HAL
open science

Analysis of pedestrian movements and gestures using an on-board camera to predict their intentions

Joseph Gesnouin

To cite this version:

Joseph Gesnouin. Analysis of pedestrian movements and gestures using an on-board camera to predict their intentions. Robotics [cs.RO]. Université Paris sciences et lettres, 2022. English. NNT : 2022UPSLM023 . tel-03813520

HAL Id: tel-03813520

https://pastel.hal.science/tel-03813520v1

Submitted on 13 Oct 2022

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.

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à MINES ParisTech

Analysis of pedestrian movements and gestures using an on-board camera to predict their intentions

Analyse des mouvements et gestes des piétons via caméra embarquée pour la prédiction de leurs intentions

Soutenue par

Joseph Gesnouin

Le 27 septembre 2022

École doctorale n°621

Ingénierie des Systèmes, Matériaux, Mécanique, Energétique

Spécialité

Informatique temps réel, robotique et automatique.

Composition du jury :

Alexandre Alahi Professeur assistant, EPFL	*Rapporteur*
Catherine Achard Professeure, Sorbonne Université	*Président du jury / Rapporteur*
Miguel Angel Sotelo Professeur, University of Alcalá	*Examinateur*
Fabien Moutarde Professeur, Mines ParisTech	*Directeur de thèse*
Steve Pechberti Ingénieur de recherche, Vedecom	*Examinateur*
Bogdan Stanciulescu Maître de conférences, Mines ParisTech	*Examinateur*

MINES ParisTech | PSL

Abstract

The autonomous vehicle (AV) is a major challenge for the mobility of tomorrow. Progress is being made every day to achieve it; however, many problems remain to be solved to achieve a safe outcome for the most vulnerable road users (VRUs). One of the major challenges faced by AVs is the ability to efficiently drive in urban environments. Such a task requires interactions between autonomous vehicles and VRUs to resolve traffic ambiguities. In order to interact with VRUs, AVs must be able to understand their intentions and predict their incoming actions.

In this dissertation, our work revolves around machine learning technology as a way to understand and predict human behaviour from visual signals and more specifically pose kinematics. Our goal is to propose an assistance system for the AV that is lightweight, scene-agnostic that could be easily implemented in any embedded device with real-time constraints.

Firstly, in the gesture and action recognition domain, we study and introduce different representations for pose kinematics, based on deep learning models as a way to efficiently leverage their spatial and temporal components while staying in an euclidean grid-space. Secondly, in the autonomous driving domain, we show that it is possible to link the posture, the walking attitude and the future behaviours of the protagonists of a scene without using the contextual information of the scene (zebra crossing, traffic light...). This allowed us to divide by a factor of 20 the inference time of existing approaches for pedestrian intention prediction while keeping the same prediction robustness. Finally, we assess the generalization capabilities of pedestrian crossing predictors and show that the classical train-test sets evaluation for pedestrian crossing prediction, *i.e.*, models being trained and tested on the same dataset, is not sufficient to efficiently compare nor conclude anything about their applicability in a real-world scenario. In order to make the research field more sustainable and representative of the real advances to come, we propose new protocols and metrics based on uncertainty estimates under domain-shift.

Résumé en Français

Le véhicule autonome est un défi majeur pour la mobilité de demain. Des progrès sont réalisés chaque jour pour y parvenir ; cependant, de nombreux problèmes restent à résoudre pour obtenir un résultat sûr pour les usagers de la route les plus vulnérables. L'un des principaux défis auxquels sont confrontés les véhicules autonomes est la capacité à conduire efficacement en milieu urbain. Une telle tâche nécessite la gestion des interactions entre les véhicules et les usagers vulnérables de la route afin de résoudre les ambiguïtés du trafic. Afin d'interagir avec ces usagers, les véhicules doivent être capables de comprendre leurs intentions et de prédire leurs actions à venir.

Dans cette thèse, notre travail s'articule autour de la technologie d'apprentissage automatique comme moyen de comprendre et de prédire le comportement humain à partir de signaux visuels et plus particulièrement de la cinématique de pose. Notre objectif est de proposer un système d'assistance au véhicule qui soit léger, agnostique à la scène et qui puisse être facilement implémenté dans n'importe quel dispositif embarqué avec des contraintes temps réel.

Premièrement, dans le domaine de la reconnaissance de gestes et d'actions, nous étudions et introduisons différentes représentations de la cinématique de pose, basées sur des modèles d'apprentissage profond afin d'exploiter efficacement leurs composantes spatiales et temporelles tout en restant dans un espace euclidien. Deuxièmement, dans le domaine de la conduite autonome, nous montrons qu'il est possible de lier la posture, l'attitude de marche et les comportements futurs des protagonistes d'une scène sans utiliser les informations contextuelles de la scène. Cela nous permet de diviser par un facteur 20 le temps d'inférence des approches existantes pour la prédiction de l'intention des piétons tout en gardant la même robustesse de prédiction. Finalement, nous évaluons la capacité de généralisation des approches de prédiction d'intention de piétons et montrons que le mode d'évaluation classique des approches pour la prédiction de traversée de piétons, n'est pas suffisante pour comparer ni conclure efficacement sur leur applicabilité lors d'un scénario réel. Nous proposons de nouveaux protocoles et de nouvelles mesures basés sur l'estimations d'incertitude afin de rendre le domaine de recherche plus durable et plus représentatif des réelles avancées à venir.

Acknowledgement

Je tiens dans un premier temps à remercier l'entièreté du jury pour avoir accepté de participer au processus d'évaluation du travail effectué ces dernières années. Je remercie plus particulièrement Catherine Achard et Alexandre Alahi pour avoir bien voulu lire et évaluer mon manuscrit de thèse ainsi que pour leurs remarques pertinentes à propos de celui-ci. Je tiens dans un second temps à exprimer ma profonde reconnaissance envers les personnes qui ont encadré mon travail de recherche. Quand on a trop à dire, on oublie comment faire les choses simplement, alors : Steve, Bogdan, Fabien, merci.

Venons-en alors à la longue et *ennuyeuse* énumération des gens qui ont eu un impact non négligeable sur la trajectoire de mon apprentissage du monde de la recherche. Marc, Guillaume B., comment ne pas commencer par vous ? J'ai le sentiment que sans vos retours, mon positionnement n'aurait pas évolué de la sorte au cours de ma thèse, pour cela, merci. Guillaume D., Raphaël R., les idées commencent souvent par un éclat d'intérêt dans les yeux d'un interlocuteur qui furent souvent les vôtres : merci.

Que celui/celle qui n'a pas eu de périodes creuses me jette la première pierre. Il n'y a pas de véritable remède à celles-ci. Lorsque la prédiction d'intention des piétons ne m'inspirait pas, nous allions à "*L'Avenir*"[1] ou bien "*Chez Bichette*". À ce sujet, je tiens à remercier l'ensemble de mes collègues chez Vedecom: Maryem, Alexis, Benoit, Emile, Francky, Lynda, Mohamed, Hugo, Laurent, Yousri, Nicolas, Guilhelm... Je n'oublie pas non plus ceux dont les affiliations industrielles varient régulièrement depuis leur départ de l'institut. Je salue donc ces passionnés de cétologie : Sylvain, Nihed, *Monsieur le président*, Victor, Evie (et re-Marc).

Outre mes escapades versaillaises, j'ai passé la majorité de mon temps dans les caves des Mines avec la plus fine des équipes de thésards : la V026 composée de Camille, Daniel, Jesus, Raphaël C., Thomas et *Monsieur Catastrophe*. Le monde des doctorants ne se résume heureusement pas à son bureau. Je salue les *pièces rapportées* de cette fine équipe : Sami, Sofiane, Sascha, Jules, Louis, Fabio, Bastien, Nathan, Jean-Pierre, Amandine, Hugo, Chloé-Agathe ainsi que l'entièreté des permanents du labo (!). Ruwen Ogien vantait "l'importance de l'odeur des croissants chauds sur la bonté humaine", je n'irai pas jusqu'à attribuer entièrement cette atmosphère conviviale qu'il m'a été possible de connaitre à nos seuls concours de gâteaux, apéros et à leur supposé effet sur nous, mais ceux-ci semblent parfaitement illustrer l'état d'esprit et le cadre dans lequel j'ai été amené à évoluer au sein de ma thèse. Pour cela, merci à tous.

Il parait que l'on perd des amis pendant la thèse, qu'involontairement, on s'éloigne de certaines et certains qui ont pour autant été présents depuis des années. À vous, je dois des excuses pour le piètre ami que j'ai pu être. Quand bien même on ne puisse pas s'excuser soit même, mon auto-flagellation intemporelle en guise de mea culpa. Philéas, Marie O., Nathalie, Malina, merci pour vos conseils, votre aide et votre honnêteté à toute épreuve cette dernière décennie. Yannis, Rayane, bien que la fenêtre de temps ciblée soit moindre que pour les zigotos du dessus, le constat est similaire.

Je tiens finalement à remercier ma famille, tribu hétéroclite en constante expansion : Gesnouin, Tat-

[1] Sémantiquement très proche de mon sujet de recherche.

inclaux, Peres, Morel, Bregeon, Agnedani... Je vous suis reconnaissant d'être à mes côtés pour cet énième rite de passage. À mes géniteurs, vous qui m'avez tout deux apporté les outils pour accéder à cette situation émancipatrice où j'ai le luxe de ne plus être inquiet de ma visibilité sociale, comment ne pas vous être reconnaissant? Merci pour cet apprentissage de la zététique : Feue Pipiole, Feu la mouche, de l'épistémologie et de la danse bretonne. J'en profite pour corroborer les dires de mon géniteur : puissent-ils inspirer le lecteur suffisamment curieux pour arriver jusqu'ici : "La science avec conscience et sans orgueil est le seul outil d'émancipation personnelle en premier lieu, collégiale sinon collective en second lieu"[2].

Pour finir, j'ai une pensée toute particulière envers ma compagne. Je te souhaite la même réussite que celle que je célèbre en écrivant ces lignes aujourd'hui. Marie, bien que cette période de vie soit sans doute révolue, elle n'aurait pas eut la même saveur sans toi. Je suis à la fois curieux de voir où nous mènera la suivante et heureux de savoir que nous y serons *"hypothétiquement"* ensemble pour nous épauler.

Joseph

[2]D'après Rabelais, Élisée Reclus, Primo Levi, Bourdieu, Pascal et Sergueï Iliouchine

Contents

1 Introduction **3**

1.1 Context 4

1.2 Objectives 5

1.3 Publications and communications 6

1.4 Outline 7

2 Human Activity Recognition With Pose-driven Deep Learning Models **9**

2.1 Historical Notes on Human Actions Understanding 10

2.2 Overview of Modern Computer Vision Modalities for Action Recognition 12

2.2.1 RGB videos 13

2.2.2 Depth data 15

2.2.3 Infrared data 16

2.2.4 Point Clouds 16

2.2.5 Pose Kinematics 16

2.3 Poses, Actions and Trajectories 21

2.4 Overview of Skeletal Sequence Modeling with Deep Neural Networks 23

2.4.1 Fully Connected Neural Networks 23

2.4.2 Recurrent Neural Networks 24

2.4.3 Convolutional Neural Networks 25

2.4.4 Spatio-temporal Attention 27

2.4.5 Graph Neural Networks 27

2.5 Representations, Inductive Biases and their roles during classification with little data . . 28

2.5.1 Importance of Explicit Temporal Modeling 28

2.5.2 Data-centric AI: the importance of the input data representation 37

2.6 Summary 43

3 From Action Recognition to Pedestrian Discrete Intention Prediction **45**

3.1 Understanding intentions and their role in predicting trajectories 46

3.2 Trajectory-based pedestrian action prediction 47

3.2.1 Related Works 48

3.2.2 From Bi-RNNs to U-RNNs . . . 50
3.2.3 Methodology . . . 50
3.2.4 Experiments . . . 51
3.2.5 Conclusion and Perspectives . . . 56
3.3 Pedestrian Discrete Intention Prediction . . . 57
3.3.1 Hit the road Jack: Human-factor perspectives on pedestrian behavior prediction . 57
3.3.2 Literature Review of State-of-the-Art . . . 60
3.3.3 Data sets for Pedestrian Intention Prediction . . . 61
3.4 Inferring crossing behavior via pose kinematics only . . . 64
3.4.1 SPI-Net: a representation-focused multi-branch deep learning network . . . 66
3.4.2 TrouSPI-Net: Spatio-temporal attention on parallel atrous convolutions . . . 76
3.5 Summary . . . 86

4 Assessing the Generalization of Pedestrian Crossing Predictors 89
4.1 Introduction . . . 90
4.2 The past, current and future state of pedestrian intention prediction benchmarks? . . . 91
4.2.1 Stone Age: prior to the release of the standardized evaluation procedures . . . 91
4.2.2 Bronze Age: one benchmark to rule them all . . . 92
4.2.3 Iron Age: identifying the generalization capabilities of our models? . . . 93
4.3 Sutor, ne ultra crepidam, or the necessity of uncertainty . . . 94
4.4 Generalization Capabilities . . . 96
4.4.1 Datasets and Implementation Details . . . 96
4.4.2 Baselines and state-of-the-art models . . . 96
4.5 New Evaluation Paradigm . . . 98
4.5.1 Cross-dataset Evaluation Results . . . 98
4.5.2 Role of pre-training in uncertainty calibration . . . 101
4.6 Improving Uncertainty Calibration . . . 102
4.6.1 Baselines from the probabilistic deep learning literature . . . 104
4.6.2 Discussion . . . 104
4.7 Summary . . . 105

5 Conclusion 107
5.1 Summary . . . 108
5.2 Future Works . . . 110

A Appendix 113
A Assessing TrouSPI-Net performance for skeletal action recognition datasets . . . 113
B Additional reliability diagrams for eleven baselines on three dataset for pedestrian discrete intention prediction . . . 114

B Résumé en français 121

C Bibliography 125

List of Figures

2.1 Roman bronze reproduction of Myron's Discobolus. The potential energy expressed in this sculpture's pose, expressing the moment of stasis just before the release, is an example of the advancement of Classical antiquity sculpture to depic motion through a static pose. 10

2.2 Drawing in Leonardo da Vinci's sketchbooks (a man going upstairs, or up a ladder). . . . 10

2.3 Chronophotography of a woman walking downstairs (Eadweard Muybridge, late 19th century) 11

2.4 Outline contours of a walking and a running subject (A) and the corresponding dot configurations (B). Picture credit [Johansson, 1973] 12

2.5 Action samples of different data modalities. Left to right: RGB, Skeleton, Depth, Infrared, and Point Cloud. 13

2.6 Illustration of RGB-based deep learning methods for action recognition: **(a)**: two-stream 2D CNN-based methods, **(b)** RNN-based methods, **(c)** 3D CNN-based methods. Image adapted from [Sun et al., 2020]. 14

2.7 An Architecture of decision level fusion of RGB stream and Depth modalities. Picture credit [Gao et al., 2019] 15

2.8 Examples of poses. Picture credit [Babu, 2019] 17

2.9 Example of difficult cases for skeletonisation algorithms: clothing and brightness on the left, occlusion and scale on the right. 17

2.10 **Top down**: consists of adding a person detector in order to identify all the articulations (keypoints) of each person and then estimate the pose according to them. **Bottom up**: consists of detecting all the keypoints in the image (*i.e* the limbs of each person), then associating these keypoints with their respective owners. Picture Credit [BeyondMinds, 2020] 18

2.11 Trajectories of the joints of a walking skeleton. Picture credit [Olsen et al., 2018] 22

2.12 Organization of the 3D skeleton data structure into a three-channel image (RGB) 26

2.13 **(left)** Convolution on 2-D grid-like data. The number of neighboring nodes is a fixed number determined by the filter size. **(right)** Generalized convolution operation on unstructured data. The number of neighboring nodes, determined by edge connectivity may vary from node to node. Picture credit [Wu et al., 2019b] 27

2.14 Pipeline of the approach: (1) we train an auto-encoder to reconstruct a sequence representing an action according to the evolution over time of the keypoints. We also add a constraint specific to the separability of classes in the latent space. (2) We then extract the weights of the encoder part up to the bottleneck represented in red and add a classifier, which transforms the encoder part into a pre-trained network on the data for action classification. 30
2.15 Example of a swipe left gesture extracted from SHREC dataset. Picture Credit [De Smedt et al., 2017]. 32
2.16 Example of actions and scenes extracted from JHMDB. Picture Credit [Liu et al., 2018]. 33
2.17 Confusion matrix obtained on SHREC 28 with a regularized auto-encoder ($\lambda = 5$). 35
2.18 Visualization of the projection of the instances and their class centroids in the latent space for SHREC dataset via T-Sne: **left** classic auto-encoder, **right** auto-encoder combined with Linear Discriminant Analysis. 36
2.19 Data structure of the skeleton representation obtained with the OpenPose library [Cao et al., 2017] 38
2.20 Kernel of a 2D convolution sliding over the pseudo-image 39
2.21 **(a)** Joints of the skeleton of a human body with the initial data structure (14 keypoints). The visiting order of the nodes is incremental:0-1-2-3-...-13. **(b)** The skeleton is transformed into a tree structure. **(c)** The tree can be unfolded into a chain whose order of visit of the nodes maintains the physical relationship of the joints: 1-0-1-8-10-12-10-8-1-9-11-13-11-9-1-2-4-6-4-2-1-3-5-7-5-3 (26 keypoints). 40
2.22 The network architecture of DD-Net [Yang et al., 2019]. "2×CNN(3, 2*filters), /2" denotes two 1D ConvNet layers (kernel size = 3, channels = 2*filters) and Maxpooling (strides = 2). Other convolutive layers are defined in the same format. GAP denotes Global Average Pooling. FC denotes Fully Connected Layers. We can change the model size by modifying the "filters" parameter. 41
2.23 The architecture of the LeNet network [LeCun et al., 1998], with the basic components of a convolutional network: convolutions, pooling, fully connected layer and a softmax classifier. 41

3.1 Trajectory-based pedestrian action prediction: the task is to forecast the future trajectories (dashed) of all the protagonists of the scene. Trajectory-based pedestrian action prediction involves a combination of individual goals and social interactions with other agents: pedestrian X1 will deviate from his primary trajectory to avoid a collision based on past trajectories of pedestrian X2. Picture credit [Kothari et al., 2021] 47
3.2 Illustration of the grid-based interaction encoding modules for trajectory-based intention prediction. (a) Occupancy pooling: each cell indicates the presence of a neighbour (b) directional pooling: each cell contains the relative velocity of the neighbour with respect to the primary pedestrian. (c) Social pooling: each cell contains the LSTM hidden-state of the neighbour. Picture credit [Kothari et al., 2021] 48
3.3 Sample from the Stanford Drone Dataset (which is not included in the Trajnet++ benchmark). The environment would play an important role in order to predict trajectories that do not go on the lawn. 49

3.4 Comparison between Bi-RNN and U-RNN architectures (blue: inputs - red: outputs - black: hidden states - green: intermediate output). U-RNN can use the information from the future during the forward pass, whereas the Bi-RNN only concatenates two naive readings in both directions. 50

3.5 Images from different datasets from which the Trajnet++ benchmark trajectories are extracted. Left: ETH-hotel dataset - Center: UCY-zara dataset - Right: UCY-students dataset. 52

3.6 Visualization of four high level defined trajectory categories and visualization of all Type III interactions. Picture credit [Kothari et al., 2021]. 53

3.7 "*Factors involved in pedestrian decision-making process at the time of crossing. The diagram is based on a meta-analysis of the past literature. The large circles refer to the major factors and small circles connected with solid lines are sub-factors. The dashed lines show the interconnection between different factors and arrows show the direction of influence*". Picture and legend credits [Rasouli and Tsotsos, 2019]. 58

3.8 Pedestrian Intention Prediction: the objective is to predict if the pedestrian will start crossing the street at some time t given the observation of length m. Figure adapted from [Kotseruba et al., 2021]. 61

3.9 Example of a multi-modal approach for pedestrian crossing prediction, in the given case the architecture is composed of five GRUs. Each of which processes a concatenation of features of different modalities and the hidden states of the GRU in the previous level. The information is then fused into the network gradually according to the complexity of the features. Picture credits [Rasouli et al., 2019b]. 62

3.10 (A) Examples of attention towards their environment and communication demonstrated by pedestrians in urban traffic scenarios. The use of pose alone in these use cases would not be a problem since the orientation of the head, the dynamics of the arms would be sufficient to capture the semantics of the scene. (B) Scenarios with irrelevant actions with no particular semantics: eating, touching or cleaning face and looking at the phone. Pose alone might not be sufficient in most cases to identify relevant or irrelevant actions (C) In larger groups, leader-followers patterns are such that only a few pedestrians look, and the rest of the group follows. Since we are dealing with pedestrian discrete intention prediction at the level of granularity of a single pedestrian, not taking the environment into account could be a problem in this type of scenario as the pose would probably not prove sufficient. 65

3.11 The multi-branch architecture of SPI-Net: the left branch focuses on the evolution of Geometric features relative to certain identified key-points over time. The second one focuses on the evolution of the spatial representation of skeletal key-points as a function of time in the Cartesian coordinate system. CNN 2D Blocks denote one 2D ConvNet layer (kernel size= 3), an AveragePooling layer and a Batchnormalization layer. Other Dense blocks are defined in the same format with a Batchnormalization layer following each Dense layer. 67

3.12 Pipeline of the approach for the Geometric branch: (**1**) we train an auto-encoder to reconstruct a sequence representing an action according to the evolution over time of the distances (represented by the red arrows) of selected keypoints (Torso, Left and Right Shoulders, Left and Right Knees). We also add a constraint specific to the separability of classes in the latent space. (**2**) We then extract the weights of the encoder part up to the bottleneck represented in red and add a classifier, which transforms the encoder part into a pre-trained network on the data for action classification. 68

3.13 The Cartesian coordinate feature is highly dependent on locations and viewpoints. When body poses are rotated or shifted, the Cartesian coordinate feature can be significantly impacted representation-wise. Meanwhile, the geometric feature (e.g., angles/distances), is location-viewpoint invariant, and thereby stays the same. This compensates for the inabilities of the Cartesian coordinate feature branch in learning temporal patterns invariant to locations and viewpoints. Picture credits [Yang et al., 2019]. 69

3.14 28 different ground-truth sequences represented in a 3-dimensional (300,25,2)-shaped tensor after the TSSI normalization. The horizontal axis of each TSSI sequence is the keypoints axis. The vertical axis of each TSSI sequence is the time axis. The x, y dimensions are mapped to RG(B) channels for visualization. The axes are kept fixed and the aspect is adjusted so that the data fit in the axes. Ground truth labels C or NC represent the Crossing or not Crossing future action of the pedestrian. 71

3.15 Behavioral Time line of a crossing pedestrian in the Joint Attention in Autonomous Driving (JAAD) data set. 72

3.16 Intention prediction accuracy of the Geometric branch alone, as a function of its λ parameter. 74

3.17 **The network architecture of TrouSPI-Net:** Its inputs consist of a sequence of 2D body poses transformed into a pseudo-image, relative pairwise distances of skeletal joints, bounding boxes, and ego-vehicle speed. U-GRUs encode every feature except pseudo-images, and each is fed into a temporal attention block. Pseudo-images are processed by parallel atrous CBAM [Woo et al., 2018] blocks with different dilation rates and then added into a single vector in order to make the size of the pseudo-images block equal to the size of the U-GRUs outputs. Modality attention is then applied to the outputs of each branch, and the weighted outputs are fed into the fully connected layer. **U-GRU blocks:** the first GRU layer does the reverse pass, we then concatenate its output with the input data and finally compute the second GRU layer's output with a forward pass. **CBAM blocks:** given an intermediate feature map extracted by atrous 2D convolutions, the module sequentially infers attention maps along two separate dimensions: channel and spatial. 78

3.18 Atrous convolutions applied to our pseudo-images: compared to regular convolutions, it involves pixel skipping, so as to cover a larger area of the input in time while staying at the same spatial resolution. This could prove useful for two use cases: **(1)** the scale of pedestrians' actions patterns might extend through time and is not limited by a specific temporal resolution, relying on atrous convolution allows TrouSPI-Net to capture features for a given pedestrian action pattern for multiple temporal resolutions and could potentially improve generalization. **(2)** Pose estimation algorithms often reconstruct temporally noisy poses when given in-the-wild video data, combining three different action extraction feature protocols for three different time ranges could have a regularizing effect on the potential pose noise obtained at a given timestamp. 79

4.1 Distribution of pedestrian bounding box height in pixel for PIE, $JAAD_{behavior}$ and $JAAD_{all}$. 93

4.2 Examples of crossing and non-crossing pedestrians from $JAAD$ and PIE datasets. The conditions under which pedestrians act from one scenario to another can differ drastically concerning input format and domain shift: pedestrian size, pedestrian positioning in the scene, illumination conditions, occlusion... 94

4.3 Pedestrian crossing prediction performance for PIE, $JAAD_{behavior}$ and $JAAD_{all}$. We show a comparison between traditional single-dataset train and test evaluation on each dataset compared to cross-dataset evaluation for eleven methods representing the diversity of architectures and modalities usually used for pedestrian crossing prediction. Ensembling denotes the average prediction given by the three models trained on each dataset for one given test set. 97

4.4 Distribution of the performance of the eleven selected approaches when evaluated in a direct train-test scenario and when evaluated in cross-dataset scenarios. 100

4.5 Critical Difference Diagram [Demšar, 2006]: first a Friedman test is performed to reject the null hypothesis, we then proceed with a post-hoc analysis based on the Wilcoxon-Holm method. We compare the robustness of classifiers over multiple training and testing sets shifts. We can see how each method ranks on average. A thick horizontal line groups a set of classifiers that are not significantly different ($\alpha = 0.1$). 100

4.6 Reliability Diagrams between I3D [Carreira and Zisserman, 2017] randomly initialized (left) and pre-trained on Sports1M [Karpathy et al., 2014](right) on PIE, $JAAD_{all}$ and $JAAD_{behavior}$ datasets. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green). 102

A.1 Confusion matrix obtained on SHREC 14 with TrouSPI-Net 114

A.2 Confusion matrix obtained on SHREC 28 with TrouSPI-Net 114

A.3 Reliability Diagrams for the eleven selected methods on PIE data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green). 115

A.4 Reliability Diagrams for the eleven selected methods on $JAAD_{behavior}$ data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green). 116
A.5 Reliability Diagrams for the eleven selected methods on $JAAD_{all}$ data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green). 117
A.6 Reliability Diagram of the Average prediction given by three individual models and their respective outputs for PIE (Ensembling), Each individual model is either trained on $PIE, JAAD_{behavior}$ or $JAAD_{all}$. 118
A.7 Reliability Diagram of the Average prediction given by three individual models and their respective outputs for $JAAD_{behavior}$ (Ensembling), Each individual model is either trained on $PIE, JAAD_{behavior}$ or $JAAD_{all}$. 119
A.8 Reliability Diagram of the Average prediction given by three individual models and their respective outputs for $JAAD_{all}$ (Ensembling), Each individual model is either trained on $PIE, JAAD_{behavior}$ or $JAAD_{all}$. 120

List of Tables

2.1 Various relational inductive biases in standard deep learning components. *An inductive bias allows a learning algorithm to prioritize one solution (or interpretation) over another, independent of the observed data.*[Battaglia et al., 2018] 23

2.2 Properties of the selected experimental datasets. 33

2.3 Performance of the given model for different encodings of the sequences on SHREC [De Smedt et al., 2017], the architecture of the model remains unchanged. 34

2.4 Performance of the given model for different encodings of the sequences on JHMDB [Jhuang et al., 2013], the architecture of the model remains unchanged. 34

2.5 Results on the SHREC dataset using the train/test split protocol 37

2.6 Results obtained via DFS normalization on SHREC [De Smedt et al., 2017], the architecture DD-NET [Yang et al., 2019] remains unchanged. 42

2.7 Results obtained via to DFS normalization on JHMDB [Jhuang et al., 2013], the architecture DD-NET [Yang et al., 2019] remains unchanged. 42

2.8 Results obtained with DFS normalization on SHREC [De Smedt et al., 2017] (3D hand skeletons) for a 2D convolutional network. 42

2.9 Results obtained with DFS normalization on JHMDB [Jhuang et al., 2013] (2D human body skeletons) for a 2D convolutional network. 42

3.1 Results for several baselines and for the best submission on the Trajnet++ public leaderboard (with respect to FDE). 54

3.2 Comparison of motion-encoding designs with respect to various interactions modules architectures on interacting trajectories of TrajNet++ real world dataset. 55

3.3 Pedestrian and Environmental Factors involved in pedestrian decision-making process in accordance with the perceptive modality used as defined in section 3.3.1. 65

3.4 Intention prediction accuracies of the Geometric branch alone, for different encodings of the sequences of inter-keypoints distances. 74

3.5 Ablation studies: classification accuracy of the Cartesian branch for pedestrian intention prediction for the crossing or not crossing task in JAAD. 74

3.6 Classification accuracies for pedestrian intention prediction for the crossing or not crossing task in JAAD. CPN [Chen et al., 2017], Alphapose [Fang et al., 2016] and Openpose [Cao et al., 2017] stand for the use of human pose estimation algorithms used by the skeleton-based features method. We have also included the results reported in [Rasouli et al., 2017b, Varytimidis et al., 2018], where CNN features are based on a non-fine-tuned AlexNet [Krizhevsky et al., 2012] and Context refers to features of the environment, not of the pedestrian itself. 75
3.7 Confusion matrix of the JAAD data set obtained by each branch of SPI-Net and SPI-Net on JAAD for the crossing or not crossing task. 76
3.8 Evaluation results for baseline and state-of-the-art models and their variants on PIE and JAAD data-sets. Dashed lines separate different types of architectures. Modalities correspond to the type of networks used in the given approach, Model Params corresponds to the size of the network compiled on the benchmark [Kotseruba et al., 2021] with Additional Costs (Optical flow, Body Pose, RGB features) already extracted. 82
3.9 Architecture variations and Ablation studies for TrouSPI-Net on PIE data-set. 83
3.10 Architecture comparison of floating-point operations per second (FLOPS) in millions, Cuda Memory Usage (CMU) in Megabytes and Weights Memory Requirements (WMR) in Megabytes. RGB features extracted by CNNs are taken into consideration during computations. 85

4.1 Pedestrian action prediction models trained and evaluated on JAAD and PIE datasets prior to the standardized benchmarks and evaluation procedures. 91
4.2 List of all the pedestrian action prediction models trained and evaluated on the standardized benchmarks 92
4.3 Average prediction given by the three models trained on each training sets for one given test-set (Ensembling). In addition to classification metrics (we use arrows to indicate which direction is better), we compare models with predictive uncertainty metrics such as Expected Calibration Error (ECE) and Maximum Calibration Error (MCE). Dashed lines separate different types of architectures 99
4.4 Average Pedestrian Crossing Prediction performance for PIE, $JAAD_{behavior}$ and $JAAD_{all}$ (5 runs). Dashed lines separate each probabilistic deep learning baseline. Each baseline is tested twice: first, in a classical train-test evaluation protocol and then tested by ensembling all three models trained on each training set to evaluate its robustness to small domain shift. We highlight the highest scores for each metric and for both evaluation protocols: train-test or ensembling. 103

A.1 Results obtained via TrouSPI-Net on SHREC [De Smedt et al., 2017]. We change the model size by modifying the filters parameter for each convolution block. 113
A.2 Results obtained via TrouSPI-Net on JHMDB [Jhuang et al., 2013]. We change the model size by modifying the filters parameter for each convolution block. 113

Chapter 1

Introduction

Il faudra toujours qu'il y ait de mauvais écrivains, car ils répondent au goût des âges non développés, non mûris ; ceux-ci ont leurs besoins aussi bien que les plus mûrs.

