x, y) = I_0(x - 2h_{max}y/W, y) \quad (\text{Eq. 3.6})$$

En effet, la rotation des symboles musicaux, qui ont une largeur typique de l'ordre de 1 interligne (environ 20 pixels) et une hauteur inférieure à 4 interlignes (80 pixels), est négligeable. Pour le vérifier, prenons l'exemple critique d'un segment vertical de hauteur 80 pixels : le décalage horizontal introduit par le biais entre les deux extrémités est inférieur à 1,5 pixels, ce qui est du même ordre de grandeur que les distorsions locales que l'on peut trouver dans la partition originale. La rotation n'est donc pas perceptible localement, à l'échelle du symbole, et nous nous contentons, par cette transformation simple, de restituer l'horizontalité globale de la partition. Les expérimentations menées sur la base de données montrent qu'en effet les algorithmes présentés par

la suite ne sont pas sensibles à la faible distorsion induite.

Conclusion

Dans la littérature, le biais n'est pas toujours corrigé : soit parce que les auteurs le supposent négligeable, y compris pour la détection des lignes de portée, soit parce que ces dernières sont localisées précisément sur toute leur longueur et que la suite de l'analyse prendra en compte les fluctuations détectées. Nous avons exposé ces différents points de vue au chapitre 1.3.2. Notre méthode débute au contraire par la détection et la correction de l'angle, ce qui nous permet de localiser ensuite les portées par projection. Elle se rapproche à cet égard de celle proposée par [Martin 92], fondée sur la maximisation des cordes. Bien que notre technique soit plus simple, nous avons constaté qu'elle fonctionne très bien. Notre résultat est en effet très fiable, car le calcul du biais provient d'une analyse globale de toute l'image. Pour la restitution de l'horizontalité, nous avons proposé d'appliquer un simple décalage vertical des colonnes image, contrairement à de nombreux auteurs, qui réalisent une rotation (par exemple [Sicard 92] [Ng, Boyle 92] [Martin 92] [Wijaya, Bainbridge 99]). Notre méthode est moins coûteuse en calcul, la rotation nécessitant une interpolation, qui peut par ailleurs ajouter des défauts. Elle s'est avérée très satisfaisante dans nos expérimentations, pour des partitions qui ont été scannées avec soin, mais sans contraintes excessives.

3.1.2. Détection et caractérisation des portées

Pour localiser les portées, il faut extraire les abscisses des cinq lignes qui les composent. On en déduit les lignes de séparation des portées, qui pourront alors être extraites et traitées séquentiellement. Notre méthode est fondée sur l'analyse de la projection horizontale de l'image redressée, avec calcul préalable de l'interligne.

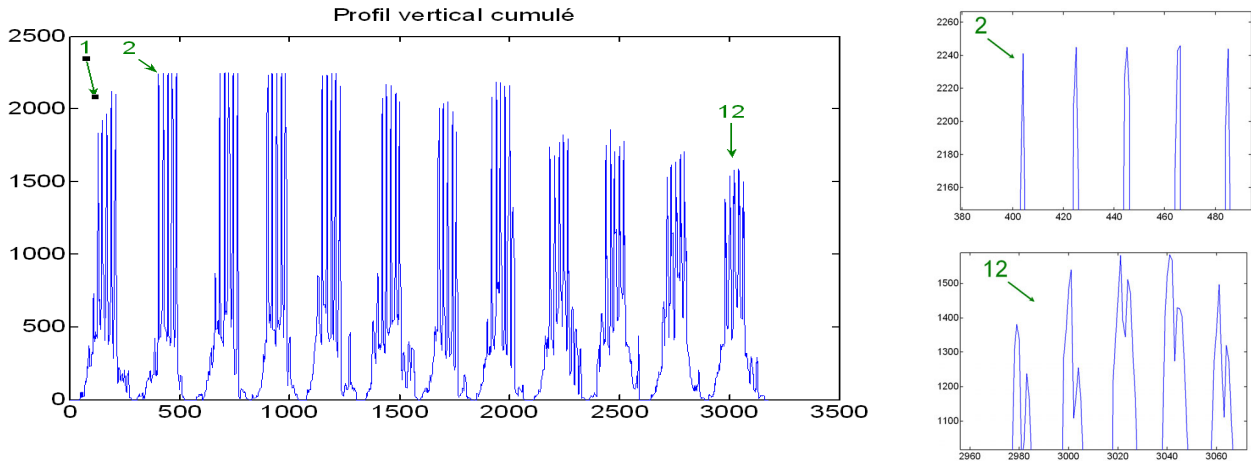
Calcul préalable de l'interligne

Comme l'inclinaison de l'image est préalablement corrigée, les lignes de portée sont globalement horizontales et nous pouvons calculer classiquement (e.g. [Fujinaga 88] [Martin 92]) le profil vertical pour les détecter (Figure 3.5) :

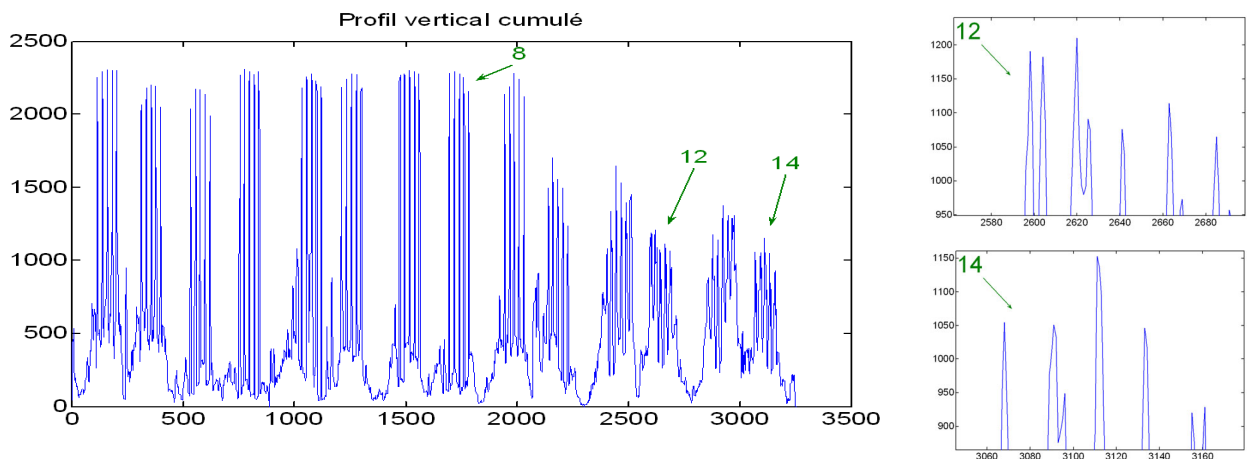
$$P_Y(x) = \sum_{y=0}^{W-1} I(x, y) \quad (\text{Eq. 3.7})$$

Le profil vertical cumulé, ou projection horizontale, calcule pour chaque ligne image la somme de tous ses pixels. Il met en évidence des groupes de 5 pics correspondant aux lignes de portée. La recherche de ces maxima n'est pas toujours aisée, car les pics peuvent être brouillés par des lignes horizontales additionnelles (barres de groupe horizontales, lignes au-dessus des mesures de renvoi), et par des courbures ou des biais locaux. La figure 3.5 présente les profils obtenus pour nos deux images tests. On constate que certains groupes sont effectivement très propres, avec des maxima qui correspondent à la longueur des lignes de portée, alors que pour d'autres portées, l'amplitude des

pics a diminué, leur épaisseur s'est élargie avec parfois l'apparition de maxima secondaires. C'est pourquoi nous préférons commencer par déduire du profil vertical l'interligne moyen, puis, connaissant ce paramètre fondamental, la position de chaque portée. La robustesse de la méthode est ainsi améliorée.



(a) Image test 1 (monodique) : à gauche, le profil vertical obtenu et à droite, un zoom sur les pics des portées 2 et 12. L'affaissement des pics de la portée 1 correspond à une courbure en début de portée, tandis que celui de la portée 12 correspond à un biais résiduel de toute la portée. Le dédoublement des pics s'explique par ce biais résiduel couplé au bruit interférent.



(b) Image test 2 (polyphonique) : les pics sont également nets pour les portées bien rectilignes et horizontales. L'affaissement des maxima et leur dédoublement correspond à un biais résiduel pour la portée 12, et à des ondulations pour la portée 14, couplés au bruit interférent.

Figure 3.5 : Exemples de profils verticaux cumulés sur les images redressées

La fonction d'autocorrélation du profil $P_Y(x)$, notée $R_{P_Y}(s)$, permet de mettre en évidence la périodicité des lignes de portée. La première valeur non nulle de s , notée s_I , qui maximise cette fonction d'autocorrélation, représente l'interligne. On peut en effet constater sur la figure 3.6 la présence d'un maximum bien net sur les deux exemples présentés.

$$R_{P_Y}(s) = \sum_x P_Y(x)P_Y(x+s) \quad (\text{Eq. 3.8})$$

$$R_{P_Y}(s_I) = \max_{s \neq 0} (R_{P_Y}(s)) \quad (\text{Eq. 3.9})$$

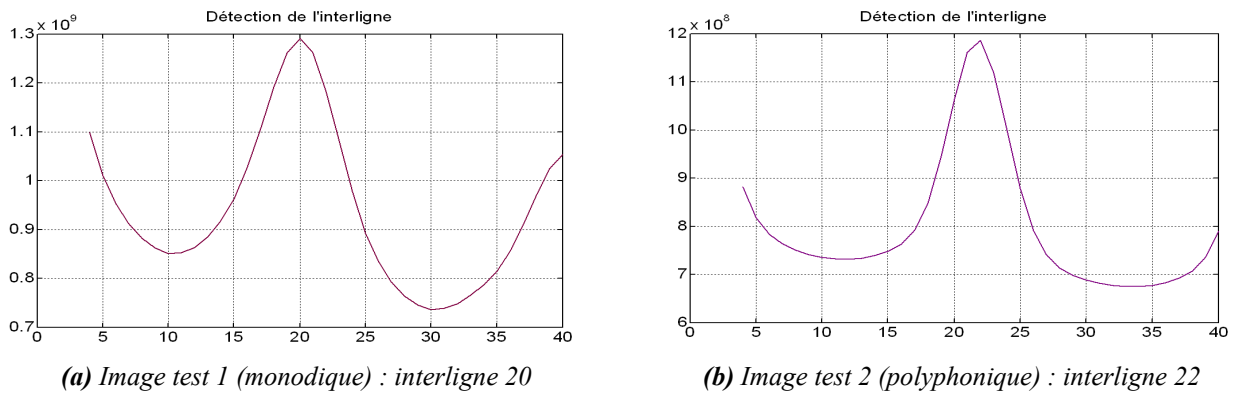


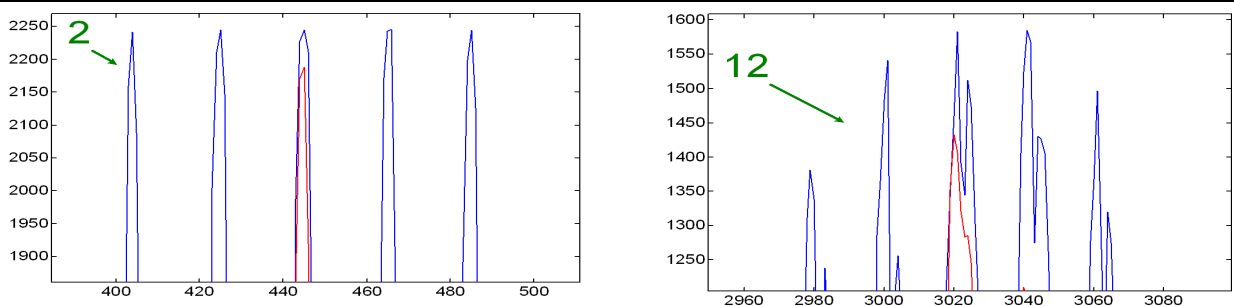
Figure 3.6 : Calcul de l'interligne moyen par autocorrélation

Détection des portées et calcul de leur position

La connaissance de l'interligne facilite beaucoup la recherche des groupes de cinq pics équidistants dans le profil. Il suffit en effet de trouver les maxima locaux de la fonction $F_{P_y}(x)$, qui correspondent à l'addition de cinq lignes de portée distantes de s_l pixels. Nous avons choisi d'effectuer le calcul sur une épaisseur de 3 pixels, ce qui permet d'être plus robuste par rapport aux petites variations de l'interligne, et de prendre en compte, au moins partiellement, l'épaisseur réelle des lignes de portée, qui est en moyenne toujours strictement supérieure à 1.

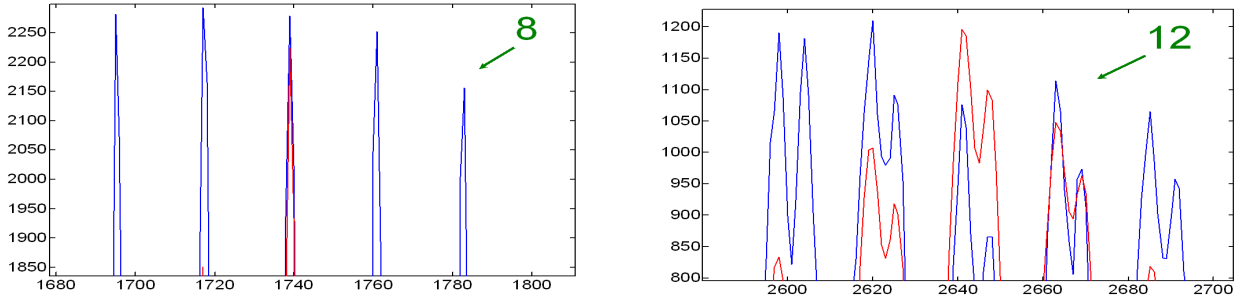
$$F_{P_y}(x) = \sum_{k=-2}^2 \sum_{i=-1}^1 P_y(x + k * s_l + i) \tag{Eq. 3.10}$$

Ces maxima locaux sont bien marqués même en présence de défauts, comme l'atteste la figure 3.7 ci-dessous. On observe donc effectivement une bonne robustesse de la méthode. Comme certaines portées présentent un biais résiduel, il faudra affiner ultérieurement ces résultats, sur chacune des portées individuellement. Dans la suite, on notera N_p le nombre de portées détectées, et $x^{(i)}$ l'ordonnée de la ligne centrale de la portée i ($1 \leq i \leq N_p$), maximum local de la fonction $F_{P_y}(x)$.



(a) Image test 1 (monodique) portée 2 : en bleu le profil vertical cumulé, en rouge la fonction $F_{P_y}(x)$ normalisée. Le maximum correspond exactement à la ligne centrale de la portée. (b) Image test 1 portée 12 : malgré la présence de pics secondaires, le maximum de la fonction $F_{P_y}(x)$ est bien marqué et correspond effectivement à la troisième ligne de portée.

Figure 3.7 (a)(b) : Détection des portées en musique monodique



(c)(d) Image test 2 (polyphonique) : les résultats sont exacts et similaires à ceux obtenus en (a) et (b), malgré la plus forte densité des symboles interférents et les imperfections des lignes de portée.

Figure 3.7 (c)(d) : Détection des portées en musique polyphonique

Calcul de l'épaisseur des lignes de portée

Nous reprenons, à l'instar d'autres auteurs (e.g. [Bellini et al. 01] [Miyao 02]), la méthode proposée par Kato et Inokuchi [Kato, Inokuchi 92], consistant à rechercher le maximum de l'histogramme des longueurs des empan noirs. Cependant, comme nous connaissons maintenant la position des portées, nous pouvons éviter de parcourir toute l'image et nous restreindre aux portées. Les zones utilisées pour le calcul de l'histogramme $H_n(e)$ sont centrées sur la troisième ligne de chaque portée, de hauteur δs_l , de largeur W . La figure 3.8 illustre les résultats obtenus sur les images de test.

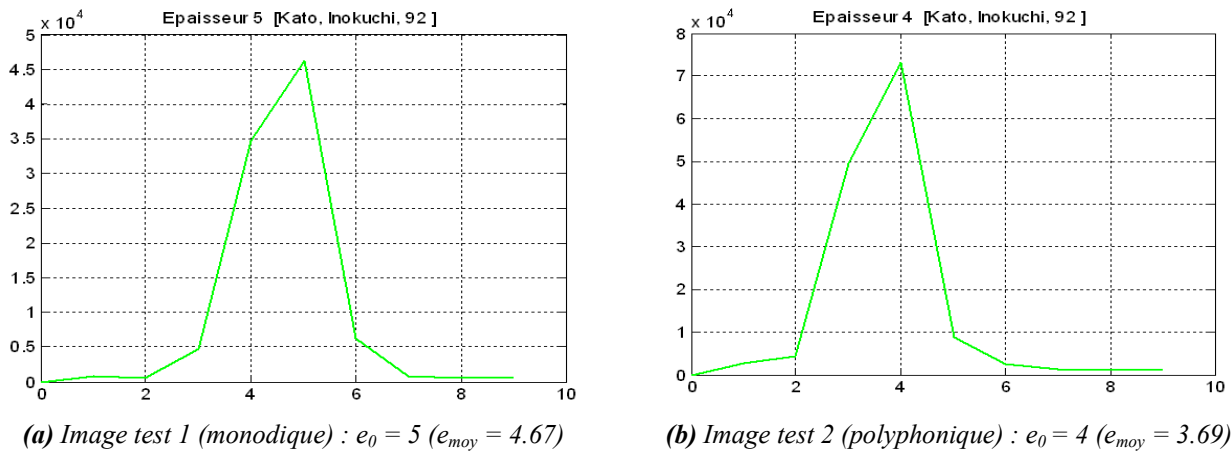


Figure 3.8: Histogramme des longueurs des empan noirs, méthode proposée par [Kato, Inokuchi 92] appliquée aux zones image centrées sur les portées

Soit e_0 l'index correspondant au maximum de l'histogramme $H_n(e)$. Cette valeur représente approximativement l'épaisseur des lignes de portée. On constate en pratique, au format d'image et à la résolution considérés, que l'épaisseur des lignes de portée varie en fait de ± 1 pixel. Une estimation de l'épaisseur moyenne des lignes de portée peut être obtenue par :

$$e_{moy} = \left(\sum_{e=e_0-1}^{e_0+1} e * H_n(e) \right) / \left(\sum_{e=e_0-1}^{e_0+1} H_n(e) \right) \quad (\text{Eq. 3.11})$$

Les paramètres d'épaisseur e_0 et e_{moy} caractérisent les lignes de portée et ils seront utilisés par la suite pour affiner leur détection et les effacer.

Extraction des sous-images correspondant chacune à une portée

Dans un premier temps, deux portée consécutives sont séparées par une droite horizontale, placée à égale distance des portées :

$$\begin{aligned} x_c^{(i)} &= \frac{x^{(i-1)} + x^{(i)}}{2} \text{ pour } 1 < i \leq Np \\ x_c^{(i)} &= 0 \text{ pour } i = 1 \end{aligned} \tag{Eq. 3.12}$$

Ce découpage simple ne fonctionne pas lorsque les portées sont très proches, avec des notes au-dessus et au-dessous de la portée, car, dans ce cas, certains symboles de la portée supérieure se retrouvent dans la portion image définie pour la portée inférieure, et vice-versa. C'est pourquoi nous étendons la zone image (Figure 3.9a) en prenant une marge de 2 interlignes au-dessus et au-dessous de la limite initiale. Soient $o_x^{(i)}$ l'origine de la sous-image extraite de l'image redressée I , et $H^{(i)}$ sa hauteur :

$$\begin{aligned} o_x^{(i)} &= x_c^{(i)} - 2s_l \text{ si } 1 \leq i < Np \\ &= 0 \text{ si } i = 1 \end{aligned} \tag{Eq. 3.13}$$

$$\begin{aligned} H^{(i)} &= x_c^{(i+1)} + 2s_l - o_x^{(i)} \text{ si } 1 \leq i < Np \\ &= H - o_x^{(i)} \text{ si } i = Np \end{aligned} \tag{Eq. 3.14}$$

L'image de la $i^{\text{ème}}$ portée, notée $I^{(i)}$ (Figure 3.9b), peut être maintenant extraite de l'image redressée I . Elle est tout simplement définie par :

$$I^{(i)}(x, y) = I(x + o_x^{(i)}, y), \quad 0 \leq x < H^{(i)}, \quad 0 \leq y < W \tag{Eq. 3.15}$$

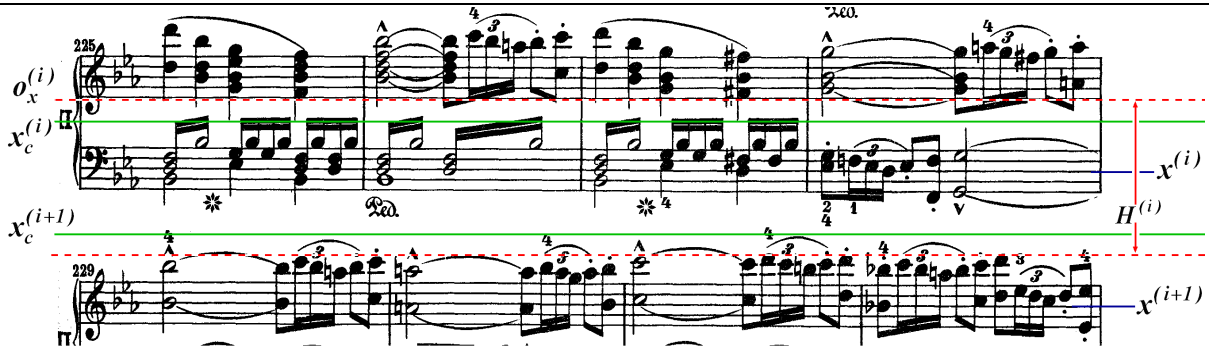
Enfin, la nouvelle abscisse de la troisième ligne de portée dans la sous-image extraite s'obtient par :

$$x_p^{(i)} = x^{(i)} - o_x^{(i)} \tag{Eq. 3.16}$$

Afin de supprimer les portions de symboles qui proviennent des portées adjacentes, nous appliquons un algorithme qui détecte tous les pixels noirs des bords, et, par croissance de région, efface tous les objets limitrophes dans la limite de la marge ajoutée. Ce procédé est illustré par les figures 3.9b et 3.9c. On remarque que la portée est correctement extraite. Il reste quelques objets qui n'appartiennent pas à la portée, mais aucun symbole ou fragment de symbole à reconnaître. En particulier, les fragments de segments verticaux, dont la détection sert à la segmentation de l'image, comme nous le verrons dans le paragraphe suivant, sont correctement éliminés.

Dans toute la suite, nous traiterons les portées individuellement. Le redressement est tout d'abord affiné, par réapplication de la méthode précédemment exposée (Equations 3.5 et 3.6) sur

l'image $I^{(i)}$, avec cependant un paramétrage adapté car l'inclinaison résiduelle à corriger est plus faible. De même, on affine la position $x_p^{(i)}$ de la troisième ligne de portée, en recalculant le profil sur l'image $I^{(i)}$ redressée (Equations 3.7 et 3.10), la valeur de l'interligne moyen s_l étant inchangée.



(a) Détermination des limites de la portée $i=10$ de l'image test 2 : en vert, les limites fixées par l'équation 3.12, en pointillés rouges les limites étendues de 2 interlignes.



(b) Sous-image $I^{(i)}$ extraite à partir de l'origine $o_x^{(i)}$ sur une hauteur $H^{(i)}$.



(c) Sous-image extraite $I^{(i)}$ après effacement des objets limitrophes.

Figure 3.9 : Extraction des portées

Calcul des ordonnées de début et de fin de portée

On réalise une projection verticale des zones de l'image situées autour de chaque ligne de la portée. A y fixé, la somme des pixels image correspondant aux lignes de portée est environ égale à $5e_{moy}$. On évalue cette somme, notée $Proj_l$, suivant l'équation 3.17, en considérant plusieurs décalages verticaux δx_p autour de la position moyenne de la portée, et on recherche la première ordonnée $y_d^{(i)}$ telle que $Proj_l$ soit toujours supérieure à $S=0.5(5e_{moy})$ sur une largeur s_l (Eq. 3.18) :

$$Proj_l(y) = \max_{\delta x_p} \sum_{k=-2}^{k=2} \sum_{\delta x = -\Delta x}^{\Delta x} I(x_p^{(i)} + ks_l + \delta x + \delta x_p, y),$$

$$\text{avec } \begin{cases} \Delta_x = E\left(\frac{e_0}{2}\right) + 1 \\ -\frac{s_l}{2} \leq \delta x_p \leq \frac{s_l}{2} \end{cases} \quad (\text{Eq. 3.17})$$

$$\forall y < y_d^{(i)}, \exists j \in [0, s_l[/ Proj_l(y - j) < 2.5e_{moy} \quad (\text{Eq. 3.18})$$

Dans l'équation 3.17, $E(x)$ désigne la partie entière de x . La plage de variation de δx_p est choisie suffisamment large ($\pm s_l/2$) pour pallier les courbures locales de l'image. Le même principe est appliqué pour déterminer la fin de la portée à l'ordonnée $y_f^{(i)}$.

La figure 3.10 illustre la méthode sur la première portée de l'image test 1. On constate que le résultat est précis. Pour des portées présentant des accolades, comme l'image test 2 (Figure 3.2), il peut y avoir une petite erreur, puisque le critère (Eq. 3.18) peut être satisfait au niveau de l'accolade. Mais cela ne pose pas de difficultés pour la suite de l'analyse, qui ne nécessite pas une grande précision sur ce paramètre.

A ce stade de l'analyse, on dispose donc de tous les paramètres qui définissent les portées dans chacune des sous-images $I^{(i)}$ ($1 \leq i \leq Np$) extraites : interligne s_l et épaisseur moyenne e_{moy} (calculés sur l'ensemble de l'image), abscisse de la troisième ligne de portée $x_p^{(i)}$, ordonnées de début et de fin de portée $y_d^{(i)}$ et $y_f^{(i)}$.

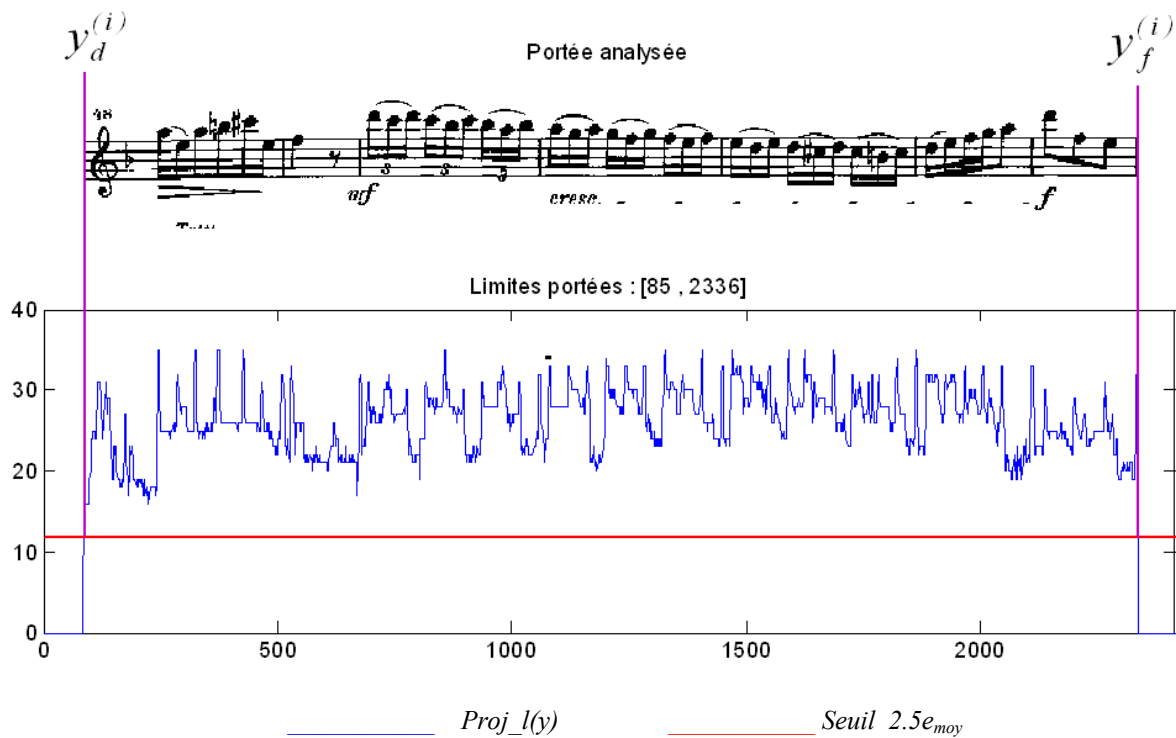


Figure 3.10 : Détection des limites de la $i^{\text{ème}}$ portée par projection verticale.

Conclusion

Nous avons exposé dans ce paragraphe une méthode qui permet de localiser les lignes de portée et d'extraire les sous-images correspondant chacune à une portée. Cette méthode est largement fondée sur les projections dans les deux directions, à l'instar de nombreux auteurs ([Kato, Inokuchi 92], [Sicard 92], [Bellini et al. 01] par exemple). Le calcul préalable de l'interligne permet, malgré les défauts résiduels, de localiser les portées sans ambiguïté à partir du profil vertical de l'image redressée. Les solutions proposées pour le calcul de l'interligne et la détection de la troisième ligne de portée sont très simples dans leur principe, mais n'avaient, à notre connaissance,

pas encore été appliquées. Le taux de réussite est de 100% sur notre base d'images, c'est-à-dire que toutes les lignes de portée ont été correctement localisées.

Il est intéressant de confronter notre méthode de détermination de l'interligne et de l'épaisseur des lignes avec celle proposée par [Kato, Inokuchi 92], qui est fondée sur la recherche d'un maximum dans l'histogramme des longueurs des empanns noirs et blancs. La figure 3.11 montre les histogrammes obtenus sur les images *test 1* et *test 2*, en analysant dans les deux cas l'intégralité de l'image.

En comparant les figures 3.8 et 3.11, on constate que les résultats obtenus pour l'épaisseur (entière) e_0 des lignes sont identiques, et cela est absolument normal puisque nous avons appliqué la méthode proposée par [Kato, Inokuchi 92]. L'unique différence réside dans le choix de la zone d'analyse, qui a été dans notre cas restreinte aux portées. On peut supposer que la précision obtenue en est généralement accrue, car les symboles fins hors portée (liaisons, lignes au-dessus des mesures de renvoi, paroles de chanson, etc.) interfèrent moins dans les mesures.

On constate par ailleurs que la somme de l'interligne (longueur des empanns blancs les plus fréquents) et de l'épaisseur des lignes (longueur des empanns noirs les plus fréquents) trouvés par la méthode [Kato, Inokuchi 92] (Figure 3.11) est égale à l'interligne obtenu avec notre méthode (Figure 3.6). Les résultats sont identiques pour 85% des images de notre base, et varient de +/- 1 pixel pour les 15% restant. Les taux de reconnaissance finals sont également similaires, que l'on applique l'une ou l'autre technique. La méthode de calcul de l'interligne proposée est donc intéressante lorsque la détection des lignes de portée se fonde sur le profil vertical, car elle ne nécessite alors qu'une dizaine de multiplications/additions (Eq. 3.8).

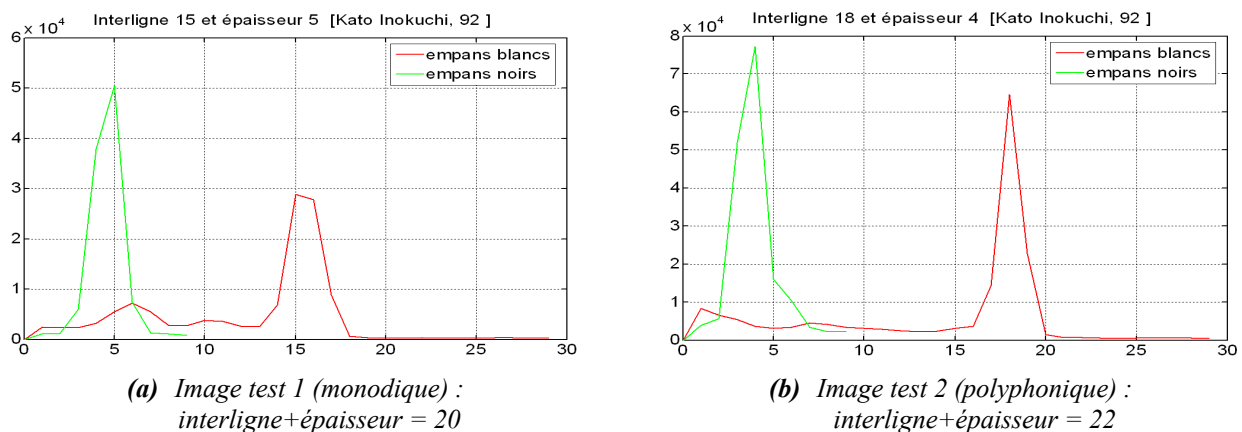


Figure 3.11 : Calcul de l'interligne et de l'épaisseur des lignes de portée, par la méthode proposée par [Kato, Inokuchi 92], appliquée sur l'intégralité de l'image

3.1.3. Poursuite des portées

Les portées ont donc été extraites de l'image source, leur biais a été corrigé, et l'interligne, l'épaisseur et la position moyenne des lignes de portée sont précisément connus. L'interligne et l'épaisseur des lignes de portée sont des paramètres dont on peut négliger les très faibles variations locales. En revanche, il arrive fréquemment que les lignes de portée soient gondolées, c'est-à-dire

que l'abscisse de la troisième ligne de portée s'éloigne localement de la position $x_p^{(i)}$ déterminée précédemment.

Pour prendre en compte les courbures résiduelles, nous appliquons un algorithme de « poursuite de portée » qui opère sur les images $I^{(i)}$ de l'extrémité gauche de la portée vers l'extrémité droite. Cet algorithme est fondé sur le calcul, en chaque ordonnée y , de la corrélation entre l'image $I^{(i)}$ et une colonne de pixels représentant la coupe d'une portée. Une corrélation simple est insuffisante à cause des symboles musicaux qui brouillent localement les résultats. C'est pourquoi nous utilisons une technique de filtrage avec facteur d'oubli, qui permet d'intégrer de manière continue les résultats de corrélation obtenus aux précédentes ordonnées. Ainsi, le résultat n'est pas sensible aux symboles interférents et il ne met en évidence que des variations lentes de la position de la portée. Dans la suite, nous appellerons masque de corrélation la colonne de pixels utilisée pour le filtrage.

Trois phases sont nécessaires à la mise en place de l'algorithme : la définition du masque, l'initialisation du filtre, la poursuite proprement dite.

Masque de corrélation en fonction de l'épaisseur des lignes

Ce masque de corrélation (Figure 3.12, Eq. 3.19), noté M_p , représente la coupe verticale d'une portée, dont les lignes ont une épaisseur e_0 et sont espacées de l'interligne s_l . Sa taille est $H_p=2 * E(2.5s_l)$, c'est-à-dire qu'elle est légèrement supérieure à la hauteur de la portée.

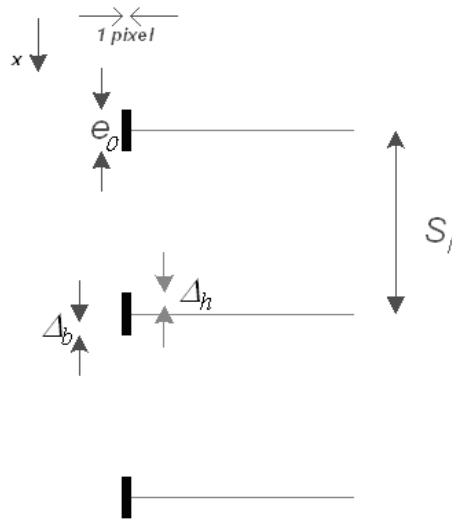


Figure 3.12 : Représentation graphique d'une portion du masque de corrélation, en fonction de l'interligne s_l et de l'épaisseur e_0 .

$$M_p(x) = \begin{cases} 1 & \text{pour } x = \frac{H_p}{2} + k * s_l + i \text{ pour } k \in [-2,2] \text{ et } i \in [-\Delta_b, \Delta_h] \\ 0 & \text{sinon} \end{cases} \quad (\text{Eq. 3.19})$$

avec $0 \leq x < H_p$, $\Delta_b = E\left(\frac{e_0}{2}\right)$ et $\Delta_b + \Delta_h + l = e_0$

Initialisation du filtrage

On initialise le filtre, en calculant la corrélation entre le masque M_p et le début de la portée, pour différents décalages x_d autour de la position moyenne de la troisième ligne de portée $x_p^{(i)}$:

$$C(x_d, y_d^{(i)}) = \frac{1}{H_p} \sum_{x=0}^{H_p-1} \left(M'_p(x) \cdot I^{(i)} \left(x + x_p^{(i)} - \frac{H_p}{2} + x_d, y_d^{(i)} \right) \right)$$

pour $-E\left(\frac{S_l}{2}\right) \leq x_d \leq E\left(\frac{S_l}{2}\right)$ (Eq. 3.20)

Dans la suite, on notera $C_{FO}(x_d, y)$ le résultat du filtrage avec facteur d'oubli, à l'ordonnée y , pour le décalage vertical x_d . Les valeurs initiales sont définies par :

$$C_{FO}(x_d, y_d^{(i)}) = C(x_d, y_d^{(i)}) \quad (\text{Eq. 3.21})$$

La position de la troisième ligne de portée à son extrémité gauche est déduite du décalage x_{d_max} qui maximise $C(x_d, y_d^{(i)})$.

Poursuite de la portée

Pour obtenir les sorties du filtre pour des y croissants, on calcule de nouveau la corrélation locale $C(x_d, y)$ suivant l'équation 3.20. Pour chaque décalage x_d , on pondère le résultat avec celui obtenu à l'itération précédente :

$$C_{FO}(x_d, y) = \alpha * C_{FO}(x_d, y-1) + (1-\alpha) * C(x_d, y) \quad (\text{Eq. 3.22})$$

Les abscisses successives $x_{FO}^{(i)}(y)$ de la troisième ligne de portée sont obtenues par maximisation sur x_d de la sortie du filtre :

$$x_{FO}^{(i)}(y) = x_p^{(i)} + x_{d_max} \quad \text{avec} \quad C_{FO}(x_{d_max}, y) = \max_{x_d} (C_{FO}(x_d, y)) \quad (\text{Eq. 3.23})$$

Le facteur α est appelé facteur d'oubli. Plus sa valeur est grande, plus les résultats de corrélation précédents ont un poids important, plus sa valeur est faible, plus l'algorithme est sensible à la corrélation courante. La valeur choisie est $\alpha=0.98$. Expérimentalement, cette valeur a permis de poursuivre les lentes variations verticales de la position de la portée, sans être sensible aux symboles musicaux.

La figure 3.13 illustre la méthode proposée. Elle montre tout d'abord que les valeurs les plus élevées en sortie du filtre ont lieu au niveau des portions de portée sans symboles. Les sorties diminuent entre ces portions, lorsque le filtre rencontre un symbole, car la corrélation locale (Equation 3.20) est faible. Les variations sont donc d'autant plus importantes que le facteur d'oubli est faible, et cela est vérifié en comparant les sorties du filtre pour $\alpha=0.80$ et $\alpha=0.98$. De même, on remarque que les positions trouvées suivent une évolution plus lisse pour le facteur d'oubli retenu

(0.98) que pour une valeur plus faible (0.8).

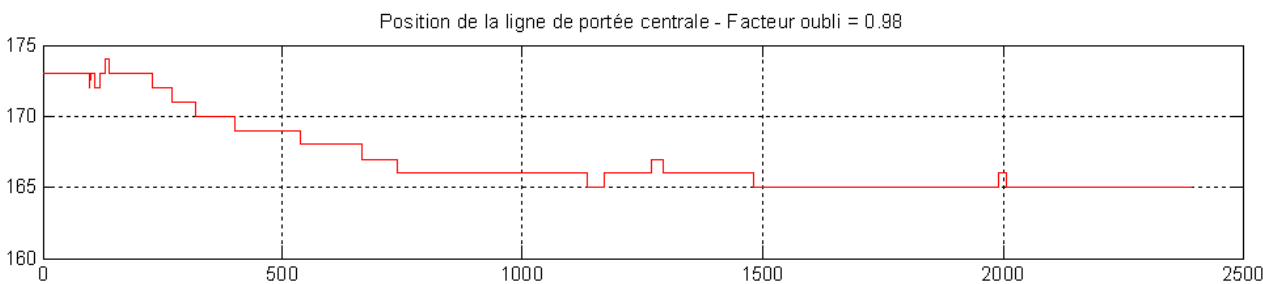
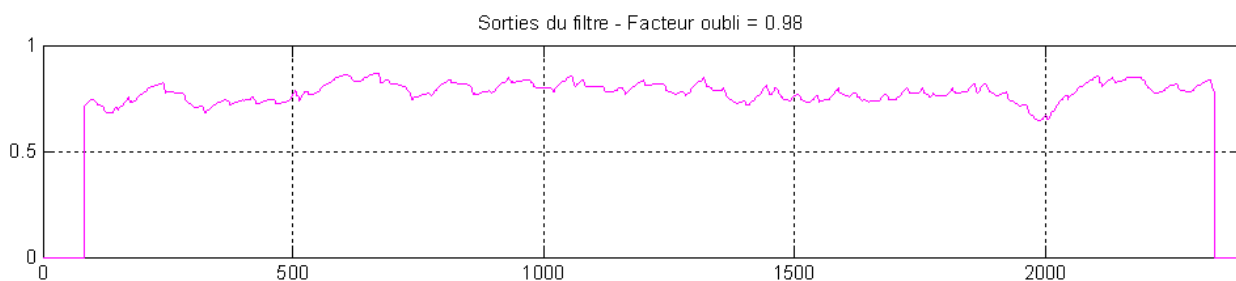
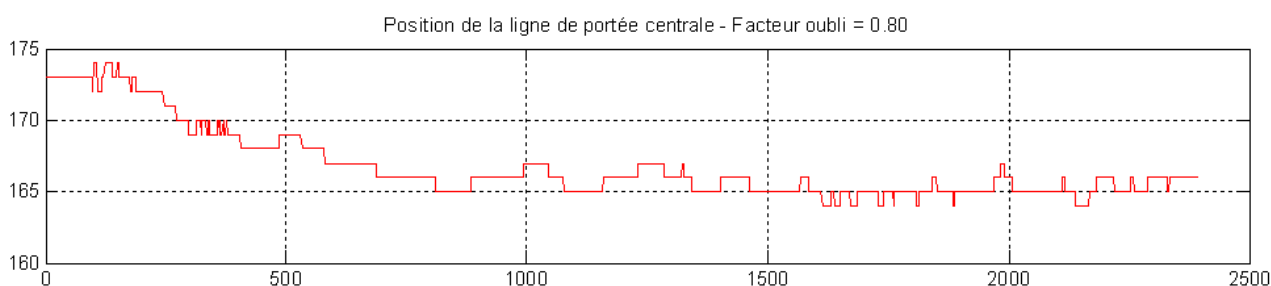
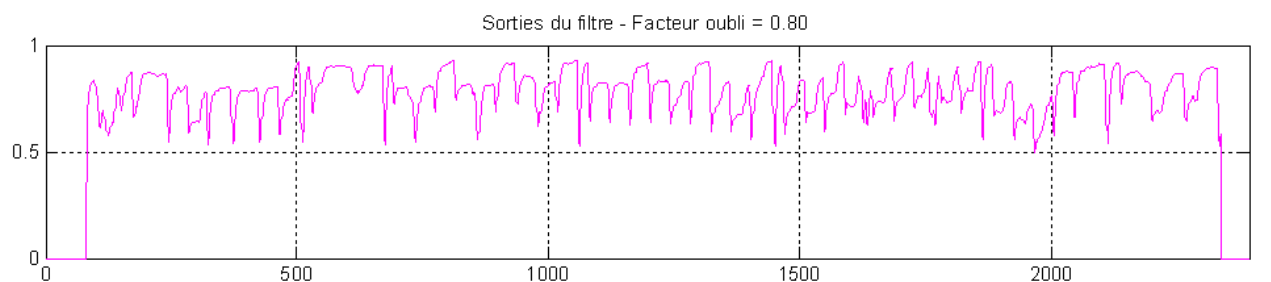
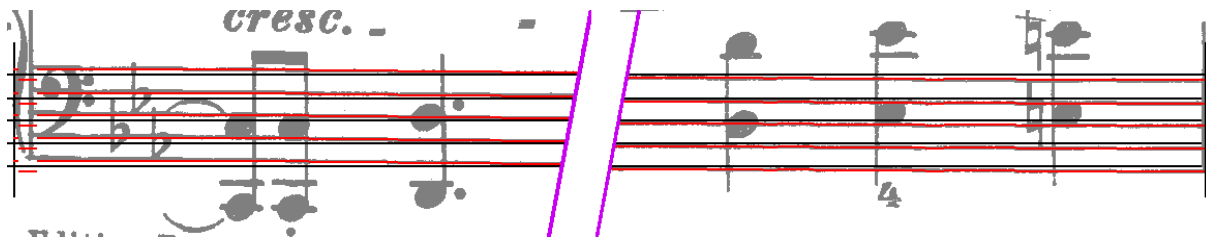


Figure 3.13 : Poursuite de portée pour deux facteurs d'oubli (0.8 et 0.98)

La figure 3.14 montre, en superposition sur l'image redressée, la position moyenne des lignes de portée (en noir), et la position précise déduite de la méthode de poursuite proposée (en rouge). On remarque que les courbures sont effectivement très bien gérées, puisque les lignes rouges se superposent parfaitement aux lignes de portée.



(a) Extrait de la portée 1 de l'image test 1 (monodique) : en noir, la position moyenne des lignes de portée (droites d'équation $x = x_p^{(i)} + k * s_l, k \in [-2, 2]$); en rouge, la poursuite des lignes de portée.



(b) Extrait de la portée 14 de l'image test 2 (polyphonique) : malgré le défaut de début de portée, dû à la petite erreur commise sur le début de portée, l'algorithme accroche rapidement les lignes de portée, et les poursuit précisément de bout en bout.

Figure 3.14 : Poursuite de portées présentant des courbures locales

Conclusion

Nous présentons une méthode novatrice de poursuite des portées, qui nous permet de faire face aux biais et courbures résiduels. Dans cette approche, il y a continuité de l'analyse via le facteur d'oubli du filtre, contrairement à la méthode de [Bellini et al. 01] qui ne semble pas prendre en compte les portions de portée partiellement occultées par des symboles, ou à la méthode de [Bainbridge, Bell 97] qui réalise une analyse très locale, ligne par ligne, d'une colonne à la suivante. Ainsi, notre solution est probablement nettement moins sensible au bruit interférent, surtout en cas de forte densité des symboles. Il aurait été très intéressant de comparer nos résultats avec ceux obtenus par un filtre de Kalman [Poulain d'Andecy et al. 94], car les deux méthodes relèvent finalement du même principe de base permettant de résoudre le problème du masquage. Notre solution est peut-être plus robuste, car elle traite d'un coup les cinq lignes équidistantes, qui ne sont a priori pas toutes masquées simultanément. Soulignons que seuls [Wijaya, Bainbridge 99] restaurent la rectitude des portées courbées.

3.1.4. Conclusion

La détection des lignes de portée est effectuée en trois étapes : calcul du biais et redressement de l'image, localisation et caractérisation globale de la portée, poursuite des lignes de portée pour affiner les résultats. Notre méthodologie s'apparente donc à celles qui sont fondées sur les projections (par exemple [Martin 92], voir chapitre 1.3.2), et à celles qui procèdent par

localisation et raffinage [Bainbridge, Bell 97]. L'originalité se situe au niveau des méthodes adoptées pour la réalisation de chaque étape, qui, projections et histogramme des longueurs des empanns exceptés, sont toutes novatrices.

Concernant les deux premières étapes, nous avons mis en œuvre des algorithmes simples, rapides à l'exécution et très robustes, car ils travaillent non pas localement, mais sur toute l'image. Sur notre base de données, toutes les portées ont été correctement localisées. La méthode de filtrage avec facteur d'oubli produit ensuite de très bons résultats. Elle est inutile pour les portées bien rectilignes, mais elle est indispensable dans le cas contraire. Nous avons pu alors constater une nette augmentation des taux de reconnaissance (symboles et durées), jusqu'à +5%. La robustesse de l'ensemble du processus de reconnaissance, par rapport aux défauts du document original ou à ceux introduits par la numérisation, a donc été considérablement améliorée.

3.2. Segmentation

Cette étape a pour objectif d'isoler les différents symboles dans l'image. Comme nous l'avons souligné dans la section 1.2, elle présente trois difficultés majeures : les symboles musicaux sont largement interconnectés par les lignes de portée, qui camouflent leur contour ; les défauts d'impression ajoutent des connexions parasites ou au contraire scindent certaines entités, en particulier les segments fins ; enfin, il faut définir le niveau de décomposition des symboles construits, typiquement les groupes de notes.

Nous avons, à l'instar de la majorité des auteurs, choisi de commencer par l'effacement des lignes de portée. Cette démarche semble en effet très naturelle, car elle provoque la déconnexion immédiate d'un grand nombre de symboles. En particulier, les silences et les rondes sont ainsi très bien isolés. Tous les autres symboles que nous souhaitons reconnaître, c'est-à-dire les noires, les blanches, les altérations et les appoggiatures, sont caractérisés par la présence d'au moins un segment vertical (Figure 3.15). La deuxième phase de la segmentation passe donc par la détection de ces segments verticaux, à partir desquels on peut appliquer un algorithme de croissance de région pour isoler chaque symbole par une boîte englobante. Bien entendu, il faut définir d'une part des règles de séparation, puisque certains symboles sont connectés entre eux, en particulier les notes groupées, et d'autre part des règles de fusion, puisque deux segments peuvent aboutir à la même région (cas des bécarrés ou des dièses qui possèdent deux segments verticaux).

















| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|--|---|---|---|
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| blanche | noire | dièse | bémol | bécarré | appoggiature | barre de mesure | pause | ½ pause | soupir | soupir | 1/2 soupir | 1/4 soupir | 1/8 soupir | point | ronde |
| Notes | | | | | | Altérations | | | | | | Silences | | | |
| Avec segment vertical | | | | | | | Sans segment vertical | | | | | | | | |

Figure 3.15 : Symboles musicaux caractérisés par au moins un segment vertical, ou sans aucun segment vertical

La segmentation doit être robuste par rapport aux défauts d'impression. Nous ne pourrons

pas obtenir une localisation parfaite à ce stade de l'analyse, mais il faut minimiser les défauts qui au mieux génèrent de l'ambiguïté pour la reconnaissance, au pire la rendent impossible. Par exemple, l'effacement abusif de pixels objet entraîne une imprécision sur la forme de l'objet, que nous modéliserons dans les étapes de plus haut niveau, tandis que la non-détection d'un segment vertical implique la non-reconnaissance du symbole.

La figure 3.16 résume les différentes étapes de la segmentation, qui sont explicitées dans les paragraphes suivants. La méthode est appliquée portée par portée, par conséquent sur chaque image $I^{(i)}$.

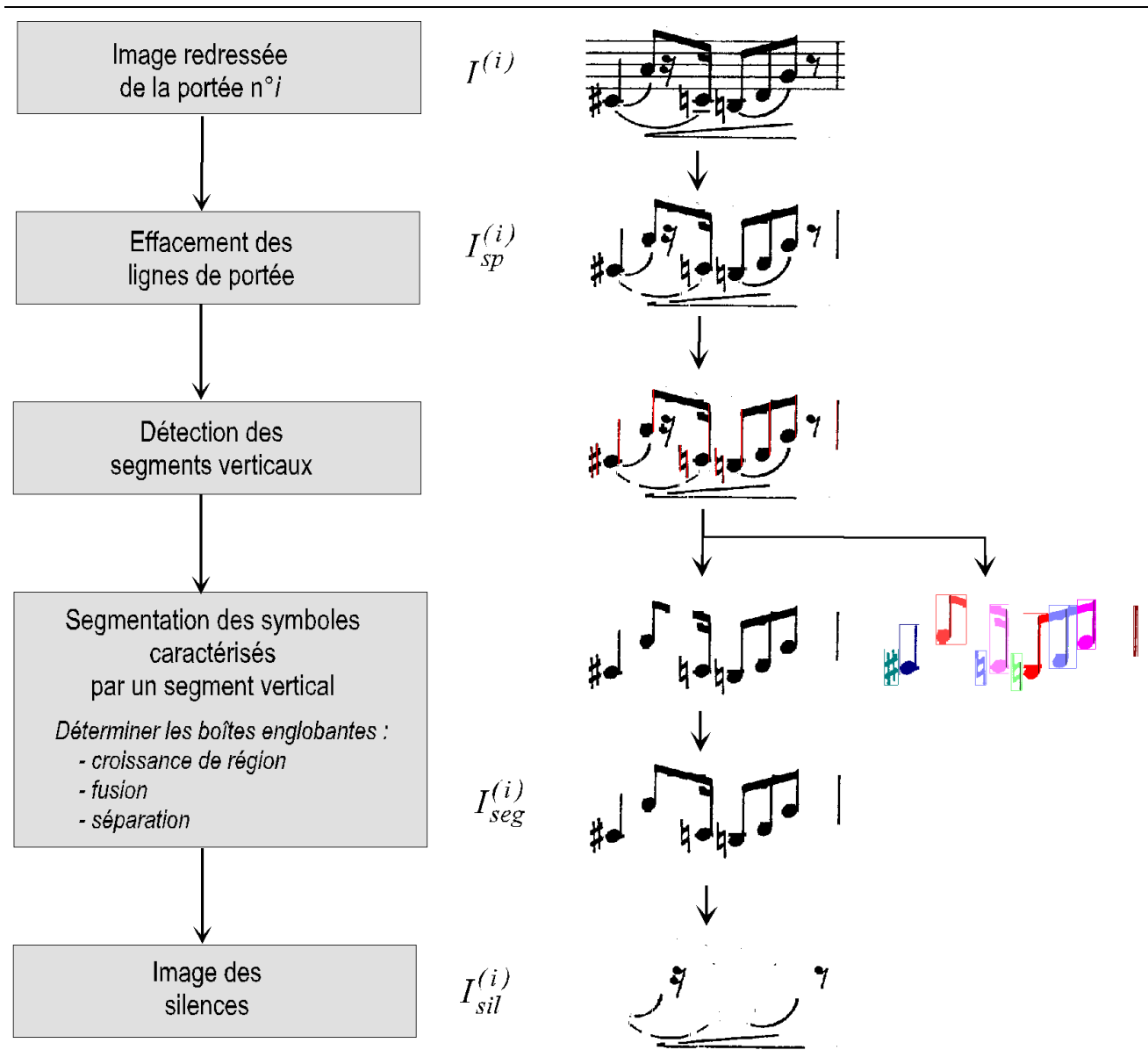


Figure 3.16 : Les différentes phases de la segmentation

3.2.1. Effacement des lignes de portée

La détection et la poursuite des lignes de portée ont permis de connaître exactement leur

position en toute ordonnée y dans la sous-image $I^{(i)}$. Nous connaissons également très précisément leur épaisseur moyenne e_{moy} . L'image sans portée $I_{sp}^{(i)}$ peut donc être obtenue en appliquant un algorithme d'effacement, qui poursuit chaque ligne de portée de la gauche de l'image vers la droite, et qui supprime toutes les colonnes de pixels noirs connexes, appelées également empan, dont la longueur est inférieure à un seuil, fixé légèrement supérieur à l'épaisseur e_{moy} :

$$s_e = \text{Arrondi}(e_{moy}) + 2 \quad (\text{Eq. 3.24})$$

Considérons un empan, situé à l'ordonnée y , et dont les extrémités se situent aux abscisses x_h et x_b . Soit x la position de la ligne de portée traitée (ligne réelle ou additionnelle), indiquée par k , à l'ordonnée y (Figure 3.17) :

$$x = x_{FO}^{(i)}(y) + k * s_l, \quad k \in [-6,6] \quad (\text{Eq. 3.25})$$

Alors le segment est effacé si et seulement si les trois critères suivants sont simultanément vérifiés :

- (1) $(x_b - x_h + 1) \leq s_e$
 - (2) $x_h > (x - s_e)$
 - (3) $x_b < (x + s_e)$
- (Eq. 3.26)

Cette règle d'effacement signifie que toute colonne de pixels qui intersecte la ligne de portée, et dont la longueur est inférieure ou égale à s_e , est considérée comme un empan de portée sans symbole superposé, et peut donc être supprimée. Sur la figure 3.17, le trait en rouge indique la position courante x de la ligne de portée. Le premier empan indiqué en vert n'est pas supprimé, car il ne vérifie pas le critère (1) de longueur. Le second empan dessiné en vert satisfait au contraire aux trois critères, et il est donc effacé.

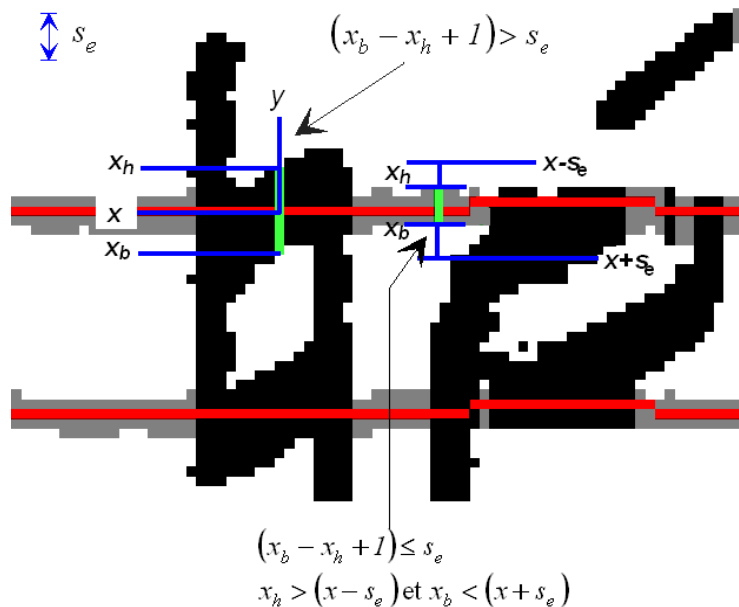


Figure 3.17 : Méthode d'effacement des lignes de portée

La méthode est appliquée sur les 5 lignes de portée, mais aussi, par extrapolation, sur les petits segments horizontaux qui supportent les notes au-dessus et au-dessous de la portée. On considère 8 lignes additionnelles, 4 au-dessus de la portée et 4 au-dessous ($k \in [-6,6]$ dans l'équation 3.25). Ainsi les connexions dues aux lignes de portée principales ou aux lignes additionnelles sont supprimées, et le traitement est parfaitement homogène pour tous les symboles, quelle que soit leur position par rapport à la portée. La figure 3.18 illustre sur un exemple les résultats que l'on obtient typiquement. Ceux-ci sont satisfaisants, dans la mesure où les portions de portée sans symboles sont effectivement très bien effacées, et les symboles sans segment vertical par conséquent bien isolés. Les groupes de notes sont également séparés des autres symboles, mais les notes sont toujours connectées entre elles par les barres de groupe dont l'épaisseur est supérieure à celle des lignes. Cela nous permettra par la suite d'identifier les groupes et d'analyser les barres pour en déduire la durée de chaque note. Certains signes de phrasé (liaisons, crescendo, etc.) sont partiellement effacés, mais ce n'est pas important car nous ne cherchons pas à les reconnaître.



Figure 3.18 : Effacement des lignes de portée, exemples de résultats.

Néanmoins, on constate des défauts au niveau des points de connexion entre les symboles et les lignes de portée : certains pixels "symbole" sont supprimés, tandis que des pixels "ligne", connexes aux symboles, demeurent.

Le premier type de défaut se produit au niveau des portions fines de symboles, superposées ou tangentes aux lignes de portée. Cela concerne les symboles creux, typiquement les têtes de note blanches, les bémols, et certaines portions de clé. On observe aussi ce phénomène, mais plus

rarement, pour certains silences (soupir, demi-soupir ou quart de soupir) ou certains crochets de note particulièrement fins au croisement d'une ligne de portée (Figure 3.19). L'effacement de ces pixels peut provoquer la fragmentation du symbole. La suppression d'un empan de barre de groupe est rarissime, car le critère d'épaisseur choisi est strictement inférieur à l'épaisseur de ces barres, et la robustesse sur la détection des groupes de notes est par conséquent assurée.



Figure 3.19 : Cas d'effacements de pixels "symbole" (défauts de type 1)

A contrario, certains pixels qui appartiennent aux lignes de portée et non aux symboles ne sont pas effacés. C'est le second type de défauts, qui se manifeste au niveau des têtes de note situées dans un interligne, lorsque les pixels du contour de la noire sont connexes aux pixels des lignes de portée (Figure 3.20). Pour les symboles creux, comme les têtes de note blanches, les deux types d'erreurs apparaissent, si bien que la forme du symbole est fortement altérée. La figure 3.20 présente d'autres exemples de pixels non éliminés, au niveau de certains silences, altérations, ou barres de groupe. On peut également constater que certaines lignes supplémentaires ne sont pas supprimées, car les interlignes ne sont pas toujours stables au-dessus ou au-dessous de la portée.

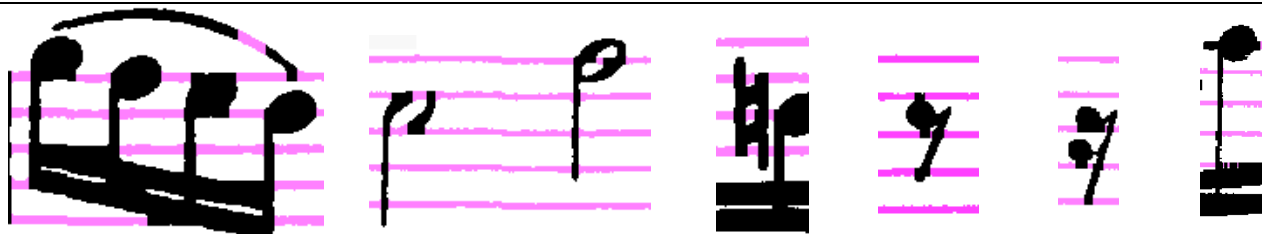


Figure 3.20 : Cas de non effacement de pixels appartenant aux lignes de portée (défauts de type 2)

Il résulte donc de cette opération une imprécision et une variabilité sur la forme des objets. Un même symbole peut prendre des formes légèrement différentes après la procédure d'effacement, suivant sa position par rapport aux lignes de portée. Par conséquent, la variabilité des symboles, déjà observée dans les documents originaux (variabilité inter et intra partition), est encore accrue. La figure 3.20 illustre cela de manière évidente sur les têtes de note blanches.

Ces problèmes semblent impossibles à éviter à ce stade de l'analyse, puisque les contours exacts des symboles sont masqués par les lignes de portée, comme le souligne [Préreau 70], et qu'il faudrait donc connaître préalablement la classe des symboles pour les segmenter plus précisément [Coüasnon 96b]. La plupart des auteurs ont mis en œuvre une procédure similaire, fondée sur un critère d'épaisseur, et ont également constaté ses limites. Certains [Carter 89] [Bainbridge, Bell 97] [Martin, Bellissant 91] ont proposé une méthode qui semble partiellement résoudre le premier type

de défaut, mais pas le second. Nous avons pour notre part limité les imperfections grâce à la caractérisation précise des portées. L'algorithme de poursuite de portée (paragraphe 3.1.3) permet de localiser précisément les lignes de portée, et ainsi d'éviter que des portions entières, décalées par rapport à la position moyenne, n'échappent au processus d'effacement. La mesure précise de l'épaisseur moyenne des lignes de portée permet de restreindre la zone d'analyse, en d'autres termes de minimiser le paramètre s_e , et donc de minimiser les suppressions abusives. Les expérimentations ont montré que le critère choisi (Eq. 3.24) aboutit en moyenne au meilleur résultat sur toute la base de données. Les images sans lignes de portée seront notées $I_{sp}^{(i)}$ dans la suite de l'exposé.

Dans la littérature, les auteurs qui ont réalisé la segmentation sans effacement des portées sont rares. L'argument avancé [Bellini et al. 2001] est essentiellement le problème de la fragmentation de symboles, qui nécessiterait de mettre en œuvre des mécanismes complexes pour leur reconstruction, et la dégradation des symboles qui correspond à une perte d'information. Dans notre méthodologie, nous tolérons les défauts de segmentation provoqués par l'effacement des lignes de portée, tout comme les nombreux auteurs qui ont adopté cette démarche. Mais nous en tiendrons compte explicitement dans les étapes ultérieures : en adoptant des méthodes de détection et de reconnaissance des symboles adaptées, robustes par rapport à ces problèmes, et surtout, grâce à l'étape de modélisation floue qui nous permet de traiter l'imprécision sur la forme et la position des objets, et de résoudre les ambiguïtés résultantes. Ce dernier aspect constitue un point original et essentiel de la méthode que nous proposons.

3.2.2. Détection des symboles caractérisés par un segment vertical

La seconde phase de la segmentation concerne tous les symboles qui sont caractérisés par la présence d'un segment vertical. On distinguera dans la suite le terme "empan vertical", qui désigne une colonne de pixels noirs connexes, et le terme "segment vertical", qui fait référence à une ligne verticale d'épaisseur supérieure ou égale à 1, donc constituée d'empans verticaux contigus. On peut également voir le segment comme une succession d'empans noirs horizontaux, de faible longueur, et verticalement alignés. Les symboles caractérisés par un segment vertical sont les notes qui possèdent une hampe (toutes les notes exceptées les rondes), les altérations et les appoggiatures (Figure 3.15). La méthode de segmentation de ces symboles est appliquée sur chaque sous-image $I_{sp}^{(i)}$, donc après effacement de la portée, et consiste en deux phases : détection des segments verticaux puis définition des rectangles englobant les symboles, par croissance de région à partir des segments détectés.

Détection des segments verticaux caractéristiques des notes et des altérations

La détection des segments verticaux doit surmonter deux difficultés majeures : les ruptures de segment et le biais. Les ruptures de segment, c'est-à-dire les interruptions durant quelques pixels, sont très fréquentes dans les documents originaux, et sont parfois introduites par la numérisation. De nombreux auteurs soulignent en particulier le problème des objets fragmentés, montrant qu'il s'agit d'un point crucial à résoudre pour espérer obtenir une bonne fiabilité du système de reconnaissance (e.g. [Coüasnon 96b], [Bainbridge, Bell 97], [Poulain d'Andecy et al. 94]). Nous

devons donc résoudre au mieux cette difficulté au niveau de la détection des segments et dans les étapes ultérieures. Le biais des segments verticaux résulte, soit de l'imperfection de l'impression du document original, soit du biais global de l'image scannée, qui n'a été que partiellement corrigé (voir paragraphe 3.1.1). Enfin, il faut remarquer que les segments verticaux sont connexes à d'autres primitives (par exemple les hampes sont connectées à une tête de note et à des barres de groupe), ou sont inclus dans des symboles (par exemple les altérations). Ils ne se présentent donc pas sous une forme linéaire sur toute leur longueur.

Les segments verticaux que nous recherchons sont caractérisés par les critères géométriques et topologiques suivants, à la taille et à la résolution image considérées :

1. Une longueur supérieure à 1.5 interligne.
2. Une épaisseur de l'ordre de 1 à 5 pixels sur les parties linéaires.
3. Un espacement entre le segment et les objets voisins d'au moins 2 pixels.
4. Un espacement entre deux segments caractéristiques d'un symbole d'au moins 1/5 d'interligne.
5. En musique monodique, on ne peut trouver verticalement qu'un seul empan correspondant à un segment de symbole musical. On fera l'hypothèse que cet empan est le plus long de la colonne considérée, hypothèse qui s'avère fondée en pratique car les autres inscriptions, en particulier les textes, sont plus petites, ou alors trop épaisses.