Nietzsche - Humain, trop humain

Contents

1.1 Context 4
1.2 Objectives 5
1.3 Publications and communications 6
1.4 Outline 7

1.1 Context

The ultimate goal of the intelligent transportation research field should be to show that robots can cohabit with humans and can efficiently share space. For instance, an unused electric scooter could share the curb with dozens of pedestrians so that it can park in a place that does not obstruct the majority of the pedestrian flow, a personal assistance robot could follow the elderly in crowded spaces such as shops so that they do not have to carry their groceries, an autonomous vehicle could easily navigate throughout the crowded streets of downtown centers of major cities while ensuring the flow of traffic... All of them while respecting arbitrary safety and ethical rules such as keeping a safe distance from the other protagonists, respecting local driving laws and/or local social norms... To navigate in urban traffic environments while remaining efficient, autonomous vehicles should be able to efficiently negotiate social interactions with the other protagonists of the scene. Hence, the ability to interpret human intentions and actions is necessary for the design of meaningful human-machine interactions since coordination between humans and machines is only possible if both parties are aware of each other's intentions or underlying motives. Consider the following scenario: you are driving down the street and come upon a person standing on the corner. How can you tell if this person is going to cross? By making a wise combination of communication, social norms, personal experience and law compliance. A driver's role is to determine whether another road user wants him to wait and let the road user cross or not based on the contextual cues and communication he provides. Of all the existing ways to communicate with one's surroundings, gesture is one, if not the most natural and easy form of communication among human beings. For instance, a person's head direction frequently reflects where he intends to travel, whilst his body orientation frequently indicates which direction he is presently going, a hand gesture could be considered an explicit form of communication with the driver to signal gratitude or dissatisfaction, the same way that establishing eye contact with the driver could be considered as an implicit form of communication to ensure that you have been seen.

Understanding the intention of the protagonists of a scene from the driver's perspective could therefore prove useful for the deployment of autonomous vehicles because:

- It would improve safety for the most vulnerable road users: knowing the intention of pedestrians to cross the road before they actually set foot on the road would allow the vehicle to warn the driver or automatically perform maneuvers. Therefore, preserving the pedestrians' integrity in a more efficient way than when triggered by an emergency stop once the pedestrians have moved on to the road and become a direct obstacle for the vehicle would be safer for all actors.

- It would ensure the flow of traffic: 98% of autonomous vehicles accidents are due to an unexpected stop of the ego-vehicle [Favarò et al., 2017]. As communication helps to disambiguate certain traffic situations, failing to understand the most basic forms of communication between road users can potentially slow down the flow of traffic.

- It would help identify unscrupulous actions such as stepping in front of the ego-vehicle to force it to stop or change its route [Färber, 2016].

1.2 Objectives

The initial objectives of this thesis are to explore deep learning approaches as a way to efficiently leverage spatial and temporal components of pedestrian poses kinematics and efficiently detect their intention of crossing in urban traffic environments. In order to delimit the scope of this thesis while trying to address theses broad objectives, we orient our research with several research directions:

Question 1 *Inductive biases are the set of assumptions a learner uses to predict results given inputs it has not yet encountered. When training deep learning architectures with little available data, should we only rely on the very composition of layers to impose relational inductive biases on the learner? Does enforcing certain constraints towards the data representation of designated hidden layers, sending informative-representation ready data to the classification network help the performance of deep learning networks for action classification?*

Question 2 *Visual skeletal representations are known to be sufficient for both humans and machines to describe and recognize biological motion, including human motion. Can pose kinematics be sufficient to serve as the only input when modeling non-trivial and non-periodic tasks related to pedestrian intention prediction?*

Question 3 *Does recent progress on pedestrian intention prediction benchmarks continue to represent meaningful generalization? What evaluation protocol and metrics should be used to go beyond accuracy in order to evaluate a model for a high-risk application with a limited amount of training data?*

First, to focus on the spatio-temporal aspect of poses kinematics, we choose to work on skeletal action recognition in a controlled environment. At the time of the beginning of this thesis, the state-of-the-art approaches for skeletal action recognition focused mainly on the sequential modeling part of the problem while relying heavily on deep-learning networks to automatically build high-level representations of the raw input. Since deep learning approaches depend heavily on the quantity and quality of data where the performance scales up with the amount of training data, the given paradigm does not encourage the community to study and improve the capabilities of deep networks with little available training data. For that reason, we choose to design deep learning architectures that act on both spatial and temporal components of the raw input and enforce the importance of engineering the data used, prior to blindly relying on automatic features learning from training examples.

Second, the complexity of an action recognition algorithm is directly impacted by the number of perception modalities it uses. Fusing multiple perceptive modalities into a single representation often lead to a high complexity, a high training time and a consequent inference time due to the presence of multiple networks extracting features for each modality (RGB, Optical Flow, Pose Dynamics...). Considering the importance for crossing prediction algorithms to run efficiently for real-time usage while being robust to a multitude of complexities and conditions, our goal is to propose a model using only one perception modality for pedestrian intention prediction that reaches the performance of multi-modal approaches.

Third, while empirically measuring the overall progress of pedestrian intention prediction algorithms over time tends to be more and more established due to the new publicly available benchmarks, know-

ing how well existing predictors react to unseen data remains an unanswered question. We question the legitimacy of the current evaluation protocols to adequately represent the applicability of evaluated pedestrian prediction models for real-world scenarii.

1.3 Publications and communications

The main publications and communications of this thesis can be synthesized as follows:

Human Activity Recognition With Pose-driven Deep Learning Models

- Gesnouin, J.; Pechberti, S.; Bresson, G.; Stanciulescu, B.; Moutarde, F. Rethinking Robust Embedding for Skeleton Human Action Recognition (GdR ISIS; Journée Action "Visage, geste, action et comportement", janvier 2021)

Pedestrian Continuous Trajectory Forecasting

- Rozenberg, R., Gesnouin, J., Moutarde, F. (2021). Asymmetrical bi-rnn for pedestrian trajectory encoding. Reconnaissance des Formes, Image, Apprentissage et Perception (RFIAP), juillet 2022.
  - Rozenberg, R., Gesnouin, J., Moutarde, F. (2021, May). Asymmetrical Bi-RNNs (U-RNNs), 2nd place solution at the Trajnet++ Challenge for pedestrian trajectory forecasting. In Workshop on Long-term Human Motion Prediction, 2021 IEEE International Conference on Robotics and Automation (ICRA).
  - Rozenberg, R., Gesnouin, J., Moutarde, F. (2021, October). Asymmetrical Bi-RNNs, 3rd place solution at the ICCV Trajnet++ Challenge. In ICCV 2021 Multi-Agent Interaction and Relational Reasoning Workshop.
  - Best presentation during the National Young Researcher's Day in Robotics 2021.

Pedestrian Discrete Intention Prediction

- Gesnouin, J., Pechberti, S., Bresson, G., Stanciulescu, B., Moutarde, F. (2020). Predicting intentions of pedestrians from 2d skeletal pose sequences with a representation-focused multi-branch deep learning network. Algorithms, 13(12), 331.
- Gesnouin, J., Pechberti, S., Stanciulcscu, B., Moutarde, F. (2021, December). TrouSPI-Net: Spatio-temporal attention on parallel atrous convolutions and U-GRUs for skeletal pedestrian crossing prediction. In 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021) (pp. 01-07). IEEE.

Uncertainty and Domain Shift evaluation of Pedestrian Crossing Predictors

- Gesnouin, J., Pechberti, S., Stanciulescu, B., Moutarde, F. (2022). Assessing Cross-dataset Generalization of Pedestrian Crossing Predictors. 33rd IEEE Intelligent Vehicles Symposium.

1.4 Outline

This thesis is laid out in five chapters:

Introduction We briefly described the context of this thesis, the problems it addresses, and the thesis's primary contributions.

Human Activity Recognition With Pose-driven Deep Learning Models In this chapter, we provide historical notes on human actions understanding and an overview of modern computer vision modalities for action recognition. We then introduce the different families and inductive biases of deep learning architectures for skeletal sequences modeling. Existing approaches fall into four broad main categories: recurrent neural networks, convolutional neural networks, attention-based associative memory neural networks and graph-neural networks. Thereafter, we question the importance of representations, inductive biases and their roles for skeletal action recognition. Firstly, we evaluate the importance of explicit temporal modeling for gesture recognition: while gestures are temporal phenomena, many gestures and actions might actually be inferred based on spatial poses only. We propose a fully-connected auto-encoder, that does not benefit from any inductive bias and enforces the mapping from inputs to outputs in the embedding via statistical regularizations. We show that the proposed approach reaches the performances of classic sequence modeling architectures on action classification tasks with little available data. Secondly, we investigate the importance of sending informative-representation ready data to a deep learning architecture, prior to the learning of multiple layers of feature hierarchies that automatically build high-level representations of the raw input. By normalizing the input data based on physical world constraints of the body structure, we show that for action classification tasks with little data, networks benefit from handcrafted features and could rely on fewer hidden layers to learn informative representations of data.

From Action Recognition to Pedestrian Intention Prediction In this chapter, we first provide an overview of existing approaches for pedestrian action prediction. The majority of existing techniques to pedestrian action prediction are trajectory-based, which means they depend on previously observed pedestrian positions to anticipate pedestrian positions in the future. These methods are successful when pedestrians have already crossed or are going to cross, i.e., these algorithms react to an action that has already occurred rather than predicting it. We first propose an asymmetrical bidirectional recurrent neural network architecture called U-RNN to encode pedestrian trajectories and evaluate its relevance to replace LSTMs for various trajectory-based models. Secondly, we address the problem of pedestrian discrete intention prediction: instead of focusing on continuous trajectories describing the expected future movement of the pedestrian and merely relying on scene dynamics to predict intentions, we define the intentions of a pedestrian as a combination of his/her high-level discrete behaviors such as his/her pose dynamics, head orientation... Finally, we show that it is possible to make the link between the posture, the walking attitude and the future behaviours of the protagonists of a scene without using the contextual information of the scene (pedestrian crossing, traffic light...). This allowed us to divide by a factor of 20 the inference speed of existing approaches for pedestrian intention prediction while keeping the same prediction robustness.

Assessing the Generalization of Pedestrian Crossing Predictors this last chapter is deliberately more exploratory. Pedestrian crossing prediction has been a topic of active research, resulting in many new algorithmic solutions, While measuring the overall progress of those solutions over time tends to be more and more established due to the new publicly available benchmark and standardized evaluation procedures, knowing how well existing predictors react to unseen data remains an unanswered question. This evaluation is imperative as serviceable crossing behavior predictors should be set to work in various scenarios without compromising pedestrian safety due to misprediction. To this end, we conduct a study based on direct cross-dataset evaluation. Our experiments show that current state-of-the-art pedestrian behavior predictors generalize poorly in cross-dataset evaluation scenarios, regardless of their robustness during a direct training-test set evaluation setting. In the light of what we observe, we argue that the future of pedestrian crossing prediction, *e.g.* reliable and generalizable implementations, should not be about tailoring models, trained with very little available data, and tested in a classical train-test scenario with the will to infer anything about their behavior in real life. It should be about evaluating models in a cross-dataset setting while considering their uncertainty estimates under domain shift.

Conclusion We summarize this thesis and identify potential future directions for our research.

Chapter 2

Human Activity Recognition With Pose-driven Deep Learning Models

Contents

2.1 Historical Notes on Human Actions Understanding 10

2.2 Overview of Modern Computer Vision Modalities for Action Recognition 12

2.2.1 RGB videos 13

2.2.2 Depth data 15

2.2.3 Infrared data 16

2.2.4 Point Clouds 16

2.2.5 Pose Kinematics 16

2.3 Poses, Actions and Trajectories 21

2.4 Overview of Skeletal Sequence Modeling with Deep Neural Networks 23

2.4.1 Fully Connected Neural Networks 23

2.4.2 Recurrent Neural Networks 24

2.4.3 Convolutional Neural Networks 25

2.4.4 Spatio-temporal Attention 27

2.4.5 Graph Neural Networks 27

2.5 Representations, Inductive Biases and their roles during classification with little data 28

2.5.1 Importance of Explicit Temporal Modeling 28

2.5.2 Data-centric AI: the importance of the input data representation 37

2.6 Summary 43

2.1 Historical Notes on Human Actions Understanding

The understanding of human gesture, both in the way it is performed and in the way it is interpreted, has been a subject of interest in several disciplinary fields, such as science and art over the centuries. Classical antiquity can be considered the birth period of the most important technical contributions to the understanding of human movement [Klette and Tee, 2008]. Motion patterns of humans were usually studied in close relation to motion patterns of animals and were typically observed in arts[1]. Most notably in the case of sculpture, artists were seeking a very accurate depiction of motion through a single static image.

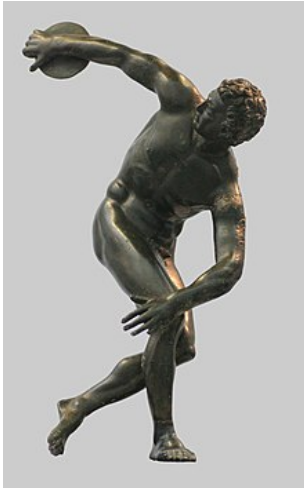


Figure 2.1: Roman bronze reproduction of Myron's Discobolus. The potential energy expressed in this sculpture's pose, expressing the moment of stasis just before the release, is an example of the advancement of Classical antiquity sculpture to depic motion through a static pose.

Later on, during the Renaissance, Leonardo da Vinci's sketchbooks contained studies about the human body and its movement. Da Vinci introduced the term kinematic trees, referring to kinematic chains that model the underlying structure of the human body, and that allow human poses to be represented by means of articulated models.

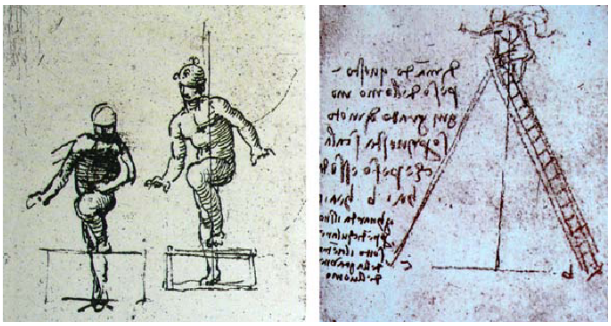


Figure 2.2: Drawing in Leonardo da Vinci's sketchbooks (a man going upstairs, or up a ladder).

Renaissance artistic currents emphasized the comprehensive development of perspective as a technique for expressing situations in a single frame or picture. As a result, artists of the time were preoccu-

[1] "Why are man and birds bipeds, but fish footless; and why do man and bird, though both bipeds, have an opposite curvature of the legs?", Aristotle - On the Parts of Animals

pied with the portrayal of the three-dimensional environment, as well as its link to the right representation of human position and motion. Projective geometry even became a mathematical theory, pioneered by Girard Desargues at the beginning of the Baroque era. Later on, Giovanni Alfonso Borelli contributed to human motion understanding with studies that applied Galilei's mechanics to analyze motion for biological purposes, which is considered as the birth of biomechanics.

Following the evolution of human gesture understanding, the most notable contributions after the birth of biomechanics are then found during the 19th century, with the advances in terms of capturing devices, and more specifically the chronophotography: a photographic technique that captures a number of phases of movements as presented in Fig 2.3.

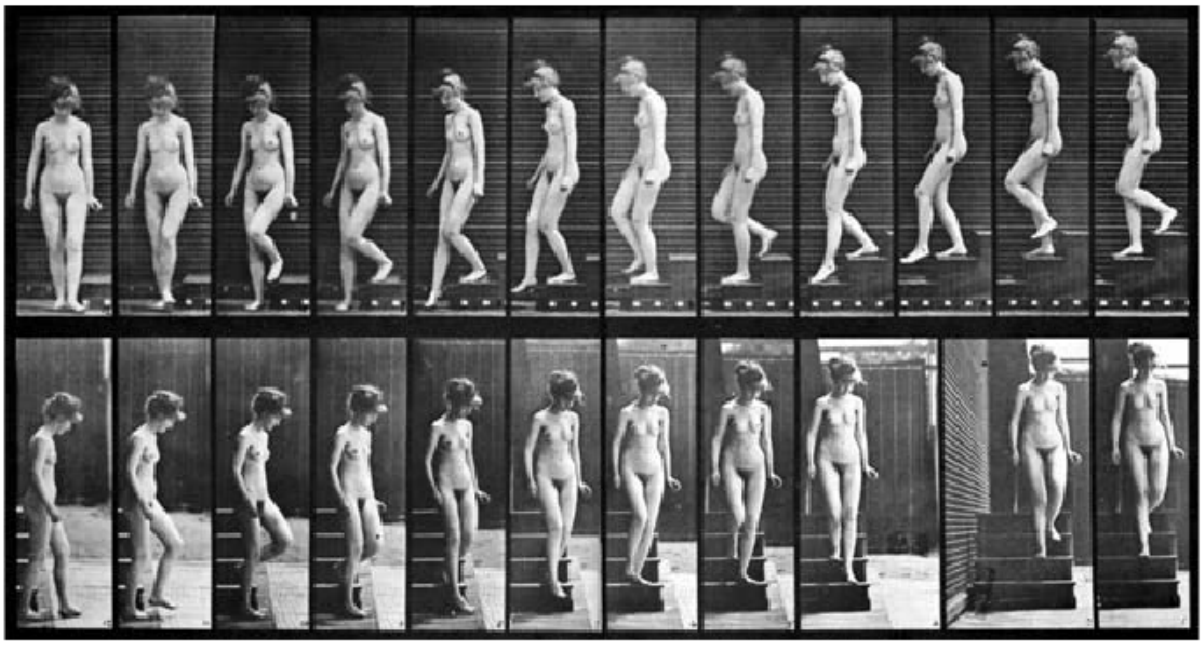


Figure 2.3: Chronophotography of a woman walking downstairs (Eadweard Muybridge, late 19th century)

Etienne-Jules Marey, and Eadweard Muybridge are two pioneers in the use of chronophotography for the study of movement. Marey invented a system that can be considered the first marker-based motion capture system. Muybridge developed a system to display the recorded series of images, pioneering motion pictures this way. His technique was applied to movement studies for different categories of activities (walking down stairs, boxing, sprinting...) and was very influential for the beginning of cinematography at the end of the 19th century as well as for art. For instance, Marcel Duchamp, Picasso and Francis Bacon were all influenced by Marey or Muybridge for their artistic development. During the second half of the 19th century, Albert Londe, used chronophotography to study the movements of patients during epileptic fits and became the first scientific medical chronophotographer [Londe, 1893].

During the 20th century, biomechanics became an independent discipline of science and research, mainly in the context of sports. Finally, the technological evolution being the one we know, the proliferation of cheap cameras and processing power of modern computers during the last 50 years have paved the way to the development of computer vision technologies to understand and describe human motions

in various domains[2]. The genesis of human motion computer analysis can probably be found in the work of [Johansson, 1973, Johansson, 1976] and presented in Fig 2.4.

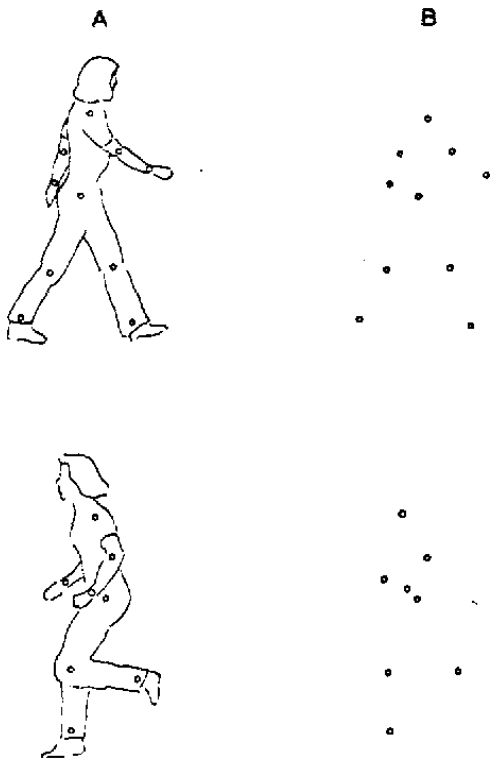


Figure 2.4: Outline contours of a walking and a running subject (A) and the corresponding dot configurations (B). Picture credit [Johansson, 1973]

The idea behind Johansson experiment is that humans can recognize human body motion actions, using the motion of the body's joints positions only. The given representation is a high-level, sparse, representation of the human body and should also be sufficient for the computer to understand the semantics of a gesture without the context. The resulting configuration of joints being rather similar to the one employed in modern marker-based motion capture systems or pose estimation models based on RGB data, Johansson's experiment is often considered as the beginning of human motion computer analysis. Finally, with the progress and diversity in terms of data acquisition sensors, the modern representations of motion in computer vision are not limited to the skeletal perception modality. In the following section, we provide an overview of modern computer vision modalities for action recognition.

2.2 Overview of Modern Computer Vision Modalities for Action Recognition

Humans' actions can be represented using various visual modalities, namely, RGB, depth, infrared, point cloud, or skeleton (see Figure 2.5). Actions can even be represented using non-visual modalities such as audio [Gao et al., 2020], radars [Chen and Ye, 2019] or even wifi-signals [Li et al., 2019c]. Each modal-

[2]In the late 1990s, there has been a whole series of work on human modelling, including ISOs, which came with image synthesis and animation in videos, and which also penetrated biomechanics and motion analysis: the H-anim project, which has become a standard in the biomechanical modelling of humans, poses and behaviour.

Figure 2.5: Action samples of different data modalities. Left to right: RGB, Skeleton, Depth, Infrared, and Point Cloud.

ity has it pros and cons depending on the application scenario and encodes different sources of useful yet different features of the scene. In this section, we provide an overview of existing perception modalities for single-modality action recognition. It is of course possible to combine several of these perception modalities to obtain a multi-modal representation of the scene, but this is at the expense of the inference speed of the model and it highly depends on the quality of the fusion or co-learning algorithm.

2.2.1 RGB videos

The RGB modality refers to images or sequences of images captured by RGB cameras to replicate what we, as humans see. It is the easiest perception modality to collect and it contains a lot of information about the context of the given recorded scene. RGB-based deep learning models have the advantage of using the most commonly used modality for action recognition. Therefore, one can benefit from huge large-scale web videos to pre-train their models for better recognition performance [Karpathy et al., 2014, Duan et al., 2020, Ghadiyaram et al., 2019]. However, due to heterogeneity in terms of backgrounds, context, inherent differences of the performers (age, physique, ethnicity...), viewpoints, scaling, and lighting conditions, action recognition from RGB data might be difficult to perform as there might be huge intra-classes differences. Moreover, RGB videos being massive data volumes, modeling the spatio-temporal components of human actions via videos has the disadvantage of leading to high computational costs. Current research in this field focuses on designing different types of deep learning frameworks to efficiently extract spatio-temporal features in a video: namely, Two-Stream 2D CNN-based methods, RNN-based methods and 3D CNN-based methods.

Two-Stream 2D CNN-based methods, as shown in Fig 2.6, learn different types of information (*e.g.*, spatial and temporal) from the input video features through separate networks and then perform late fusion to obtain the performed action. Some works proposed to enhance the vanilla version of Two-Stream [Simonyan and Zisserman, 2014] by using a third stream to add the motion saliency stream on top of the appearance information and motion information streams [Zong et al., 2021], others proposed to reduce the computational costs of the overall approach by either feeding low-resolution RGB frames to speed up the computation [Karpathy et al., 2014] or either avoiding computing perfectly accurate optical flow [Zhang et al., 2016, Piergiovanni and Ryoo, 2019]. Finally, some works tried to enhance the long-term-dependency-modeling capacity of two-stream networks as this is their main drawback by simply dividing each action video into three segments and processed each segment with a two-stream network. To produce the video-level prediction, each segment's score is then fused with average pooling [Wang et al., 2016a] or element-wise multiplication [Diba et al., 2017b].