Les défauts tolérés par rapport au segment "idéal" sont :

6. La présence d'une ou plusieurs ruptures, de 2 pixels au maximum.
7. Un faible biais. Dans le cas de segments très fins, il n'existe alors pas d'empan vertical qui parcourt le segment sur toute sa longueur.

Cette analyse suggère de calculer une carte des empan noirs verticaux pour l'analyse de longueur (point 1), de filtrer l'image afin d'analyser l'épaisseur des segments et de valider le critère d'espacement (points 2 et 3). Une analyse du voisinage des extrémités des empan détectés permet de reconnecter des segments interrompus, ou des empan horizontalement décalés à cause du biais (points 6 et 7), tandis que les critères 4 et 5 permettent de ne retenir que les segments pertinents, et de les caractériser par un unique empan vertical.

Nous allons maintenant décrire en détail les différentes phases de la méthode. Celles-ci sont illustrées ci-dessous, sur une partition qui présente des traits très épais (Figure 3.21a), et sur une partition imprimée avec des traits très fins (Figure 3.22a). Les partitions analysées se situent généralement entre ces deux cas extrêmes.

La première phase consiste à parcourir toute l'image, colonne par colonne, et à créer une carte codant la longueur des empan verticaux noirs détectés. Soit $I_v^{(i)}(x,y)$ cette carte (Figures 3.21b et 3.22b).

$$I_v^{(i)}(x,y) = 0 \text{ si } I_{sp}^{(i)}(x,y) = 0 \quad (\text{Eq. 3.27})$$

$$I_v^{(i)}(x,y) = l, \text{ avec } l \text{ la longueur de l'empan vertical contenant } (x,y)$$

Dans la seconde phase, les empan horizontaux susceptibles d'appartenir à un segment vertical sont extraits par convolution du négatif de l'image $I_{sp}^{(i)}$, noté $\overline{I_{sp}^{(i)}}$, avec le noyau N_l , et intersection de l'image résultat avec l'image source $I_{sp}^{(i)}$:

$$I_l^{(i)}(x, y) = I_{sp}^{(i)}(x, y) \sum_{j=-4}^4 \left(\overline{I_{sp}^{(i)}(x, y + j)} N_l(j) \right)$$

$$N_l = \frac{1}{4} [1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1] \quad (\text{Eq.3.28})$$

La valeur maximale (1.0) est obtenue pour les pixels centrés sur un empan horizontal de longueur inférieure ou égale à 5, et séparé des objets voisins d'au moins 2 pixels blancs de part et d'autre. Le filtre met donc typiquement en évidence les pixels appartenant aux segments verticaux. Sur la figure 3.22c (traits fins), on constate que les pixels des segments verticaux recherchés sont des maxima (en rouge) ; sur la figure 3.21c (traits gras), les valeurs obtenues varient entre 0.5 et 1.0.

Dans la troisième phase, on réalise une fermeture verticale d'ordre 2. Afin d'éviter de connecter des objets qui doivent être effectivement bien séparés, il faut vérifier que les deux empan concernés sont de type ligne. La règle est la suivante : deux empan de la colonne y , séparés de 1 ou 2 pixels blancs, sont connectés si leurs extrémités voisines sont toutes les deux des maxima dans l'image $I_l^{(i)}$ (Figure 3.22d). La longueur de l'empan obtenu par cette fusion est mise à jour dans la carte $I_v^{(i)}$. On calcule également à ce stade une carte de la longueur des empan horizontaux, notée $I_h^{(i)}$.

Dans la quatrième phase, on recherche dans chaque colonne de la carte $I_v^{(i)}$ l'empan le plus long. Soit l la longueur de l'empan considéré, dont les extrémités sont situées aux abscisses x_h et x_b . Cet empan est retenu si sa longueur $l=(x_b-x_h+1)$ est supérieure à 1.5 interligne (point 1), et s'il satisfait globalement aux critères d'épaisseur et d'espacement (points 2 et 3) :

$$l > 1.5 s_l \text{ et } \frac{\sum_{x=x_h}^{x_b} I_l^{(i)}(x, y)}{l} > \frac{1}{4} \Rightarrow \text{empan retenu} \quad (\text{Eq. 3.29})$$

Les figures 3.21e et 3.22e montrent les résultats obtenus. On remarque que le critère choisi est un bon compromis. Il permet de supprimer les empan trop courts pour appartenir aux segments recherchés, ainsi que les empan correspondant à des objets trop épais (typiquement les empan inclus dans des barres de groupe de notes, connectées à cause des défauts d'impression), sans toutefois supprimer les empan significatifs des segments recherchés.

La cinquième phase permet de retenir un empan unique par segment vertical. Pour cela, une fenêtre d'analyse de largeur $s_l/5$ parcourt horizontalement l'image. Dans cette fenêtre, on retient parmi les empan restants l'empan le plus long, et on supprime les autres. Les figures 3.21f et 3.22f illustrent les résultats obtenus. A ce stade de l'analyse, on constate qu'aucun des segments verticaux significatifs de la présence de symboles musicaux n'est manqué ; en revanche, il reste quelques

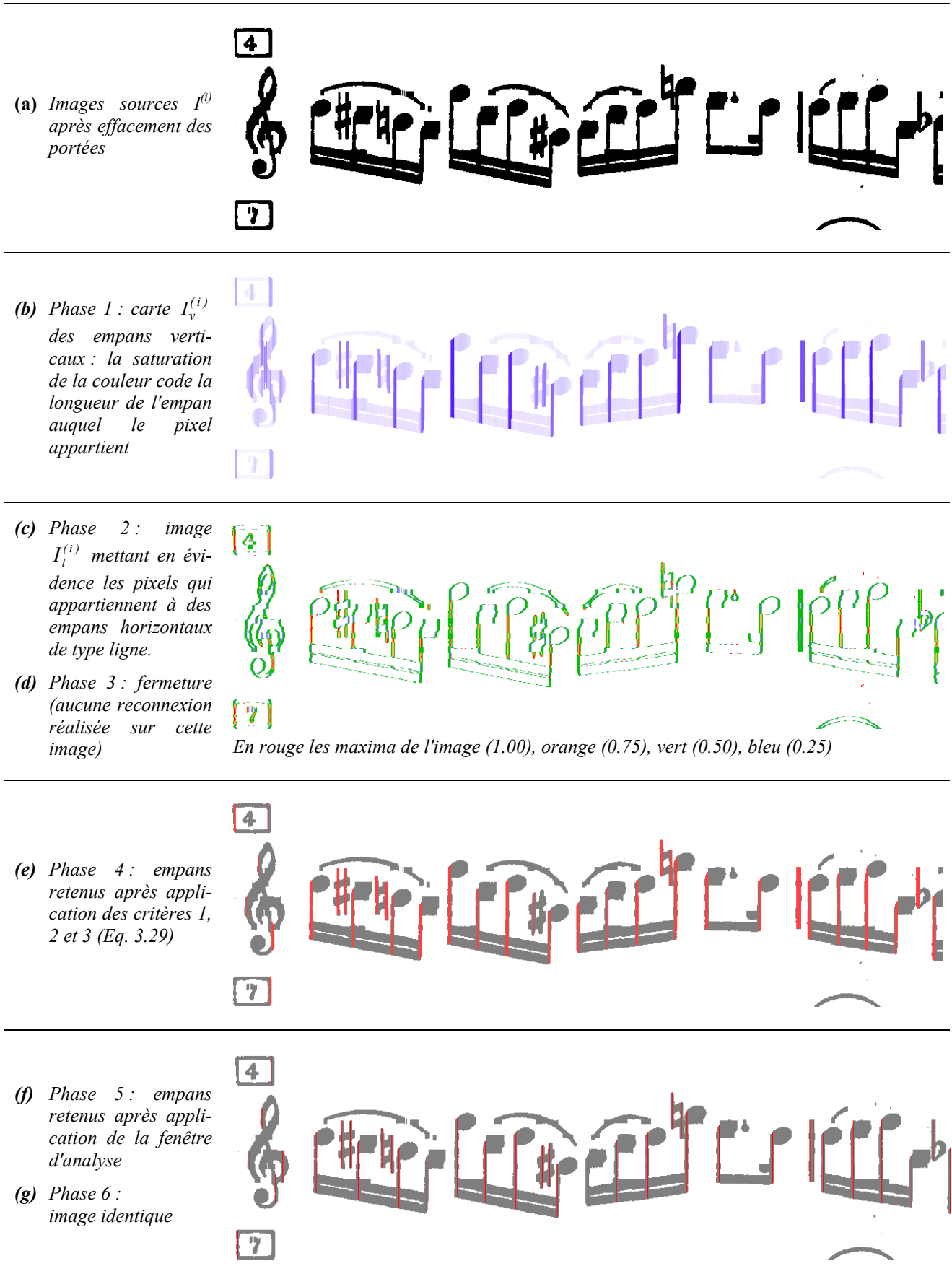


Figure 3.21 : Détection des segments verticaux (cas de traits très épais)

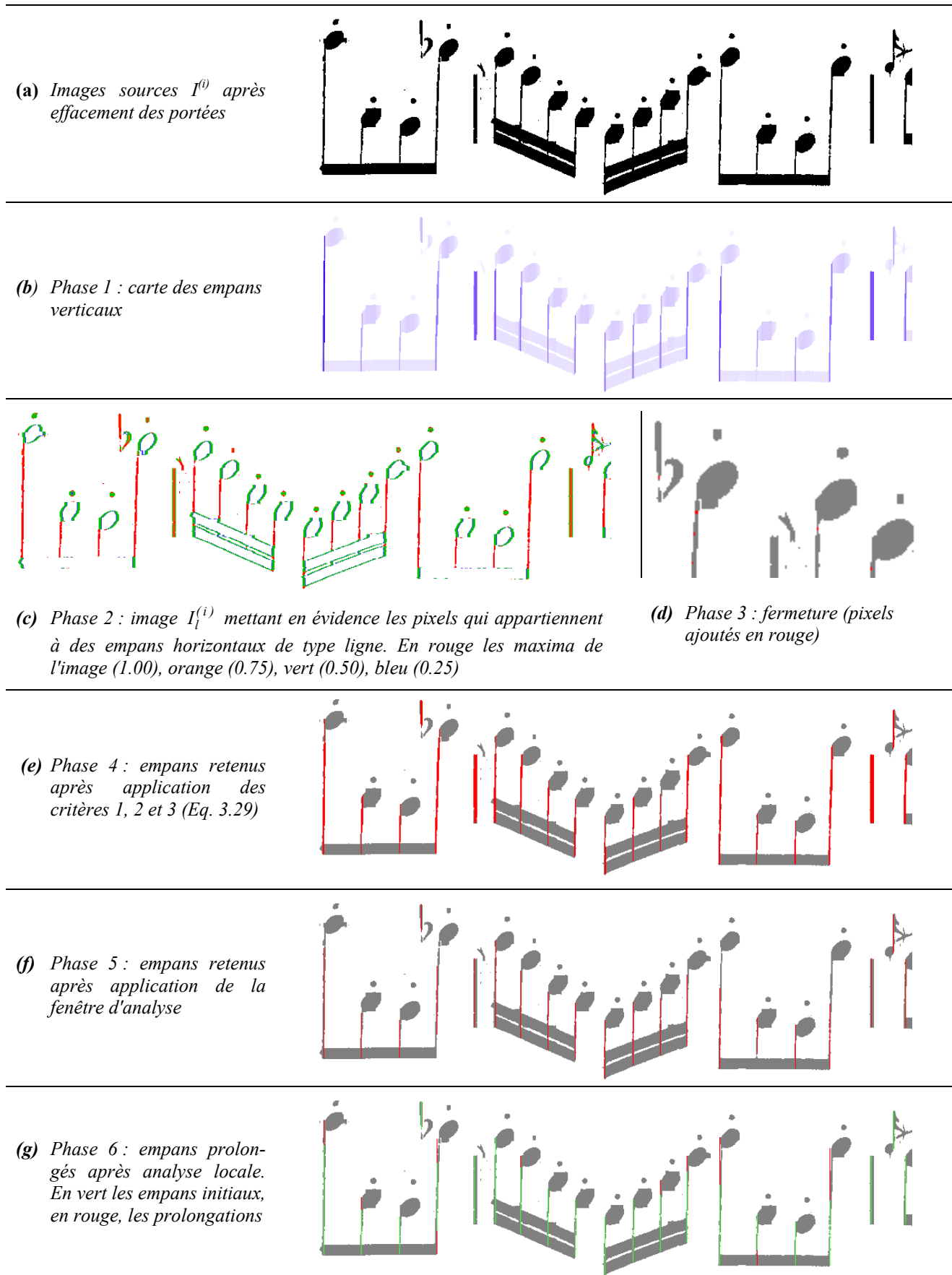


Figure 3.22 : Détection des segments verticaux (cas de traits très fins)

fausses détections, au niveau de la clé de sol notamment, et dans le cas de segments très épais (barre de mesure de la figure 3.21f). Ces fausses détections ne sont pas vraiment pénalisantes, car elles pourront être éliminées dans les étapes ultérieures : par fusion de boîtes englobantes identiques, ou, à défaut, lors de la modélisation floue qui détectera une incohérence graphique. L'essentiel est de ne pas manquer de segment vertical, car alors le symbole correspondant ne serait pas détecté, et donc irrémédiablement non reconnu. On remarque également que les segments très fins présentant un biais ne sont pas détectés sur toute leur longueur (Figure 3.22f).

L'objet de la sixième et dernière étape est donc d'affiner ces résultats en associant des empanx verticaux qui appartiennent à un même segment, mais qui sont horizontalement décalés à cause du biais. Le critère de fusion porte sur les épaisseurs des segments correspondants, qui doivent être similaires.

Considérons de nouveau un empanx retenu à l'étape précédente, situé à l'ordonnée y et d'extrémités x_h et x_b . On peut estimer l'épaisseur moyenne e_{p0} du segment correspondant en moyennant les longueurs des empanx horizontaux caractéristiques des lignes verticales (maxima dans $I_l^{(i)}$), situés entre x_h et x_b :

$$e_{p0} = \frac{1}{N} \sum_{\substack{x_h \leq x \leq x_b \\ I_l^{(i)}(x,y)=1.0}} I_h^{(i)}(x,y), N = \text{Card}\{(x,y) / x_h \leq x \leq x_b \text{ et } I_l^{(i)}(x,y)=1.0\} \quad (\text{Eq 3.30})$$

On évalue ensuite le voisinage, aux coordonnées $(x_h - l, y - l)$, $(x_h - l, y + l)$, $(x_b + l, y - l)$ et $(x_b + l, y + l)$. Prenons par exemple le voisinage supérieur. Si un empanx vertical est présent à gauche (en $y-l$), absent à droite (en $y+l$), et si l'épaisseur moyenne e_p du segment, calculée suivant l'équation 3.30, est comparable à e_{p0} (même épaisseur à 1.0 près), alors l'empanx principal est prolongé, et les coordonnées des extrémités sont remises à jour. Le même principe est appliqué pour le voisinage inférieur, et le procédé est réitéré tant que des empanx voisins peuvent être fusionnés. La figure 3.22g montre l'importance de cette ultime étape dans le cas des partitions imprimées avec des traits très fins.

Les symboles détectés sont numérotés (indice s) et sont stockés dans une structure mémorisant les paramètres trouvés : l'ordonnée du segment, dorénavant notée $y_p(s)$, l'abscisse de l'extrémité supérieure, notée $x_{ph}(s)$, et l'abscisse de l'extrémité inférieure, notée $x_{pb}(s)$.

La méthode de détection des segments verticaux est donc réalisée à partir de trois images extraites de l'image source $I_{sp}^{(i)}$: la carte des longueurs des empanx verticaux $I_v^{(i)}$, la carte des longueurs des empanx horizontaux $I_h^{(i)}$, et l'image filtrée $I_l^{(i)}$ qui extrait les empanx horizontaux satisfaisant à un critère caractérisant les lignes. Elle permet de surmonter les principales difficultés, c'est-à-dire les cas de rupture de segments et les problèmes de biais sur les segments fins, dans la mesure où ces défauts sont dans la limite tolérée. Les cas de non-détection, qui doivent être absolument évités, sont extrêmement rares et correspondent à des cas extrêmes de dégradation de l'image.

La figure 3.23 montre des exemples de segments imparfaitement détectés, à cause de ruptures ou de déconnexions supérieures à 2 pixels : fragmentation ou effacement d'une portion de segment qui

conduit à une détection partielle, déconnexion de la tête de note ou des barres de groupe. Il faut noter que ces défauts sont présents dans le document original et ne proviennent pas de l'algorithme d'effacement des lignes de portée. Tolérer des ruptures supérieures, de 3 pixels ou plus, conduit à des reconnections erronées, et nuit globalement à la reconnaissance. Un autre défaut est la détection multiple de segments, pour les impressions en traits très gras, ou la fausse détection, par exemple de crochets ou de silences. Toutes ces imperfections seront prises en compte dans les étapes ultérieures, au niveau de la méthode d'analyse des symboles (chapitre 4) et de la modélisation floue qui permettra de lever les ambiguïtés de classification (chapitre 5).

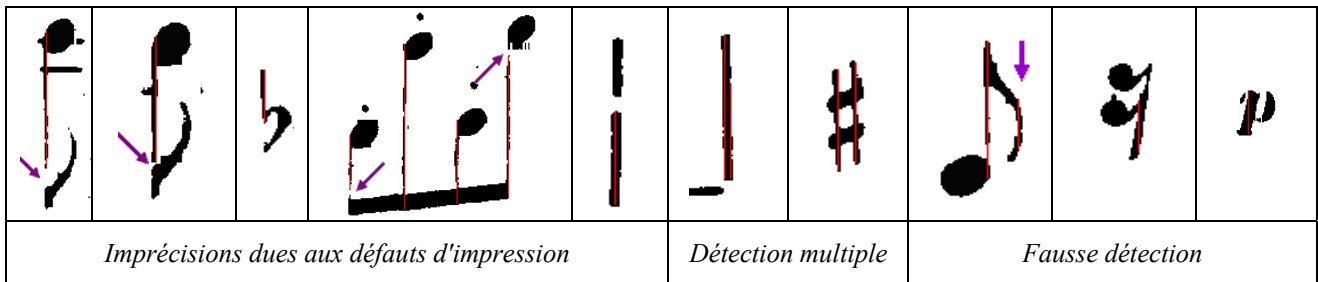


Figure 3.23 : Cas de segments imparfaitement détectés ou improprement détectés

Boîtes englobantes délimitant les symboles caractérisés par un segment vertical

L'empan principal de chaque segment ayant été précisément localisé, il peut servir de germe pour un algorithme de croissance de région, qui agglomère tous les pixels noirs connexes à cet empan, au sens des 8-voisins. On peut ainsi délimiter le symbole par une boîte englobante. Comme certains symboles sont encore connectés malgré l'effacement des lignes de portée, en particulier les notes reliées par des barres de groupe, on limite la croissance de part et d'autre de l'empan à 1.5 interligne. La largeur d'un symbole est en effet toujours inférieure à cette valeur, de chaque côté du segment vertical. L'autre condition d'arrêt de la croissance de région est l'absence de pixels noirs connexes à la région détectée. Les paramètres obtenus sont, pour l'objet indicé s , $(x_h(s), y_g(s))$ et $(x_b(s), y_d(s))$: ils définissent les coordonnées des coins supérieur gauche et inférieur droit de la boîte englobante (Figure 3.24). Les figures 3.25 et 3.26 illustrent les différentes étapes de la segmentation des symboles caractérisés par un segment vertical. Les résultats obtenus après la croissance de région sont représentés en (b).

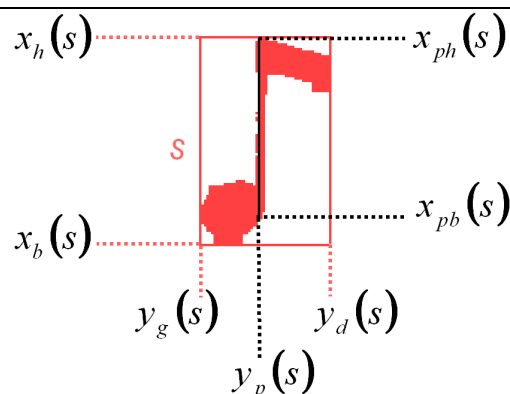
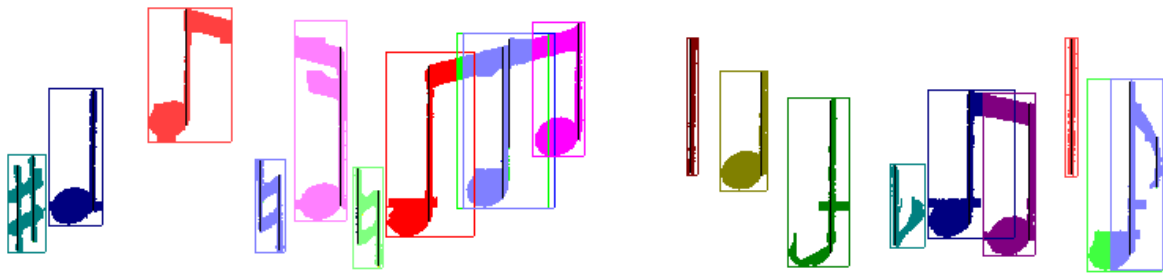


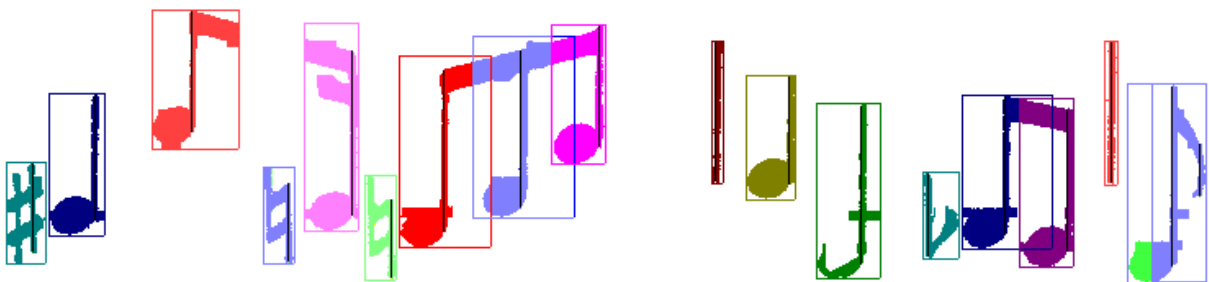
Figure 3.24 : Croissance de région à partir du segment principal. Les symboles sont ainsi limités par une boîte englobante.



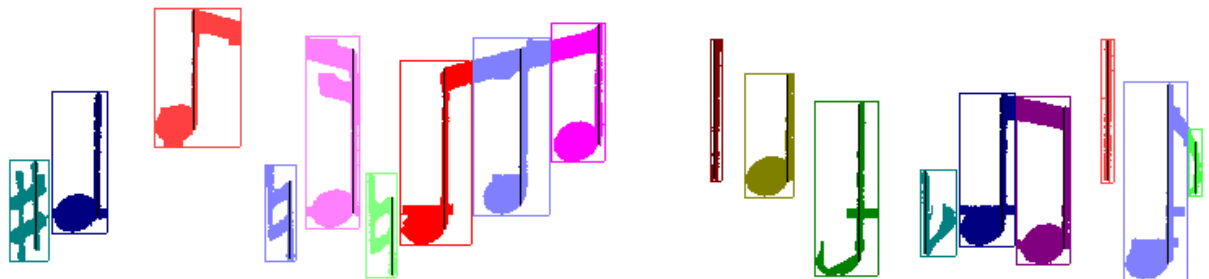
(a) Image après effacement des lignes de portée



(b) *Étiquettes obtenues par croissance de région à partir du segment vertical et boîtes englobantes.* Certains symboles sont détectés deux fois (dièse, bécarres, hampe épaisse) et conduisent à des boîtes englobantes identiques (cas des symboles isolés) ou presque superposées (cas de la hampe). Les boîtes englobantes des symboles connectés se chevauchent.



(c) *Étiquettes et boîtes englobantes après fusion.* Les boîtes englobantes similaires sont fusionnées. Le segment détecté sur le crochet de la dernière note n'est pas résolu, car les deux segments détectés pour cette note ne conduisent pas à des boîtes englobantes similaires.

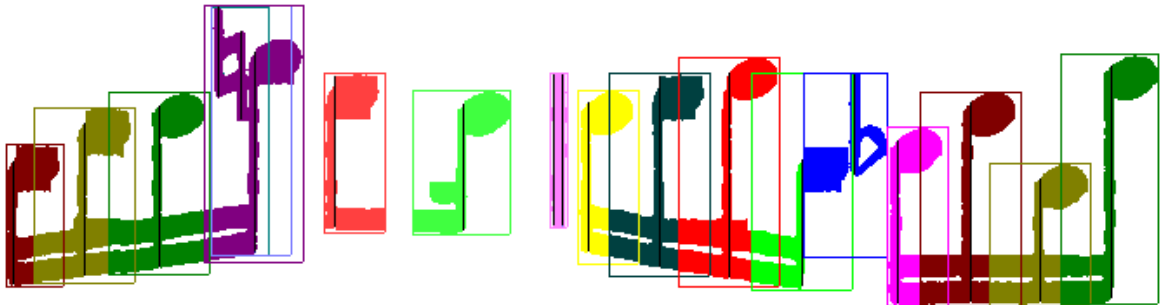


(d) *Étiquettes et boîtes englobantes après séparation.* Les notes incluses dans des groupes sont bien séparées les unes des autres. Le défaut au niveau de la dernière note devra être résolu dans les étapes ultérieures.

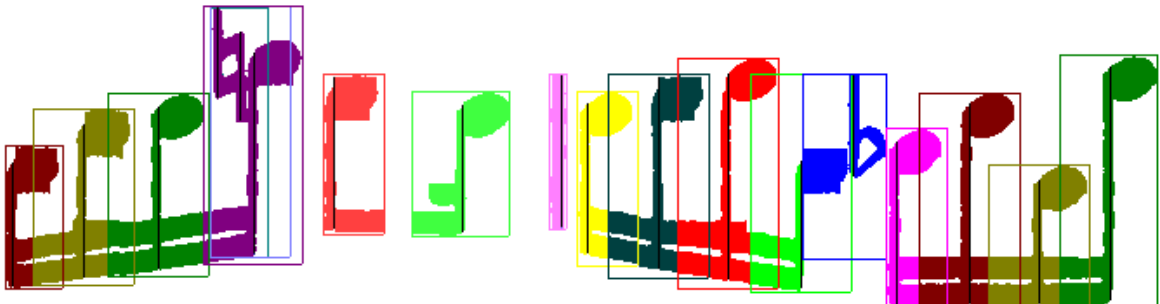
Figure 3.25: Segmentation des symboles caractérisés par un segment vertical



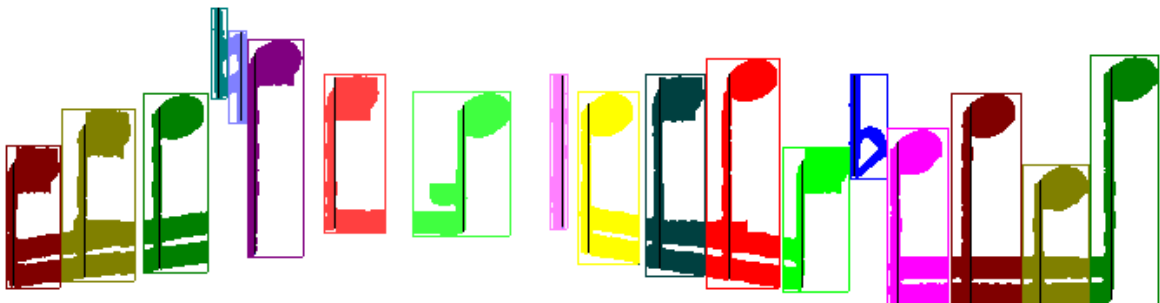
(a) Image après effacement des lignes de portée



(b) *Étiquettes obtenues par croissance de région à partir du segment vertical et boîtes englobantes. Certains symboles sont détectés deux fois (bécarre, barre de mesure) et conduisent à des boîtes englobantes identiques si le symbole est bien isolé. En revanche, les cas des symboles connectés conduisent à des boîtes englobantes distinctes qui se chevauchent (cas du bécarre et du bémol).*



(c) *Étiquettes et boîtes englobantes après fusion. Les boîtes englobantes similaires sont fusionnées. Ainsi la double détection de la barre de mesure est bien résolue. En revanche, le bécarre est toujours détecté deux fois, à cause de la connexion avec la note suivante.*



(d) *Étiquettes et boîtes englobantes après séparation. Les notes incluses dans des groupes sont bien séparées. Le bécarre est scindé, défaut qui devra être résolu dans les étapes ultérieures, mais il est bien déconnecté de la note suivante.*

Figure 3.26: Segmentation des symboles caractérisés par un segment vertical, cas d'une partition présentant de nombreuses connexions parasites

Les symboles isolés (figures 3.25b et 3.26b) sont correctement délimités par la boîte englobante. En revanche, pour les groupes de notes, les limites de chaque rectangle dépendent du degré de proximité des notes. Lorsque deux notes consécutives sont très proches, alors les deux boîtes englobantes se chevauchent. Il faut donc ajouter une étape permettant de les séparer. Les connexions parasites entre symboles voisins syntaxiquement séparés conduisent aussi à des chevauchements. Enfin, lorsque deux segments ont été détectés par symbole, typiquement dans le cas des bécarrés et des dièses, de certaines barres de mesure ou hampes épaisses, alors on obtient deux rectangles similaires et superposés. Deux traitements supplémentaires sont donc réalisés pour affiner ces premiers résultats.

Une première règle est appliquée pour fusionner des boîtes englobantes provenant des segments verticaux indicés par s et s' et correspondant à un même symbole. Lorsque les côtés des rectangles sont situés à des ordonnées identiques, à un tiers d'interligne près, alors les objets sont fusionnés, c'est-à-dire que l'objet s est conservé, alors que l'objet s' est supprimé, puisqu'il est considéré comme identique au précédent. Avec les notations proposées, cette règle est exprimée par les relations suivantes :

$$\begin{cases} |y_g(s) - y_g(s')| \leq s_l / 3 \\ |y_d(s) - y_d(s')| \leq s_l / 3 \end{cases} \Rightarrow \text{fusion} \quad (\text{Eq. 3.31})$$

La fusion fonctionne très bien pour les symboles qui sont bien séparés de leurs voisins : voir par exemple les bécarrés et dièses bien isolés de la figure 3.25c, ou la barre de mesure de la figure 3.26c. Dans le cas de la note dont la hampe a été détectée deux fois (Figure 3.25c), les écarts entre les côtés des rectangles englobants sont égaux à l'écart entre les deux emfans détectés ($s_l/5$), donc bien inférieurs à la marge choisie ($s_l/3$), et la fusion a été par conséquent opérée avec succès. Lorsque deux symboles distincts sont improprement connectés (voir par exemple le bécarré et le bémol de la figure 3.26c), le critère de fusion n'est pas satisfait avec les paramètres choisis : les cadres ne peuvent englober simultanément les deux symboles à cause de la limitation de la croissance de région ($1.5s_l$) et de la distance qui sépare les emfans détectés (supérieure à $s_l/3$). On peut constater néanmoins que, pour ces mêmes raisons, certains cas de double détection ne sont pas correctement résolus : par exemple la dernière note de la figure 3.25c détectée deux fois, par sa hampe et son crochet, ou encore le bécarré de la figure 3.26c, également détecté deux fois, à cause de la connexion à la note suivante. Ces défauts sont cependant préférables aux fusions inappropriées : celles-ci entraînent irrémédiablement la non-reconnaissance du symbole éliminé alors que la sur-détection pourra être résolue ultérieurement, par l'évaluation de règles graphiques et syntaxiques. Les paramètres ont donc été expérimentalement optimisés dans cette optique.

La seconde règle permet de traiter les superpositions de boîtes englobantes consécutives, indicées par s et s' . Un profil horizontal $P(y)$ est calculé sur la zone comprise entre les deux segments verticaux ($y_p(s)$ et $y_p(s')$) et délimitée horizontalement par les côtés les plus extrêmes des boîtes englobantes (Figure 3.27). Le minimum est ensuite calculé. Soient y_1 et y_2 les ordonnées telles que :

$$\forall y < y_1, P(y) > \text{Min}_y(P(y)), \forall y > y_2, P(y) > \text{Min}_y(P(y)) \quad (\text{Eq. 3.32})$$

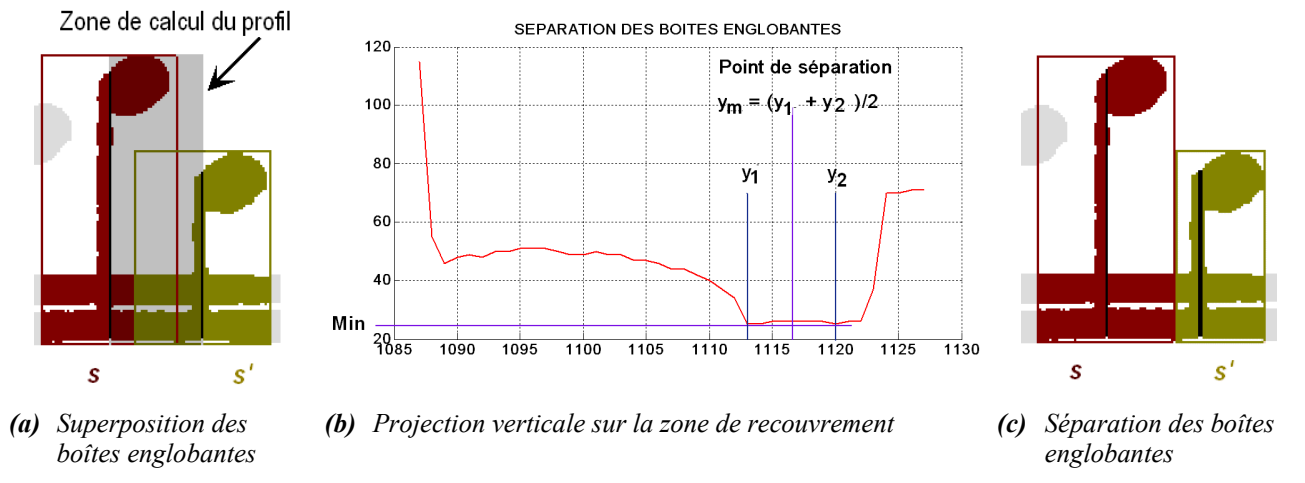


Figure 3.27 : Séparation des boîtes englobantes

Le point de séparation y_m est choisi exactement au milieu. Les ordonnées des boîtes englobantes sont remises à jour suivant cette valeur, la distance entre le côté vertical et l'empan vertical étant toujours limitée à $1.5s_l$:

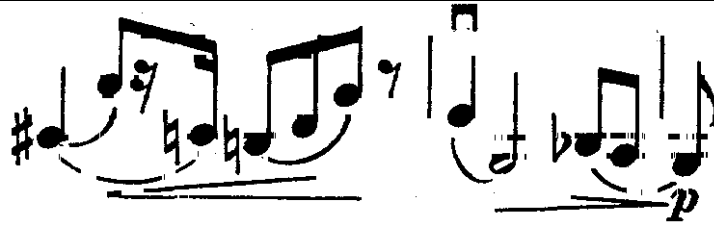
$$\begin{aligned} y_d(s) &= \text{Min}(y_m, y_p(s) + 1.5s_l) \\ y_g(s') &= \text{Max}(y_m, y_p(s') - 1.5s_l) \end{aligned} \quad (\text{Eq. 3.33})$$

Les figures 3.25d et 3.26d illustrent les résultats finals obtenus. Ceux-ci sont satisfaisants puisque les symboles sont bien délimités par une boîte englobante, à l'exception des cas de détections multiples non résolus qui se traduisent par un fractionnement du symbole concerné. Mais soulignons de nouveau que la méthode de classification choisie et l'évaluation de règles graphiques permettront néanmoins de résoudre ces ambiguïtés.

3.2.3. Images des silences

La croissance de région est ensuite poursuivie, en relâchant le critère d'arrêt portant sur la distance maximale à l'empan vertical. On obtient ainsi une image qui contient tous les pixels connexes aux empan verticaux détectés, notée $I_{seg}^{(i)}$ (Figure 3.28).

La différence entre l'image $I_{sp}^{(i)}$ (portées éliminées) et l'image $I_{seg}^{(i)}$ est ensuite calculée. L'image résultante, notée $I_{sil}^{(i)}$, contient tous les symboles qui ne sont pas caractérisés par un segment vertical, c'est-à-dire les silences, les points et les rondes, ainsi que des signes et inscriptions diverses, et quelques résidus de symboles qui ont été fractionnés lors de l'effacement des lignes de portée. Cette nouvelle image sera utilisée pour la reconnaissance des silences.



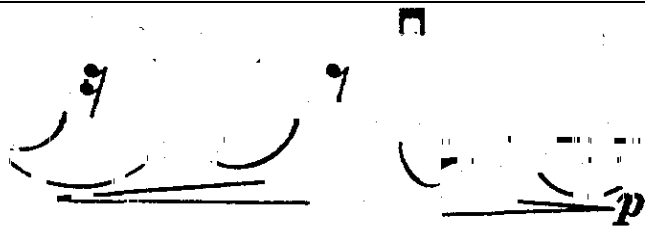
(a) Image après effacement des lignes de portée $I_{sp}^{(i)}$



(b) Image après croissance de région limitée à $1.5s_l$



(c) Image après poursuite de la croissance de région $I_{seg}^{(i)}$



(d) Image des silences $I_{sil}^{(i)}$

Figure 3.28 : Obtention des images de silences

3.2.4. Résultats et conclusion

Nous avons donc présenté une méthode de segmentation qui permet de détecter les symboles caractérisés par un segment vertical, de les délimiter par une boîte englobante, et de créer une image qui contient tous les autres symboles, en particulier les silences, les points et les rondes. La segmentation, à la différence de la plupart des systèmes présentés dans la littérature (paragraphe 1.3.3), ne va pas jusqu'à la décomposition des symboles composés en primitives élémentaires : les composantes d'un groupe de notes ne sont séparées qu'au niveau de la note (tête de note plus hampe), sans localisation des barres de groupe. Notons l'usage, à cet effet, de projections verticales, à l'instar de nombreux auteurs (e.g. [Bellini et al. 01]).

Comme cela a été réalisé dans de nombreux systèmes de la bibliographie (paragraphe 1.3.3), nous avons donc mis au point une méthodologie qui procède par effacement des lignes de portée, et fonde la segmentation sur l'extraction préalable de certaines composantes de l'image. Mais

l'analogie s'arrête à ce niveau, car l'objectif n'est pas d'étiqueter les segments verticaux (e.g. [Kato, Inokuchi 92]), ni d'extraire les primitives composant les symboles par cycles de classification/effacement (e.g. [Ramel et al. 94][Sicard 92]), mais de délimiter la plupart des symboles par une boîte englobante, à partir du segment détecté. Ainsi, il n'y a aucune réelle imbrication entre segmentation et reconnaissance, et gérer de manière rigoureuse l'ambiguïté résultant d'imprécisions de segmentation reste possible.

La détection des segments verticaux a été réalisée avec un soin tout particulier, afin de garantir robustesse et précision de l'extraction. En effet, ces résultats sont à la base de tout le processus de segmentation, et ils sont essentiels à l'analyse des symboles correspondants, comme nous l'expliquerons dans le chapitre suivant. Grâce aux critères choisis, les cas irrattrapables de non-détection d'un symbole sont rarissimes. Néanmoins, la segmentation finale n'est pas toujours parfaite, comme l'illustre la figure 3.29 : élargissement de la boîte englobante (a)(b)(c)(d), due à des inscriptions qui croisent les symboles, telles les liaisons ; réduction ou fragmentation de la boîte englobante à cause de l'effacement de pixels lors de la suppression des lignes de portée (d)(e) ; fragmentation de symboles due à leur double détection combinée avec des connexions parasites (c)(f) ; fausses détections (g). Ces imperfections résultent des défauts d'impression du document original ou sont induites par l'effacement des lignes de portée. Elles ne peuvent être résolues à ce stade de l'analyse, aucune information contextuelle n'étant disponible.

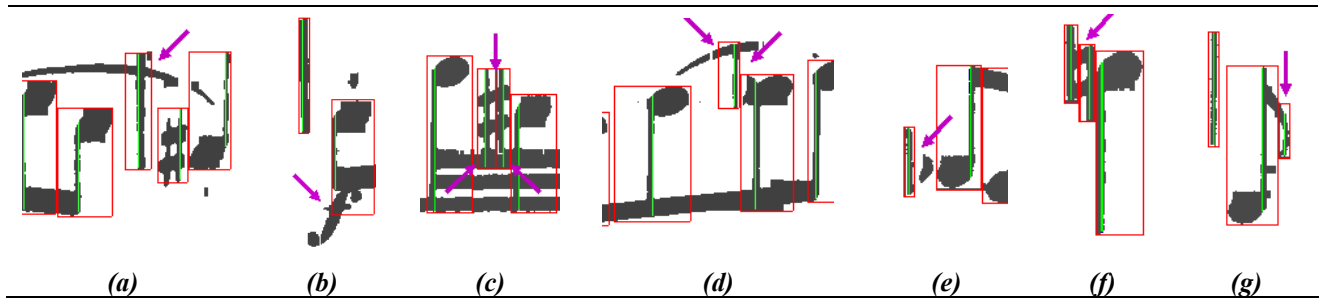


Figure 3.29 : Imprécisions sur les boîtes englobantes

L'ambiguïté résultante sera donc explicitement prise en compte dans les étapes ultérieures :

- en premier lieu au niveau de l'analyse des symboles (chapitre 4) : la méthodologie sera définie sachant que des imprécisions sont possibles. En particulier, toute méthode de classification qui nécessiterait la connaissance exacte des frontières des symboles ne saurait être fiable. C'est pourquoi les boîtes englobantes ne seront pas utilisées pour la classification proprement dite : elles serviront juste d'indicateurs sur les dimensions des symboles caractérisés par un segment vertical et permettront de déduire les zones dans lesquelles des silences et des rondes peuvent être recherchés (paragraphe 4.3).
- En prenant en compte explicitement les imprécisions de segmentation lors de l'étape de modélisation floue : variabilité des symboles accrue à cause de l'effacement des lignes de portée, imprécision sur la position relative des objets, à considérer lors de l'évaluation de règles graphiques.
- En procédant par génération d'hypothèses et évaluation de la cohérence graphique et syntaxique de toutes les combinaisons possibles pour la prise de décision finale. Ainsi les fausses détections et les ambiguïtés de classification seront résolues par l'introduction du contexte complet.

CHAPITRE 4

Analyse individuelle des symboles

La segmentation a conduit à la localisation, par une boîte englobante, des symboles caractérisés par un segment vertical, et à la génération d'une image contenant les silences. Ceux-ci n'ont pas été segmentés, mais, sachant qu'ils se situent dans les espaces libres entre les boîtes englobantes, et, en musique monodique généralement autour de la troisième ligne de portée, les zones de recherche sont finalement assez bien définies.

Nous allons maintenant tenter de classer les symboles. La méthode part du principe qu'une classification exacte ne peut être réalisée en analysant chaque symbole individuellement, étant donné toutes les sources d'ambiguïté (chapitre 2). C'est pourquoi l'analyse présentée ne conduit pas à une décision unique, mais à un ensemble d'hypothèses de reconnaissance. C'est la modélisation des sources d'imprécision et l'intégration des règles musicales qui permettra de lever les ambiguïtés et de choisir la solution correcte parmi toutes les combinaisons d'hypothèses générées (chapitre 5). En ce qui concerne la méthode d'analyse proprement dite, il apparaît clairement qu'elle doit permettre de surmonter les imprécisions de segmentation qui n'ont pu être résolues. Ces considérations ont conduit à choisir de mettre en correspondance les symboles de la partition avec des modèles de classe prédéfinis, par calcul de corrélation.

4.1. Mise en correspondance avec des modèles

Un grand nombre de méthodes peuvent être envisagées pour la classification des objets segmentés, comme nous avons pu le constater dans l'étude bibliographique. La plupart sont fondées sur une sous-segmentation des objets composés (groupes de notes), la reconnaissance des primitives extraites, et leur réassemblage d'après des règles qui expriment la structure des groupes, autrement dit la position relative des primitives [Bainbridge, Bell 03] [Coüason, Camillerapp 94] [Droettboom et al. 02] [Fahmy, Blostein 98] [Kato, Inokuchi 90] [Ng, Boyle 96]. Les primitives elles-mêmes (tête de note, hampe, barre de groupe, crochet) et les autres symboles (silences, points, altérations) sont classés de manières très diverses. On peut distinguer deux grandes catégories : les méthodes structurelles et les méthodes de mise en correspondance de l'image avec des modèles. Les méthodes structurelles sont très présentes dans la littérature : classification d'après un vecteur de caractéristiques géométriques et topologiques [Armand 93] [Carter 89] [Fujinaga 97] [Kato, Inokuchi 90] [Ng, Boyle 96], analyse de profils locaux [Bainbridge, Bell 96] [Fujinaga 88] [Reed,

Parker 96], extraction et analyse de squelettes [Martin 92] [Randriamahefa et al. 93]. Elles nécessitent de connaître précisément la localisation de la forme analysée, et semblent par conséquent très sensibles aux défauts de segmentation, en particulier à la fragmentation. D'autre part, la sous-segmentation des symboles construits en primitives semble extrêmement difficile à réaliser de manière fiable, de même que la résolution des cas de connexions parasites entre symboles syntaxiquement séparés. Toutes ces difficultés paraissent impossibles à résoudre à ce stade de l'analyse, sans aucune information contextuelle. C'est pourquoi certains auteurs imbriquent segmentation et classification dans des algorithmes complexes [Coüasnon, Camillerapp 94] [Ng, Boyle 96], ou introduisent une rétroaction, de manière à revoir certaines décisions après détection d'incohérences durant l'analyse sémantique [Ferrand et al. 99] [Kato, Inokuchi 90] [McPherson, Bainbridge 01]. Le problème principal est que ces méthodes se fondent finalement sur des informations qui restent très locales, et ne prennent donc pas en compte toute l'information contextuelle.

Nous avons donc opté pour le second type de méthode d'analyse, la mise en correspondance de l'image avec des modèles de classe, qui peut être réalisée par corrélation (template matching) [Bainbridge, Bell 96] [Reed, Parker 96] [Martin 92], ou par réseau de neurones [Bellini et al. 01] [Su et al. 01] [Martin 92], et qui présente l'avantage de mieux tolérer les défauts de segmentation. Plus précisément, nous proposons de générer des hypothèses de reconnaissance à partir de scores de corrélation calculés entre les objets de la partition et des modèles de référence, les zones d'analyse étant déduites des résultats obtenus en segmentation. L'équation 4.1 définit la corrélation normalisée entre un modèle M^k de la classe k (Figure 4.1), de dimensions $d_x^k . d_y^k$, d'origine (i_k, j_k) , avec l'objet s à la position (x, y) dans l'image analysée I :

$$C_s^k(x, y) = \frac{I}{d_x^k . d_y^k} \sum_{(i, j) \in M^k} M^k(i, j) . I'(i, j) \quad (\text{Eq. 4.1})$$

avec $M^k(i, j) = \begin{cases} -1 & \text{pour un pixel blanc} \\ 1 & \text{pour un pixel noir} \end{cases}, 0 \leq i < d_x^k, 0 \leq j < d_y^k$

et I' , la sous-image extraite de I , autour de (x, y) , de taille $d_x^k . d_y^k$:

$$I'(i, j) = \begin{cases} -1 & \text{si } I(x + i - i_k, y + j - j_k) = 0 \\ 1 & \text{si } I(x + i - i_k, y + j - j_k) = 1 \end{cases}, 0 \leq i < d_x^k, 0 \leq j < d_y^k$$

En cas de parfaite superposition entre la forme et le modèle, le score de corrélation est maximal et égal à 1. Il décroît avec le nombre de pixels qui diffèrent. Ce score de corrélation est calculé pour différentes positions (x, y) , et seul le plus haut score, noté $C^k(s)$, obtenu à la position (x_k, y_k) , est retenu : il représente le degré de similarité entre le modèle et la forme analysée et permet d'obtenir sa localisation précise :

$$C^k(s) = C_s^k(x_k, y_k) = \max_{(x, y)} C_s^k(x, y) \quad (\text{Eq. 4.2})$$

Les modèles de la figure 4.1 sont définis pour la taille et la résolution d'image considérées, et nous permettent d'éviter des remises à l'échelle. Pour traiter d'autres formats, il conviendra de

définir d'autres ensembles de modèles, qui pourront être choisis en fonction de l'interligne s_l trouvé.

Les origines (i_k, j_k) (en rouge sur la figure 4.1) des modèles M^k ont été choisies ainsi :

- pour les barres de mesure : au centre du modèle. Ainsi l'abscisse x_k doit se situer sur la troisième ligne de portée.
- pour les têtes de note (blanche, noire), et les rondes : au centre du modèle. L'abscisse x_k doit donc se situer sur une ligne de portée ou au milieu de l'interligne. Elle permettra de déduire directement la hauteur d'une note.
- pour les silences : au centre dans la direction horizontale, et au point d'intersection avec la troisième ligne de portée dans la direction verticale.
- pour les altérations : au centre pour le dièse et le bémol ; pour le bécarre au centre de la boucle, et pour les appoggiatures, sur le segment vertical dans la direction horizontale et au milieu de la tête dans la direction verticale. De nouveau, l'abscisse x_k permet de déduire directement la hauteur de ces symboles.
- pour les points : au centre du modèle.

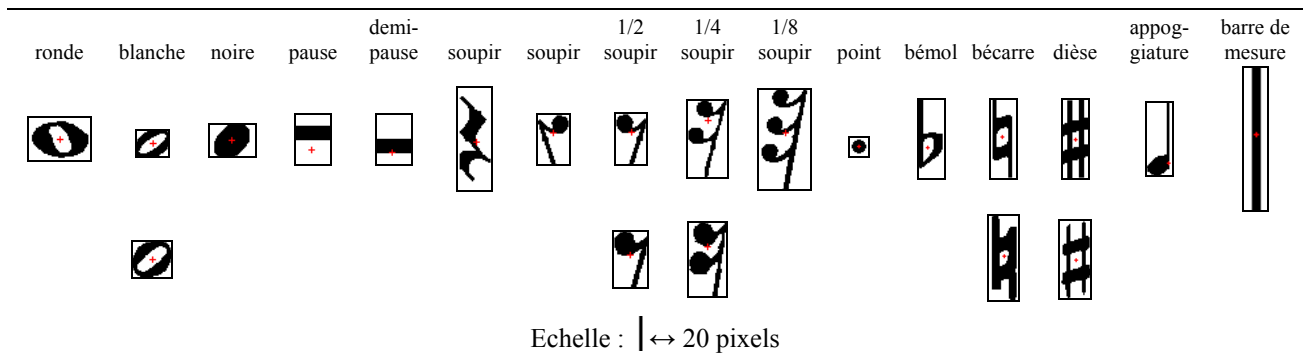


Figure 4.1 : Modèles de références M^k . Un ou deux modèles sont définis par classe k .

La méthode proposée présente un certain nombre d'avantages. Tout d'abord, elle ne nécessite pas de connaître précisément la localisation des formes à reconnaître. En particulier, elle tolère les problèmes de fragmentations ou de connexions parasites : il suffit en effet que l'objet soit détecté et relativement bien localisé pour obtenir des résultats significatifs. Dans notre cas, la détection robuste des segments verticaux assure ce prérequis pour tous les symboles particulièrement sujets à ces défauts (notes, altérations), puisque le score de corrélation peut être calculé sur de petites zones déduites de la position du segment vertical. Par exemple, une tête de note sera recherchée à ses deux extrémités, en étendant suffisamment la zone de calcul pour tolérer une déconnexion éventuelle. Ce mécanisme, complété d'une analyse des barres de groupe (paragraphe 4.2.6), permet de surcroît d'éviter des procédures complexes de sous-segmentation/ reconstruction des groupes de notes.

Il faut également souligner que la connaissance de l'écriture musicale permet de préciser les zones de calcul de corrélation. En effet, la position des symboles est définie dans la théorie musicale par rapport aux lignes de portée : ainsi, les barres de mesure se situent exactement entre la première et la cinquième ligne de portée, les silences, en musique monodique, sont placés autour de la troisième ligne (sauf dans le cas de silences inclus dans des groupes de notes), et les têtes de note aux extrémités de la hampe. En résumé, les zones de calcul de corrélation peuvent être définies en fonction de la localisation du symbole et de la classe k testée, et le processus est ainsi optimisé en

termes de coût de calcul et de fiabilité.

Un autre avantage de la méthode est qu'elle permet d'adapter facilement les modèles de classe à la partition analysée. Il est en effet possible de tester plusieurs modèles par classe, et de choisir le plus adapté. Nous en avons retenu deux pour les symboles qui présentent une forte variabilité (Figure 4.1). De plus, la modélisation floue, proposée dans le chapitre suivant, permet de définir automatiquement des modèles de classe adaptés à la partition traitée, à partir des scores de corrélation obtenus, car ceux-ci indiquent un degré de ressemblance global entre les symboles de la partition et les modèles génériques du programme. Nous verrons également dans le chapitre 6 qu'il est possible d'adapter les modèles M^k eux-mêmes, grâce à un apprentissage supervisé réalisé sur un extrait de la partition, et d'affiner les paramètres du programme pour la reconnaissance de cette partition.

Enfin, la méthode fournit la localisation des symboles dans chaque hypothèse de classification, et ces résultats pourront être exploités pour l'évaluation de règles graphiques.

Nous allons maintenant décrire plus en détail les différentes phases de l'analyse individuelle des symboles. Nous distinguerons l'analyse des symboles caractérisés par un segment vertical, qui ont été localisés par une boîte englobante, de l'analyse des autres symboles.

4.2. Analyse des symboles caractérisés par un segment vertical

Les segments verticaux ont été extraits et les symboles correspondants ont été délimités par des boîtes englobantes. Ces différentes informations sont utilisées en préclassification, de manière à éviter des tests incohérents par rapport à la connaissance a priori que nous avons des symboles musicaux et de leur position sur la portée. Chaque objet s est ensuite corrélé avec les modèles des classes jugées possibles, sur des zones définies pour chaque classe en fonction de la position du segment vertical et de la théorie musicale (structure et position du symbole sur la portée). Les scores de corrélation obtenus pour chaque objet s conduisent à la génération d'hypothèses de reconnaissance. Les modèles de classe (Figure 4.1) sont génériques et utilisés pour toutes les partitions, quelle que soit l'édition. Seul le modèle de barre de mesure est défini de manière dynamique en fonction des caractéristiques des portées analysées.

4.2.1. Préclassification

Les boîtes englobantes fournissent des informations très intéressantes sur les dimensions des objets. Il a été montré que celles-ci sont caractéristiques de la classe du symbole, et qu'elles peuvent être utilisées comme paramètres discriminants [Préreau 70] [Fujinaga 97] [Carter 89] [Ng, Boyle 96]. Etant donné les imprécisions inévitables que nous avons constatées, boîtes englobantes et segments verticaux ne sont utilisés qu'en préclassification, de manière à éviter des corrélations incohérentes, coûteuses en calcul et génératrices d'ambiguïté. Cette préclassification sera d'ailleurs relâchée si aucune des classes testées ne donne de résultats significatifs. Quatre groupes ont été définis : les symboles de type altération (bémol, dièse, bécarre, appoggiature), les symboles de type note (blanche, noire, croche, etc.), les symboles de type soupir (1^{er} modèle de soupir de la figure 4.1), et

les barres de mesure. Tout symbole caractérisé par un segment vertical appartient à l'un de ces groupes. Rappelons les notations relatives aux paramètres extraits lors de la segmentation (Figure 4.2) : extrémités du segment vertical $(x_{ph}(s), y_p(s))$ et $(x_{pb}(s), y_p(s))$, coins supérieur gauche $(x_h(s), y_g(s))$ et inférieur droit $(x_b(s), y_d(s))$ de la boîte englobante (paragraphe 3.2.2). La position des lignes de portée est également connue (Eq. 3.23) au niveau de l'objet s ($x_{FO}^{(i)}(y_p(s)) + ks_I, k \in [-2, 2]$).

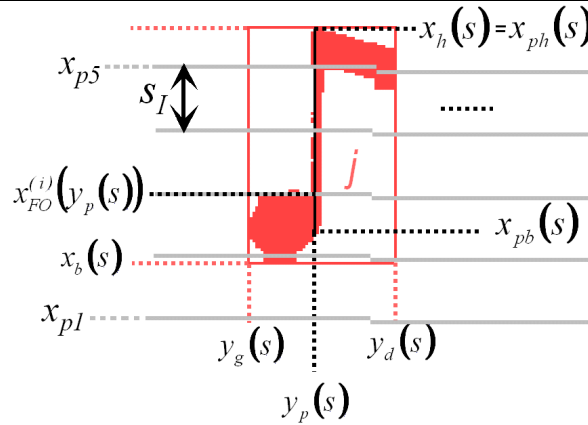




Figure 4.2 : Paramètres extraits lors de la segmentation

Les critères de préclassification portent sur la hauteur et la largeur des boîtes englobantes, sur la longueur du segment vertical, sur la position de l'objet par rapport à la portée. Cinq critères ont été définis par groupe. Certains sont stricts, c'est-à-dire qu'ils doivent être absolument vérifiés pour que l'objet appartienne au groupe, les autres non. Si le nombre de critères satisfaits est supérieur ou égal à trois, et que toute condition nécessaire est vérifiée, alors la corrélation est effectuée sur tous les modèles du groupe. Les tableaux ci-dessous résument les critères définis pour chaque groupe, avec en grisé les critères obligatoires. Dans ces tableaux, on notera, pour plus de lisibilité (Figure 4.2), x_{p1} la position de la ligne de portée inférieure au niveau de l'objet considéré ($x_{p1} = x_{FO}^{(i)}(y_p(s)) + 2s_I$), et x_{p5} la position de la ligne de portée supérieure ($x_{p5} = x_{FO}^{(i)}(y_p(s)) - 2s_I$).

| | Groupe Barres de mesure (B) | |
|-----------|--|---|
| C1 | $ x_{pb}(s) - x_{p1} < 0.2s_I$ OU $(H - x_{pb}(s)) \leq \text{Max}(0, \tan(\theta)y_p(s))$ | Segment connecté à la ligne inférieure ou au bas de l'image (barre de système) |
| C2 | $ x_{ph}(s) - x_{p5} < 0.2s_I$ OU $x_{ph}(s) \leq \text{Max}(0, \tan(\theta)y_p(s))$ | Segment connecté à la ligne supérieure ou au haut de l'image (barre de système) |
| C3 | $(y_d(s) - y_g(s)) < 0.6s_I$ | Objet étroit |
| C4 | $ x_h(s) - x_{p5} < 0.2s_I$ | Boîte englobante connectée à la ligne de portée supérieure |
| C5 | $ x_b(s) - x_{p1} < 0.2s_I$ | Boîte englobante connectée à la ligne de portée inférieure |

|  | Groupe Notes (N) |
|---|---|
| C1 | $(x_{pb}(s) - x_{ph}(s)) \geq 2s_I$ <i>Longueur minimale du segment</i> |
| C2 | $(x_b(s) - x_h(s)) \geq 3s_I$ <i>Hauteur minimale du cadre</i> |
| C3 | $(y_d(s) - y_g(s)) > 1.2s_I$ <i>Largeur minimale du cadre</i> |
| C4 | $ x_{pb}(s) - x_{pl} \geq 0.2s_I$ OU $ x_{ph}(s) - x_{ps} \geq 0.2s_I$ <i>Segment non connecté aux lignes de portée inférieure et supérieure simultanément</i> |
| C5 | $((x_b(s) - x_{ps}) > 0.2s_I$ ET $(x_{ps} - x_h(s)) > 0.2s_I$) OU $((x_b(s) - x_{pl}) > 0.2s_I$ ET $(x_{pl} - x_h(s)) > 0.2s_I$) ET $(x_{pb}(s) - x_{ph}(s)) > 3s_I$ <i>Intersection avec la ligne supérieure ou inférieure de la portée, avec longueur minimale du segment</i> |

|  | Groupe Silences (S) |
|---|--|
| C1 | $x_{ps} \leq x_{ph}(s)$ ET $x_{pl} \geq x_{pb}(s)$ <i>Objet centré sur la portée</i> |
| C2 | $(x_{pb}(s) - x_{ph}(s)) < 3s_I$ <i>Longueur maximale du segment</i> |
| C3 | $(x_b(s) - x_h(s)) < 3.5s_I$ <i>Hauteur maximale du cadre</i> |
| C4 | $(y_d(s) - y_g(s)) < 1.3s_I$ <i>Largeur maximale du cadre</i> |
| C5 | $(0 < (x_h(s) - x_{ps}) < s_I)$ OU $(0 < (x_{pl} - x_b(s)) < s_I)$ <i>Cadre assez proche de la ligne de portée supérieure ou inférieure</i> |


|  | Groupe Altérations (A) |
|---|---|
| C1 | $(x_{pb}(s) - x_{ph}(s)) < 3.5s_I$ <i>Longueur maximale du segment</i> |
| C2 | $(x_{pb}(s) - x_{ph}(s)) \geq 1.5s_I$ <i>Longueur minimale du segment</i> |
| C3 | $(x_b(s) - x_h(s)) < 4s_I$ <i>Hauteur maximale du cadre</i> |
| C4 | $(y_d(s) - y_g(s)) < 1.3s_I$ <i>Largeur maximale du cadre</i> |
| C5 | $((x_b(s) - x_h(s)) - (x_{pb}(s) - x_{ph}(s))) < s_I$ <i>Longueur du segment peu différente de la hauteur du cadre</i> |

Tableau 4.1 : Critères de préclassification

L'objectif n'est pas tant de préclasser les objets, que d'éliminer d'emblée des hypothèses impossibles. C'est en particulier le rôle des conditions nécessaires : en effet, il est inutile de chercher une barre de mesure si le segment vertical n'est pas connecté aux lignes extrêmes de portée, ou de rechercher un soupir au-dessus de la portée. Les autres critères n'ont pas à être tous simultanément vérifiés, soit parce que ce n'est généralement pas le cas (de manière évidente, une note ne se positionne pas toujours sur la portée comme défini en C5), soit parce que c'est la conséquence d'un défaut de segmentation que l'on peut tolérer : si une liaison croise une barre de

mesure, alors le point C3 portant sur la largeur maximale du cadre englobant ne sera pas vérifié. Mais on peut espérer que ses côtés supérieur et/ou inférieur (C4, C5) soient quand même sur les lignes de portée extrêmes, et que le nombre de critères satisfaits soit finalement suffisant.

En même temps, les critères sont suffisamment discriminants pour différencier les objets. Par exemple, une altération peut remplir la condition nécessaire C1 du groupe notes (segment plus long que $2s_l$), mais ne pas satisfaire à au moins trois des critères restants, puisque les caractéristiques de taille indiquées pour les altérations et pour les notes sont plutôt antinomiques.

Il peut arriver que des défauts de segmentation, souvent dus à des connexions parasites avec des objets voisins, faussent totalement les informations sur la dimension de l'objet, et qu'un symbole ne soit pas admis dans le groupe qui lui correspond. Dans ce cas, les scores de corrélation obtenus sur les modèles testés s'avèreront insuffisants, et l'objet sera corrélé avec tous les modèles M^k .

La figure 4.3 illustre la segmentation opérée sur plusieurs extraits de partitions, et les résultats de préclassification obtenus. On constate que ceux-ci permettent effectivement de restreindre les corrélations aux classes pertinentes, même en cas d'imprécision sur les boîtes englobantes. Nos expérimentations, menées sur toute la base de données, montrent que le nombre de corrélations effectuées est considérablement réduit, d'un facteur 2. On a également pu vérifier la robustesse par rapport aux nombreux paramètres qui ont été définis (chapitre 7, section 7.3.4).

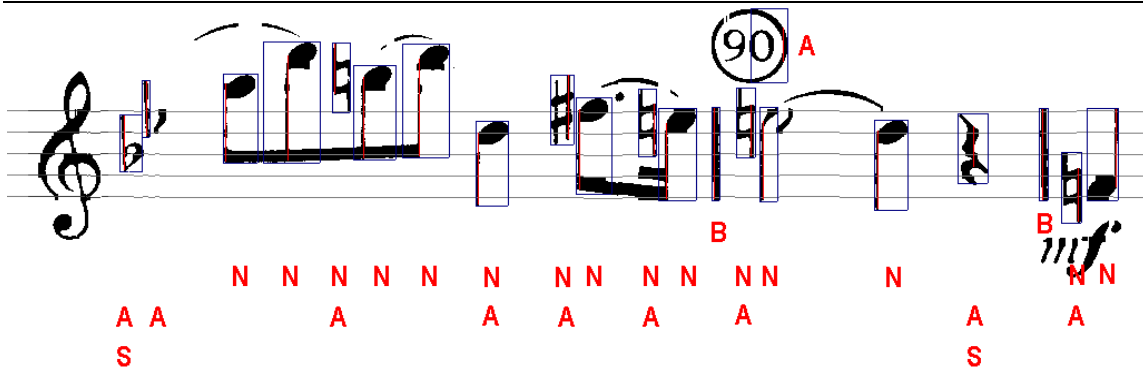
4.2.2. Zones de calcul de la corrélation

Chaque objet s est corrélé avec tous les modèles de classe des groupes dans lesquels il est admis, suivant les équations 4.1 et 4.2. Les images utilisées sont les images sans portées ($I = I_{sp}^{(i)}$ dans l'équation 4.1). Les zones de corrélation, c'est-à-dire les plages de variation de x et y , ont été définies pour chaque groupe, par rapport à la position du segment vertical détecté, $y_p(s)$, $x_{ph}(s)$ et $x_{pb}(s)$, connaissant l'origine (i_k, j_k) des modèles M^k . Notons (x_0, y_0) les coordonnées du centre de la zone, Δx et Δy les plages de variation dans les deux directions autour de cette position centrale. Le tableau 4.2 résume la définition des zones de corrélation.

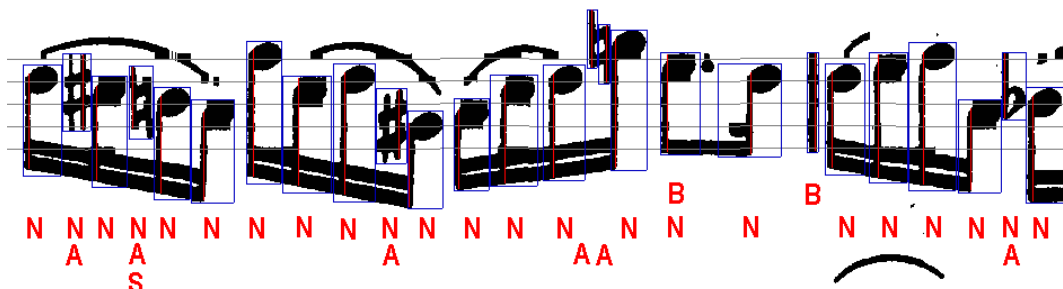
Commentons ce tableau. Les silences (en fait limités aux soupirs) sont recherchés sur la ligne centrale de la portée, à l'ordonnée du segment vertical, avec une plage de variation de $\pm s_l/2$ dans les deux directions. Le même principe est appliqué pour les barres de mesure. Comme le modèle de barre de mesure est déduit des paramètres de la portée (distance entre les lignes extrêmes, paragraphe 4.2.3), et que la position de la portée est précisément connue, aucune variation dans la direction verticale n'est autorisée.