Considering that Two-Stream approaches barely handle long-term dependencies, RNN-based models aim to efficiently model the long-term temporal dynamics in video sequences. As shown in Fig 2.6,

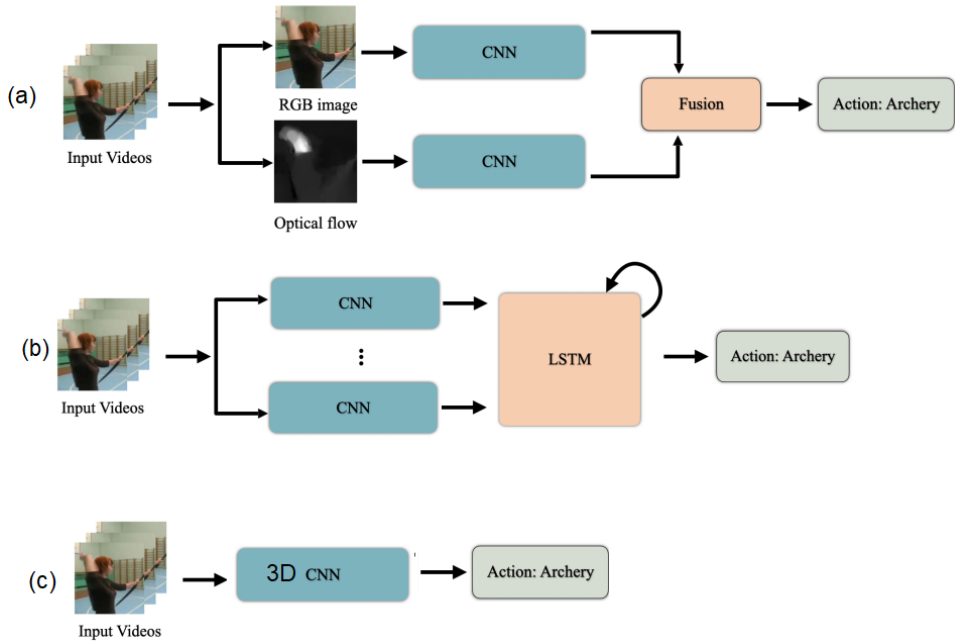


Figure 2.6: Illustration of RGB-based deep learning methods for action recognition: **(a)**: two-stream 2D CNN-based methods, **(b)** RNN-based methods, **(c)** 3D CNN-based methods. Image adapted from [Sun et al., 2020].

RNN-based models use convolutions as a feature extracting method for each frame in the sequence, extracted features are then used as input for a LSTM layer to consider the temporal relationships of each frame features. [Baccouche et al., 2011] introduced a two-steps scheme automatically learning spatio-temporal features via 3D convolutions and uses them to classify the entire sequence by using recurrent neural networks. Similarly, [Donahue et al., 2015] proposed a 2D convolutional neural network to extract frame-level RGB features followed by LSTMs to generate the overall action label. Some works extended the existing approaches by using GRUs [Shi et al., 2017, Dwibedi et al., 2018] or using Bi-directional LSTM instead of regular LSTM in order to learn both the forward and backward temporal information of an action [Ullah et al., 2017, He et al., 2021].

The last type of deep learning framework to model human motion in RGB camera streams consists of scaling 2D convolutions to 3D convolutions, thus capturing simultaneously the spatial and temporal context information in videos [Ji et al., 2012]. Every flavor-of-the-month deep-learning architecture for image classification got its 3 dimensional variant: DenseNet [Huang et al., 2017] got Temporal 3D CNN [Diba et al., 2017a], Resnet [He et al., 2016] got 3D ResNets[Hara et al., 2017], EfficientNet [Tan and Le, 2019] got EfficientNet3D [Kopuklu et al., 2019]... Some works investigated the combination of 3D convolutions with two-stream designs [Carreira and Zisserman, 2017, Wang et al., 2017], thus drastically increasing the complexity and computational burden of the proposed approaches. In the opposite direction, some works proposed to reduce the computational complexity and parameters size of the given methods by factoring 3D convolutions [Qiu et al., 2017, Xie et al., 2018] or by inserting temporal information over 2D Capsule Network with a zero computational cost instead of relying on 3D Convolutions [Voillemin et al., 2021].

In conclusion, the RGB video modality is one of the most explored modality for action recognition,

which makes it a modality of choice when dealing with tasks with not enough training data as one can benefit from huge large-scale datasets to pre-train their architectures. It is the easiest modality to collect and use as it provides a lot of information and context around the action. However, due to huge intra-classes differences, it is highly sensitive to viewpoint, background and illumination conditions. Last but not least, dealing with high data volume such as raw videos makes the architectures expensive computationally which can be an issue when aiming at real-time scenarios.

2.2.2 Depth data

Depth maps are images in which the pixel values describe the distance between a given viewpoint and the scene's points. A depth camera is the sensor device that is used to create a depth image. The main advantage of depth information is that it provides 3d structural information and geometric shape information of the scene compared to raw RGB. However, it lacks colors and texture information which is problematic as it has been shown that convolutional neural networks tend to classify images by texture rather than by shape [Hermann et al., 2020]. Another drawback of the Depth data modality is that it has a limited workable distance which constraints the usage of depth data in non-controlled, open environments. Due to the lack of texture information, the depth modality is most of the time used in combination with its corresponding RGB stream as they are complementary in terms of provided information: *e,g* RGB+D. The question now resides in the fusion algorithm strategy (early, intermediate, late... [Guerry et al., 2017]) used to efficiently combine RGB stream and depth data as presented in Fig 2.7.

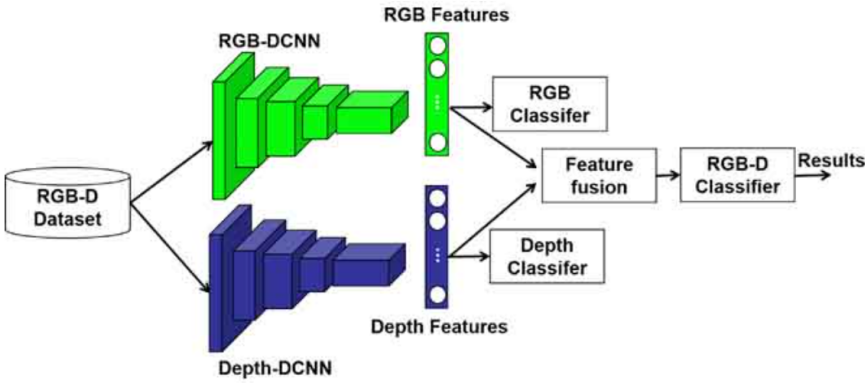


Figure 2.7: An Architecture of decision level fusion of RGB stream and Depth modalities. Picture credit [Gao et al., 2019]

Due to the availability of low-cost and reliable sensors during the last decade such as the Microsoft's Kinect [Zhang, 2012], there has been a rise in the domain of multi-modal RGB+D deep learning approaches for action classification [Imran and Kumar, 2016, Wang et al., 2018, Wang et al., 2020a].

However, most of the time, datasets with depth data are recorded in controlled environments: [Shahroudy et al., 2016, De Smedt et al., 2017]. In the context of pedestrian intention prediction, at the beginning of the thesis, there was not a single academic dataset containing depth data from stereo vision for urban traffic environments. Therefore, we will not expand more on the fusion possibilities between depth, RGB and even skeleton modalities. It would have been possible to use depth estimation algorithms from monocular vision such as P3Depth [Patil et al., 2022], AdaBins [Bhat et al., 2021] or TransDepth [Yang et al., 2021] but it would be at the cost of a computationally expensive task.

2.2.3 Infrared data

Infrared sensors do not need to rely on external ambient light, and thus are particularly suitable for any tasks performed in dark environments, when the lightning conditions are not sufficient enough for other perception modalities to be applicable such as during the night. However, infrared images may suffer from low contrast and low signal-to-noise ratio, which makes it challenging to consider, for robust human action recognition. Moreover, infrared cameras are sensitive to sunlight. Overall infrared cameras should be considered as a specific use case sensor: when illuminations conditions are not good enough to process efficiently what the camera sees. We will not expand much on the architectures used for gesture recognition via infrared data. Firstly because they are very similar to those used in RGB streams: RNN-based models [Kawashima et al., 2017, Imran and Raman, 2019] two-stream models [Mehta et al., 2021] and 3D convolutions [Shah et al., 2018]. Secondly because throughout the thesis, we do not consider nightly environments to be one interesting enough use case: the tasks of predicting pedestrian intentions is complicated enough with good conditions, the point of adding a specific use case requiring a specific sensor would not help much.

2.2.4 Point Clouds

A point cloud is a set of data points in space. The points may represent a 3D shape or object under a spatial reference system. There are two main ways to obtain 3D point cloud data, using 3D sensors such as LiDARs, or using image-based 3D reconstruction. As a 3D data modality, point cloud can efficiently represent the latent geometric structure and distance information of object surfaces, which provide additional cues for gesture recognition. Nevertheless, similarly to depth data, point clouds are lacking color and texture information. Secondly, they are highly complex structures and processing all the points within the point cloud sequence to leverage the spatio-temporal textures of a gesture is often computationally expensive. One straightforward technique for extracting spatio-temporal information from point cloud sequences is to transform point cloud sequences to 3D point clouds and use static point cloud methods (e.g. PointNet++ [Qi et al., 2017] got its 3 dimensional variant 3DV-PointNet++ [Wang et al., 2020b]). However, by transforming point cloud sequences into a static 3D point cloud, one might lose spatio-temporal information in the process. Some methods, attempted to extract dynamic features from point cloud sequences, either by disentangling space and time in point cloud sequences [Min et al., 2020, Fan et al., 2021], or by leveraging both time and space components conjointly [Liu et al., 2019, Fan et al., 2022]. However, similarly to the depth modality, we will not expand much more on the Point Clouds modality as it would not have been possible to use considering the current state of academic datasets for pedestrian intention prediction.

2.2.5 Pose Kinematics

The detection and pose estimation of humans is the first and necessary step in pose-based action recognition, of which posture analysis is an essential component. Nowadays, pose estimation approaches are not limited to the use of motion capture systems or depth cameras. RGB data can be used to infer 2D body poses [Cao et al., 2017], 3D body poses [Martinez et al., 2017] and even track people in real-time [Xiu et al., 2018]. This breakthrough has stimulated the skeletal modality interest since it proved

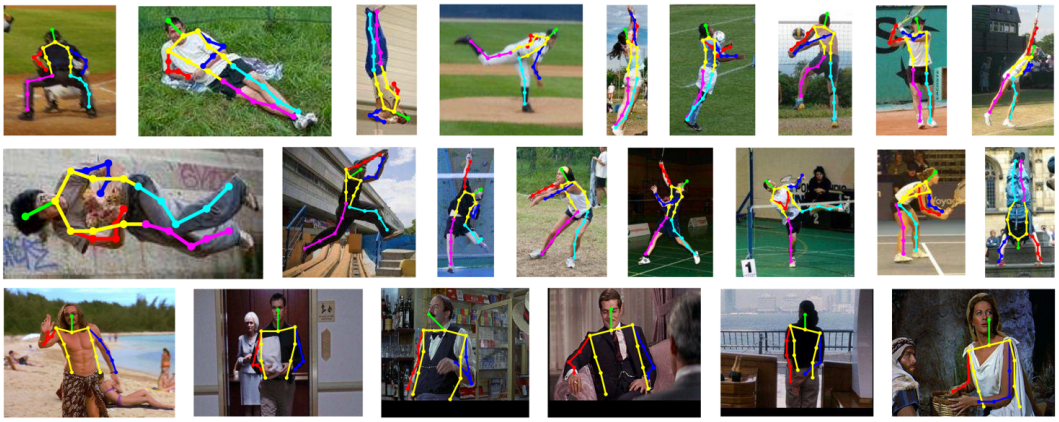


Figure 2.8: Examples of poses. Picture credit [Babu, 2019]

to be sufficient to describe and understand the motion of a given action without any background context. This has made pose-based action recognition preferred over other modalities on a huge amount of real-time scenarios for human action recognition such as human-robot interaction [Mazhar et al., 2018, Bujalance Martin and Moutarde, 2019], medical rehabilitative applications [Mousavi Hondori and Khademi, 2014, Chang et al., 2011] and even sports analytics [Wang et al., 2019, Zecha et al., 2018].

Figure 2.9: Example of difficult cases for skeletonisation algorithms: clothing and brightness on the left, occlusion and scale on the right.

However, the detection task is complicated by the variability of people's appearance (clothing, pose, etc.), as well as by occlusion phenomena, due to the crowd and the setting, or due to problems of scale in the image. Approaches built on deep learning methods are far more promising in overcoming these difficulties than are statistical approaches. Since [Toshev and Szegedy, 2014], most pose estimation approaches have universally adopted convolution networks as their main building block, largely replacing statistical approaches. In the current state of research in 2D multi-person pose estimation, two types of methods are deeply differentiated: Top-Down methods and Bottom-up methods, those two approaches are detailed on Fig 2.10.

2.2.5.1 Bottom Up Approaches

Bottom up approaches are generally based on matching algorithms, graph theory, combinatorial optimization or even greedy algorithms allowing to assign to each keypoint a unique person in the image once all the keypoints have been obtained thanks to convolutive approaches.

Figure 2.10: **Top down**: consists of adding a person detector in order to identify all the articulations (keypoints) of each person and then estimate the pose according to them. **Bottom up**: consists of detecting all the keypoints in the image (*i.e* the limbs of each person), then associating these keypoints with their respective owners. Picture Credit [BeyondMinds, 2020]

For instance, [Pishchulin et al., 2015] propose to extract candidate keypoints thanks to classical architectures such as Faster RCNN [Ren et al., 2015] or Dense CNN [Huang et al., 2017] and then solve the problem of classification (e.g. elbow, knee, head...) and allocation of these keypoints thanks to linear programming and a combinatorial simplex optimization algorithm. [Cao et al., 2017] propose an approach that detects keypoints in the image using a two-branch convolutional architecture: one inferring a set of 18 heatmaps, each representing a particular part of the human pose skeleton, and the other branch inferring a set of 38 *Part Affinity Fields* representing the degree of association between the keypoints. Bipartite graphs are then generated based on these outputs and the Hungarian algorithm is used to prune the graph and thus optimally assign each keypoint to a single person. [Newell et al., 2017] propose an approach that teaches a network to simultaneously produce keypoint detections and assignments thanks to two cost functions: one for detection and one for matching. [Insafutdinov et al., 2016] propose a bottom up approach able to work on a sequence of images in a sequential way and are no longer restricted to the frame by frame approach of the previous methods. After having detected the keypoints of each frame, they are associated in the form of a spatio-temporal graph, the problem of assigning keypoints to a person is then assimilated to a minimum cut problem in graph theory. Finally, [Kreiss et al., 2019, Kreiss et al., 2021] propose a multi-person 2D human pose estimation that addresses failure modes that are particularly prevalent in the transportation domain, *i.e.* crowded images in low resolution with partially occluded pedestrians that occupy a small portion of the image.

2.2.5.2 Top Down Approches

Unlike Bottom Up approaches which rely on the quality of their matching algorithm, Top Down methods rely on the inference quality of their person detection model. Computationally speaking, these methods increase significantly in execution time according to the number of people in the image but are gener-

ally more robust. Mask R-CNN [He et al., 2017], initially a segmentation model, can be modified at the output of the mask to obtain poses. The basic architecture first extracts features from an image using convolutions. These features are used by a Region Proposal Network (RPN) to obtain candidate bounding boxes for the presence of objects. By combining the information about the location of the person thanks to the bounding boxes and the keypoints obtained from the mask, we obtain the skeleton of the human pose for each person present in the image. [Iqbal and Gall, 2016] show that the multi-pose estimation problem can be formulated as a set of association problems for each of the detected persons in the image. Thus, for each detected person, one can generate a skeleton using a single-person pose estimator inside the bounding box position of the given protagonist. Since this approach does not take into account occlusion or truncation problems for each of the obtained bounding boxes, they reuse the linear programming system of [Pishchulin et al., 2015] on each of the graphs inferred for each bounding box. Since the size of the graphs is much smaller than in the Bottom Up approach, the speed of inference obtained is globally better. [Papandreou et al., 2017] propose a top down approach, where keypoint estimation is not performed by regression for each keypoint but by estimating heatmaps and a magnitude vector allowing to best target the keypoint position in the heatmap. This approach allows to potentially obtain several keypoints of the same class in the same bounding box and thus, to a certain extent, overcome the occlusion problem for the estimation of poses in a 2D space. AlphaPose [Fang et al., 2016] propose an approach to facilitate pose estimation when the obtained person detection bounding boxes are inaccurate. Since pose estimation is performed on the boundig box obtained from a detection algorithm, errors in the localization of these bounding boxes when using the detector can result in a non-optimal operation of the pose extraction algorithm. The authors therefore propose the use of a Symmetric Spatial Transformer Network (SSTN) to extract a quality area from an inaccurate bounding box. A pose estimator is then used on this extracted region and the resulting coordinates are transformed to match the original space.

2.2.5.3 The question of dimension

One of the main limitations of a 2D approach is the ability to treat occlusions between pedestrians in a 2-dimensional space. Therefore, in order to improve pose detection, the question of adding a third dimension may arise. The methods for 3D pose estimation are much less mature than those for 2D pose estimation. One of the main reasons to date would be the lack of available reliable datasets [Yang et al., 2018a]. However, some types of approaches can be discerned: estimating a 2D pose and reconstructing a 3D pose, directly performing the regression of a 3D pose, or treating the 3D pose estimation problem jointly or even iteratively with the regression of a 2D pose.

[Chen and Ramanan, 2016] propose, for example, an architecture passing through the estimation of an intermediate pose in 2D and estimating the value of the pose depth using the k-nearest neighbors' algorithm on a 3D pose database. [Martinez et al., 2017] show that translating points in a 2-dimensional space into a 3-dimensional space is a task that can be solved with a simple multilayer perceptron (MLP). Analogously, [Nie et al., 2017] predict the depth of human joints based on their 2D locations using recurrent approaches (Long short-term memory - LSTM).

On the other hand, [Li and Chan, 2014] propose a convolutional approach directly performing a regression of the skeleton in 3 dimensions. [Sun et al., 2017] propose to base their regression on the joins of the keypoints and not the keypoints of the skeleton. [Tekin et al., 2016] train an auto-encoder on

skeletons with a sparse latent space of greater dimension than the input's. Then they regress with a convolutional neural network taking as input the image corresponding to the skeleton, the values of the latent space for a given training instance and use the decoder part in order to retrieve the pose in 3D. [Mehta et al., 2018] propose a 3D pose regression method reusing the same bottom-up approach as [Cao et al., 2017] to associate keypoints and define *occlusion-robust pose-map* (ORPM), allowing to infer the whole-body pose even in case of strong partial occlusions by other persons or objects.

Finally, [Simo-Serra et al., 2013] propose to partition the 3D pose estimation problem by simultaneously estimating the 2D and 3D poses and then combining the results obtained. [Tome et al., 2017] propose an iterative refinement method in which 3D inferences help refine and improve 2D estimates, and then translate the prediction from 2-dimensional space into a 3-dimensional space. In [Rogez et al., 2019], the estimation of 2D/3D human poses is performed jointly through a localization-classification-regression (LCR-Net) architecture. The localization is performed thanks to an RPN, suggesting candidate poses at different locations of the image, and a classifier evaluates the plausibility of the different pose proposals. Finally a regression refines the pose proposals in 2D and 3D. Similar work has been proposed by [Benzine et al., 2019, Benzine et al., 2020] to efficiently extract conjointly 2D and 3D poses for a possibly large number of people at low resolution. Therefore trying to scale up the current focus of single-person pose estimation or estimation of 3D pose of few people at high resolution.

2.2.5.4 The issue of sequentiality

On the basis of the multi-person pose estimators described above, it is natural to seek to extend them from the frame-by-frame approach and thus take into account the sequential information present in the sequence.

Many of the approaches are based on the graph partitioning work of [Pishchulin et al., 2015] and [Insafutdinov et al., 2016] for the frame by frame approach. The notable difference to take into account sequentiality is to extend the graph of keypoints in space at each frame into a spatio-temporal graph. However, current linear programming solvers take a long time to converge and real-time usage becomes complicated or impossible considering the size of the graph for a video. A current line of research tends to explore more efficient and scalable top-down solutions by first estimating the pose of several people in each image and then linking them in terms of appearance similarity and temporal relationship. Thus, [Xiu et al., 2018] propose PoseFlow, a sequential approach to AlphaPose [Fang et al., 2016] by maximizing the overall confidence of pose inference for a temporal sequence. First, a top-down estimation of the poses for each image is performed. Pose Flows are constructed by associating poses that correspond to the same person across frames through an estimate of the distance between poses. Using a sliding window, the poses of each person in the frame are normalized to the previous and next positions in the video. [Ning and Huang, 2019] propose a top-down approach where the matching of poses to a person refers to a distance between two poses based on the optical flow. [Xiao et al., 2018] propose a top-down approach in which the identification of a person over time is based on two complementary pieces of information: spatial coherence and pose coherence. Tracking and identification are performed using a geometric convolutional Siamese network to determine the degree of similarity between features extracted from two poses. [Raaj et al., 2019] reuse the Part Affinity Field principle for static frames of [Cao et al., 2017] in a sequential form thanks to a recurrent architecture where the network uses as input

the heatmaps of the previous frames to estimate those of the current frame.

Overall, Top Down sequential methods based on pose matching should be prioritized because the location of a person may vary according to a sudden change of camera angle, while his or her pose will remain almost the same, in order to obtain a better tracking of the protagonist of the scene and not to mix the skeletal kinematics of two people at different moments in the sequence but close in space, which would render the second part of the approach: biased by using wrong data.

2.3 Poses, Actions and Trajectories

In our context of interpreting actions from estimated poses kinematics (see section 2.2.5), we define here the different concepts derived from pose-based modeling, following the nomenclature proposed by [Picard, 2011] and extended by [Barnachon, 2013]:

A gesture is described as a movement that conveys a purpose. Unlike a mechanical system, the distinction comes from the conscious will of the human being who consciously produces the gesture.

An articulation/joint; a, is a given n-dimensional point in a reference frame. With $n \in [2;3]$, for readability purposes we continue the specification of each concepts for $n = 3$.

$$a = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.1}$$

A pose, P, is the n-dimensional position of all the given articulations at a given time t. We define a pose as the union of the positions of the articulations for a given timestamp t as:

$$P(t) = \bigcup_{a \in \mathcal{A}} a(t) = \bigcup_{a \in \mathcal{A}} \begin{bmatrix} x_a(t) \\ y_a(t) \\ z_a(t) \end{bmatrix} \tag{2.2}$$

Where $P(t)$ is the given pose at time t, $x_a(t), y_a(t), z_a(t)$ are respectively the coordinates on the X, Y, Z axis at time t, $\mathcal{A}$ is the set of articulations of interest and $a(t)$ the articulation at a time t. A pose $P(t)$ can also be defined under the form of a $|\mathcal{A}| \times n$-shaped vector:

$$P(t) = \begin{bmatrix} a_1(t) \\ a_2(t) \\ \vdots \\ a_{|\mathcal{A}|}(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ y_1(t) \\ z_1(t) \\ \vdots \\ x_{|\mathcal{A}|}(t) \\ y_{|\mathcal{A}|}(t) \\ z_{|\mathcal{A}|}(t) \end{bmatrix} \tag{2.3}$$

An action, A, is a temporal sequence of poses sharing a common semantic, *i.e.* a movement conveying a meaning. An action can therefore be defined as its corresponding, ordered, sequence of poses from

t_0 to t_N where N is the finite number of poses in the sequence:

$$A = \bigcup_{t=t_0}^{t_N} P(t) = \begin{bmatrix} a_1(t_0) & \dots & a_1(t_N) \\ a_2(t_0) & \dots & a_2(t_N) \\ \vdots & \ddots & \vdots \\ a_{|\mathcal{A}|}(t_0) & \dots & a_{|\mathcal{A}|}(t_N) \end{bmatrix} = \begin{bmatrix} x_1(t_0) & x_1(t_1) & \dots & x_1(t_N) \\ y_1(t_0) & y_1(t_1) & \dots & y_1(t_N) \\ z_1(t_0) & z_1(t_1) & \dots & z_1(t_N) \\ \vdots & \vdots & \ddots & \vdots \\ x_{|\mathcal{A}|}(t_0) & x_{|\mathcal{A}|}(t_1) & \dots & x_{|\mathcal{A}|}(t_N) \\ y_{|\mathcal{A}|}(t_0) & y_{|\mathcal{A}|}(t_1) & \dots & y_{|\mathcal{A}|}(t_N) \\ \underbrace{z_{|\mathcal{A}|}(t_0)}_{\text{Pose 1}} & \underbrace{z_{|\mathcal{A}|}(t_1)}_{\text{Pose 2}} & \dots & \underbrace{z_{|\mathcal{A}|}(t_N)}_{\text{Pose } N} \end{bmatrix} \tag{2.4}$$

Trajectories, the lines of the matrix representation of an action are the expression of the trajectories of the articulations over time, on each axis of the n-dimensional world. The trajectories of an action, carry more information about the action than a pose. This is the transposition of Johansson's postulate [Johansson, 1973]. It is therefore possible to see an action as a set of n-dimensional or one-dimensional trajectories:

$$A = \bigcup_{t=t_0}^{t_N} \bigcup_{a \in \mathcal{A}} a(t) = \bigcup_{t=t_0} \bigcup_{a \in \mathcal{A}} \begin{bmatrix} x_a(t) \\ y_a(t) \\ z_a(t) \end{bmatrix} \tag{2.5}$$

with $a(t)$ the n-dimensional trajectory of the articulation a over time, $x_a(t), y_a(t) and z_a(t)$ the one-dimensional trajectory of the articulation a over time on the X, Y, Z axis.

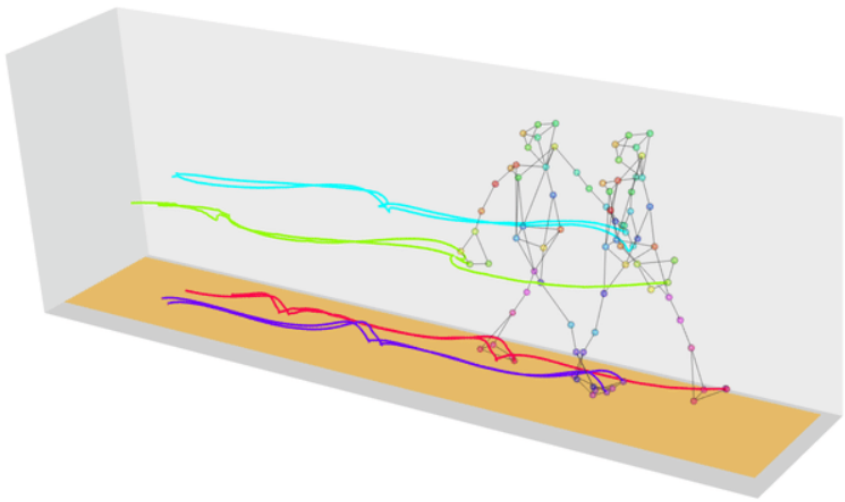


Figure 2.11: Trajectories of the joints of a walking skeleton. Picture credit [Olsen et al., 2018]

Action recognition is a classification problem. The goal is to assign a label to the temporal sequence of poses, *e.g*, assuming a training set of actions and their respective labels: $\mathcal{D} = \{(\boldsymbol{A}_i, y_i)\}_{i=1}^{N}$ composed of N training samples. The goal is to learn a mapping function f that can correctly predict the label y of the input action A.

2.4 Overview of Skeletal Sequence Modeling with Deep Neural Networks

In the previous sections, we listed the different types of existing algorithms to obtain the skeleton of a person based on a RGB stream. We then defined the concepts to move from pose extraction to the analysis of poses kinematics. In this section, we review a list of approaches based on "in-depth" learning of action recognition on skeletal data. Those approaches for skeleton-based action recognition can be split into four categories: recurrent-based architectures, convolutions-based architectures, attention-based approaches and graph-based approaches.

Whereas all those approaches were all designed to perform the same task: action recognition, their very composition in terms of layers provides a different type of relational inductive bias: how do they hierarchically process features? The performance of action recognition approaches not only results from the training data but also from their sequence-focused design and their corresponding assumptions towards the data. As shown in Table 2.1 the choice of a good neural network architecture and its corresponding inductive biases is crucial to model sequences.

Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations

Table 2.1: Various relational inductive biases in standard deep learning components. *An inductive bias allows a learning algorithm to prioritize one solution (or interpretation) over another, independent of the observed data.*[Battaglia et al., 2018]

When training deep learning models, anything that imposes constraints on the learning trajectory can be considered as an inductive bias. Any non-relational inductive biases used in deep learning include for instance: activation functions, regularization's such as weight decay, dropout, batch and layer normalization, data augmentation, optimizers...

Given all the possible combinations and effects of each non-component inductive bias that can only be evaluated empirically, no trend is easily distinguishable in terms of identifying the best architecture to model sequence for skeletal action recognition. However, each of those categories has its pros and cons by design that we will explain in the following sections.

2.4.1 Fully Connected Neural Networks

According to the universal approximation theorem [Hornik, 1991], any bounded function can be approximated as well as one wants with a shallow Neural Network containing only one hidden layer. As such, one may even use a trivial feed-forward neural network such as a Multi-Layer Perceptron (MLP) to model sequences, like any other type of data. However, the stronger the inductive bias, the better the training sample efficiency. Considering that fully connected neural networks have weak relational inductive bias, the design of those architectures do not emphasizes a minimal a priori assumptions about the data and therefore would lead to a data-intensive training compared to approaches designed to consider the temporal phenomena of action recognition. For instance, when modeling time series in real world scenarii,

sequences are usually not stationary: the interpretation of a given feature might depend on earlier features or the timestamp t they appeared at. Such assumptions about the data is not taken in consideration with MLPs. Similarly, their general relation pattern (All-to-all) does not naturally benefit from regularities, like periodicity, that may exist in time series data. Finally, fully connected layers require a fixed number of inputs, whereas in real world scenarii, the same actions can be carried out under different time windows sizes. Due to theses shortcomings, and limitations when compared to more complex neural network architectures to perform gesture recognition, vanilla multi-layer perceptrons are hardly ever used on their own [Li et al., 2019b]. Nevertheless, it is worth noting that when fully connected neural networks are coupled with attention mechanisms (see section 2.4.4), such as in the Transformer architecture [Vaswani et al., 2017], those models achieve similar / better performance in domains involving sequential data processing than convolutional and recurrent neural networks. However, those approaches are more data-intensive in regards to the amount of available training data compared to architectures providing assumptions about the input data.[3] For instance, for related sequential task such as Natural Language Processing, Transformers perform extremely well on many tasks with enough training data and computation [Devlin et al., 2018, Keskar et al., 2019], but several studies have shown that LSTMs can perform better than Transformers on tasks requiring sensitivity to hierarchical structure, especially with limited amount of training data [Tran et al., 2018, Dehghani et al., 2018].

2.4.2 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) combine feedforward neural networks with hidden states which one can view as dynamic memories. This allows RNNs to exhibit temporal dynamic behavior[4] where the hidden state is used to process variable length sequence of inputs. In contrast to Fully Connected Networks, RNNs can process sequences of arbitrary length, which made them the reference approaches for sequence modeling in speech recognition, digital signal processing, video processing and natural language processing. Similarly, most deep-learning approaches for gesture recognition also use recurrent cells such as LSTMs [Hochreiter and Schmidhuber, 1997] or GRUs [Cho et al., 2014]. For those approaches, the skeleton is represented in the form of a sequence and state neural networks are applied to it. The difference between each approach resides in designing novel neural network architecture based on RNNs, LSTMs or GRUs to achieve action recognition. For instance, [Du et al., 2015b] have classified the time series skeletal data by hierarchically combining the predictions of several RNNs subnetworks. Treating every body part of the skeleton sequence independently, then aggregating more body parts together until the final classification. [Shahroudy et al., 2016] propose a similar approach by using five different part-aware Long Short Term Memory (LSTM) networks based on subsets of joints, to capture local information about the skeleton sequences. Instead of aggregating the networks such as in [Du et al., 2015b], a fusion algorithm is used to merge them. [Shukla et al., 2017] propose a hierarchical recurrent architecture roughly equivalent to [Du et al., 2015b] and [Shahroudy et al., 2016] but reduce manually the number of joints at the input of the model, some of them being considered superfluous and carrying little information. This reduction in the number of input joints then leads to a reduced set of parameters and reduces the model inference time without degrading the quality of the classifier.

[3] Therefore unusable when learning classification using little data.
[4] *explicit temporal modeling*

[Wang and Wang, 2017] has proposed a method based on two streams of RNNs, one to learn the spatial dependencies in the kinematics of skeleton joints, the other one to learn the temporal dependencies of the pose kinematics. These are then combined following a late fusion strategy with an end-to-end trainable network. In order to achieve better performance for skeleton-based action recognition with RNNs, some studies introduce new features for skeleton sequence instead of using the regular trajectories of the articulations in the Cartesian coordinate system. For instance, [Liu et al., 2016] use a tree-structure-based traversal method for better representation of human skeletons while introducing a gating mechanism within LSTM cells to improve recognition robustness. From the articulations positional information, [Zhang et al., 2017b] generate eight geometric indicators and evaluate them with a three-layer LSTM network. [Zhang et al., 2017a] propose an adaptive recurrent network with a LSTM architecture, allowing the network to adapt to the most appropriate end-to-end observational viewpoints in order to manage large variations in the orientation of actions. [Avola et al., 2018] exploit the geometric characteristics of the angles of the joints learned with a LSTM architecture. However, recurrent cells are relatively slow and difficult to train due to the well-known gradient vanishing and exploding problems and hardly manage to learn long-term dependencies [Li et al., 2018]. Moreover, studies using RNNs when dealing with action recognition tasks have the issue of placing lesser emphasis on spatial features of the skeleton sequence as they pay much more attention and place much more importance on the trajectories of each articulation and therefore the sequential part of the problem. Therefore, recurrent architectures might fail to capture the spectrum of actions that could be inferred based on spatial poses only.

2.4.3 Convolutional Neural Networks

Since recurrent cells are relatively slow and difficult to train and to use in real-time due to their lack of parallelization, Convolutional Neural Networks (CNNs) have become an interesting solution given their advantages in terms of parallel computing, and efficiency in learning characteristics and speed.

CNNs are a class of feedforward neural networks where the neural network uses a convolution operation in place of a matrix multiplication. CNNs tend to exhibit good performance on data with a grid-like topology such as images and time series, as they respectively can be viewed as a field of vectors taking values over an evenly spaced 1D grid (time) or 2D grid (spatial pixel grid). Initially designed for 2D grids such as images, CNN can be applied over sequences, as their relational inductive bias will find temporal regularities via their locality. The interpretation of a given feature might depend on earlier/future features or the timestamp t they appeared at, which can be viewed as a "time" neighbor of the current timestamp.

When dealing with action recognition tasks with Convolutional Neural Networks, two kinds of solutions arise: transforming the input data in a 2D-grid-like manner or a 1-D grid-like manner.

For instance, convolutions can be performed on skeletons kinematics represented as pseudo-images, so that standard 2D convolutions can be applied, or any other spatio-temporal version of CNNs such as 3D convolutions. Since skeletal data are small elements, it is possible to organize a sequence of skeletal features chronologically in an image that retains the original information of the skeletal kinematics as illustrated in Figure 2.12.

The general idea of this type of approach is to structure the data in order to give them the expected form (a sequence of images) and thus classify these images using standard computer vision methods. Such motion formalism to represent skeletal sequences by compact image-like inputs was first proposed

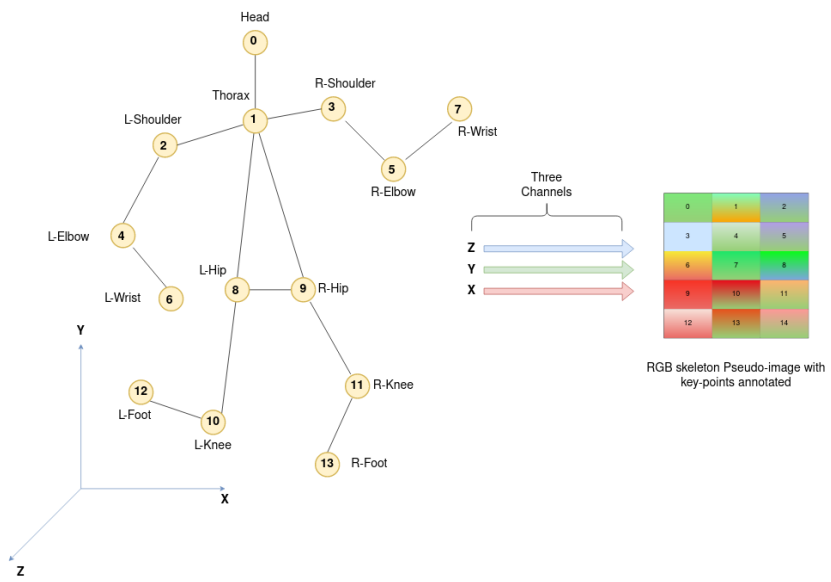


Figure 2.12: Organization of the 3D skeleton data structure into a three-channel image (RGB)

by [Elias et al., 2015], alongside with [Du et al., 2015a] where a special insistence has been given to features representation and data normalization to improve instance indexing. Pulling in the same direction, [Wang et al., 2016b] propose skeleton sequences that have been further encoded into the Hue Saturation Value (HSV) space. To overcome the issue of the semantic continuity of pixels in the generated pseudo-images, [Li et al., 2017b] proposes a skeleton transformer layer before the CNN's to learn the optimal representation of pseudo-images for the network. [Li et al., 2017a] used the pairwise distances between joints encoded into RGB images, known as the Joint Distance Map as input features to account for view invariance. [Ke et al., 2017] propose to transform a skeleton sequence into three video clips, the CNN characteristics of the three clips are then merged into a single characteristics vector which is finally sent to a softmax function for classification. [Pham et al., 2018] propose to use a residual network ([He et al., 2016]) with for input the transformed normalized skeleton in the RGB space. [Cao et al., 2018] propose to classify the image obtained thanks to gated convolutions. Finally, [Banerjee et al., 2020] has reduced the number of channels to one in grayscale instead of three in RGB. Moving away from the image domain while keeping the notion of sequential modeling via convolutions, other CNN-based approaches use them in 1D format: [Bai et al., 2018] show that convolution networks can match or even surpass the performance of recurrent networks for typical sequential modeling tasks. Therefore, [Devineau et al., 2018] propose an architecture based on parallel convolutions capable of capturing features at different temporal resolutions. This results in a three-branch convolutional model that takes as input the positions of skeletal joints at different speeds and the distances in pairs between joints. [Weng et al., 2018] propose a deformable convolutional neural network with one-dimensional convolutions capable of discovering combinations of information-carrying joints to avoid joints whose semantics contribute little to the model. [Yang et al., 2019] propose a Double-feature Double-motion network where skeletal kinematics are processed either in the Cartesian coordinate system under different time shifts with 1D CNN or processed in a location-viewpoint invariant manner.

2.4.4 Spatio-temporal Attention

Human perception focuses on the most relevant parts of an image in order to acquire information to understand its semantics. For machine learning, this phenomenon is artificially recreated by a mechanism of attention: conceptually, attention can be interpreted in a broad sense as a vector of weights of importance. In the context of action recognition, attention can be used to weight the importance of certain moments of the action in order to classify it, or to weight the importance of certain skeletal joints. Nevertheless, it is not a deep learning component but rather a mechanism used to provide pertinent features to the other components of a neural network. Most of the existing works combine the attention mechanism with regular deep learning components for time series modeling. For instance, [Fan et al., 2019] propose an attention mechanism for multiview fusion of skeletons preprocessed by LSTMs cells. [Maghoumi and LaViola Jr, 2019] propose to stack GRUs with a global attention mechanism as well as two fully connected layers. [Song et al., 2017] propose a model based on LSTM and RNN and combine spatial and temporal global attention: a network focuses on the discriminating articulations of each frame, the other network weights the attention levels of the results for each instant in order to focus on the important frames. [Fan et al., 2019] use action information from multiple viewpoints to improve recognition performance and provide an attention mechanism for multi-view fusion of skeletons sent to LSTMs. Similarly, it is also possible to use attention with convolutions. Thus, [Hou et al., 2018] propose a convolutional network learning different levels of attention for each spatio-temporal feature extracted by the convolution filters for each frame of the sequence.

2.4.5 Graph Neural Networks

The evolution over time of the skeleton of the human body can be considered in the form of a dynamic graph. So far, research in deep-learning for action recognition on skeletal data has focused mainly on Euclidean data. The non-Euclidean nature of data in graph format makes the use of basic operations, such as convolution, difficult to perform. However, convolutions have by definition the ability to extract local spatial features and could use the skeleton data structure in graph format for the classification of human actions. Such ability fits perfectly to Graph-type data structures since they are, by definition, locally connected structures: the set of neighbors of a node.

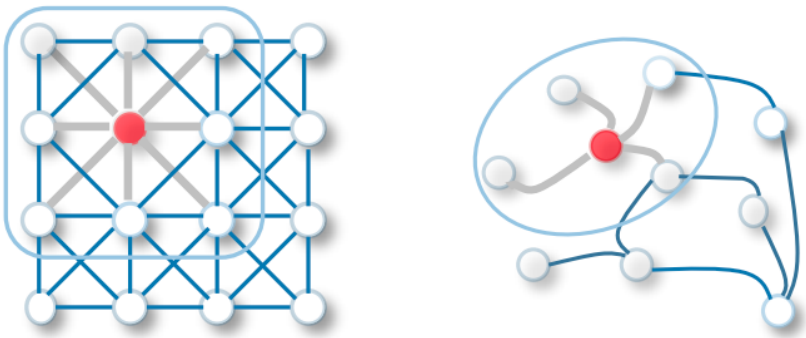


Figure 2.13: **(left)** Convolution on 2-D grid-like data. The number of neighboring nodes is a fixed number determined by the filter size. **(right)** Generalized convolution operation on unstructured data. The number of neighboring nodes, determined by edge connectivity may vary from node to node. Picture credit [Wu et al., 2019b]

To this extent, representing the skeleton in the form of a graph can have the advantage of not exploiting non-existent neighborhood links between joints, but preserving coherent spatial semantics for the skeleton compared to approaches using Euclidean data structures such as RNNs or CNNs. Geometric Deep-Learning [Gori et al., 2005, Scarselli et al., 2008, Bronstein et al., 2017] refers to techniques attempting to generalize deep structured neural networks to non-Euclidean domains such as graphs. [Wu et al., 2019b] provide a state of the art on geometric deep-learning and propose a taxonomy to differentiate geometric networks into four categories: recurrent, convolutional, auto-encoder and spatio-temporal. Such formalism to represent the skeletal sequences in a non-euclidean data structure while applying deep-learning to unstructured data was first proposed by [Li et al., 2018]. [Zhang et al., 2018] propose to apply convolutions on the edges of a graph corresponding to skeletal bones in order to preserve spatial semantics. Thereupon, architectures trying to combine both spatial and temporal graph convolutions throughout one network were designed [Wu et al., 2019a]. Similarly, [Yan et al., 2018] extend the spatial convolutions of graphs into spatio-temporal convolutions. They propose a convolutional spatio-temporal approach including time-bound joints in the convolutional block in addition to spatially bound joints. Self-attention mechanisms can be utilized to improve the modeling capacity of graph neural networks as well [Shi et al., 2019, Li et al., 2019a]. Finally, [Si et al., 2019] propose to cumulate attention to a CNN-LSTM geometric network, capitalizing all the approaches presented previously in a single network. While many skeleton-based action recognition methods adopt graph convolutional networks to extract features on top of pose kinematics, the boost in model accuracy observed is hindered by potential drawbacks such as computational complexity, resource consumption or interoperability. Current research trends aim at reducing the computational complexity and ressource consumption of graph neural networks for skeletal action recognition. For instance, [Ye et al., 2020] propose to combine both graph neural networks and standard convolutions in an euclidean grid space to reduce the computational costs of the overall approach. [Cheng et al., 2020] propose a new form of graph convolution that provides improvements in terms of computational complexity and memory consumption of graph neural networks by reducing by nearly a factor of 10 both of those performance indicators compared to standard graph convolution. Nevertheless, when considering the model speed as one priority, the boost in model accuracy compared to architectures using euclidean data structure is questionable. Specifically considering that most of the standard deep-learning operations are already implemented in any deep learning frameworks and also in any neural hardware solutions for embedded devices: the knowledge of the optimization of euclidean data structure networks is conserved compared to approaches based on Graph Networks where basic operations need to be redefined and one might lose speed efficiency in the process.

2.5 Representations, Inductive Biases and their roles during classification with little data

2.5.1 Importance of Explicit Temporal Modeling

2.5.1.1 Introduction

We have seen in section 2.4 that existing skeletal action prediction models focus mainly on the sequential modeling part of the problem by modeling the trajectories of joints over time. Moreover, those

approaches rely heavily on deep-learning networks and their inductive biases to learn informative representations of those trajectories [5] by stacking layers, leading to more and more complex architectures over the years. Since deep learning approaches depend heavily on the quantity and quality of data where the performance of approaches scales up with the amount of training data, the current paradigm does not encourage the community to study and improve the capabilities of deep networks for tasks with little data available. In this work, we propose to go back to *"It is all about embedding and standardization in machine-learning"*: once one finds a way to standardize and represent data in a more adequate way, any classifier might be able to obtain good results as long as the input data is informative. We start from several assumptions:

- The primary role of the hidden layers is to realize a composition of non-linear transformations in the hope of finding an embedding adapted to the data format that preserves a maximum of its semantics from representation to output. Hence, one could for instance enforce the representation of a specific designated hidden layer by combining classical statistical representation and learned representation.
- By addressing the issue of data representation, we discover, not only the mapping from representation to output, but also the representation itself. Therefore, one avoids a "blind" learning of this representation: by optimizing a precise network on a precise dataset and by testing a huge set of hyper-parameters, one is assured of good results from representation to outputs. The more reliable the data representation, the more trivial the architecture used for classification can be and the less prone to over-fitting it becomes. Moreover, we can therefore potentially reduce the size of our network and thus by definition reduce its inference time, which can be interesting in the context of a real-time setting.

Based on previous works, showing that the addition of a non-supervised regularization during a classification problem with little data allows a better generalization of networks [Brigato and Iocchi, 2021]. We question if we can efficiently initialize a network similarly to unsupervised pre-training researches [Ranzato et al., 2006, Hinton and Salakhutdinov, 2006, Hinton, 2007] and combine two different ideas:

- The idea that the choice of initial parameters can have a regularizing effect on the model (and therefore improve optimization).
- The idea that learning about the input distribution can help with learning about the mapping from inputs to outputs. We refer to this as representation learning and the classical method used for such a task is the auto-encoder: a neural network that is trained to attempt to copy its input to its output by learning a representation h in a low-dimensional manifold of a learning example x and approximately recovers x from h through a transition function.

Hence we propose an auto-encoder as presented in Fig 2.14 with a separability constraint term that focuses on two completely different pieces of information in the data:

- The inherent structure of the data captured in an unsupervised manner thanks to the reconstruction capability of the auto-encoder and its abstraction ability. Some of the important and discriminating information in the data set would then be retained in a low-dimensional manifold.

[5] We refer to the concept of gathering knowledge from experience without the need of human operators as semantics.

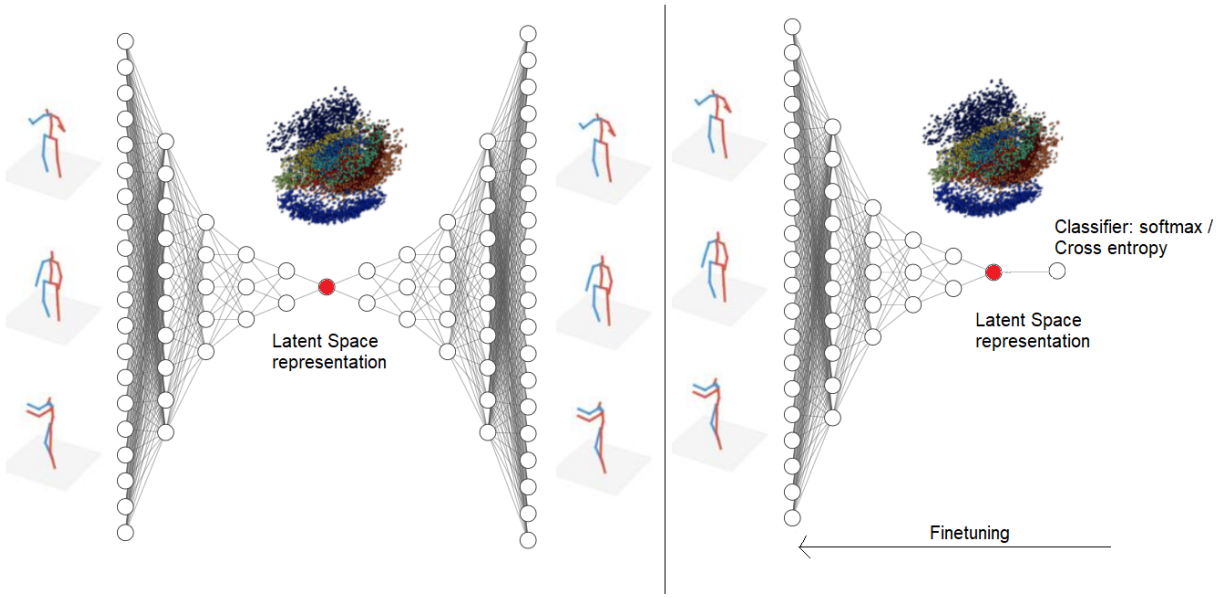


Figure 2.14: Pipeline of the approach: (1) we train an auto-encoder to reconstruct a sequence representing an action according to the evolution over time of the keypoints. We also add a constraint specific to the separability of classes in the latent space. (2) We then extract the weights of the encoder part up to the bottleneck represented in red and add a classifier, which transforms the encoder part into a pre-trained network on the data for action classification.

- A first separability draft of classes thanks to Linear Discriminant Analysis projection of the instances in the latent space to improve the maninfold's representation.

The idea behind our approach is that it leads to a better encoder weights initialization if we combine statistics and semantics as we enforce certain constraints towards the data representation of designated hidden layer: the latent space.

To demonstrate that the question of data representation is almost as important as sequential modeling for such task, and according to the universal approximation theorem [Hornik, 1991](any bounded function can be approximated as well as one wants with a shallow Neural Network), we use the simplest form of an autoencoder, a feed-forward neural network such as a Multi-Layer Perceptron (MLP) to model sequences with no explicit temporal modeling.

2.5.1.2 Formalization

Formally, we define the problem as follows:

$$\min_{\theta_1,\theta_2}, \left\| \mathbf{X} - g_{\theta_2}\left(f_{\theta 1}(\mathbf{X})\right) \right\|^2 \tag{2.6}$$

Equation (2.6) is the usual reconstruction function of an auto-encoder with $\mathbf{X}$ a data matrix, θ_1, θ_2 the parameters of the encoder and decoder blocks and $f(), g()$ are respectively the transition functions such that:

$$\begin{aligned} f_{\theta 1} &: \mathbf{X} \rightarrow \mathcal{F} \\ g_{\theta_2} &: \mathcal{F} \rightarrow \mathbf{X} \end{aligned} \tag{2.7}$$

Where $\mathcal{F}$ is the feature space which can be regarded as a compressed representation of the input

matrix $\mathbf{X}$. We refer to $\mathcal{F}$ as the bottleneck or the latent space of the auto-encoder.

We then add a statistical supervised constraint specific to the separability of classes in the cost function: with $\mathbf{S}$ being the projection matrix of the instances in the latent space obtained with a linear discriminant analysis (LDA) and λ a weighting parameter as presented in equation (2.8):

$$\min_{\theta_1,\theta_2,\mathbf{S}} \left\|\mathbf{X} - g_{\theta_2}\left(f_{\theta_1}(\mathbf{X})\right)\right\|^2 + \lambda \left\|f_{\theta_1}(\mathbf{X}) - \mathbf{S}_{f_{\theta 1}}(\mathbf{X})\right\|^2 \tag{2.8}$$

The given formula is only applicable if $dim(\mathcal{F}) = M - 1$ with M being the number of classes in the dataset. The training method is a simple iterative algorithm, optimizing an appropriate objective function. This algorithm is based on two updating steps according to the scheme written in the pseudo-code below:

Algorithm 1: Auto-encoder with statistical separability constraint training algorithm

Input: data matrix $\mathbf{X}$, ground truth labels y, weighting parameter λ, loss threshold ε
Initialization of the encoder and decoder parameters θ_1 and θ_2;
while $\left\|\mathbf{X} - g_{\theta_2}\left(f_{\theta_1}(\mathbf{X})\right)\right\|^2 + \lambda \left\|f_{\theta_1}(\mathbf{X}) - \mathbf{S}_{f_{\theta 1}}(\mathbf{X})\right\|^2 > \varepsilon$ **do**
  Update θ_1 and θ_2 using the auto-encoder.;
  Update S using Linear Discriminant Analysis on $f_{\theta 1}(\mathbf{X})$ data matrix and y.;
end
Result: θ_1, parameters of the encoder block

We choose the value of λ for the supervised separability constraint part empirically, by modifying its value for different trainings and evaluate its gain for later stages. Once the training of the auto-encoder has been performed, we recover the weights of the encoder part: θ_1 and add a linear classifier, such as a softmax regression classifier right after the bottleneck. In order to use the encoder as a classifier, we train the given modified network by minimizing the cross-entropy loss $\mathscr{L}_{CE}$ between the empirical distribution defined by the training set and the probability distribution defined by the model being trained:

$$\mathscr{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{m=1}^{M} \mathbb{1}_{y_i \in C_m} \log\left(p_{\text{model}}\left(y_i \in C_m\right)\right) \tag{2.9}$$

Where N is the total number of instances, M is the cardinal of the set of classes $\{C_1, \ldots, C_m\}$, $\mathbb{1}_{y_i \in C_m}$ is the value of the indicator function of the i-th sequence belonging to the m-th class and $p_{\text{model}}\left(y_i \in C_m\right)$ is the softmax probability that the i-th sequence belongs to the class C_m. The encoder $f_{\theta 1}(X)$ now learns to provide a representation to the softmax regression classifier and has been pre-trained to do so. Since supervised training of feedforward networks does not involve imposing any condition on the learned intermediate representations, we expect our statistical regularization to make the classification task easier without any explicit temporal modeling.