Les têtes de note sont recherchées aux extrémités supérieure droite et inférieure gauche du segment vertical révélant la hampe. Les deux cas ont donc été distingués dans le tableau. La plage de variation Δx dans la direction verticale est assez large, afin de pallier les problèmes éventuels de déconnexion entre hampe et tête de note.

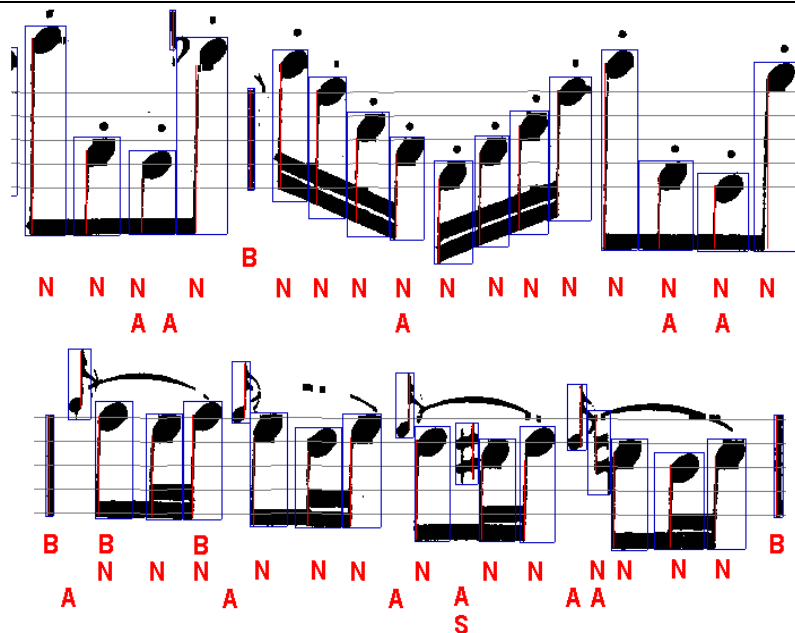
Pour les altérations, on remarque que la plage de recherche dans la direction verticale est également importante, puisqu'elle s'étend sur toute la hauteur du segment vertical. Cela permet de surmonter les problèmes d'effacement partiel (cas du bémol de la figure 3.23) ou des imprécisions dues à des



(a) Cas d'une image de bonne qualité. Les défauts de segmentation sont dus à l'effacement des lignes de portée : fragmentation du second bémol de l'armure et de la blanche. Néanmoins, la préclassification est correcte sur ces objets. Un ou deux groupes seulement sont retenus par symbole, limitant le nombre de corrélations à effectuer et fiabilisant les résultats.



(b) Cas d'une image imprimée en traits épais, avec de nombreuses connexions parasites entre altérations et notes (exemple de la figure 3.21). La segmentation est malgré tout de bonne qualité, conduisant à des présélections qui sont très pertinentes. Le deuxième bémol est détecté deux fois dans le groupe "altérations". Cette préclassification est exacte, mais la double détection devra être résolue ultérieurement.



(c) Cas d'une image imprimée en traits très fins (cf Figure 3.22), avec des biais, des pixels "objet" effacés (hampes, altérations), mais aussi des connexions parasites entre objets : voir par exemple les connexions dues aux liaisons de phrasé, ou la succession appoggiature, bémol, note de la fin de la mesure de la deuxième portée. Malgré quelques imprécisions sur les boîtes englobantes, les préclassifications sont également très pertinentes.

Figure 4.3 : Résultats de préclassification obtenus sur une portée.

Légende : B = barres, N= notes, A = altérations, S = silences

connexions parasites (à une barre de groupe par exemple, dièse de la figure 3.29c). La plage de variation Δy dans la direction horizontale est grande pour les dièses et bécarrés ($s_I/2$), puisque le segment détecté peut être à droite ou à gauche du centre du symbole, mais très faible pour les bémols ($s_I/8$) qui ne présentent pas cette ambiguïté. Pour les appoggiatures, la zone de recherche se situe à l'extrémité inférieure du segment vertical.

| | x_0 | y_0 | Δx | Δy |
|---|--|----------|--|--------------------|
| Barres de mesure | $x_{FO}^{(i)}(y_p(s))$ | $y_p(s)$ | 0 | $s_I/5$ |
| | $x = x_0$ | | $y_0 - \Delta y < y < y_0 + \Delta y$ | |
| Notes | $x_{ph}(s)$ | $y_p(s)$ | s_I | $3s_I/4$ |
| | $x_{pb}(s)$ | | | |
| | $x_0 - \Delta x < x < x_0 + \Delta x$ | | $y_0 \leq y < y_0 + \Delta y$ si $x_0 = x_{ph}(s)$ $y_0 - \Delta y < y \leq y_0$ si $x_0 = x_{pb}(s)$ | |
| Silences | $x_{FO}^{(i)}(y_p(s))$ | $y_p(s)$ | $s_I/2$ | $s_I/2$ |
| | $x_0 - \Delta x < x < x_0 + \Delta x$ | | $y_0 - \Delta y < y < y_0 + \Delta y$ | |
| Altérations sauf appoggiatures | $\frac{x_{ph}(s) + x_{pb}(s)}{2}$ | $y_p(s)$ | $\frac{x_{pb}(s) - x_{ph}(s)}{2}$ | $s_I/2$ ou $s_I/8$ |
| | $x_0 - \Delta x < x \leq x_0 + \Delta x$ | | $y_0 - \Delta y < y < y_0 + \Delta y$ | |
| Appoggiatures | $x_{pb}(s)$ | $y_p(s)$ | $s_I/4$ | $s_I/10$ |
| | $x_0 - \Delta x < x < x_0 + \Delta x$ | | $y_0 - \Delta y < y < y_0 + \Delta y$ | |

Tableau 4.2 : Zones de calcul de corrélation entre l'objet s et le modèle M^k en fonction de la classe k et de la position du segment vertical

Ces définitions introduisent donc de l'information structurelle dans la méthode de classification ; elles permettent également de surmonter des défauts d'impression, de tolérer une certaine variabilité sur la forme des symboles ou sur leur position par rapport à la portée. Des résultats seront présentés dans le chapitre 7 (section 7.3.4), montrant la robustesse de la méthode par rapport aux paramètres.

4.2.3. Cas des barres de mesure

Les barres de mesure (classe $k=0$) sont extraites comme les autres symboles, par préclassification et corrélation. La différence réside dans le choix du modèle de corrélation M^0 , qui est adapté à la hauteur de la portée analysée, et qui est défini pour différentes épaisseurs. On peut en effet observer une forte variabilité de l'épaisseur des barres de mesure. Or, il est indispensable d'effectuer une détection fiable et non ambiguë de ces symboles, car la modélisation floue des règles de musique et la décision seront appliquées mesure par mesure. La corrélation est donc effectuée autour de chaque segment identifié comme barre de mesure possible en préclassification, sur la zone définie dans le tableau 4.2, pour plusieurs modèles M^0 déterminés comme suit:

$$M^0(i, j) = 1 \quad \text{pour } 0 \leq i < 4s_l + e_0, p_b \leq j < p_b + p_n$$

$$= -1 \quad \text{pour } 0 \leq i < 4s_l + e_0, 0 \leq j < p_b \text{ ou } p_b + p_n \leq j < 2p_b + p_n$$
(Eq. 4.3)

avec $p_b = 3$ et $3 \leq p_n \leq 12$

Il s'agit donc d'un segment noir d'épaisseur p_n variable, précédé et suivi de $p_b=3$ colonnes de pixels blancs. Ces paramètres sont valables pour la taille et la résolution d'image considérées, mais pourraient être exprimés en fonction de l'interligne s_l , par application d'un facteur d'échelle. L'origine est située au centre du modèle. Le score de corrélation final $C^0(s)$ provient de la maximisation sur l'épaisseur p_n et sur l'ordonnée y . L'épaisseur optimale trouvée pourrait être utilisée pour distinguer les barres simples des barres finales plus épaisses, mais cette distinction n'a pas encore été introduite.

4.2.4. Génération d'hypothèses

Nous disposons donc pour chaque symbole s caractérisé par un segment vertical de scores de corrélation $C^k(s)$ avec des modèles de référence M^k . Un ensemble de règles de sélection d'hypothèses est appliqué de manière à retenir au plus trois hypothèses de reconnaissance (H1, H2, H3), avec éventuellement la possibilité qu'il n'y ait pas de symbole (H0). Notons $C^{k1}(s)$, $C^{k2}(s)$ et $C^{k3}(s)$ les trois plus hauts scores obtenus, classés par ordre décroissant. Le tableau 4.3. résume les règles de sélection appliquées :

| | | | |
|-----------|---|---|----------------------|
| | Si $C^{k1}(s) \geq t_a(k_1)$ | Si $t_a(k_1) > C^{k1}(s) \geq t_m$ | Si $C^{k1}(s) < t_m$ |
| H0 | | Pas de symbole (-) | Pas de symbole (-) |
| H1 | Classe du modèle M^{k1} | Classe du modèle M^{k1} | |
| H2 | Classe de M^{k2} si $\begin{cases} t_m \leq C^{k2}(s) \\ (C^{k1}(s) - C^{k2}(s)) < t_a \end{cases}$ | Classe de M^{k2} si $\begin{cases} t_m \leq C^{k2}(s) \\ (C^{k1}(s) - C^{k2}(s)) < t_a \end{cases}$ | |
| H3 | Classe de M^{k3} si $\begin{cases} t_m \leq C^{k3}(s) \\ (C^{k1}(s) - C^{k3}(s)) < t_a \end{cases}$ | Classe de M^{k3} si $\begin{cases} t_m \leq C^{k3}(s) \\ (C^{k1}(s) - C^{k3}(s)) < t_a \end{cases}$ | |

Tableau 4.3 : Règles de sélection d'hypothèses de reconnaissance

Le seuil t_m est le score de corrélation minimal qui doit être atteint pour qu'une hypothèse soit retenue ; le paramètre t_a est un seuil d'ambiguïté qui permet de garder en hypothèses H2 ou H3 les modèles dont les scores de corrélation sont proches du premier. Nous avons fixé $t_m=0.3$ et $t_a=0.3$. Ces valeurs ont été optimisées expérimentalement : pour des valeurs de t_m plus élevées, l'hypothèse correcte est plus souvent éliminée, et pour des valeurs plus faibles, beaucoup plus d'hypothèses sont retenues, alourdissant le coût de calcul. En pratique, les cas d'élimination d'une hypothèse exacte sont très rares. Le choix de t_a résulte d'un compromis similaire. Lorsque le plus haut score de corrélation $C^{k1}(s)$ obtenu est plus faible que le seuil de décision $t_d(k_1)$, alors on autorise la possibilité qu'il n'y ait pas de symbole à cet endroit (-) en H0 dans le tableau 4.3 et dans les tableaux suivants). Les seuils de décision $t_d(k)$ sont définis pour chaque classe k par :

$$t_d(k) = \alpha_k * t_d \text{ avec } t_d = 0.5 \quad (\text{Eq. 4.4})$$

Les coefficients α_k , optimisés expérimentalement, permettent de prendre en compte, pour chaque classe, la sensibilité du score de corrélation aux variations de fonte et la probabilité de fausses détections. Par exemple, α_k est grand (1.3) pour le bémol car le score de corrélation entre le bémol et un objet quelconque peut être élevé, et que ce symbole varie peu. En revanche, α_k vaut 0.9 pour un dièse car il est fréquent que les modèles génériques ne soient pas bien adaptés à la partition, la variabilité des dièses étant en effet très importante. Tous les coefficients α_k sont compris entre 0.8 et 1.4, comme indiqué dans le tableau ci dessous :

| Classe k | α_k | $t_d(k)$ | Classe k | α_k | $t_d(k)$ |
|----------|------------|----------|----------|------------|----------|
| | 1.4 | 0.70 | ♪ | 1.4 | 0.70 |
| ● | 1.1 | 0.55 | ♯ | 0.9 | 0.45 |
| ◊ | 0.8 | 0.40 | ♯ | 0.9 | 0.45 |
| ♯ | 1.3 | 0.65 | ♭ | 1.3 | 0.65 |

Tableau 4.4 : Seuils de décision $t_d(k)$ pour les classes caractérisées par un segment vertical

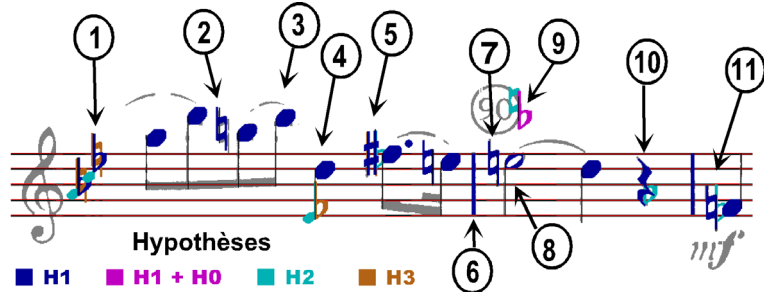
Si aucun des modèles k testés n'aboutit à un score de corrélation supérieur au seuil de décision $t_d(k)$, alors tous les autres modèles sont également testés, et la méthode de génération d'hypothèses est appliquée sur tous les scores de corrélation obtenus. Ainsi, des défauts entraînant une mauvaise définition du cadre englobant puis une préclassification erronée ont moins d'incidence sur les hypothèses générées.

La figure 4.4 illustre les résultats obtenus sur les exemples de la figure 4.3. Les hypothèses de classification sont superposées à l'image originale. Certains scores de corrélation sont également précisés dans les tableaux.

Les résultats sont très bons sur le premier exemple (a) : on peut constater que chaque symbole obtient le score de corrélation le plus élevé avec le modèle de sa classe, et que les seuils de décision $t_d(k)$ suffiraient dans ce cas à prendre la bonne décision. L'ambiguïté est faible car l'image est de bonne qualité et les modèles de classe sont bien adaptés à cette partition. Les résultats obtenus pour le bémol (objet 1) sont significatifs, malgré la fragmentation due à l'effacement des lignes de portée, grâce à la détermination de la zone de calcul par rapport au segment vertical. Il en est de même pour la blanche (objet 8).

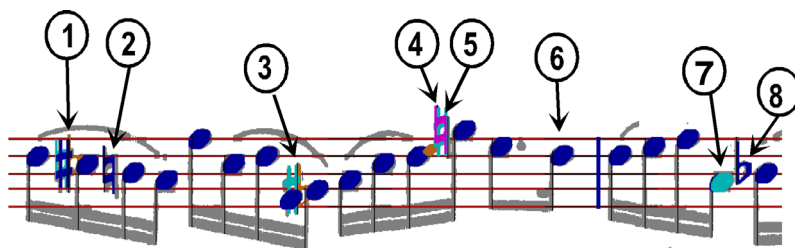
On constate des performances comparables sur le troisième exemple (c) ; notamment l'objet 3 est correctement analysé en dépit de la déconnexion entre la tête de note noire et sa hampe.

En revanche, il y a davantage d'ambiguïté pour la portée (b), pour deux raisons : d'une part, les modèles de classe sont moins ressemblants aux symboles de cette partition, imprimée en traits gras, d'autre part, les imprécisions de segmentation ou la forte proximité de certains symboles conduit à des hypothèses multiples sur certains d'entre eux. Ce problème pourra être résolu par l'introduction des règles graphiques.



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|----|-----------------|-----------------|----------------|-----------------|-----------------|----------|-----------------|--------------|-----------------|-----------------|-----------------|
| H0 | | | | | | | | | (-) | | |
| H1 | \flat 0.78 | \natural 0.86 | \bullet 0.81 | \bullet 0.76 | \sharp 0.66 | $ $ 0.91 | \natural 0.83 | \circ 0.51 | \flat 0.51 | \natural 0.79 | \natural 0.81 |
| H2 | \natural 0.52 | | | \natural 0.66 | \flat 0.62 | | | | \natural 0.35 | \flat 0.52 | \flat 0.62 |
| H3 | \natural 0.50 | | | \flat 0.52 | \natural 0.53 | | | | | | |

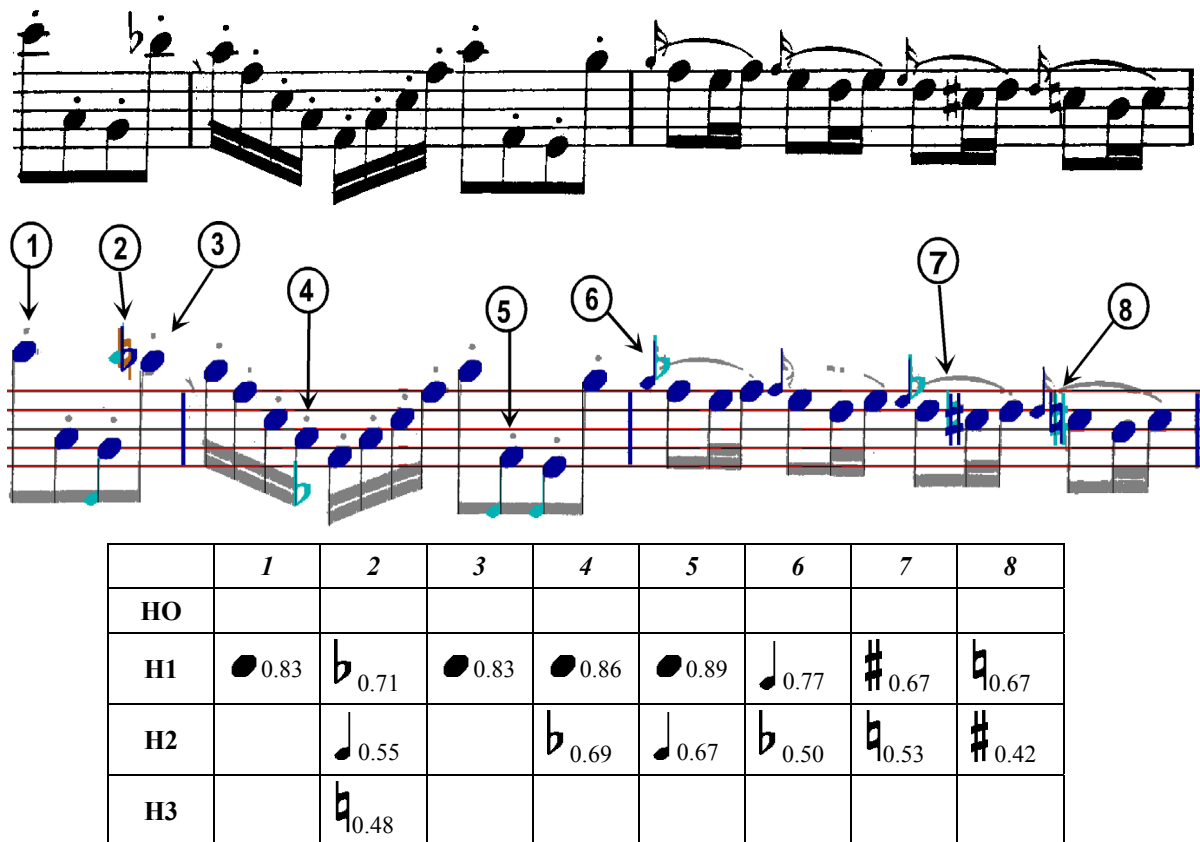
(a) Les hypothèses de reconnaissance sont pertinentes. L'ambiguïté entre les scores de corrélation n'est pas très importante car les modèles de classe sont très ressemblants aux symboles de la partition. Pour l'objet 9, qui ne correspond pas à un symbole devant être reconnu, on a bien une hypothèse H0 autorisant l'absence de symbole à cet endroit.



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----|---------------|--------------|----------------|-----------------|---------------|----------------|----------------|----------------|
| H0 | | | | | | | | |
| H1 | \sharp 0.68 | \flat 0.83 | \bullet 0.58 | \flat 0.77 | \flat 0.77 | \bullet 0.79 | \bullet 0.67 | \flat 0.84 |
| H2 | \flat 0.58 | | \sharp 0.55 | \sharp 0.56 | \sharp 0.56 | | | \bullet 0.60 |
| H3 | \flat 0.48 | | \flat 0.48 | \natural 0.56 | | | | |

(b) On remarque davantage d'ambiguïté entre les scores de corrélation, car les modèles M^k ne sont pas très bien adaptés à cette impression en traits gras. De plus, les connexions entre objets voisins introduisent une sur-classification de certains d'entre eux. Par exemple, les objets 4 et 5, détectés deux fois, sont en fait un unique

bécarre mais aboutissent à des hypothèses de reconnaissance superposées. Ce problème sera facilement résolu grâce à l'introduction de règles graphiques. L'hypothèse H2 faite sur l'objet 8 est une noire, car la zone de corrélation en bas à gauche du segment du bémol comprend la tête de note précédente! De nouveau, cette ambiguïté pourra être résolue lors de la modélisation floue.



(c) Toutes les notes sont bien reconnues, malgré une déconnexion entre la hampe et la tête de note pour l'objet 3, grâce à la définition de la zone de corrélation aux extrémités de la hampe, tolérant ce type de défaut.

Figure 4.4: Génération d'hypothèses de reconnaissance

4.2.5. Analyse de la hauteur des notes et altérations

La hauteur des notes et des altérations s'obtient très simplement par les coordonnées (x_k, y_k) trouvées (Equation 4.2). Connaissant la position de la troisième ligne de portée $x_{F0}^{(i)}(y_k)$, on extrapole toutes les positions possibles, par addition et soustraction d'un multiple du demi-interligne $(s/2)$, et on retient la plus proche de x_k . La hauteur (do, ré, mi, etc.) se déduit de la clé, qui est un paramètre d'entrée de notre programme.

4.2.6. Durée des notes : résultats préliminaires

Trouver toutes les durées est impossible à ce stade de l'analyse, car les barres de groupe n'ont pas été extraites : seules les hampes et les têtes de note sont potentiellement identifiées. Les groupes de notes ne seront intégralement constitués que lors de la modélisation floue, puisque c'est à ce niveau que les symboles ne sont plus considérés individuellement mais les uns par rapport aux autres, et l'analyse des durées sera donc finalisée lors de cette étape.

Cependant, les hypothèses de reconnaissance précédemment générées constituent d'ores et déjà des informations qui permettent d'amorcer la reconstitution des groupes de notes. D'autre part, un point de durée peut être recherché dans le voisinage de chaque tête de note retenue en hypothèse de classification.

Détection des notes reliées par au moins une barre de groupe

Les barres de groupe sont très difficiles à détecter et à classer, car leur taille et leur forme sont très variables. Elles interfèrent aussi largement avec les lignes de portée, et présentent de nombreux défauts d'impression (connexions parasites, déconnexion de la hampe). Elles sont également assemblées de différentes manières, suivant les éditions. La figure 4.5 illustre ces remarques sur les images obtenues après suppression des lignes de portée.



Figure 4.5 : Exemples de barres de groupe, après suppression des lignes de portée

Par conséquent, plutôt que de chercher à segmenter et classer les barres de groupe, nous proposons une méthode qui se contente de vérifier la présence d'un segment d'épaisseur adéquate, qui connecte les extrémités des hampe de toute paire de symboles voisins, classés "noires" en hypothèse H1, H2, ou H3. Des relations de connexité auraient pu être établies d'après la croissance de région (paragraphe 3.2.3). Pour une meilleure précision et une plus grande fiabilité, nous proposons un algorithme spécifique, fondé sur une transformation de Hough modifiée.

Considérons deux segments verticaux, supposés être les hampe de têtes de note noires, s et s' . La position de chaque segment est parfaitement connue (coordonnées $y_p(s)$, $x_{ph}(s)$, $x_{pb}(s)$ et $y_p(s')$, $x_{ph}(s')$, $x_{pb}(s')$ établies en 3.2.2) et la position des têtes de note l'est également (résultats de la corrélation, Eq. 4.2). Les barres de groupe doivent être recherchées à l'extrémité opposée à la tête de note.

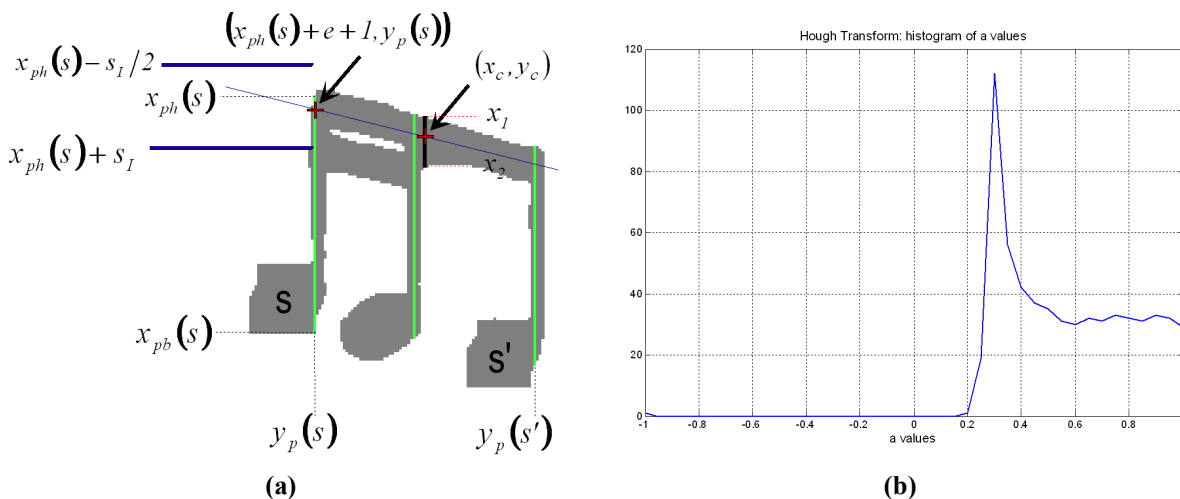


Figure 4.6 : Détection des barres de groupe

Un algorithme de croissance de région est de nouveau appliqué sur l'image sans portée, de gauche à droite, donc pour des ordonnées y_c croissantes. Considérons le cas pour lequel la tête de note est vers le bas. L'ensemble des pixels noirs situés en $y_p(s)$, et dont l'abscisse est comprise entre $(x_{ph}(s)-s_l/2)$ et $(x_{ph}(s)+s_l)$, est utilisé comme germe. Cet intervalle correspond à la position de la barre de groupe externe, avec une marge qui autorise une faible déconnexion de la hampe. Notons $e=s_l/4$ la demi-épaisseur minimale d'une barre de groupe. Soient un pixel noir de coordonnées (x_c, y_c) ($y_p(s) \leq y_c \leq y_p(s')$) aggloméré à la région, et, x_1 et x_2 les abscisses des extrémités de l'empan contenant (x_c, y_c) . Si les distances de ce point aux extrémités de l'empan sont toutes deux supérieures à e , alors la présence d'une barre de groupe centrée sur (x_c, y_c) , et d'épaisseur strictement supérieure à $2e$, peut être envisagée. On calcule donc la pente a de la droite qui passe par le point (x_c, y_c) et par l'extrémité de la hampe de l'objet s (pixel en $(x_{ph}(s)+e+I, y_p(s))$) :

$$a = \frac{x_c - (x_{ph}(s) + e + I)}{y_c - y_p(s)} \quad \text{si} \quad \begin{cases} (x_c - x_1) \geq e \\ (x_2 - x_c) \geq e \end{cases} \quad (\text{Eq. 4.5})$$

Les valeurs de a trouvées sont quantifiées sur l'intervalle $[-1, 1]$ (angles compris entre -45° et 45°), avec un pas de 0.05, et accumulées dans un histogramme (Figure 4.6b). Cet histogramme, dans l'hypothèse où la croissance de région a atteint l'objet s' ($y_c=y_p(s')$), permet de déterminer l'équation $x = a_{opt}y + b_{opt}$ de la ligne médiane de la barre de groupe : il suffit de rechercher l'indice a_{opt} du maximum de l'histogramme et d'en déduire b_{opt} par :

$$b_{opt} = x_{ph}(s) + e + I - a_{opt}y_p(s) \quad (\text{Eq. 4.6})$$

Un dernier critère est testé, afin de valider ces paramètres. Le nombre N_{pn} de pixels noirs situés sur le segment centré sur la droite $x = a_{opt}y + b_{opt}$ et d'épaisseur $2e+I$, est compté. Si le rapport indiqué en équation 4.7 est supérieur à 0.8, alors la présence d'une barre de groupe, reliant s et s' , dans l'hypothèse où il s'agit de noires, est validée et ses paramètres sont mémorisés.

$$q = \frac{N_{pn}}{(y_p(s') - y_p(s) + I)(2e + I)} \quad (\text{Eq. 4.7})$$

Afin d'accroître la fiabilité de la méthode, deux histogrammes sont en fait calculés : le premier, comme indiqué précédemment, et le second de manière similaire, mais en considérant les droites qui passent par l'extrémité de la hampe de l'objet s' . Les paramètres retenus sont ceux qui maximisent le rapport q (Eq. 4.7). Ainsi, il suffit que la barre de groupe soit assez bien connectée à l'une des deux hampes au moins pour être bien détectée, et la robustesse de la méthode est accrue. A noter qu'il peut y avoir d'autres barres de groupe, entre les têtes de notes et la barre analysée (cas des doubles, triples, quadruples croches), mais leur présence n'est pas vérifiée. Les résultats obtenus seront par la suite affinés, lors de la modélisation floue, afin de déduire l'intégralité des groupes de notes dans les différentes configurations d'hypothèses, et d'analyser précisément la durée de chacune des notes qui les composent (paragraphe 5.4.3).

Premières hypothèses de durée

Une première hypothèse sur la durée de la note est néanmoins établie, en considérant chaque hypothèse indépendamment des autres, simplement en dénombrant le nombre de crochets ou de barres de groupe, par analyse de petites sections de part et d'autre de la hampe. Ces sections (Figure 4.7) sont déterminées en fonction du segment vertical ($y_p(s), x_{ph}(s), x_{pb}(s)$), des limites supérieures et inférieures de la boîte englobante ($x_h(s), x_b(s)$), et des coordonnées de la tête de note (x_k, y_k) :

$$y = y_p(s) \pm 0.25s_I$$

$$\text{Si } |x_{pb}(s) - x_k| < |x_k - x_{ph}(s)| : \begin{cases} x_{l1} = \text{Min}(x_{ph}(s), x_h(s)) \\ x_{l2} = \text{Min}(x_k - s_I, x_{l1} + 3s_I) \end{cases}$$

$$\text{Sinon} \begin{cases} x_{l2} = \text{Max}(x_{pb}(s), x_b(s)) \\ x_{l1} = \text{Max}(x_k + s_I, x_{l2} - 3s_I) \end{cases} \quad (\text{Eq. 4.8})$$

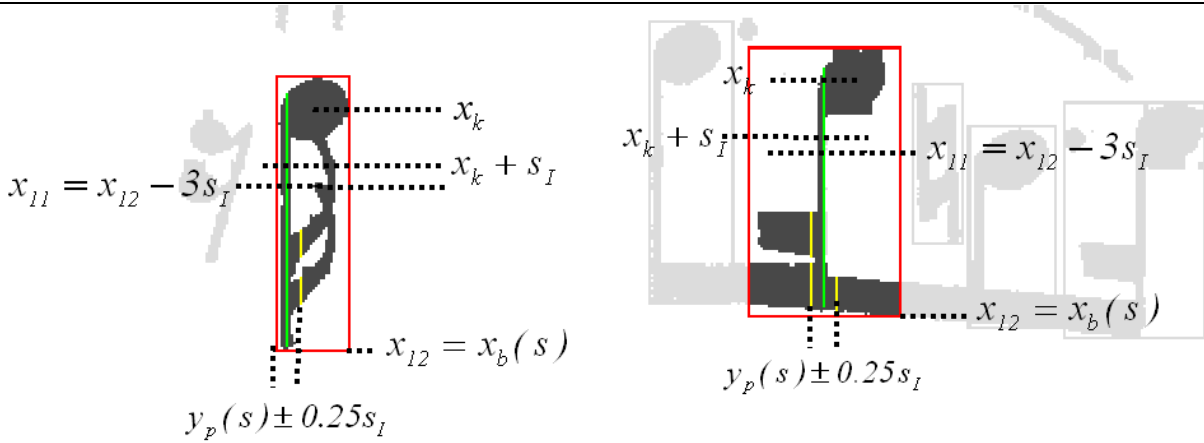


Figure 4.7 : Première estimation de la durée des noirs

Ces résultats préliminaires ne sont pas totalement fiables pour les notes groupées, car la zone d'analyse n'est pas assez précise dans la direction verticale. Néanmoins, ils sont utiles à la reconnaissance des silences, au niveau de la sélection d'hypothèses, comme nous le verrons au paragraphe 4.3. Pour améliorer les performances, les durées seront recalculées pour chaque hypothèse de groupe de notes, lors de la modélisation floue (paragraphe 5.4.3), d'après la position exacte de la barre de groupe externe.

La durée des notes peut être modifiée par la présence d'un point de durée, placé après la tête de note (noire ou blanche). La détection des points de durée est également fondée sur un calcul de corrélation effectué entre l'image analysée et le modèle de point, sur une zone déduite de la position (x_k, y_k) de la tête de note :

$$\begin{cases} x_k - s_I < x < x_k + s_I / 2 \\ y_k + s_I / 2 < y < y_k + 5s_I / 2 \end{cases} \quad (\text{Eq. 4.9})$$

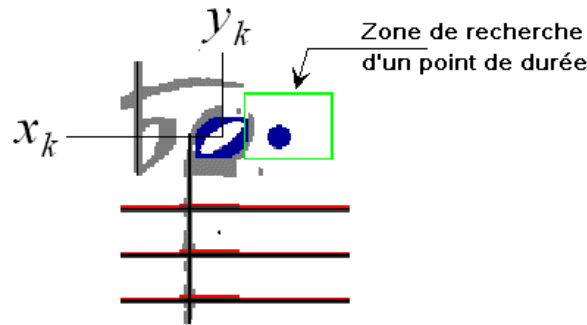


Figure 4.8: Zone de recherche d'un point allongeant la durée d'une note

Deux conditions nécessaires doivent être réunies pour mémoriser un point en hypothèse de reconnaissance : les dimensions de la boîte englobante doivent être inférieures à $0.75s_l$ dans les deux directions, et le score de corrélation doit être supérieure au seuil $t'_m=0.5$. Le premier critère, de présélection, permet d'éviter la détection de points sur des traits (typiquement l'extrémité d'un crochet). Le seuil de décision a été fixé à $t_d(k)=0.6$.

4.2.7. Conclusion

Nous présentons pour la reconnaissance des symboles caractérisés par un segment vertical une méthode qui introduit de la connaissance structurelle a priori (dimensions et structure des symboles, position par rapport aux lignes de portée) en préclassification, mais qui est fondée sur la corrélation avec des modèles de référence. Aucune décision n'est prise : au maximum 4 hypothèses de reconnaissance ont été générées par segment vertical, et la solution finale ne sera choisie qu'après évaluation de leur cohérence, lors de la modélisation floue des règles musicales.

L'intérêt de la présélection d'hypothèses est qu'elle permet de diminuer considérablement le coût de calcul. Comme elle est réalisée dans notre système avec beaucoup de souplesse, elle n'élimine que très rarement la classe correcte, et, dans ce cas, l'erreur peut être rattrapée car les scores de corrélation indiquent qu'il vaut mieux effectuer la comparaison avec tous les modèles M^k . Globalement, l'ambiguïté est considérablement réduite et les résultats de corrélation sont plus facilement interprétables. Les taux de reconnaissance obtenus sur notre base d'images prouvent en effet l'intérêt de cette phase, puisqu'ils chutent de près de 4% si on la supprime.

Il est à noter que les zones de calcul de corrélation sont définies par rapport aux segments verticaux, qui ont été détectés de manière très robuste. On obtient donc des résultats significatifs, même en cas de connexions entre objets voisins, ou de fragmentation de symbole. Les plages d'analyse suffisamment larges dans la direction verticale améliorent encore la robustesse, puisqu'elles autorisent des dégradations aux extrémités du segment vertical. On peut au total affirmer que la méthode est très robuste aux défauts d'impression.

Enfin, soulignons que les notes sont partiellement reconstruites, puisque la hampe a été révélée par la détection du segment vertical, que la tête de note a été recherchée dans les zones admissibles aux extrémités de ce segment, et que des barres de groupe, reliant des noires par paires, ont été identifiées. Des critères de position et de proximité ont été introduits pour arriver à ces résultats, mais aucune décision finale n'est encore prise. Les hypothèses de groupes de notes ne seront générées qu'ultérieurement, lors de la modélisation floue. L'assemblage complet des groupes

de notes, la vérification de la cohérence interne de chacun et par rapport aux autres symboles de la mesure, seront alors réalisés dans chaque configuration d'hypothèses. La méthode permet donc de traiter les symboles composés, sans mécanisme fondé sur des grammaires, et de les valider, non seulement sur des critères locaux d'assemblage, mais aussi de manière globale.

4.3. Analyse des autres symboles

Les symboles restants (silences, rondes) sont recherchés dans les zones libres autour des boîtes englobantes. Les silences se situent toujours autour de la troisième ligne de portée en musique monodique, mais ils peuvent être décalés lorsqu'ils sont inclus dans un groupe de notes. Ce dernier cas ne concerne néanmoins que les demi-soupirs, quarts de soupir et huitièmes de soupir. La première étape consiste donc à définir les zones de présence possible d'un silence ou d'une ronde, la seconde à calculer les corrélations avec les modèles de référence de ces symboles, afin de générer des hypothèses de reconnaissance. La méthode est comparable à celle pratiquée pour les symboles caractérisés par un segment vertical.

4.3.1. Zones de corrélation pour les silences situés sur la troisième ligne de portée et les rondes

Soient s_n et s_{n+1} deux symboles successifs caractérisés par un segment vertical. L'espace séparant leurs boîtes englobantes sera testé pour la détection d'un silence si :

$$y_g(s_{n+1}) - y_d(s_n) > 2s_l \quad (\text{Eq. 4.10})$$

Les limites $[y_{n1}, y_{n2}]$ de la zone de calcul de corrélation sont alors définies par :

$$\begin{aligned} y_{n1} &= y_d(s_n) + s_l / 2 \\ y_{n2} &= y_g(s_{n+1}) - s_l / 2 \end{aligned} \quad (\text{Eq. 4.11})$$

Toute boîte englobante correspondant à un symbole pour lequel aucune classe n'a obtenu un score de corrélation supérieur à son seuil de décision est ignorée : puisque l'on autorise l'hypothèse qu'il n'y ait pas de symbole caractérisé par un segment vertical à cette ordonnée (hypothèse H0), cela signifie également qu'un silence pourrait être présent. Une seule exception est faite à cette règle : lorsque le segment vertical intersecte la troisième ligne de portée et que sa longueur est supérieure à $2.75s_l$, alors on considère que ce symbole ne peut être un silence (i.e. ce n'est pas une fausse détection d'un segment vertical sur un silence, comme illustrée en figure 3.23), et par conséquent aucune nouvelle recherche ne doit être faite.

La figure 4.9 illustre la méthode sur quelques portées. Les boîtes englobantes et les segments verticaux sont indiqués en rouge pour les symboles qui ont obtenu un score de corrélation supérieur au seuil de décision (pas d'hypothèse H0), et en vert pour les autres (H0 permise). Les zones de recherche d'un silence centré sur la portée sont grisées. On constate qu'elles englobent effectivement les silences non inclus dans des groupes de notes. Sur la troisième portée, on constate

également que les boîtes englobantes de l'appoggiature et de la blanche, en vert, ne sont pas grisées, car la longueur du segment exclut la possibilité d'un silence. Au contraire, la zone sous l'indication de tempo de la première portée est autorisée, ce qui permettra de détecter la pause. Certaines zones sont grisées bien qu'aucun silence ne soit présent, car elles vérifient la condition 4.10.

Figure 4.9 : Zones de recherche de silences centrés sur la portée (parties grisées).

4.3.2. Zones de corrélation pour les silences inclus dans des groupes de notes

Les intervalles $[y'_{n1}, y'_{n2}]$ de recherche de silences inclus dans les groupes de notes sont définis d'après les paires d'objets successifs s_n et $s_{n'}$ ($n' > n$), tous deux classés "noires" dans l'un des niveaux d'hypothèses (paragraphe 4.2.4), et interconnectés par une barre de groupe (paragraphe 4.2.6), d'après les critères suivants :

$$y_p(s_{n'}) - y_p(s_n) > 3.5s_l \Rightarrow \begin{cases} y'_{n1} = y_p(s_n) + s_l / 2 \\ y'_{n2} = y_p(s_{n'}) - s_l / 2 \end{cases} \quad (\text{Eq. 4.12})$$

La figure 4.10 illustre la méthode, avec les conventions de la figure 4.9, sur l'image $I_{sp}^{(i)}$ et sur l'image $I_{sil}^{(i)}$. On constate de nouveau que les silences inclus dans les groupes de notes sont bien dans des plages de recherche autorisées.



Figure 4.10 : Zones de recherche de silences inclus dans des groupes de notes

4.3.3. Génération d'hypothèses de reconnaissance (silences et rondes)

Les hypothèses de reconnaissance sont générées de nouveau à partir des scores de corrélation obtenus entre les modèles de référence (Figure 4.1) et l'image $I_{sp}^{(i)}$ pour les silences entre les groupes de notes ainsi que les rondes, et l'image $I_{sil}^{(i)}$ pour les silences inclus dans des groupes de notes, afin que les barres de groupe n'interfèrent pas dans l'analyse.

Tous les modèles de silences sont corrélés pour les ordonnées y incluses dans les intervalles $[y_{n1}, y_{n2}]$, autour de la troisième ligne de portée ($x_0 = x_{F0}^{(i)}(y)$) avec des décalages verticaux de $\pm s_I/2$ pour les soupirs, demi-soupirs, quarts de soupir, huitièmes de soupir, et de $\pm s_I/5$ pour les pauses et les demi-pauses.

$$x_{F0}^{(i)}(y) - \Delta x \leq x \leq x_{F0}^{(i)}(y) + \Delta x \text{ avec } \Delta x = s_I / 2 \text{ ou } \Delta x = s_I / 5 \quad (\text{Eq. 4.13})$$

Les rondes sont recherchées sur toute ligne de portée ou interligne, sur une plage verticale de $\pm s_I/5$:

$$x_{F0}^{(i)}(y) + m \frac{s_I}{2} - \frac{s_I}{5} \leq x \leq x_{F0}^{(i)}(y) + m \frac{s_I}{2} + \frac{s_I}{5} \text{ avec } m \in [-11, 11] \quad (\text{Eq. 4.14})$$

Pour les silences inclus dans les groupes de notes ($[y'_{n1}, y'_{n2}]$), la zone d'analyse est étendue dans la direction verticale (Eq. 4.15) à toute la portée et seuls les modèles de classe demi-soupir, quart de soupir et huitième de soupir sont testés.

$$x_{F0}^{(i)}(y) - 2s_I \leq x \leq x_{F0}^{(i)}(y) + 2s_I \quad (\text{Eq. 4.15})$$

Les scores de corrélation sont mémorisés pour chaque classe et pour chaque ordonnée y dans

un tableau. Les pics de corrélation sont recherchés puis comparés, afin d'extraire des hypothèses de reconnaissance. Le score minimal de corrélation t'_m est fixé à 0.5, le seuil d'ambiguïté t_a à 0.2, et les seuils de décision $t_d(k)$ sont tous égaux à 0.6. La règle de génération d'une hypothèse H0 (absence de symbole) est cependant légèrement modifiée, car elle tient compte du nombre de temps totalisés par les symboles déjà sélectionnés en hypothèse H1, comme nous allons le voir ci-dessous.

La figure 4.11 montre les scores de corrélation obtenus, pour les deux premières portées de la figure 4.9. Pour plus de lisibilité, seules quatre classes sont représentées, et tous les scores de corrélation inférieurs au seuil minimal $t'_m=0.5$ (en rouge sur la figure) sont mis à 0. La ligne magenta représente le seuil de décision $t_d(k)=0.6$. Les pics de corrélation (maxima locaux marqués par une étoile rouge) correspondent effectivement aux silences présents dans l'image, avec cependant des ambiguïtés puisque l'on observe des pics de corrélation voisins pour des classes différentes.

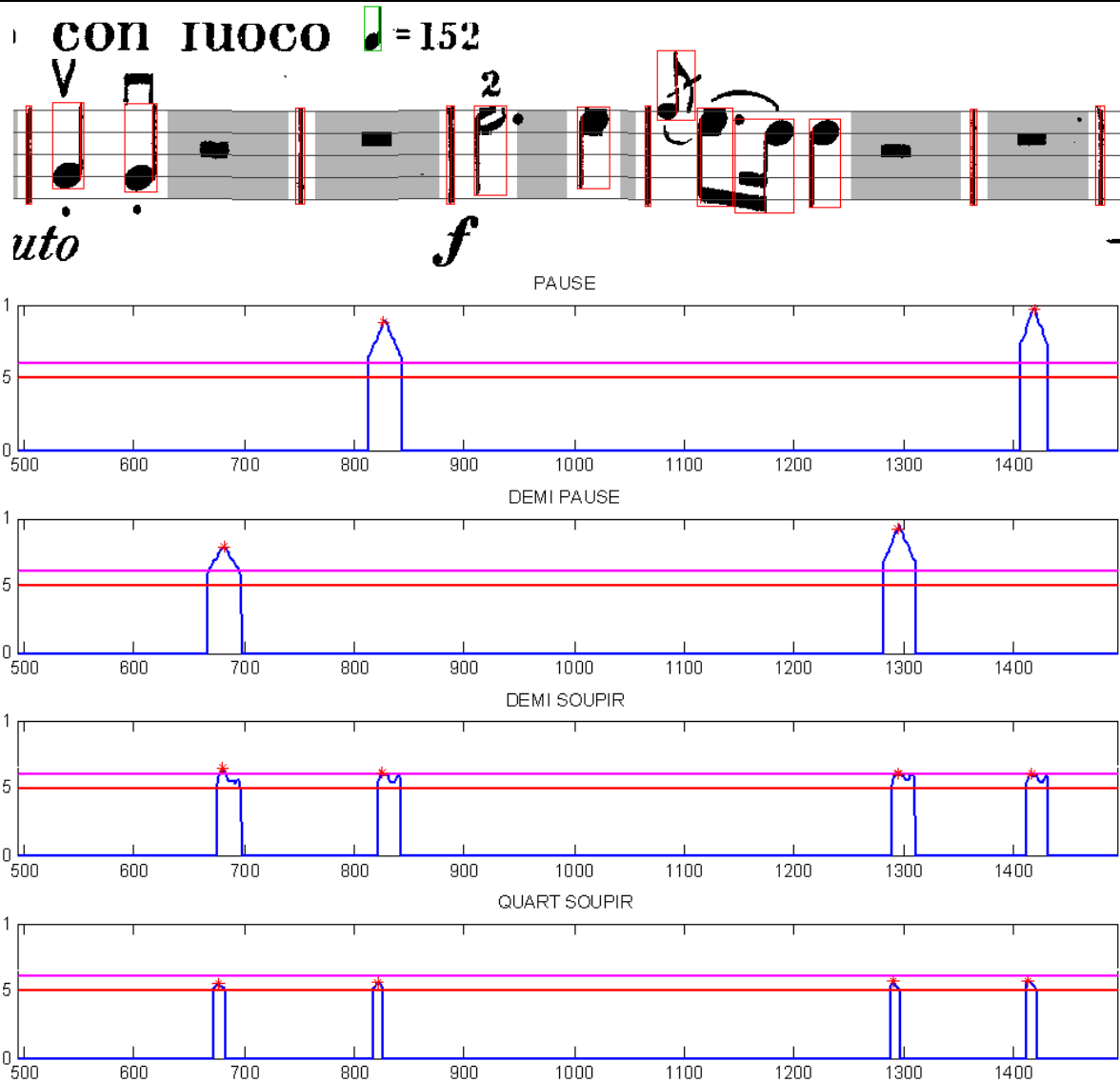
Soit D la durée totale des notes classées en hypothèse H1, et D_m la durée manquante dans la mesure (nombre de temps par mesure auquel on retranche D). Les règles de sélection d'hypothèses sont appliquées de la manière suivante :

- Rechercher le score de corrélation maximal. La classe correspondante est mémorisée en hypothèse H1 si le score de corrélation est supérieur au seuil minimal t'_m . Si le score de corrélation est inférieur au seuil de décision $t_d(k)$, ou que la durée de ce silence est supérieure à la durée manquante D_m , alors la possibilité qu'il n'y ait pas de symbole à cet endroit est mémorisée en hypothèse H0. Dans le cas contraire, la durée du silence est retranchée à D_m .
- Examiner les scores de corrélation obtenus par les autres classes, au voisinage de ce pic de corrélation (même ordonnée à $\pm s_l$). Le second maximum local est mémorisé en hypothèse H2 si la différence entre les scores de corrélation est inférieure au seuil d'ambiguïté t_a (0.3). Le troisième maximum local du voisinage est mémorisé en H3, suivant le même principe. Tout autre maximum local du voisinage sera ignoré.
- Répéter tant que des maxima locaux non traités sont supérieurs au seuil minimal t_m .

Les hypothèses de silences sont donc générées par ordre décroissant de scores de corrélation, et la durée de la mesure est calculée au fur et à mesure, en ne considérant que les hypothèses H1. Bien entendu, cette durée n'est généralement pas exacte, mais c'est un ordre de grandeur utile, qui permet de ne pas mémoriser des hypothèses de silence infondées, sans laisser la possibilité d'une fausse détection. Ce cas se produirait, par exemple, au niveau d'une liaison de phrasé tangente à la quatrième ligne de portée, qui provoquerait un fort pic de corrélation avec le modèle de pause. Cette variante s'est avérée plus efficace que la méthode appliquée pour les symboles caractérisés par un segment vertical, consistant à positionner des seuils de décision variant suivant la classe.

Les hypothèses générées par les règles de sélection sont indiquées dans les tableaux (Figure 4.11). On remarque que la solution correcte est effectivement toujours mémorisée, mais qu'il peut y avoir une assez grande ambiguïté, en particulier entre les demi-soupirs et des quarts de soupir de la seconde portée. Ce phénomène est dû à l'intercorrélation importante qui existe entre ces deux symboles, et il est accentué par la variabilité des polices. Les pauses et les demi-pauses de la

première portée sont en revanche très bien distinguées, bien que ces symboles ne diffèrent que par leur position sur la portée, car celle-ci a été précisément localisée et la zone de corrélation restreinte dans la direction verticale. Le seuil d'ambiguïté suffit à éliminer toute autre classe dans cet exemple.



Hypothèses retenues

| | | | | |
|-------|--------|--------|--------|--------|
| y_k | 680 | 826 | 1293 | 1418 |
| H0 | | | | |
| H1 | ■ 0.80 | ■ 0.89 | ■ 0.94 | ■ 0.97 |
| H2 | | | | |
| H3 | | | | |

Figure 4.11a : Exemples de scores de corrélation obtenus pour les silences (portée 1).

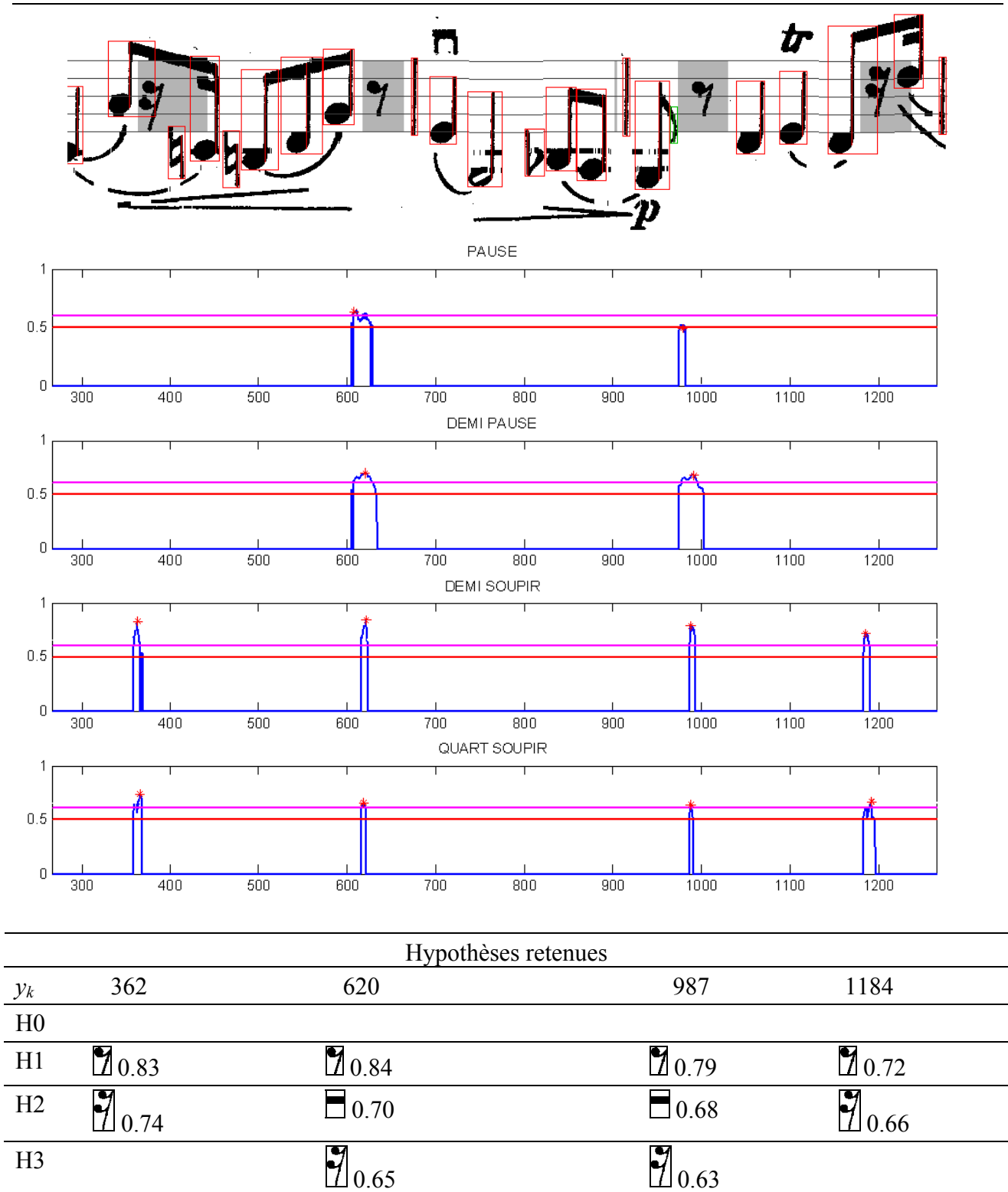


Figure 4.11b : Exemples de scores de corrélation obtenus pour les silences (portée 2).

4.3.4. Points allongeant la durée des silences

La recherche des points allongeant la durée des silences s'effectue, comme pour les notes, en déterminant une zone de corrélation à partir de la position du silence. Celle-ci est définie en

fonction des coordonnées (x_k, y_k) du silence comme suit :

$$\begin{cases} x_k - s_I \leq x \leq x_k \\ y_k + \frac{s_I}{2} \leq y \leq y_k + \frac{7}{4}s_I \end{cases} \quad (\text{Eq. 4.16})$$

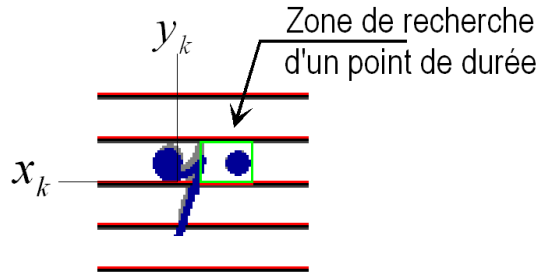


Figure 4.12 : Zone de recherche des points allongeant la durée des silences

Un point sera recherché après une ronde, suivant la méthode proposée pour les noires et les blanches (paragraphe 4.2.6, Eq. 4.9), si la métrique le permet.

Les règles appliquées pour la génération d'une hypothèse sont identiques à celles définies pour des points allongeant les notes : dimensions de la boîte englobante inférieures à $0.75s_I$, score minimal de corrélation $t'_m=0.5$, seuil de décision $t_d(k)=0.6$.

4.3.5. Conclusion

La méthode proposée pour l'analyse des silences est comparable à celle des symboles caractérisés par un segment vertical, puisqu'elle procède par corrélation avec des modèles de référence et génération d'hypothèses de reconnaissance. Il y a cependant une certaine dépendance par rapport aux résultats précédemment obtenus : les zones de corrélation sont définies en fonction des symboles caractérisés par un segment vertical, et la génération d'une hypothèse H0 (absence de silence) dépend du nombre de temps déjà totalisés dans la mesure. Ayant souligné dans l'étude bibliographique la nécessité de ne pas fonder la reconnaissance de certains objets sur celle d'autres objets, il convient donc de justifier ces choix :

- Au niveau de la segmentation : étant donné que les silences sont très bien séparés des autres symboles, le seul cas d'ambiguïté, d'ailleurs très rare, est la détection d'un segment vertical au niveau d'un silence (Figure 3.23). Les deux seuls cas d'erreurs qui en découlent, avec les choix indiqués au paragraphes 4.3.1 et 4.3.2, sont :
 - le segment vertical a conduit à des hypothèses de reconnaissance, sans hypothèse H0 : donc à la reconnaissance a priori certaine d'un symbole caractérisé par un segment vertical. C'est presque impossible avec les seuils de décision choisis.
 - le segment vertical a conduit à des hypothèses de reconnaissance, avec hypothèse H0, mais sa longueur est supérieure à $2.75s_I$. C'est également quasiment impossible.

Ces deux cas de figure sont irrattrapables, mais ils n'apparaissent a priori jamais. Dans tous les autres cas de détection d'un segment sur un silence, les corrélations avec des modèles de silence

sont effectuées, et c'est l'étape de décision qui statuera. La dépendance est donc négligeable.

- Au niveau de la génération d'hypothèses : aucune hypothèse de silence n'est rejetée sur la base d'informations incomplètes ou insuffisamment fiables, comme le nombre de temps totalisés par les notes. Cette information n'influence que la génération d'hypothèses H0 (pas de silence). Toutes les hypothèses, déduites des scores de corrélation, sont maintenues, et c'est encore l'étape finale de décision qui tranchera.

En conséquence, la reconnaissance des silences n'est pas conditionnée par les résultats obtenus sur les autres symboles, essentiellement grâce à la méthode fondée sur la génération d'hypothèses de reconnaissance, sans prise de décision immédiate sur la base d'un contexte limité.

En revanche, il faut noter que la segmentation des silences n'a pas été finalisée, et c'est sans doute un point qu'il faudra améliorer. En effet, il serait aisé, avec les résultats déjà obtenus, de délimiter les symboles par une boîte englobante, par une analyse de connexité, pour restreindre les zones de corrélation. Le coût de calcul serait diminué de manière importante. D'autre part, il serait intéressant de tenter une préclassification, d'après les dimensions et la position de la boîte englobante, de manière à réduire encore le coût de calcul, et à fiabiliser les résultats. En particulier, on pourrait ainsi rejeter rapidement les liaisons et éliminer toute confusion avec des pauses ou demi-pauses. Des ambiguïtés entre pause et demi-soupir, pause et quart de soupir (figure 4.11b), et autres paires de classes de symboles, seraient également supprimées.

4.4. Choix du modèle de classe en fonction de la partition

Les scores de corrélation révèlent un taux de ressemblance moyen entre un symbole de la partition et le modèle. Le principal avantage de cette méthode d'analyse par rapport aux méthodes structurelles est qu'elle permet de mieux surmonter les imprécisions de segmentation. L'inconvénient est que les scores de corrélation chutent rapidement dès que les modèles de classe M^k diffèrent des symboles de la partition, typiquement à cause de la variabilité des polices. Cet aspect est pris en considération dans les règles de sélection d'hypothèses et dans la modélisation floue des classes de symboles (chapitre 5). Cela s'est néanmoins avéré insuffisant pour les classes de symboles qui présentent de très fortes variabilités entre éditions différentes, en particulier pour les blanches, dièses, bécarres, demi-soupirs et quarts de soupir. Un second modèle générique a alors été introduit, de manière à mieux couvrir les différents styles d'édition (Figure 4.1). Afin de ne pas alourdir inutilement les calculs, les deux modèles sont testés simultanément au début de l'analyse, et les scores de corrélation comparés de manière à sélectionner le modèle le plus approprié : dès que l'un des deux modèles a obtenu 5 fois le plus haut score de corrélation, à chaque fois supérieur au seuil de décision, alors il est choisi pour la suite comme modèle de classe unique, et le second modèle est définitivement abandonné.

4.5. Exemples et conclusion

Toutes les hypothèses générées d'après les paragraphes 4.2 et 4.3 sont réordonnées suivant

l'axe horizontal. Deux objets consécutifs, qui ont conduit à des hypothèses de reconnaissance identiques, sont fusionnés, et un certain nombre de cas de double détection (paragraphe 3.2.2 et 3.2.4) sont ainsi résolus. On dispose finalement pour chaque mesure d'un ensemble d'hypothèses, dont la cohérence mutuelle pourra être évaluée par l'introduction des règles de la théorie musicale.

Si nous nous reportons à l'étude bibliographique faite aux paragraphes 1.3.3 et 1.3.4, nous pouvons dégager les points forts de la méthode proposée, qui permettent de gérer l'ambiguïté de manière rigoureuse :

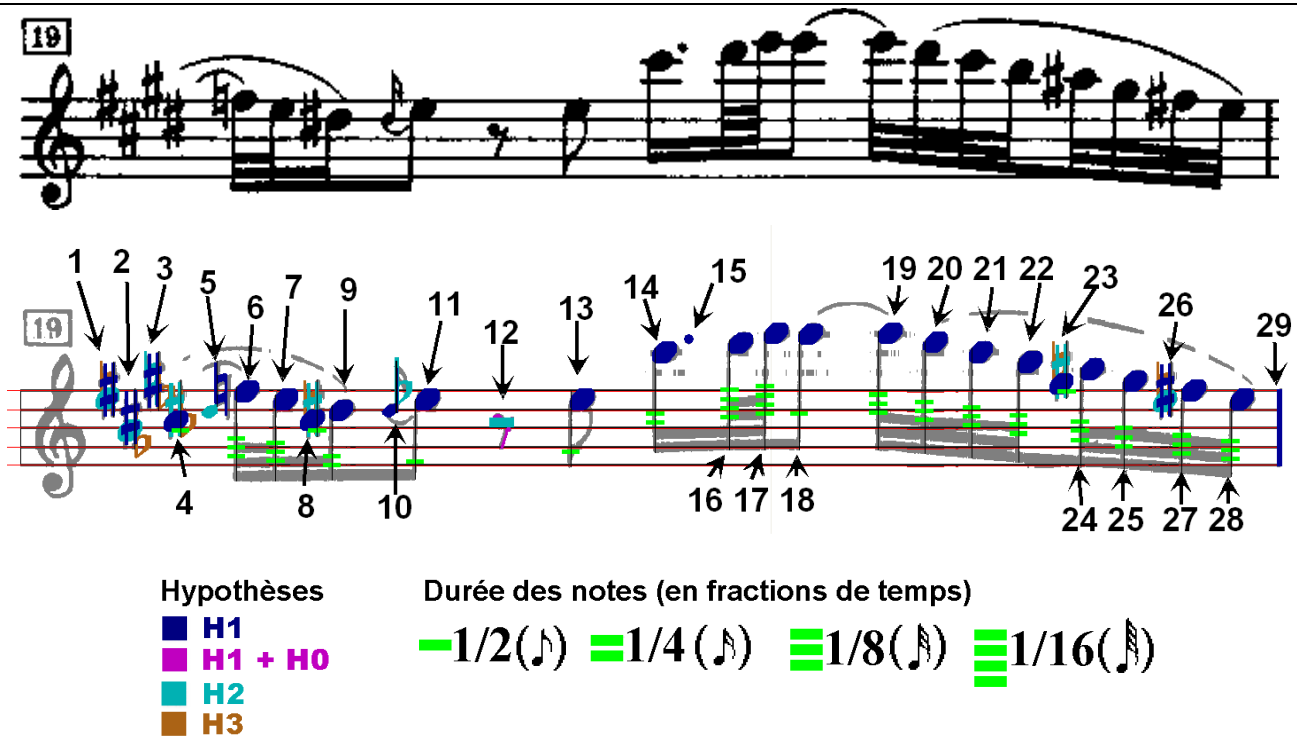
- Il y a indépendance entre les étapes de segmentation et de reconnaissance. L'analyse de l'image est cependant réalisée en prenant en compte d'éventuelles imprécisions de segmentation.
- La connaissance a priori sur la structure et la position des symboles est introduite dans l'analyse, mais de manière souple, et sur chaque objet indépendamment des autres. Aucune règle musicale n'est utilisée pour accepter ou rejeter des hypothèses de classification. Il s'agit donc bien d'une analyse individuelle des symboles, sans introduction ponctuelle d'informations contextuelles incomplètes et incertaines.
- Des hypothèses de reconnaissance sont proposées mais aucune décision n'est prise. On pourra donc évaluer les différentes combinaisons d'hypothèses en introduisant l'intégralité du contexte (règles graphiques et syntaxiques). Notons qu'avec les paramètres choisis, il est très rare que la classe d'un objet ne soit pas retenue (paragraphe 7.2).
- La technique d'analyse est homogène. On dispose ainsi pour chaque hypothèse de reconnaissance d'un score de corrélation et de coordonnées dans l'image, et toutes ces informations pourront servir à l'évaluation finale.

La figure 4.13 illustre, sur une mesure, les hypothèses de reconnaissance obtenues, indiquées en superposition sur l'image originale. Les scores de corrélation sont indiqués dans le tableau. Cet exemple montre qu'effectivement une analyse de plus haut niveau est nécessaire, car les scores de corrélation obtenus pour des hypothèses concurrentes peuvent être dans certains cas très ambigus : ils sont parfois presque égaux et le choix du plus haut score de corrélation ne permet manifestement pas toujours d'obtenir la bonne solution (voir par exemple les symboles 4, 8 et 23). Néanmoins, la solution correcte est dans l'ensemble des configurations possibles.

Il est évident que l'utilisation des règles d'écriture musicale (paragraphe 1.1) peut aider à lever les ambiguïtés et à rejeter des configurations d'hypothèses incohérentes. Par exemple, la règle n°6 sur les altérations à la clé permettra de retenir l'hypothèse correcte H2 (dièse) pour l'objet 4. La règle graphique n°1 sur la position d'une altération par rapport à la note pénalisera les hypothèses qui ne satisfont pas au critère d'alignement, comme l'hypothèse H2 "bémol" pour l'objet 10. Les règles syntaxiques n°6 et n°7 permettront de vérifier la cohérence des altérations accidentelles 5 et 26, entre elles et par rapport à la tonalité. Les règles de métrique n°4 et n°5 conduiront à valider la cohérence des groupes de notes, de manière à rejeter l'hypothèse H1 de croche pour l'objet 23 et à retenir l'hypothèse H1 de demi-soupir pour l'objet 12.