Architecture Details

In order to evaluate the quality of representation, we chose to take the easiest possible autoencoder architecture: a feed-forward multilayer perceptron. Naturally, it is possible to perform the same work on

convolutional or sequential autoencoders in order to capture the sequentiality of the action or to minimise the size of the network and thus potentially improve the accuracy and the inference time. The architecture consists of an encoder part and a decoder part. The decoder is symmetrical to the encoder part and is defined as such:

- An input layer of dimension 2112 representing the flattened tensor of a complete gesture for 32 time steps, 22 keypoints and 3 dimensions or of dimension 960 for 32 time steps, 15 keypoints and 2 dimensions depending on the input format of the skeletal kinematics.
- 5 dense layers of feature extractions whose dimension decreases by a power of two for each layer (512, 256, 128, 64, 32).
- A bottleneck of dimension $M - 1$, with M being the cardinal of the set of classes for each dataset.

To address the vanishing gradient problem, each perceptron in the given auto-encoder network uses the LeakyRelu [Maas, 2013] activation function.

Optimizer, Batching, Regularizations

The given auto-encoder is regularized with dropout [Srivastava et al., 2014] with $p = 0.1$, L_2 regularization with $\lambda = 10^{-1}$ and batch-normalization [Ioffe and Szegedy, 2015]. For both training phases, we use Adam [Kingma and Ba, 2014], following the hyper-parameters recommendations of the authors. Regarding the number of training epochs, we train the given network with an annealing learning rate that drops from 10^{-3} to 10^{-8} and use early stopping: if the validation loss does not improve during the 50 last steps by more than 0.01%, we stop the training in order to prevent the network from over-fitting the training data.

2.5.1.3 Experiments

Datasets

We select two skeleton-based action recognition datasets, SHREC dataset [De Smedt et al., 2017] and JHMDB dataset [Jhuang et al., 2013] to evaluate our regularized autoencoder from different perspectives (see Table 2.2).

Figure 2.15: Example of a swipe left gesture extracted from SHREC dataset. Picture Credit [De Smedt et al., 2017].

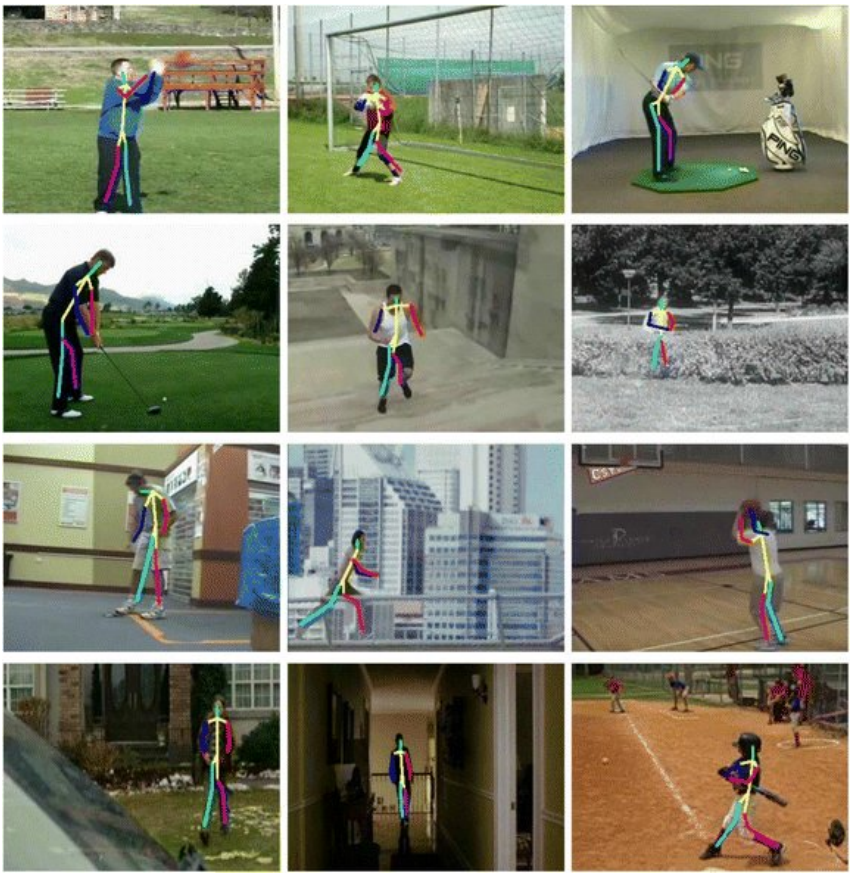


Figure 2.16: Example of actions and scenes extracted from JHMDB. Picture Credit [Liu et al., 2018].

Table 2.2: Properties of the selected experimental datasets.

	SHREC Dataset	**JHMDB Dataset**
Number of samples	2800	928
Training/Testing Protocol	1 Training Set 1 Testing Set	3 Splits Training/ Testing Sets
Dimension of skeletons	3D	2D
Subject	Hand	Body
Number of actions	14 and 28	21

The SHREC dataset is composed of hand gesture sequences for supervised action classification such as Grab, Tap, Expand, Pinch, Swipe... Each gesture falls into one of 14 categories and can be performed with either only one finger or with the whole hand (hence the 14 or 28 classes depending on the number of fingers used). It contains a total of 2800 examples, being performed by 28 different participants in total. Each gesture might extend through time and is not limited by a specific temporal resolution. For each sequence a depth image of the scene is provided at each time step, alongside with both a 2D and a 3D skeletal representation of the hand which is derived from RGB-D data in a controlled environment. Although multiple perception modalities are available (*i.e.*, RGB-D data and pose kinematics), only the skeletal information is used during our experiments. Following the splits provided by the original paper [De Smedt et al., 2017], The dataset is split into 1960 train sequences (70% of the dataset) and 840 test

sequences (30% of the dataset).

The Joint-annotated Human Motion Data Base (JHMDB) is composed of 2D skeletons that are obtained from RGB videos in non-controlled environments and therefore is more representative of more general cases for real-world scenarii. JHMDB is composed of manually annotated body gesture sequences collected from various sources, such as movies, public databases such as Google, Youtube videos... Each gesture sequence falls into one of 21 categories such as brushing hair, clapping, jumping, golfing, catching... The dataset contains a total of 928 samples and the evaluation is done by cross-validation ($N = 3$) according to the splits provided by [Yang et al., 2019].

Classification Metrics

For each model variation, we provide the model accuracy on the test set or the average accuracy of the N splits during cross-validation.

Results and Discussion

Table 2.3: Performance of the given model for different encodings of the sequences on SHREC [De Smedt et al., 2017], the architecture of the model remains unchanged.

Method	Parameters	Accuracy on SHREC 14	Accuracy on SHREC 28
LDA on features branch input	-	33.0%	27.6%
LDA on Classic Encoder ($\lambda = 0$)	-	37.9%	42.8%
LDA on Regularized Encoder ($\lambda = 5$)	-	43.5%	35.6%
Encoder (He initialization)	1.2M	91.2%	85.2%
Classic auto-encoder ($\lambda = 0$)	-	91.5%	85.9%
Regularized auto-encoder (λ =1)	-	91.9%	**87.6%**
Regularized auto-encoder ($\lambda = 2.5$)	-	92.4%	86.9%
Regularized auto-encoder ($\lambda = 5$)	-	**92.5%**	87.1%
Regularized auto-encoder ($\lambda = 7.5$)	-	91.9%	86.4%
Regularized auto-encoder ($\lambda = 10$)	-	90.9%	85.2%

Table 2.4: Performance of the given model for different encodings of the sequences on JHMDB [Jhuang et al., 2013], the architecture of the model remains unchanged.

Method	Parameters	Average accuracy of 3 splits
Chained Net [Zolfaghari et al., 2017]	17.50 M	56.8%
EHPI [Ludl et al., 2019]	1.22 M	65.5%
PoTion [Choutas et al., 2018]	4.87 M	67.9%
DD-Net [Yang et al., 2019]	1.82M	**77.2**%
Encoder (He initialization)	0.67M	65.2%
Classic auto-encoder ($\lambda = 0$)	-	66.4%
Regularized auto-encoder ($\lambda = 1$)	-	66.2%
Regularized auto-encoder ($\lambda = 2.5$)	-	68.3%
Regularized auto-encoder ($\lambda = 5$)	-	67.9%
Regularized auto-encoder ($\lambda = 7.5$)	-	66.5%

For this study, it was of interest to investigate if using the data $f_{\theta_1}(\mathbf{X})$ projected into the latent space provided more information compared to the initial input features X without fine-tuning the entire approach and updating the weights of the encoder. Table 2.3 shows that, by using the same classifier

on the projected data $f_{\theta_1}(\mathbf{X})$ from a classical auto-encoder, a simple linear discriminant analysis finds slightly more information in the data than when trained with the initial input features X. Moreover, the latent space representation obtained by our regularized auto-encoder seems to be a little bit more informative than a regular auto-encoder latent space representation. Afterward, from Tables 2.3 and 2.4 we evaluate the necessity of using a pre-trained encoder network for classification initialized with an auto-encoder training. By comparing the results from the same network with He weights initialization [He et al., 2015] prior to any auto-encoder training to the entire approach, we show that using an auto-encoder to initialize the network's weights helps to a certain extent the network's accuracy. Finally, we evaluate the correspondence between the value of λ for the supervised separability constraint part and its prediction accuracy.

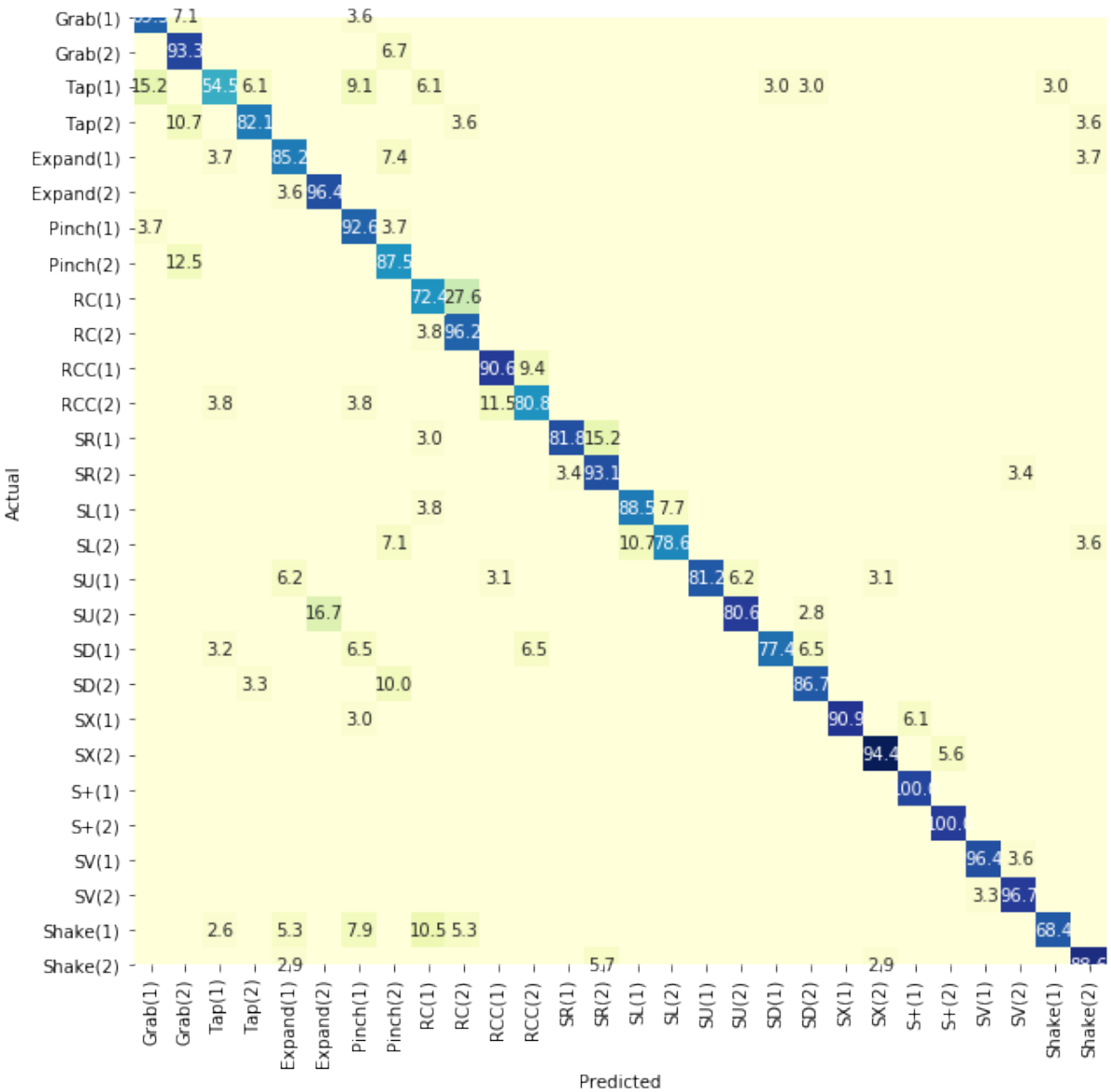


Figure 2.17: Confusion matrix obtained on SHREC 28 with a regularized auto-encoder ($\lambda = 5$).

In addition to the results presented in table 2.3, table 2.4 and table 2.5, the confusion matrix of the best performing regularized encoder on SHREC 28 is available in Figure 2.17. The given confusion

matrix shows that the proposed model is robust to each action class regardless of its complexity and suggests the model can accommodate a wide range of skeleton-based action recognition scenarii while being completely agnostic sequentially since the model is designed without any explicit biases towards sequential data modeling.

From the qualitative plots in figure 2.18, we show that projecting the data to the latent space with our regularized auto-encoder provides a more visible separation of the centroid of each class compared to a vanilla auto-encoder. Here we have fixed the size of the latent space ($C - 1 = 13$ dimensions). By mapping all the samples into the latent space, each of the C classes in the dataset would cluster in a $(C - 1)$-dimensional hyper-ellipsoid.

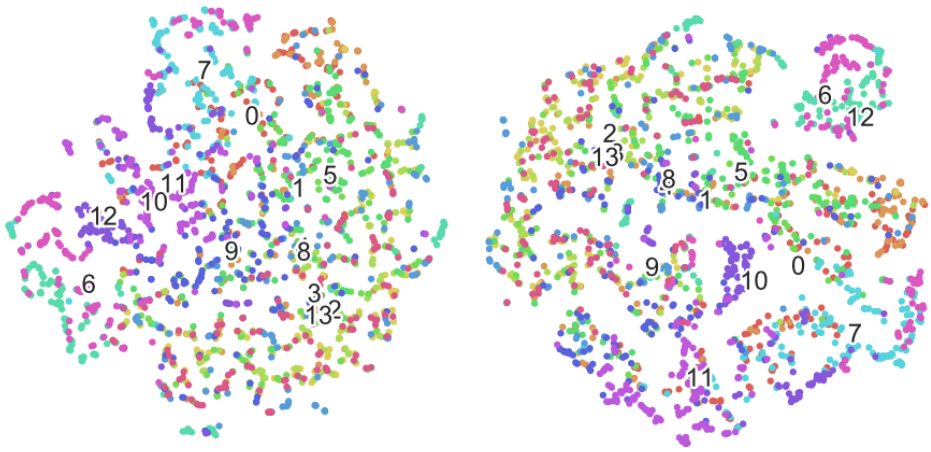


Figure 2.18: Visualization of the projection of the instances and their class centroids in the latent space for SHREC dataset via T-Sne: **left** classic auto-encoder, **right** auto-encoder combined with Linear Discriminant Analysis.

2.5.1.4 Conclusions and Perspectives

We have presented here an approach for skeleton action recognition with no sequential modeling at all that focuses on the question of data representation: while gestures are temporal phenomena, many gestures and actions might actually be inferred based on spatial poses only. To demonstrate that the question of data representation is almost as important as sequential modeling for such task, we use the simplest form of an autoencoder (a feedforward, non-recurrent neural network similar to single layer perceptrons that participate in multilayer perceptrons) to reconstruct the actions. We add to the reconstruction cost function of the autoencoder a statistical supervised regularization with a Linear Discriminant Analysis. This allows to condition the projection of the instances in the latent space upon their class. We then obtain, in addition to a reduced in size representation of the action, a first draft of the separability of the classes in the latent space. We then extract the encoder part of the trained autoencoder and evaluate its classification ability.

We tested our approach on two public databases: the SHREC database (3D Hand Gesture Recognition) and the JHMDB database (2D Body Action). On both databases, results match state of the art for skeleton action recognition tasks while being the fastest approach proposed (according to [Yang et al., 2019], up to 4 times faster than the fastest one). We therefore show that a trivial model focusing on the repre-

Model	Accuracy (14)	Accuracy (28)
[Oreifej and Liu, 2013]	78.5	74
[Devanne et al., 2014]	79.6	62
[De Smedt et al., 2017]	82.9	71.9
[Ohn-Bar and Trivedi, 2013]	83.9	76.5
[Weng et al., 2018]	85.8	80.2
[De Smedt et al., 2016] (SoCJ + HoHD + HoWR)	88.2	81.9
[Caputo et al., 2018]	89.5	–
[Boulahia et al., 2017]	90.5	80.5
[Hou et al., 2018] (Res-TCN)	91.1	87.3
Non Pretrained - Encoder (He initialization)	91.2	85.2
[Devineau et al., 2018] (SkelNet)	91.3	84.4
[Chen et al., 2019]	91.3	86.6
Classic auto-encoder ($\lambda = 0$)	91.5	85.9
[Nguyen et al., 2019]	92.38	86.31
Regularized dense auto-encoder ($\lambda = 5$)	92.5	87.1
[Hou et al., 2018] (STA-Res-TCN)	93.6	90.7
[Maghoumi and LaViola Jr, 2019]	94.5	91.4
[Yang et al., 2019]	94.6	**91.9**
[Avola et al., 2018]	**97.62**	91.4

Table 2.5: Results on the SHREC dataset using the train/test split protocol

sentation of its data with statistical regularization can compete with more complex approaches such as state neural networks or convolutional neural networks for skeleton action recognition.

As for future direction, since there is no explicit temporal modeling in the current approach, one could update that approach to models where temporal modeling is explicitly taken in consideration to improve performance. Secondly the training protocol of our approach has the disadvantage of operating with two training phases: one could fruitfully explore the usage of a dynamic loss function where the regularization effect decreases over time to make the approach easier to train. Finally, the proposed approach is based on linear discriminant analysis, as we need to get the projection matrix S from the LDA in order to compute the regularized part of the cost function. One could try to explore the same work with non-linear separability constraints instead of linear ones. However, switching from linear discriminant analysis to quadratic is impossible because of how the homogeneity of variance/covariance is not respected in quadratic discriminant analysis and we would not be able to project the samples. Another research direction would therefore be the possibility to use contrastive learning as an alternative to the LDA-term, which would remove the constraint on the bottleneck dimension.

2.5.2 Data-centric AI: the importance of the input data representation

2.5.2.1 Introduction

Many modern deep learning methods follow an "end-to-end" design philosophy that emphasizes minimal a priori representational and computational assumptions, which explains why they tend to be so data-intensive. When performing action recognition, neural networks are designed to extract temporal features from gestures and then merge them hierarchically depending on their sequence-focused design in order to perform the final classification. Intermediate representations of the gestures are entirely learned by the model and its corresponding inductive biases, without any manual intervention. However, since model

representations are based on the input data representation, finding an appropriate input representation is crucial to leverage the full potential of the network. In this subsection, we evaluate how the original input representation influences the final model accuracy.

The majority of data structure format of poses available through pose estimation approaches ignores the physical dependency relationships between joints and adds false connections between body joints that are not physically linked.

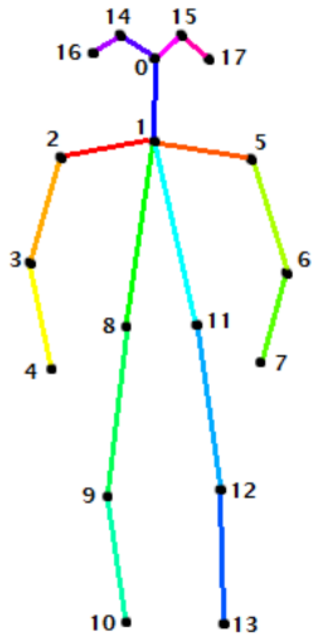


Figure 2.19: Data structure of the skeleton representation obtained with the OpenPose library [Cao et al., 2017]

Figure 2.19, shows that keeping the skeleton obtained by classical pose estimation algorithms, without resorting to a transformation could reduce the quality of classification results: some joint pairs, although incrementally following each other in the data structure used, have no valid reason to be: for example, the end of the left arm and the right shoulder (*nodes 4 and 5*) or the end of the right arm and the left hip (*nodes 7 and 8*). A large majority of current works seem to neglect the importance of this spatial representation and only focus on the temporal side: the trajectories of the joints, without questioning the spatio-temporal data structure in input. Following the work of [Liu et al., 2016] and [Yang et al., 2018b], we carried out a Depth-First Search (DFS) on a graph hub representing the skeleton, as it makes it possible to obtain a tree/graph structure exploiting only neighborhood links between existing joints without using graph neural networks.

2.5.2.2 Formalization

Pose kinematics are defined as a vector:

$$\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m) \in \mathbf{R}^{m \times N \times d} \tag{2.10}$$

where m is the sequence duration, N is the count of key-points, and d is the dimension of each

key-point. All sequences of skeletons can then be sampled in the form of a 3-dimensional (m, N, d)-shaped tensor representing a 2D image-like spatio-temporal continuous representation of the sequence of poses. The horizontal axis of each pseudo-image represents the key-points axis while the vertical axis represents the time axis. (x, y, z) dimensions of each key-point are then mapped to RGB channels. Such representations allow us to extract spatio-temporal features using standard computer vision methods such as convolution in 2D grid spaces.

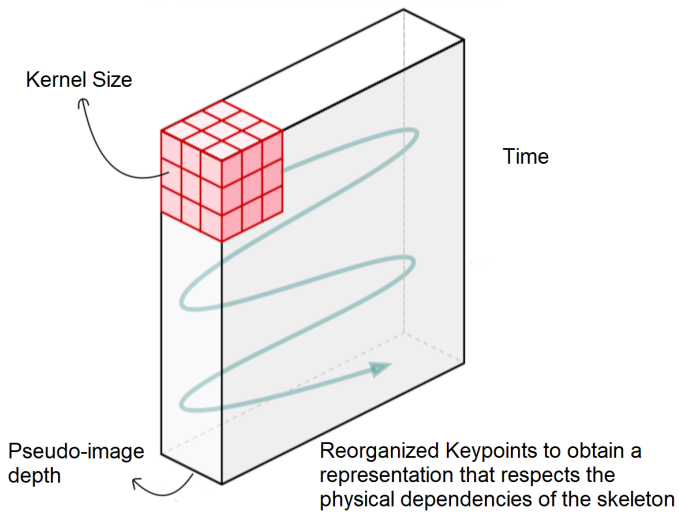


Figure 2.20: Kernel of a 2D convolution sliding over the pseudo-image

As shown in Fig 2.20, the convolution window only focuses on features that are locally connected. by reorganizing the 3-dimensional (m, N, d)-shaped tensor format, one could benefit from the spatio-temporal features extractions mechanisms of convolutions on pseudo-images in a Euclidean Grid-space while leveraging the full potential of 2D convolutions on skeletal kinematics as the input data will take in consideration the design of convolutions to respect the physical world constraints of the body structure throughout the entire feature extraction process.

Algorithm 2: Explore

```
Input: graph G, node s
mark the node s;
print(s);
foreach node t son of s do
    if t is not marked then
        Explore(G,t);
    else
end
```

To do so, we carry out a DFS (Algorithm 3) on each pose in order to obtain a representation that respects the physical dependencies of the skeleton. Exploring a DFS from a s vertex works as follows. The algorithm follows a path in the graph until it reaches a leaf or a previously visited vertex. The algorithm then returns to the last vertex where it was possible to follow another path and continues exploring. The exploration stops when all the vertices since s have been visited. Such transformation preserves both spatial and temporal relationships by repeating the joints and re-indexing them while avoiding as

Algorithm 3: Depth-First Search (DFS)

```
Input: graph G
foreach node s ∈ 𝒢 do
    if s is not marked then
        explore(G,s);
    else
end
```

many as possible redundancies, which are inevitable when one wishes to preserve the spatial structure. As shown in the figure 2.21, the keypoints are organized to respect the spatio-temporal structure of the action, in the context of a 2D convolution, the window will focus for example on three keypoints such as the knee, the foot and the hip and their position for any moment of the sequence.

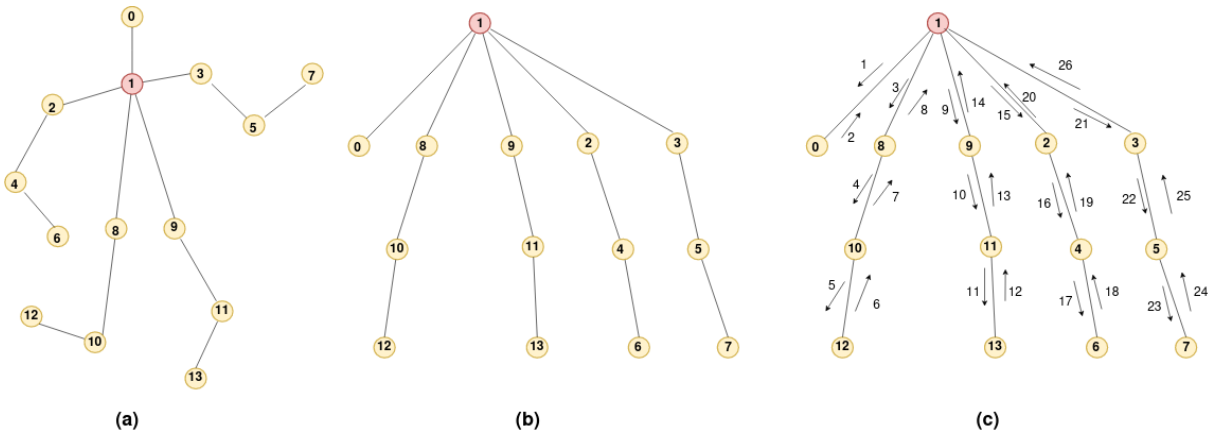


Figure 2.21: **(a)** Joints of the skeleton of a human body with the initial data structure (14 keypoints). The visiting order of the nodes is incremental:0-1-2-3-...-13. **(b)** The skeleton is transformed into a tree structure. **(c)** The tree can be unfolded into a chain whose order of visit of the nodes maintains the physical relationship of the joints: 1-0-1-8-10-12-10-8-1-9-11-13-11-9-1-2-4-6-4-2-1-3-5-7-5-3 (26 keypoints).

2.5.2.3 Experiments

To investigate the added value of this input representation for skeletal action recognition, we compare our results with or without shifted input to existing classification models for the same experimental conditions for both 1D convolutions and 2D convolutions models. We recreated the experimental training conditions of [Yang et al., 2019] as well as on their corresponding architecture presented in 2.22 for 1D Convolutions and used a simple Lenet-5 [LeCun et al., 1998] for 2D convolutions whose architecture is presented in 2.23.

Datasets

Similarly to our expriments considering the importance of explicit temporal modeling (see2.5.1), we use the same skeleton-based action recognition datasets, SHREC [De Smedt et al., 2017] and JHMDB[Jhuang et al., 2013 to evaluate the given data representation from different perspectives (see subsection 2.5.1.3).