C'est pourquoi toutes ces règles musicales sont modélisées et servent à l'évaluation de toutes les configurations d'hypothèses obtenues sur chaque mesure. On constate aussi, sur cet exemple,

que chaque symbole intervient dans plusieurs règles, et qu'il interagit avec des symboles distants. Cela prouve l'intérêt de ne prendre aucune décision individuelle, mais au contraire de générer des hypothèses de reconnaissance, et de vérifier la cohérence globale par l'évaluation simultanée de l'ensemble des règles musicales. C'est l'objet de la modélisation floue, exposée dans le chapitre 5. Cette modélisation s'appuie sur la théorie des possibilités et des ensembles flous [Dubois, Prade 80], afin de prendre en compte les imprécisions sur les résultats obtenus à l'issue de l'analyse individuelle des symboles ainsi que les degrés de souplesse des règles de musique.



| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| H0 | | | | | | | | (-) | | | | | | |
| H1 | ♩ 0.65 | ♩ 0.73 | ♩ 0.85 | ♩ 0.60 | ♩ 0.83 | ♩ 0.77 | ♩ 0.77 | ♩ 0.74 | ♩ 0.73 | ♩ 0.83 | ♩ 0.65 | ♩ 0.83 | ♩ 0.63 | ♩ 0.85 |
| H2 | ♯ 0.62 | ♩ 0.46 | | ♯ 0.58 | | ♭ 0.48 | | — 0.63 | | | ♯ 0.65 | | ♩ 0.56 | |
| H3 | ♭ 0.57 | | | ♭ 0.54 | | | | | | | ♭ 0.54 | | ♭ 0.53 | |

Figure 4.13 : Hypothèses de reconnaissance sur une mesure

CHAPITRE 5

Modélisation floue

L'analyse individuelle des symboles a abouti à un ensemble d'hypothèses de reconnaissance, attribué à chaque objet segmenté. Nous arrivons maintenant à la phase d'interprétation de haut niveau, dont l'objectif est d'analyser les informations extraites de l'image, de modéliser les règles musicales, et de prendre une décision par optimisation de tous les critères.

5.1. Objectifs

Nous avons identifié dans le paragraphe 1.2 les spécificités de l'édition musicale, sources d'imprécision et d'incertitude : la variabilité des symboles (inter et intra partition), la grande variété des arrangements de notes possibles, le masquage partiel des symboles par les lignes de portée, les défauts liés à l'impression et à la numérisation. Les techniques de segmentation et de reconnaissance ont été définies pour surmonter au mieux ces difficultés, en particulier par l'introduction de nombreuses connaissances a priori relatives à la mise en page, à la structure et à la position des symboles. Néanmoins, une imprécision sur la forme et la position des objets segmentés, et par conséquent une incertitude sur leur classe, ne peuvent être totalement évitées, pour toutes les raisons précédemment évoquées, et les conclusions données dans la littérature convergent sur ce point (e.g. [Ng, Boyle 96] [Watkins 96] [Fahmy, Blostein 98] [Bainbridge, Wijaya 99] [McPherson 02]).

Les deux premières étapes, la segmentation et l'analyse individuelle des symboles, permettent cependant de définir un ensemble d'hypothèses de reconnaissance, contenant les symboles recherchés (voir l'évaluation présentée au chapitre 7, paragraphe 7.2). Les résultats sont donc pertinents mais encore insuffisants. Deux nouveaux axes peuvent alors être exploités pour l'extraction de la solution : d'une part analyser les scores de corrélation obtenus sur toute la partition, de manière à mieux définir les modèles de classe et à les adapter à cette partition ; d'autre part modéliser et intégrer les règles musicales qui définissent les relations entre les symboles, afin d'évaluer la cohérence des symboles dans les différentes configurations d'hypothèses.

Ces deux axes sont tout à fait novateurs par rapport à la bibliographie. A notre connaissance, il n'y a pas d'exemple, dans la littérature, de systèmes qui adaptent leurs modèles de classe à la partition traitée. Fujinaga apporte une réponse au problème de la variabilité des polices, par une voie différente, en proposant un système évolutif capable d'apprendre de nouveaux prototypes

[Fujinaga et al. 98] [Sayeed Choudhury et al. 01]. Il faut cependant réaliser un apprentissage hors ligne. En ce qui concerne la modélisation et l'intégration des règles musicales, les méthodes proposées sont généralement fondées sur des grammaires, avec, pour objectifs principaux, la reconstruction des notes et la restitution de la sémantique [Bainbridge, Bell 03] [Baumann 95] [Coüasnon 96a] [Fahmy, Blostein 98]. Les règles modélisées pour la reconnaissance sont essentiellement des règles graphiques, locales, relatives à la structure des symboles, ou au positionnement des attributs des notes par rapport à ces dernières. Les décisions prises sont également très locales, puisqu'elles résultent du test d'un prédicat portant sur des symboles proches. On constate donc généralement les limitations suivantes :

- Toutes les règles musicales ne sont pas modélisées ni intégrées. En particulier, les règles syntaxiques (cohérence des altérations et de la tonalité, organisation rythmique des groupes de notes) ne sont pas testées, la vérification du nombre de temps dans la mesure exceptée.
- La décision ne procède pas de l'évaluation globale de tout le contexte, mais de décisions locales successives.
- L'imprécision et la flexibilité des règles musicales ne sont pas modélisées. Par exemple, la position d'une altération par rapport à une note est déclarée correcte ou incorrecte, alors que l'on constate en pratique des variations notables, voire des positions a priori interdites (chevauchement d'une altération et de la tête de note par exemple).
- L'incertitude sur la classe des primitives est peu prise en compte.

Quelques projets ont tenté de surmonter ces limitations. Watkins propose une grammaire floue, modélisant le caractère graduel des règles graphiques, en remplaçant les prédicats binaires par des fonctions de certitude, et tente de propager l'incertitude jusqu'à la prise de décision [Watkins 96]. Les critères modélisés sont cependant limités à la structure des notes. L'incertitude sur la classe des primitives est prise en compte dans les grammaires, lorsque différentes classes sont proposées pour un même objet [Fahmy, Blostein 98], néanmoins les décisions restent locales et n'intègrent pas tout le contexte. L'incertitude a également été formalisée dans le cadre de la théorie des probabilités par Stückelberg, mais de manière très prospective [Stückelberg, Doerman 99]. Enfin, une architecture bidirectionnelle permet de réviser des résultats obtenus, par détection d'incohérences dans les modules d'interprétation de haut niveau. Mais les systèmes présentés restent également très prospectifs [Stückelberg et al. 97], ou montrent des exemples de corrections locales qui ne font toujours pas intervenir l'ensemble du contexte et n'intègrent pas toutes les règles [McPherson, Bainbridge 01] [Ferrand et al. 99] [Kato, Inokuchi 90].

Les besoins d'approches syntaxiques, pour la réduction de l'ambiguïté, sont maintenant reconnus (e.g. [Kato, Inokuchi 92] [Fahmy, Blostein 98] [Ferrand et al. 99] [McPherson, Bainbridge 01]). Les méthodes proposées jusqu'à présent se heurtent aux difficultés suivantes :

- La difficulté de modéliser l'incertitude relative à l'étiquetage des primitives, et de la propager de bout en bout.
- La difficulté d'aller au-delà de la reconstruction de symboles à partir de primitives, et de proposer des solutions qui intègrent les critères relatifs aux relations entre les symboles. Cette difficulté est liée à la nature des règles musicales (paragraphe 1.2) : leur flexibilité ou leur imprécision, le fait qu'elles peuvent mettre en jeu un grand nombre de symboles proches ou distants, l'hétérogénéité des informations (règles graphiques ou syntaxiques), l'interdépendance des règles, dans le sens où plusieurs règles peuvent s'appliquer sur un même

- symbole, tout en impliquant des ensembles de symboles différents.
- La difficulté de fusionner toutes ces informations afin de prendre une décision globale.

Nous proposons dans ce chapitre une méthode, fondée sur la théorie des ensembles flous et des possibilités [Dubois, Prade 80], qui tente de répondre à ces questions. L'objectif est de prendre en compte l'imprécision des informations extraites de la partition musicale, l'imprécision et la flexibilité des règles musicales, l'incertitude qui en résulte, de modéliser et d'intégrer l'ensemble des règles musicales afin de prendre une décision globale, par optimisation de tous les critères.

La théorie des ensembles flous et des possibilités offre un formalisme bien adapté à notre problématique. Elle permet en effet de représenter et de traiter l'information spatiale imprécise [Bloch 00] [Bloch, Maître 97] [Krishnapuram, Keller 92], de représenter et de fusionner des informations très hétérogènes, issues directement de l'image ou provenant de connaissances génériques [Dubois et al. 99]. Des fonctions d'appartenance et des distributions de possibilité seront définies pour la représentation des classes et des différentes règles musicales, en prenant en compte l'imprécision des informations extraites (la forme et la position des objets), et en modélisant l'imprécision des règles musicales (comme la position relative des symboles, qui est mal définie) et leur flexibilité (comme le rappel non obligatoire d'altérations).

Un autre point fort de cette théorie est qu'elle permet de représenter dans un même cadre des idées de similarité, de préférence, de plausibilité, d'incertitude [Dubois, Prade 01]. Diverses sémantiques sont utilisées dans notre approche. Une sémantique de similarité permet de modéliser les classes de symboles, par comparaison d'un symbole à un prototype de chaque classe. Une sémantique de plausibilité est utilisée pour la modélisation de la position relative des symboles, ou pour l'évaluation de la cohérence des altérations. Une sémantique de préférence permet de modéliser de façon simple et efficace les contraintes souples telles que le regroupement des notes. Enfin, une sémantique de degré de confiance est utilisée dans la phase de fusion, fournissant l'évaluation d'une hypothèse, exprimée comme une affectation d'un groupe de symboles à des classes. La souplesse et la variété des opérateurs de combinaison permettent en effet de fusionner toutes ces informations hétérogènes, bien qu'elles ne jouent pas le même rôle et n'aient pas nécessairement le même poids [Dubois, Prade 80][Bloch 96] [Bloch 03].

Nous pouvons ainsi proposer un système de reconnaissance qui intègre de bout en bout toutes les sources d'imprécision et d'incertitude, afin d'éviter des décisions locales fondées sur un contexte incomplet, et de ne pas perdre d'information. Une décision globale peut ainsi être prise après fusion de tous les éléments d'information, conduisant à une solution cohérente par rapport à la théorie musicale.

La suite de ce chapitre s'organise en 4 parties. Nous décrivons tout d'abord la modélisation floue des classes de symboles, l'évaluation de la cohérence graphique et syntaxique, la fusion et la décision. Nous terminerons par quelques exemples de décisions élaborées sur des mesures particulières, afin d'illustrer l'ensemble de la méthode proposée. Les résultats complets seront présentés au chapitre 7.

Les mesures suivantes (Figures 5.1 et 5.2), extraites d'une même partition, serviront d'exemples tout au long de ce chapitre. Elles présentent des défauts qui font typiquement échouer

les logiciels d'OMR : une variabilité au niveau de la forme et de la position relative des symboles, des connexions parasites entre primitives, des objets dont la signification est ambiguë (les points de staccato qui peuvent être confondus avec des points de durée). Par conséquent, on peut effectivement constater dans le tableau 5.1 une forte ambiguïté des scores de corrélation, d'autant que les modèles génériques (Figure 4.1) ne sont pas très bien adaptés à la fonte de cette partition.

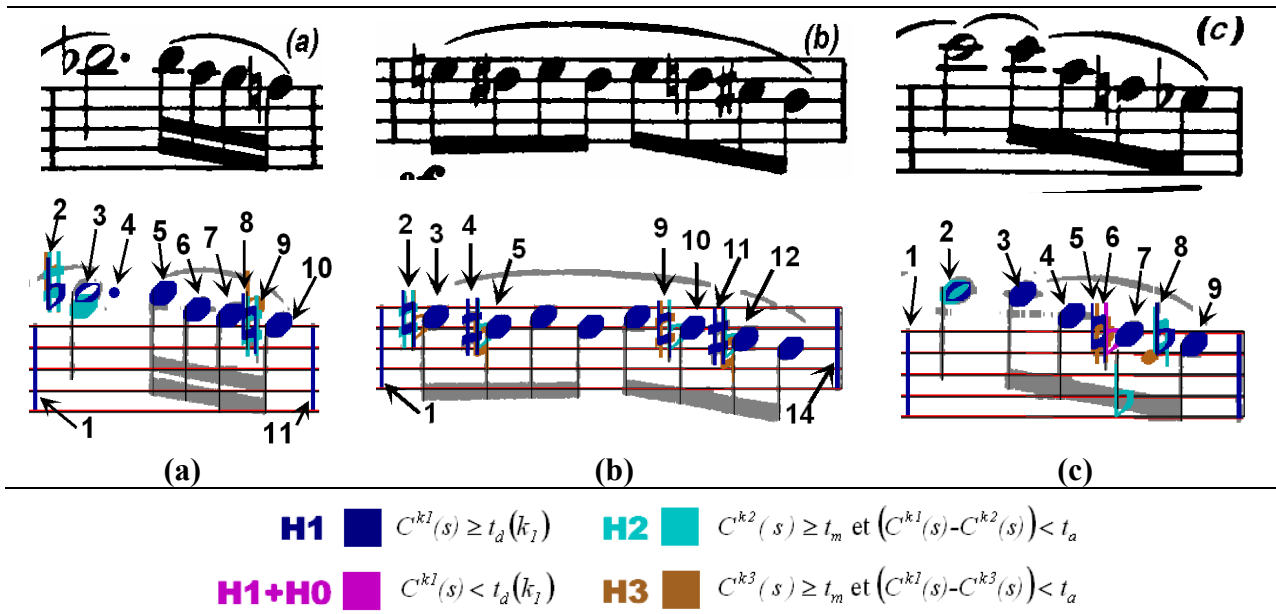


Figure 5.1 : Image source et hypothèses de reconnaissance

| | Mesure (a) | | | Mesure (b) | | | | Mesure (c) | | | |
|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | 2 | 3 | 8 | 2 | 4 | 9 | 11 | 2 | 5 | 6 | 8 |
| H0 | | | | | | | | | | (-) | |
| H1 | <i>b 0.67</i> | <i>o 0.66</i> | <i>b 0.59</i> | <i>b 0.62</i> | <i># 0.58</i> | <i>b 0.59</i> | <i># 0.59</i> | <i>o 0.59</i> | <i>b 0.49</i> | <i>b 0.56</i> | <i>b 0.72</i> |
| H2 | <i># 0.49</i> | <i>o 0.40</i> | <i># 0.49</i> | <i># 0.52</i> | <i>b 0.52</i> | <i>b 0.58</i> | <i>b 0.48</i> | <i>o 0.38</i> | <i># 0.42</i> | <i>b 0.49</i> | <i>b 0.53</i> |
| H3 | <i>b 0.47</i> | | <i>b 0.39</i> | <i>b 0.50</i> | <i>b 0.41</i> | <i># 0.54</i> | <i>b 0.36</i> | | <i>b 0.41</i> | <i># 0.42</i> | <i>o 0.45</i> |

Tableau 5.1 : Hypothèses et scores de corrélation. L'hypothèse correcte est en gras et en italique.

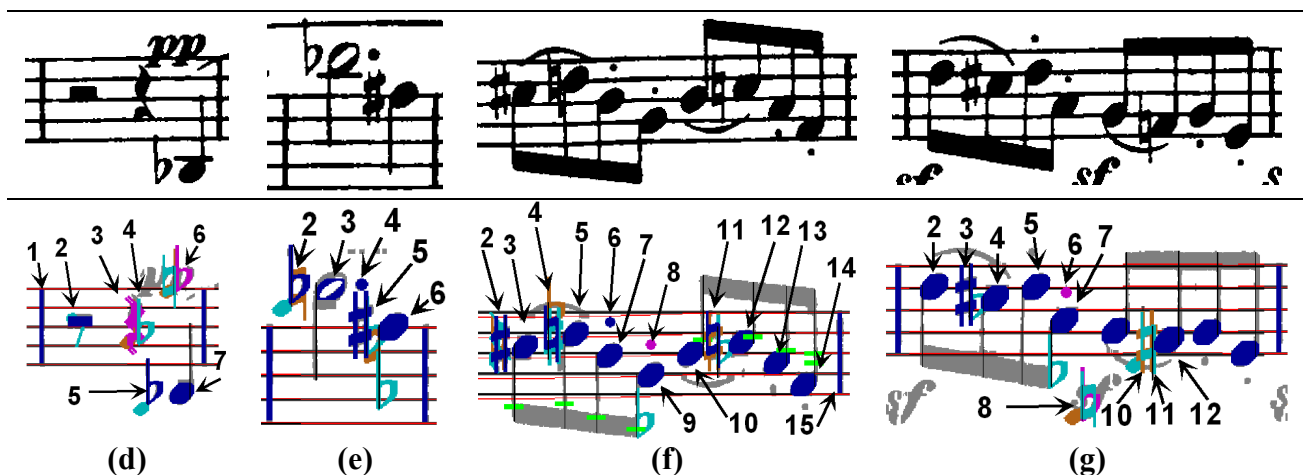


Figure 5.2 : Autres mesures extraites de la partition

5.2. Modélisation des classes de symboles

La modélisation floue des classes de symboles poursuit un double objectif : d'une part, adapter le modèle de classe à la partition analysée, en d'autres termes, traiter le problème de la variabilité inter-partitions (typographies variables), d'autre part prendre en compte les imprécisions sur la forme des objets segmentés. Celles-ci existent dans le document original (variabilité intra-partition, défauts d'impression) et sont de plus amplifiées par l'effacement des lignes de portée (interférences avec les lignes de portée).

L'étape d'analyse individuelle des symboles a conduit, pour chaque objet s , à un ensemble d'hypothèses de reconnaissance, chacune attribuant une classe k à l'objet s , avec les scores de corrélation $C^k(s)$ correspondant (Eq. 4.2). Chaque score de corrélation $C^k(s)$ représente un degré de similarité entre l'objet s et un modèle M^k de la classe k . Le degré de possibilité que l'objet s appartienne à la classe k est d'autant plus élevé que ce score de corrélation est grand. Nous définissons donc, pour chaque classe k , une distribution de possibilité d'appartenance à la classe comme une fonction croissante du score de corrélation :

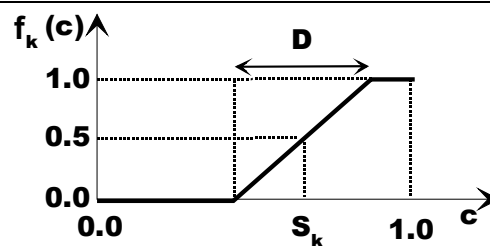


Figure 5.3 : Distribution de possibilité de la classe k

Le degré de possibilité $\pi_k(s)$ que l'objet s appartienne à la classe k est donc obtenu par :

$$\pi_k(s) = f_k(C^k(s)) \quad (\text{Eq. 5.1})$$

La forme de la distribution de possibilité dépend de deux paramètres : D , qui représente la largeur de la zone d'incertitude, sur laquelle le degré de possibilité d'appartenance à la classe est strictement compris entre 0 et 1, et S_k , le milieu de cette zone. Le paramètre D est invariant et égal à 0.4, quelles que soient la classe k et la partition analysée. En revanche, le paramètre S_k est appris à partir des scores de corrélation qui ont été obtenus sur toute la page analysée :

$$S_k = \frac{t_d(k) + D/2 + n(k)m(k)}{n(k) + 1} \quad (\text{Eq. 5.2})$$

Dans cette équation, $n(k)$ représente le nombre d'objets qui ont obtenu leur plus haut score de corrélation avec le modèle M^k de la classe k , ce score étant supérieur ou égal au seuil de décision $t_d(k)$ (paragraphes 4.2.4 et 4.3.3). En d'autres termes, il s'agit du nombre d'objets qui ont été retenus en hypothèse H1, sans hypothèse H0 (Tableau 4.3), donc qui majoritairement appartiennent effectivement à la classe k . La moyenne $m(k)$ de ces scores de corrélation représente donc un degré de similarité moyen entre le modèle M^k et les symboles de la classe k dans la partition. Le paramètre

S_k tend vers $m(k)$ lorsque le nombre $n(k)$ est suffisamment grand, et on affecte un degré de possibilité égal à 0.5 en ce point.

Supposons maintenant que le modèle de classe M^k soit peu adapté à la partition. Dans ce cas, S_k est à peine supérieur à $t_d(k)$. Prenons $S_k = t_d(k)$ pour simplifier. Le degré de possibilité d'appartenance à la classe k est donc nul pour tout objet s ayant obtenu un score de corrélation $C^k(s)$ compris entre 0.0 et $t_d(k)-D/2$, et l'hypothèse est alors considérée comme tout à fait impossible. Il augmente ensuite linéairement, et atteint 1.0 en $t_d(k)+D/2$, score de corrélation à partir duquel l'hypothèse est jugée tout à fait possible.

Lorsque le modèle de classe est au contraire fort ressemblant aux symboles analysés, alors la distribution de possibilité est décalée vers la droite. Il y a ainsi adaptation du modèle à la partition traitée, et donc prise en compte de la variabilité inter-partitions.

La zone d'incertitude, centrée sur S_k , permet de modéliser la variabilité intra-partition. Sa largeur D représente l'écart maximal typique que l'on peut observer entre deux scores de corrélation obtenus par deux objets extraits d'une même partition, et de classes identiques.

Enfin, il faut souligner qu'il n'est pas nécessaire d'estimer précisément la forme de la distribution de possibilité. Il est surtout important qu'elle ne soit pas binaire, et que l'ordre soit préservé. Expérimentalement, nous avons effectivement constaté une bonne robustesse par rapport aux paramètres. Cela peut être expliqué, d'une part par le fait que les informations sont imprécises, si bien que leur représentation peut l'être également, et d'autre part par l'influence modeste que joue chaque élément d'information lorsqu'il est combiné à beaucoup d'autres, comme c'est le cas ici.

La figure 5.4 indique les distributions de possibilité des classes dièse, bémol et bécarre de la partition prise en exemple, et montre comment elles sont appliquées à l'objet 4 de la mesure (b) (Figure 5.1). Les résultats sont conformes à ce que l'on attend : les paramètres S_k définissant ces distributions sont effectivement représentatifs de la forte ressemblance du modèle de la classe bémol aux symboles de la partition, et de la plus faible adéquation des modèles de bécarre et de dièse ; en conséquence, les hypothèses "bémol" et "bécarre" sont correctement éliminées pour l'objet 4, et seule l'hypothèse correcte (dièse) obtient un degré de possibilité non nul.

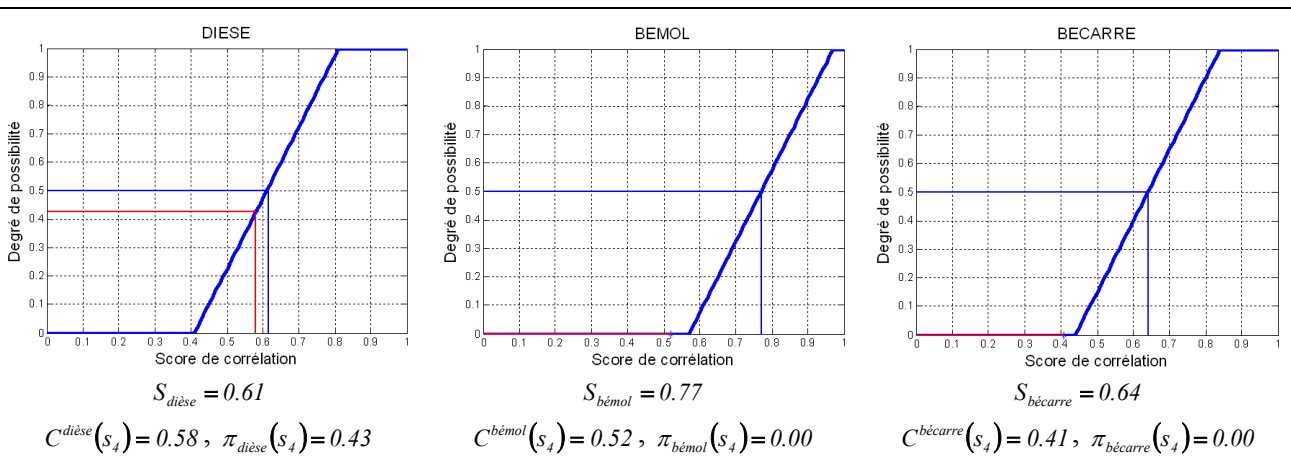


Figure 5.4 : Exemples de distributions de possibilité apprises à partir des scores de corrélation

Le tableau 5.2 indique les degrés de possibilité obtenus pour les symboles de la figure 5.1, étudiés dans le tableau 5.1. En comparant ces deux tableaux, on peut remarquer que les degrés de possibilité d'appartenance aux classes présentent généralement moins d'ambiguïté que les scores de corrélation. Notamment, beaucoup d'hypothèses de reconnaissance obtiennent un degré de possibilité égal à 0. Certaines hypothèses de reconnaissance correctes sont maintenant mieux mises en évidence : les blanches des mesures (a) et (c), les dièses 4 et 11 de la mesure (b). D'autres, en revanche, demeurent ambiguës, par exemple le bémol de la mesure (a), ou le bécarré 9 de la mesure (b), et c'est l'évaluation des règles musicales qui permettra de choisir la bonne hypothèse. On peut aussi remarquer que le rang peut changer : par exemple, l'hypothèse H1 (bémol) de l'objet 6 de la mesure (c) est maintenant a priori éliminée au profit de l'hypothèse correcte H2 (bécarré).

| | Mesure (a) | | | Mesure (b) | | | | Mesure (c) | | | |
|----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | 2 | 3 | 8 | 2 | 4 | 9 | 11 | 2 | 5 | 6 | 8 |
| H0 | | | | | | | | | | (-) | |
| H1 | \flat 0.25 | \natural 0.53 | \natural 0.38 | \natural 0.45 | \sharp 0.43 | \natural 0.38 | \sharp 0.45 | \natural 0.35 | \natural 0.13 | \flat 0.00 | \flat 0.38 |
| H2 | \sharp 0.20 | \bullet 0.00 | \sharp 0.20 | \sharp 0.25 | \flat 0.00 | \flat 0.03 | \flat 0.00 | \bullet 0.00 | \sharp 0.03 | \natural 0.13 | \natural 0.23 |
| H3 | \natural 0.07 | | \flat 0.00 | \flat 0.00 | \natural 0.00 | \sharp 0.33 | \natural 0.00 | | \flat 0.00 | \sharp 0.03 | \natural 0.00 |

Tableau 5.2 : Degrés de possibilité d'appartenance aux classes

Les degrés de possibilité des barres de mesure sont fixés à 1, puisque aucune remise en cause de ces objets n'est envisagée.

5.3. Cohérence graphique

Chaque objet a jusqu'à présent été traité individuellement, sans prise en compte d'informations contextuelles. Nous introduisons maintenant les relations graphiques entre les symboles de la partition. Etant donné une combinaison d'hypothèses de reconnaissance, l'objectif est de calculer des degrés de compatibilité entre chaque objet et tous les autres objets considérés, exprimant dans quelle mesure cet objet satisfait aux règles musicales codifiant les positions relatives des symboles musicaux.

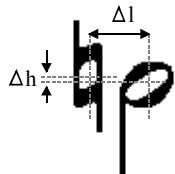
La méthode adoptée procède pas à pas, par évaluation de critères locaux puis fusion. Dans un premier temps, on considère chaque symbole et ses voisins proches dans la mesure. Deux critères sont évalués pour chaque paire (relation binaire), l'un dans la direction verticale, l'autre dans la direction horizontale, puis fusionnés. Les méthodes d'évaluation et de fusion expriment les règles graphiques de la théorie musicale (paragraphe 1.1), et elles dépendent donc des hypothèses de classe considérées. Ces premiers résultats sont de nouveau fusionnés pour exprimer des relations graphiques d'ordre supérieur, c'est-à-dire impliquant plus de deux symboles : on calcule ainsi le degré de compatibilité graphique de chaque objet avec tous ses voisins dans la mesure. Ces résultats seront combinés avec les autres éléments d'information, pour évaluer le degré de possibilité final de la configuration d'hypothèses.

L'étape d'analyse individuelle, fondée sur le template matching, a permis de localiser chaque symbole dans chacune des hypothèses de reconnaissance (Equation 4.2). La cohérence graphique de chaque configuration d'hypothèses peut donc être appréciée à partir des coordonnées obtenues. Il n'est cependant pas souhaitable d'évaluer les règles graphiques de manière stricte, étant donné les sources d'imprécisions : imprécision des coordonnées elles-mêmes, due à la variabilité des symboles (typographies différentes, défauts d'impression et de segmentation), mais surtout, imprécision des règles musicales. Prenons l'exemple des altérations : la règle spécifie qu'une altération accidentelle doit être placée devant la note, et à la même hauteur. La distance qui sépare ces deux objets n'est pas codifiée de manière stricte, et dépend au contraire beaucoup de la densité des symboles, ou simplement de la mise en page, comme on peut le remarquer en comparant les bécarres des mesures (f) et (g) de la figure 5.2. Bien que la position dans la direction verticale soit exprimée de manière précise, il y a en pratique une certaine tolérance dans l'application de la règle, puisque des décalages plus ou moins importants par rapport à la position théorique peuvent être constatés (bécarre de la mesure (c), Figure 5.1). C'est pourquoi toutes les règles graphiques ont été exprimées sous la forme de relations floues, les coefficients de compatibilité obtenus prenant des valeurs, non pas binaires, mais comprises entre 0 et 1, reflétant le degré de satisfaction de la règle.

Nous allons dans la suite de ce paragraphe expliciter la modélisation des différentes règles graphiques décrites au paragraphe 1.1, puis la méthode de fusion qui aboutit à un unique coefficient par symbole, exprimant sa compatibilité avec les autres objets de la mesure.

5.3.1. Compatibilité graphique entre une altération accidentelle et une note

Une altération accidentelle doit être placée devant une note, à la même hauteur. Notons $(x_n^{k_n}, y_n^{k_n})$ et $(x_m^{k_m}, y_m^{k_m})$ les coordonnées des objets s_n et s_m ($m > n$) dans les hypothèses de classe k_n et k_m (position du maximum de corrélation, Eq. 4.2), Δl et Δh les décalages entre les deux symboles :



$$\begin{cases} \Delta l = y_m^{k_m} - y_n^{k_n} \\ \Delta h = |x_m^{k_m} - x_n^{k_n}| \end{cases} \quad (\text{Eq. 5.3})$$

Le degré de compatibilité exprimant la possibilité qu'un objet s_n soit une altération de classe k_n et que l'objet s_m soit une note de classe k_m est calculé par :

$$C_p(s_n^{k_n}, s_m^{k_m}) = \begin{cases} \alpha_l f_l(\Delta l) + \alpha_h f_h(\Delta h) & \text{si } f_l(\Delta l) > 0 \text{ et } f_h(\Delta h) > 0 \\ 0 & \text{sinon} \end{cases} \quad (\text{Eq. 5.4})$$

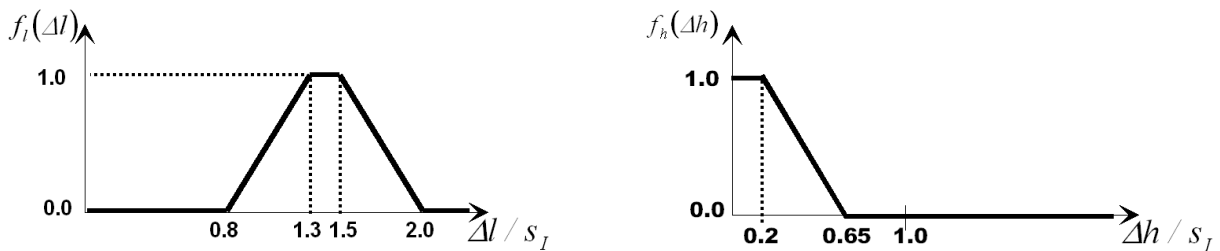


Figure 5.5 : Compatibilité graphique entre une altération accidentelle et une note

Les fonctions f_i et f_h (Figure 5.5) définissent les décalages admissibles dans les deux directions. La combinaison utilisée dans l'équation 5.4 exclut les cas où l'un des deux critères n'est pas du tout satisfait. Cette condition permet d'écartier définitivement des symboles incohérents, comme les objets 6(d) et 8(g), ou encore les bémols sur les hampes, qui ne sont suivis d'aucune hypothèse de note. Dans tous les autres cas, il s'agit d'un compromis : les positions étant compatibles dans les deux directions, le coefficient de compatibilité graphique est calculé par une somme pondérée, les poids exprimant l'importance relative des deux critères. Nous avons choisi $\alpha_i = 0.2$ et $\alpha_h = 0.8$, car le décalage dans la direction horizontale n'est pas aussi significatif que celui dans la direction verticale.

La figure 5.6 illustre le principe sur l'objet 2 de la mesure (a). Le degré de possibilité que l'objet 2 soit un bémol suivi d'une blanche (1.00) est supérieur au degré de possibilité de la combinaison dièse/blanche (0.76), ce qui correspond bien aux résultats attendus. L'introduction de cette règle musicale contribue donc à renforcer l'hypothèse correcte (H1, bémol), et à la différencier de l'hypothèse H2 incorrecte (dièse), ce qui est très intéressant étant donné l'ambiguïté des degrés de possibilité d'appartenance aux classes (respectivement 0.25 et 0.20 dans le tableau 5.2).

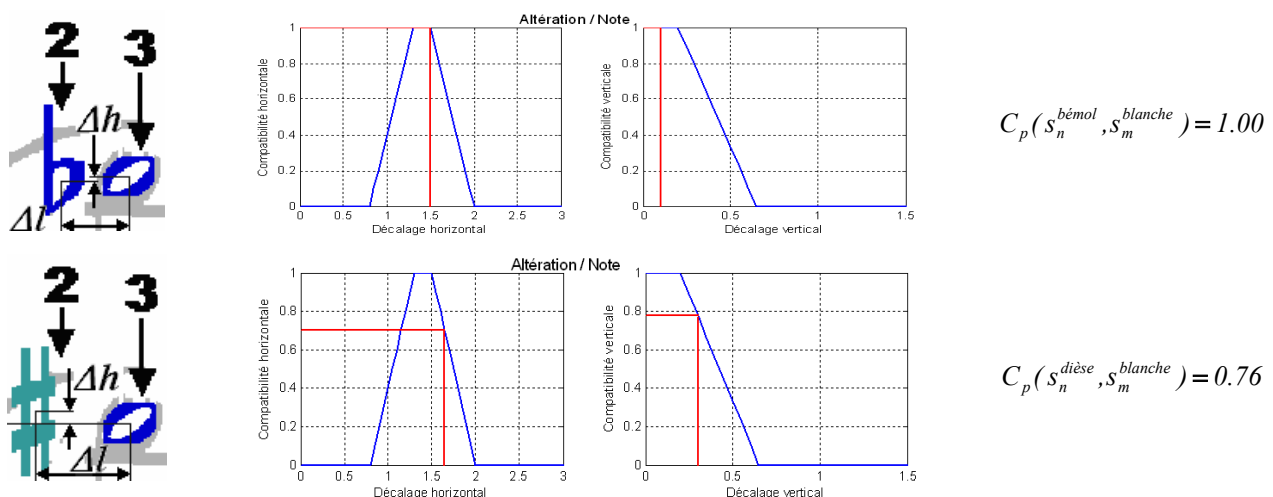


Figure 5.6 : Compatibilité graphique : comparaison des résultats pour deux hypothèses concurrentes

Le tableau 5.3 indique quelques coefficients de compatibilité :

| $s_n \downarrow s_m \rightarrow$ | Noire | Noire | Noire | Noire | Noire | Noire |
|----------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Bécarre | 0.00 | 1.00 | 0.72 | 0.96 | 1.00 | 0.86 |
| Dièse | 1.00 | 1.00 | | 0.93 | 1.00 | 0.89 |
| Bémol | 0.00 | 0.00 | 0.91 | 0.00 | 0.46 | |

Tableau 5.3 : Exemples de coefficients de compatibilité graphique entre une altération et une note. En gras les coefficients de compatibilité qui correspondent à l'hypothèse exacte.

Ces résultats amènent quelques commentaires :

- Les coefficients de compatibilité sont maximaux lorsque l'altération est à la position usuelle (dièse 4(b), bécarres 9(b) et 11(f)) et diminuent lorsqu'elle s'en éloigne (bécarres 4(f) et 11(g)). L'utilisation de valeurs comprises entre 0 et 1, plutôt que des seuils binaires, permet de ne pas écarter des hypothèses exactes, mais qui ne sont pas exactement à la position attendue.
- Le coefficient de compatibilité le plus élevé correspond généralement à l'hypothèse exacte (dans tous les cas sur les exemples proposés, sauf pour le bécarre 11(g)), ce qui contribue à la renforcer.
- Un certain nombre d'hypothèses fausses sont éliminées, car les coefficients sont nuls : par exemple, les hypothèses "bécarre" et "bémol" pour l'objet 4 de la mesure (b).
- Toutes les ambiguïtés ne sont cependant pas résolues. Pour l'objet 4(b), seule la classe dièse obtient un degré de compatibilité non nul et l'ambiguïté est donc parfaitement levée ; au contraire, les hypothèses "bécarre" et "dièse" pour l'objet 9(b), qui conduisent à des degrés de possibilité d'appartenance aux classes presque identiques (0.38 et 0.33 dans le tableau 5.2), ne peuvent toujours pas être départagées.

Les mesures (f) et (g) montrent d'autres configurations qui ne peuvent être complètement résolues par les degrés de possibilité d'appartenance aux classes et l'évaluation de règles graphiques, et qui nécessiteront l'introduction des règles syntaxiques portant sur les altérations et la tonalité.

5.3.2. Compatibilité graphique entre une appoggiature et une note

Une méthode similaire est appliquée pour les appoggiatures. La fonction f_h a cependant été modifiée, puisqu'une appoggiature est le plus souvent décalée d'un demi-interligne, parfois davantage (Figure 5.7). Le coefficient de compatibilité exprimant le degré de possibilité qu'un objet s_n soit une appoggiature et qu'un objet voisin s_m soit une note de classe k_m est toujours défini par l'équation 5.4, avec cette fois $\alpha_l=0.5$ et $\alpha_h=0.5$: les deux critères ont maintenant la même importance.

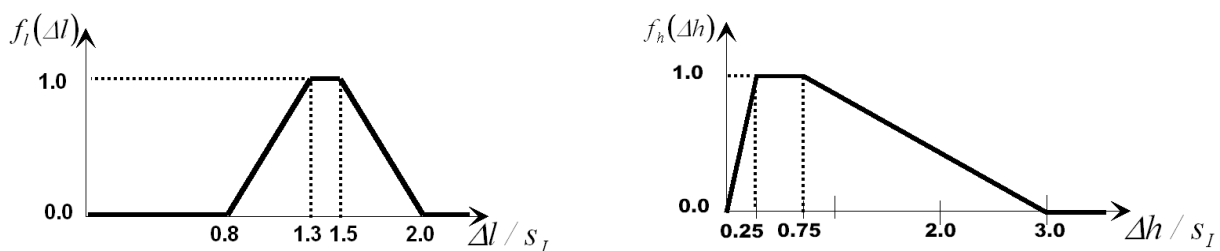


Figure 5.7 : Compatibilité graphique entre une appoggiature et une note

Le tableau 5.4 donne quelques exemples. La comparaison avec les résultats obtenus sur les altérations montre que ces fausses appoggiatures ont un coefficient de compatibilité inférieur à ceux des altérations exactes (8(c) 5(d) et 10(g)), ce qui de nouveau renforce les bonnes hypothèses, ou sont bien éliminées lorsque leur position est incompatible avec la tête de note suivante (8(g)). Les critères graphiques apportent cependant peu d'information dans le cas du symbole 10 de la mesure (g), et la décision portera davantage sur les règles syntaxiques et les degrés de possibilité

d'appartenance aux classes.

| | 8/9(c) | 5/7(d) | 8/9(g) | 10/12(g) |
|----------------------------------|--|--|--|--|
| $s_n \downarrow s_m \rightarrow$ | <i>Noire</i> | <i>Noire</i> | <i>Noire</i> | <i>Noire</i> |
| | $\begin{cases} f_i(\Delta l) = 0.40 \\ f_i(\Delta h) = 1.00 \end{cases}$ | $\begin{cases} f_i(\Delta l) = 0.36 \\ f_i(\Delta h) = 1.00 \end{cases}$ | $\begin{cases} f_i(\Delta l) = 0.82 \\ f_i(\Delta h) = 0.00 \end{cases}$ | $\begin{cases} f_i(\Delta l) = 0.82 \\ f_i(\Delta h) = 0.88 \end{cases}$ |
| <i>Appoggiature</i> | 0.70 | 0.68 | 0.00 | 0.85 |
| <i>Bécarre</i> | 0.72 | | 0.00 | 0.86 |
| <i>Dièse</i> | | | | 0.89 |
| <i>Bémol</i> | 0.91 | 1.00 | 0.00 | |

Tableau 5.4 : Exemples de coefficients de compatibilité graphique entre une appoggiature et une note, et comparaison avec les coefficients obtenus par les hypothèses d'altérations. En gras les coefficients qui correspondent à l'hypothèse exacte

5.3.3. Compatibilité graphique entre une note et un point de durée

Les points allongeant la durée des notes sont recherchés, durant l'étape d'analyse individuelle des symboles, sur une petite zone proche de la tête de note. La localisation typique est indiquée en gris clair sur la figure 5.8a. On peut néanmoins trouver des points beaucoup plus proches ou plus éloignés, car de nouveau, la distance séparant les deux symboles n'est pas fixée de manière précise par la théorie musicale. En pratique, on constate qu'elle peut varier de manière très significative, en fonction de l'édition ou de la densité locale de la partition, et c'est pourquoi la zone de recherche des points de durée est assez étendue (rectangle de la figure 5.8a). Un coefficient de compatibilité graphique a cependant été défini, de manière à privilégier les configurations courantes, sans éliminer celles qui sont plus rares (Figure 5.8b, Equation 5.5).

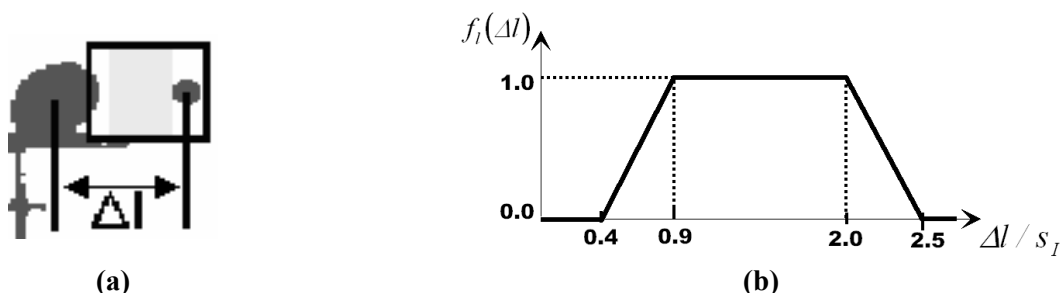


Figure 5.8 : (a) Zone de recherche d'un point de durée, (b) Compatibilité graphique entre une note et un point de durée.

$$C_p(s_n^{k_n}, s_m^{k_m}) = f_l(\Delta l) \quad (\text{Eq. 5.5})$$

Le tableau 5.5 montre trois exemples. Le point de durée 4(a) a un coefficient de

compatibilité maximal avec la note qui le précède. Mais c'est également le cas des points de staccato 6(f) et 6(g), qui ont une position très ambiguë.


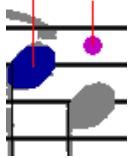
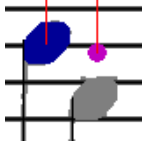
| | | | |
|----------------------------------|---|--|---|
| |  |  |  |
| | 3/4(a) | 5/6(f) | 5/6(g) |
| $s_m \downarrow s_n \rightarrow$ | <i>Blanche</i> | <i>Noire</i> | <i>Noire</i> |
| <i>Point de durée</i> | 1.00 | 1.00 | 0.96 |

Tableau 5.5 : Exemples de coefficients de compatibilité graphique entre une note et un point de durée

Il est donc nécessaire de modéliser également la règle graphique portant sur les points de staccato, afin de trouver la vraie signification des points extraits lors de l'analyse individuelle des symboles : point qui allonge la note précédente, ou point de phrasé qui agit sur la note suivante (note piquée).

5.3.4. Compatibilité graphique entre un point et une note de son voisinage

Le degré de possibilité qu'un objet s_n soit un point de durée et que l'objet s_m soit une note est fonction du décalage horizontal et vertical entre ces deux symboles, avec cette nouvelle définition :

$$\begin{cases} \Delta l = |y_m^{k_m} - y_n^{k_n}| \\ \Delta h = |x_m^{k_m} - x_n^{k_n}| \end{cases} \quad (\text{Eq. 5.6})$$

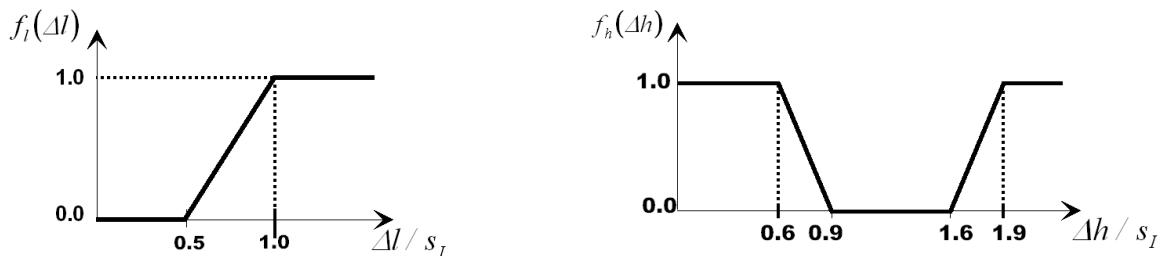


Figure 5.9 : Compatibilité graphique entre un point de durée et une note de son voisinage

$$C_p(s_n^{k_n}, s_m^{k_m}) = \text{Max}[f_l(\Delta l), f_h(\Delta h)] \quad (\text{Eq. 5.7})$$

Le degré de compatibilité est donc élevé, dès lors que l'un des deux critères est vérifié. La figure 5.10 illustre la nécessité d'une combinaison plus indulgente que les précédentes.

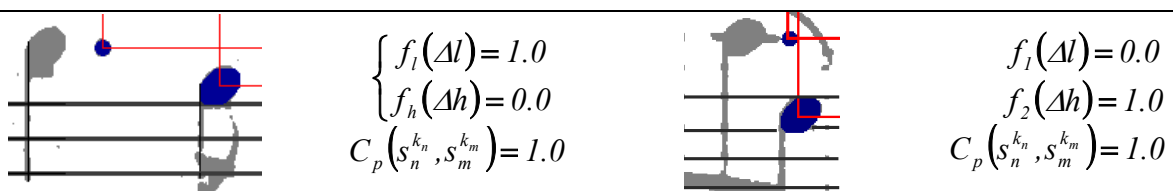


Figure 5.10 : Compatibilité graphique entre un point et une note de son voisinage

La figure 5.11 montre des résultats sur les mesures des figures 5.1 et 5.2. Les points de durée 4(a) et 4(e) sont effectivement déclarés totalement compatibles avec les notes voisines. En revanche, le point de staccato 6(g) est totalement éliminé. Le point 6(f) reste ambigu : étant un peu trop haut par rapport à la tête de note, il obtient un coefficient de compatibilité faible mais non nul, reflétant qu'il est toujours possible de l'interpréter comme un point de durée. L'introduction des règles de métrique contribuera à la résolution de ce cas.

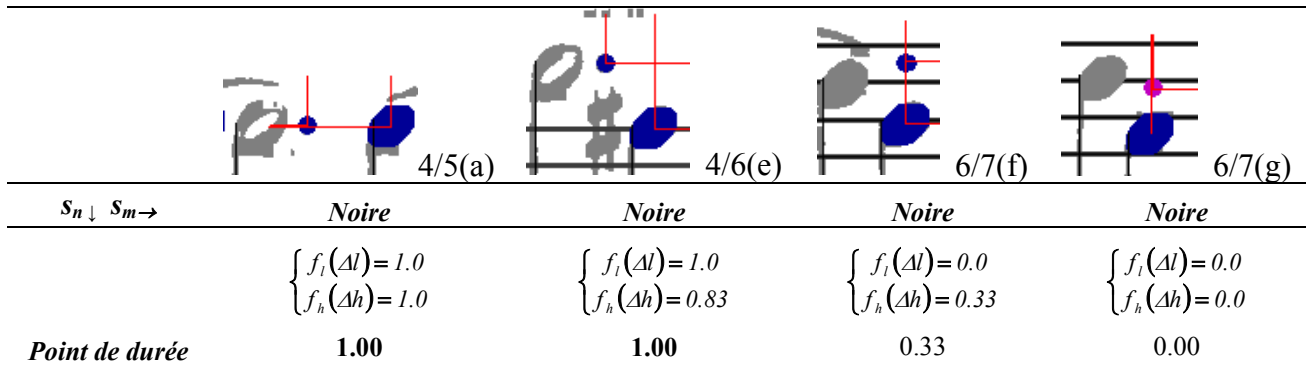
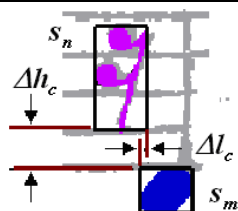


Figure 5.11 : Exemples de compatibilité graphique entre un point et une note de son voisinage

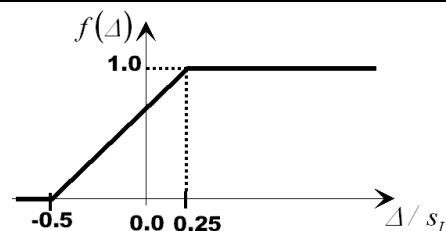
5.3.5. Compatibilité graphique entre deux symboles quelconques

Toutes les règles graphiques spécifiques à des classes ont été définies. Pour toute autre paire d'objets, il suffit simplement d'exprimer que deux symboles voisins ne doivent théoriquement pas se chevaucher. On estime les positions des boîtes englobantes de deux objets successifs s_n et s_m ($m > n$), à partir de leurs coordonnées $(x_n^{k_n}, y_n^{k_n})$ et $(x_m^{k_m}, y_m^{k_m})$, et des dimensions typiques des symboles des classes k_n et k_m considérées (Figure 5.12a). Soit Δl_c le décalage entre le bord droit de l'objet s_n et le bord gauche de l'objet s_m . Les valeurs admissibles sont données par la fonction f (Figure 5.12b). Celle-ci permet d'autoriser des décalages négatifs, afin de prendre en compte l'imprécision de l'estimation, et d'autoriser les chevauchements que l'on peut trouver dans les partitions de forte densité. Dans la direction verticale, on doit considérer plusieurs cas suivant la position relative des rectangles englobants. Par exemple, si s_n est au-dessus de s_m , alors le décalage vertical Δh_c est la différence entre le bord supérieur de s_m et le bord inférieur de s_n . Les valeurs admissibles pour Δh_c sont de nouveau définies par la fonction f . Le coefficient de compatibilité final est le maximum des deux critères, car un seul suffit pour que les deux objets soient bien séparés :

$$C_p(s_n^{k_n}, s_m^{k_m}) = \text{Max}[f(\Delta l_c), f(\Delta h_c)] \quad (\text{Eq. 5.8})$$



(a) Décalages entre deux symboles quelconques



(b) Compatibilité graphique dans les deux directions

Figure 5.12 : Compatibilité graphique entre deux symboles quelconques

La figure 5.13 illustre le procédé sur quelques exemples. Les deux premiers montrent qu'un chevauchement peu important, dû à une forte densité de symboles, est pénalisé mais reste possible (objets 7 et 8 de la mesure (a)), contrairement aux symboles qui se superposent vraiment, comme les objets 3 et 4 de la mesure (d). Tous les degrés sont donc permis, de la valeur maximale 1 qui correspond à une bonne séparation, à des valeurs nulles qui correspondent sans ambiguïté à des cas interdits dans l'édition des partitions monodiques. Le coefficient de compatibilité est maximal dès lors que la séparation est bonne dans une direction au moins, ce qui permet de ne pas rejeter une configuration insolite (mesure (e), avec un point de durée juste au-dessus d'une altération).

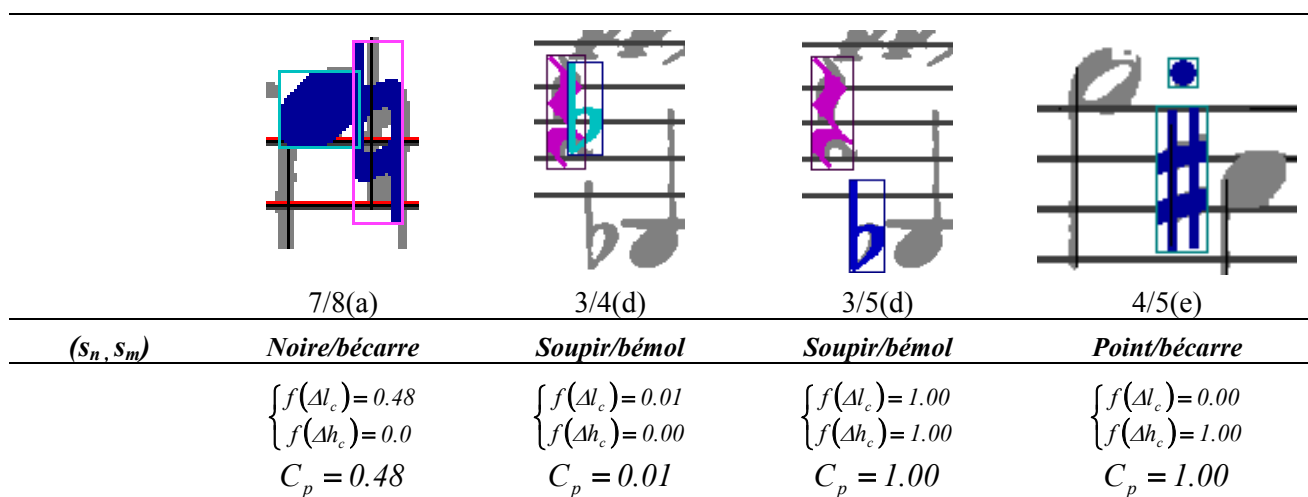


Figure 5.13 : Coefficients de compatibilité graphique entre deux objets de classes quelconques

5.3.6. Modification des hypothèses de reconnaissance

Les différents coefficients de compatibilité graphique fournissent des informations indiquant qu'il peut être judicieux, dans certains cas, d'ajouter une hypothèse H0 (absence de symbole), si celle-ci n'est pas présente. Les exemples présentés montrent en effet que des fausses détections, ou encore des superpositions dues à la sur-segmentation d'un objet, se traduisent généralement par des coefficients de compatibilité nuls ou faibles.

Reprenons les différentes règles. Tout d'abord celles qui portent sur la compatibilité d'une altération ou d'une appoggiature avec la note suivante. Lorsqu'un objet $s_n^{k_n}$ est classé en dièse, bécarre, bémol ou appoggiature en hypothèse H1, et qu'aucune hypothèse de note $s_m^{k_m}$ n'aboutit à un coefficient de compatibilité non nul ($C_p(s_n^{k_n}, s_m^{k_m}) = 0$ selon l'équation 5.4), alors l'hypothèse H0 est ajoutée, si elle n'était pas présente : cette situation suggère en effet que l'objet s_n est une fausse détection, et il faut donc introduire la possibilité qu'il n'y ait pas de symbole à cet endroit.

Considérons ensuite la compatibilité d'un point. Si le coefficient de compatibilité graphique entre ce point et l'une des notes est strictement inférieur à 1 (Eq. 5.7), alors il y a un doute sur la nature de ce point, qui est peut-être un point de staccato. Dans ce cas, l'hypothèse H0 est ajoutée si elle n'était pas présente, donnant la possibilité d'ignorer l'objet en tant que point de durée. Cette modification des hypothèses est réalisée sur l'objet 6(f) (Figure 5.11), puisqu'il obtient un coefficient de compatibilité avec la noire 7(f) égal à 0.33. La décision finale combinera cette

information aux critères syntaxiques portant sur les groupes de notes et la métrique pour lever l'ambiguïté.

La dernière règle, concernant la compatibilité graphique de deux symboles quelconques, révèle plutôt des défauts de segmentation. Rappelons en effet que celle-ci n'est pas parfaite, et qu'il arrive qu'un même objet ait été détecté deux fois, essentiellement à cause des connexions parasites entre symboles voisins (Figure 3.29). Les cas de double détection, qui conduisent à des hypothèses de reconnaissance identiques, ont été résolus (paragraphe 4.5). Les autres ont, au contraire, été laissés en suspens. Il arrive également que la même hypothèse de reconnaissance soit générée pour deux objets consécutifs très proches (Figure 4.4b), si les zones de corrélation se chevauchent. Ces erreurs peuvent maintenant être détectées, puisque deux hypothèses superposées aboutissent nécessairement à un coefficient de compatibilité graphique nul. Lorsque ces deux hypothèses sont toutes les deux de niveau H1, elles s'excluent mutuellement. Il faut donc ajouter, pour les deux objets, l'hypothèse H0 (absence de symbole), si elle n'était pas présente. Ainsi, l'algorithme de décision aura la totale possibilité de choisir l'une ou l'autre hypothèse, ou bien aucune.

Dans la mesure (c) de la figure 5.1, le bécarre est détecté deux fois. Le coefficient de compatibilité graphique entre les objets 5 (H1) et 6 (H2), est nul. L'hypothèse H0 est donc ajoutée pour l'objet 5. Ce n'était pas nécessaire dans cet exemple, puisque l'hypothèse H0 était déjà présente pour l'objet 6, permettant de retenir l'hypothèse H1 pour l'objet 5, et l'hypothèse H0 pour l'objet 6. C'est en revanche indispensable dans d'autres cas, pour que l'algorithme de décision puisse aboutir à la configuration exacte.

5.3.7. Fusion : compatibilité graphique d'un symbole avec tous ses voisins

La compatibilité graphique de chaque objet avec tous ses voisins a donc été évaluée, paire par paire, en fonction des hypothèses de classe. L'étape suivante consiste à fusionner ces résultats, de sorte que chaque objet obtienne un unique coefficient de compatibilité graphique avec tous ses voisins dans la mesure, dans chaque configuration d'hypothèses. Celui-ci est défini par :

$$C_p(s_n^{k_n}) = \left[\min_{j < n} C_p(s_j^{k_j}, s_n^{k_n}) \right] \cdot \left[\min_{l > n} C_p(s_n^{k_n}, s_l^{k_l}) \right] \quad (\text{Eq. 5.9})$$

Il s'agit du produit de deux termes, le premier représentant la compatibilité graphique de l'objet s_n avec tous ses voisins antérieurs dans la mesure, le second sa compatibilité graphique avec ses voisins postérieurs. L'utilisation de l'opérateur de conjonction *min* dans chaque terme exprime que le symbole s_n doit être simultanément compatible avec tous les autres. Contrairement à d'autres t-normes, comme le produit, il permet d'obtenir des résultats comparables quel que soit le nombre d'objets impliqués. Pour combiner les deux termes, on emploie cette fois le produit, car cet opérateur a un comportement plus sévère et permet de mieux différencier les configurations d'hypothèses.

Cette fusion conclut l'évaluation des règles graphiques. Ces résultats seront ensuite combinés avec les degrés de possibilité d'appartenance aux classes, et les résultats portant sur les règles syntaxiques (paragraphe 5.4), afin de prendre une décision (paragraphe 5.5). On peut d'ores

et déjà remarquer que la méthode permet de comparer la compatibilité graphique de toutes les configurations d'hypothèses, de manière globale sur toute la mesure, et qu'elle va donc au-delà de l'évaluation de règles binaires et locales. Les exemples présentés ont également montré que la modélisation floue des règles graphiques donne des résultats significatifs, dans le sens où les combinaisons d'hypothèses exactes obtiennent les plus hauts coefficients de compatibilité, et que celles qui ne satisfont pas rigoureusement à la théorie musicale sont certes pénalisées, mais non rejetées. Les différentes fonctions proposées ont été définies expérimentalement, à partir de l'observation des partitions musicales, et validées par les simulations réalisées sur toute la base d'images. Il n'y a pas eu d'optimisation globale des paramètres qui les définissent, mais on a pu vérifier qu'une petite variation des paramètres conduit à des décisions similaires, prouvant la robustesse de la méthode. Comme nous l'avons déjà souligné au paragraphe 5.2, l'un des intérêts du formalisme flou est qu'il ne nécessite pas l'ajustement précis des distributions de possibilité, car finalement, le plus important est la relation d'ordre établie plutôt que les valeurs en elles-mêmes.

5.4. Cohérence syntaxique

Nous introduisons dans cette partie les règles musicales relatives à la tonalité, les altérations, et la métrique. Ces règles correspondent à des informations globales de la partition. De ce fait, elles impliquent généralement un grand nombre de symboles, graphiquement distants dans l'image. D'autre part, nous avons souligné leur flexibilité dans le paragraphe 1.1. La méthode proposée permet de surmonter ces deux difficultés, et constitue l'une des innovations majeures par rapport à la bibliographie, les systèmes présentés jusqu'à présent n'intégrant pas encore ce type d'information dans le cœur même du processus de reconnaissance. Elle est de nouveau fondée sur la théorie des ensembles flous et des possibilités, qui est très bien adaptée à la modélisation de contraintes souples. Les différentes règles sont testées, sur chaque configuration d'hypothèses de reconnaissance, et un degré de possibilité est affecté à chaque objet concerné par la règle, exprimant sa compatibilité avec les autres objets intervenant dans l'évaluation de la règle.

5.4.1. Armure

La tonalité est donnée en paramètre d'entrée du programme. La règle 6 (paragraphe 1.1) indique qu'une succession d'altérations, dièses ou bémols, suivant un ordre prédéfini, doit être placée juste après la clé. Il s'agit donc d'une contrainte stricte, et un coefficient de compatibilité binaire, noté $C_s(s_n^{k_n})$, est donc affecté aux altérations à la clé : 1 si l'altération satisfait à la règle, 0 dans le cas contraire.

5.4.2. Altérations accidentelles

Les règles introduites dans ce paragraphe concernent toutes les autres hypothèses d'altérations, qui sont donc a priori placées devant une tête de note. Si un symbole appartient à l'une des classes d'altérations (bémol, dièse, ou bécarre), alors il doit être cohérent d'une part avec la tonalité, d'autre part avec les autres altérations de la partition. D'après les règles 6 et 7 (paragraphe

1.1), il suffit en fait de considérer les altérations de même hauteur, à l'octave près¹ : l'altération éventuellement présente dans l'armure, les altérations précédentes dans la mesure, et, le cas échéant, dans les mesures antérieures, mais proches. Un degré de possibilité $C_s(s_n^{k_n})$ est donc affecté à chaque altération, en fonction de la configuration.

Considérons une combinaison d'hypothèses dans laquelle un symbole s_n est une altération de classe k_n (bémol, bécarre ou dièse), ce symbole étant précédé dans la mesure d'une autre altération s_m ($m < n$), de classe k_m , et de même hauteur. Supposons également dans un premier temps qu'il n'y a pas d'altération à la clé. Le tableau 5.6 indique le degré de possibilité $C_s(s_n^{k_n})$ attribué à l'objet s_n :

| | s_n =dièse | s_n =bécarre | s_n =bémol |
|-----------------|--------------|----------------|--------------|
| s_m = aucune | 0.75 | 0.5 | 0.75 |
| s_m = dièse | 0.5 | 1.0 | 0.0 |
| s_m = bécarre | 1.0 | 0.5 | 1.0 |
| s_m = bémol | 0.0 | 1.0 | 0.5 |

Tableau 5.6 : Coefficients de compatibilité syntaxique entre deux altérations de même hauteur présentes dans la même mesure, sans altération à la clé.

Les configurations les plus usuelles sont les suivantes : lorsqu'un dièse ou un bémol apparaît pour la première fois dans la mesure, ou lorsqu'un bécarre annule un dièse ou un bémol. Les coefficients de compatibilité attribués sont respectivement de 0.75 dans le premier cas, et de 1.0 dans le second : ils sont supérieurs à 0.5, puisque ces configurations sont toutes deux parfaitement valides, mais un poids plus grand est attribué à la seconde, afin de favoriser toute interaction cohérente dans la mesure. Il est également possible que l'altération s_m rappelle la première, de manière à faciliter la lecture. Cette configuration est possible, mais la présence de s_m n'est pas obligatoire, et on lui attribue donc un degré de possibilité moyen (0.5). Enfin, certaines configurations sont a priori impossibles (degré nul), comme la présence d'un bémol après un dièse.

Le tableau 5.7 indique les degrés de possibilité définis, suivant un raisonnement similaire, lorsqu'un dièse est présent dans l'armure, à la même hauteur que s_m et s_n . Le degré de possibilité est nul lorsqu'il correspond à une configuration impossible (par exemple, un dièse à la clé et un bémol dans la mesure), est égal à 0.5 pour une association possible mais non obligatoire (comme le rappel d'une altération déjà dans l'armure), est maximal pour une interaction cohérente (par exemple, un bécarre annulant le dièse à la clé). La configuration de la dernière ligne ne se produit a priori jamais, puisque l'objet s_m ne peut être un bémol, sachant qu'il y a un dièse à la clé. L'objet s_m est donc mal classé, et, en l'absence d'information fiable, on reprend les degrés de possibilité de la première ligne.

| | s_n =dièse | s_n =bécarre | s_n =bémol |
|-----------------|--------------|----------------|--------------|
| s_m = aucune | 0.5 | 1.0 | 0.0 |
| s_m = dièse | 0.5 | 1.0 | 0.0 |
| s_m = bécarre | 1.0 | 0.5 | 0.0 |
| s_m = bémol | 0.5 | 1.0 | 0.0 |

Tableau 5.7 : Coefficients de compatibilité syntaxique entre deux altérations de même hauteur présentes dans la même mesure, avec un dièse à la clé

¹ Dans la suite, "de même hauteur" signifiera toujours "à l'octave près", mais nous omettrons de le préciser.

Il suffit d'interchanger bémol et dièse dans le tableau 5.7 pour traiter le cas où un bémol est à la clé.

On considère enfin les altérations dans les mesures précédentes. Cette configuration n'est examinée que s'il n'y a pas d'altération de même hauteur que s_n la précédant dans la mesure ou dans l'armure. Le tableau 5.8 remplace donc la première ligne du tableau 5.6, lorsqu'une altération s_m de même hauteur est néanmoins présente dans une mesure précédente. De nouveau, les configurations impossibles sont affectées d'un degré de possibilité nul (bémol/dièse et dièse/bémol), égal à 0.5 dans tous les autres cas : par exemple, la présence d'un bécarré annulant un dièse dans une mesure précédente n'est pas obligatoire, et elle devient tout aussi possible qu'une configuration dièse/dièse.

| | S_n =bémol | s_n =bécarré | s_n =dièse |
|-----------------|--------------|----------------|--------------|
| s_m = bémol | 0.5 | 0.5 | 0.0 |
| s_m = bécarré | 0.5 | 0.5 | 0.5 |
| s_m = dièse | 0.0 | 0.5 | 0.5 |

Tableau 5.8 : Coefficients de compatibilité syntaxique entre deux altérations de même hauteur, présentes dans des mesures différentes, sans altération à la clé.

Lorsqu'une altération est présente à la clé, les altérations dans les mesures précédentes ne sont jamais prises en compte, car l'information donnée par l'armure est prédominante. La première ligne du tableau 5.7 est donc toujours appliquée, si aucune altération s_m ne précède l'altération s_n dans la mesure.

Prenons maintenant l'exemple des objets 4 et 9 de la mesure (b) (Figure 5.1), pour lesquels 3 hypothèses de classe (bémol, dièse et bécarré) sont générées par objet. Au total, neuf combinaisons doivent être évaluées, sachant qu'il n'y a pas d'altération de même hauteur à la clé :

| | | | | | | | | | | | |
|---|---|--------------|--------------|---|---|--------------|--------------|---|---|--------------|--------------|
| 4 | 9 | $C_s(s_4^k)$ | $C_s(s_9^k)$ | 4 | 9 | $C_s(s_4^k)$ | $C_s(s_9^k)$ | 4 | 9 | $C_s(s_4^k)$ | $C_s(s_9^k)$ |
| # | # | 0.75 | 0.5 | b | # | 0.75 | 0.0 | ♯ | # | 0.5 | 1.0 |
| # | b | 0.75 | 0.0 | b | b | 0.75 | 0.5 | ♯ | b | 0.5 | 1.0 |
| # | ♯ | 0.75 | 1.0 | b | ♯ | 0.75 | 1.0 | ♯ | ♯ | 0.5 | 0.5 |

Tableau 5.9 : Exemple de degrés de possibilité obtenus sur la mesure (b) pour les objets 4 et 9.
Les cases grisées correspondent aux hypothèses d'altération qui ont une compatibilité graphique nulle avec la note suivante

On voit sur cet exemple comment les deux objets interagissent. En moyenne, les deux meilleures configurations sont un dièse pour l'objet 4, suivi d'un bécarré pour l'objet 9, ou un bémol pour l'objet 4, suivi d'un bécarré pour l'objet 9. Si on introduit les règles graphiques (tableau 5.3), la seconde possibilité est éliminée car l'hypothèse bémol a un coefficient de compatibilité graphique nul avec la note altérée. On pressent donc que les critères graphiques et syntaxiques fusionnés vont conduire à la solution correcte, c'est-à-dire dièse pour l'objet 4, bécarré pour l'objet 9.

5.4.3. Métrique

La métrique est généralement introduite tout à la fin du processus de reconnaissance, pour

détecter, voire corriger des erreurs (e.g. [Couïasnon, Rétif 95] [Droettboom et al. 02] [Ferrand et al. 99]). Il s'agit généralement de compter le nombre de temps par mesure, qui doit nécessairement correspondre à la signature temporelle (règle 4 dans le paragraphe 1.1), puis d'ajouter des critères, comme l'alignement vertical en musique polyphonique, afin d'effectuer des corrections.

Nous proposons au contraire d'intégrer les règles 4 et 5 (paragraphe 1.1) relatives à la métrique dans l'algorithme de reconnaissance. La règle 4 ne sera évaluée que lors de la décision. En revanche, la règle 5, relative aux regroupements de notes, est modélisée pour améliorer l'interprétation des durées de note.

La méthode est fondée sur la détection des barres de groupe, présentée au paragraphe 4.2.6, qui a permis de valider la présence d'au moins une barre de groupe reliant deux objets successifs, dans l'hypothèse où il s'agit de noires. Ces premiers résultats peuvent être utilisés pour former les groupes complets. Dans la mesure (c) par exemple, les symboles 3, 4, 7 et 9 sont tous classés "noire" en hypothèse H1, et connectés par paires, d'après les équations 4.5, 4.6 et 4.7. On peut donc en déduire qu'ils forment un groupe de 4 notes. Les paramètres de la barre de groupe la plus extrême sont affinés, par la recherche du segment de pente a , qui relie les extrémités de la première et de la dernière note, et qui maximise le rapport donné en équation 4.7. La plage de variation de a est déduite des pentes des segments reliant les notes deux à deux. Le paramètre b est toujours défini par l'équation 4.6. Ainsi, la barre de groupe externe est précisément localisée.

Ces résultats sont utilisés pour recalculer la durée de chacune des noires s_n du groupe. De nouveau, il s'agit simplement de compter le nombre de barres de groupe de part et d'autre de la hampe, comme exposé au paragraphe 4.2.6, mais sur une section $[x_{11}, x_{12}]$ déduite, non plus du cadre englobant, mais de la position exacte de la barre de groupe (Figure 5.14) :

$$y = y_p(s_n) \pm 0.25s_I$$

$$\text{Si } |x_{pb}(s_n) - x_n^{k_n}| < |x_n^{k_n} - x_{ph}(s_n)| : \begin{cases} x_{12} = ay + b - 0.3s_I \\ x_{11} = \text{Min}(x_n^{k_n} - s_I, x_{12} + 3s_I) \end{cases}$$

(tête de note en bas)

$$\text{Sinon (tête de note en haut)} : \begin{cases} x_{12} = ay + b + 0.3s_I \\ x_{11} = \text{Max}(x_n^{k_n} + s_I, x_{12} - 3s_I) \end{cases} \quad (\text{Eq. 5.10})$$

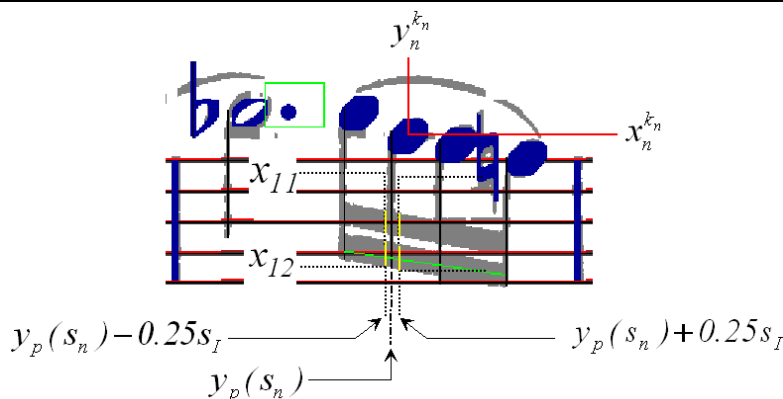


Figure 5.14 : Détermination de la durée d'une noire incluse dans un groupe de notes en fonction de la position de la barre de groupe externe.

Cette opération est effectuée sur chaque nouvelle hypothèse de groupement de notes et mémorisée, si bien qu'elle n'a pas à être réitérée.

Les points de durée présents dans la configuration d'hypothèses sont ensuite affectés aux notes correspondantes. Il est également nécessaire de considérer les silences (demi-soupirs, quarts de soupir, etc.), puisqu'ils peuvent remplacer des notes dans les groupes. Lorsqu'un silence est graphiquement inclus dans le groupe, on peut affirmer qu'il en fait effectivement partie. En revanche, lorsqu'il est placé devant le groupe, deux cas sont à considérer : avec ou sans le silence. Une fois le groupe défini, sa durée totale est calculée. Si elle n'est pas conforme aux durées usuelles pour la métrique de la partition (règle 5 du paragraphe 1.1), alors l'organisation rythmique du groupe est comparée à celles des groupes habituels, et deux nouvelles hypothèses sont générées : la première accroît la durée du groupe jusqu'à la durée usuelle immédiatement supérieure, en changeant un nombre minimal de valeurs ; la seconde porte la durée du groupe à la valeur usuelle immédiatement inférieure. Naturellement, les corrections proposées prennent en compte les points de durée et la classe des symboles, noire ou silence : l'interprétation d'un silence n'est jamais modifiée, et une nouvelle durée de note n'est proposée que si elle est cohérente avec la présence ou l'absence d'un point de durée.

Un degré de possibilité $C_d^{Hl}(g)$ est ensuite affecté à chaque groupe g , dans chaque hypothèse H^l de durée, suivant le nombre $L(g)$ de notes et de silences constituant le groupe, le nombre $l(g)$ de modifications réalisées, et la durée du groupe :

$$C_d^{Hl}(g) = C_l^{Hl}(g) * \pi_d^{Hl}(g) \text{ avec } C_l^{Hl}(g) = 1.0 - \frac{l(g)}{L(g)} \quad (\text{Eq. 5.11})$$

Ce degré est le produit de deux coefficients. Le premier, $C_l^{Hl}(g)$, n'évalue pas directement la validité d'une hypothèse par rapport à la règle 5, mais par rapport à l'interprétation initiale faite sur les durées, qui est considérée comme étant fiable. Ainsi, plus la nouvelle interprétation diffère de la première, plus le degré de possibilité décroît. Le second terme, $\pi_d^{Hl}(g)$, évalue au contraire la possibilité de la durée du groupe. Deux valeurs peuvent être prises : 1.0, si la durée est usuelle, ce qui est le cas pour toute hypothèse correspondant à une proposition de correction, ou 0.5, ce qui est généralement le cas pour les hypothèses initiales.

Prenons un exemple. Considérons un groupe de quatre symboles, soit un quart de soupir suivi de 3 doubles croches groupées ($\text{♩} \underline{\text{♩♩♩}}$), dans une métrique binaire. Supposons l'interprétation initiale de ce groupe exacte, soit $(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}) = 1$. Les hypothèses suivantes sont proposées :




| | | $C_l^{Hl}(g)$ | $\pi_d^{Hl}(g)$ | $C_d^{Hl}(g)$ | Durée |
|----------|---|---------------|-----------------|---------------|--------------|
| H^{l0} |  | 1.00 | 1.0 | 1.00 | 0.75 ou 1.00 |
| H^{l1} |  | 2/3 = 0.67 | 1.0 | 0.67 | 1.00 |
| H^{l2} |  | 1.00 | 1.0 | 1.00 | 0.50 |

Tableau 5.10 : Hypothèses de durée (interprétation initiale exacte)

L'hypothèse initiale H^{l0} ne semble pas correcte si on considère les trois notes seulement (0.75 temps), ce qui justifie de proposer les corrections H^{l1} et H^{l2} (la gestion des triolets est expliquée un peu plus loin) ; mais cette hypothèse est tout à fait possible si on inclut le quart de soupir dans le groupe (1 temps), et c'est pourquoi $\pi_d^{H^{l0}}(g)$ prend la valeur 1. Le degré de possibilité final, $C_d^{H^{l0}}(g)$, est maximal pour les hypothèses H^{l0} et H^{l2} , et c'est la métrique qui les départagera.

Supposons maintenant qu'une erreur de durée ait été faite dans l'interprétation initiale : $(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8}) = \frac{7}{8}$. De nouvelles hypothèses sont proposées, portant non seulement sur le groupe de trois notes, mais aussi sur le groupe incluant le quart de soupir, puisque celui-ci n'atteint plus une durée usuelle :






| | | $C_i^{H^l}(g)$ | $\pi_d^{H^l}(g)$ | $C_d^{H^l}(g)$ | Durée |
|----------|--|----------------|------------------|----------------|------------|
| H^{l0} |  | 1.00 | 0.5 | 0.50 | 5/8 ou 7/8 |
| H^{l1} |  | 2/3 = 0.67 | 1.0 | 0.67 | 1.00 |
| H^{l2} |  | 2/3 = 0.67 | 1.0 | 0.67 | 0.50 |
| H^{l3} |  | 3/4 = 0.75 | 1.0 | 0.75 | 1.00 |
| H^{l4} |  | 1/2 = 0.50 | 1.0 | 0.50 | 0.50 |

Tableau 5.11 : Hypothèses de durée (interprétation initiale fausse)

Le degré de possibilité $C_d^{H^l}(g)$ maximal est maintenant obtenu pour l'hypothèse H^{l3} , qui correspond à la bonne solution. Comme l'hypothèse H^{l0} ne parvient jamais à une durée usuelle, que ce soit avec le silence ou sans lui, le degré de possibilité $\pi_d^{H^{l0}}(g)$ de ce groupe est égal à 0.5. Ce choix permet de ne pas rejeter complètement cette hypothèse, puisque aucune règle stricte de la notation musicale n'indique que les groupes doivent satisfaire à un découpage temporel rigoureux en temps ou fraction de temps, et que cette convention peut être relâchée, pour des questions de phrasé par exemple. Mais on lui affecte un degré de possibilité inférieur, reflétant qu'un groupe atteignant une durée usuelle est a priori préférable.

Appliquons maintenant la méthode sur la mesure (c) (Figure 5.15). Les durées initiales des noires 3, 4, 7 et 9 sont $(\frac{1}{2} + \frac{1}{4} + \frac{1}{2} + \frac{1}{2})$, qui conduisent à une durée totale de 1.25 temps. Le degré de possibilité de cette configuration, notée H^{l0} , est $C_d^{H^{l0}}(g) = 0.5$, et deux nouvelles propositions H^{l1} et H^{l2} sont faites : $(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4})$ avec un degré de possibilité $C_d^{H^{l1}}(g) = 0.25$, et $(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2})$ avec un degré de possibilité $C_d^{H^{l2}}(g) = 0.75$. La seconde correction est donc a priori préférée aux autres hypothèses, et la règle 4, portant sur le nombre de temps par mesure, appliquée durant l'étape de décision, servira à confirmer ce choix.

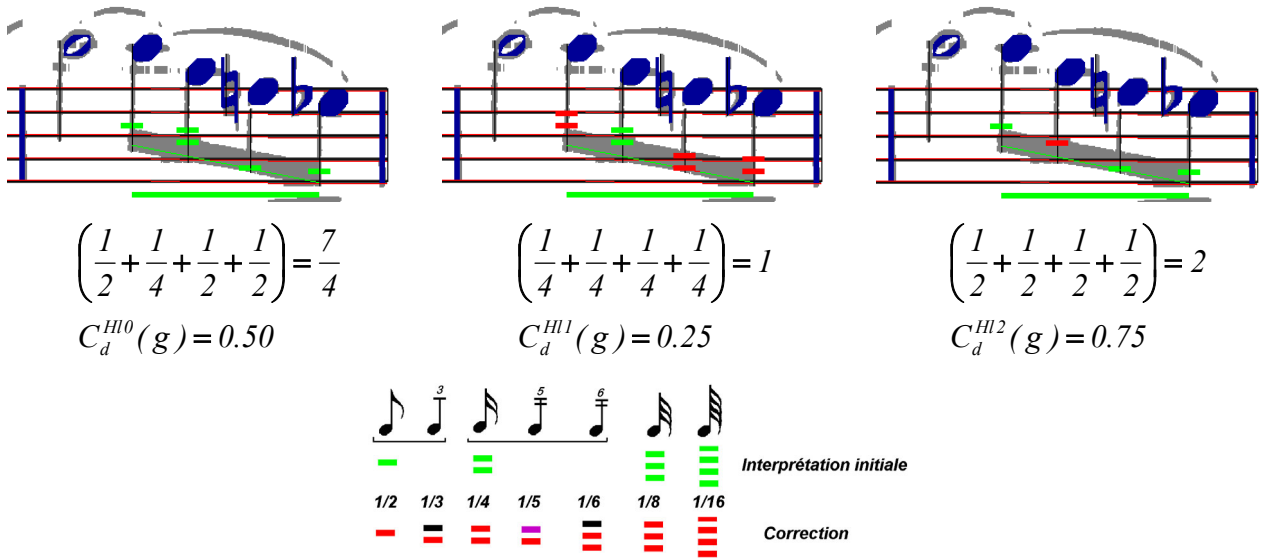


Figure 5.15 : Exemple d'hypothèses de durée (mesure (c))

Cette méthode permet de résoudre le problème des n-olets, dont l'interprétation, sans contexte, est délicate, voire impossible lorsque le petit nombre placé à proximité de la barre de groupe est omis, ce qui est très fréquent. Considérons le cas simple d'un triolet dans une métrique binaire (groupe de 3 croches qui vaut au total 1 temps, Figure 5.16). Si le nombre de barres de groupe est correctement calculé, alors l'interprétation initiale est $\left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right)$, avec un degré de possibilité $C_d^{HI0}(g)$ égal à 0.5 ($C_l^{HI0}(g) = 1$ et $\pi_d^{HI0}(g) = 0.5$). Une seconde hypothèse est générée, $\left(\frac{1}{3} + \frac{1}{3} + \frac{1}{3}\right)$, avec un degré de possibilité maximal, car il ne s'agit pas d'une correction faite sur un décompte erroné des barres de groupe, et que la durée totale (1 temps) est tout à fait possible.

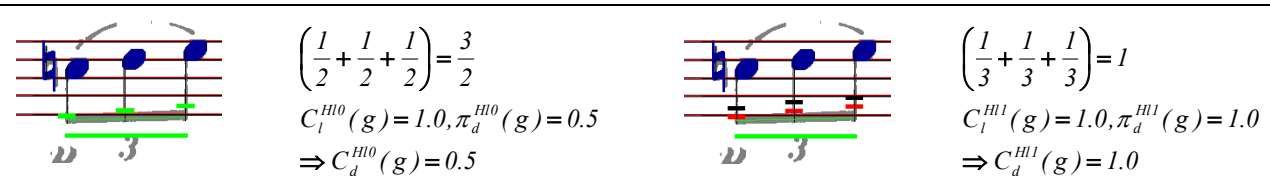


Figure 5.16 : Cas des triolets

La méthode proposée conduit à de bons résultats, notamment pour les n-olets qui sont ainsi très bien reconnus, sans aucune analyse supplémentaire de l'image. Les corrections proposées et choisies par l'algorithme de décision vont effectivement dans le sens d'une nette augmentation du taux de reconnaissance des durées des notes noires (chapitre 7, paragraphe 7.3). Ces résultats reposent beaucoup sur la méthode de détection des barres de groupe, qui est fiable, et qui conduit à une interprétation également fiable des durées. Ainsi, le nombre d'erreurs est faible, et les corrections peuvent être proposées sans trop d'ambiguïté. Dans l'exemple présenté (Figure 5.15), un seul modèle de groupe conduit à une durée de 2 temps, avec une seule modification. Tous les groupes de 2 notes, de 4 notes ou plus, avec une seule durée erronée, sont généralement sans ambiguïté. Les groupes de 3 notes sont plus délicats. Par exemple, l'interprétation initiale

$\left(\frac{1}{8} + \frac{1}{4} + \frac{1}{4}\right) = \frac{5}{8}$ peut être remplacée par $\left(\frac{1}{8} + \frac{1}{8} + \frac{1}{4}\right) = \frac{1}{2}$, ou $\left(\frac{1}{8} + \frac{1}{4} + \frac{1}{8}\right) = \frac{1}{2}$, ou encore $\left(\frac{1}{6} + \frac{1}{6} + \frac{1}{6}\right) = \frac{1}{2}$, avec dans les trois cas un degré de possibilité $C_d^{Hl}(g)$ égal à 0.67. Dans de telles situations, c'est le groupe le plus fréquent qui est proposé comme correction, $\left(\frac{1}{8} + \frac{1}{8} + \frac{1}{4}\right) = \frac{1}{2}$ dans cet exemple.

La figure 5.17 montre des corrections qui ont été réalisées, illustrant la pertinence de la méthode proposée.

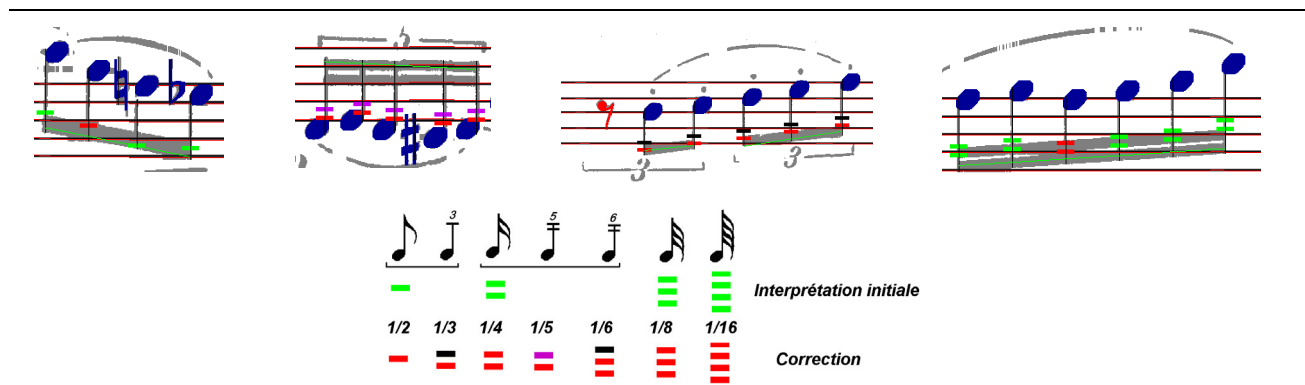


Figure 5.17 : Exemples de corrections de durée pouvant être effectuées par la méthode proposée

5.5. Fusion des informations et décision

La modélisation floue des classes de symboles, et des règles graphiques et syntaxiques de la musique, a abouti à un ensemble de degrés de possibilité et de coefficients de compatibilité, permettant d'évaluer les hypothèses de reconnaissance et leur cohérence mutuelle. L'étape suivante consiste à fusionner toutes ces informations, puis à rechercher la configuration optimale par rapport à tous ces critères, de manière à prendre une décision fiable, cohérente par rapport à la théorie musicale.

L'optimisation globale est réalisée sur chaque mesure, afin de diminuer la complexité du processus de décision. Cette subdivision d'un problème d'optimisation globale en sous-problèmes est très naturelle dans notre application, puisque la mesure correspond elle-même à la décomposition de la musique : en particulier, les symboles sont liés par les règles musicales essentiellement au niveau de la mesure.

Considérons une mesure. Une ou plusieurs hypothèses de reconnaissance ont été générées pour chaque symbole détecté, et toutes les combinaisons d'hypothèses doivent être séquentiellement évaluées. Notons N le nombre d'objets de la mesure, et $k(n,j)$ la classe attribuée à l'objet s_n dans la configuration j ($n=1..N$). Plusieurs hypothèses H^d de durées ont également pu être générées pour cette configuration, combinant cette fois les hypothèses faites sur les $N(j, H^d)$ groupes de notes et silences, indicés par g ($g=1..N(j, H^d)$). La décision est réalisée en deux étapes :

- Fusion de tous les degrés de possibilité et coefficients de compatibilité,
- Décision par maximisation de la fonction résultante.

5.5.1. Fusion

Il est nécessaire dans un premier temps de vérifier la cohérence globale de chaque combinaison d'hypothèses.

Test de cohérence globale

Les conditions nécessaires à cette cohérence sont les suivantes :

- Chaque point de durée doit pouvoir être rapporté à une note ou un silence, en d'autres termes, être dans la zone de recherche d'une note ou d'un silence de la configuration.
- Chaque altération, qui n'est pas une altération à la clé, doit être suivie d'une note, c'est-à-dire avoir un coefficient de compatibilité graphique non nul avec au moins une tête de note (Equation 5.4).
- Chaque objet doit être graphiquement compatible avec tous ses voisins, c'est-à-dire avoir un coefficient de compatibilité graphique final (Equation 5.9) non nul.

Toute configuration d'hypothèses qui ne satisfait pas à ces trois conditions nécessaires est immédiatement éliminée. Pour toutes les autres, les différents degrés de possibilité sont progressivement fusionnés, comme indiqué ci-dessous.

Cohérence des symboles

Le coefficient de compatibilité global $C_t^{(j)}(s_n^{k(n,j)})$ d'un objet s_n , classé en classe $k(n,j)$ dans la configuration j , se déduit de son coefficient de compatibilité graphique $C_p(s_n^{k(n,j)})$ avec les autres symboles (Equation 5.9), fusionné dans le cas des altérations à son coefficient de compatibilité syntaxique $C_s(s_n^{k(n,j)})$ (paragraphe 5.4.1 et 5.4.2) :

- Pour les dièses, bécarres et bémols accidentels :

$$C_t^{(j)}(s_n^{k(n,j)}) = \frac{1}{2} [C_p(s_n^{k(n,j)}) + C_s(s_n^{k(n,j)})] \quad \text{si } C_p(s_n^{k(n,j)}) > 0.5$$

$$C_t^{(j)}(s_n^{k(n,j)}) = C_p(s_n^{k(n,j)}) \quad \text{sinon} \quad \text{Eq. 5.12}$$

- Pour les altérations de tonalité

$$C_t^{(j)}(s_n^{k(n,j)}) = C_p(s_n^{k(n,j)}) C_s(s_n^{k(n,j)}) \quad \text{Eq. 5.13}$$

- Pour les appoggiatures :

$$C_t^{(j)}(s_n^{k(n,j)}) = \frac{1}{2} [C_p(s_n^{k(n,j)}) + 0.5] \quad \text{Eq. 5.14}$$

- Pour les autres classes :

$$C_t^{(j)}(s_n^{k(n,j)}) = C_p(s_n^{k(n,j)}) \quad \text{Eq. 5.15}$$

Pour les altérations accidentelles, la fusion des critères graphiques et syntaxiques est donc réalisée par une moyenne, avec une condition qui exprime que la compatibilité syntaxique n'est prise en compte que si l'altération est à peu près correctement placée par rapport à la note. Cette condition étant réalisée, l'opérateur fournit un résultat qui donne une importance égale aux deux critères. Les appoggiatures étant souvent confondues avec les altérations, on moyenne leur coefficient de compatibilité graphique avec un pseudo coefficient de compatibilité syntaxique, égal à 0.5. Pour les altérations de tonalité, le produit permet de rejeter toute altération à la clé syntaxiquement incorrecte, donc fautive de manière certaine, ou graphiquement incompatible. Pour toutes les autres classes, il n'y a aucun critère syntaxique à prendre en compte.

Le degré de possibilité de l'objet s_n , classé en classe $k(n,j)$ dans la configuration j , est ensuite exprimé comme le produit de son degré de possibilité d'appartenance à la classe $k(n,j)$ (Eq. 5.1) et de sa compatibilité graphique et syntaxique avec les autres symboles de la mesure dans cette configuration (Eq. 5.15) :

$$\pi(s_n^{k(n,j)}, j) = \pi_{k(n,j)}(s_n^{k(n,j)}) \cdot C_t^{(j)}(s_n^{k(n,j)}) \quad \text{Eq. 5.16}$$

Ces premiers résultats sont ensuite moyennés pour former le degré de possibilité final de la configuration de symboles j :

$$Conf_r(j) = \frac{1}{N} \sum_{n=1}^N \pi(s_n^{k(n,j)}, j) \quad \text{Eq. 5.17}$$

Le coefficient $Conf_r(j)$ exprime le degré de possibilité global de la configuration j de symboles, les critères relatifs aux durées des notes devant encore être ajoutés. La fusion est réalisée pour chaque symbole (Equation 5.16) par l'opérateur multiplication (t-norme), exprimant que les deux critères, degré de possibilité d'appartenance aux classes et cohérence par rapport aux autres objets de la mesure, doivent être simultanément vérifiés. Il s'agit d'une règle sévère. Lorsque l'hypothèse H_0 est choisie (absence de symbole), on pose $\pi(s_n^{k(n,j)}, j) = 0$, ce qui correspond à l'idée qu'un objet analysé doit nécessairement correspondre à un symbole à reconnaître. La fusion sur les symboles de la mesure est ensuite obtenue par une moyenne. Cette fois, il s'agit plutôt d'un compromis, qui permet de ne pas rejeter une configuration d'hypothèses incluant un symbole "très peu possible". Une égale importance est donnée à tous les symboles de la mesure.

Cohérence temporelle

Pour chaque configuration de symboles j , il peut y avoir au plus cinq hypothèses faites sur les durées de chaque groupe de notes g : l'hypothèse initiale, deux corrections en considérant les notes et les silences inclus dans le groupe, mais sans aucun silence qui le précède, deux corrections en considérant ce même groupe, avec le silence qui le précède, s'il existe. Un degré de possibilité $C_d^{(j,H^d)}(g)$ a été attribué à chacune de ces hypothèses, suivant l'équation 5.11. Toutes les combinaisons d'hypothèses de durée de la configuration j , notées (j,H^d) , sont évaluées par fusion des degrés de possibilité $C_d^{(j,H^d)}(g)$:

$$Conf_d(j, H^d) = \left[\frac{1}{N(j, H^d)} \sum_{g=1}^{N(j, H^d)} C_d^{(j, H^d)}(g) \right] \left[1 - \frac{N(j, H^d)}{N'(j, H^d)} \right] \quad (\text{Eq. 5.18})$$

Le premier terme représente la moyenne des coefficients $C_d^{(j, H^d)}(g)$ obtenus sur les groupes de notes de la configuration de durée (j, H^d) . Il est ensuite multiplié par un deuxième facteur, dans lequel $N'(j, H^d)$ représente le nombre total de notes groupées. Le second terme est donc d'autant plus élevé que les notes sont rassemblées en peu de groupes. Cela permet d'exclure les configurations pour lesquelles une mauvaise interprétation d'un symbole scinde un groupe cohérent en deux : le cas par exemple d'une altération placée devant une note incluse dans un groupe, confondue avec une note isolée.

Degré de possibilité final

Le résultat final est déduit des degrés de possibilité portant sur les symboles (Eq. 5.17) et sur les durées (Eq. 5.18) :

$$Conf(j, H^d) = Conf_r(j) * Conf_d(j, H^d) \quad (\text{Eq. 5.19})$$

C'est le produit des deux critères, exprimant qu'ils doivent être simultanément vérifiés. L'utilisation du produit, au lieu d'une t-norme telle que le minimum, ou encore d'une moyenne, rend cette règle plus sévère.

5.5.2. Décision

Toutes les informations, relatives aux symboles ou au contexte musical, ont donc été fusionnées en un unique coefficient, qui exprime le degré de possibilité de la configuration (j, H^d) . Une seule règle n'a pas encore été évaluée : il s'agit de la durée de la mesure, et de sa conformité par rapport à la métrique, donnée en paramètre d'entrée du programme.

Soit $D(j, H^d)$ la somme des durées de tous les symboles, qui est aussi la somme des durées des groupes de notes et de silences (dépendant de j et H^d), et de tous les silences isolés (dépendant seulement de j). L'algorithme de décision choisit la configuration (j, H^d) qui satisfait au mieux aux deux critères suivants, indiqués par ordre de priorité :

- la durée totale $D(j, H^d)$ de la mesure est exacte,
- le degré de possibilité $Conf(j, H^d)$ est maximisé.

Cela signifie que l'algorithme choisit, parmi les configurations qui satisfont à la métrique, celle qui maximise $Conf(j, H^d)$. Si aucune configuration n'obtient un nombre de temps correct, alors on maximise simplement $Conf(j, H^d)$.

Ainsi, la dernière règle musicale (règle 4), qui est stricte, est incorporée dans l'étape finale de décision. Toutes les règles présentées au paragraphe 1.1 ont donc été exprimées et participent à la décision. La modélisation floue a permis de fusionner des critères très hétérogènes : les

informations sur la forme des symboles, des informations contextuelles d'ordre graphique ou syntaxique, et cela quel que soit le nombre de symboles impliqués, qu'ils soient proches ou distants dans la mesure. Ainsi, l'ambiguïté constatée à l'issue de l'analyse individuelle des symboles est considérablement réduite, et une décision cohérente par rapport à la théorie musicale peut être prise, par optimisation globale de tous les critères.

L'inconvénient d'une telle méthodologie est le risque d'explosion combinatoire : le nombre total de configurations est égal au produit du nombre d'hypothèses faites sur chaque objet analysé. Le coût de calcul peut donc être rédhibitoire pour des mesures qui contiennent beaucoup de symboles et qui présentent une forte ambiguïté : jusqu'à $5 \cdot 10^6$ configurations à évaluer dans nos expérimentations. Un garde-fou très grossier a donc été mis en place : les hypothèses les moins possibles sont supprimées, pour que le nombre maximal de combinaisons n'excède pas 10^5 . Naturellement, des optimisations plus fines peuvent être trouvées. Soulignons également que la combinatoire est généralement tout à fait acceptable (en moyenne 350 combinaisons générées par mesure). Actuellement, le temps moyen de la modélisation floue et de l'étape de décision est d'environ 0.3 seconde par portée, sur un Pentium 4 à 3.2 GHz.

5.6. Exemples

Nous présentons dans ce paragraphe quelques exemples illustrant la méthode : modélisation floue, fusion et décision. Nous traiterons tout d'abord la mesure (c) de la figure 5.1, puis la mesure (f) de la figure 5.2, et enfin la mesure présentée en conclusion du chapitre 4 (Figure 4.13)

5.6.1. Exemple 1

Considérons la mesure (c), et quelques combinaisons d'hypothèses (Figure 5.18, Tableaux 5.12). Le nombre total de configurations de symboles est égal à 192, mais seules 42 d'entre elles satisfont aux conditions nécessaires. Une configuration jugée d'emblée impossible est, par exemple, la configuration j_1 (Tableau 5.12a), pour laquelle les deux bécarres superposés ont chacun un coefficient de compatibilité graphique nul. Cet exemple montre comment les cas de double détection laissés en suspens sont maintenant bien résolus.

Prenons maintenant deux configurations particulières d'hypothèses, j_2 et j_3 , parmi les 18 possibles, et comparons-les. Les tableaux 5.12b et 5.12c présentent les résultats issus de la modélisation floue. Les hypothèses sur les durées des notes seront données ultérieurement. A noter qu'un fa dièse est à la clé dans cette partition.

Les configurations j_2 et j_3 diffèrent par les objets 2, 5, et 8. Les degrés de possibilité d'appartenance aux classes sont dans tous les cas favorables à la configuration correcte, à savoir j_2 . Les coefficients de compatibilité $C_i(s_n^{k(n,j)})$ sont également meilleurs pour les altérations 5 et 8 correctes : dans le cas de l'altération 5, c'est le critère syntaxique qui fait la différence, alors que

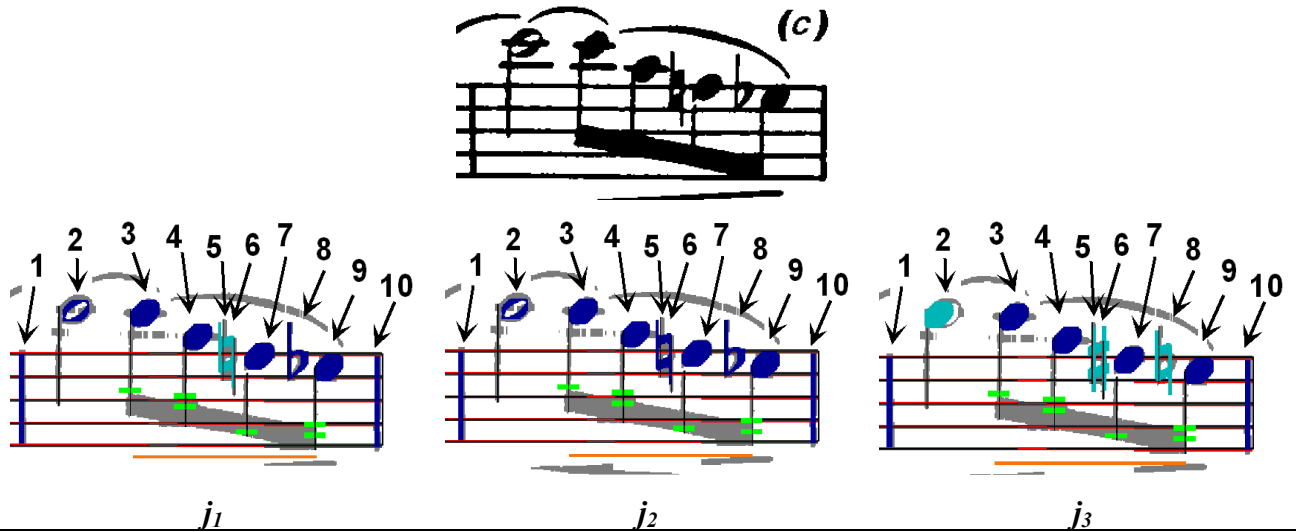


Figure 5.18 : Image originale et configurations d'hypothèses pour la mesure (c)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|
| | | o | ● | ● | h | h | ● | b | ● | |
| $\pi_{k(n,j_1)}(s_n^{k(n,j_1)})$ | 1.00 | 0.35 | 0.32 | 0.25 | 0.13 | 0.13 | 0.65 | 0.38 | 0.62 | 1.00 |
| $C_p(s_n^{k(n,j_1)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 | 0.00 | 1.00 | 0.91 | 0.91 | |
| $C_s(s_n^{k(n,j_1)})$ | | | | | 1.00 | 0.50 | | 0.50 | | |
| $C_t(s_n^{k(n,j_1)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.25 | 1.00 | 0.71 | 0.91 | 1.00 |

(a) Configuration j_1

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|
| | | o | ● | ● | h | (-) | ● | b | ● | |
| $\pi_{k(n,j_2)}(s_n^{k(n,j_2)})$ | 1.00 | 0.35 | 0.32 | 0.50 | 0.13 | | 0.65 | 0.38 | 0.62 | 1.00 |
| $C_p(s_n^{k(n,j_2)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 0.91 | 0.91 | |
| $C_s(s_n^{k(n,j_2)})$ | | | | | 1.00 | | | 0.50 | | |
| $C_t(s_n^{k(n,j_2)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 0.71 | 0.91 | 1.00 |
| $\pi(s_n^{k(n,j_2)}, j_2)$ | 1.00 | 0.35 | 0.32 | 0.50 | 0.13 | 0.00 | 0.65 | 0.27 | 0.56 | 1.00 |

(b) Configuration j_2 (exacte)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|
| | | ● | ● | ● | # | (-) | ● | h | ● | |
| $\pi_{k(n,j_3)}(s_n^{k(n,j_3)})$ | 1.00 | 0.00 | 0.32 | 0.50 | 0.03 | | 0.65 | 0.23 | 0.62 | 1.00 |
| $C_p(s_n^{k(n,j_3)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | | 1.00 | 0.71 | 0.71 | 1.00 |
| $C_s(s_n^{k(n,j_3)})$ | | | | | 0.50 | | | 0.50 | | |
| $C_t(s_n^{k(n,j_3)})$ | 1.00 | 1.00 | 1.00 | 1.00 | 0.75 | | 1.00 | 0.61 | 0.71 | 1.00 |
| $\pi(s_n^{k(n,j_3)}, j_3)$ | 1.00 | 0.00 | 0.32 | 0.50 | 0.02 | 0.00 | 0.65 | 0.14 | 0.44 | 1.00 |

(c) Configuration j_3

Tableaux 5.12 : Degrés de possibilité et coefficients de compatibilité dans 3 configurations différentes

pour l'altération 8, c'est le critère graphique. Remarquons que la note 9 a un coefficient de compatibilité graphique égal à celui de l'altération qui la précède, car c'est la relation altération/note qui est la moins bien satisfaite, les autres coefficients de compatibilité impliqués dans l'équation 5.9 étant tous égaux à 1. Au total, la configuration j_2 obtient un degré de possibilité $Conf_r(j_2) = 4,78 / 10 = 0.48$ contre $Conf_r(j_3) = 4.07 / 10 = 0.41$.

Considérons maintenant l'aspect temporel, en commençant par la configuration j_2 . Le premier groupe de notes est en fait une blanche isolée, de 2 temps. Son degré de possibilité est $C_d^{H^0}(g_1) = 1.0$. Le second groupe, comprenant quatre croches, est mal interprété à cause de l'épaississement local de la barre de groupe. Sa durée est de 1.75 temps, et trois hypothèses sont générées avec les degrés de possibilité suivants : $C_d^{H^0}(g_2) = 0.5$ pour l'hypothèse initiale, $C_d^{H^1}(g_2) = 0.25$ et $C_d^{H^2}(g_2) = 0.75$ pour les deux autres (paragraphe 5.4.3, Figure 5.15). En combinant les hypothèses sur les deux groupes, on obtient donc trois hypothèses de durée pour la mesure dans la configuration j_2 (Equation 5.18) :

$$\begin{array}{lll}
 (j_2, H^{d0}) & D(j_2, H^{d0}) = \left(2 \right) + \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{2} + \frac{1}{2} \right) = 3.75 & Conf_d(j_2, H^{d0}) = \left[\frac{1.0 + 0.5}{2} \right] \left[1 - \frac{2}{5} \right] = 0.45 \\
 (j_2, H^{d1}) & D(j_2, H^{d1}) = \left(2 \right) + \left(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} \right) = 3.00 & Conf_d(j_2, H^{d1}) = \left[\frac{1.0 + 0.25}{2} \right] \left[1 - \frac{2}{5} \right] = 0.38 \\
 (j_2, H^{d2}) & D(j_2, H^{d2}) = \left(2 \right) + \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} \right) = 4.00 & Conf_d(j_2, H^{d2}) = \left[\frac{1.0 + 0.75}{2} \right] \left[1 - \frac{2}{5} \right] = 0.53
 \end{array}$$

Les résultats sur les durées sont identiques pour la configuration j_3 , sauf que la durée est diminuée d'un temps puisque le premier groupe est une noire. Une seule configuration permet donc d'atteindre une durée de mesure correcte : la configuration (j_2, H^{d2}) , qui obtient par ailleurs le plus grand degré de possibilité final : $Conf(j_2, H^{d2}) = 0.48 * 0.53 = 0,25$.

Conclusion

Cet exemple montre la pertinence de la modélisation floue et des méthodes de fusion, qui conduisent à un degré de possibilité final maximal pour la configuration exacte. Il illustre également la complémentarité des règles graphiques et des règles syntaxiques, en particulier pour l'analyse des altérations : chaque critère apporte un élément d'information, plus ou moins discriminant, et participe à la décision finale.

Il faut également souligner l'importance des coefficients de compatibilité graphique, qui permettent de détecter les défauts graves de segmentation et de les résoudre, et de la règle stricte portant sur la métrique, qui élimine à elle seule un grand nombre de configurations, et qui contribue à la validation des corrections de durées.

5.6.2. Exemple 2

Considérons maintenant la mesure (f). La figure 5.19 et le tableau 5.13 montrent les hypothèses initiales, avec les scores de corrélation associés :

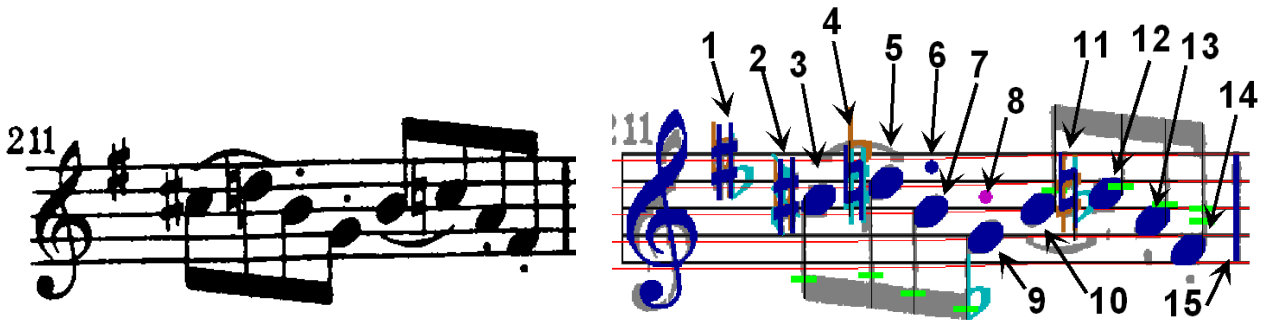


Figure 5.19 : Image originale et hypothèses de reconnaissance superposées à l'image originale (Mesure (f))

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| H0 | | | | | | (-) | | (-) | | | | | | | |
| H1 | # | # | • | b | • | . | • | . | • | • | b | • | • | • | |
| | 0.65 | 0.68 | 0.75 | 0.49 | 0.76 | 0.62 | 0.93 | 0.53 | 0.89 | 0.88 | 0.64 | 0.86 | 0.87 | 0.87 | 0.93 |
| H2 | b | z | | # | | | | | b | | b | | | | |
| | 0.58 | 0.51 | | 0.47 | | | | | 0.64 | | 0.57 | | | | |
| H3 | b | b | | b | | | | | | | # | | | | |
| | 0.51 | 0.50 | | 0.41 | | | | | | | 0.50 | | | | |

Tableau 5.13 : Hypothèses de reconnaissance et scores de corrélation (Mesure (f))

Le nombre total de configurations est égal à 648. L'hypothèse H0 est présente pour l'objet 6, bien que le point ait obtenu un score de corrélation supérieur au seuil de décision, parce que le coefficient de compatibilité graphique avec la note 7 est strictement inférieur à 1 (paragraphe 5.3.6). Certaines configurations sont immédiatement éliminées : toutes celles qui incluent les bémols 1, 4 ou 9, le bécarre 1 ou le point 8 (compatibilité graphique nulle). Finalement, seules 36 configurations sont valides. Examinons quatre d'entre elles, notées j_1 , j_2 , j_3 et j_4 .

Configuration j_1 :

La configuration j_1 , qui est en fait la solution (Figure 5.20 et tableau 5.14), satisfait à toutes les conditions nécessaires, et elle peut donc être évaluée. Quelques résultats méritent d'être commentés. Les degrés de possibilité d'appartenance aux classes sont faibles pour les objets 4 et 5, à cause des défauts d'impression locaux (pixels noirs connectant ces deux symboles, augmentant la variabilité intra-partition). Les coefficients de compatibilité graphique sont parfois strictement inférieurs à 1, reflétant la qualité médiocre de la mise en page et de l'impression : altérations positionnées trop près de la note altérée (objets 2 et 4), ou de l'objet précédent (objet 11), espace entre les noires faible (objets 13 et 14). La modélisation floue permet donc de prendre en compte tous ces défauts, sans toutefois rejeter la configuration d'hypothèses. Enfin, on peut noter l'intérêt de l'intégration des règles portant sur les altérations : en particulier, le coefficient de compatibilité syntaxique est maximal pour le bécarre 11, mettant en évidence sa parfaite cohérence avec le dièse 2 (annulation de cette altération). Ce dernier a lui-même un coefficient élevé (0.75), puisqu'il

apporte une information compatible avec la tonalité. Le degré de possibilité de cette configuration, pour la partie symbolique, est $Conf_r(j_1) = 6.32 / 15 = 0.42$.

La durée initiale de la mesure n'est pas compatible avec la métrique (4 temps par mesure), à cause de la mauvaise interprétation faite sur la dernière croche. Deux groupes de notes sont détectés. Le premier (notes 3, 5, 7 et 9) atteint une durée usuelle (1 temps), et aucune correction n'est donc proposée. Le second (notes 10, 12, 13 et 14), a en revanche une durée égale à 1.75 temps $\left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{4}\right)$, et deux corrections sont proposées, $\left(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}\right)$ et $\left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right)$. Il y a donc au total 3 combinaisons de durée pour la configuration de symboles j_1 , évaluées suivant l'équation 5.18 :

$$\begin{array}{ll} (j_1, H^{d0}) & D(j_1, H^{d0}) = \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{4}\right) = 3.75 & Conf_d(j_1, H^{d0}) = \left[\frac{1.0 + 0.5}{2}\right] \left[1 - \frac{2}{8}\right] = 0.56 \\ (j_1, H^{d1}) & D(j_1, H^{d1}) = \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}\right) = 3.00 & Conf_d(j_1, H^{d1}) = \left[\frac{1.0 + 0.25}{2}\right] \left[1 - \frac{2}{8}\right] = 0.47 \\ (j_1, H^{d2}) & D(j_1, H^{d2}) = \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right) = 4.00 & Conf_d(j_1, H^{d2}) = \left[\frac{1.0 + 0.75}{2}\right] \left[1 - \frac{2}{8}\right] = 0.66 \end{array}$$

Seule la combinaison H^{d2} permet d'atteindre la durée de 4 temps dans la mesure. La configuration (j_1, H^{d2}) peut donc être retenue, avec un degré de possibilité global $Conf(j_1, H^{d2}) = 0.42 * 0.66 = 0.28$.

Configuration j_2 :

Considérons maintenant la deuxième configuration d'hypothèses, j_2 , qui ne diffère de la précédente que par les objets 4 et 11, passés de bécarres à dièses (Figure 5.21). Le tableau 5.15 indique les nouveaux résultats (en orange, les coefficients qui ont changé).

La comparaison des deux tableaux montre que les degrés de possibilité et les coefficients de compatibilité graphique sont favorables à la solution correcte, de manière très significative pour l'objet 11, beaucoup moins pour l'objet 4.

Le coefficient de compatibilité syntaxique est égal à 0.5 pour l'altération 4, que ce soit dans l'hypothèse dièse ou dans l'hypothèses bécarre : en effet, un ré dièse a été reconnu dans la mesure précédente, et l'une des hypothèses ne peut être préférée à l'autre (tableau 5.8). Ce critère n'est donc pas discriminant pour l'objet 4, mais une application stricte de la théorie musicale aurait rejeté la solution correcte (bécarre inutile). De plus, la prise en compte des altérations dans les mesures précédentes conduit à choisir le bécarre de préférence au dièse (degré de possibilité final de 0.07 contre 0.05).

On remarque l'interaction forte entre les objets 2 et 11, très distants dans la mesure : le coefficient de compatibilité syntaxique qui était maximal lorsque l'objet 11 était classé bécarre (bécarre qui annule un dièse), est maintenant égal à 0.5 lorsque l'objet 11 est classé en dièse, puisque cette fois, il s'agit d'un rappel d'altération non obligatoire.

La modélisation des règles sur les altérations est donc un élément important dans la prise de décision finale, permettant de renforcer les configurations cohérentes tout en prenant en compte la souplesse des règles. Le degré de possibilité de la configuration j_2 est $Conf_r(j_2) = 5.85 / 15 = 0.39$, donc bien inférieur à $Conf_r(j_1)$. L'analyse syntaxique des durées

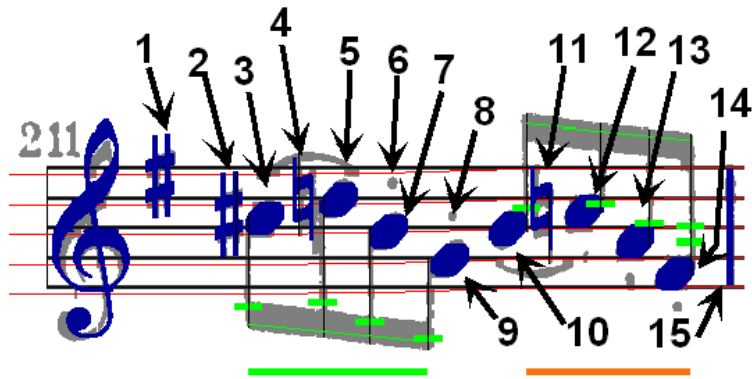


Figure 5.20 : Hypothèses de reconnaissance dans la configuration j_1 (Mesure (f))

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|----------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | # | # | ♮ | ♭ | ♮ | (-) | ♮ | (-) | ♮ | ♮ | ♭ | ♮ | ♮ | ♮ | |
| $\pi_{k(n,j_1)}(s_n^{k(n,j_1)})$ | 0.53 | 0.60 | 0.22 | 0.10 | 0.25 | | 0.78 | | 0.66 | 0.63 | 0.47 | 0.56 | 0.59 | 0.59 | 1.00 |
| $C_p(s_n^{k(n,j_1)})$ | 1.00 | 0.98 | 0.98 | 0.96 | 0.96 | | 1.00 | | 1.00 | 0.83 | 0.83 | 1.00 | 0.67 | 0.67 | 1.00 |
| $C_s(s_n^{k(n,j_1)})$ | 1.00 | 0.75 | | 0.50 | | | | | | | 1.00 | | | | |
| $C_t(s_n^{k(n,j_1)})$ | 1.00 | 0.87 | 0.98 | 0.73 | 0.96 | | 1.00 | | 1.00 | 0.83 | 0.92 | 1.00 | 0.67 | 0.67 | |
| $\pi(s_n^{k(n,j_1)}, j_1)$ | 0.53 | 0.52 | 0.21 | 0.07 | 0.24 | 0.00 | 0.78 | 0.00 | 0.66 | 0.52 | 0.44 | 0.56 | 0.40 | 0.40 | 1.00 |

Tableau 5.14 : Configuration j_1 (correcte)

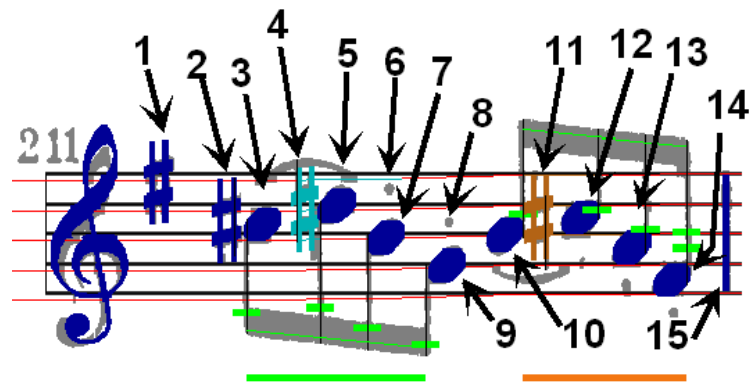


Figure 5.21 : Hypothèses de reconnaissance dans la configuration j_2 (Mesure (f))

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|----------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | # | # | ♮ | # | ♮ | (-) | ♮ | (-) | ♮ | ♮ | # | ♮ | ♮ | ♮ | |
| $\pi_{k(n,j_2)}(s_n^{k(n,j_2)})$ | 0.53 | 0.60 | 0.22 | 0.08 | 0.25 | | 0.78 | | 0.66 | 0.63 | 0.15 | 0.56 | 0.59 | 0.59 | 1.00 |
| $C_p(s_n^{k(n,j_2)})$ | 1.00 | 0.98 | 0.98 | 0.93 | 0.93 | | 1.00 | | 1.00 | 0.67 | 0.67 | 1.00 | 0.67 | 0.67 | 1.00 |
| $C_s(s_n^{k(n,j_2)})$ | 1.00 | 0.75 | | 0.50 | | | | | | | 0.50 | | | | |
| $C_t(s_n^{k(n,j_2)})$ | 1.00 | 0.87 | 0.98 | 0.71 | 0.93 | | 1.00 | | 1.00 | 0.67 | 0.59 | | | | |
| $\pi(s_n^{k(n,j_2)}, j_2)$ | 0.53 | 0.52 | 0.21 | 0.05 | 0.23 | 0.00 | 0.78 | 0.00 | 0.66 | 0.42 | 0.09 | 0.56 | 0.40 | 0.40 | 1.00 |

Tableau 5.15 : Configuration j_2 (incorrecte : deux erreurs sur les altérations)

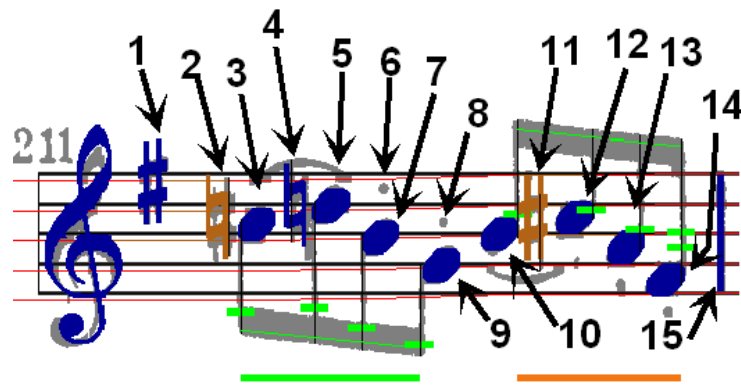


Figure 5.22 : Hypothèses de reconnaissance dans la configuration j_3 (Mesure (f))

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| | # | b | • | b | • | (-) | • | (-) | • | • | # | • | • | • | |
| $\pi_{k(n,j_3)}(s_n^{k(n,j_3)})$ | 0.53 | 0.13 | 0.22 | 0.10 | 0.25 | | 0.78 | | 0.66 | 0.63 | 0.15 | 0.56 | 0.59 | 0.59 | 1.00 |
| $C_p(s_n^{k(n,j_3)})$ | 1.00 | 0.91 | 0.91 | 0.96 | 0.96 | | 1.00 | | 1.00 | 0.67 | 0.67 | 1.00 | 0.67 | 0.67 | 1.00 |
| $C_s(s_n^{k(n,j_3)})$ | 1.00 | 0.50 | | 0.50 | | | | | | | 1.00 | | | | |
| $C_t(s_n^{k(n,j_3)})$ | 1.00 | 0.71 | 0.91 | 0.73 | 0.96 | | 1.00 | | 1.00 | 0.67 | 0.83 | | | | |
| $\pi(s_n^{k(n,j_3)}, j_3)$ | 0.53 | 0.09 | 0.20 | 0.07 | 0.24 | 0.00 | 0.78 | 0.00 | 0.66 | 0.42 | 0.13 | 0.56 | 0.40 | 0.40 | 1.00 |

Tableau 5.16 : Configuration j_3 (incorrecte)

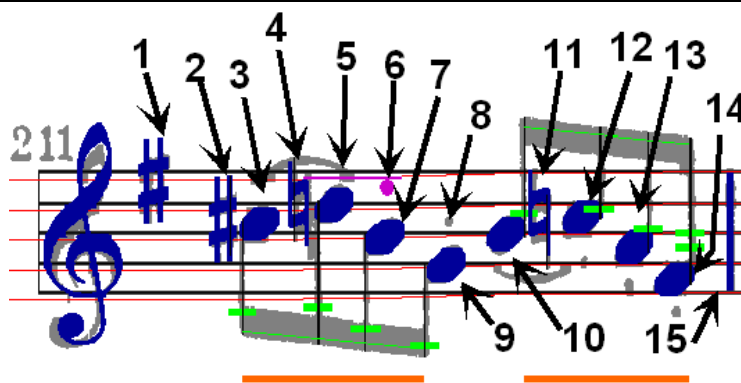


Figure 5.23 : Hypothèses de reconnaissance dans la configuration j_4 (Mesure (f))

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| | # | # | • | b | • | . | • | (-) | • | • | b | • | • | • | |
| $\pi_{k(n,j_4)}(s_n^{k(n,j_4)})$ | 0.53 | 0.60 | 0.22 | 0.10 | 0.25 | 0.05 | 0.78 | | 0.66 | 0.63 | 0.47 | 0.56 | 0.59 | 0.59 | 1.00 |
| $C_p(s_n^{k(n,j_4)})$ | 1.00 | 0.98 | 0.98 | 0.96 | 0.96 | 0.33 | 0.33 | | 1.00 | 0.83 | 0.83 | 1.00 | 0.67 | 0.67 | 1.00 |
| $C_s(s_n^{k(n,j_4)})$ | 1.00 | 0.75 | | 0.50 | | | | | | | 1.00 | | | | |
| $C_t(s_n^{k(n,j_4)})$ | 1.00 | 0.87 | 0.98 | 0.73 | 0.96 | | 0.33 | | 1.00 | 0.83 | 0.92 | 1.00 | 0.67 | 0.67 | |
| $\pi(s_n^{k(n,j_4)}, j_4)$ | 0.53 | 0.52 | 0.21 | 0.07 | 0.24 | 0.12 | 0.24 | 0.00 | 0.66 | 0.52 | 0.44 | 0.56 | 0.40 | 0.40 | 1.00 |

Tableau 5.17 : Configuration j_4 (incorrecte)

est identique à celle faite en j_1 , et au total, la configuration (j_1, H^{d2}) est donc toujours retenue.

Configuration j_3 :

Cette troisième configuration j_3 est illustrée par la figure 5.22 et le tableau 5.16. Par rapport à la configuration j_1 , l'objet 2 est classé en bécarré au lieu de dièse, et l'objet 11 en dièse au lieu de bécarré.

Les altérations 2 et 11 sont globalement compatibles avec la tonalité, et cohérentes entre elles. C'est pourquoi les coefficients de compatibilité $C_i(s_n^{k(n-j_i)})$ sont assez élevés (0.71 et 0.83), peu inférieurs aux valeurs prises dans la configuration j_1 (0.87 et 0.92). La différence est quand même significative, les critères graphiques et syntaxiques renforçant tous la bonne solution. Les degrés de possibilité d'appartenance aux classes sont quant à eux nettement moins ambigus que les scores de corrélation, et nettement favorables à la configuration j_1 . Le degré de possibilité de la configuration j_3 est égal à $Conf_r(j_3) = 5.45 / 15 = 0.36$ et donc bien inférieur à celui obtenu pour la configuration correcte j_1 . L'analyse des durées est toujours identique à celle de la configuration j_1 , et cette dernière est donc toujours retenue.

Configuration j_4 :

Terminons par la configuration j_4 , qui obtient un nombre de temps correct, à cause de la confusion faite sur le point de staccato et de l'erreur sur la durée de la noire 14 (Figure 5.23, Tableau 5.17).

La position du point (objet 6) est très ambiguë, et il est impossible de classer cet objet de façon certaine, en point de durée ou en point de staccato, sans prise en compte du contexte. Le contexte local est intégré sous la forme des degrés de compatibilité graphique entre le point et les notes 5 et 7 : le coefficient de compatibilité graphique avec la note précédente (objet 5) est égal à 1.0, et le coefficient de compatibilité avec la note suivante (objet 7) est non nul (0.33). Les deux interprétations, point de durée ou de staccato sont donc possibles. La configuration j_4 est donc valide, mais pénalisée par le faible degré de possibilité final de la note 7, peu compatible avec le point : $Conf_r(j_4) = 5.82 / 15 = 0.39$ ($Conf_r(j_1) = 0.42$). L'hypothèse H0 (absence de symbole) est donc préférée lorsque les classifications proposées s'avèrent peu compatibles avec les autres symboles de la mesure.

La durée totale de la mesure est cependant correcte (4 temps), mais incorrectement répartie. Aucune correction ne peut être proposée pour le premier groupe, car il n'existe pas d'arrangement usuel avec un unique point de durée. Son degré de possibilité est égal à 0.5. Pour le second groupe, les corrections sont identiques à celles proposées dans la configuration j_1 . On obtient donc encore trois configurations de durées :

$$\begin{array}{ll}
 (j_4, H^{d0}) & D(j_4, H^{d0}) = \left(\frac{1}{2} + \frac{3}{4} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{4}\right) = 4.00 & Conf_d(j_4, H^{d0}) = \left[\frac{0.5+0.5}{2}\right] \left[1 - \frac{2}{8}\right] = 0.38 \\
 (j_4, H^{d1}) & D(j_4, H^{d1}) = \left(\frac{1}{2} + \frac{3}{4} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}\right) = 3.25 & Conf_d(j_4, H^{d1}) = \left[\frac{0.5+0.25}{2}\right] \left[1 - \frac{2}{8}\right] = 0.28 \\
 (j_4, H^{d2}) & D(j_4, H^{d2}) = \left(\frac{1}{2} + \frac{3}{4} + \frac{1}{2} + \frac{1}{2}\right) + \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right) = 4.25 & Conf_d(j_4, H^{d2}) = \left[\frac{0.5+0.75}{2}\right] \left[1 - \frac{2}{8}\right] = 0.47
 \end{array}$$

Seule la configuration (j_4, H^{d0}) aboutit donc à un nombre correct de temps, avec un degré de

possibilité de 0.38. Le degré de possibilité final est égal à $Conf(j_4, H^{d0}) = 0.39 * 0.38 = 0.15$, cette fois nettement inférieur à $Conf(j_1, H^{d2}) = 0.28$. Nous pouvons donc observer sur cet exemple la pertinence de la modélisation floue qui, bien que n'écartant pas d'emblée la possibilité d'un point de durée, permet finalement de rejeter cette configuration au profit de la bonne solution : par l'évaluation de degrés de compatibilité graphique locaux, mais aussi par l'ajout d'un contexte plus large portant sur les groupes de notes.

Décision et conclusion

La configuration d'hypothèse retenue est la configuration j_1 , avec une correction de durée sur la dernière note (j_1, H^{d2}). C'est effectivement la solution exacte. Cet exemple montre comment la fusion des différents critères permet de comparer la cohérence globale des différentes configurations d'hypothèses, et finalement d'aboutir à la bonne interprétation, malgré la qualité médiocre du document original. Les défauts d'impression et de mise en page se traduisent par des degrés de possibilité d'appartenance aux classes, et des coefficients de compatibilité graphique assez faibles, mais les objets concernés ne sont pas rejetés, grâce à la modélisation floue. Les critères syntaxiques apportent des informations complémentaires, plus globales, et contribuent à départager des configurations concurrentes. On a pu constater la pertinence des modèles proposés, qui prennent en compte la souplesse de l'écriture musicale, en particulier au niveau des altérations.

Tous ces critères, une fois fusionnés, permettent d'établir une relation d'ordre entre les différentes configurations, et d'aboutir à la bonne solution, par maximisation. L'importance relative de chaque degré de possibilité ou coefficient de compatibilité dépend de la configuration testée, et cela prouve l'intérêt d'une optimisation globale.

On remarque également sur cet exemple le haut degré de dépendance entre les symboles, liés par des règles de notation qui agissent localement (règles graphiques), ou entre symboles distants (règles syntaxiques). La modélisation floue permet de fusionner toutes ces informations hétérogènes, et c'est l'un des aspects les plus importants de la méthode que nous proposons.

5.6.3. Exemple 3

Reprenons l'exemple clôturant le chapitre 4 (Figures 4.13 et 5.24), pour lequel la classe réelle de quelques objets n'est pas mémorisée en hypothèse H1. Le choix correspondant au plus haut score de corrélation n'aboutit donc pas à la solution. Certains scores de corrélation sont indiqués dans le tableau 5.18, et les degrés de possibilité d'appartenance aux classes dans le tableau 5.19 (en gras, les valeurs pour les hypothèses exactes). On constate immédiatement une réduction de l'ambiguïté. Comparons trois des configurations qui satisfont aux conditions préliminaires.

Configuration j_1 :

La configuration d'hypothèses correcte, notée j_1 , est décrite par la figure 5.25 et le tableau 5.20. On constate une très bonne compatibilité graphique, ainsi qu'une très bonne cohérence entre les altérations et la tonalité. En particulier la séquence bécarre (objet 5) annulant le dièse à la clé, puis dièse (objet 26) annulant le premier bécarre, se concrétise par des coefficients de compatibilité syntaxique maximaux. Le degré de possibilité final est égal à $Conf_r(j_1) = 0.48$. Il n'y a pas

d'erreurs de durée, et un degré de possibilité maximal est donc attribué à chaque groupe de notes. Par conséquent, $Conf_d(j_1, H^{d0}) = [1 - 4/17] = 0.76$. La configuration est donc retenue, avec un degré de possibilité final égal à $Conf(j_1, H^{d0}) = 0.37$.