Architecture Details

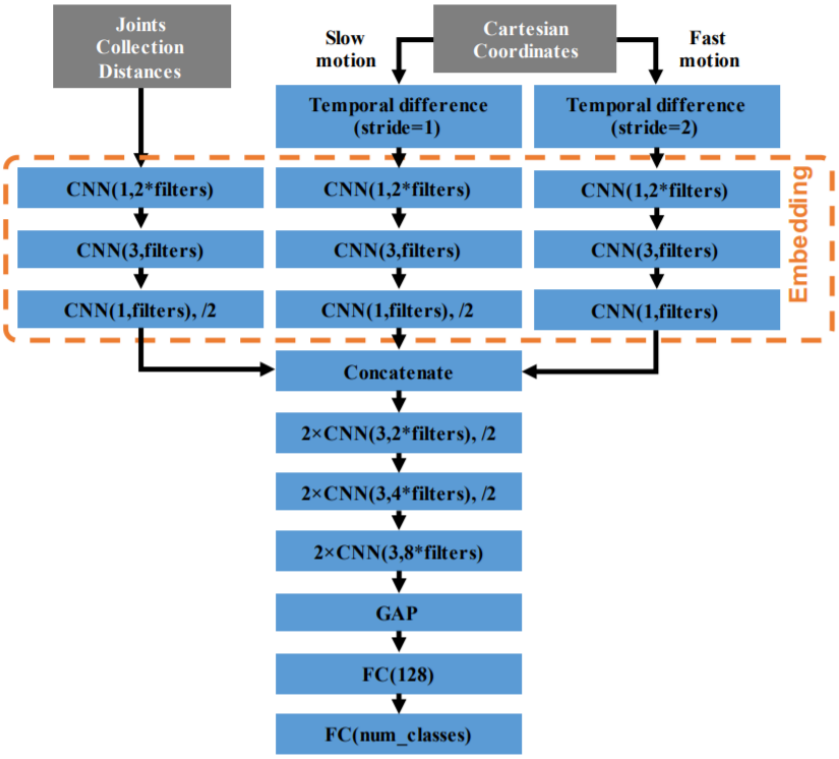


Figure 2.22: The network architecture of DD-Net [Yang et al., 2019]. "2×CNN(3, 2*filters), /2" denotes two 1D ConvNet layers (kernel size = 3, channels = 2*filters) and Maxpooling (strides = 2). Other convolutive layers are defined in the same format. GAP denotes Global Average Pooling. FC denotes Fully Connected Layers. We can change the model size by modifying the "filters" parameter.

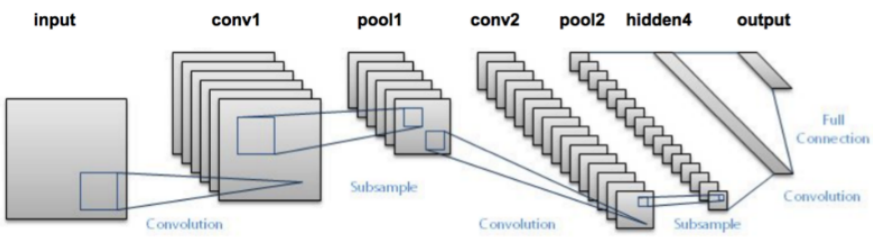


Figure 2.23: The architecture of the LeNet network [LeCun et al., 1998], with the basic components of a convolutional network: convolutions, pooling, fully connected layer and a softmax classifier.

- 1D convolutions: The experimental conditions do not differ from those of [Yang et al., 2019], the corresponding architecture is presented in Figure 2.22

- 2D convolutions: In contrast to the original Lenet architecture [LeCun et al., 1998] presented in Figure 2.23, we add dropout [Srivastava et al., 2014] with $p = 0.5$ for each layer. After feature extraction from the convolutional layers, batch-normalization [Ioffe and Szegedy, 2015] is added for each dense layer of the architecture. Training is performed with the Adam optimizer [Kingma and Ba, 2014] following the recommendations of the paper and with a learning rate of 0.01. We automatically reduce the learning rate by half once stuck in a plateau after 5 epochs not improving the loss.

Results

Table 2.6: Results obtained via DFS normalization on SHREC [De Smedt et al., 2017], the architecture DD-NET [Yang et al., 2019] remains unchanged.

Method	Parameters	14 classes	28 classes
DD-NET (64 filters)	1.82M	94.6%	91.9%
DD-NET (32 filters)	0.50M	93.5%	90.4%
DD-NET (16 filters)	0.15M	91.8%	90.0%
DFS-DD-NET (64 filters)	1.84M	**95.9%**	**92.4%**
DFS-DD-NET (32 filters)	0.51M	94.7%	92.0%
DFS-DD-NET (16 filters)	0.16M	93.1%	90.5%

Table 2.7: Results obtained via to DFS normalization on JHMDB [Jhuang et al., 2013], the architecture DD-NET [Yang et al., 2019] remains unchanged.

Method	Parameters	Results
Chained Net [Zolfaghari et al., 2017]	17.5M	56.8%
EHPI [Ludl et al., 2019]	1.22M	65.5%
POTION [Choutas et al., 2018]	4.87M	67.9%
DD-Net (filters 64)	1.82M	**77.2%**
DD-Net (filters 32)	0.5M	73.7%
DD-Net (filters 16)	0.15M	65.7%
DFS-DD-NET (filters 64)	1.81M	76.7%
DFS-DD-NET (filters 32)	0.5M	**77.2%**
DFS-DD-NET (filters 16)	0.15M	66.4%

Table 2.8: Results obtained with DFS normalization on SHREC [De Smedt et al., 2017] (3D hand skeletons) for a 2D convolutional network.

Method	Kernel Size	Parameters	Accuracy 14	Accuracy 28
Lenet-5 [LeCun et al., 1998]	3x3	0.10M	90.9%	83.8%
Lenet-5	5x5	0.10M	91.9%	85.5%
Lenet-5	7x7	0.10M	92.3%	84.5%
Lenet-5	9x9	0.10M	90.9%	87.6%
DFS-Lenet-5	3x3	0.18M	90.9%	**89.1%**
DFS-Lenet-5	5x5	0.18M	92.9%	88.1%
DFS-Lenet-5	7x7	0.18M	**93.4%**	88.6%
DFS-Lenet-5	9x9	0.18M	91.5%	87.6%

Table 2.9: Results obtained with DFS normalization on JHMDB [Jhuang et al., 2013] (2D human body skeletons) for a 2D convolutional network.

Method	Kernel Size	Parameters	Accuracy
Lenet-5 [LeCun et al., 1998]	3x3	0.07M	65.1%
Lenet-5	5x5	-	67.7%
Lenet-5	7x7	-	69.7%
Lenet-5	9x9	-	69.4%
DFS-Lenet-5	3x3	0.13M	63.5%
DFS-Lenet-5	5x5	-	65.2%
DFS-Lenet-5	7x7	-	**71.9%**
DFS-Lenet-5	9x9	-	68.1%

For this study, it was of interest to investigate if using the reorganized data to respect the physical constraints of the body structure while using convolutions would help to leverage the potential of deep learning networks without modifying their initial structure or components. We note here an improvement in accuracy for both datasets, for different input dimensions (2d and 3d body poses) and for different convolution dimensions as shown in Tables 2.6 and 2.7 for 1D convolutions and Tables 2.8 and 2.9 for 2D convolutions. By choosing a complex model for the 1D convolution and a trivial model for the 2D convolution, we hypothesize that this transformation can be beneficial whatever the level of complexity of the model. However, this transformation significantly modifies the size of the network input and

consequently, the parameter size of the approaches will increase. This increase is only slightly visible on DD-Net as there is no dense layer. For Lenet-5 the size of the model increases by a factor of two, due to the fully connected part of the architecture. It is therefore advisable to avoid the use of dense layers as much as possible for this approach or to work on networks of reduced size. The boost in performance for such transformation seems to be more marked on SHREC than on JHMDB. One explanation is that the level of abstraction and precision is not the same between these datasets. The number of keypoints in SHREC is higher than in JHMDB and therefore the organization of the keypoints could provide more information. Such transformation is therefore relevant as it only changes the size of the image input and therefore does not change the network's architecture, nor its inductive biases while becoming better for the task it was designed for: action recognition. Finally, this transformation requires an a priori knowledge of the organization of the input data structure (*e,g*: elbow: keypoint 1, head: keypoint 2, ...). In the context of the thesis, this does not pose any specific constraints. However, it will be impossible to perform this work on datasets where the pose data structure is not specified.

2.6 Summary

In this chapter, we have presented an overview of modern computer vision modalities for action recognition. We specifically focus on the simple yet informative skeleton modality, as it has been proven to be sufficient to describe and understand the motion of a given action without any background context. Thereafter, we explore the different neural network architectures used for sequential modeling of pose kinematics. Recurrent approaches have long been considered as the *de facto* approaches for obtaining good performance on sequences with neural networks. However, this assumption has been radically challenged by recent research in the field. Apart from their ability to match recurrent approaches for sequence modeling, convolutions have the particularity of having local connections, which are effective in reducing the number of parameters and thus accelerating convergence, a useful property when dealing with small databases. Moreover, as the parameters of the convolution kernels are shared throughout the convolution space, convolutions can handle both fixed length and variable length inputs in the same way as recurrent networks, which makes them an approach of choice in tasks where the inference speed remains important. Nevertheless, compared to the most promising approaches for skeletal action recognition, *e,g* graph neural networks, convolutions in an Euclidean grid space seem to neglect the importance of the spatio-temporal input data structure. However, considering that gesture recognition algorithms need to be reliable and fast enough to be computed in real-time on embedded devices, convolutions still seem to have the high ground compared to graph neural networks. The knowledge of the optimization of euclidean data structure networks is conserved compared to approaches where basic operations need to be redefined and one might lose speed efficiency in the process. Finally, we question the importance of representations, inductive biases and their roles in skeletal action recognition. Firstly, we evaluate the importance of explicit temporal modeling for gesture recognition. We propose a fully-connected autoencoder, that does not benefit from any relational inductive bias and enforces the mapping from inputs to outputs in the embedding via statistical regularizations. We show that the proposed approach reaches the performances of classic sequence modeling architectures on action classification tasks with little available data. Secondly, we investigate the importance of sending informative-representation ready data to a deep learning architecture in a 1D-2D grid space. By transforming the input data based on physical

world constraints of the body structure prior to the learning of multiple layers of feature hierarchies that automatically build high-level representations of the raw input, we show that finding an appropriate input representation is crucial to leverage the full potential of a deep learning network for action recognition.

Chapter 3

From Action Recognition to Pedestrian Discrete Intention Prediction

Contents

3.1 Understanding intentions and their role in predicting trajectories 46

3.2 Trajectory-based pedestrian action prediction 47

3.2.1 Related Works 48

3.2.2 From Bi-RNNs to U-RNNs 50

3.2.3 Methodology 50

3.2.4 Experiments 51

3.2.5 Conclusion and Perspectives 56

3.3 Pedestrian Discrete Intention Prediction 57

3.3.1 Hit the road Jack: Human-factor perspectives on pedestrian behavior prediction 57

3.3.2 Literature Review of State-of-the-Art 60

3.3.3 Data sets for Pedestrian Intention Prediction 61

3.4 Inferring crossing behavior via pose kinematics only 64

3.4.1 SPI-Net: a representation-focused multi-branch deep learning network 66

3.4.2 TrouSPI-Net: Spatio-temporal attention on parallel atrous convolutions 76

3.5 Summary 86

3.1 Understanding intentions and their role in predicting trajectories

Consider the following scenario: you're driving down the street and come upon a person standing on the corner. How can you tell if they are going to cross? Even for human drivers, interpreting the intentions and behaviors of other road users can be difficult and complex. A driver's role is to determine whether another road user wants you to wait and let them cross, if they are waiting for you to cross after you pass, or if they are simply waiting for something else. Even then, what a person signals may not be the same as what they end up doing. Pedestrian intention prediction is the corresponding area of research that seeks to automatically determine the underlying motives of pedestrians and their incoming actions/positions.

The majority of existing techniques for pedestrian action prediction are trajectory-based [Alahi et al., 2016] [Bhattacharyya et al., 2018, Kothari et al., 2021], which means they depend on previously observed pedestrian positions to anticipate pedestrian positions in the future. These methods are successful when pedestrians have already crossed or are going to cross, i.e., these algorithms react to an action that has already begun rather than predicting it. For instance, past trajectories of a pedestrian might not always play a role in its underlying objectives: when a pedestrian is waiting at the kerb, he may have no intention of crossing the street at all, or he could have the intention to cross the street but could not manage to do so because of the dynamics of the scene, or he could have the intention to cross the street and manage to do so. One way to take into account this inability of trajectory-based approaches is to determine the intention of the pedestrian before he/she even initiates his/her action. This would provide additional information about the pedestrian's intention that does not depend on the dynamics of the scene and his past positions. Such discrete information could then be used by trajectory-based approaches to enhance their forecasting performance, or more broadly, used by any vehicle planning module in crowded urban traffic environments. In this chapter, we split the pedestrian intention prediction task as a combination of high-level discrete behaviors as well as continuous trajectories describing the expected future movement of the pedestrian:

Trajectory Forecasting, the goal is to predict the value of a sequence at a time index t_p based on previous values only, *e.g*, based on values whose temporal index t is such as $t < t_p$. Trajectory-based pedestrian action prediction modules aim at forecasting the future trajectories of all the pedestrians in a scene based on their past trajectories. This corresponds to answering the open question "Where will the pedestrian be?" This is therefore a regression problem.

Pedestrian Discrete Intention Prediction, while action recognition consists of using a complete sequence of poses to label an action (see section 2.3), intention prediction predicts from an incomplete sequence to label an intention (i.e., before the pedestrian crosses). In the current state of academic data sets, this is equivalent to answering the closed question: "Will the pedestrian cross the street?" This is therefore a classification problem. The prediction can rely on multiple sources of information, including visual features of the pedestrians and their surroundings, pedestrian kinematics, spatial positioning of the pedestrian based on 2D bounding box locations, optical flow and ego-vehicle speed.

As part of our contributions presented in this chapter, we first detail the different types of existing ap-

proaches for pedestrian trajectory forecasting, we then introduce an asymmetrical bidirectional recurrent neural network architecture called U-RNN to encode pedestrian trajectories and evaluate its relevance to replace LSTMs for various trajectory-based models. Secondly, instead of focusing on continuous trajectories describing the expected future movement of the pedestrian and merely relying on past trajectories to predict intentions, we address the problem of pedestrian discrete intention prediction. The complexity of a pedestrian discrete intention prediction algorithm is directly impacted by the number of perception modalities it uses. Fusing multiple perceptive modalities into a single representation often leads to a high complexity, a high training time and a consequent inference time due to the presence of multiple networks extracting features for each modality (RGB, Optical Flow, Pose Dynamics...). Considering the importance of crossing prediction algorithms to run efficiently for real-time usage while being robust to a multitude of complexities and conditions, our goal is to propose a model using only pose kinematics for pedestrian intention prediction that reaches the performance of multi-modal approaches.

3.2 Trajectory-based pedestrian action prediction

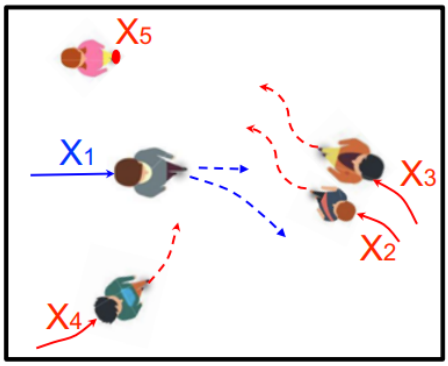


Figure 3.1: Trajectory-based pedestrian action prediction: the task is to forecast the future trajectories (dashed) of all the protagonists of the scene. Trajectory-based pedestrian action prediction involves a combination of individual goals and social interactions with other agents: pedestrian X1 will deviate from his primary trajectory to avoid a collision based on past trajectories of pedestrian X2. Picture credit [Kothari et al., 2021]

Pedestrian trajectory prediction from past positions using social interactions has been steadily receiving attention by the research community, as it plays a crucial role in various applications leading to the deployment of intelligent transport systems [Uber, 2020, Waymo, 2021]. Following the success of Social LSTM [Alahi et al., 2016] in trajectory forecasting in crowded scenes, a variety of approaches has been proposed that focused on efficiently leveraging social interactions from a scene [Ma et al., 2016, Gupta et al., 2018, Bartoli et al., 2018, Pfeiffer et al., 2018, Vemula et al., 2018, Kothari et al., 2021]. In this section, we elude the question of improving social interactions models, and focus on the encoding of the trajectories of individual pedestrians by using U-RNNs (our asymmetrical Bi-RNNs) instead of regular LSTMs. Using the recent Trajnet++ benchmark [Kothari et al., 2021] and with respect to various available learning architectures that forecast pedestrians trajectories, we evaluate the effectiveness of U-RNNs for efficient pedestrian trajectories encoding. We then provide insight into designing improved motion encoders prior to the application of interaction modules for the task of pedestrian trajectory pre-

diction.

3.2.1 Related Works

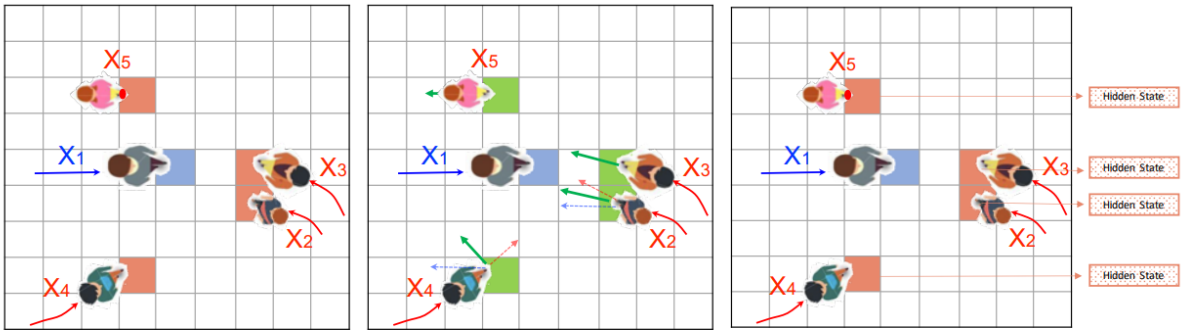
3.2.1.1 Encoder-Interaction-Decoder pipeline

The most common pipeline for pedestrian trajectory prediction consists of:

1. A **sequence encoder** for the past coordinates of each pedestrian independently. The encoder is usually a RNN, such as LSTMs [Alahi et al., 2016, Zhang et al., 2019, Zhao et al., 2019, Choi and Dariush, 2019, Kothari et al., 2021], or GRUs [Hong et al., 2019, Rhinehart et al., 2019, Rhinehart et al., 2018].

2. An **interaction module** for taking into account the neighbors trajectories. The most common way to take into account the effect of interactions between agents in their trajectories is to decode the past positions while pooling on a spatial grid with either the neighbors' positions, their relative velocities [Kothari et al., 2021], or their RNN hidden states [Alahi et al., 2016] (see Fig 3.2).

3. A **decoder** that predicts future coordinates. A common approach is to use a RNN for decoding. Some authors found that this can lead to error accumulation, and that a simple multi-layer perceptron (MLP) that predicts simultaneously all future positions performs better [Becker et al., 2019]. However, taking into account interactions between pedestrians requires to predict the coordinates one step at a time, so RNNs are generally preferred.



(a) Occupancy pooling (b) Directional pooling (c) Social pooling

Figure 3.2: Illustration of the grid-based interaction encoding modules for trajectory-based intention prediction. (a) Occupancy pooling: each cell indicates the presence of a neighbour (b) directional pooling: each cell contains the relative velocity of the neighbour with respect to the primary pedestrian. (c) Social pooling: each cell contains the LSTM hidden-state of the neighbour. Picture credit [Kothari et al., 2021]

Most of past years' research focused on improving the interaction module, with only limited new methods since [Alahi et al., 2016], or on developing approaches that take inspiration from popular frameworks such as Transformers [Giuliari et al., 2021] or contrastive learning [Liu et al., 2020b] in order to deter the model from predicting colliding or too uncomfortable trajectories. However, little work has been published on the influence of the encoder and thus on the importance of past coordinates, even if it would be easily applicable to all models that use this pipeline.

3.2.1.2 Alternative approaches.

Learning-free algorithms. The straight line at constant speed using the last known velocity is a reasonable approximation for the problem at hand [Schöller et al., 2020], given that we only try to predict the next few seconds. More complex learning-free methods can also be successfully applied, some generic, such as the Kalman Filter, and some specific, such as Optimal Reciprocal Collision Avoidance (ORCA) [van den Berg et al., 2011], which ensures that trajectories do not collide, which is not necessarily the case with other methods, especially the straight line.

Other methods. Even though non-RNN methods cannot take advantage of the research on interaction modules, alternative machine learning approaches have been developed. Convolutional Neural Networks are faster than RNN-based methods due to parallelization, but the performances are significantly lower [Nikhil and Tran Morris, 2018]. Some authors have explored the popular Transformers architecture, but the results are inferior to those of RNNs with state-of-the-art social interaction modules [Giuliari et al., 2021]. Research has also been conducted on applying Inverse Reinforcement Learning (IRL) to the pedestrian trajectory prediction problem [Fernando et al., 2019], even though retrieving the pedestrian cost function requires much more computation than learning a predictor.

Figure 3.3: Sample from the Stanford Drone Dataset (which is not included in the Trajnet++ benchmark). The environment would play an important role in order to predict trajectories that do not go on the lawn.

3.2.1.3 What information is relevant?

Scene context as an additional modality. The Trajnet++ dataset does not include the pedestrians' environment, but some argue that it is sometimes necessary in order to predict trajectories correctly [Becker et al., 2019]. Indeed, in situations such as the one in Fig. 3.3, it would be very difficult to predict plausible trajectories since the environment would play an important role in order to predict trajectories that do not go on the lawn. However, the environment's additional information seems to make generalization more difficult [Schöller et al., 2020].

Neighbors past coordinates. Most methods make use of neighbors past and present positions. However, it seems that knowing even future neighbors positions is useless in terms of prediction error [Schöller et al., 2020]. Indeed, global trajectories are not that much affected by interactions. Still, ne-

glecting the influence of neighbors inevitably leads to collisions: relevant metrics for pedestrian trajectory prediction take this into account in addition to purely spatial errors, in order to produce physically feasible trajectories.

3.2.2 From Bi-RNNs to U-RNNs

U-RNN is a bidirectional recurrent neural network architecture that was informally introduced in [Ahmet, 2020] under the form of U-GRUs for Knowledge Tracing. The objective of this work is to investigate whether U-RNNs could replace regular RNNs or Bi-RNNs for trajectory encoding. Bi-RNNs [Schuster and Paliwal, 1997] address a drawback of Recurrent Neural Networks (RNNs), which is that they cannot take the future into account when they encode an input, which may be desirable for some cases. For example, in the case of pedestrian trajectory prediction, one could expect that some movements are influenced by anticipation of a potential obstacle [Xue et al., 2017, Yao et al., 2021]. Bi-RNNs produce two outputs, one that is obtained by reading the input forward and one by reading the input backwards. Concatenation or some other operation is then applied.

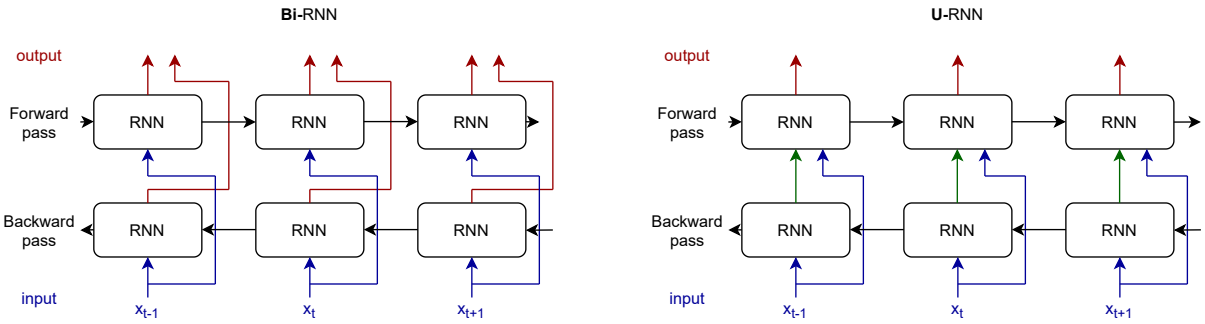


Figure 3.4: Comparison between Bi-RNN and U-RNN architectures (blue: inputs - red: outputs - black: hidden states - green: intermediate output). U-RNN can use the information from the future during the forward pass, whereas the Bi-RNN only concatenates two naive readings in both directions.

However, an aspect of Bi-RNNs that could be undesirable is the architecture's symmetry in both time directions. Bi-RNNs are often used in natural language processing, where the order of the words is almost exclusively determined by grammatical rules and not by temporal sequentiality. However, in trajectory prediction, the data has a preferred direction in time: the forward direction. Another potential drawback of Bi-RNNs is that their output is simply the concatenation of two naive readings of the input in both directions. In consequence, Bi-RNNs never actually read an input by knowing what happens in the future. Conversely, the idea behind U-RNN, illustrated in Fig. 3.4, is to first do a backward pass, and then use during the forward pass information about the future. By using an asymmetrical Bi-RNN to encode pedestrian trajectories, we accumulate information while knowing which part of the information will be useful in the future as it should be relevant to do so if the forward direction is the preferred direction of the data.

3.2.3 Methodology

We based our experiments on the Trajnet++ LSTM baseline [Kothari et al., 2021] with respect to a variety of interaction modules: *directional*, *occupancy* and *social* pooling (see Fig 3.2). All hyper-parameters

except for the encoder remained unchanged. For clarification purposes, we further explain our methodology for the *directional* pooling case.

Input embedding: The input data consists of coordinates $(x_t)_{t\in[[1,T_{obs}]]}$ for each pedestrian (with $(x_t) \in \mathbb{R}^2$). In order to allow easier generalization, we use velocities $(v_t)_{t\in[[1,T_{obs}-1]]}$ instead with $v_t = x_{t+1} - x_t$. From the trajectory velocities $(v_t)_t$ of a single pedestrian, we obtain the trajectory embeddings $(e_t)_{t\in[[1,T_{obs}-1]]}$ with $e_t = f(v_t, W_e)$ where f is a single-layer perceptron, and W_e are learnable weights that are shared among pedestrians.

U-RNN architecture: The backward and forward hidden states $(h_t^b)_{t\in[[1,T_{obs}-1]]}$ and $(h_t^f)_{t\in[[1,T_{obs}-1]]}$ are obtained according to these equations:

$$\begin{aligned} h_{t-1}^b &= RNN\left(h_t^b, e_t, W_b\right) \\ h_{t+1}^f &= RNN\left(h_t^f, \left[e_t, h_t^b\right], W_f\right) \end{aligned} \tag{3.1}$$

where W_b and W_f are learnable weights that are shared among pedestrians, and $[\cdot,\cdot]$ denotes concatenation. The last hidden state $h_{T_{obs}}^f$ is then used as the encoding of the sequence.