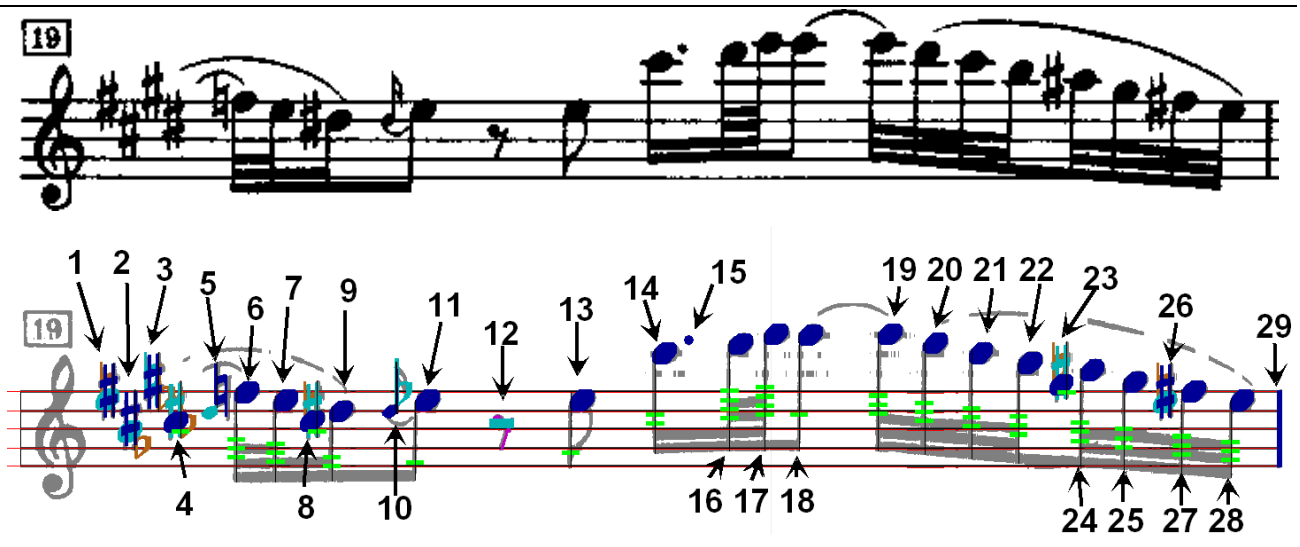


Figure 5.24 : Image originale et hypothèses de reconnaissance (Ex. Chapitre 4)

| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| H0 | | | | | | | | (-) | | | | | | |
| H1 | | | | | | | | | | | | | | |
| | 0.65 | 0.73 | 0.85 | 0.60 | 0.83 | 0.77 | 0.77 | 0.74 | 0.73 | 0.83 | 0.65 | 0.83 | 0.63 | 0.85 |
| H2 | | | | | | | | | | | | | | |
| | 0.62 | 0.46 | | 0.58 | | 0.48 | | 0.63 | | | 0.65 | | 0.56 | |
| H3 | | | | | | | | | | | | | | |
| | 0.57 | | | 0.54 | | | | | | | 0.54 | | 0.53 | |

Tableau 5.18 : Hypothèses de reconnaissance et scores de corrélation

| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| H0 | | | | | | | | (-) | | | | | | |
| H1 | | | | | | | | | | | | | | |
| | 0.08 | 0.55 | 0.61 | 0.00 | 0.55 | 0.50 | 0.40 | 0.55 | 0.24 | 0.63 | 0.08 | 0.55 | 0.45 | 0.61 |
| H2 | | | | | | | | | | | | | | |
| | 0.43 | 0.00 | | 0.33 | | 0.00 | | 0.08 | | | 0.50 | | 0.00 | |
| H3 | | | | | | | | | | | | | | |
| | 0.57 | | | 0.08 | | | | | | | 0.08 | | 0.05 | |

Tableau 5.19 : Hypothèses de reconnaissance et degrés de possibilité d'appartenance aux classes

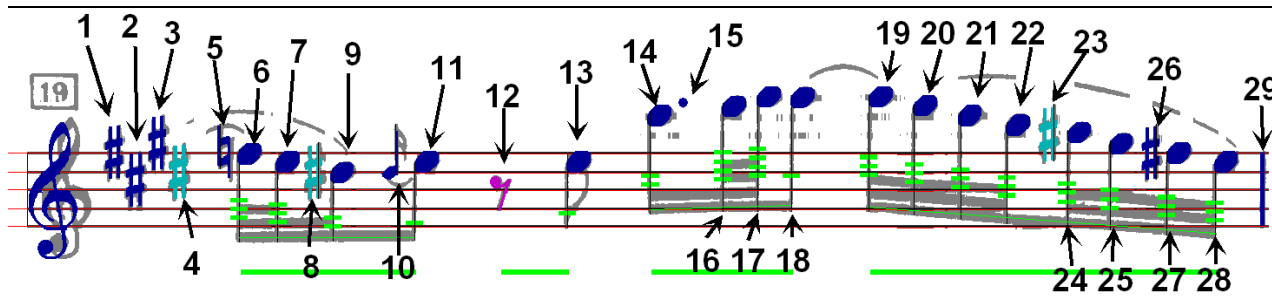


Figure 5.25 : Hypothèses de reconnaissance dans la configuration j_1 (correcte)

| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | # | b | ● | # | ● | ♪ | ● | 7 | ● | . | # | ● | # | ● |
| $\pi_{k(n,j_1)}(s_n^{k(n,j_1)})$ | 0.43 | 0.55 | 0.61 | 0.33 | 0.55 | 0.50 | 0.39 | 0.55 | 0.29 | 0.63 | 0.50 | 0.55 | 0.45 | 0.61 |
| $C_p(s_n^{k(n,j_1)})$ | 1.00 | 1.00 | 1.00 | 0.98 | 0.98 | 0.80 | 0.80 | 1.00 | 1.00 | 1.00 | 0.95 | 0.95 | 0.98 | 0.98 |
| $C_s(s_n^{k(n,j_1)})$ | 1.00 | 1.00 | | 0.50 | | | | | | | 0.75 | | 1.00 | |
| $C_t(s_n^{k(n,j_1)})$ | 1.00 | 1.00 | 1.00 | 0.74 | 0.98 | | 0.80 | 1.00 | 1.00 | 1.00 | 0.73 | 0.95 | 0.99 | 0.98 |
| $\pi(s_n^{k(n,j_1)}, j_1)$ | 0.43 | 0.55 | 0.61 | 0.24 | 0.54 | 0.65 | 0.32 | 0.55 | 0.29 | 0.63 | 0.36 | 0.53 | 0.45 | 0.59 |

Tableau 5.20 : Configuration j_1 (correcte)

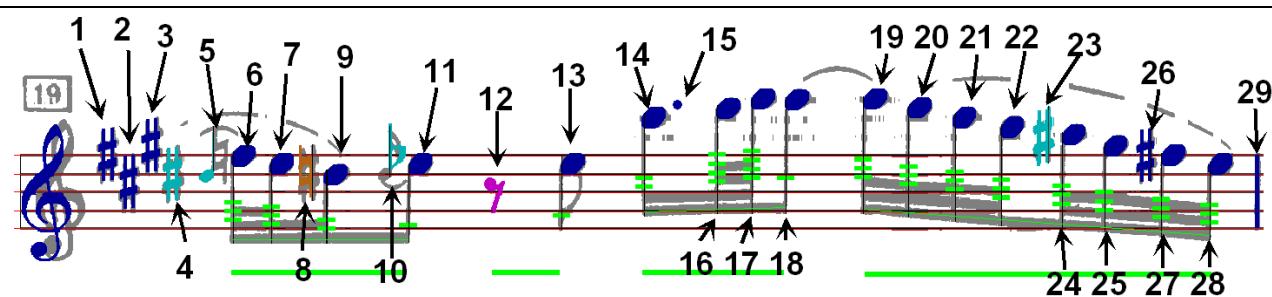


Figure 5.26 : Hypothèses de reconnaissance dans la configuration j_2 (3 erreurs)

| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | # | ♪ | ● | b | ● | b | ● | 7 | ● | . | # | ● | # | ● |
| $\pi_{k(n,j_2)}(s_n^{k(n,j_2)})$ | 0.43 | 0.00 | 0.61 | 0.08 | 0.55 | 0.00 | 0.39 | 0.55 | 0.29 | 0.63 | 0.50 | 0.55 | 0.45 | 0.61 |
| $C_p(s_n^{k(n,j_2)})$ | 1.00 | 0.74 | 0.74 | 0.96 | 0.96 | 0.47 | 0.47 | 1.00 | 1.00 | 1.00 | 0.95 | 0.95 | 0.98 | 0.98 |
| $C_s(s_n^{k(n,j_2)})$ | 1.00 | | | 1.00 | | 0.75 | | | | | 0.75 | | 0.50 | |
| $C_t(s_n^{k(n,j_2)})$ | 1.00 | 0.62 | 0.74 | 0.98 | 0.96 | 0.47 | 0.47 | 1.00 | 1.00 | 1.00 | 0.73 | 0.95 | 0.74 | 0.98 |
| $\pi(s_n^{k(n,j_2)}, j_2)$ | 0.43 | 0.00 | 0.45 | 0.07 | 0.53 | 0.00 | 0.18 | 0.55 | 0.29 | 0.63 | 0.36 | 0.53 | 0.33 | 0.59 |

Tableau 5.21 : Configuration j_2 (3 erreurs)

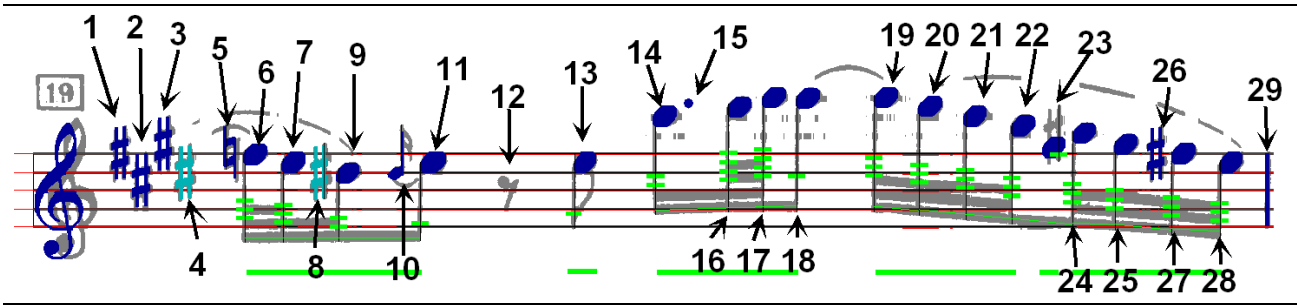


Figure 5.27 : Hypothèses de reconnaissance dans la configuration j_3 (2 erreurs)

| | 4 | 5 | 6 | 8 | 9 | 10 | 11 | 12 | 14 | 15 | 23 | 24 | 26 | 27 |
|----------------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| | # | b | • | # | • | • | • | (-) | • | • | • | • | # | • |
| $\pi_{k(n,j_3)}(s_n^{k(n,j_3)})$ | 0.43 | 0.55 | 0.61 | 0.33 | 0.55 | 0.50 | 0.39 | | 0.29 | 0.63 | 0.08 | 0.55 | 0.45 | 0.61 |
| $C_p(s_n^{k(n,j_3)})$ | 1.00 | 1.00 | 1.00 | 0.98 | 0.98 | 0.80 | 0.80 | | 1.00 | 1.00 | 1.00 | 0.95 | 0.98 | 0.98 |
| $C_s(s_n^{k(n,j_3)})$ | 1.00 | 1.00 | | 0.50 | | | | | | | | | 1.00 | |
| $C_t(s_n^{k(n,j_3)})$ | 1.00 | 1.00 | 1.00 | 0.74 | 0.98 | | 0.80 | | 1.00 | 1.00 | 1.00 | 0.95 | 0.99 | 0.98 |
| $\pi(s_n^{k(n,j_3)}, j_3)$ | 0.43 | 0.55 | 0.61 | 0.24 | 0.54 | 0.65 | 0.32 | 0.00 | 0.29 | 0.63 | 0.08 | 0.53 | 0.45 | 0.59 |

Tableau 5.22 : Configuration j_3 (2 erreurs)

Configuration j_2 :

Comparons les résultats avec ceux obtenus dans une autre configuration d'hypothèses j_2 (Figure 5.26, tableau 5.21). On voit sur cette mesure l'importance des degrés de possibilité d'appartenance aux classes, qui pénalisent très fortement les hypothèses erronées. C'est ce critère qui élimine correctement le bécarre 8, qui obtient par ailleurs de très bons coefficients de compatibilité graphique et syntaxique (bécarre qui annule l'altération à la clé), en moyenne supérieur au coefficient $C_t(s_n^{k(n,j_1)})$ obtenu pour le dièse pourtant exact (0.98 contre 0.74). Le bémol 10 ayant un coefficient de compatibilité graphique inférieur à 0.5 avec la note suivante, sa compatibilité syntaxique (0.75 : première apparition d'un mi bémol) n'est pas prise en compte. La comparaison des degrés de compatibilité syntaxique obtenus par l'objet 26, dans les configurations j_1 et j_2 , illustre de nouveau comment les altérations interagissent dans cette modélisation.

Le degré de possibilité final de cette configuration est $Conf_r(j_2) = 9.03 / 29 = 0.31$, ce qui la classe nettement derrière la configuration exacte.

Configuration j_3 :

La configuration j_3 (Figure 5.27, Tableau 5.22) illustre davantage la fusion des règles relatives aux groupes de notes. Deux erreurs sont présentes : l'omission du demi-soupir 12, et la confusion faite sur le dièse 23, classé en croche. Elles se compensent d'un point de vue temporel, si bien que la durée totale de la configuration est correcte (4 temps).

Le degré de possibilité de cette configuration, pour la partie symbolique, est $Conf_r(j_3) = 13.12 / 29 = 0.45$, contre 0.48 pour la configuration j_1 . Six groupes de notes ont été trouvés cette fois, puisque aucune barre de groupe reliant les notes 22 et 23 ou 23 et 24 n'a été détectée. Chacun de ces groupes g_i a une durée totale parfaitement cohérente avec la métrique,

respectivement 1, 0.5, 1.0, 0.5, 0.5 et 0.5 temps, et on obtient donc pour chacun un degré de possibilité $C_d^{H^0}(g_i)$ égal à 1. Le degré de possibilité final relatif à la métrique est cette fois $Conf_d(j_4, H^{d0}) = [1 - 6/17] = 0.65$, bien inférieur à $Conf_d(j_1, H^{d0}) = 0.76$. Le second facteur de l'équation 5.18 permet donc, à bon escient, de renforcer la configuration (j_1, H^{d0}) par rapport à la configuration (j_3, H^{d0}) . Le degré de possibilité final est de $Conf(j_3, H^{d0}) = 0.45 * 0.65 = 0.30$ contre $Conf(j_1, H^{d0}) = 0.37$. La discrimination entre les configurations j_1 et j_2 est plus nette après l'incorporation du critère syntaxique relatif aux groupements de notes.

Conclusion

Cet exemple illustre particulièrement l'intérêt de la modélisation floue des classes de symboles, puisque les degrés de possibilité d'appartenance aux classes sont beaucoup plus pertinents que les scores de corrélation. Il montre de nouveau l'importance des règles syntaxiques, notamment de l'évaluation du découpage temporel de la mesure. Une décision portant sur chaque symbole pris individuellement, sur la base des scores de corrélation, aurait été erronée ; la modélisation floue permet au contraire d'extraire la configuration d'hypothèses correcte, par une analyse pertinente des scores de corrélation obtenus sur toute la page de musique, et par l'intégration des règles musicales dans le processus de décision.

5.7. Conclusion

Nous avons proposé dans ce chapitre une méthodologie complète, fondée sur la théorie des ensembles flous et des possibilités, permettant de modéliser les classes de symboles et d'intégrer les principales règles graphiques et syntaxiques de la notation musicale. Cette méthodologie apporte une réponse à un certain nombre de problèmes essentiels, qui n'avaient pas encore été traités ou suffisamment formalisés dans la littérature : la prise en compte de la variabilité des polices de symboles, des imprécisions sur la forme et la position des objets (dues en particulier aux difficultés de segmentation), de la souplesse des règles musicales. Les règles syntaxiques concernant les altérations et la métrique ont pu être intégrées dans le processus de décision, malgré leur très grande flexibilité, et bien qu'elles concernent de nombreux symboles distants dans la mesure. Ce point est particulièrement novateur. Un autre point fort de la méthode proposée est qu'elle permet de fusionner des informations très hétérogènes, de manière à prendre une décision globale, cohérente par rapport à la notation. Les nombreux exemples présentés ont illustré comment cette méthode conduit, à partir d'un ensemble d'hypothèses de reconnaissance, à l'interprétation correcte. Ils ont en particulier démontré l'importance de l'optimisation globale, évaluant tout le contexte. Cet aspect est également très novateur, la plupart des systèmes présentés procédant par décisions locales successives. Les résultats obtenus sur toute la base de données seront présentés dans le chapitre 7 (paragraphe 7.3).

CHAPITRE 6

Améliorations de la robustesse

Nous avons présenté dans les chapitres précédents un système complet de reconnaissance, procédant séquentiellement en trois étapes : prétraitements et segmentation de l'image, génération d'hypothèses de reconnaissance, modélisation et intégration des règles musicales permettant d'évaluer les différentes configurations d'hypothèses et de prendre une décision. Ce système est fondamentalement unidirectionnel. Des procédures rétroactives sont néanmoins proposées dans ce chapitre, afin d'améliorer la robustesse de la méthode proposée.

Les taux de reconnaissance (Chapitre 7, paragraphe 7.3), indiquent une bonne fiabilité du système. Néanmoins, la fiabilité d'un système de reconnaissance ne se mesure pas exclusivement par ce biais. Il est également très important que ce système soit capable de donner des indications sur les erreurs potentielles, afin de faciliter la correction. En effet, même si les taux de reconnaissance sont bons, la vérification systématique de tous les résultats est une tâche extrêmement longue et fastidieuse, qui finalement diminue considérablement le gain de temps réalisé par rapport à une édition entièrement manuelle. Un objectif important est donc l'indication automatique d'erreurs potentielles.

Un deuxième axe d'amélioration concerne toutes les procédures qui permettent d'adapter le système de reconnaissance à une partition particulière, dans le but d'améliorer sa reconnaissance. Ce point se rapporte essentiellement à l'apprentissage des modèles de classe, spécifiques à la partition analysée. Les nombreux paramètres définis lors des différentes étapes restent quant à eux inchangés, puisqu'ils modélisent des connaissances génériques sur l'écriture musicale. Cet apprentissage peut être réalisé à partir d'un extrait de la partition, reconnu par le logiciel d'OMR et corrigé par l'utilisateur. Les procédures d'indication d'erreurs facilitent l'intervention de ce dernier, et une telle démarche est certainement très bien acceptée en pratique, si elle conduit à de réels gains de reconnaissance sur le reste de la partition. Notons que ces procédures d'adaptation ne doivent pas conduire à des modèles trop restrictifs, et que la souplesse de la méthodologie doit être maintenue, notamment pour gérer la variabilité intra-partition.

6.1. Détection automatique d'erreurs

Les erreurs de reconnaissance sont de quatre sortes : symbole ajouté, confusion, symbole

manquant, erreur de durée de note. Nous proposons d'analyser la solution retenue par l'algorithme de décision, afin d'indiquer à l'utilisateur les symboles potentiellement erronés. Les critères utilisés sont les degrés de possibilité d'appartenance aux classes, la décomposition rythmique de la mesure et la compatibilité graphique.

6.1.1. Indication des ajouts et des confusions potentiels

Considérons de nouveau chaque symbole s_n^k , classé en classe k par l'algorithme de décision, avec un degré de possibilité $\pi_k(s_n^k)$ d'appartenance à la classe k (Equation 5.1). Une faible valeur de $\pi_k(s_n^k)$ peut être révélatrice d'une erreur de classification du symbole s_n . La règle suivante est donc appliquée : si $\pi_k(s_n^k) < t_s^k$, alors le symbole s_n est indiqué comme potentiellement faux.

Les seuils t_s^k ont été déterminés par apprentissage. Les bases d'apprentissage et de test ont été constituées à partir des hypothèses de reconnaissance : elles incluent les hypothèses exactes, et les hypothèses erronées mais graphiquement possibles. Par exemple, les hypothèses d'altérations accidentelles qui ne sont compatibles avec aucune note sont écartées, toutes les autres sont intégrées. La moitié des exemples de chaque classe est utilisée en apprentissage, l'autre moitié en généralisation. L'apprentissage a été réalisé par une optimisation globale, avec le critère suivant : maximisation du taux de détection d'erreurs, pour un taux de fausses alarmes global inférieur à 2.0%. Ce pourcentage semble en effet raisonnable puisqu'il correspond à l'indication superflue de moins de 10 symboles par page de musique. Il est parfaitement acceptable en pratique, s'il permet d'éviter une vérification systématique de tous les symboles par un pointage performant des erreurs effectivement commises.

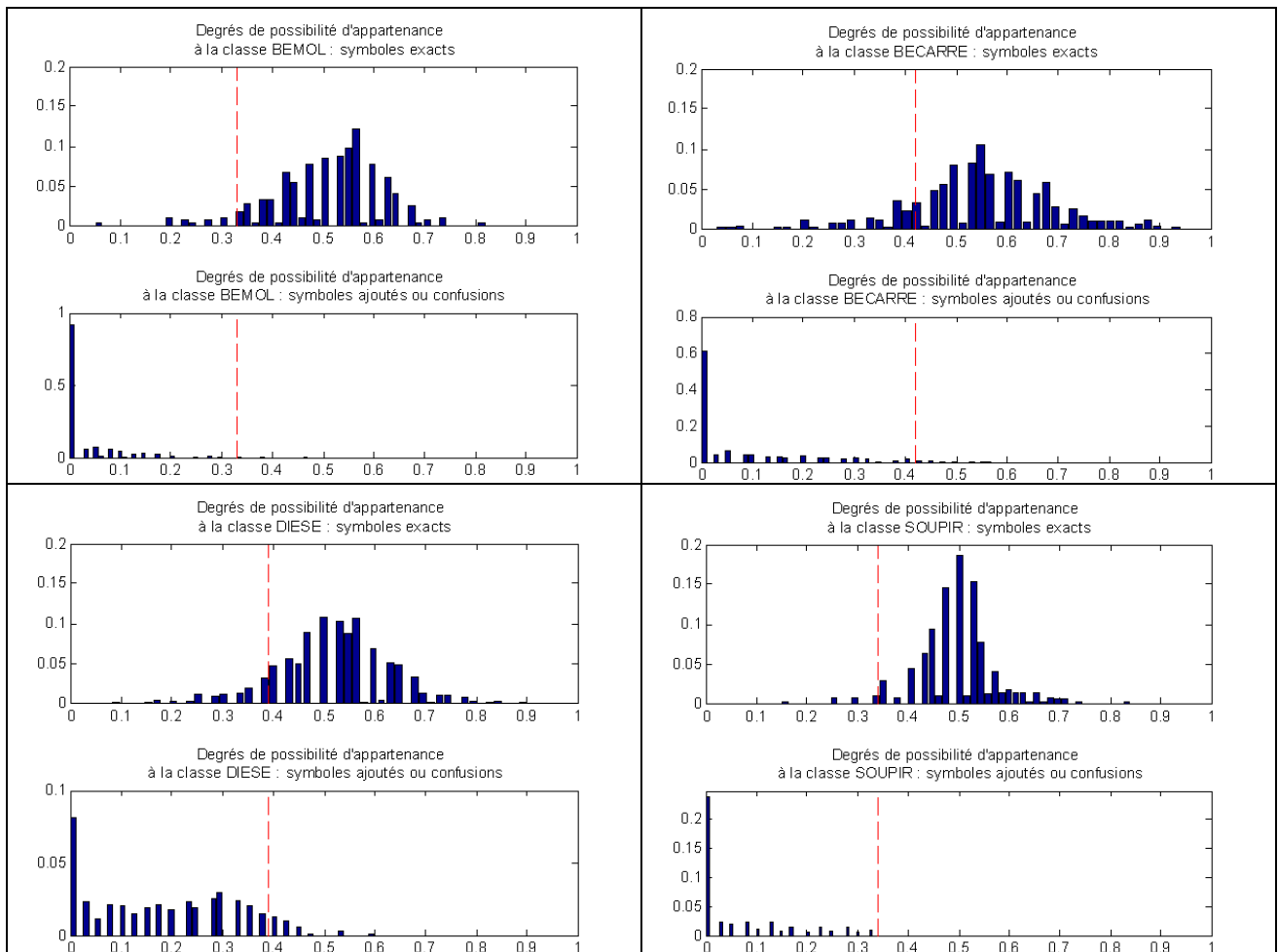
Le tableau 6.1 indique le jeu final de coefficients obtenu, avec le taux de détection d'erreurs et le taux de fausses alarmes, par classe, en apprentissage et en généralisation. La classe des barres de mesure n'est pas représentée, car les degrés de possibilité d'appartenance aux classes ne sont pas évalués (voir paragraphe 5.1). Les classes "soupir" (♯), "huitième de soupir" et "ronde", ne sont pas non plus traitées, car le nombre d'erreurs est trop faible pour permettre un apprentissage significatif. Le taux global de détection des erreurs, sur les douze classes traitées, est de 95.1% en apprentissage, et 95.0% en généralisation, avec des taux de fausses alarmes égaux à 1.9% en apprentissage et 2.1% en généralisation. Ces résultats préliminaires laissent donc présager une bonne efficacité de la méthode proposée. Les taux obtenus en sortie du programme de reconnaissance seront exposés dans le chapitre suivant (paragraphe 7.6).

La figure 6.1 indique, pour quelques classes k , la répartition des degrés de possibilité d'appartenance aux classes, sur l'ensemble des symboles de la base d'apprentissage qui ont été correctement classifiés, et sur l'ensemble des symboles ajoutés ou erronés, de la base d'apprentissage également. Les histogrammes présentés ont été normalisés par rapport au nombre de prototypes de chaque ensemble. On constate que les distributions sont assez bien séparées, ce qui justifie la méthode choisie. Le recouvrement se traduit par des non-détections et des fausses alarmes. L'optimisation globale permet de trouver le meilleur compromis, compte tenu des fréquences d'occurrence relatives des classes, et de leur probabilité d'erreur.

| Classe k | t_s^k | Taux Détections | Taux Fausses alarmes |
|------------|---------|-----------------|----------------------|
| b | 0.33 | 99.2 | 4.0 |
| | | 98.7 | 4.0 |
| q | 0.42 | 96.8 | 13.3 |
| | | 97.1 | 12.6 |
| ♪ | 0.35 | 97.3 | 10.9 |
| | | 97.1 | 14.5 |
| # | 0.39 | 91.9 | 10.2 |
| | | 90.7 | 11.8 |
| ● | 0.17 | 94.9 | 0.3 |
| | | 94.9 | 0.4 |
| ○ | 0.12 | 76.7 | 1.3 |
| | | 86.7 | 1.6 |
| TOTAL | | 95.1 | 1.9 |
| | | 95.0 | 2.1 |

| Classe k | t_s^k | Taux Détections | Taux Fausses alarmes |
|------------|---------|-----------------|----------------------|
| - | 0.38 | 93.8 | 0 |
| | | 93.8 | 0 |
| - | 0.35 | 85.5 | 4.2 |
| | | 84.1 | 1.3 |
| ~ | 0.34 | 100 | 2.6 |
| | | 98.3 | 2.6 |
| . | 0.35 | 93.0 | 5.9 |
| | | 93.0 | 6.8 |
| 7 | 0.30 | 82.4 | 2.3 |
| | | 82.4 | 4.1 |
| 7 | 0.36 | 99.6 | 27.9 |
| | | 100.0 | 22.1 |

Tableau 6.1 : Seuils t_s^k pour l'indication des erreurs potentielles, et résultats sur la base d'apprentissage (cases blanches) et sur la base de test (cases grises)



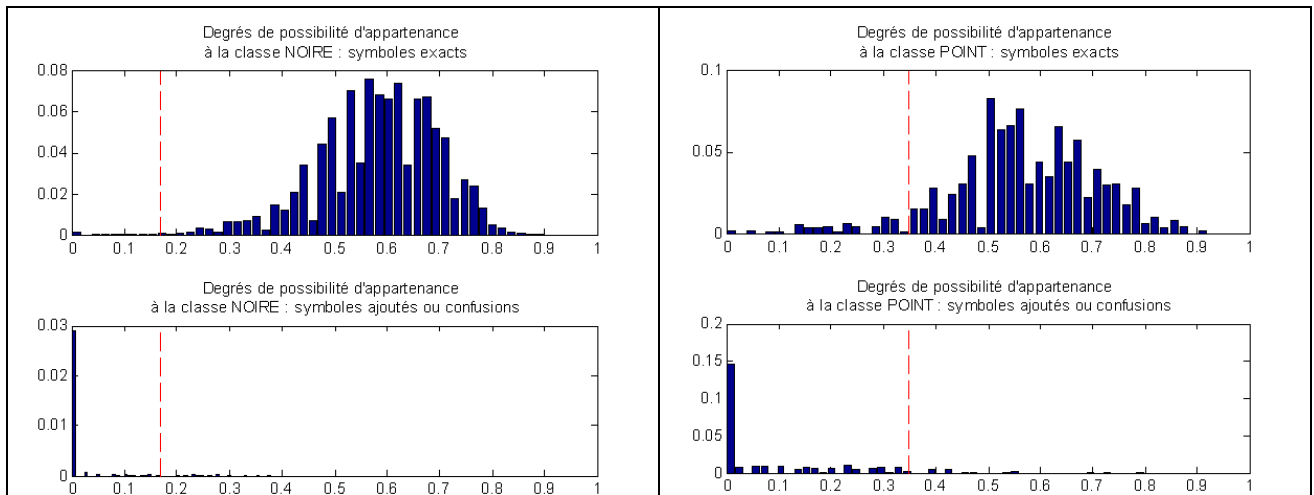


Figure 6.1 : Degrés de possibilité d'appartenance aux classes, pour les hypothèses exactes et pour des hypothèses erronées, et seuils t_s^k (pointillés rouges) servant à l'indication des erreurs potentielles.

La figure 6.2 donne quelques exemples d'indications de symboles ajoutés ou mal classés. Les trois exemples de gauche sont corrects, alors que l'indication sur la blanche de la dernière mesure est une fausse alarme.

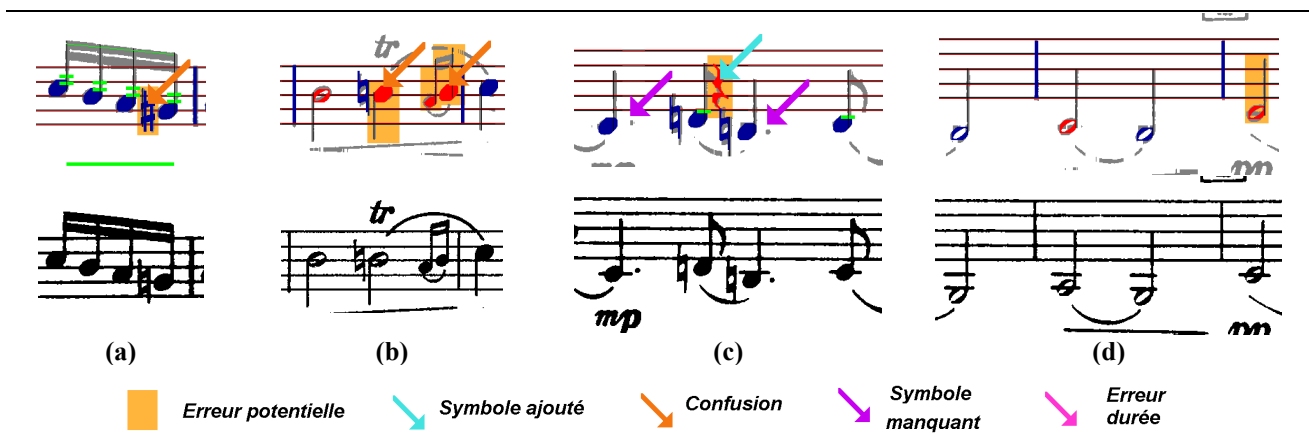


Figure 6.2 : Exemples d'indications d'erreurs (ajouts ou confusions)

6.1.2. Détection des symboles manquants

Ce type d'erreur peut avoir deux causes : soit la non-détection du symbole lors de l'étape de segmentation, soit le rejet des hypothèses de classification (i.e. choix de l'hypothèse H_0 : "absence de symbole"). Le premier cas ne peut plus être repéré. En revanche, l'analyse des objets détectés mais rejetés peut donner une indication sur les symboles manquants.

Un symbole s_n détecté et non retenu est donc indiqué comme potentiellement manquant s'il satisfait aux conditions suivantes :

- Le symbole est situé à une distance inférieure à deux interlignes des lignes extrêmes de la portée : ce test permet d'éviter de nombreuses fausses alarmes relatives aux inscriptions

diverses présentes entre les portées (titres, indications de phrasé, etc.).

- Le symbole est graphiquement compatible avec les hypothèses de classification retenues.

Le second critère reprend les résultats de la modélisation floue des règles graphiques. Le coefficient $C_p(s_n^k)$, exprimant la compatibilité graphique du symbole s_n classé en classe k , est calculé (Equation 5.9), en considérant les symboles retenus par l'algorithme de décision, et en supposant que la classe k du symbole s_n testé est celle qui maximise le degré de possibilité $\pi_k(s_n^k)$ d'appartenance à la classe (Equation 5.1). Le symbole est indiqué comme potentiellement manquant si le coefficient $C_p(s_n^k)$ est non nul. Plus de la moitié des symboles manquants sont ainsi correctement indiqués, avec un taux de fausses alarmes de 1% (paragraphe 7.6). La figure 6.3 donne des exemples. Les indications sur les quatre mesures de gauche sont pertinentes, tandis que celle de la mesure de droite est une fausse alarme.

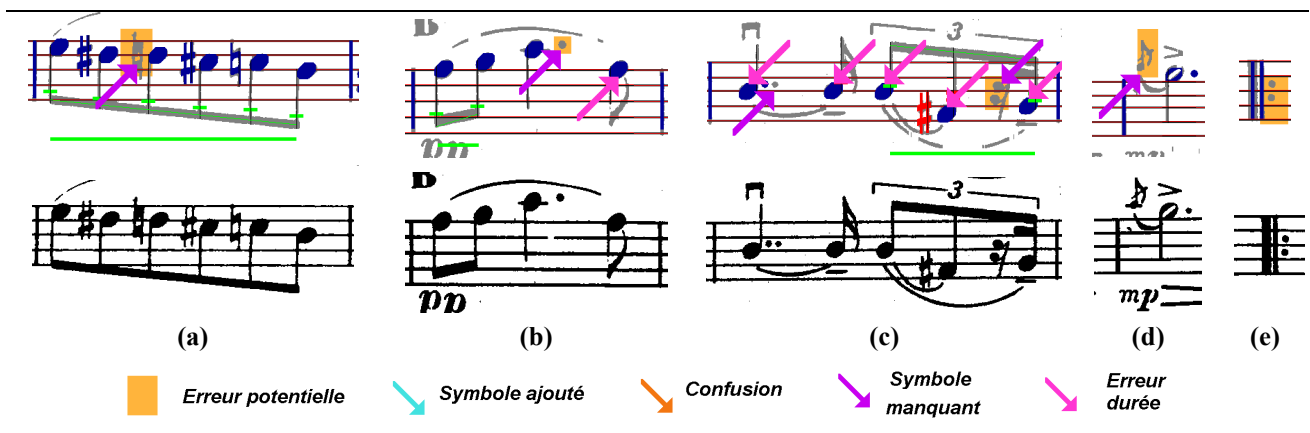


Figure 6.3 : Exemples d'indications d'erreurs (symboles manquants)

6.1.3. Analyse de la rythmique

La cohérence rythmique de la mesure constitue le dernier critère permettant de repérer des erreurs de classification, ou de calcul de durée. Nous indiquons tout d'abord toutes les mesures qui ne satisfont pas à la contrainte stricte de métrique (règle 4, paragraphe 1.1).

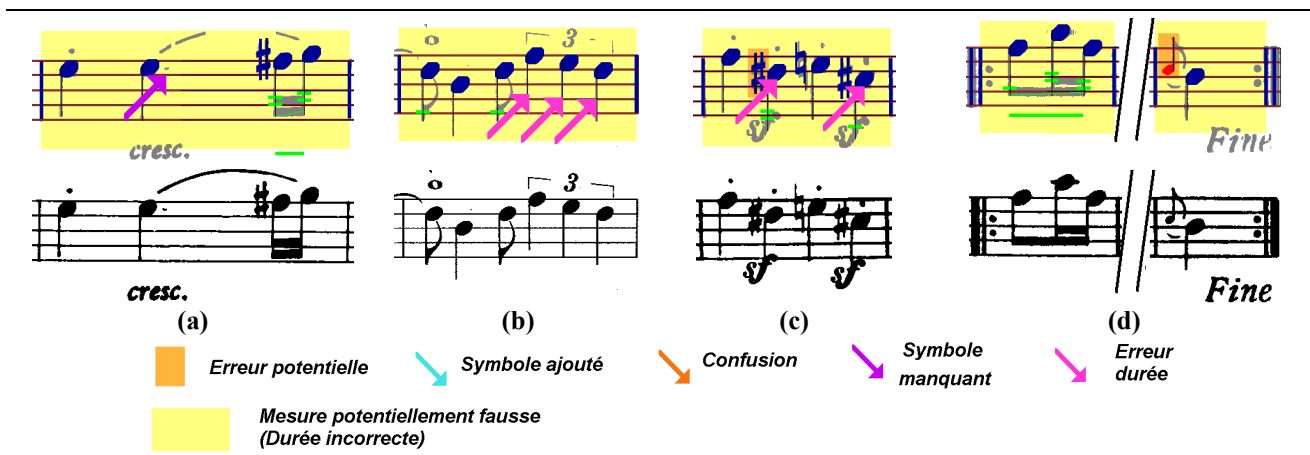


Figure 6.4 : Exemples d'indications de mesures erronées

La figure 6.4 montre quatre exemples : les trois premières indications sont pertinentes et permettent de localiser des erreurs de reconnaissance (point de durée manquant et erreurs de durée), tandis que les deux dernières sont de fausses alarmes, car il s'agit d'une reprise, les durées des deux mesures se complétant correctement.

La décomposition rythmique de chaque mesure est ensuite analysée. La durée minimale des groupes de notes nous permet de fixer un pas de découpage, égal à 1.0 ou 0.5 dans une métrique binaire, 1.5 ou 0.5 dans une métrique ternaire. Les groupes de notes sont ensuite associés aux silences voisins ou aux notes isolées voisines, de manière à ce que la durée totale de chacune des associations soit égale à un multiple du pas. Celles qui ne satisfont pas à ce découpage idéal sont indiquées comme fausses. Tous les autres groupes, dont le nombre total de temps semble correct, mais dont la répartition des durées est inhabituelle, compte tenu de la signature temporelle, sont également pointés comme potentiellement erronés. Par exemple, un groupe "croche pointée / double croche / croche" est parfaitement admissible dans une métrique ternaire, mais indiqué comme potentiellement erroné dans une métrique binaire avec un pas de 0.5.

La figure 6.5 montre quelques exemples d'indications d'erreurs dans une métrique binaire. Dans le premier exemple (a), le pas de découpage de la dernière mesure est égal à 1 temps. Les deux premiers groupes de notes ont une durée de 1.5 temps (3 croches), et sont donc indiqués comme potentiellement faux. C'est correct car il s'agit en fait de triolets (le quatrième temps est dans la première mesure qui doit être reprise). Dans le deuxième exemple (b), le pas de découpage est de 0.5 temps. Le groupe de 3 croches (1.5 temps) est marqué erroné, bien que sa durée soit 3 fois celle du pas, car ce regroupement est inhabituel dans une métrique binaire à 4 temps par mesure (4/4). Il est permis dans une métrique ternaire ou une métrique 3/4, comme dans le dernier exemple (c).

(a) Signature temporelle : 4/4

(b) : 4/4

(c) : 3/4

Durée des notes

| | | | | | | |
|-----|-----|-----|-----|-----|-----|------|
| | | | | | | |
| 1/2 | 1/3 | 1/4 | 1/5 | 1/6 | 1/8 | 1/16 |

Mesure potentiellement fausse (Durée incorrecte)

Erreur potentielle

Groupe cohérent

Groupe potentiellement faux (Erreur de durée)

Figure 6.5 : Exemples d'indications de durées erronées dans une métrique binaire

La figure 6.6 donne des exemples dans une métrique ternaire. La durée de la mesure (a) est globalement bonne car des erreurs de durée se compensent. Elles sont facilement repérées par les indications de groupe erroné. Dans le second exemple, les trois erreurs de durée (triolet non reconnu) sont aussi détectées (groupe potentiellement faux et durée inexacte de la mesure).

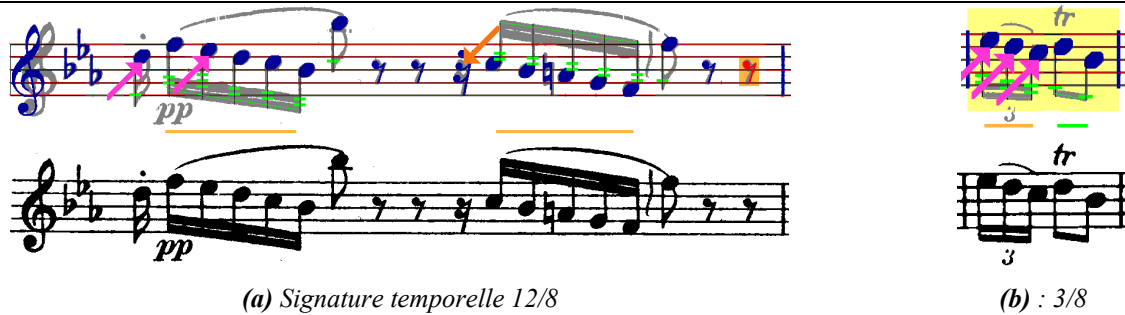


Figure 6.6 : Exemples d'indications de durées erronées dans une métrique ternaire

Les fausses alarmes sont négligeables, et correspondent à une mauvaise détection d'un groupe de notes, ou à la non-modélisation d'un rythme. Les non-détections proviennent de groupes de notes présentant des erreurs mais malgré tout rythmiquement cohérents. Dans le premier exemple de la figure 6.7, le triolet n'est pas reconnu, mais le groupe formé d'un quart de soupir et de 5 doubles croches semble correct dans une métrique ternaire. De même, le groupe de 3 croches du deuxième exemple est parfaitement valide, et les erreurs ne peuvent être détectées.

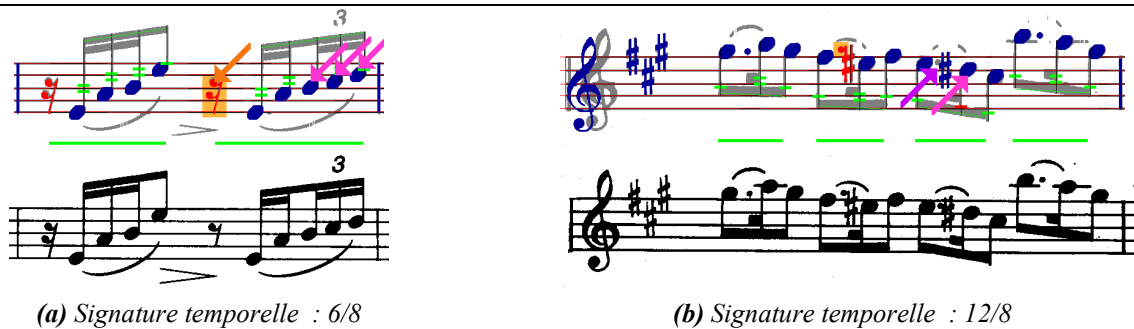


Figure 6.7 : Exemples de non-détection d'erreurs de durée

Notons que les critères rythmiques utilisés s'appliquent à l'écriture classique uniquement, contrairement aux critères exposés dans les paragraphes précédents, qui sont généraux.

6.1.4. Conclusion

Différents résultats, obtenus essentiellement lors de l'étape de modélisation floue, sont donc ré-exploités, afin de détecter les erreurs potentielles : les degrés de possibilité d'appartenance aux classes pour les symboles retenus par l'algorithme de décision, les coefficients de compatibilité graphique pour les symboles détectés mais rejetés lors de la décision, les groupes de notes et la modélisation des groupements rythmiques usuels. Cette démarche est très novatrice par rapport à la bibliographie, aucune méthode proposée n'introduisant des critères autres que la vérification de la durée des mesures, ou, en musique polyphonique, de l'alignement vertical des différentes voix [Coüasnon, Rétif 95] [Ferrand et al. 99] [Droettboom et al. 02]. Différents exemples ont illustré l'intérêt de la méthodologie, qui permet de repérer très rapidement des erreurs de reconnaissance, directement, ou indirectement. Dans certains cas en effet, une indication pertinente conduit à vérifier le reste de la mesure, et à trouver d'autres erreurs, qui sont corrélées à la première : dans la figure 6.3(b) par exemple, l'absence du point permet de remarquer immédiatement l'erreur de durée

commise sur la note suivante. Les figures 6.3(c), 6.5(b) et 6.6(a) illustrent également ce propos. Une évaluation plus précise des résultats sera donnée dans le chapitre 7 (paragraphe 7.6).

6.2. Adaptation à la partition analysée

La variabilité des typographies est un facteur important d'ambiguïté. Deux axes ont été prévus pour pallier cette difficulté : d'une part, deux modèles génériques sont définis pour les classes qui présentent la plus forte variabilité, de manière à mieux couvrir les différentes éditions (Figure 4.1 et paragraphe 4.4) ; d'autre part, la modélisation floue permet d'adapter le modèle de chaque classe à la partition traitée (paragraphe 5.2). Ces procédures ne nécessitent aucune intervention de l'utilisateur.

Cependant, elles ne peuvent pas fonctionner correctement lorsque les modèles génériques sont trop différents des symboles de la partition, et il est alors nécessaire de réaliser un apprentissage. D'autre part, même si les taux de reconnaissance sont globalement satisfaisants, un apprentissage des modèles peut néanmoins améliorer les résultats et constituer un gain de temps appréciable lorsque de grands volumes sont à traiter. Notons enfin que cette démarche permet de constituer des jeux de modèles, qui peuvent être mémorisés dans une base de données, et réutilisés en fonction de l'édition de la partition.

La procédure d'apprentissage est simple. L'utilisateur sélectionne quelques portées, représentatives des symboles musicaux, et corrige manuellement les erreurs faites par le programme de reconnaissance. Cette étape permet de définir un ensemble de prototypes, dont on connaît la classe et les coordonnées dans l'image. L'objectif de l'apprentissage est d'en déduire de nouveaux modèles de classe, plus conformes à l'édition traitée, et d'ajuster les paramètres du programme liés à ces modèles.

6.2.1. Apprentissage des modèles de classe

La procédure d'apprentissage prend en paramètres d'entrée l'image de la partition, et la liste des prototypes (classe et position dans l'image). Le programme de reconnaissance a également déterminé les modèles génériques (Figure 4.1) les mieux adaptés. Cette information est également passée à la procédure d'apprentissage.

Les portées sont tout d'abord redressées et les lignes supprimées (paragraphe 3.2.1). Les symboles corrigés par l'utilisateur sont corrélés avec le modèle générique de classe correspondant, autour de la position indiquée, afin de rechercher la position exacte du maximum de corrélation. Cette position est déjà connue pour tous les autres symboles, qui avaient été correctement classés par le programme de reconnaissance. Chaque couple de coordonnées permet d'extraire de l'image de la partition une sous-image contenant un prototype. Ces petites images sont ensuite moyennées sur chaque classe, et binarisées, avec un seuil égal à 0.5. On obtient ainsi un ensemble de modèles, notés M_a^k , représentatifs de chaque classe k .

Quelques précautions doivent cependant être prises sur certaines classes pour obtenir le résultat escompté.

Tout d'abord, les modèles de têtes de note, noires ou blanches, ne doivent pas inclure des portions de hampe. Pour cela, on applique une symétrie centrale sur chaque image extraite de la partition, et les deux images, l'image initiale et l'image symétrique, participent toutes les deux à la moyenne. Ainsi, on obtient des modèles de classe parfaitement symétriques, ne représentant que la tête de note, que les hampes des prototypes soient dirigées vers le haut ou vers le bas, et quelle que soit la proportion des deux cas. Le principe est aussi appliqué aux rondes, pour le respect de la symétrie du modèle uniquement.

D'autre part, les symboles creux (bémols, blanches, rondes) sont souvent détériorés par la procédure d'effacement des portées, lorsqu'ils sont situés dans un interligne. Pour ces classes, deux images moyennes sont calculées, la première sur les symboles centrés sur une ligne de portée, et la seconde sur les symboles placés dans un interligne. Ces deux images sont binarisées, puis recombinaées, pour former le modèle final : la première image définit tous les pixels susceptibles d'être effacés, tandis que tous les autres sont obtenus par un ET logique entre les deux images.

A priori tous les prototypes d'une même classe se ressemblent, puisqu'ils sont extraits d'une même partition, et les moyennés a donc bien un sens. Il serait cependant judicieux d'ajouter un critère validant chaque prototype, avant de l'intégrer dans la moyenne. Cela ne s'est pas avéré nécessaire dans nos expérimentations, mais pourrait être réalisé pour plus de fiabilité.

La figure 6.8 illustre la méthode d'apprentissage. La partition à reconnaître comprend 16 pages de musique, soient 177 portées. Le taux de reconnaissance global est bon, mais deux classes, les dièses et les quarts de soupir, ne sont pas très bien reconnues (taux inférieurs à 95%), ce qui justifie un apprentissage. La figure 6.8b indique les 11 portées utilisées pour l'apprentissage, extraites de trois pages, choisies pour contenir les différentes classes de symboles. La figure 6.8c montre les résultats de reconnaissance initiaux. Ces résultats ont été vérifiés par l'utilisateur et corrigés, afin de constituer la liste des prototypes. Celle-ci a été passée en paramètre de la procédure d'apprentissage, qui en a déduit les modèles de classe indiqués dans la figure 6.8a.

6.2.2. Apprentissage des paramètres

La seconde phase de l'apprentissage consiste à ajuster les paramètres liés aux modèles de classe. Il s'agit donc des seuils de décision $t_d(k)$ qui interviennent dans la génération d'hypothèses (Tableau 4.3), et dans la définition des distributions de possibilité d'appartenance aux classes (Eq. 5.2). Les nouveaux modèles M_a^k sont corrélés avec les images d'apprentissage, sans portée. Notons $C_k(s_n)$ le score de corrélation entre le modèle M_a^k et le $n^{\text{ième}}$ prototype ($0 \leq n < N_k$) de la classe k . Comme les symboles présentent toujours une variabilité dans la partition, on observe, sur chaque classe k , des variations du score de corrélation autour de la valeur moyenne C_k^m , définie par :

$$C_k^m = \frac{1}{N_k} \sum_{n=0}^{N_k-1} C_k(s_n) \quad (\text{Eq. 6.1})$$



(a) Modèles de classe déduits de l'apprentissage


(b) Portées utilisées pour l'apprentissage

Durée des notes

 Mesure potentiellement fausse (Durée incorrecte)

 Erreur potentielle

 Groupe cohérent

 Groupe potentiellement faux (Erreur de durée)

(c) Résultats de classification : ces résultats sont corrigés par l'utilisateur pour réaliser l'apprentissage.

Figure 6.8 : Exemple d'apprentissage de modèles de classe.

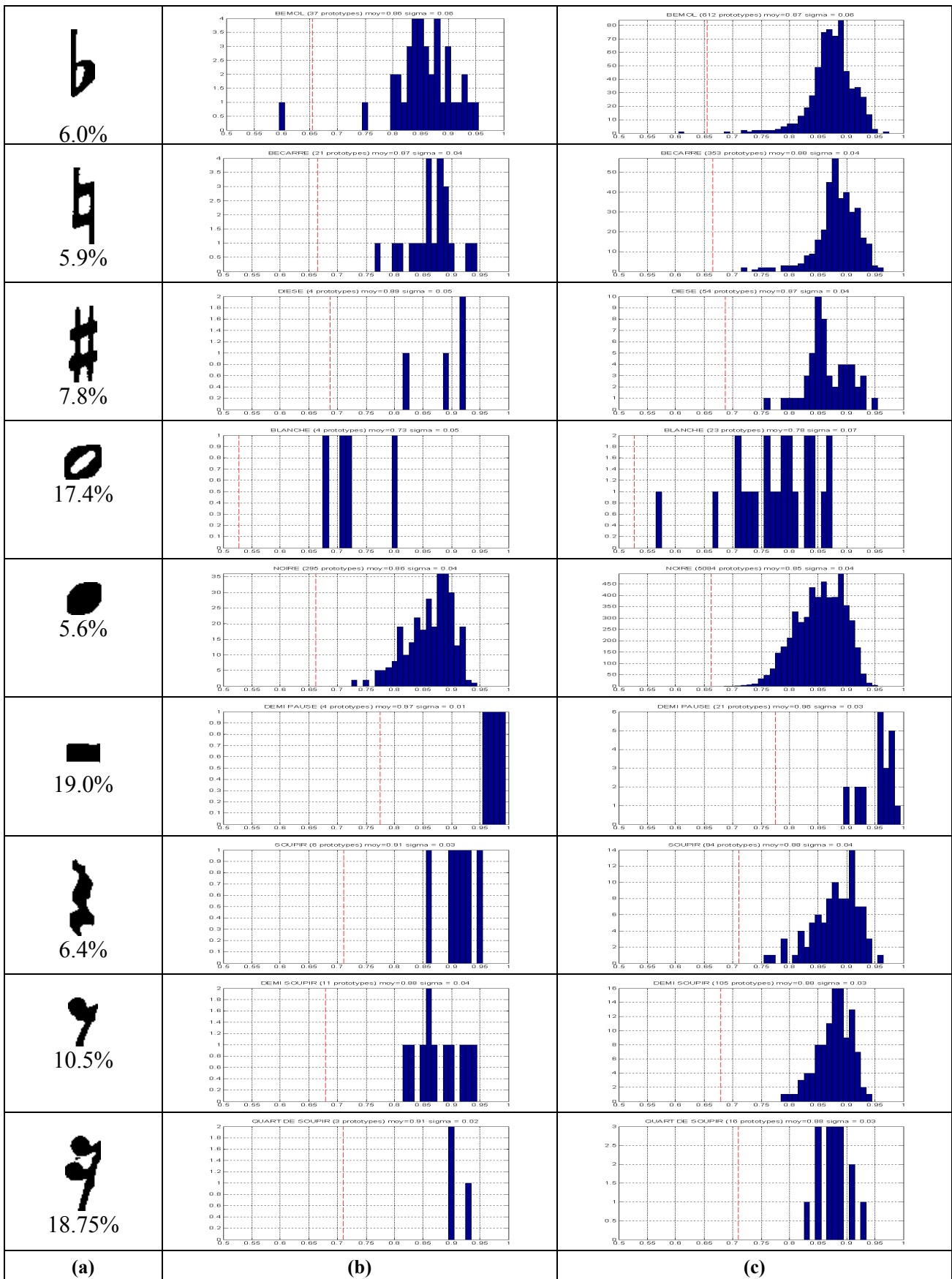


Figure 6.9: Exemple d'apprentissage de symboles : (a) Exemples de modèles appris M_a^k et proportion de prototypes extraits de la partition pour l'apprentissage ; (b) Histogrammes des scores de corrélation entre les modèles M_a^k et les prototypes d'apprentissage ; en pointillés rouges, les seuils de décision ; (c) Histogrammes calculés sur toute la partition.

Le seuil de décision $t_d(k)$ est ensuite calculé en fonction du paramètre D (paragraphe 5.2), qui représente l'écart maximal typique que l'on peut observer entre les scores de corrélation obtenus par des symboles de même classe, dans une même partition :

$$t_d(k) = C_k^m - D / 2 \quad (\text{Eq. 6.2})$$

La méthode de reconnaissance en généralisation est inchangée. Elle utilise simplement les nouveaux modèles de classe, avec les nouveaux seuils de décision $t_d(k)$. En particulier, les distributions de possibilité d'appartenance aux classes sont toujours apprises sur chaque page de musique analysée (Eq. 5.2), afin d'ajuster le paramètre S_k . Il ne s'agit en effet que d'un ajustement, puisque S_k , en l'absence de symboles classés en classe k en hypothèse H1 ($n(k)=0$ dans l'équation 5.2), prend la valeur moyenne C_k^m . Aucun des autres paramètres du programme ne dépend des modèles de classe, et ils ne sont donc pas modifiés.

L'apprentissage conduit à de bons résultats si le nombre de prototypes appris par classe est suffisant. Les expérimentations ont montré que 5 prototypes, en comptant les symétriques pour les classes "noire", "blanche" et "ronde", sont suffisants. Mais les résultats sont bien sûr d'autant plus fiables et précis que la base d'apprentissage est importante. La figure 6.9 illustre la méthode appliquée à la partition de la figure 6.8. La comparaison des scores de corrélation, obtenus sur la base d'apprentissage (colonne de gauche) et sur toute la partition (colonne de droite), prouve que les modèles appris sont effectivement représentatifs. On observe toujours une variabilité des scores de corrélation, ce qui montre que la modélisation floue des classes est, en dépit de l'apprentissage de nouveaux modèles, toujours pertinente.

6.2.3. Conclusion

La variabilité importante des polices de symboles est une difficulté majeure, identifiée dans de nombreux articles (e.g. [Fujinaga 88] [Bainbridge, Bell 96]). Bien que les systèmes présentés soient pour la plupart conçus pour être les plus généraux possibles, on peut affirmer qu'une source importante d'erreurs est due à cette caractéristique de l'édition musicale, et qu'il est nécessaire de proposer des procédures d'apprentissage des modèles de classe. Très peu de solutions ont cependant été proposées dans la littérature. Un seul auteur, à notre connaissance, traite réellement de ce problème : Fujinaga, dont le système, fondé sur l'extraction de caractéristiques et la décision par le plus proche voisin, peut apprendre de nouveaux prototypes et ajuster sa règle de décision par un algorithme génétique [Fujinaga 97]. Nous proposons une autre méthode, consistant à apprendre des modèles de classe utilisés pour une analyse par corrélation, ainsi que les paramètres liés à ces modèles. Cet apprentissage est spécifique à une partition donnée, mais on peut supposer qu'il peut être réutilisé pour d'autres partitions provenant de la même édition.

Grâce à l'apprentissage, le système de reconnaissance gagne en robustesse à deux niveaux :

- Il y a une diminution de l'ambiguïté des scores de corrélation, puisque les modèles de classe sont plus ressemblants aux symboles de la partition.
- La modélisation floue des classes de symboles est plus fine, car les seuils de décision $t_d(k)$ appris permettent d'ores et déjà de définir des distributions de possibilité d'appartenance aux

classes adaptées à la partition, ces distributions étant ensuite affinées grâce aux résultats produits par l'analyse des symboles de toute la partition.

L'apprentissage nécessite une intervention limitée de l'utilisateur. Dans les expérimentations, et pour des raisons pratiques de programmation, des portées entières ont été sélectionnées pour qu'elles incluent suffisamment de symboles de chaque classe en un nombre minimal de portées. Cette tâche n'est pas nécessaire : avec une interface graphique, il suffit que l'utilisateur pointe des symboles, jusqu'à ce que le nombre de prototypes par classe soit suffisant. L'apprentissage étant ensuite complètement automatique, on peut donc affirmer que la procédure est simple et rapide à réaliser. Un gain substantiel, en termes de taux de reconnaissance, a été obtenu dans les expérimentations réalisées. Des résultats précis seront présentés dans le chapitre 7 (paragraphe 7.7).

6.3. Conclusion

Nous avons proposé dans ce chapitre deux axes d'amélioration d'un système d'OMR : l'indication automatique d'erreurs potentielles et l'apprentissage supervisé d'une partition donnée, permettant de gagner en robustesse et en facilité d'utilisation. Ces voies ont été peu explorées jusqu'à présent, bien qu'on puisse affirmer qu'elles sont essentielles : Lutz, dans le cadre de la création d'une large base de données musicales [Lutz 04], rapporte qu'il faut à des musiciens expérimentés environ 1/4 d'heure pour rééditer correctement une page de musique scannée et reconnue par le logiciel commercial PhotoScore [PhotoScore]. Cette expérience montre qu'il est indispensable d'améliorer la fiabilité du système d'OMR, en passant si nécessaire par des procédures d'apprentissage, et en facilitant la recherche des erreurs. Les propositions faites dans ce chapitre vont dans ce sens et sont donc très pertinentes.

Les modèles de classe appris peuvent être sauvegardés et réutilisés. La procédure d'apprentissage, couplée à la méthode de sélection automatique de modèles (paragraphe 4.4), permet donc de compléter et d'affiner le programme d'OMR, au fur et à mesure de son utilisation. On peut également imaginer que l'utilisateur extraie lui-même de la base de données les modèles appropriés, de manière plus ou moins assistée.

Enfin, il faut de nouveau souligner l'intérêt de la modélisation floue, dont les résultats sont largement repris pour l'indication des erreurs potentielles.

CHAPITRE 7

Résultats

L'objet de ce chapitre est d'évaluer les différentes étapes de la méthode, de manière objective, sur une large base d'images. Comme nous l'avons mentionné au premier chapitre, les systèmes d'OMR présentés dans la littérature sont très rarement évalués. Le cas échéant, l'évaluation est réalisée sur une base de données restreinte, qui ne permet pas de vérifier la généralité de la méthodologie, en particulier de ses différents paramètres [Blostein, Baird 92]. Notons également qu'il n'existe pas de base d'images de référence, ni de méthode standard d'évaluation d'un logiciel d'OMR. Il a donc fallu constituer cette base, et définir des critères d'évaluation.

Une large base de données a été constituée, avec un grand souci de généralité (paragraphe 2.2), afin d'analyser les résultats obtenus en sortie de l'étape d'analyse individuelle des symboles, et de fournir des taux de reconnaissance. L'objectif est double : évaluer la fiabilité du système proposé, mais aussi analyser finement la méthode et repérer les sources d'erreurs. Des résultats de reconnaissance seront également comparés à ceux produits par un logiciel du commerce, Smartscore, sur quelques exemples [SmartScore 06]. Différentes statistiques seront ensuite données sur l'indication des erreurs potentielles. Enfin, l'apport de l'apprentissage sera illustré sur trois cas.

7.1. Conditions d'expérimentation et données en sortie du système

7.1.1. Conditions d'expérimentation

La base de test contient plus d'une centaine de partitions, qui représentent 1191 portées et plus de 48000 symboles à reconnaître. Rappelons que le système ne reconnaît pour l'instant ni la clé, ni la tonalité, ni la signature temporelle. Ces indications sont donc fournies par l'utilisateur. Le programme est lancé page par page, même si certaines sont extraites de la même partition, et consécutives. Il n'y a dans les images testées aucun changement de clé, de tonalité ou de métrique. C'est une restriction importante qu'il faudra lever par la suite. Le système décrit dans les chapitres précédents est exécuté sur toute la base, sans aucun ajustement de paramètres, ni intervention de l'utilisateur. Notons enfin que les symboles qui ont servi à la mise au point de la méthode représentent une faible proportion de cette base.

7.1.2. Données en sortie du programme

Le programme fournit une image des symboles reconnus (superposés à l'image source), un fichier Midi qui permet d'entendre la mélodie, ainsi qu'un fichier texte qui décrit les résultats de reconnaissance : type de symbole, hauteur et durée, position dans l'image. Ce fichier texte correspond à la représentation symbolique de l'image. Comparé au fichier corrigé (reconnaissance parfaite), il permet de calculer les statistiques qui seront indiquées par la suite.

La création d'un fichier Midi suppose de restituer l'interprétation de haut niveau, notamment la hauteur et la durée des notes, compte tenu de leurs attributs. L'analyse sémantique est très simple à réaliser à partir des informations extraites, car les relations structurelles et syntaxiques ont auparavant été établies :

- Les groupes de notes sont déjà construits et les durées calculées (paragraphe 4.2.6 et 5.4.3).
- l'attribution des points aux notes et aux silences est réalisée lors de la reconnaissance elle-même (paragraphe 4.2.6 et 4.3.4).
- Les altérations à la clé sont bien différenciées des altérations accidentelles lors de la modélisation floue (paragraphe 5.4.1 et 5.4.2).
- L'attribution des altérations accidentelles aux notes est très simple, puisque toutes les altérations qui ne sont pas correctement positionnées par rapport à une note sont éliminées par la modélisation floue (paragraphe 5.3.1). Cette information se propage très simplement sur le reste de la mesure, en musique monodique.

Comme, de plus, la clé et la tonalité sont données en paramètres d'entrée, et ne changent pas, la restitution de la sémantique ne présente aucune ambiguïté.

Il existe néanmoins des sources d'erreurs sur la hauteur des notes, dues à des imprécisions de localisation portant, soit sur les lignes de portée (paragraphe 3.1.3), soit sur les têtes de note (paragraphe 4.2.5). En ce qui concerne le premier cas, il faut noter que l'espace entre les lignes additionnelles au-dessus ou au-dessous de la portée peut varier très nettement, et que ce problème n'a pas encore été traité.

7.1.3. Méthode d'évaluation de la précision et de la fiabilité du système

D'après les remarques précédentes, on peut considérer qu'une évaluation au niveau symbolique, sur l'ensemble des classes à reconnaître (Figure 4.1), complétée d'une vérification de la durée et de la hauteur des notes, est suffisante pour estimer la fiabilité et la précision du système. On considèrera donc 5 types d'erreurs :

- symbole ajouté : symbole qui ne correspond à aucun objet de l'image devant être reconnu : par exemple un symbole confondu avec une lettre d'un texte, un point de durée dû à un bruit ou à un point de staccato, etc.
- symbole manquant : symbole qui aurait dû être reconnu, mais pour lequel aucune classe n'a été attribuée.
- confusion : symbole détecté mais mal reconnu : la classe qui lui a été attribuée n'est pas la bonne.

- durée de note erronée, due à une mauvaise interprétation des crochets ou des barres de groupe, à la non-détection d'un triolet, etc.
- hauteur de note erronée : cette erreur a deux causes possibles : soit la position de la tête de note par rapport à la portée n'est pas suffisamment précise, soit une erreur a été commise sur une altération précédant la note dans la mesure, ou sur la détection d'une barre de mesure. On ne considèrera que le premier cas, les autres étant redondants.

Soulignons que la correction de ces erreurs, réalisée de manière individuelle, est suffisante, puisqu'il n'y a pas d'ambiguïté à résoudre pour la restitution de la sémantique. Un fichier Midi généré après correction des cinq types d'erreurs mentionnés produit donc la mélodie exacte. Naturellement, cette remarque ne serait plus valable pour des systèmes reconnaissant les partitions polyphoniques ou davantage de symboles, comme les ornements, car il y aurait alors plus d'ambiguïté dans l'analyse sémantique. L'évaluation que nous proposons ne serait plus totalement représentative de la qualité de la musique reconstituée, ni de la charge de travail nécessaire aux corrections. Il faudrait alors compléter la méthode par une évaluation réalisée à un niveau d'abstraction plus élevé [Ng et al. 04].

7.2. Résultats sur l'analyse individuelle des symboles

L'objectif de ces premiers tests est d'évaluer la qualité des hypothèses de reconnaissance. En particulier, il s'agit de vérifier que les symboles sont bien détectés, et que l'ensemble des hypothèses de reconnaissance inclut effectivement les classes exactes.

7.2.1. Résultats et analyse

Le tableau 7.1 indique la répartition des hypothèses de reconnaissance, par classe, puis sur tous les symboles : par exemple, la colonne H1 indique dans quelle proportion la classe correcte est présente dans le niveau d'hypothèse H1, sans hypothèse H0 (score de corrélation supérieur au seuil de décision $t_d(k)$). La somme des quatre colonnes "H1", "H0+H1", "H2" et "H3" donne le pourcentage de symboles dont la classe est bien dans les hypothèses de reconnaissance.

Sur le total des symboles, 99.68% ont été correctement analysés. Cela signifie qu'au moins 0.32% des erreurs finales sont faites lors de la segmentation ou de l'analyse individuelle des symboles.

Pour les classes peu ambiguës (typiquement les noires, les barres de mesure), l'hypothèse correcte est située à plus de 99% dans le niveau H1, avec ou sans hypothèses H0, c'est-à-dire que le modèle de classe correspondant obtient le plus haut score de corrélation. En revanche, les classes qui présentent davantage de variabilité (typiquement les altérations, les appoggiatures, les blanches, les quarts et huitièmes de soupir) ont davantage d'hypothèses correctes dans les niveaux H2 ou H3. Ce tableau prouve donc qu'il est nécessaire de générer plusieurs hypothèses de reconnaissance par objet : au total, 0.91% des hypothèses correctes ne correspondent pas au score de corrélation maximal, 2.95% des symboles ne sont pas reconnus de manière certaine (dans la colonne "H1"). On













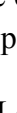


| Classe | HYPOTHESES CORRECTES (%) | | | | TOTAL (%) |
|---|--------------------------|--------------|-------------|-------------|--------------|
| | H1 | H1+H0 | H2 | H3 | |
|  | 99.49 | 0.06 | 0.03 | 0.06 | 99.64 |
|  | 96.35 | 02.39 | 0.90 | 0.06 | 99.70 |
|  | 89.85 | 07.79 | 01.46 | 0.32 | 99.43 |
|  | 45.07 | 39.20 | 5.87 | 1.88 | 92.02 |
|  | 89.84 | 05.88 | 3.39 | 0.62 | 99.73 |
|  | 99.79 | 0.14 | 0.04 | 0.00 | 99.97 |
|  | 80.49 | 0.70 | 14.46 | 2.70 | 98.34 |
|  | 77.42 | 20.74 | 0.92 | 0.00 | 99.08 |
|  | 83.17 | 15.01 | 0.27 | 0.00 | 98.45 |
|  | 57.14 | 0.00 | 42.86 | 0.00 | 100.00 |
|  | 72.00 | 10.67 | 15.33 | 0.00 | 98.00 |
|  | 90.90 | 6.48 | 1.50 | 0.00 | 98.88 |
|  | 93.90 | 4.78 | 0.84 | 0.00 | 99.52 |
|  | 98.20 | 1.80 | 0.00 | 0.00 | 100.00 |
|  | 71.72 | 26.21 | 0.69 | 0.00 | 98.62 |
| TOTAL | 96.73 | 02.04 | 0.78 | 0.13 | 99.68 |

Tableau 7.1 : Répartition des hypothèses de reconnaissance, par classe, et sur tous les symboles

constate cependant que le niveau d'hypothèse H3 semble inutile en ce qui concerne les symboles qui ne sont pas caractérisés par un segment vertical (silences, points, rondes). Il a donc été supprimé.

Le tableau 7.2 donne davantage de détails sur les erreurs. La première colonne indique le pourcentage de symboles qui ne sont pas dans les hypothèses de reconnaissance. Ces erreurs sont de deux sortes : soit le symbole n'est pas détecté, soit il a été détecté, mais sa classe n'a pas été retenue. Les colonnes suivantes, "symbole non détecté" et "confusion", indiquent la proportion des deux types d'erreurs. Enfin, les trois dernières colonnes évaluent le taux de symboles ajoutés, en distinguant deux cas : soit il s'agit d'un symbole qui n'a pas à être reconnu ("Ajout"), soit il s'agit d'une sur-détection (un symbole qui doit être reconnu et qui est détecté plusieurs fois).

Les cas d'hypothèses manquantes ont plusieurs origines. La plus courante est une très mauvaise impression ou une forte dégradation du document. La conséquence est, soit une segmentation fautive (Figure 7.1), soit un rejet de la bonne hypothèse car les critères de préclassification ne sont pas suffisamment satisfaits ou le score de corrélation est trop faible (Figure 7.2). On peut cependant noter que les cas de connexions parasites sont généralement très bien

| | Classe non présente dans les hypothèses de reconnaissance | | | Hypothèses ajoutées | | |
|-------|---|---------------------|-----------|---------------------|---------------|-------|
| | TOTAL | Symbole non détecté | Confusion | TOTAL | Sur-détection | Ajout |
| | 0.36 | 0.17 | 0.19 | 0.64 | 0.11 | 0.53 |
| b | 0.30 | 0.24 | 0.06 | 83.86 | 0.24 | 83.62 |
| q | 0.57 | 0.41 | 0.16 | 15.26 | 7.14 | 8.12 |
| ♪ | 7.98 | 1.64 | 6.34 | 95.77 | 0.00 | 95.77 |
| # | 0.27 | 0.16 | 0.12 | 6.38 | 4.55 | 1.83 |
| ● | 0.03 | 0.02 | 0.01 | 0.25 | 0.09 | 0.16 |
| o | 1.66 | 1.13 | 0.52 | 0.52 | 0.00 | 0.52 |
| ○ | 0.92 | 0.00 | 0.92 | 0.00 | 0.00 | 0.00 |
| . | 1.55 | 1.28 | 0.27 | 48.55 | 0.32 | 48.23 |
| y | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| y | 2.00 | 1.33 | 0.67 | 32.67 | 0.00 | 32.67 |
| γ | 1.12 | 0.75 | 0.37 | 9.98 | 0.00 | 9.98 |
| z | 0.48 | 0.12 | 0.36 | 37.46 | 21.05 | 35.41 |
| - | 0.00 | 0.00 | 0.00 | 37.39 | 0.00 | 37.39 |
| - | 1.38 | 0.69 | 0.69 | 55.86 | 0.00 | 55.86 |
| TOTAL | 0.32 | 0.17 | 0.15 | 8.20 | 0.88 | 7.32 |

Tableau 7.2 : Erreurs dans les hypothèses de reconnaissance

résolus. Au contraire, les effacements importants de pixels conduisent presque toujours à une erreur. Le rejet d'hypothèses exactes peut également être dû à l'inadéquation des modèles génériques de classe, combinée à une impression de qualité moyenne et/ou des imprécisions dans l'effacement des lignes de portée (Figure 7.3).

Enfin, certains choix qui ont été faits pour la segmentation et la préclassification, sont l'origine de quelques erreurs. La figure 7.4 résume les principaux cas. Ils sont néanmoins très marginaux par rapport aux précédentes sources d'erreurs, à l'exception du problème plus récurrent de la préclassification erronée de certaines blanches (d).

On constate dans le tableau 7.2 que de nombreux symboles sont ajoutés. Les sur-détections ont différentes origines : la plus fréquente concerne les dièses et les bécarrés, qui, présentant deux segments verticaux, peuvent être détectés deux fois (paragraphe 3.2.2, Figure 3.29). Les soupirs font également l'objet de plusieurs détections, puisqu'ils peuvent être analysés en tant que symbole caractérisé par un segment vertical (paragraphe 4.2), ou en tant que silence (paragraphe 4.3). Cette

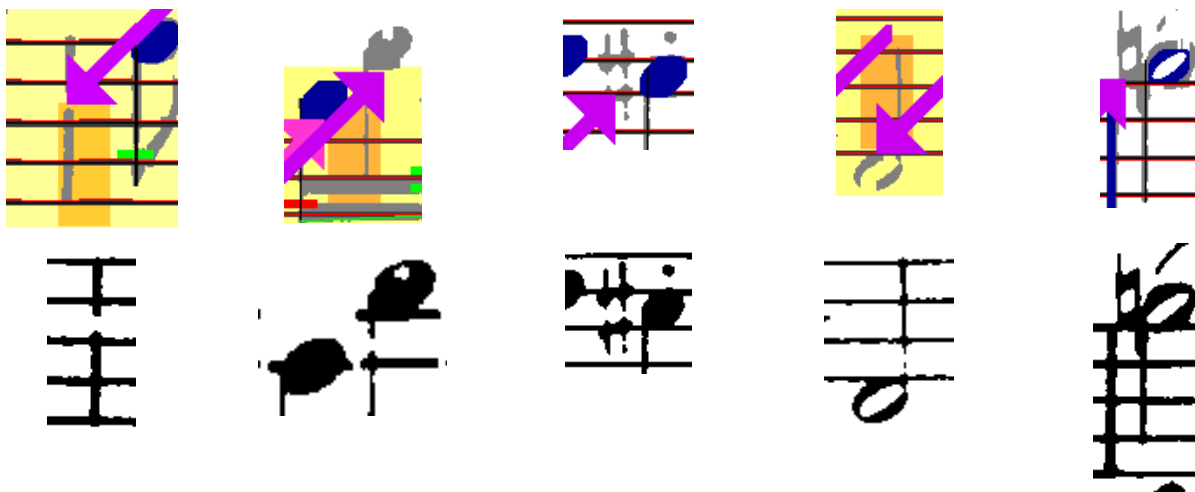


Figure 7.1 : Exemples de défauts graves de segmentation, dus à de fortes dégradations de l'image

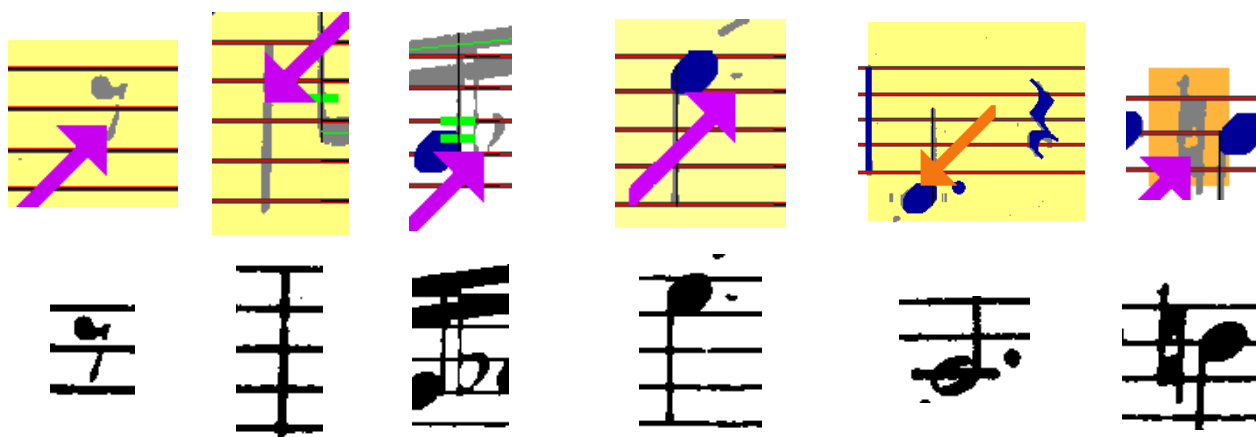


Figure 7.2 : Exemples de symboles dont la classe n'est pas présente dans les hypothèses, à cause de la mauvaise qualité du document original. (rejet de la classe en préclassification (Tableau 4.1) ou lors de la sélection d'hypothèses (Tableau 4.3)).

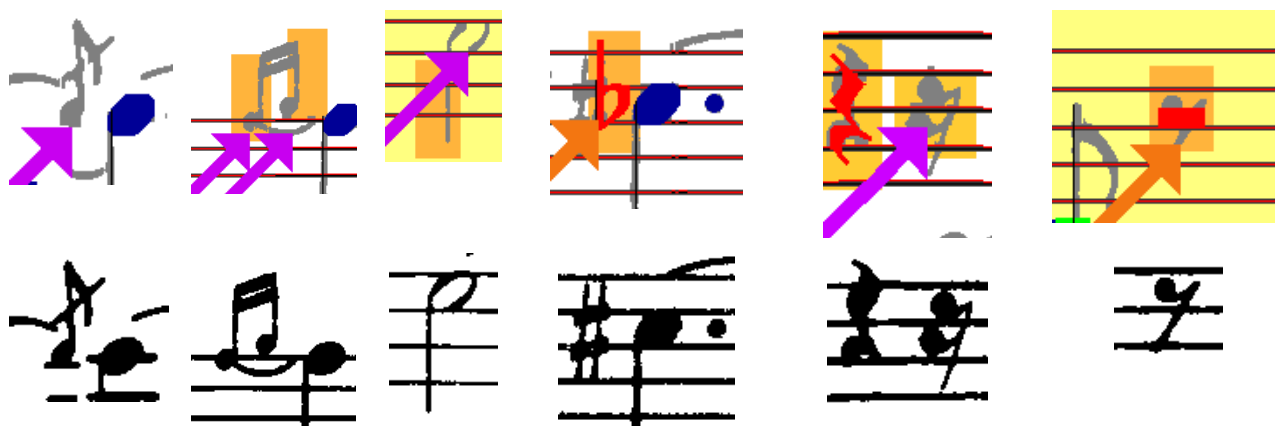


Figure 7.3 : Exemples de symboles inadaptés aux modèles de classe. Les scores de corrélation obtenus par les modèles de référence sont trop faibles, et ne passent pas les règles de sélection d'hypothèses (Tableau 4.3).

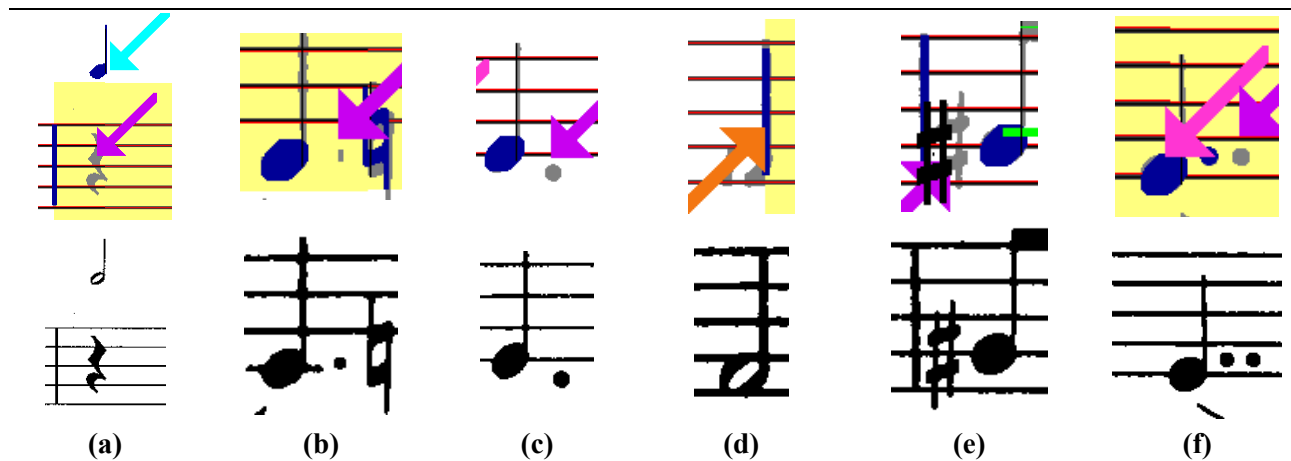


Figure 7.4 : Exemples d'erreurs liées à la méthodologie et au document

(a) La détection certaine (hypothèse H1 sans hypothèse H0) d'une appoggiature empêche la recherche d'un silence au-dessous (paragraphe 4.3.1); (b) L'effacement de tous les emports courts satisfaisant aux équations 3.25 et 3.26, conduit à effacer partiellement le point au-dessous de la portée, car il n'est pas à une position usuelle; (c) Le point n'est pas dans la zone de recherche définie par l'équation 4.9; (d) L'effacement des lignes de portée dégrade la tête de note blanche; en conséquence, la possibilité d'une note est rejetée en préclassification (Tableau 4.1). Comme par ailleurs une barre de mesure est détectée avec un score de corrélation suffisant, cette erreur n'est pas rattrapée (paragraphe 4.2.4); (e) Le dièse n'est pas détecté au bon endroit (résultat en noir). Cette erreur est due à 3 causes : la présence de la barre de mesure à cette distance, un modèle de classe insuffisamment ressemblant, la plage de corrélation assez large ($s/2$ dans la direction horizontale, tableau 4.2); (f) Un unique point est recherché après les têtes de note ou les silences, et le second point n'est pas détecté.

redondance devrait être simple à éliminer, car les soupirs sont généralement bien isolés des autres symboles. Des segments verticaux très épais conduisent également à des sur-détections. Cela concerne notamment les bémols et les notes.

Les diverses inscriptions qui ne correspondent pas à des symboles à reconnaître, en particulier les textes, conduisent à des ajouts de symboles, essentiellement de bémols et d'appoggiatures. Les ajouts de silences sont dus à des confusions avec des liaisons, des queues de note, ou d'autres signes sur la portée. En ce sens, une segmentation préalable des silences, par analyse de connexité, suivie d'une préclassification, réduirait considérablement le taux de ces hypothèses supplémentaires. Enfin, les points ajoutés proviennent de bruits, de fragments de lignes de portée additionnelles incomplètement effacées, ou encore des points de staccato. La figure 7.5 montre quelques exemples de ces hypothèses inutiles.

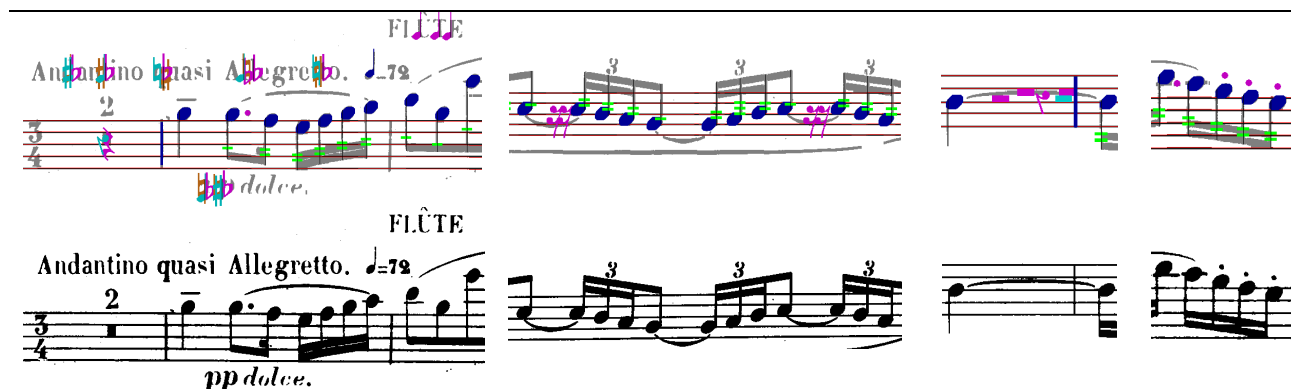


Figure 7.5 : Exemples d'hypothèses ajoutées

Le tableau 7.3 indique le nombre moyen d'hypothèses qui ont été générées par symbole

correctement analysé. Les taux évoluent entre 1.1, pour la classe "noire" qui présente le moins d'ambiguïté, et 2.4 pour la classe "appoggiature" qui présente le plus d'ambiguïté. Ils sont pour la plupart inférieurs à 2, ce qui tend à prouver que les critères utilisés pour la sélection d'hypothèses sont bien définis. Des seuils plus bas résoudraient quelques-uns des problèmes illustrés dans la figure 7.3, mais conduiraient à davantage d'hypothèses, donc également à davantage d'ambiguïté dans les étapes ultérieures. Les expérimentations ont montré que les choix qui ont été faits constituent le meilleur compromis.

| Classe | Taux | Classe | Taux |
|--------|------|--------------|-------------|
| | 1.00 | • | 1.17 |
| b | 1.38 | ♯ | 1.71 |
| q | 2.26 | 7 | 2.10 |
| ♪ | 2.40 | 7 | 1.95 |
| # | 2.04 | 7 | 1.92 |
| • | 1.10 | - | 1.46 |
| ◊ | 1.80 | - | 1.76 |
| ◊ | 1.36 | TOTAL | 1.19 |

Tableau 7.3 : Nombre moyen d'hypothèses par symbole bien détecté

7.2.2. Conclusion

Les prétraitements, la segmentation et l'analyse individuelle des symboles conduisent donc à de bons résultats, avec néanmoins 0.32% de symboles incorrects et 8.20% de symboles ajoutés. Les tests réalisés sur une large base de données tendent à prouver que les nombreux paramètres qui ont été définis ne sont pas restrictifs, mais qu'ils modélisent correctement la notation musicale (voir également les tests de robustesse décrits dans la section 7.3.4). L'axe principal d'amélioration consisterait à réduire le nombre de fausses détections. On peut, pour cela, envisager une détection préalable des textes [Fletcher, Kasturi 88], ainsi qu'une segmentation et une préclassification des symboles qui ne sont pas caractérisés par un segment vertical. Il est à noter que la méthode de segmentation des autres symboles est performante. Elle permet généralement de surmonter le problème des connexions parasites. En revanche, elle échoue en cas de dégradations trop importantes des segments (pixels noirs effacés), et il faudrait, pour les partitions présentant ces défauts, envisager des techniques de restauration. Il serait également intéressant d'étudier plus en détail les différentes polices, afin d'optimiser les modèles génériques de classe et d'en proposer éventuellement davantage. L'ambiguïté serait alors probablement réduite.

7.3. Taux de reconnaissance

Dans ce paragraphe, nous indiquons différentes statistiques, calculées sur les résultats obtenus en sortie du programme, permettant de mesurer la fiabilité du système, et d'évaluer l'apport

de la modélisation floue.

7.3.1. Evaluation du système et analyse des résultats

Le tableau 7.4 donne les résultats de reconnaissance par classe et par type d'erreur (E), en distinguant trois types d'erreurs, les symboles manquants ($(E)=(M)$), les confusions ($(E)=(C)$), et les ajouts de symboles ($(E)=(A)$) :

$$r_k^{(E)}(k) = \frac{\text{Nombre de symboles erronés de la classe } k}{\text{Nombre total de symboles de la classe } k} * 100 \quad (\text{Eq. 7.1})$$

La fiabilité du système peut se mesurer par des taux de reconnaissance calculés sur chaque classe (Equation 7.2), complétés des taux de symboles ajoutés $r_k^{(A)}(k)$ (Equation 7.1). Ces résultats sont dans les colonnes grisées du tableau 7.4.

$$\tau_k(k) = 100 - (r_k^{(M)}(k) + r_k^{(C)}(k)) \quad (\text{Eq. 7.2})$$

Le tableau 7.5 donne les mêmes informations, mais rapportées cette fois au nombre total de symboles (Equations 7.3 et 7.4). Cette présentation est également intéressante car les différentes classes ont des fréquences d'occurrence très différentes.

$$r^{(E)}(k) = \frac{\text{Nombre de symboles erronés de la classe } k}{\text{Nombre total de symboles}} * 100 \quad (\text{Eq. 7.3})$$

$$\tau(k) = 100 - (r^{(M)}(k) + r^{(C)}(k)) \quad (\text{Eq. 7.4})$$

Le taux de reconnaissance global des symboles, sur toute la base, est égal à 99.20%, les erreurs provenant des confusions (0.2%) et des symboles manquants (0.6%). Il y a également 0.30% de symboles ajoutés. La figure 7.6 indique l'histogramme des taux de reconnaissance obtenus par page de musique. Un tiers d'entre elles obtiennent un taux supérieur à 99.75%, et tous les taux sont supérieurs à 91%. Tous ces résultats prouvent la robustesse de la méthode. D'après le paragraphe précédent, 0.32% d'erreurs proviennent de la segmentation ou de l'analyse individuelle des symboles ; 0.48% d'erreurs proviennent donc de l'étape de décision, mais il faut souligner que ces nouvelles erreurs peuvent être la conséquence des premières, puisque les symboles sont corrélés dans la mesure, notamment liés par la métrique.

Commentons plus en détail les résultats. Le tableau 7.4 montre que les différentes classes de symboles obtiennent un taux de reconnaissance supérieur à 90%, les appoggiatures et les huitièmes de soupir exceptés. Les appoggiatures sont difficiles à reconnaître, à cause de leur forte variabilité (Figure 7.3), et également, de la grande imprécision sur leur localisation : on constate en effet que le modèle graphique flou proposé permet de résoudre l'ambiguïté entre les appoggiatures et les altérations accidentelles (Tableau 7.6), mais qu'il rejette de nombreuses appoggiatures, lorsque l'espacement avec la note dans la direction horizontale est important. Les confusions sont généralement faites entre des symboles de même durée (entre altérations et appoggiatures, entre soupir et noire (Figure 7.7a), etc.) et/ou présentant une forte inter-corrélation (bécarre et dièse (b),

| Classe | $r_k^{(M)}(k)$ | $r_k^{(C)}(k)$ | | $\tau_k(k)$ | $r_k^{(A)}(k)$ |
|--------|----------------|----------------|-----|---------------|----------------|
| | 0.34 | 0.06 | ● | 99.60 | 0.33 |
| b | 0.72 | 0.36 | q | 98.92 | 0.18 |
| q | 1.10 | 1.05 | # | 97.65 | 0.08 |
| ♪ | 31.92 | 3.52 | b ● | 64.55 | 8.68 |
| # | 0.43 | 0.62 | q | 98.95 | 0.00 |
| ● | 0.03 | 0.02 | ♪ ~ | 99.95 | 0.05 |
| o | 0.52 | 1.48 | ● | 98.00 | 0.52 |
| o | 1.84 | 3.33 | | 97.70 | 0.00 |
| . | 2.36 | 0.16 | | 97.48 | 0.70 |
| ~ | 0.00 | 14.29 | q | 85.71 | 0.00 |
| ~ | 6.00 | 3.33 | q | 90.67 | 0.00 |
| q | 1.75 | 0.873 | q - | 97.38 | 0.75 |
| ~ | - | - | | - | 0.48 |
| ~ | 0.48 | 0.12 | ● | 99.40 | 3.11 |
| - | 0.00 | 0.00 | | 100.00 | 4.05 |
| - | 1.38 | 0.00 | | 98.62 | 3.45 |

Tableau 7.4 : Résultats de reconnaissance sur chaque classe : pourcentages de symboles manquants $r_k^{(M)}(k)$, pourcentages de confusions $r_k^{(C)}(k)$ et indications sur les principales confusions, taux de reconnaissance $\tau_k(k)$, pourcentages de symboles ajoutés $r_k^{(A)}(k)$

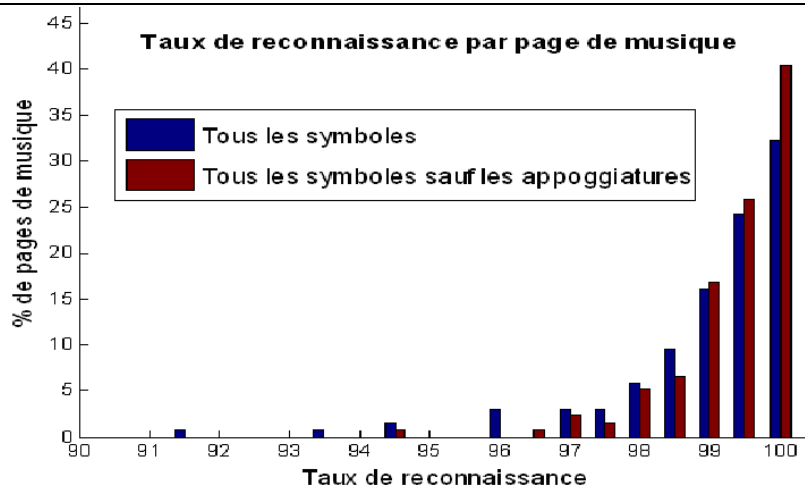


Figure 7.6 : Histogramme des taux de reconnaissance, obtenu sur la base de données comprenant une centaine de pages de musique

| Classe | $r^{(M)}(k)$ | $r^{(C)}(k)$ | $\tau(k)$ | $r^{(A)}(k)$ |
|--------------|--------------|--------------|--------------|--------------|
| | 0.05 | 0.01 | 13.25 | 0.04 |
| b | 0.02 | 0.01 | 3.43 | 0.01 |
| q | 0.03 | 0.03 | 2.49 | $<10^{-2}$ |
| ♪ | 0.28 | 0.03 | 0.57 | 0.08 |
| # | 0.02 | 0.03 | 5.27 | 0.00 |
| ● | 0.02 | 0.01 | 63.24 | 0.03 |
| o | 0.01 | 0.04 | 2.33 | 0.01 |
| ○ | 0.01 | $<10^{-2}$ | 0.44 | 0.00 |
| . | 0.09 | 0.01 | 3.77 | 0.03 |
| · | 0.00 | $<10^{-2}$ | 0.01 | 0.00 |
| 7 | 0.02 | 0.01 | 0.28 | 0.00 |
| 9 | 0.03 | 0.01 | 1.62 | 0.01 |
| 7 | 0.00 | 0.00 | 0.00 | $<10^{-2}$ |
| 7 | 0.01 | $<10^{-2}$ | 1.72 | 0.05 |
| ■ | 0.00 | 0.00 | 0.46 | 0.02 |
| ■ | $<10^{-2}$ | 0.00 | 0.30 | 0.01 |
| TOTAL | 0.60 | 0.20 | 99.20 | 0.30 |

Tableau 7.5 : Résultats de reconnaissance rapportés à tous les symboles : pourcentages de symboles manquants $r^{(M)}(k)$, pourcentages de confusions $r^{(C)}(k)$, taux de reconnaissance $\tau(k)$, pourcentages de symboles ajoutés $r^{(A)}(k)$

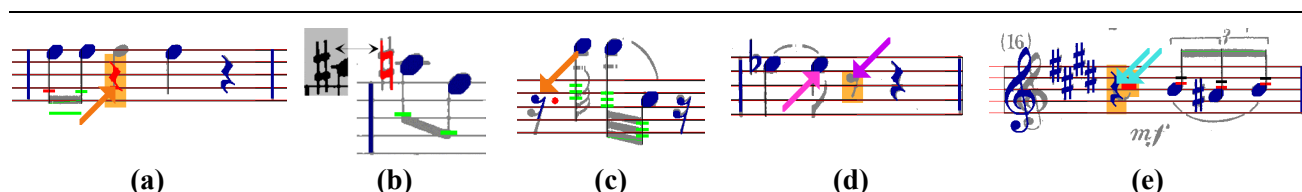


Figure 7.7 : Exemples d'erreurs

demi-soupir et quart de soupir (c), noire et blanche, etc.). Les symboles manquants proviennent des non-détections, ou encore du non-choix d'une hypothèse exacte, compensant une autre erreur, par exemple une erreur de durée (d). Il faut en effet noter que de nombreuses erreurs se compensent d'un point de vue temporel, à cause de la règle stricte concernant le nombre de temps par mesure, appliquée en priorité dans l'étape de décision (paragraphe 5.5.2).

Les taux de reconnaissance des altérations sont satisfaisants, grâce à la modélisation floue des règles graphiques et syntaxiques. Le tableau 7.6 présente plus en détail les résultats obtenus sur

les altérations accidentelles (sans les altérations à la clé) et les appoggiatures, par une matrice de confusion (Equation 7.5) : les taux obtenus pour les bémols, bécarres et dièses sont tous supérieurs à 97%, et ces résultats semblent nettement meilleurs que ceux présentés dans la littérature (e.g. [Bainbridge, Wijaya 99]). Les confusions les plus fréquentes sont entre bécarre et dièse, ou appoggiature et bémol. En effet, les règles graphiques ne départagent pas les hypothèses lorsqu'elles sont parfaitement superposées, et les règles syntaxiques ne permettent pas toujours de lever l'ambiguïté (rappel d'altération).

$$C(i,j) = \frac{\text{Nombre de symboles de la classe } i \text{ classés en classe } j}{\text{Nombre de symboles de la classe } i} * 100 \quad (\text{Eq. 7.5})$$

| | b | q | # | ♪ | Autres | Manquants | Ajoutés |
|----------|--------------|--------------|--------------|--------------|--------|-----------|---------|
| b | 97.06 | 0.82 | 0.00 | 0.16 | 0.00 | 1.96 | 0.49 |
| q | 0.00 | 97.65 | 0.97 | 0.08 | 0.00 | 1.30 | 0.08 |
| # | 0.06 | 0.68 | 98.33 | 0.00 | 0.25 | 0.68 | 0.00 |
| ♪ | 2.35 | 0.00 | 0.00 | 64.56 | 1.17 | 31.92 | 8.68 |

Tableau 7.6 : Matrice de confusion des altérations accidentelles.

On constate enfin, en comparant les tableaux 7.2 et 7.4, que la plupart des hypothèses ajoutées sont correctement éliminées. Beaucoup d'erreurs concernent les appoggiatures (tableaux 7.4 et 7.5), de nouveau à cause de la difficulté de modélisation de leur position. Les autres ajouts sont généralement présents dans les anacrouses ou les mesures de reprise : le nombre total de temps de la mesure étant inférieur à celui attendu, l'algorithme tend à ajouter des symboles, souvent des silences, parfois des notes (Figure 7.7e). En revanche, le nombre d'altérations ajoutées est négligeable, la modélisation floue écartant toutes les hypothèses incohérentes.

7.3.2. Hauteur et durée des notes

Une note est correctement interprétée lorsque sa hauteur et sa durée sont exactes. Le tableau 7.7 résume les taux de reconnaissance obtenus, par classe et sur le total des notes, en ne comptabilisant que les erreurs directes : mauvais positionnement de la note par rapport à la portée, et mauvaise interprétation des crochets, des barres de groupe ou des n-olets. Les erreurs indirectes, dues par exemple à l'absence d'un point de durée, ou à la classification erronée d'une altération, ne sont pas prises en compte, car elles ont déjà été comptabilisées.




| | Hauteur | Durée |
|---|--------------|--------------|
|  | 99.00 | 99.28 |
|  | 98.52 | 100.00 |
|  | 98.16 | 100.00 |
| TOTAL | 98.98 | 99.31 |

Tableau 7.7 : Interprétation de la hauteur et de la durée des notes (pourcentages)

La précision de la détection des lignes de portée (paragraphe 3.1.3) explique les assez bons résultats obtenus sur la hauteur des notes. La quasi-totalité des erreurs est due aux espacements inégaux des petites lignes additionnelles au-dessus ou au-dessous de la portée.

99.28% des noires, croches, etc. ont une durée correcte, malgré les interconnexions entre barres de groupe, et la présence de nombreux n-olets dans la base de données. La détection précise de la barre de groupe la plus externe, et la modélisation floue des groupes de notes, ont permis d'atteindre cette fiabilité. Les erreurs commises compensent souvent des erreurs de reconnaissance (Figure 7.8a) ; elles sont parfois dues à certains rythmes rares qui ne sont pas encore modélisés (b)(c), aux anacrouses et aux mesures de reprise qui amènent de fausses corrections (d), ou au contraire rejettent les corrections pour satisfaire au nombre de temps par mesure (e). Notons que les triolets de noires ou de blanches ne sont pas encore gérés.

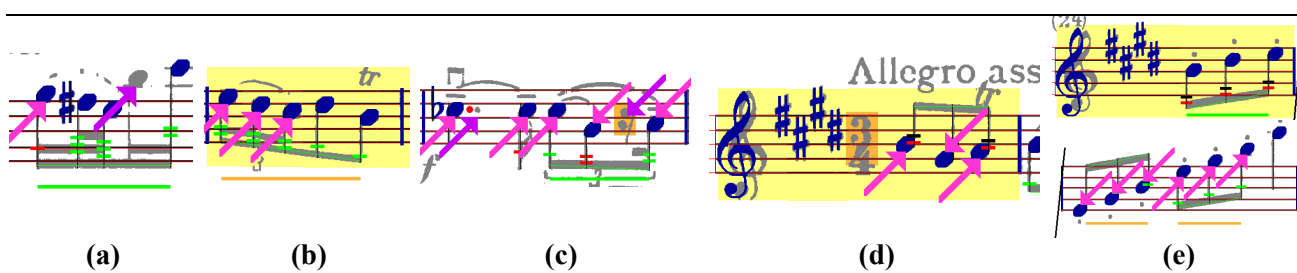


Figure 7.8 : Exemples d'erreurs sur la durée des notes

7.3.3. Apport de la modélisation floue

Le programme de reconnaissance a été lancé sur toute la base de données, en ne gardant que les deux règles strictes : la règle sur les altérations de tonalité et celle sur le nombre de temps par mesure (règles 4 et 6 du paragraphe 1.1). La décision consiste alors à choisir la combinaison d'hypothèses satisfaisant à la métrique, et maximisant le score de corrélation moyen. Le tableau 7.8 compare les résultats obtenus, avec et sans modélisation floue.

| Classes | | Avec modèle flou | | Sans modèle flou | | Apport modèle flou | |
|---------------|---|------------------|--------------|------------------|--------------|--------------------|---------------|
| | | $\tau(k)$ | $r^{(A)}(k)$ | $\tau(k)$ | $r^{(A)}(k)$ | $\tau(k)$ | $r^{(A)}(k)$ |
| Barres | | 99.59 | 0.33 | 99.59 | 0.33 | 0.00 | 0.00 |
| Altérations | \flat \natural \sharp | 97.86 | 0.12 | 89.93 | 43.08 | +7,93 | -42.96 |
| Appoggiatures | ♪ | 64.55 | 8.69 | 67.37 | 50.00 | -2,82 | -41.31 |
| Points | . | 97.48 | 0.70 | 96.95 | 4.66 | +0,53 | -3.96 |
| Notes | \bullet \circ \circ | 99.87 | 0.04 | 99.54 | 0.27 | +0.33 | -0.23 |
| Silences | ♪ ♪ ♪ ♪ ♪ ♪ ♪ ♪ | 98.01 | 2.31 | 95.51 | 5.13 | +2.50 | -2.82 |
| TOTAL | | 99.20 | 0.30 | 98.32 | 4.10 | +0.88 | -3.80 |

| | | | |
|---------------------------------------|--------------|--------------|--------------|
| DUREES (noires, croches, etc.) | 99.28 | 95.59 | +3.69 |
|---------------------------------------|--------------|--------------|--------------|

Tableau 7.8 : Apport de la modélisation floue (pourcentages)

La modélisation floue améliore les résultats sur toutes les classes (sauf la reconnaissance des appoggiatures) et les durées. Le mauvais résultat sur les appoggiatures s'explique par la difficulté à modéliser leur position par rapport à la note : la modélisation actuelle en rejette un grand nombre. En revanche, le gain obtenu sur les altérations, qui n'interviennent pas dans la métrique, est particulièrement significatif. Au total, le taux de reconnaissance des symboles est augmenté de 0.88%, le taux de symboles ajoutés est diminué de 3.8%, l'interprétation de la durée des notes noires s'améliore de 3.7%. En pratique, cela correspond, en moyenne, à une réduction de plus de 28 erreurs à corriger sur une page de musique de 10 portées (Tableau 7.9) : 13 erreurs sur les symboles à reconnaître, 15 erreurs sur les ajouts. Ces résultats prouvent l'intérêt de la modélisation floue, qui par ailleurs ne représente en moyenne que 10 % de la durée totale consacrée aux prétraitements et à l'analyse individuelle des symboles (paragraphe 7.4).

| | Confusions et symb. manquants | Ajouts | Durées | TOTAL |
|-------------------|-------------------------------|-------------|------------|-------------|
| Sans modèle flou | 6.8 | 16.6 | 11.3 | 34.7 |
| Avec modèle flou | 3.2 | 1.2 | 1.8 | 6.2 |
| DIFFERENCE | 3.6 | 15.4 | 9.5 | 28.5 |

Tableau 7.9 : Evaluation du nombre moyen d'erreurs sur une page de musique de 10 portées

7.3.4. Robustesse aux paramètres

Les méthodes appliquées pour la segmentation et l'analyse individuelle des symboles mettent en jeu de nombreux paramètres, qui ont été déduits de l'observation des partitions musicales. Ils représentent des connaissances a priori sur la notation. Les résultats présentés dans cette section vérifient la robustesse du système par rapport à ces paramètres.

Le tableau 7.10 indique les taux de reconnaissance qui sont obtenus en faisant varier les différents seuils utilisés en préclassification (Tableau 4.1) : dans le sens du relâchement ou en les rendant plus sévères. Relâché de 10% signifie par exemple que le test $x < S$ est devenu $x < 1.1S$.

| | Relâchés 20% | Relâchés 10% | Initiaux | Tendus 10% | Tendus 20% |
|---------------------------------------|---------------|---------------|---------------|---------------|---------------|
| Taux de reconnaissance | 98,05% | 99.02% | 99.20% | 99.03% | 98.39% |
| Taux de symboles ajoutés | 0.40% | 0.36% | 0.30% | 0.31% | 0.33% |
| Durées (noires, croches, etc.) | 98.96% | 99.27% | 99.28% | 99.15% | 99.10% |

Tableau 7.10 : Robustesse par rapport aux paramètres de préclassification

On constate une bonne stabilité des résultats, globalement et sur chacune des classes, jusqu'à une variation de 10%. Au-delà, les résultats commencent à chuter, soit parce que les critères sont trop sévères (tendus 20%) et ne prennent pas assez en compte la variabilité de l'écriture musicale, soit parce que la discrimination devient trop faible (relâchés 20%).

Des tests similaires ont été réalisés pour vérifier la robustesse par rapport à la définition des zones de corrélation, en fonction de la classe (Tableau 4.2) :

| | Restreintes 25% | Restreintes 10% | Initiales | Etendues 10% | Etendues 25% | Etendues 50% |
|--------------------------------|--------------------|--------------------|-----------|-----------------|-----------------|-----------------|
| Taux de reconnaissance | 98.67% | 98.89% | 99.20% | 99.17% | 99.17% | 99.04% |
| Taux de symboles ajoutés | 0.33% | 0.29% | 0.30% | 0.32% | 0.34% | 0.40% |
| Durées (noires, croches, etc.) | 99.13% | 99.24% | 99.28% | 99.26% | 99.22% | 99.03% |

Tableau 7.11 : Robustesse par rapport à la définition des zones de corrélation

L'extension des zones de corrélation a une influence très faible sur les taux de reconnaissance finals. Ces résultats confirment la capacité de la méthode à reconnaître les symboles sans localisation préalable précise. Le coût de calcul accru justifie cependant les choix qui ont été faits.

Les performances diminuent lorsque les zones de corrélation sont réduites. Il faut souligner que cette perte est due essentiellement aux appoggiatures seules (-25% à -30% de perte sur cette classe) : la plage de variation initialement très faible dans la direction horizontale ($s_i/10$, Tableau 4.2) n'autorise pas une réduction supplémentaire. Le même phénomène se produit pour les bémols, dans une moindre mesure (-1% à -1.5%). En revanche, les résultats restent stables sur les autres classes.

On a également constaté une très bonne robustesse par rapport aux paramètres définissant les distributions de possibilité. Cela s'explique très simplement par le fait que la modélisation floue permet justement d'éviter de positionner des seuils d'acceptation ou de rejet, et que le plus important est finalement la relation d'ordre établie plutôt que les valeurs des degrés de possibilité en elles-mêmes. D'autre part, comme de nombreux critères sont fusionnés, l'influence de chaque paramètre se voit encore diminuée.

7.4. Temps de calcul

Le traitement complet d'une page de musique de 10 portées prend en moyenne 35 secondes, sur un Pentium 4 à 3.2 GHz : 30 secondes pour les prétraitements, la segmentation et l'analyse individuelle des symboles, 3 secondes pour la modélisation floue et la décision, 2 secondes pour tout le reste, dont la génération des données de sortie.

En moyenne, 350 combinaisons d'hypothèses de reconnaissance sont générées par mesure. La combinatoire de l'étape de décision est donc statistiquement tout à fait acceptable. D'autre part, l'interprétation de haut niveau prend 10 fois moins de temps que les traitements et l'analyse de bas niveau. Tous ces chiffres montrent que la méthode proposée, procédant par génération d'hypothèses puis décision, n'introduit pas une complexité rédhibitoire, et qu'elle est d'un point de vue pratique parfaitement applicable. Néanmoins, comme le nombre de configurations d'hypothèses croît exponentiellement, il serait souhaitable de trouver des heuristiques qui permettraient de limiter

intelligemment le nombre de combinaisons à explorer.

7.5. Comparaison avec un logiciel du commerce

Bien que les publicités faites sur les logiciels commerciaux d'OMR affichent de très bons taux de reconnaissance, les utilisateurs remarquent que les erreurs peuvent être, en pratique, très nombreuses sur les partitions présentant les difficultés mentionnées au chapitre 1. C'est pourquoi il est intéressant d'effectuer quelques comparaisons, afin de vérifier l'apport de la méthode présentée. Le logiciel testé est SmartScore 5.0 Pro Demo [SmartScore 06].

La figure 7.9 indique des résultats sur quelques mesures extraites d'une même partition :

Figure 7.9 : Comparaison avec [SmartScore 06].

Notre méthode (1^{ère} ligne): 0 erreur, SmartScore (2^{ème} ligne): 16 erreurs

Ces exemples montrent que la méthode proposée permet de résoudre des cas pour lesquels SmartScore échoue : la résolution de l'ambiguïté de classification (c)(f), en particulier entre points de staccato et points de durée (a)(b), la reconnaissance des altérations (a)(b)(d)(f)(g), le problème des symboles qui se touchent (b)(d)(f), la détection et la cohérence rythmique des groupes de notes (b)(d)(e) et de la mesure (a)(b)(c)(d)(e)(f). Les taux de reconnaissance obtenus sur les deux pages complètes de musique, dont sont extraits ces exemples, sont de 92% pour SmartScore, 98.7% pour notre programme. La durée des croches est correcte à 85.3% avec SmartScore, à 99.3% avec notre programme. Ces exemples, qui sont représentatifs des erreurs typiques commises par SmartScore, tendent à prouver que l'ambiguïté est mieux résolue avec la méthode proposée, grâce notamment à la modélisation floue des règles de musique et à leur intégration dans le processus de décision.

Il est important d'insister sur la qualité de la reconnaissance des groupes de notes. Celle-ci dépend bien sûr de la fiabilité des algorithmes permettant d'extraire les primitives : têtes de note, hampes, barres de groupe, points, silences. Mais elle est également représentative de la capacité du système à reconstruire les relations liant ces primitives, et donc de sa capacité à passer d'une analyse symbolique à une analyse syntaxique et sémantique de plus haut niveau. Les erreurs commises par SmartScore, illustrées dans les figures 7.9(a)(b)(d)(e), sont extrêmement fréquentes, même sur des partitions bien imprimées. Notre méthode permet au contraire de reconstruire les groupes de manière très fiable, de corriger des erreurs de durée isolées, de bien interpréter les points ainsi que les silences remplaçant des notes dans le groupe. La figure 7.10 montre d'autres exemples illustrant ce propos.

Figure 7.10 : Reconnaissance des groupes de notes et de silences (partition originale en 1^{ère} ligne) : comparaison de la méthode proposée (2^{ème} ligne) avec SmartScore (3^{ème} ligne)

Légende :

| | | | | |
|-----|-----|-----|-----|-----|
| | | | | |
| 1/2 | 1/3 | 2/7 | 1/4 | 1/5 |

Figure 7.11 : Reconnaissance des triolets : comparaison de la méthode proposée (1^{ère} ligne) avec SmartScore (2^{ème} ligne)

Figure 7.12 : Reconnaissance des n-plets. Notre système (1^{ère} ligne) reconnaît mieux ces rythmes que SmartScore (2^{ème} ligne).

La reconnaissance des triolets nécessite de cocher une option dans les paramètres d'entrée de SmartScore. On constate que cela provoque des erreurs (Figure 7.11a), et que cela ne permet pas de résoudre correctement tous les triolets (Figure 7.11b), notamment ceux qui ne sont pas explicitement indiqués (Figure 7.11c). Enfin, les quintolets, sextolets, septolets, etc. sont très mal reconnus, de même que les triolets inclus dans des groupes. La figure 7.12 en donne des exemples. Rappelons que la reconnaissance des n-olets ne nécessite aucune indication dans les paramètres d'entrée de notre système, sauf dans le dernier cas (Figure 7.12d).

Voyons enfin quels sont les points forts de SmartScore par rapport à notre système. Tout d'abord, ce logiciel permet de reconnaître des classes de symboles qui ne sont pas encore intégrées dans notre méthode : doubles points, doubles bémols, doubles dièses, liaisons, barres de reprise, clé, signature temporelle, ornements, etc. D'autre part, les principales erreurs commises par notre programme, mais évitées par SmartScore, sont faites sur les mesures qui n'ont pas le nombre de temps requis par la métrique : les anacrouses et les mesures de reprise. Notre système a tendance à ajouter des symboles ou à effectuer des corrections inappropriées, contrairement à SmartScore, qui néanmoins ne résout pas toujours correctement tous ces cas. La figure 7.13 indique quelques exemples.

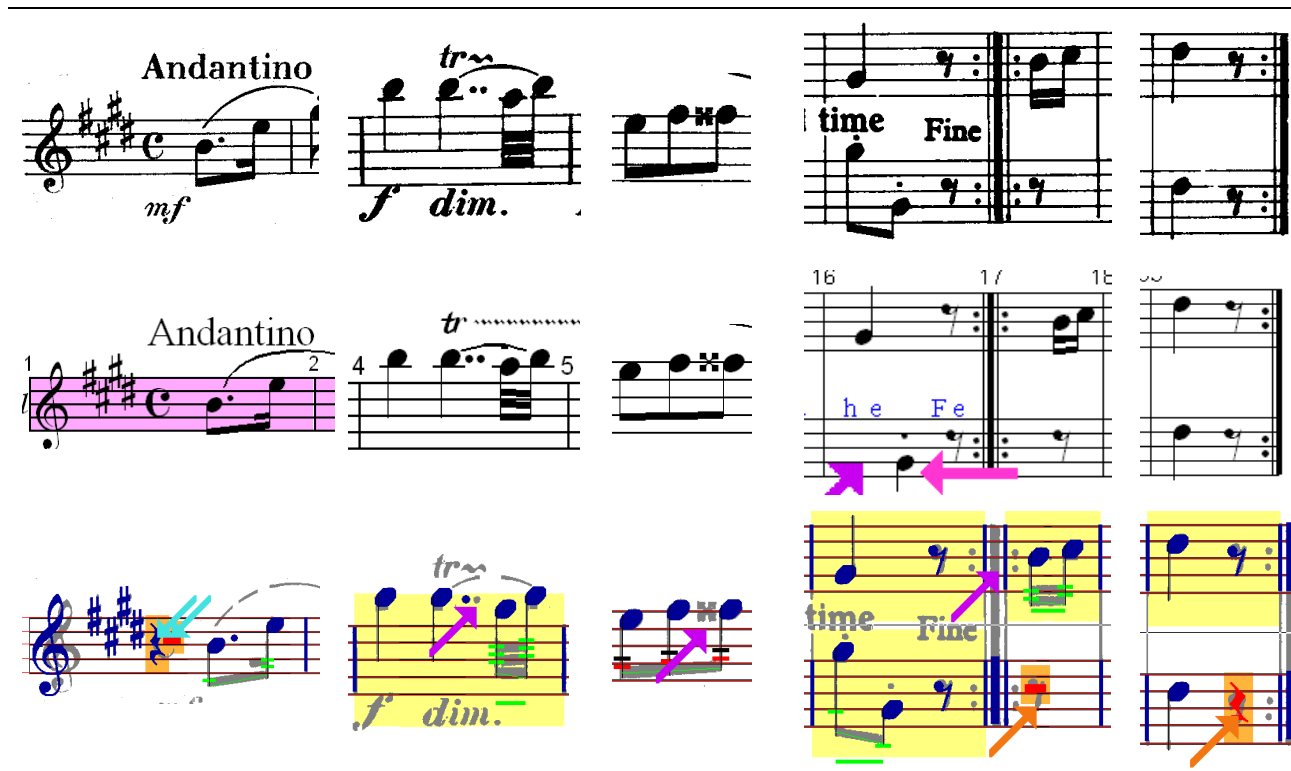


Figure 7.13 : Exemples de configurations pouvant être reconnues par SmartScore, encore non gérées par notre système. 1^{ère} ligne : original, 2^{ème} ligne : SmartScore, 3^{ème} ligne : notre système.

7.6. Résultats sur l'indication des erreurs potentielles

Nous présentons dans ce paragraphe des statistiques sur l'indication des erreurs potentielles.

Nous ne considérons que les erreurs pouvant actuellement être détectées : confusions, ajouts, symboles manquants, erreurs de durée de note. Les erreurs commises sur la hauteur des notes ne sont pas considérées dans les résultats présentés, puisqu'elles ne sont pas encore gérées.

Les pourcentages de détection d'erreurs, donnés dans les colonnes 2, 3 et 4 du tableau 7.12, sont calculés par rapport au nombre d'erreurs de chaque type, puis sur le total des erreurs (dernière ligne) ; les pourcentages de fausses alarmes (dernière colonne) sont rapportés au nombre total de symboles à reconnaître. Une erreur est dite "indirectement détectée" lorsqu'une autre erreur est présente dans la même mesure et que celle-ci a été correctement pointée, ou lorsque la mesure est indiquée comme potentiellement fausse.

52% des erreurs sont directement détectées, 32% sont facilement repérées grâce aux précédentes indications. Les confusions sont particulièrement bien traitées. Au total, 84% des erreurs sont indiquées, au moins approximativement, pour un taux de fausses alarmes de 2.6%. Cela signifie concrètement que, sur une page de musique de 10 portées, il y a en moyenne environ 5 erreurs détectées, 1 erreur non détectée, et 10 fausses alarmes. Sachant qu'une telle partition contient plus de 400 symboles, on peut dire que la méthode proposée permet effectivement de faciliter la correction manuelle du résultat de reconnaissance. Notons que deux tiers des fausses alarmes sur les symboles manquants sont des points (dus à des bruits). En introduisant un critère supplémentaire sur cette classe (degré de possibilité strictement positif), le taux de fausses alarmes décroît de 0.2% (0.88% au lieu de 1.08%). Ainsi, le nombre moyen d'indications erronées devient inférieur à 10, le nombre de détections correctes restant quant à lui inchangé.

| Erreur : | Directement détectée (%) | Indirectement détectée (%) | Détectée (%) | Fausse alarmes (%) |
|------------------|--------------------------|----------------------------|--------------|----------------------------|
| Symbole ajouté | 48.28 | 34.48 | 82.76 | 1.50 |
| Confusion | 78.57 | 15.31 | 93.88 | |
| Symbole manquant | 60.07 | 21.88 | 81.94 | 1.08 |
| Erreur de durée | 32.27 | 50.45 | 82.73 | <10⁻² |
| TOTAL | 52.06 | 31.82 | 83.89 | 2.58 |

Tableau 7.12 : Indication des erreurs potentielles

7.7. Evaluation de la méthode d'apprentissage supervisé

Les expérimentations présentées dans ce chapitre concernent la méthode d'apprentissage supervisé. Celle-ci a été appliquée lorsqu'une ou quelques classes de symboles obtiennent des taux de reconnaissance médiocres avec les modèles génériques.

Le tableau 7.13 indique les taux de reconnaissance obtenus sur la partition illustrant la méthode présentée au paragraphe 6.2. Rappelons que cette partition comprend 177 portées, 11 d'entre elles ayant servi à l'apprentissage. Les classes "dièse", "bécarre" et "quart de soupir" sont

maintenant parfaitement reconnues. Les résultats sur d'autres classes se sont également améliorés. Le taux de reconnaissance global passe de 99.69% à 99.96% ; le taux de symboles ajoutés diminue, de 0.53% à 0.11%.

| Classe | Sans apprentissage | Apprentissage supervisé | Classe | Sans apprentissage | Apprentissage supervisé |
|--------|--------------------|-------------------------|--------|--------------------|-------------------------|
| b | 99.51 | 99.84 | . | 97.65 | 98.23 |
| h | 99.43 | 100.00 | 7 | 93.75 | 100.00 |
| # | 90.77 | 100.00 | ● | 99.94 | 99.98 |

Tableau 7.13 : Taux de reconnaissance avant et après apprentissage supervisé (exemple 1, 16 pages de musique). Les autres taux de reconnaissance sont inchangés.

Le second exemple (Tableau 7.14) n'inclut que deux pages de musique (21 portées). Il est cependant très intéressant car certains taux de reconnaissance sont particulièrement faibles. Ces mauvais résultats sont dus à une impression en traits très gras, qui ne convient pas aux modèles génériques de classe. 7 portées ont servi à l'apprentissage. On constate de nouveau une nette amélioration : le taux de reconnaissance global augmente, de 95.64% à 98.09%, le taux de symboles ajoutés diminue, de 3.13% à 2.32%. Les résultats finals ne sont cependant pas encore très bons, à cause d'un manque de fiabilité dans la détection des barres de mesure (obliques, dépassant parfois la portée).

| Classe | Sans apprentissage | Apprentissage supervisé | Classe | Sans apprentissage | Apprentissage supervisé |
|--------|--------------------|-------------------------|--------|--------------------|-------------------------|
| # | 91.67 | 100.00 | 7 | 87.50 | 100.00 |
| 7 | 0.00 | 71.43 | . | 95.24 | 100.00 |
| 7 | 81.25 | 100.00 | | | |

Tableau 7.14 : Taux de reconnaissance avant et après apprentissage supervisé (exemple 2, 2 pages de musique, 21 portées).

Terminons par un troisième exemple. La partition comprend cette fois 49 portées, et 5 d'entre elles ont participé à l'apprentissage. Les erreurs de reconnaissance portaient sur la distinction entre dièse et bécarré. Le tableau 7.15 indique les résultats obtenus pour ces deux classes, les autres étant inchangés. Le taux de reconnaissance global passe de 99.69% à 99.87%, le taux de symboles ajoutés diminue, de 0.35% à 0.09%.

| Classe | Sans apprentissage | Apprentissage supervisé | Classe | Sans apprentissage | Apprentissage supervisé |
|--------|--------------------|-------------------------|--------|--------------------|-------------------------|
| # | 97.2 | 99.3 | h | 81.82 | 100.00 |

Tableau 7.15 : Taux de reconnaissance avant et après apprentissage supervisé (exemple 3, 5 pages de musique, 49 portées).

Ces résultats prouvent de nouveau l'intérêt de l'apprentissage. Celui-ci est moins évident lorsque les résultats de reconnaissance sont plutôt satisfaisants pour toutes les classes, et que les erreurs proviennent davantage de la mauvaise qualité de l'impression que de l'inadéquation des modèles génériques de classe. En effet, la modélisation floue permet de tolérer une certaine variabilité entre les symboles de la partition et les modèles, et l'intégration des règles graphiques et

syntaxiques contribue à lever l'ambiguïté restante. On conclura donc que l'apprentissage est souhaitable lorsque de nombreuses confusions sont commises sur une ou plusieurs classes, l'apprentissage des modèles servant à améliorer la discrimination.

7.8. Conclusion

Nous avons présenté dans ce chapitre de nombreux résultats permettant d'évaluer les différentes étapes du système de reconnaissance, sa robustesse et sa fiabilité. Ces résultats étant obtenus sur une large base de données, on peut penser qu'ils sont représentatifs. En particulier, ils valident les nombreux paramètres utilisés pour la segmentation et l'analyse individuelle des symboles, et prouvent l'intérêt de la modélisation floue. Sur la centaine de pages de musique testées, le taux de reconnaissance des symboles est de 99.2%, avec 0.3% de symboles ajoutés. 99.3% des notes ont une durée exacte. Sur une partition de 10 portées (plus de 400 symboles), ces taux correspondent en moyenne à 6 erreurs à corriger. Cinq d'entre elles sont indiquées par le programme, et 10 fausses indications sont inutilement vérifiées. Des améliorations sont à réaliser au niveau de l'interprétation de la hauteur des notes (1% d'erreurs) et de l'indication automatique de ce type d'erreurs.

L'exécution du programme de reconnaissance prend en moyenne 35 secondes sur un Pentium 4 à 3.2 GHZ, bien qu'aucune optimisation n'ait été effectuée. Une simple restructuration du code conduirait déjà à des gains de temps.

Tous ces éléments tendent à prouver que la méthode conduit à des résultats fiables, utilisables en pratique, au moins sur des images qui ont été obtenues à partir de partitions qui n'ont pas été physiquement dégradées, et qui ont été correctement numérisés. On a pu d'ailleurs constater que cette méthode apporte des réponses à des problèmes qui ne sont pas résolus par SmartScore, l'un des logiciels commerciaux les plus performants [SmartScore 06].

Quelques exemples ont également illustré l'intérêt de la procédure d'apprentissage, qui permet, moyennant une intervention limitée de l'utilisateur, d'apprendre de nouveaux modèles de classe et d'améliorer les résultats.

Les points forts de la méthode, ainsi que les axes d'amélioration, seront discutés en détail dans le chapitre de conclusion.

CHAPITRE 8

Conclusion

Nous avons décrit un système complet de reconnaissance de partitions imprimées, dans le cas de la musique monodique, et proposé des procédures permettant de gagner en fiabilité. Ce système a été testé sur une large base de partitions. L'objet de cette conclusion est de résumer les principales caractéristiques de la méthode proposée, de dégager les contributions et les axes d'amélioration.

8.1. Méthode proposée et caractéristiques

Le système comprend trois modules de traitement et d'analyse : les prétraitements et la segmentation de l'image, l'analyse individuelle des symboles, l'analyse de haut niveau aboutissant à la décision. A l'instar de nombreux systèmes présentés dans la littérature, il s'agit d'une analyse ascendante, mais contrairement à la plupart d'entre eux, les différentes étapes sont bien séparées. Cette séparation nette répond à l'un des objectifs qui avaient été fixés, dans le but d'intégrer de manière rigoureuse les connaissances a priori qui peuvent être utilisées pour la reconnaissance, et de mieux gérer l'ambiguïté. Chaque étape comprend des méthodes innovantes, conçues pour répondre aux problèmes spécifiques du domaine de l'OMR, identifiés dans le chapitre 1.

Les prétraitements sont limités au redressement de l'image, le biais étant déduit d'un simple calcul d'autocorrélation. Les efforts ont ensuite porté sur la qualité de la segmentation : nous avons proposé tout d'abord une méthode de filtrage permettant de localiser précisément les lignes de portée, capable de surmonter les défauts usuels (biais résiduel, courbures, variations d'épaisseur), robuste aux symboles interférents. Les résultats obtenus sont essentiels pour la suite, puisqu'ils permettent d'amorcer la segmentation par l'effacement des lignes de portée, et qu'ils sont utilisés pour définir la position des primitives et des symboles relativement à la portée. La suppression des lignes de portée suffit à séparer les symboles isolés, tels les silences, les rondes, les points. Tous les autres symboles sont caractérisés par la présence d'au moins un segment vertical, qui sert à leur localisation. A cet effet, nous avons proposé un détecteur de segment vertical, robuste aux principaux défauts d'impression : biais, faibles ruptures, connexions parasites. Cette méthode est efficace, notamment en cas de connexions entre symboles théoriquement séparés, typiquement les têtes de note et les altérations. Le fractionnement assez fréquent des hampes, ainsi que leur biais, sont également bien gérés. Les résultats sont bons, dans le sens où peu de symboles sont manqués.

Néanmoins des défauts ne peuvent être évités à ce stade de l'analyse : sur-détections, imprécisions sur la forme des objets (effacement des lignes de portée imparfait) ou sur leur localisation (taille de la boîte englobante).

La seconde étape réalise l'analyse individuelle des symboles, par corrélation avec des modèles de référence. Les zones de corrélation sont déduites des résultats de segmentation. Cette méthode d'analyse n'avait encore jamais été utilisée de manière systématique, pour la reconnaissance de toutes les classes de symboles. Elle permet de tolérer les défauts d'impression et les imprécisions de segmentation, et de générer des hypothèses de reconnaissance pertinentes, comme le démontrent les résultats obtenus sur toute la base de données. Des hypothèses sont également émises sur la présence de barres de groupe reliant les noires potentielles. Une caractéristique essentielle de la méthode que nous proposons est en effet de ne prendre aucune décision définitive à ce niveau, que ce soit sur les symboles isolés ou sur les symboles composés. Au contraire, toute l'ambiguïté de classification, qui résulte des défauts de l'image, des imprécisions de segmentation, de la variabilité des symboles, est maintenue. C'est la modélisation et l'intégration des règles musicales qui permettra de choisir une combinaison d'hypothèses de reconnaissance, conforme à la syntaxe musicale, par vérification de leur cohérence mutuelle.

Il est à noter que de nombreux paramètres, modélisant des connaissances a priori, ont été définis dans les étapes de segmentation et d'analyse des symboles. Pour la détection des lignes de portée par exemple, nous avons fait l'hypothèse qu'elles sont constituées de 5 lignes horizontales équidistantes. De même, des hypothèses ont été émises sur la longueur minimale des segments verticaux, la largeur maximale des symboles, la structure des groupes de notes, la position des symboles sur la portée, etc. Tous ces paramètres ont été définis avec suffisamment de souplesse pour couvrir les différents styles d'édition, et autoriser des défauts d'impression et de mise en page. Les résultats obtenus sur toute notre base de données montrent qu'ils sont effectivement pertinents, puisque les hypothèses générées incluent la classe exacte pour 99.7% des symboles. Comme ils ont été normalisés par rapport à la valeur de l'interligne, ils pourraient également s'appliquer à d'autres tailles ou résolutions d'image, ce qui est un critère de généralité important de la méthode. L'autre caractéristique essentielle de notre système est qu'aucune règle, régissant les relations entre les symboles, n'a encore été intégrée à ce stade : il y a effectivement une séparation totale entre l'analyse de bas niveau et l'analyse de haut niveau, les primitives musicales ayant été jusqu'à présent analysées séparément les unes des autres, sans ajout d'aucun contexte.

L'objet de l'analyse de haut niveau est de lever l'ambiguïté des hypothèses de reconnaissance, en interprétant les résultats obtenus, et en incorporant les règles musicales. C'est une démarche particulièrement originale, la tendance générale étant plutôt d'utiliser des méthodes rétroactives pour revoir des résultats de classification. Nous pensons cependant qu'elle est très performante, car la décision peut être prise en intégrant la totalité du contexte, au contraire des méthodes rétroactives qui utilisent des informations plus partielles. De plus, procéder par génération d'hypothèses de reconnaissance et décision prend tout son sens dans notre système, puisque notre modélisation va au-delà des règles graphiques locales et qu'elle inclut des règles syntaxiques, impliquant de nombreux symboles distants.

Les règles de la notation musicale sont difficiles à modéliser, à cause de leur flexibilité, de leur hétérogénéité, du fait qu'elles s'appliquent à des niveaux d'abstraction différents et qu'elles concernent un nombre variable de symboles, plus ou moins éloignés dans l'image. Les systèmes proposés jusqu'à présent se sont surtout concentrés sur la modélisation des règles structurelles, permettant de recomposer les groupes de notes à partir de primitives préalablement reconnues, ainsi que sur les règles graphiques, telles que la position d'une altération par rapport à la tête de note. La modélisation que nous proposons, fondée sur la théorie des ensembles flous et des possibilités, apporte une réponse à des problèmes importants, insuffisamment résolus dans la littérature : l'intégration des règles syntaxiques malgré leur flexibilité, la fusion de toutes les informations dans un même formalisme, la prise en compte de toutes les sources d'incertitude (dues à l'imprécision des informations extraites ou aux connaissances génériques elles-mêmes). En outre, cette modélisation nous permet de traiter la variabilité des symboles, en adaptant le modèle de classe à la partition traitée, sur la base des scores de corrélation obtenus sur toute la page de musique. Notons pour terminer que la reconstruction des groupes de notes ne se limite pas à la vérification de critères graphiques locaux d'assemblage, mais qu'elle est au contraire finalisée après introduction de tout le contexte. Tous ces aspects sont novateurs par rapport à la bibliographie.

Les résultats obtenus démontrent la pertinence des modèles proposés, qui conduisent à un taux de reconnaissance moyen de 99.2% (avec 0.3% de symboles ajoutés). Par ailleurs, 99.3% des notes ont une durée exacte, 99% ont une hauteur correcte. Les images de bonne qualité sont généralement très bien, voire parfaitement, reconnues. Les autres présentent davantage d'ambiguïté, mais sont néanmoins assez bien interprétées (taux de reconnaissance généralement supérieurs à 98%). Ces résultats reposent sur les méthodes d'extraction des informations (segmentation qui surmonte les défauts d'impression courants combinée au template matching), sur la modélisation floue et la structure du système qui permettent de tolérer des imprécisions et des incertitudes, de les propager de bout en bout, jusqu'à la décision finale, prise globalement par optimisation de tous les critères.