Decoder: For decoding, we used a RNN and directional pooling, with a learnable $GridPooling$ function that involves average pooling and a linear embedding, all of which we do not detail here and was implemented by [Kothari et al., 2021]. The predicted positions $(o_t^i)_{t\in[[1,T_{pred}]]}$ of pedestrian i are obtained according to these equations:

$$\begin{aligned} h_1^i &= h_{T_{obs}}^{f,i} \\ e_t^i &= f\left(v_t^i, W_e\right) \\ I_t^i &= \text{GridPooling}\left(v_t^{-i}\right) \\ h_{t+1}^i &= RNN\left(h_t^i, \left[e_t^i, I_t^i\right], W_d\right) \\ o_t^i &= g\left(h_t^i, W_{out}\right) \end{aligned} \tag{3.2}$$

where $(v_t^i)_t$, $(e_t^i)_t$, $(I_t^i)_t$, $(h_t^i)_t$ are respectively the velocities, velocity embeddings, interaction embeddings and decoder hidden states for pedestrian i, W_e, W_d and W_{out} are learnable weights that are shared among pedestrians (W_e being the same as for the encoder), v^{-i} denotes velocities of pedestrians other than i and $[\cdot,\cdot]$ denotes concatenation.

3.2.4 Experiments

There are several datasets that are aimed at evaluating pedestrian motion prediction, with very diverse characteristics [Rudenko et al., 2020]. We chose the Trajnet++ benchmark [Kothari et al., 2021], which aggregates several common pedestrian trajectories datasets, emphasizes the importance of quantifying the physical feasibility of a model prediction and only evaluates trajectories where there are interactions between pedestrians.

Data. Trajnet++ data consists of trajectories that have been extracted from real-life videos and that are under the form of spatial coordinates. The framerate is 2.5 frames per second. Fig. 3.5 illustrates

Figure 3.5: Images from different datasets from which the Trajnet++ benchmark trajectories are extracted. Left: ETH-hotel dataset - Center: UCY-zara dataset - Right: UCY-students dataset.

sample images from videos from which the spatial coordinates were extracted. The datasets that are used are:

- ETH [Pellegrini et al., 2010], itself subdivided into ETH-hotel and ETH-uni. ~650 tracks extracted from 25 min of video.
- UCY [Leal-Taixé et al., 2014], itself subdivided into UCY-zara and UCY-students. ~700 tracks extracted from 16 min of video.
- WildTrack [Chavdarova et al., 2018], ~650 tracks extracted from an hour of video.
- L-CAS [Sun et al., 2018], ~1100 tracks extracted from 49 min of video.
- CFF [Alahi et al., 2014], Large-scale dataset of ~42 million trajectories extracted from real-world train stations.
- In addition, synthetic data generated using ORCA [van den Berg et al., 2011] is also used.

Task. The goal is to predict the spatial coordinates of pedestrians in the near future (12 frames, i.e. 4.8 seconds), using only the near past (9 frames, i.e. 3.6 seconds). In each scene (set of different agents' trajectories over a given duration), a primary pedestrian is designated for evaluation purposes.

Categories. The scenes in the data are subdivided into categories with respect to the primary pedestrian of the scene, as Fig. 3.6 illustrates. Type I and Type II denote respectively static primary pedestrian trajectories and trajectories that are correctly predicted with an extended Kalman filter. Type III is the benchmark's type of interest, as it regroups all scenes where the primary pedestrian has interactions with other agents. Type IV is used for the remaining scenes, where the primary pedestrian trajectory seems unpredictable even when given the social environment. In addition to the four main types, Type III is further subdivided into four categories that describe the main type of interaction that is occurring: Leader-follower (the primary pedestrian follows someone else), Collision avoidance (the primary pedestrian had to avoid someone else), Group (the primary pedestrian is part of a group) and Others.

Metrics. There are four main metrics. Two are spatial errors: Average Displacement Error (ADE) and

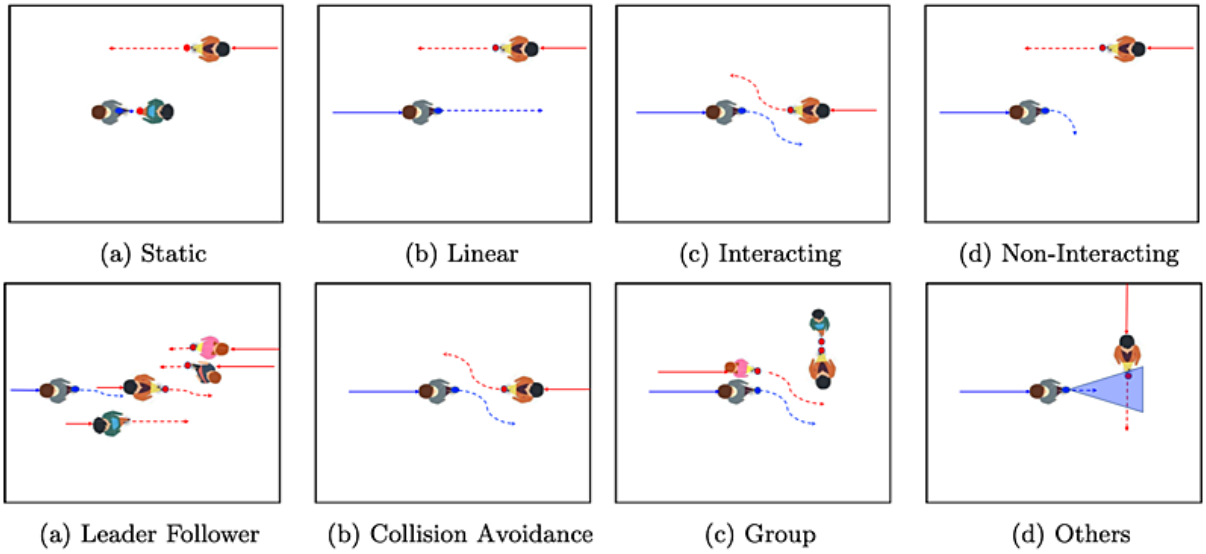


Figure 3.6: Visualization of four high level defined trajectory categories and visualization of all Type III interactions. Picture credit [Kothari et al., 2021].

Final Displacement Error (FDE), which are expressed in meters. The other two are collision errors: Prediction Collision (Col-I) and Ground Truth Collision (Col-II), which are expressed in percentage. Col-I is the fraction of collisions between the primary pedestrian's predicted trajectory and the other pedestrians predicted trajectories, and thus represents how physically realistic the predicted scene is, regardless of reality. Col-II, on the other hand, is the fraction of collisions between the primary pedestrian predicted trajectory and the other pedestrians *real* trajectories. Therefore, it represents how physically realistic the predictions are individually.

Evaluation. According to the Trajnet++ benchmark, the performance is evaluated on ~3000 scenes from ETH and UCY datasets, as well as on ~4000 synthetic scenes. The benchmark gives metrics for each type and sub-type of scene. The score that is chosen in order to compare models on the public leaderboard is FDE computed on Type III (Interacting) scenes from the real datasets (~1700 scenes), with Col-I as the secondary score (computed on the same data). Until the end of March 2021, the secondary score was FDE computed on Type III scenes from the synthetic dataset, but it was abandoned because predicting synthetic trajectories had become a solved problem. On the contrary, while performances seem to have reached a limit with respect to FDE (more than one meter on a 4.8 seconds horizon), the current challenge is to be able to predict physically feasible scenes while keeping a good FDE.

3.2.4.1 Baselines

We used the following baselines for comparison purposes:

- **Learning-free methods.** We considered Kalman filter [Kalman, 1960], constant velocity [Schöller et al., 2020] and ORCA [van den Berg et al., 2011].
- **Vanilla LSTM.** An architecture with a LSTM encoder, a LSTM decoder, and no interaction module (each pedestrian is considered independently).

- **AMENet** [Cheng et al., 2021], a conditional variational auto-encoder based on attentive dynamic maps for interaction modeling, **AIN** [Zhu et al., 2020], an encoder-decoder pipeline focusing on global spatio-temporal interactions and **PecNet** [Mangalam et al., 2020], a conditioned-on-goal endpoint variational auto-encoder. We reference the scores that are on the public leaderboard for AMENet and the ones referenced in [Kothari et al., 2021] for AIN and PecNet.
- **Social NCE** [Liu et al., 2020b], best submission on the public leaderboard, with respect to FDE. It uses social pooling and contrastive learning. We reference the scores that are on the public leaderboard.

Table 3.1: Results for several baselines and for the best submission on the Trajnet++ public leaderboard (with respect to FDE).

Model	ADE (m)	FDE (m)	Col-I (%)	Col-II (%)
Kalman filter	0.87	1.69	**0**	19.5
Constant velocity [Schöller et al., 2020]	0.68	1.42	14.3	15.2
ORCA [van den Berg et al., 2011]	0.72	1.42	**0**	**11.3**
Vanilla LSTM	0.67	1.43	15.2	12.3
AMENet [Cheng et al., 2021]	0.62	1.30	14.1	16.9
AIN [Zhu et al., 2020]	0.62	1.24	10.7	17.1
PecNet [Mangalam et al., 2020]	0.57	1.18	15.0	14.3
Social NCE [Liu et al., 2020b]	**0.53**	**1.14**	5.3	**11.3**

Table 3.1 shows the results on the four metrics and helps understand the pros and cons of each method. In terms of FDE, the Kalman filter is by far the worst of all, almost 30 cm behind constant velocity (but Type III scenes, on which evaluation is performed, are by definition scenes where trajectories cannot be correctly predicted using a Kalman filter). The constant velocity method is both extremely simple and reasonably effective, but at the cost of high collision rates. ORCA allows to completely get rid of collisions without sacrificing FDE. Vanilla LSTM is completely irrelevant, since it is worse even than the constant velocity method, highlighting how the potential of RNNs can only be revealed by using interaction encoders. Finally, the best submission on the leaderboard reaches a FDE that is 30 cm below the constant velocity method, with a Col-I of only 5%; however, as we said, ADE and FDE are still relatively high in absolute terms.

3.2.4.2 Implementation details

For training, we used ETH, UCY, WildTrack, L-CAS, and only part of CFF datasets, totalling ~29000 scenes in the training set and ~5000 scenes in the validation set. In the training procedure, we decrease the learning rate when the validation loss reaches a plateau, and also apply early-stopping when the validation loss stops decreasing for several epochs. We also use rotation augmentation as a data augmentation technique to regularize all the models[1].

We did not code everything from scratch, but rather built on top of the numerous baselines that are available with Trajnet++. Since out goal was not to beat the state-of-the-art but rather to allow meaningful comparison between different motion encoders, comparisons of given approaches are relevant given the same interaction module and hyper-parameter settings.

[1] Our implementation of the asymmetrical Bi-RNNs for the Trajnet++ benchmark is available at: github.com/JosephGesnouin/Asymmetrical-Bi-RNNs-to-encode-pedestrian-trajectories.

We tested the following architectures, denoted by their Encoder-Decoder structure. For each architecture, RNN can be replaced by either GRU [Cho et al., 2014] or LSTM [Hochreiter and Schmidhuber, 1997]:

- **RNN - RNN.** A common baseline.
- **Bi-RNN - RNN.** We used concatenation in order to fuse the outputs of the Bi-RNN, since it worked better than summation.
- **U-RNN - RNN.** The architecture described in Section 3.2.3.
- **reversed U-RNN - RNN.** The backward pass and forward pass are inverted in the U-RNN, in order to investigate if there is indeed a preferred direction of U-RNNs according to the data.

We used default number of parameters that were similar to the baselines in [Kothari et al., 2021] and did not change between different models. However, this led to LSTM models having higher total number of parameters than their GRU counterparts, but it did not affect our conclusions. The order of magnitude of the uncertainties on the metrics were ± 1 cm on ADE and FDE, ± 0.5% on Col-I and ± 1% Col-II.

3.2.4.3 Results

Table 3.2: Comparison of motion-encoding designs with respect to various interactions modules architectures on interacting trajectories of TrajNet++ real world dataset.

Model (Encoder - Decoder)	**Interaction**	**ADE (m)** ± 0.01 m	**FDE (m)** ± 0.01 m	**Col-I (%)** ± 0.5%	**Col-II (%)** ± 1%
Constant velocity [Schöller et al., 2020]	None	0.68	1.42	14.3	15.2
None - GRU	Dir. pooling [Kothari et al., 2021]	0.63	1.33	6.9	12.1
LSTM - LSTM	Occ. pooling [Alahi et al., 2016]	0.58	1.23	11.5	13.9
U-LSTM - LSTM	Occ. pooling	**0.57**	**1.22**	**10.2**	14.9
GRU - GRU	Dir. pooling [Kothari et al., 2021]	0.58	1.24	6.5	12.4
Bi-GRU - GRU	Dir. pooling	0.59	1.26	6.7	11.7
U-GRU - GRU	Dir. pooling	0.58	1.25	6.5	11.7
reversed U-GRU - GRU	Dir. pooling	0.58	1.25	6.5	**11.0**
LSTM - LSTM	Dir. pooling	0.58	1.25	6.4	11.4
Bi-LSTM - LSTM	Dir. pooling	0.59	1.28	6.2	11.9
U-LSTM - LSTM	Dir. pooling	**0.56**	**1.22**	**5.2**	11.9
reversed U-LSTM - LSTM	Dir. pooling	0.58	1.26	6.6	**11.1**
LSTM - LSTM	Soc. pooling [Alahi et al., 2016]	0.55	1.18	6.9	12.7
U-LSTM - LSTM	Soc. pooling	**0.53**	**1.15**	**6.5**	11.5
Social NCE [Liu et al., 2020b]	Soc. pool. [Alahi et al., 2016] + contr. learning	**0.53**	**1.14**	**5.3**	11.3

In Table 3.2, we present the results that we obtained during our experiments. The first thing to notice is that using a simple RNN decoder with *directional* pooling, even without an encoder, improved FDE by 10 cm and cuts Col-I by half compared to the Constant velocity model or to Vanilla LSTM. Secondly, adding a RNN encoder for past coordinates helped improving performance, which indicates that there is indeed relevant information in past positions. This suggests that pedestrians engage in complex trajectories that may span on relatively long durations.

Note that the proposed asymmetrical architecture is independent of the chosen recurrent unit. We observed in preliminary experiments that the encoder's architecture did not seem to have any impact, with

identical performances of GRU - GRU, Bi-GRU - GRU, U-GRU - GRU and reversed U-GRU - GRU architectures. At first glance, one could conclude that the information contained in past coordinates may be too redundant to allow to detect any difference between encoder architectures, as there would be no further information to extract. Or that contrary to vehicles for example, pedestrian trajectories are too irregular to make good use of past information. However, experiments with LSTMs gave different results. LSTM - LSTM and Bi-LSTM - LSTM performed similarly as GRU architectures, but using a U-LSTM encoder helped get significantly better ADE, FDE and Col-I for *directional* pooling, suggesting that there was indeed unused information in past trajectories. Regarding Col-II, the best architectures seem to differ compared to the other metrics, but this appears to be non-significant given the small score differences and the order of magnitude of the standard deviations.

The better performance of U-LSTM compared to U-GRU strongly indicates that the additional information extracted by the U-RNN architecture came from long-term dependencies. Moreover, the hypothesis we proposed, that the non-symmetrical architecture of U-RNN should better leverage information by using the preferred direction of the data is supported by the absence of performance improvement when using a reversed U-LSTM encoder.

Since it was clear that, for the *directional* pooling case, the proposed Asymmetrical Bi-RNNs motion encoder performed better than regular LSTMs which are the *de facto* RNNs for trajectory encoding, we experimented U-LSTMs with *occupancy* and *social* pooling. In both experiments, our sequence encoder yielded significantly better results compared to regular LSTMs for every available metric (ADE,FDE, Col I). This suggests that the proposed architecture is a viable alternative to LSTMs for trajectory encoding.

3.2.5 Conclusion and Perspectives

We proposed a sequence encoder based on Asymmetrical Bi-RNNs to predict future pedestrians trajectories using naturalistic pedestrian scenes data from the widely studied Trajnet++ dataset. Contrary to many previous trajectory-based pedestrian action prediction approaches that proposed new interactions modules, our work solely relies on proposing a new sequence encoder that could easily be applied to all models that use the encoder-decoder pipeline for pedestrian trajectory forecasting, while taking advantage of the research on interactions and multi-modal trajectory prediction. The proposed sequence encoder was shown to achieve better prediction accuracy than previous sequence encoders such as LSTMs for a variety of existing approaches and interactions modules. This suggests that there is still room for improvement in coordinates-only approaches, and indicates that interactions are not the only aspect on which pedestrian trajectory prediction can progress. Although this work is highly preliminary, our quantitative results could open many perspectives for future research. The success of Asymmetrical Bi-LSTMs compared to Asymmetrical Bi-GRUs suggests that this boost may come from using information with long-term dependencies, confirming that some pedestrians movements are influenced by long-term anticipation. We believe that these results constitute a promising baseline to replace LSTMs for a variety of approaches and could be used to significantly improve current trajectory prediction algorithms.

3.3 Pedestrian Discrete Intention Prediction

We have seen in section 3.1 that trajectory-based methods are successful when pedestrians have already crossed or are going to cross, *i.e.*, these algorithms react to an action that has already begun rather than predicting it. In this section, we formally define pedestrian discrete intention prediction.

Determining the pedestrians' discrete intention is mandatory. From this information, their trajectory can be further estimated to understand the pedestrians' next actions or positions, which can greatly reduce the risk of accidents. For instance, knowing the intention of pedestrians to cross the road before they actually set a foot on the road would allow the vehicle to warn the driver or automatically perform maneuvers. Therefore, preserving the pedestrians' integrity in a more efficient way than when triggered by an emergency stop once the pedestrians have moved on to the road and become a direct obstacle for the vehicle would be safer for all actors.

We first detail the factors that influence pedestrian behavior. We then provide a literature review of the existing learning-based approaches for pedestrian discrete intention prediction. Lastly, we list the available academic datasets for Pedestrian Discrete Intention Prediction.

3.3.1 Hit the road Jack: Human-factor perspectives on pedestrian behavior prediction

As the understanding of pedestrian behaviour could be used in the design of autonomous driving systems, we detail the factors that influence pedestrian behavior into two groups, the ones that directly relate to pedestrians and the environmental ones, as shown in Fig 3.7.

3.3.1.1 Pedestrian Factors

Demographics

- Gender is one of the most important factor influencing the way to cross [Moore, 1953, Heimstra et al., 1969, Holland and Hill, 2007]. Women tend to cross with a lower speed compared to men [Ishaque and Noland, 2008], tend to use zebra crossing more often [Moore, 1953], qualities such as caution [Heimstra et al., 1969, Holland and Hill, 2007] and higher law compliance [Tom and Granié, 2011] are generally more prevalent for women than for men when it comes to crossing. Similarly, the attention pattern between men and women differs in what they look at just before crossing: men tend to look at vehicles while women tend to look at traffic lights and other pedestrians [Tom and Granié, 2011]. This information is to be nuanced: the relative attention of a pedestrian is also impacted by speed, law compliance, age and road structure [Geruschat et al., 2003]... (*i.e.*: Pedestrians who crossed against the light looked at the cars, while others fixated on the traffic light.)

- Age is another factor influencing the way to cross. Old people walk slower and do not have a steady velocity [Goldhammer et al., 2014]. They are more cautious which means that they pay more attention to the traffic prior to crossing. On the other hand, younger pedestrians are less predictable than their elders when it comes to knowing what they are going to do [Holland and Hill, 2007].

Walking state

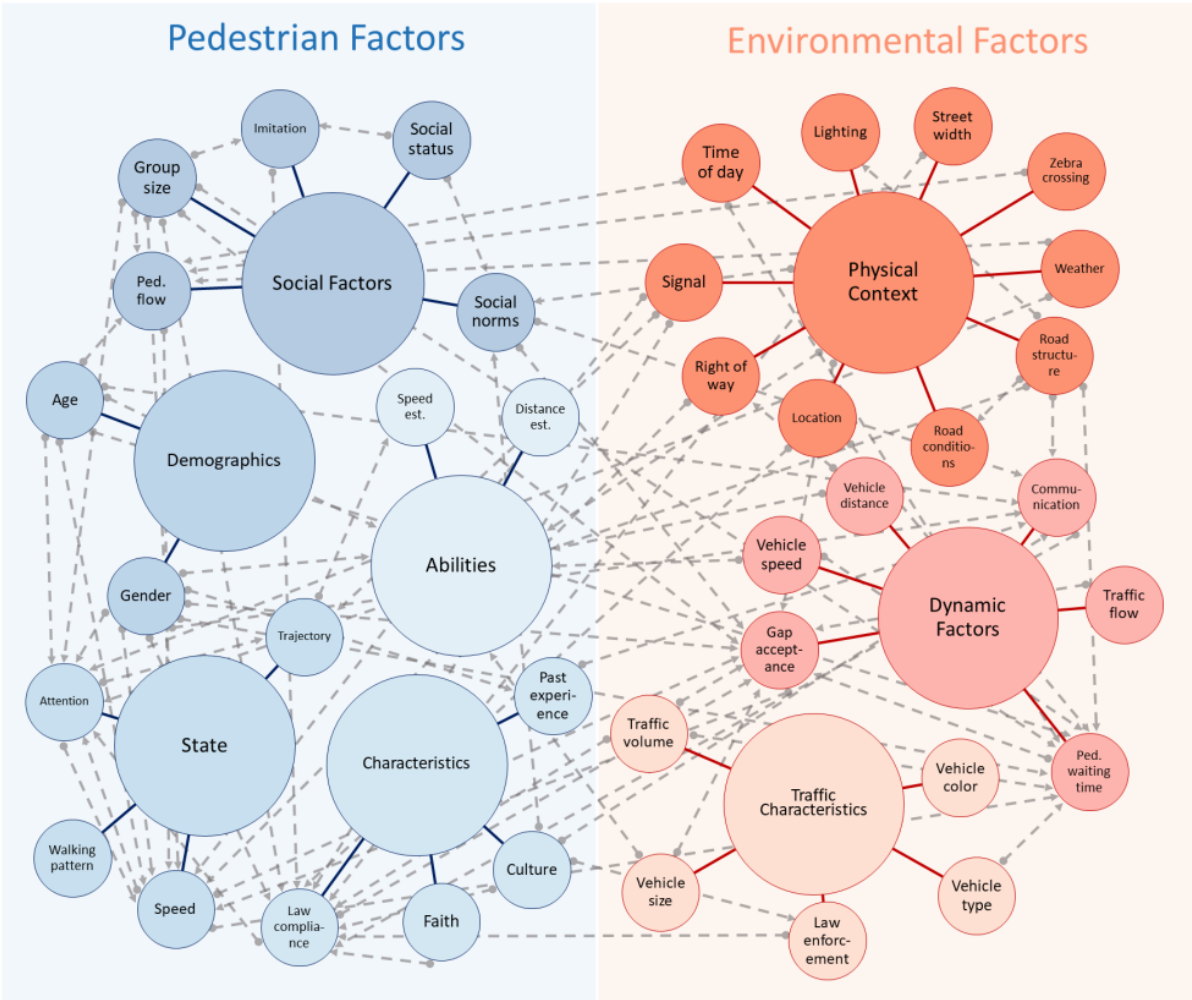


Figure 3.7: "*Factors involved in pedestrian decision-making process at the time of crossing. The diagram is based on a meta-analysis of the past literature. The large circles refer to the major factors and small circles connected with solid lines are sub-factors. The dashed lines show the interconnection between different factors and arrows show the direction of influence*". Picture and legend credits [Rasouli and Tsotsos, 2019].

- The capacity to assess speed and distance can have an impact on how pedestrians perceive their surroundings and, as a result, how they react to them. Pedestrians, on average, judge vehicle distance better than vehicle speed [Sun et al., 2015]. Walking pedestrians are less conservative about crossing compared to standing ones, one reason for this would be that their speed influences their sense of speed and distance estimation [Oudejans et al., 1996].

- Trajectory: pedestrians' ability to estimate speed is also affected by their walking direction. When pedestrians walk in the same direction as vehicles, they are more likely to make risky decisions about whether or not to cross [Schmidt and Faerber, 2009].

- Pedestrian's speed: pedestrian usually walk faster during crossing compared to when they walk on the kerb [Tian et al., 2013]. Speed is also influenced by the density of people [DiPietro and King, 1970], age [Goldhammer et al., 2014], time of the day and road structure [Willis et al., 2004].

Characteristics

- Culture plays an important role in pedestrian behaviors as it establishes a set of social standards. Variations in social norms exist, obviously between different countries but also within the same country [Björklund and Åberg, 2005]. For instance, each culture could assign different levels of importance to traffic issues (*e.g* speeding and jaywalking between swedish and chinese drivers [Lindgren et al., 2008]), could have a different gap acceptance times[2] (*e.g* indians cross on average between 2s to 8s whereas germans cross between 3s to 7s time to collision [Schmidt and Faerber, 2009]) or could perceive and analyze a situation differently (*e.g* americans judge traffic behavior based on pedestrian features, but indians place more emphasis on contextual elements such as traffic circumstances, road structure... [Clay, 1995])
- Faith and religion seem to play a role in pedestrian behavior as well. [Rosenbloom et al., 2004] show that ultra-orthodox pedestrians are three times more likely to break traffic laws.
- Law compliance (*e.g* crossing at red light, jaywalking) can be influenced by demographics but also physical factors (*e.g* the location of a designated crosswalk. [Sisiopiku and Akin, 2003]).

3.3.1.2 Environmental Factors

Physical context

- The presence of traffic signals or zebra crossings has a significant impact on how drivers behave [Moore, 1953], and as previously stated, impacts the degree of law compliance of pedestrians [Sisiopiku and Akin, 2003]. Pedestrians tend to have different trajectory patterns at unsignalized crossing (*e.g* cross diagonally [Tom and Granié, 2011], tend to walk faster [Lam et al., 1995]). Pedestrians also tend to have different attention pattern (*e.g* pedestrians look at vehicles 69.5% of the time at signalized and 86% of the time at unsignalized intersections [Tom and Granié, 2011]).
- Road structure and street width impact the level of risk affordance for both drivers and pedestrians: while pedestrians pay more attention prior to crossing in wide streets [Oudejans et al., 1996], drivers are also expected to change their behaviors depending on road structure, which inevitably influences pedestrian's expectations [Björklund and Åberg, 2005].
- Meteorological conditions influence pedestrians' behavior. Bad weather directly impacts the speed estimation capability of pedestrians which makes them less risk-averse than usual for crossing [Sun et al., 2015]. Road conditions, such as wet roads caused by rain or icy road caused by snow, can affect both drivers' and pedestrians' movements [Moore, 1953]. Illumination conditions (*e.g* day or night) impact both drivers and pedestrians visual functions leading them to make riskier decisions [Harrell, 1991].

Traffic context

- Traffic density affects both pedestrians and drivers [Schmidt and Faerber, 2009]. To put it in a nutshell, the higher the density of traffic, the lower the chance of pedestrians to cross against the signal [Ishaque and Noland, 2008].

[2] How much of a gap in traffic (usually in time) pedestrians deem safe to cross.

- Vehicle characteristics such as vehicle size also play a role when it comes to crossing. Pedestrians are more cautious when dealing with a larger vehicle [Das et al., 2005]. Pedestrians are also more likely to underestimate the vehicle's arrival time as the vehicle's size grows [Caird and Hancock, 1994].
- Vehicle type (*e.g* motorcycle, vans, cars...) influence pedestrians waiting time before crossing. Each gender is differently influenced by vehicle type when making a crossing decision [Caird and Hancock, 1994].