Des procédures permettant de gagner en robustesse et en facilité d'utilisation ont également été proposées, complétant le système de reconnaissance. Tout d'abord, la plupart des erreurs sont directement ou indirectement indiquées, d'après les résultats de la modélisation floue. La correction manuelle, lourde et fastidieuse, est ainsi facilitée. L'apprentissage d'une partition particulière est également possible. Il permet d'ajuster les modèles de classe et les paramètres liés à ces modèles, et de gagner en fiabilité. Les expérimentations ont montré l'intérêt d'appliquer cette procédure lorsque certaines classes de symboles ne sont pas reconnues de façon satisfaisante, à cause d'une trop grande différence entre les symboles de la partition et les modèles de référence. Ces deux axes sont aussi très novateurs par rapport à la bibliographie.

8.2. Compléments

8.2.1. Améliorations diverses

Un axe d'amélioration important concerne la segmentation des symboles qui ne sont pas

caractérisés par un segment vertical. Dans le système actuel, la corrélation est effectuée sur tous les espaces libres entre les boîtes englobantes des autres symboles. En conséquence, une proportion importante des hypothèses générées ne correspond à aucun silence. Il serait donc intéressant d'effectuer une segmentation complète, par analyse de connexité, suivie éventuellement d'une préclassification. La corrélation serait effectuée sur des zones plus restreintes, avec les modèles de classe appropriés, comme pour les symboles caractérisés par un segment vertical. Les hypothèses seraient beaucoup plus pertinentes et les taux de reconnaissance probablement améliorés. En particulier, le taux de symboles ajoutés serait considérablement réduit.

Les résultats sur l'interprétation de la hauteur des notes ne sont pas totalement satisfaisants, avec 99% de réussite. Pour améliorer ce taux, il faudrait détecter les petites lignes additionnelles au-dessus et au-dessous de la portée, et non pas se contenter de les extrapoler.

Enfin, nous avons déjà mentionné la possibilité de diminuer le coût de calcul au niveau de l'évaluation des hypothèses de reconnaissance : tout d'abord en rejetant d'emblée toutes les hypothèses qui sont absolument impossibles (par exemple les altérations qui ne sont compatibles avec aucune note), éventuellement en déterminant des heuristiques qui permettent d'accélérer le processus de décision, sans perte notable au niveau du taux de reconnaissance.

8.2.2. Compléments dans l'analyse des symboles

Le système proposé n'est pas complet. En effet les doubles points, les doubles bémols et les doubles dièses ne sont pas encore gérés. Intégrer ces configurations supposerait quelques ajustements dans l'analyse de bas niveau (recherche d'un second point, modèle de classe du double dièse), et l'intégration de nouvelles règles dans l'analyse de haut niveau : règles graphiques sur la position des doubles altérations ou des deux points, règles syntaxiques sur la cohérence de ces symboles par rapport aux autres informations extraites et aux paramètres globaux. Des modèles flous peuvent être définis, dans la continuité de ce qui a été présenté, et intégrés sans difficulté.


Afin de compléter tous les symboles essentiels à la restitution de la mélodie, il faudrait également différencier les barres de mesure : barres simples, finales, barres de reprise. La méthode de détection des barres de mesure permet de connaître leur épaisseur. En ajoutant la recherche des deux points de part et d'autre de la troisième ligne de portée, et la détection de barres consécutives, il devient possible d'émettre des hypothèses sur la nature des barres de mesure.

8.2.3. Reconnaissance automatique des informations globales

La clé, la tonalité et la métrique sont données en paramètres d'entrée du programme. Ces informations ne sont pas lourdes à entrer, et conduisent à une plus grande fiabilité. Néanmoins, les changements en cours de partition ne sont pas possibles, et cette limitation nuit à la généralité du système.

L'analyse par corrélation n'est probablement pas adaptée à la reconnaissance des clés, à

cause de leur très grande variabilité. Il serait plus approprié de réaliser une segmentation puis une analyse structurelle. Cette technique est a priori possible car ces symboles sont bien séparés des autres objets. En ce qui concerne la signature temporelle, on peut penser aux méthodes utilisées pour la reconnaissance de chiffres et de caractères. Le problème est néanmoins complexifié par la présence des lignes de portée, dont l'effacement peut dégrader, voire fractionner les symboles (par exemple la clé de fa), rendant la reconnaissance ambiguë. Dans ce cas, l'idéal serait d'émettre des hypothèses de reconnaissance, et d'extraire la solution dans l'étape d'interprétation de haut niveau : en validant les positions relatives de la clé, des altérations de tonalité et de la signature temporelle, en examinant la hauteur des altérations de tonalité compte tenu de la clé, en vérifiant que la métrique trouvée est compatible avec l'interprétation des notes et des silences, etc. Cette prise en compte de l'ambiguïté dans la reconnaissance des informations globales nécessiterait de modifier la structure du système. Ce point sera discuté dans le paragraphe suivant.

La dernière information globale (optionnelle), est l'indication de la présence de n-olets dans des groupes de notes plus larges : présence de rythmes tels que "croche / triolet de doubles croches" . Sans indication donnée par l'utilisateur, un tel groupe est corrigé en 4 doubles croches ; le cas échéant, il est laissé tel quel, car ce rythme est permis. Afin de gérer complètement automatiquement tous les modèles de rythme, il devient nécessaire de rechercher la présence du chiffre n indiquant le n-olet. Le résultat de cette analyse doit également être intégré dans le modèle flou, pour prendre en compte l'incertitude sur la reconnaissance, et pour autoriser l'absence du chiffre, qui est fréquente.

8.3. Perspectives

Nous indiquons dans ce paragraphe des perspectives plus larges que les compléments mentionnés précédemment. Ces propositions devraient conduire à une plus grande fiabilité du système et permettre d'étendre la méthode aux partitions polyphoniques.

8.3.1. Reconnaissance à partir d'images dégradées

Les performances du système actuel chutent considérablement lorsque les images analysées sont fortement dégradées, à cause de la mauvaise qualité du document original, ou d'une mauvaise acquisition. Actuellement, les défauts tolérés sont les faibles ruptures, les connexions parasites. La reconnaissance échoue lorsque de nombreux pixels sont effacés, surtout au niveau des segments verticaux. La figure 8.1 illustre ce propos. On remarque que les petites coupures, les déconnexions entre hampe et barres de groupe, entre hampe et tête de note, sont bien supportées. En revanche, la note qui possède une hampe partiellement effacée n'est pas reconnue. Ce défaut arrive lorsque le document original est dégradé (cas de partitions anciennes) ou que la numérisation a été bâclée. Pour améliorer la qualité de l'image, il faudrait tout d'abord envisager une acquisition en niveaux de gris, suivie d'une binarisation adaptative, c'est-à-dire à seuil variable, déterminé pour chaque pixel en fonction des pixels voisins. Typiquement, cet algorithme permettrait de mettre à 1 (noir) des pixels gris qui semblent faire partie de segments, et de mettre à 0 (blanc) des pixels gris isolés

(bruit). On pourrait aussi appliquer des méthodes de restauration des structures linéaires dans les cas les plus graves, lorsque des portions entières de segments sont effacées dans la partition originale.

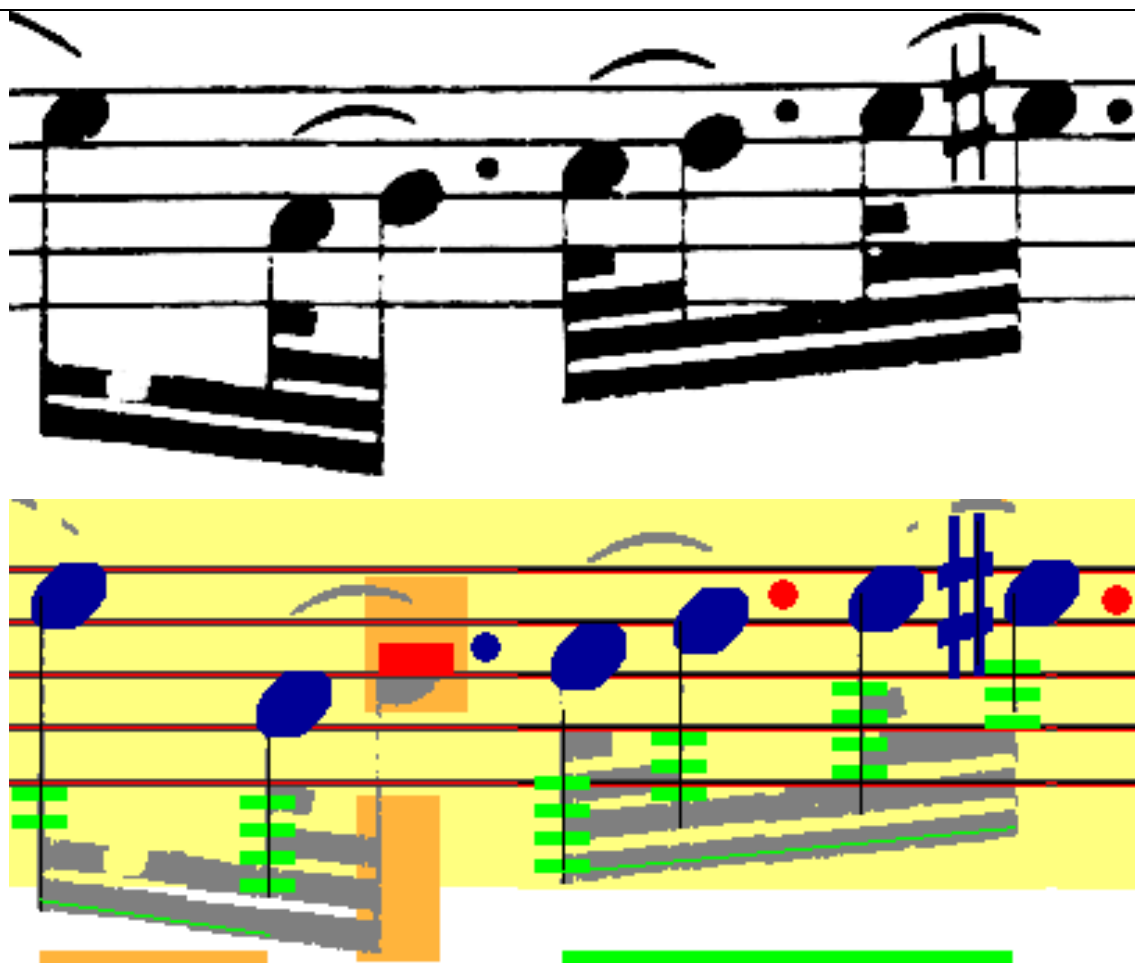


Figure 8.1 : Cas des images fortement dégradées.

8.3.2. Intégration d'informations structurelles

La reconnaissance des symboles est fondée sur les résultats de corrélation avec des modèles de référence. Nous avons beaucoup insisté sur la pertinence de cette méthode, qui, combinée à la détection robuste des segments verticaux, permet de mieux surmonter les défauts d'impression et les imprécisions de segmentation, et de générer des hypothèses de reconnaissance exploitables. On constate néanmoins des limitations. En effet, les scores de corrélation mesurent un taux de ressemblance moyen entre deux formes. Comme certaines classes de symboles sont fortement corrélées (bécarre et dièse, noire et blanche, demi-soupir et quart de soupir, etc.), le score de corrélation obtenu entre le symbole testé et un modèle d'une classe différente peut être assez élevé ; dans le même temps, le score de corrélation entre le symbole et le modèle générique de classe peut être, quant à lui, relativement faible, à cause de la variabilité des polices. D'où l'ambiguïté constatée, que nous levons par la vérification des règles musicales. Introduire également des informations structurelles compléterait l'analyse et conduirait probablement à une meilleure discrimination. Par exemple, les têtes de note noires ne devraient théoriquement pas inclure de pixels blancs (en tout

cas en pratique moins que les têtes de note blanches, même en présence de bruit), et ce critère très simple permettrait de contribuer au choix d'une classe plutôt qu'une autre. Cette idée ne remet pas du tout en cause l'intérêt du template matching : il s'agit simplement d'extraire explicitement des caractéristiques qui sont plus ou moins cachées dans les scores de corrélation, pour contribuer à lever l'ambiguïté entre des hypothèses concurrentes. Les questions qui se posent sont les suivantes : comment définir la zone image sur laquelle est calculé le vecteur d'attributs, quels attributs choisir, et comment les intégrer dans la décision? On peut probablement définir la zone image à partir des résultats de segmentation (boîte englobante), de la position correspondant au maximum de corrélation, et des dimensions typiques du symbole dans l'hypothèse de classe considérée. De nombreux attributs ont déjà été testés dans la littérature (e.g. [Fujinaga 97]), mais la sélection des plus pertinents dépend certainement des classes testées. Enfin, les résultats devront être intégrés dans le modèle flou, afin d'éviter des décisions rigides qui ne permettent pas de surmonter les difficultés liées à la qualité de l'image. Cette nouvelle voie semble importante à explorer, car nous avons effectivement constaté des confusions entre symboles corrélés (paragraphe 7.3), erreurs que nous pouvons espérer ainsi éviter.

8.3.3. Structure du système de reconnaissance

La structure du système actuel permet de prendre des décisions, mesure par mesure. Si nous introduisons l'extraction automatique des informations globales (clé, tonalité, métrique), alors il semble nécessaire d'ajouter un niveau d'interprétation, permettant de valider la compatibilité entre ces informations et les hypothèses de reconnaissance obtenues sur les mesures impliquées. Par exemple, la décomposition rythmique est révélatrice de la métrique [Ng et al. 95], et il est donc possible de vérifier que l'interprétation des notes, des silences et de la signature temporelle (déduite de la reconnaissance des chiffres) est cohérente. De même, il y a de fortes relations liant la clé, l'armure et les altérations accidentelles. La modélisation actuelle des règles musicales intègre presque tous ces concepts, mais les informations globales sont supposées certaines. La figure 8.2 indique ce que pourrait être la nouvelle architecture permettant l'extraction et la validation de ces informations.

Par rapport à la figure 2.3, les entrées spécifiques sont réduites à l'image. L'extraction des informations globales (clé, signature temporelle et armure) est réalisée dans la partie analyse d'image. En cas d'ambiguïté, différentes hypothèses peuvent être émises. La décision par mesure, telle que nous l'avons présentée, est effectuée pour chaque combinaison (clé, métrique, tonalité), et les résultats sont fusionnés sur toutes les mesures impliquées : c'est le niveau d'interprétation ajouté, qui permet d'évaluer la cohérence des informations globales par rapport aux informations locales extraites des mesures. Des règles portant sur la cohérence des informations globales entre elles (par exemple la clé par rapport aux altérations de tonalité), représentées par la flèche rouge, sont également intégrées, et la fusion de toutes ces informations conduit à la décision finale. C'est également à ce niveau que peuvent être gérées les mesures de reprise.

Naturellement, la génération d'hypothèses pour la reconnaissance des informations globales conduit à une complexité accrue au niveau de la décision. On peut supposer que la combinatoire

reste acceptable, car l'ambiguïté n'est probablement pas très élevée (par exemple, les clés sont très différentes entre elles et très différentes des autres symboles) et des règles strictes permettent d'écarter d'emblée des hypothèses impossibles (comme une armure incompatible avec la clé).

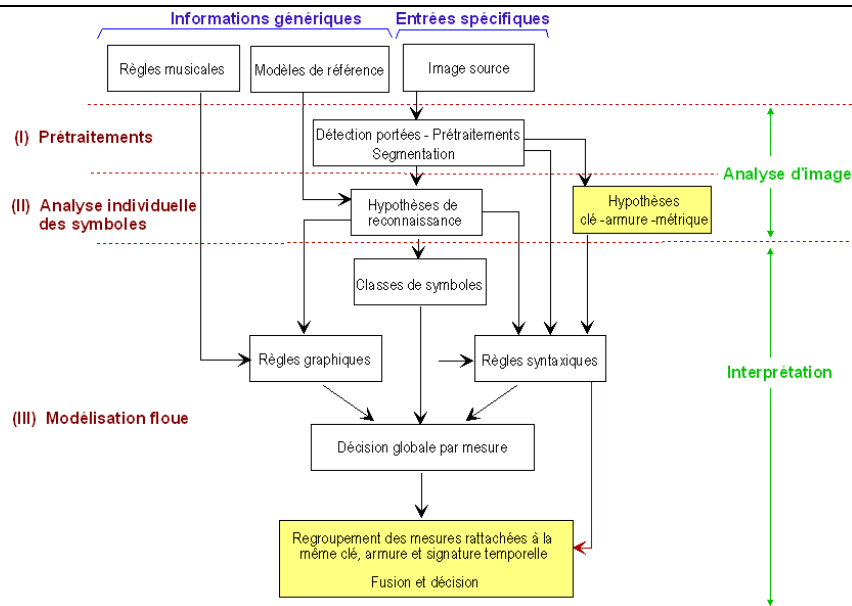


Figure 8.2 : Modification de la structure du système, pour l'extraction des informations globales

8.4. Extension à la musique polyphonique

Nous avons restreint notre étude à la musique monodique. D'un point de vue applicatif, c'est une hypothèse de travail très restrictive. Il convient maintenant d'étudier dans quelle mesure cette contrainte peut être relâchée, et également quels bénéfices peuvent être tirés de la méthode que nous avons présentée, pour la reconnaissance des partitions polyphoniques.

La première extension consiste à reconnaître les accords (Figure 8.3). Il faudrait tout d'abord autoriser la détection de plus d'un empan vertical dans une même colonne image, puisque les symboles caractérisés par un segment vertical peuvent maintenant se superposer (cas d'altérations l'une en dessous de l'autre (e)(g)). Il faudrait également étendre la zone de recherche des têtes de note, tout le long de la hampe.

L'assemblage des têtes de note suit des règles, qui peuvent être utilisées pour la reconnaissance : les têtes de note sont du même côté de la hampe, à gauche pour les hampes montantes (f), à droite pour les hampes descendantes (a), sauf lorsque l'accord contient une seconde (deux notes consécutives de la gamme) (b)(c)(d)(g). D'autre part, les notes d'un accord ont toutes la même durée, ce qui constitue une information supplémentaire exploitable, pour l'extraction des points notamment. Ceux-ci doivent être alignés verticalement, cette règle étant appliquée avec plus ou moins de précision (d)(e). Les règles floues modélisant la position d'une altération par rapport à la tête de note doivent prendre en compte la présence d'autres altérations dans l'accord, puisque les décalages horizontaux peuvent être augmentés pour éviter des chevauchements (d)(e)(f)(g). Elles doivent également considérer les notes situées de l'autre côté de la hampe (c). On constate donc la

nécessité de fusionner ces différentes informations, d'ordre graphique et syntaxique, afin de vérifier la cohérence globale des notes de l'accord et de leurs attributs.

Les principales modifications consistent donc à relâcher certaines contraintes dans les modules de bas-niveau, et à étendre les règles de musique. Le système proposé semble tout à fait approprié pour la modélisation, l'intégration et la fusion de ces nouvelles règles.

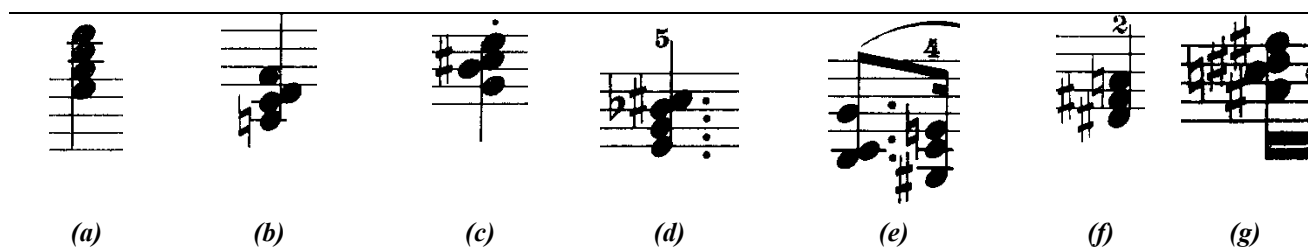


Figure 8.3 : Exemples d'accords

Le passage à des partitions réellement polyphoniques, typiquement les partitions de piano (Figure 8.4) semble plus délicat. Examinons chacune des étapes, afin de déterminer quels ajustements supplémentaires doivent être faits.

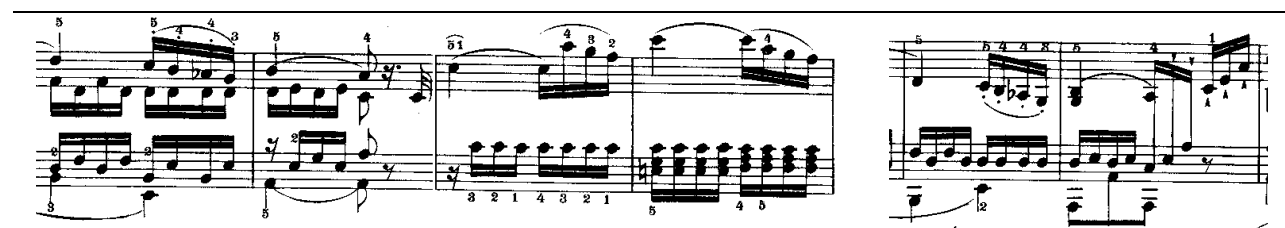


Figure 8.4 : Partition polyphonique (piano)

Il faut tout d'abord extraire les systèmes, par la reconnaissance des barres de mesure qui traversent et relient les portées. Cette étape ne devrait pas poser de difficultés ; au contraire, la fiabilité de la reconnaissance est probablement meilleure, car il n'y a plus de confusion possible avec les hampes.

A priori, il n'y a pas de nouvelles modifications à apporter dans les étapes de segmentation et d'analyse individuelle des symboles. Il faut juste relâcher la contrainte relative à la position des silences non inclus dans des groupes, puisqu'ils ne sont plus nécessairement centrés sur la portée. Comme les connexions entre objets sont d'autant plus fréquentes que la partition est dense, on peut en déduire que la recherche des segments verticaux constitue toujours un bon critère de détection. L'analyse des objets segmentés par template matching semble toujours pertinente, justifiée par la difficulté, encore accrue, de segmenter précisément les objets.

L'étape suivante consiste à séparer les différentes voix, compte tenu des hypothèses de reconnaissance. Il faut pour cela évaluer de nouvelles règles : position relative des symboles dans les deux directions, sens des hampes, barres de groupe, etc. C'est certainement une étape complexe, comme on peut l'imaginer en étudiant les mesures de la figure 8.4. Par exemple, le nombre de voix n'est pas constant, certaines voix passent d'une portée à l'autre, etc. La décision peut toujours être réalisée mesure par mesure, mais en considérant toutes les voix du système. Une méthode, applicable aux hypothèses de reconnaissance, pourrait être la suivante :

- Considérer chaque mesure du système (au sens une mesure correspondant à un instrument) indépendamment des autres.
 - Pour chaque combinaison d'hypothèses de reconnaissance :
 - Emettre des hypothèses sur la séparation des voix dans les portées polyphoniques, en testant les critères mentionnés ci-dessus. Les mesures des portées monodiques du système correspondent à une voix unique, sans ambiguïté.
 - Pour chaque hypothèse de séparation, évaluer les règles graphiques et syntaxiques sur chacune des voix, indépendamment des autres. Ces règles sont celles décrites dans le mémoire, augmentées des critères relatifs aux accords.
 - Retenir les configurations d'hypothèses possibles, rejeter toutes les autres.
- Combiner les mesures du système entre elles :
 - Combiner les différentes configurations de voix qui ont été retenues dans l'étape précédente. Evaluer des règles graphiques et syntaxiques sur la cohérence des voix entre elles, typiquement l'alignement vertical, d'un point de vue graphique (position relative des symboles) et d'un point de vue syntaxique (alignement temporel, compatibilité des altérations, etc.).
 - Fusionner toutes les informations pour prendre une décision globale, par optimisation de tous les critères.

Le passage à la musique polyphonique semble donc complexe, mais possible avec notre méthode. Les principales modifications sont à réaliser dans les niveaux d'interprétation de haut niveau, puisque de nouveaux critères sont à considérer. Le formalisme proposé est bien adapté à l'intégration de ces nouvelles règles, qui présentent des caractéristiques similaires aux règles déjà modélisées. Par exemple, l'alignement graphique des symboles des différentes voix est défini avec une certaine tolérance ; les degrés de possibilité affectés aux groupes de notes peuvent être fusionnés sur l'ensemble des voix, et des critères syntaxiques ajoutés pour évaluer, avec flexibilité, leur cohérence mutuelle ; en musique classique, on peut envisager de modéliser des règles d'harmonie, souples par nature, etc.

On peut craindre une explosion combinatoire du nombre de configurations d'hypothèses à explorer. Néanmoins on peut également supposer que l'ambiguïté au niveau de la séparation des voix n'est pas très importante, puisqu'une erreur commise sur un symbole n'affecte pas nécessairement le découpage. De plus, le traitement préliminaire voix par voix permet de ne retenir que quelques combinaisons d'hypothèses pertinentes pour chaque voix du système, avant de les recombinaisonner. Enfin, les nouveaux critères ajoutent un contexte très fort, qui permet de croiser les informations entre les voix, et par conséquent d'écartier d'emblée des configurations impossibles dans la dernière étape.

D'une manière générale, on peut dire que l'augmentation de l'ambiguïté, conséquence inévitable du nombre accru de symboles, est compensée par l'intégration de contraintes supplémentaires.

La méthode proposée pour le traitement des partitions monodiques semble donc tout à fait extensible aux partitions polyphoniques : d'une part, parce que la structure du système le permet, d'autre part parce que les arguments avancés sont assez généraux, et que le traitement de partitions plus complexes peut tirer profit des concepts présentés.

BIBLIOGRAPHIE

- [Andronico et al. 82] A. Andronico, A. Ciampa, On automatic pattern recognition and acquisition of printed music, *International Computer Music Conference (ICMC)*, pp. 245-278, Venice, Italy, 1982.
- [Armand 93] J.-P. Armand, Musical score recognition : a hierarchical and recursive approach, *2nd International Conference on Document Analysis and Recognition (ICDAR)*, pp. 906-909, 1993.
- [Bainbridge 96] D. Bainbridge, Optical music recognition: a generalized approach, *Second New Zealand Computer Science Graduate Conference*, 1996.
- [Bainbridge, Bell 96] D. Bainbridge, T.C. Bell, An extensible optical music recognition system, *Nineteenth Australasian Computer Science Conference*, pp. 308-317, Melbourne, Australia, 1996.
- [Bainbridge 97] D. Bainbridge, *Extensible optical music recognition*, PhD thesis, Department of Computer Science, University of Canterbury, New Zealand, 1997.
- [Bainbridge, Bell 97] D. Bainbridge, T.C. Bell, Dealing with superimposed objects in optical music recognition, *Sixth International Conference on Image Processing and its Application*, pp. 756-760, Dublin, Ireland, 1997.
- [Bainbridge, Carter 97] D. Bainbridge, N. Carter, Automatic reading of music notation, *Handbook on Optical Character Recognition and Document Image Analysis*, Bunke, World Scientific, pp. 583-603, 1997.
- [Bainbridge, Wijaya 99] D. Bainbridge, K. Wijaya, Bulk processing of optically scanned music. *Seventh International Conference on Image Processing and its Applications*, pp. 474-478, Manchester, UK, 1999.
- [Bainbridge, Bell 03] D. Bainbridge, T. Bell, A music notation construction engine for optical music recognition, *Software Practice and Experience (SP&E)* 33(2), pp. 173-200, 2003.
- [Baumann, 95] S. Baumann, A simplified attributed graph grammar for high level music recognition, *Third International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1080-1083, Montréal, Canada, 1995.
- [Baumann, Dengel 92] S. Baumann, A. Dengel, Transforming printed piano music into midi, *IAPR Workshop on SSPR*, Bern, Switzerland, 1992.
- [Baumann, Tombre 95] S. Baumann, K. Tombre, Report of the Line Drawing and Music Recognition Working Group, *Document Analysis Systems (DAS'94)*, Eds. A. L. Spitz and A. Dengel, pp. 462-464. World Scientific, 1995.

Bibliographie

- [Bellini et al. 01] P. Bellini, I. Bruno, P. Nesi, Optical Music Sheet Segmentation, *International Conference on WEB Delivering of Music*, pp. 183-190, Florence, Italy, 2001.
- [Bloch 96] I. Bloch, Information combination operators for data fusion : a comparative review with classification, *IEEE Transactions on Systems, Man, and Cybernetics*, 26(1) pp. 52-67, 1996.
- [Bloch 00] I. Bloch, Fusion of numerical and structural image information in medical imaging in the framework of fuzzy sets, *P. Szczepaniak et al. (Eds.), Fuzzy Systems in Medicine, Series Studies in Fuzziness and Soft Computing*, pp. 429-447, Springer, Berlin, 2000.
- [Bloch 03] I. Bloch, Théorie des ensembles flous et des possibilités, *Fusion d'informations en traitement du signal et des images*, sous la direction de I. Bloch, Ed. Hermes Science, pp. 149-217, 2003.
- [Bloch, Maître 97] I. Bloch, H. Maître, Fusion of image information under imprecision, *B. Bouchon-Meunier, (Ed.), Aggregation and Fusion of Imperfect Information, Series Studies in Fuzziness*, pp. 189-213, Physica Verlag, Springer, 1997.
- [Blostein, Baird 92] D. Blostein, H. Baird, A critical survey of music image analysis, *H.S. Baird et al., (Eds), Structured Document Image Analysis*, pp. 405-434, Springer, Berlin, 1992.
- [Blostein, Haken 99] D. Blostein, L. Haken, Using diagram generation software to improve diagram recognition : a case study of music notation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21 (11), pp. 1121-1135, 1999.
- [Carter 89] N. P. Carter, *Automatic recognition of printed music in the context of electronic publishing*, Phd Thesis, University of Surrey, UK, 1989.
- [Carter, Bacon 92] N. Carter, R. Bacon, Automatic recognition of printed music. *H.S. Baird, H. Bunke, K. Yamamoto, (Eds.), Structured Document Image Analysis*, pp. 456-465, Springer, Berlin, 1992.
- [Clarke et al. 88] A. T. Clarke, B. M. Brown and M. P. Thorne, Inexpensive optical character recognition of music notation: a new alternative for publishers, *Computers in Music Research Conference*, pp 84-87, Lancaster, UK, 1988.
- [Coüasnon 91] B. Coüasnon, *Réseaux de neurones appliqués à la reconnaissance de partitions musicales*, rapport de DEA, Irisa, Université de Rennes I, 1991.
- [Coüasnon 96a] B. Coüasnon, Formalisation grammaticale de la connaissance a priori pour l'analyse de documents : application aux partitions d'orchestre, *Reconnaissance des Formes et Intelligence Artificielle (RFIA'96)*, pp. 465-474, Rennes, France, 1996.
- [Coüasnon 96b] B. Coüasnon, *Segmentation et reconnaissance de documents guidées par le connaissance a priori : application aux partitions musicales*, Thèse de l'Université de Rennes 1, 1996.
- [Coüasnon, Camillerapp 94] B. Couasnon, J. Camillerapp, Using grammars to segment and recognize music scores, *International Association for Pattern Recognition Workshop on Document Analysis Systems*, pp. 15-27, Kaiserslautern, Germany, 1994.
- [Coüasnon, Rétif 95] B. Coüasnon, B. Rétif, Using a Grammar for a Reliable Full Score Recognition System, *International Computer Music Conference*, pp. 187-194, Banff, Canada, 1995.
- [Danhauser 96] A. Danhauser, *Théorie de la Musique*, Ed. Lemoine, 1996.
- [Droettboom et al. 02] M. Droettboom, I. Fujinaga, K. MacMillan, Optical music interpretation, *Statistical, Structural and Syntactic Pattern Recognition Conference*, pp. 362-370, 2002.

- [Dubois, Prade 80] D. Dubois and H. Prade, *Fuzzy sets and systems : theory and applications*, Academic Press, New-York, 1980.
- [Dubois, Prade 01] D. Dubois, H. Prade, La problématique scientifique du traitement de l'information, *Information-Interaction-Intelligence*, vol. 1, N°2, 2001.
- [Dubois et al. 99] D. Dubois, H. Prade, R. Yager, Merging Fuzzy Information, *J.C. Bezdek, D. Dubois and H. Prade (Eds), Handbook of Fuzzy Sets Series, Approximate Reasoning and Information Systems*, Chapter 6, Kluwer, 1999.
- [Fahmy, Blostein 91] H. Fahmy, D. Blostein, A graph grammar for high level recognition of music notation, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 70-78, 1991.
- [Fahmy, Blostein 98] H. Fahmy, D. Blostein, A graph-rewriting paradigm for discrete relaxation : application to sheet-music recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 12, No. 6, pp. 763-799, 1998.
- [Ferrand et al. 99] M. Ferrand, J.A. Leite, A. Cardoso, Improving optical music recognition by means of abductive constraint logic programming, *EPIA*, pp. 342-356, 1999.
- [Fletcher, Kasturi 88] L. A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed text/graphics images, *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)* 10(6), pp. 910-918, 1988.
- [Fotinea et al. 00] S. E. Fotinea, G. Giakoupis, A. Liveris, S. Bakamidis, G. Carayannis, An optical notation recognition system for printed music based on template matching and high level reasoning, *6th Recherche d'Informations Assistée par Ordinateur (RIA0'00)*, Paris, 2000.
- [Fujinaga 88] I. Fujinaga. *Optical Music Recognition using projections*, Master thesis, McGill University, Faculty of Music, Montreal, Canada, 1988.
- [Fujinaga et al. 92] I. Fujinaga, B. Alphonse, B. Pennycook, G. Diener, Interactive optical music recognition. *International Computer Music Conference*, pp. 117-120, 1992.
- [Fujinaga 95] I. Fujinaga, Exemplar-based learning in adaptive optical music recognition system, *International Computer Music Conference*, pp. 55-56, 1995.
- [Fujinaga 97] I. Fujinaga, *Adaptive optical music recognition*, Ph.D. Dissertation, McGill University, 1997.
- [Fujinaga et al. 98] I. Fujinaga, S. Moore, D. S. Sullivan, Implementation of exemplar-based learning model for music cognition, *International Conference on Music Perception and Cognition*, pp. 171-179, 1998.
- [Genfang, Shunren 03] C. Genfang, X. Shunren, The study and prototype system of printed music recognition, *International Conference on Neural Networks and Signal Processing*, pp. 1002-1008, China, 2003.
- [Hori et al. 99] T. Hori, S. Wada, H. Tai, S.Y. Kung, Automatic music score recognition/play system based on decision based neural network, *IEEE Signal Processing Society, Third Workshop on Multimedia Signal Processing*, pp. 183-184, Copenhagen, Denmark, 1999.
- [Interactive Music Network] http://www.interactivemusicnetwork.org/wg_imaging/documents.html
- [Kassler 72] M. Kassler, Optical character recognition of printed music: a review of two dissertations, *Perspectives of New Music*, Vol.11 n°2, pp. 250-254, 1972.

Bibliographie

- [Kato, Inokuchi 90] H. Kato, S. Inokuchi, The recognition system of printed piano using musical knowledge and constraints, *IAPR Workshop on Syntactic and Structural Pattern Recognition*, pp. 231-248, Murray Hill NJ, 1990.
- [Kato, Inokuchi 92] H. Kato, S. Inokuchi, A recognition system for printed piano using musical knowledge and constraints, *H.S. Baird et al., (Eds), Structured Document Image Analysis*, pp. 231-248, Springer, Berlin, 1992.
- [Krishnapuram, Keller 92] R. Krishnapuram, J. M. Keller, Fuzzy set theoretic approach to computer vision : an overview, *Int. Conf. on Fuzzy Systems*, pp 135-142, San Diego, CA, 1992.
- [Lutz 04] M. Lutz, The Maine music box: a pilot project to create a digital music library, *Library Hi Tech*, Vol. 22, n°3, pp. 283-294, 2004.
- [Mahoney 82] J. V. Mahoney, *Automatic analysis of musical score images*, B. S. thesis, Dept. of Computer Science and Engineering, Massachusetts Institute of Technology, 1982.
- [Marinai, Nesi 99] S. Marinai, P. Nesi, Projection based segmentation of musical sheets, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 515-518, Bangalore India, 1999.
- [Martin 89] P. Martin, Reconnaissance de partitions musicales et réseaux de neurones : une étude, *Septième congrès Reconnaissance des Formes et Intelligence Artificielle (RFIA)*, pp. 217-226, Paris, France, 1989.
- [Martin 92] P. Martin, *Réseau de neurones artificiels : application à la reconnaissance optique de partitions musicales*, Thèse IMAG, Grenoble, 1992.
- [Martin, Bellissant 91] P. Martin, C. Bellissant, Low-level analysis of music drawing images, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 417-425, Saint-Malo, France, 1991.
- [Matsushima et al. 85] T. Matsushima, I. Sonomoto, T. Harada, K. Kanamori, S. Ohteru, Automated high speed recognition of printed music (WABOT-2 vision system), *International Conference on Advanced Robotics (ICAR)*, pp. 477- 482, Shiba Koen Minato-ku, Tokyo, 1985.
- [McPherson, Bainbridge 01] J.R. McPherson, D. Bainbridge, Coordinating knowledge within an optical music recognition system, *The Fourth New Zealand Computer Science Research Students' Conference*, pp. 50-58, Christchurch, NZ, 2001.
- [McPherson 02] J. R. McPherson, Introducing feedback into an optical music recognition system, *Third International Conference on Music Information Retrieval*, Paris, France, 2002.
- [Miyao, Nakano 95] H. Miyao, Y. Nakano, Head and stem extraction from printed music scores using a neural network approach, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 1074-1079, Montreal, Canada, 1995.
- [Miyao 02] H. Miyao, Stave extraction for printed music scores, *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, pp. 562-568, Manchester, UK, 2002.
- [Modayur 91] B. R. Modayur, Restricted domain music score recognition using mathematical morphology, *International Conference on Symbolic and Logical Computing*, Madison, S. Dakota, 1991.
- [Modayur 96] B. Modayur, Music score recognition – A selective attention approach using mathematical morphology, *Technical report*, Electrical Engineering Dept., University of Washington, Seattle, 1996.

- [Musitek] Musitek, Music Imaging Technologies, <http://www.musitek.com/>
- [Ng, Boyle 92] K. C. Ng, R. D. Boyle, Segmentation of Music Primitives, *D.C. Hogg and R.D. Boyle (eds), Proceedings of the British Machine Vision Conference (BMVC'92)*, Leeds, UK, pp. 472–480. Springer-Verlag London.
- [Ng et al. 95] K. C. Ng, R. D. Boyle, D. Cooper, Low and high level approaches to optical music score recognition, *IEE Colloquium on Document Image Processing and Multimedia Environment*, pp. 3/1-3/6, 1995.
- [Ng, Boyle 96] K. C. Ng, R. D. Boyle, Recognition and reconstruction of primitives in music scores, *Image and Vision Computing 14(1)*, pp. 39-46, 1996.
- [Ng et al. 04] K. C. Ng, J. Barthelemy, B. Ong, I. Bruno, P. Nesi, *The Interactive-Music Network, DE4.7.1, CIMS: Coding Images of Music Sheets*, Section 7 OMR evaluation, http://www.interactivemusicnetwork.org/wg_imaging/upload/musicnetwork-de4-7-1-coding-images-of-music-v2-8-20040208.pdf
- [O³MR] <http://www.dsi.unifi.it/%7Ehpcn/wwwomr/le.html>
- [PhotoScore] <http://www.neuratron.com/photoscore.htm>
- [Poulain d'Andecy et al. 94] V. Poulain d'Andecy, J. Camillerapp, I. Leplumey, Kalman filtering for segment detection : application to music scores analysis, *ICPR, 12th Int. Conf. on Pattern Recognition (IAPR)*, pp. 301-305, Jerusalem, Israel, 1994.
- [Poulain d'Andecy et al. 95] V. Poulain d'Andecy, J. Camillerapp, I. Leplumey, Analyse de partitions musicales, *L'Ecrit et le Document, Traitement du Signal, Vol. 12, n°6*, pp. 653-661, 1995.
- [Prerau 70] D. S. Prerau, *Computer pattern recognition of standard engraved music notation*, PhD thesis, Massachusetts Institute of Technology, 1970.
- [Pruslin 66] D. Pruslin, *Automatic recognition of sheet music*, PhD thesis, Massachusetts Institute of Technology, 1966.
- [Ramel et al. 94] J.Y. Ramel, N. Vincent, H. Emptoz, Reconnaissance de partitions musicales, *Colloque National sur l'Ecrit et le Document (CNED)*, pp. 325-334, 1994.
- [Randriamahefa et al. 93] R. Randriamahefa, J. P. Cocquerez, C. Fluhr, F. Pépin, S. Philipp, Printed Music Recognition, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 898-901, Tsukuba Science City, Japan, 1993.
- [Reed, Parker 96] K. T. Reed, J. R. Parker, Automatic computer recognition of printed music, *International Conference on Pattern Recognition (ICPR)*, pp. 803-807, 1996.
- [Roach, Tatem 88] J. W. Roach, J.E. Tatem, Using domain knowledge in low-level visual processing to interpret handwritten music : an experiment, *Pattern Recognition*, Vol. 21, N°1, pp. 33-44, 1988.
- [Sicard 92] E. Sicard, An efficient method for the recognition of printed music, *11th IAPR International Conference on Pattern Recognition*, Vol. III ,pp. 573-576, Netherlands,1992.
- [SmartScore 06] SmartScore 5.0 Pro Demo (2006) <http://www.musitek.com>, 2006
- [Stückelberg, Doerman 99] M.V. Stückelberg, D. Doermann, On musical score recognition using probabilistic reasoning, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, pp. 115-118, Bangalore, India,1999.

Bibliographie

- [Stückelberg et al. 97] M.V. Stückelberg, C. Pellegrini, M. Hilaro. An architecture for musical score recognition using high-level domain knowledge, *Int. Conf. on Document Analysis and Recognition (ICDAR)*, Vol. 2, pp. 813-818, 1997.
- [Sayeed Choudhury et al. 00] G. Sayeed Choudhury, T. DiLauro, M. Droettboom, I. Fujinaga, B. Harrington, K. MacMillan, Optical music recognition system within a large-scale digitization project, *International Conference on Music Information Retrieval (ICMC)*, 2000.
- [Sayeed Choudhury et al. 01] G. Sayeed Choudhury, T. DiLauro, M. Droettboom, I. Fujinaga, K. MacMillan, Strike up the score, deriving searchable and playable digital formats from sheet music, *D-Lib Magazine* 7 (2), 2001.
- [Su et al. 01] M. C. Su, C. Y. Tew, H. H. Chen, Musical symbol recognition using SOM-based fuzzy systems, *International Fuzzy System Association Conference (IFSA/NAFIPS)*, Vol. 4, pp. 2150-2153, 2001.
- [Watkins 96] G. Watkins, The use of fuzzy graph grammar for recognising noisy two-dimensional images, *North American Fuzzy Information Processing Society Conference (NAFIPS)*, pp. 415-419, 1996.
- [Wijaya, Bainbridge, 99] K. Wijaya, D. Bainbridge, Staff line restoration, *Seventh International Conference on Image Processing and Its Applications*, pp. 760-764, Manchester, U.K. 1999
- [Wong, Choi 94] Y. S. Wong, A.K.O. Choi, A two-level model-based object recognition technique, *Int. Symposium on Speech, Image Processing and Neural Networks*, pp. 319-322, 1994.

PUBLICATIONS

Publications relatives à la thèse

F. Rossant, Une méthode globale pour la reconnaissance de partitions musicales, *Gretsi 2001*, Vol. 2, pp. 95-98, Toulouse, France, 2001

F. Rossant, I. Bloch, Reconnaissance de partitions musicales par modélisation floue et intégration de règles musicales, *Gretsi 2001*, Vol. 2, pp. 99-102, Toulouse, France, 2001

F. Rossant, A global method for music symbol recognition in typeset music sheets, *Pattern Recognition Letters*, Vol. 23/10, pp. 1129-1141, 2002.

F. Rossant, I. Bloch, Modélisation floue pour la reconnaissance de partitions musicales, *Logique Floue et ses Applications (LFA)*, pp. 253-260, Montpellier, France, 2003

F. Rossant, I. Bloch, A fuzzy model for optical recognition of musical scores, *Fuzzy Sets and Systems*, Vol. 14, pp. 165-201, 2004

F. Rossant, I. Bloch, Amélioration de la reconnaissance de partitions musicales par modélisation floue et indication des erreurs possibles, *Gretsi 2005*, pp. 937-940, Louvain-la Neuve, Belgique, 2005.

F. Rossant, I. Bloch, Optical music recognition based on a fuzzy modeling of symbol classes and music writing rules, *International Conference on Image Processing (ICIP)*, Vol. 2, pp. 538-541, Genova, Italy, 2005.

F. Rossant, I. Bloch, Robust and adaptive OMR system including fuzzy modeling, fusion of musical rules, and possible error detection, *EURASIP Journal of Applied Processing*, accepté en Août 2006.

Autres publications

E. Rydgren, A. Amara, F. Amiel, T. Ea, F. Rossant, Iris features extraction using wavelet packets, *International Conference on Image Processing (ICIP)*, Vol. 2, pp. 861-864, Singapor, 2004.

F. Rossant, M. Torres Eslava, T. Ea, F. Amiel, A. Amara, Iris identification and robustness evaluation of a wavelet packets based algorithm, *International Conference on Image Processing (ICIP)*, Vol. 3, pp. 257-260, Genova, Italy, 2005.

F. Rossant, M. Torres Eslava, T. Ea, F. Amiel, A. Amara, Identification par analyse en paquets d'ondelettes de l'iris et tests de robustesse, *Gretsi 2005*, pp. 9-12, Louvain-la-Neuve, Belgique, 2005.

Publications

T. Ea, A. Valentian, F. Rossant, F. Amiel, A. Amara, Algorithm implementation for iris identification, *48th Midwest Symposium on Circuits and Systems (MWSCAS)*, Vol.2., pp. 1207-1210, Cincinnati, Ohio, 2005.

T. Ea, A. Valentian, F. Amiel, F. Rossant, A. Amara, Implementation on SoPC of algorithms dedicated to iris identification, *Conference On Design of Circuits and Integrated Systems (DCIS)*, Lisboa, Portugal, 2005

T. Ea, F. Amiel, A. Michalowska, F. Rossant, A. Amara, Erosion and dilatation implementation for Iris recognition system using different techniques on SoPC, DCIS'06, Barcelona, Spain, 2006

T. Ea, F. Amiel, A. Michalowska, F. Rossant, A. Amara, Contribution of Custom Instructions on SoPC for iris recognition application, ICECS 06, Nice, France, 2006.


















M. Terré, F. Rossant, B. Mikovicova, L. Féty, CDMA modem, Texas Instruments Developer Conference, Huston, 2002.

B. Mikovicova, F. Rossant, J.Y. Delabbaye, Joint phase carrier and information bits estimation, *European DSP Education and Research Symposium (EDERS)*, Birmingham, 2004

ANNEXE

Dans cette annexe sont présentés quelques exemples de reconnaissance obtenus sur notre base d'images. Les résultats sont comparés avec ceux réalisés par SmartScore [SmartScore 06].

Légende (images produites par notre méthode) :

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---------------------------|---|--|
|  | Symbole manquant |  | Confusion |  | Symbole ajouté |  | Erreur de durée |  | Erreur de hauteur | | |
| Durée des notes | | |  | Mesure potentiellement fausse (Durée incorrecte) | | | | | | | |
|  |  |  |  |  |  |  |  |  | Erreur potentielle |  | Groupe cohérent |
| 1/2 | 1/3 | 2/7 | 1/4 | 1/5 | 1/6 | 1/8 | 1/16 | | |  | Groupe potentiellement faux (Erreur de durée) |

Exemple 1 :

The musical score for Example 1 consists of six staves of music. The notation includes various rhythmic values and dynamic markings. Key features include:

- Staff 1: Starts with a *pp* marking and a *tr* (trill) marking.
- Staff 2: Measure 35 is circled. Dynamic marking *mp* is present.
- Staff 3: Measure 40 is circled. Dynamic markings *pp* and *mf* are present.
- Staff 4: Measure 45 is circled.
- Staff 5: Measure 50 is circled.
- Staff 6: Measure 55 is circled. Dynamic marking *f* is present. The piece ends with a first ending bracket labeled '1'.

Notre méthode :

SmartScore :

Exemple 2 :

Musical score for Example 2, measures 5 to 16. The score is written in treble clef with a key signature of three sharps (F#, C#, G#). It features various musical notations including triplets, trills (tr), and slurs. The measures are numbered 5, 7, 9, 11, 13, and 15.

Notre méthode :

Musical score for 'Notre méthode', measures 5 to 16. This version includes fingerings (blue dots) and other annotations (green and red lines) to guide the performer. A red arrow points to a double sharp symbol (F#) in measure 18, with the text "[#] double dièse" written in red. The score is written in treble clef with a key signature of three sharps (F#, C#, G#). The measures are numbered 5, 7, 9, 11, 13, and 15.

SmartScore : option "triplets" non cochée ou cochée

The image displays two musical staves side-by-side, comparing the output of SmartScore with the 'Triplets' option unchecked (top) and checked (bottom). The top staff has a checkbox labeled 'Triplets' that is empty. The bottom staff has a checkbox labeled 'Triplets' that is checked. The musical notation is identical, but the interpretation of triplet markings differs. In the top staff, annotations include '[+] point' (red arrow pointing down), '[-] 4 Triplets' (orange box with four orange arrows pointing up), '[C] quart soupir -> dièse' (red arrow pointing down), '[-] appoggiature' (red arrow pointing down), '[-] double dièse' (red arrow pointing down), '[+] blanche' (red arrow pointing down), '[-] appoggiature' (red arrow pointing down), '[-] appoggiature' (red arrow pointing down), '[-] appoggiature' (red arrow pointing down), and '[-] 13 Triplets' (orange box with thirteen orange arrows pointing up). In the bottom staff, annotations include '[+] point' (red arrow pointing down), '[+] Triolet' (red arrow pointing up to a bracketed triplet), '[-] quart soupir' (red arrow pointing down), and '[+] Triolet' (red arrow pointing up to a bracketed triplet). The musical notation includes various notes, rests, and dynamic markings like 'b' and 'f'.

Exemple 3 :

Herausgegeben von Kurt Soldan

Andante

4

7

10 *tr* Presto

15

19

28

Notre méthode :

Herausgegeben von Kurt Soldan

Andante

4

7

10 *tr* Presto

15

19

28

SmartScore :

A musical score for a piece in C major, marked 'Andante'. The score consists of 25 measures across six staves. The notes are highlighted in pink. Annotations include:

- Measure 1: **[+] point** (red arrow pointing down), **[C] dièse** (red arrow pointing down).
- Measure 4: **[-] point** (red arrow pointing down).
- Measure 5: **Durée** (red arrow pointing up).
- Measure 6: **[-] bémol** (red arrow pointing down).
- Measure 10: **[-] 2 noires** (two red arrows pointing down), **[+] ronde** (red arrow pointing down).
- Measure 11: **Durée** (red arrow pointing down).
- Measure 12: **Durée** (red arrow pointing down).
- Measure 19: **[-] barre** (red arrow pointing down), **[C] bécarré** (red arrow pointing down).

Other markings include 'Andante', 'res G', and '798x490'.

Exemple 4 :

A musical score for a piece in C major, consisting of 25 measures across five staves. The notes are in black. Measure markers are placed in boxes above the staves:

- Measure 10
- Measure 15
- Measure 20
- Measure 25

The score includes various musical notations such as slurs, accents, and a dynamic marking 'p' at the end.

Notre méthode :

The image shows five staves of musical notation. The first staff is labeled with a box containing the number '10'. The second staff is labeled with a box containing '15'. The third staff is labeled with a box containing '20'. The fifth staff is labeled with a box containing '25'. The notation includes various note values, rests, and accidentals. There are several annotations: orange boxes around specific notes, pink arrows pointing to notes, and green horizontal lines drawn below the staves. Some notes have a 'v' (accents) above them.

SmartScore :

[-] 5 bécarres

The image shows five staves of musical notation, numbered 9 through 28. Above the first staff, the text '[-] 5 bécarres' is written in red. Five red arrows point downwards to specific notes on the staves: the first arrow points to the note on staff 12, the second to staff 14, the third to staff 18, the fourth to staff 26, and the fifth to staff 28. The notation includes various note values, rests, and accidentals.

Exemple 5 :

*) Die kleinen dynamischen Zeichen sind Vor- | *) Les petits signes dynamiques constituent des | *) The dynamics in small type are the editor's

Notre méthode :

4 confusions dièse / bécarré sur 5 bien indiquées.

SmartScore

Notre méthode après apprentissage :

Reconnaissance parfaite.

33 *pp* *p* *pp*

37 *p* *pp* *mf*

40

44 *f* Tutti

Exemple 6 :

Très modéré

mf

p


Retenu

p

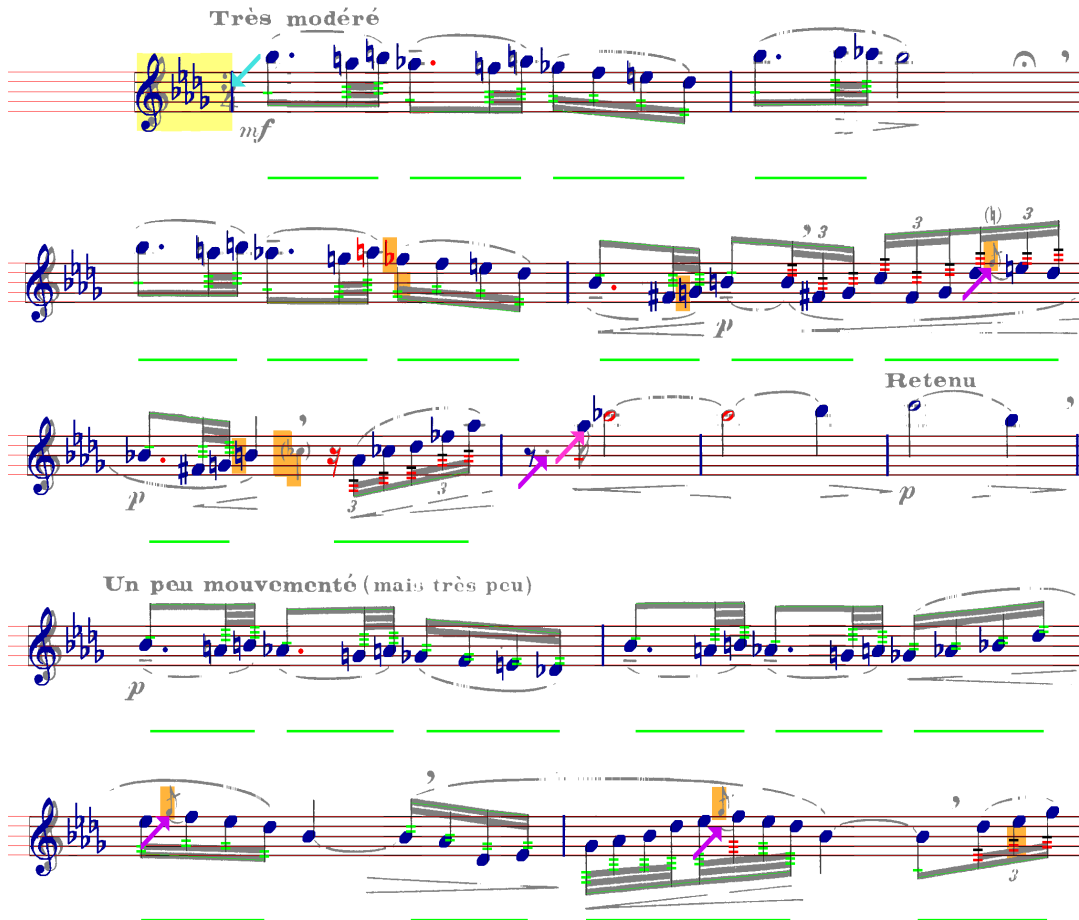
Un peu mouvementé (mais très peu)

p

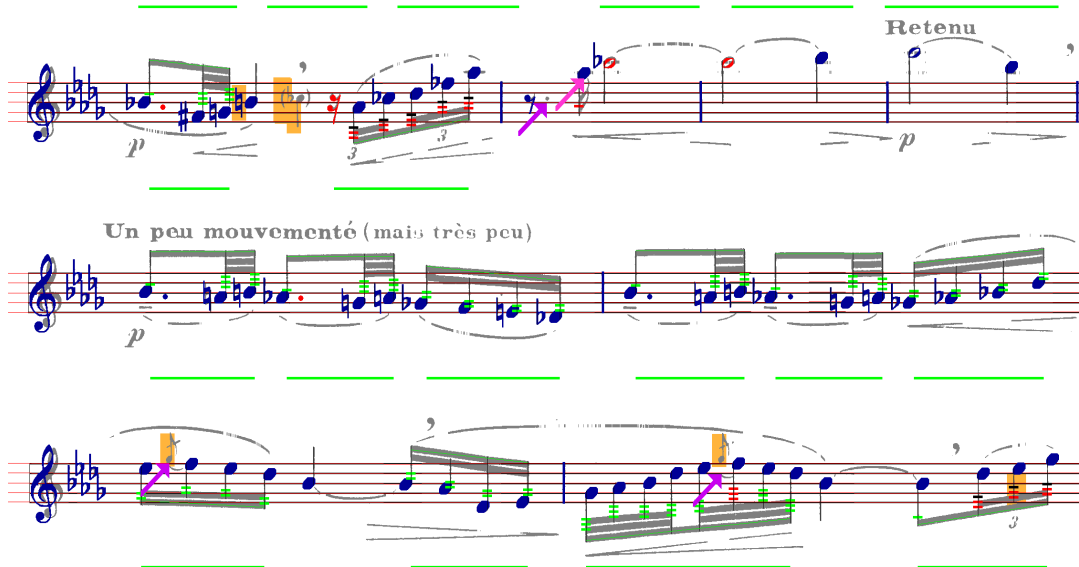
Notre méthode : option "triolet dans groupes" cochée

Cette option permet d'inclure dans l'analyse des modèles rythmiques rares comprenant un triole. Les groupes "croche / triolet de doubles croches"  sont ainsi bien interprétés.

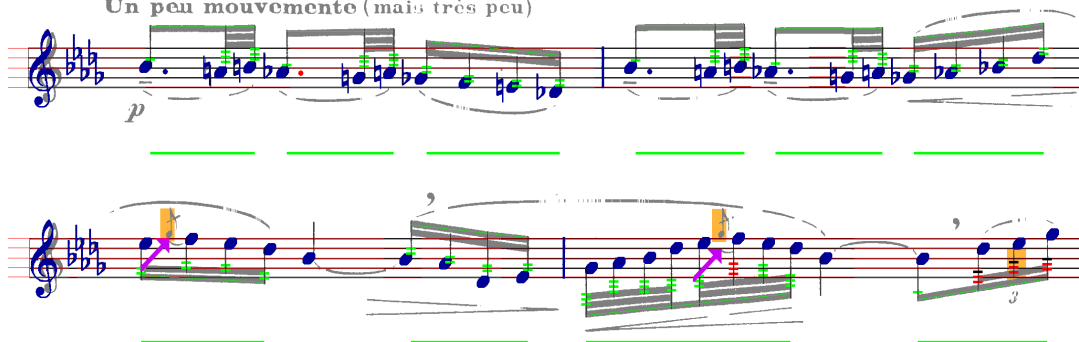
Très modéré



Retenu



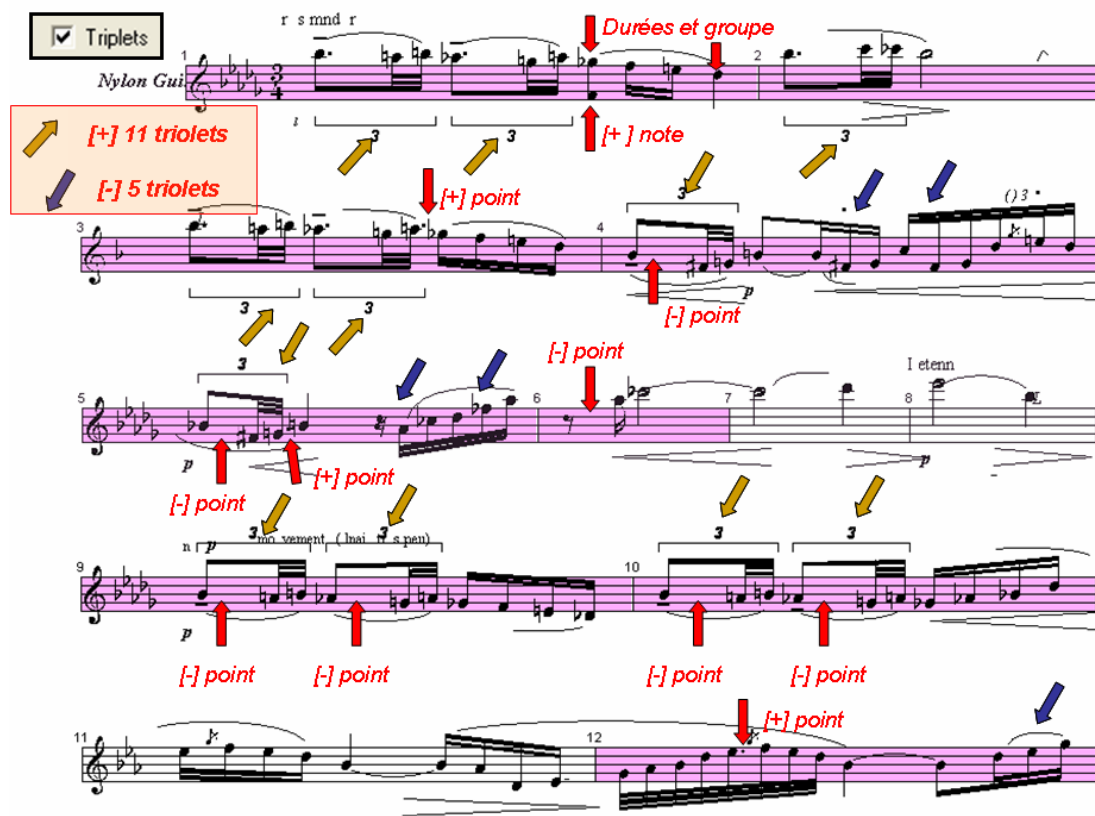
Un peu mouvementé (mais très peu)



SmartScore :

Triplets

Nylon Gui



[+] 11 triolets

[-] 5 triolets

[+] Durées et groupe

[+] note

[+] point

[-] point

[-] point

[-] point

[-] point

[-] point

[-] point

[+] point

Exemple 7

% Lento dolcissimo
p
cresc. - - - - - *f*
FIN
 Più lento $\text{♩} = 76$
p espressivo
mf

Notre méthode :

Cette partition est l'une des partitions de la base les moins bien analysées. Taux de reconnaissance (sur toute la page) : 95.9%, avec 0.64% de symboles ajoutés ; durées : 97.0% ; hauteurs : 95.1%. Les erreurs sur les silences induisent des erreurs de durée. En introduisant d'autres critères de reconnaissance (paragraphe 8.3.2), on peut espérer avoir une meilleure discrimination et améliorer l'ensemble des résultats.

% Lento dolcissimo
p
cresc. - - - - - *f*
FIN
 Più lento $\text{♩} = 76$
p espressivo
mf

SmartScore :

[+] blanche

[-] 17 blanches / blanches pointées

[+] 2 silences

[-] quart soupir

[+] bécarre

[-] bémol

Groupe et durées

Exemple 8 :

espress.

mf

p

f

ff

p dolce

rit. a tempo

rull.

ppp

Notre méthode :

Certains rythmes ne sont pas encore modélisés et ne peuvent donc pas être reconnus : les doubles points, les triolets irréguliers. Les quintolets, les sextolets et les septolets sont en revanche très bien analysés, même lorsqu'ils ne sont pas explicitement indiqués.

A musical score in G major, 4/4 time, spanning measures 13 to 77. The score is annotated with various musical symbols and performance instructions. Key annotations include:

- Measure 13: Triplet of eighth notes, marked *espress*.
- Measure 17: Triplet of eighth notes, marked *f*.
- Measure 40: Triplet of eighth notes, marked *f*.
- Measure 45: Triplet of eighth notes, marked *p*.
- Measure 51: Triplet of eighth notes, marked *mf*.
- Measure 60: Triplet of eighth notes, marked *p*, with a *rit. a tempo* instruction.
- Measure 69: Triplet of eighth notes, marked *ff*.
- Measure 77: Triplet of eighth notes, marked *p*.

 Other annotations include dynamics like *p dolce*, *pp*, and *ppp*, and performance instructions like *rall* and *V* (breath mark).

SmartScore :

SmartScore analysis of the musical score, showing detected triplets and other annotations. A checkbox labeled "Triplets" is checked. The score is annotated with:

- Yellow arrows pointing to detected triplets in measures 14, 15, 16, 17, 18, 19, 20, 41, 42, 43, 44, 45, 46, 47, 48, 49, 51, 52, 53, 54, 55, 56, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79.
- Red arrows pointing to detected silences in measures 40 and 53, labeled "[+] Silence".
- Red arrows pointing to detected appoggiature in measures 53 and 70, labeled "[+] Appoggiature".
- Red arrows pointing to detected dièses in measures 53 and 77, labeled "[+] Dièse".

 Dynamics and performance instructions are also visible, such as *p*, *mf*, *f*, *ff*, *p dolce*, *pp*, and *ppp*.

Exemple 9 :

Musical score for Example 9, measures 79-159. The score is written in treble clef with a key signature of one sharp (F#). The dynamics and markings are as follows:

- Measure 79: *pp*, *cresc.*, *sf*, *sf*, *cresc.*, *sfp*
- Measure 86: *cresc.*, *f*, *p*
- Measure 90: *f*, *pp*
- Measure 96: *ff*, *p*
- Measure 104: *f*, *sf*, *tr*
- Measure 110: *sf*, *tr*
- Measure 116: *ff*, *p*, *cresc.*
- Measure 124: *p*
- Measure 131: *cresc.*, *ff*, *sf*
- Measure 137: *sf*, *pp*, *pp*
- Measure 144: *p*, *f*
- Measure 150: *ff*, *sf*, *sf*, *ff*, *ff*
- Measure 159: *sfp*, *f*

Notre méthode :

Taux de reconnaissance (sur 2 pages) : 99.3% ; durées : 99.3%

79 *pp* *cresc.* *sf* *sf* *pp* *cresc.* *sfp*

86 *cresc.* *f* *p*

90 *mp*

96 *ff* *p*

104 *f* *sf*

110 *sf*

116 *ff* *p* *cresc.*

124 *p*

131 *cresc.* *ff* *sf*

137 *sf* *pp* *pp*

144 *p* *f*

150 *ff* *sf* *ff* *ff*

150 *sf* *f* B 55*

SmartScore :

Taux de reconnaissance (sur 2 pages) : 92.0% ; durées : 85.3%

A musical score for guitar, consisting of 11 staves of music. The score is written in treble clef with a key signature of one sharp (F#). The music is annotated with various symbols:

- Red arrows:** Point to specific notes or groups of notes across all staves.
- Purple arrows:** Point to specific notes, often in the lower register or during complex passages.
- Pink arrows:** Point to specific notes, often in the lower register.
- Red double arrows:** Point to specific notes, often in the lower register.
- Dynamic markings:** *pp*, *f*, *ff*, *p*, *ppp*, *fp*, *f*.
- Tempo/Performance markings:** $1/27 \times 230$.
- Shaded areas:** A pink shaded area covers measures 14-15 on the second staff, and another pink shaded area covers measures 76-77 on the tenth staff.