Dynamic factors

- Gap acceptance, or how much of a gap in traffic (usually in time) pedestrians deem safe to cross, is one of the most important dynamic factors. The combination of vehicle speed and vehicle distance from the pedestrian defines how far the vehicle is from the pedestrian [Das et al., 2005]. Pedestrians usually do not cross when the gap acceptance is below 3s and are very likely to cross when it is higher than 7s [DiPietro and King, 1970]. As denoted previously, gap acceptance highly depends on social factors, law compliance, street width...
- Waiting time impact on crossing behavior is subject to controversies. [Sun et al., 2003] argue that the longer a pedestrian wait prior to his crossing, the more impatient he becomes and the lower his gap acceptance becomes. On the other hand, [Wang et al., 2010] claim that changes in gap acceptance are not explained by waiting time alone. Waiting time should therefore be studied in conjunction with other factors such as pedestrian characteristics (gender, age, walking speed), road structure, or location.
- Communication is considered as one of the main factors in resolving traffic ambiguities [Wilde, 1980]. To indicate that they will not concede to pedestrians their right to cross, cars maintain or increase their speed. As a result, pedestrian intention to cross may differ depending on the driver's behavior. Conversely, when drivers stop their cars before they are legally required to, they signal their intention to give pedestrians the opportunity to cross [Dey and Terken, 2017]. The presence of eye contact amongst road users has been demonstrated to promote compliance with instructions and laws. At crosswalks, for example, drivers who make eye contact with pedestrians are more likely to give pedestrians the opportunity to cross [Guéguen et al., 2015].

3.3.2 Literature Review of State-of-the-Art

Being a sub-problem within action recognition, most of the existing approaches for pedestrian crossing prediction, as defined in Fig 3.8, rely on the same modalities used for the latter[3], including visual features of the pedestrians and their surroundings, pedestrian kinematics, spatial positioning of the pedestrian based on 2D bounding box locations, optical flow, semantic segmentation, and ego-vehicle speed.

Early works formulated the problem as a static image classification problem with either support vector machine [Köhler et al., 2012, Köhler et al., 2013] or 2D Convolutions [Rasouli et al., 2017b, Varytimidis et al., 2018], using only the last frame in the observation sequence to predict binary crossing behaviors. More successful approaches were designed to take into account temporal coherence in short-term motions of visual features of the pedestrians by using ConvLSTMs [Shi et al., 2015, Gujjar and Vaughan, 2019], 3D Convolutions [Tran et al., 2014, Carreira and Zisserman, 2017, Chaabane et al., 2020], or Spatio-Temporal

[3]We refer the reader to chapter 2 for an extensive review of visual modalities and architectures available at hand.

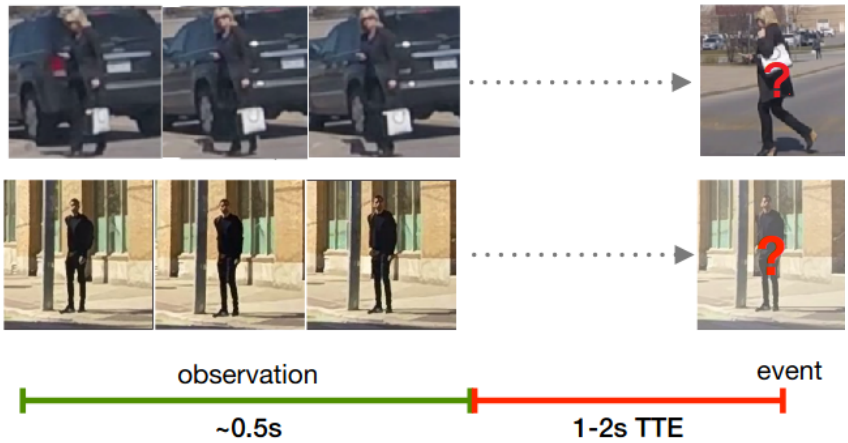


Figure 3.8: Pedestrian Intention Prediction: the objective is to predict if the pedestrian will start crossing the street at some time t given the observation of length m. Figure adapted from [Kotseruba et al., 2021].

DenseNet [Saleh et al., 2019]. [Pop et al., 2019] propose to extract spatial information with convolutive layers, then consider temporal dynamics with recurrent layers and propose a new metric for pedestrians dynamics evaluation: the time to cross (TTC) prediction. Some works are based on state-of-the-art generative methods in deep learning, focusing on the future representation of the action, and then classifying the action in its globality: [Gujjar and Vaughan, 2019] and [Chaabane et al., 2020] process the classification of the crossing action by feeding the predicted frames of their future frame prediction auto-encoder network into a classification network. However, those kinds of approaches have a major drawback: since background context is included, they are noise sensitive. Moreover, predicting future frames of a given scene can be time-consuming considering the type and the structure of the approach proposed which can be a bit delicate in a real-time scenario. Approaches trying to minimize the inference time of their models by avoiding the usage of RGB images were explored: [Achaji et al., 2021] proposes a transformer using only spatial positioning of the pedestrian based on 2D bounding box locations. Crossing prediction based on kinematics only was also explored with various available learning architectures to monitor the temporal evolution of skeletal joints such as convolutions [Ranga et al., 2020], recurrent cells [Marginean et al., 2019, Ghori et al., 2018] or graph-based models [Cadena et al., 2019].

More recently, approaches combining multiple sources of information emerged as shown in Fig 3.9. By combining several of these perception modalities in order to obtain a multi-modal representation of the scene, one obtain approaches that are often very discriminative and powerful for action prediction. However, this is at the expense of the inference's speed of the model and it highly depends on the quality of the fusion or co-learning algorithm. Therefore, multi-modal approaches differ by the way they merge the available sources, e.g. scenes, trajectories, poses and ego-vehicle speed, and the learning architecture used to infer a crossing prediction, e.g. RNN-based models [Kotseruba et al., 2020, Bhattacharyya et al., 2018, Yue-Hei Ng et al., 2015, Rasouli et al., 2019b, Kotseruba et al., 2021, Yang et al., 2022, Ranga et al., 2020] or Transformer-based models [Lorenzo et al., 2021a, Lorenzo et al., 2021b].

3.3.3 Data sets for Pedestrian Intention Prediction

Dataset collection and annotation is a time and labor-intensive operation, however annotated datasets are critical for deep learning breakthroughs. Deep learning depends heavily on the quantity and quality

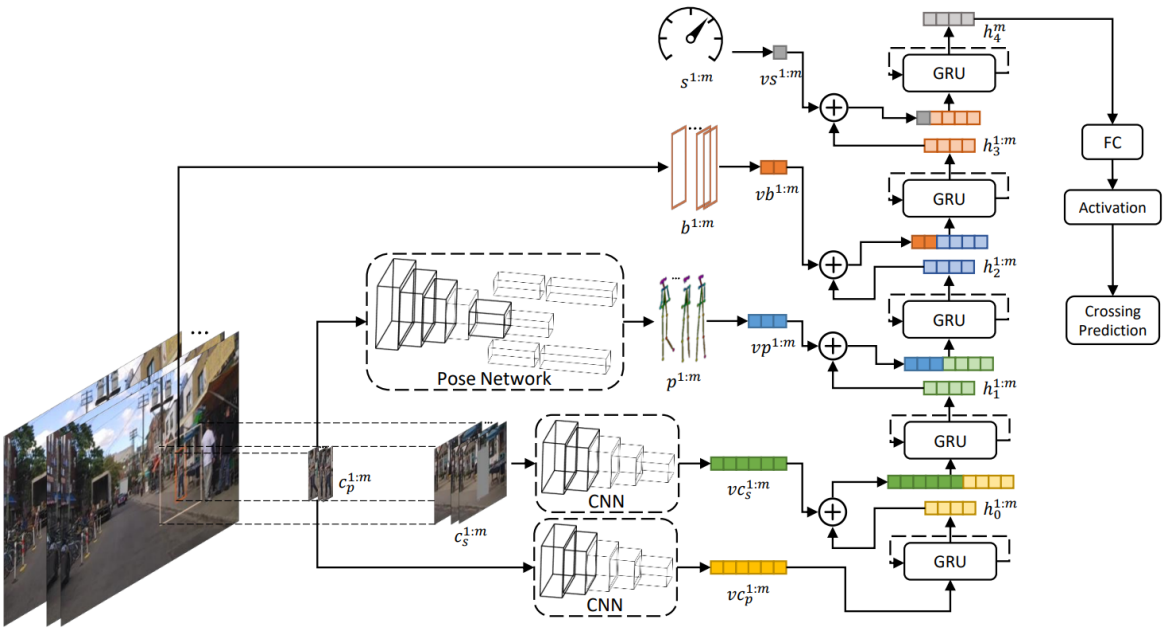


Figure 3.9: Example of a multi-modal approach for pedestrian crossing prediction, in the given case the architecture is composed of five GRUs. Each of which processes a concatenation of features of different modalities and the hidden states of the GRU in the previous level. The information is then fused into the network gradually according to the complexity of the features. Picture credits [Rasouli et al., 2019b].

of data where the performance of approaches scales up with the amount of training data (*e.g.* image classification keeps improving at least up-to billions of samples [Mahajan et al., 2018]). In contrast to trajectory-based approaches, pedestrian discrete intention prediction is generally less mature. There are fewer scientific contributions in this area of research because there is simply less properly annotated data available and it does not really fits the current paradigm. This echoes our desire in chapter 2 to work on deep learning with little available data. Nevertheless, several data sets for pedestrian crossing prediction have been developed throughout the years, including:

- **Daimler** [Schneider and Gavrila, 2013], the smallest data set available with only 68 clips of scripted actions in front of the ego-vehicle in a controlled environment with no occlusions. This one is considered trivial and was mainly used for pre-deep learning era (*e.g.* statistical approaches such as SVMs).

- **CASR** [Fang and López, 2020], which stands for Cyclist Arm Signal Recognition. The second smallest data set available with 229 arm signal actions of cyclists on videos of approximately 10 seconds each. This one is not exactly a pedestrian dataset, but focuses on vulnerable road users in the broadest sense.

- **JAAD** [Rasouli et al., 2017b], the first realistic dataset from the ego-view that include annotations for both the pedestrian and the video context. It includes pedestrian positionnal information via bounding boxes, walking states (e.g., walking, crossing, looking), appearance (e.g., clothing, group size), and demographics (e.g., age, group, gender). JAAD contains 346 clips of 5-10 seconds (30 Hz) each at daytime in the streets of downtown centers of North America and Eastern Europe. The

number of pedestrians with behavioral annotations is 686 while the total number of pedestrians is 2786.

- **PIE** [Rasouli et al., 2019a], also obtained by a vehicle-mounted camera as it navigates through crowded urban traffic environments: it contains 6 hours of continuous footage (30Hz) and provides similar annotations for all pedestrians sufficiently close to the road regardless of their intent to cross in front of the ego-vehicle and provides more diverse behaviors of pedestrians than JAAD. Additionally, PIE proposes ground-truth ego-vehicle speed, gps coordinates, and heading direction collected from the vehicle as well as relevant elements of infrastructure (traffic lights, signs and zebra crossings). The number of pedestrians with behavior annotations is 1842 making PIE the largest publicly available dataset for studying pedestrian behavior in traffic.

- **TITAN** [Malla et al., 2020], contains 700 clips ranging from 10 to 20 seconds (10 Hz) in highly interactive urban driving scenarios in Tokyo, Japan. The dataset includes 50 labels including vehicle states and actions, pedestrian age groups, and targeted pedestrian action attributes that are organized hierarchically corresponding to atomic, simple/complex-contextual, transportive, and communicative actions.

- **STIP** [Liu et al., 2020a], includes over 900 hours of driving scene videos of front, right, and left cameras, while the vehicle was driving in dense areas of five cities in the United States. The videos were annotated at 2fps with pedestrian bounding boxes and labels of crossing/not-crossing the street.

Affected by a notorious lack of interest as the research area was in a shortage of annotated data compared to its continuous trajectory-based version, the field of research has suffered for a long time from the absence of common evaluation protocols and standardized benchmarks, making the task of comparing performance between approaches complex if not impossible to achieve. Even if their reported performance was evaluated on the same data-sets, they were reported under different experimental conditions such as:

- observation length: ranging from a single frame to 10 seconds of observation at most.

- prediction horizon: ranging from one frame after the observation to a few seconds depending on the approach.

- observation endpoint: sometimes stopping the observation before the event of crossing, sometimes considering the entire event: leading to the term prediction no longer applying since the action is already taking place.

- pedestrian selection and splits methods varying to ensure a balanced data-set in terms of crossing / not crossing distribution, leading to completely different splits based on the same data-sets and hardly comparable.

- Modalities input: for instance pose kinematics could come from different pose estimation algorithms, optical flow could be perfectly computed or estimated...

The lack of a common evaluation criterion, of normalized modalities inputs, of a common observation frames selection method, and common prediction horizons made the task of comparing each approach's robustness difficult if not impossible to realize during the first part of the thesis. After a while, common evaluation protocols and modalities inputs for three datasets [Kotseruba et al., 2021] were proposed to advance research on pedestrian action prediction further and obtain a fair comparison between all the upcoming methods[4].

In the following section, we introduce the two approaches we developed for pedestrian intention prediction throughout the thesis. As the evaluation protocols [Kotseruba et al., 2021] were made available online only after the publication of the first contribution, we detail for each approach the initial evaluation protocol used. We then compare both approaches with the standard evaluation procedures [Kotseruba et al., 2021].

3.4 Inferring crossing behavior via pose kinematics only

The topic of pedestrian crossing prediction has attracted significant interest in computer vision and robotics communities but remains a difficult research topic due to the great variation and complexity of its input data. Although many approaches have been proposed which report interesting results on pedestrian crossing prediction, most of the existing methods may suffer from a large model size and slow inference speed by aggregating multiple forms of perception modalities extracted by additional networks such as background context, optical flow, or pose estimation information [Piccoli et al., 2020, Kotseruba et al., 2021, Yue-Hei Ng et al., 2015].

However, in such decisive applications, a desirable action prediction model should run efficiently for real-time usage and should also be robust to a multitude of complexities and conditions. To alleviate this issue, we propose two models using only one additional network to compute poses and disregard the other perception modalities. Pose kinematics provide a compact and structured approach to represent human pose information that would otherwise be encoded in pixels. These pose kinematics could provide enough information to efficiently infer someone's activities and intentions, such as whether or not they intend to cross the street. Based on section 3.3.1, we propose Table 3.3 listing the pedestrian and environmental factors involved in pedestrian decision-making process in accordance with the perceptive modality used. A person's head direction, for example, frequently reflects where he intends to travel, whilst his body orientation indicates which direction he is presently going (See Fig 3.10).

Our contributions are summarized in the following:

- We propose SPI-Net and TrouSPI-Net: two scene-agnostic, lightweight, multi-branch approaches that rely on pose kinematics to predict crossing behaviors. The proposed approaches could be applied following the application of any additional network to compute pedestrian body poses and could be easily implemented in any embedded devices with real-time constraints since it only uses standard deep-learning operations in an euclidean grid space. SPI-Net and TrouSPI-Net are both

[4]We are getting ahead of ourselves in this chapter by saying that there is still some work to be done on the benchmark and we refer the curious reader to Chapter 4 for more information.

Perceptive Modality	Social Factors	Demographics	Abilities	State	Characteristics	Physical Context	Dynamic Factors	Traffic Characteristics
2D bounding Box locations	x	x	x	Trajectory, Speed	x	x	x	x
Visual features of the pedestrians and their surroundings	Ped. Flow, Group Size, Social Norms	"Age, Gender"	x	Attention, Walking Pattern	"Faith", "Culture"	Street width, Zebra crossing, Weather, Road structure, Conditions, Location, Signal, Time of day, Lighting	"Vehicle Distance", "Vehicle Speed"," Traffic flow", "Communication"	Traffic Volume
Pedestrian Pose Kinematics	x	x	x	Attention, Walking pattern	x	x	Communication	x
Optical Flow	Group Size, Ped. Flow, Social Norms	x	x	Attention, Walking Pattern	x	Street width, Zebra crossing, Road structure, Location	x	Traffic Volume
Ego Vehicle Speed	x	x	x	x	x	x	Vehicle Speed, Communication	x

Table 3.3: Pedestrian and Environmental Factors involved in pedestrian decision-making process in accordance with the perceptive modality used as defined in section 3.3.1.

Figure 3.10: (A) Examples of attention towards their environment and communication demonstrated by pedestrians in urban traffic scenarios. The use of pose alone in these use cases would not be a problem since the orientation of the head, the dynamics of the arms would be sufficient to capture the semantics of the scene. (B) Scenarios with irrelevant actions with no particular semantics: eating, touching or cleaning face and looking at the phone. Pose alone might not be sufficient in most cases to identify relevant or irrelevant actions (C) In larger groups, leader-followers patterns are such that only a few pedestrians look, and the rest of the group follows. Since we are dealing with pedestrian discrete intention prediction at the level of granularity of a single pedestrian, not taking the environment into account could be a problem in this type of scenario as the pose would probably not prove sufficient.

robust to a multitude of complexities and conditions (e.g., weather, location) as it only relies on pedestrians' kinematics.

- We first represent a skeleton sequence as a 2D image-like spatio-temporal continuous representation as presented in section 2.5.2. For our second contribution, as the scale of pedestrians' actions patterns might extend through time and is not limited by a specific temporal resolution, we extract spatio-temporal features by relying on parallel processing of 2D atrous convolutions enhanced with self-attention for multiple dilation rates. This allows TrouSPI-Net to capture features for a given pedestrian action pattern for multiple temporal resolutions.

- We secondly represent a skeleton sequence as its evolution of Euclidean pairwise distances of skeletal joints over time and encode them either with a fully connected encoder architecture as presented in section 2.5.1 for SPI-Net, either with U-GRUs, as presented in section 3.2: a non-symmetrical bidirectional recurrent architecture designed to exploit the bidirectional temporal context and long-term temporal information for challenging skeletal dynamics having similar patterns but different outputs. In both contributions, this compensates for the inabilities of the first stream in learning temporal patterns invariant to locations and viewpoints.

- Evaluations of both SPI-Net and TrouSPI-Net have been conducted with the freshly proposed common evaluation criteria [Kotseruba et al., 2021] on two standard benchmarks for pedestrian behaviors prediction: Joint Attention in Autonomous Driving (JAAD) [Rasouli et al., 2017b, Rasouli et al., 2017a] and Pedestrian Intention and trajectory Estimation (PIE) [Rasouli et al., 2019a] public data-sets. Architecture variations and branch ablations are also presented to provide insight into our proposed multi-branch approach.

3.4.1 SPI-Net: a representation-focused multi-branch deep learning network

In this work, we propose to go back to *"It is all about embedding and standardization in machine-learning"*: once one finds a way to standardize and represent data more adequately, any classifier might be able to obtain good results as long as the input data is informative. By normalizing the input data, creating global-motion features and location-viewpoint invariant features or enforcing certain constraints towards the data representation of designated hidden layers, we send informative-representation ready data to the classification network. It allows us to rely on fewer hidden layers to learn informative representations of data and therefore reduce the complexity of the network compared to other approaches. Since we choose to rely on a reduced number of hidden layers, we can focus on the inference time of our model, which is mandatory since we take the model speed as one of our priorities.

The network architecture of SPI-Net is shown in Figure 3.11. In the following section, we explain our motivation for designing input features and network structures of SPI-Net. The network is divided into two branches: one focuses on the evolution of Euclidean distances relative to certain identified key-points over time, the other focuses on the evolution of the spatial representation of skeletal key-points as a function of time in the Cartesian coordinate system. The first branch corresponds to the encoder part of an auto-encoder initially trained to reconstruct an action according to the evolution over time of

selected key-point distances. We add to the auto-encoder cost function a statistical supervised separability constraint to perform better separation between instances according to their class in the latent space (see section 2.5.1). In the second branch, a 2D convolutional network, we represent a skeleton sequence as a pseudo-image. This allows us to extract spatio-temporal features using standard computer vision deep-learning methods (see section 2.5.2). We then perform a late fusion on those two branches and fine-tune the entire approach in order to evaluate the model's performance.

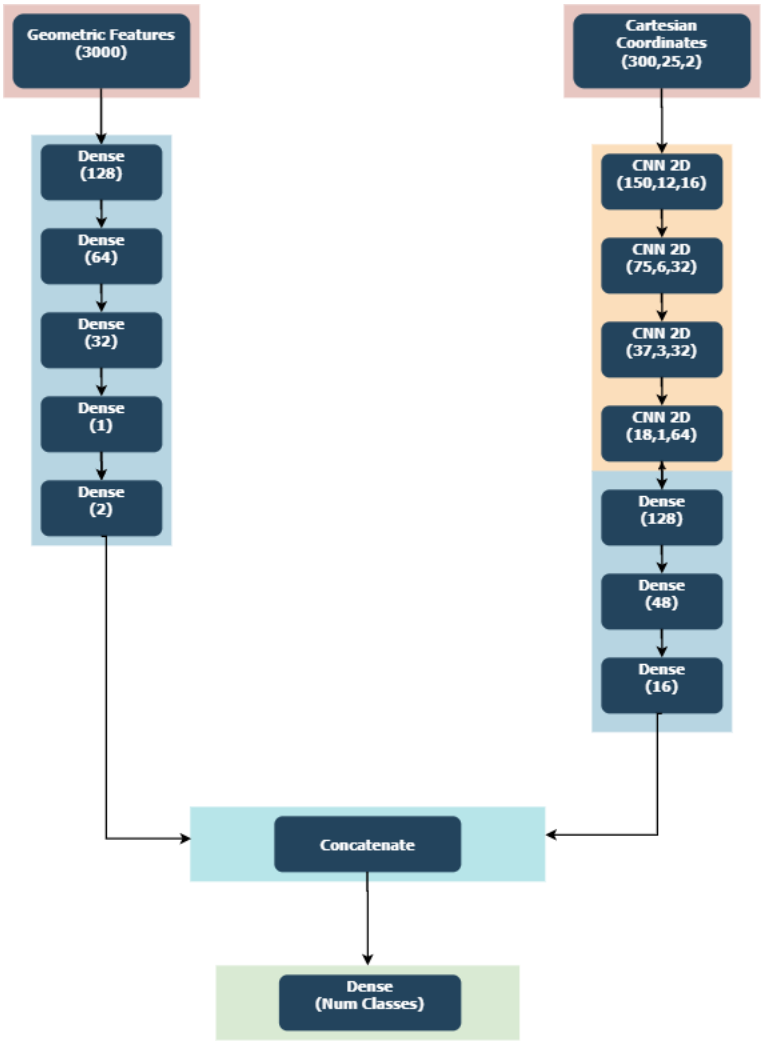


Figure 3.11: The multi-branch architecture of SPI-Net: the left branch focuses on the evolution of Geometric features relative to certain identified key-points over time. The second one focuses on the evolution of the spatial representation of skeletal key-points as a function of time in the Cartesian coordinate system. CNN 2D Blocks denote one 2D ConvNet layer (kernel size= 3), an AveragePooling layer and a Batchnormalization layer. Other Dense blocks are defined in the same format with a Batchnormalization layer following each Dense layer.

3.4.1.1 Geometric Features Branch

For the Geometric Features branch, we use the simplest form of an auto-encoder presented in Fig 3.12: a trivial feed-forward non-recurrent neural network to reconstruct an action according to the evolution of the Euclidean distances of five given key-points over time: Torso, Left and Right Shoulders, Left and

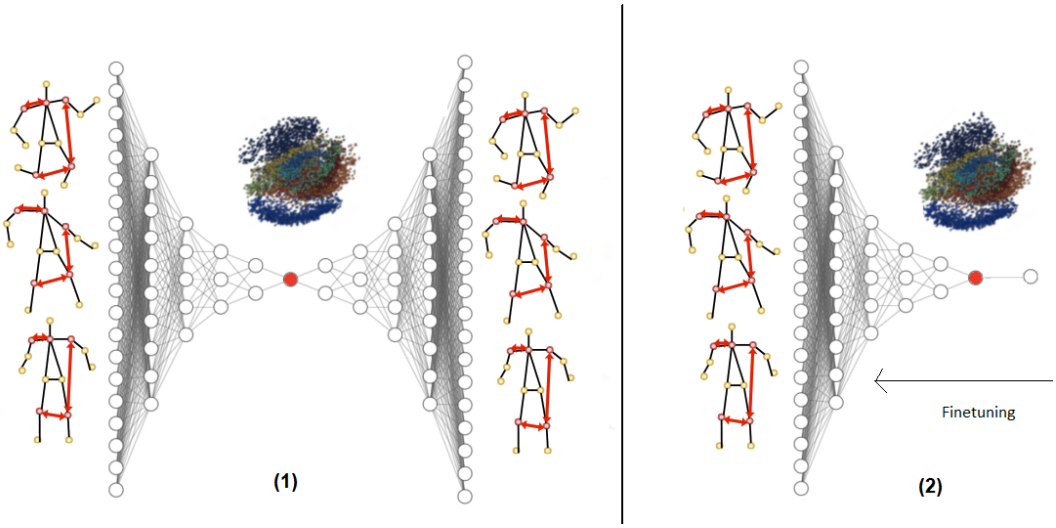


Figure 3.12: Pipeline of the approach for the Geometric branch: (**1**) we train an auto-encoder to reconstruct a sequence representing an action according to the evolution over time of the distances (represented by the red arrows) of selected keypoints (Torso, Left and Right Shoulders, Left and Right Knees). We also add a constraint specific to the separability of classes in the latent space. (**2**) We then extract the weights of the encoder part up to the bottleneck represented in red and add a classifier, which transforms the encoder part into a pre-trained network on the data for action classification.

Right Knees. The given key-points were selected in order to extract specific information for the model such as pedestrian's orientation or pedestrian's dynamics over time.

A considerable amount of literature has been published on modelling pedestrian's attention towards its environment as an input to infer its crossing intention [Rehder et al., 2014, Köhler et al., 2015, Flohr et al., 2015, Schulz and Stiefelhagen, 2015, Dey and Terken, 2017, Rasouli et al., 2018] mainly by focusing on specific key-points such as the head and more specifically its orientation. [Rasouli et al., 2018] show that across all the possible forms of attention and communication a pedestrian could use, the most notable one is to look in the direction of the approaching vehicle: for a collision incoming within the next few seconds, pedestrians always tend to look at the vehicle before crossing [Rasouli et al., 2017a]. Therefore, such head orientation input is not necessarily useful for the particular task of intention prediction since it is almost always recurrent information and would be redundant as this information would also be easily available in the Cartesian Coordinates features branch. In that regard, [Schulz and Stiefelhagen, 2015] report that head detection is not particularly useful for the particular task of intention prediction. Similar results were reported in [Rasouli et al., 2017b]: specifically focusing on the head for modelling pedestrian's attention does not seem to bring better performance for the task of intention prediction. Key-points such as elbows or wrists were considered as well in order to capture specific attention behaviors of pedestrians relying on hand gestures to communicate their intention of crossing to the driver. However, it has been shown that pedestrians mainly use explicit communication such as hand gestures to signal gratitude or dissatisfaction following the driver's action [Dey and Terken, 2017]. Such a specific gesture happens too late for our current intention prediction task as the pedestrian would be already either crossing or not at that time. In fact, [Schneemann and Heinemann, 2016] discovered that evident attention indicators used by humans for inferring crossing intentions such as the head orientation of pedestrians are not always sufficient. Even more, they concluded that *"a lack of information about the pedestrian's posture*

and body movement results in a delayed detection of the pedestrians changing their crossing intention". In conformity with this conclusion, we chose to capture different information for the Geometric features branch. Instead of extracting pedestrian's awareness features towards its environment, we try to capture pedestrian's orientation features and pedestrian's dynamics features over time based on relative distances of their key-points. Therefore the torso and shoulders key-points are preferred over the head, elbows or wrists to model the pedestrian orientation towards his environment. Besides, knees key-points are taken into consideration in order to determine if the given pedestrian is walking or standing in the scene and therefore capture its dynamics. By selecting a lower amount of key-points than the ones available in the complete body structure, we reduce the inference time of the Geometric features branch without degrading its quality for classification.

To avoid redundancy in our distances matrix and to minimize the geometric branch input size, we use the Joint Collection Distances (JCD shown in Fig 3.13) [Li et al., 2017b, Yang et al., 2019] feature to represent our vector of distances over time.

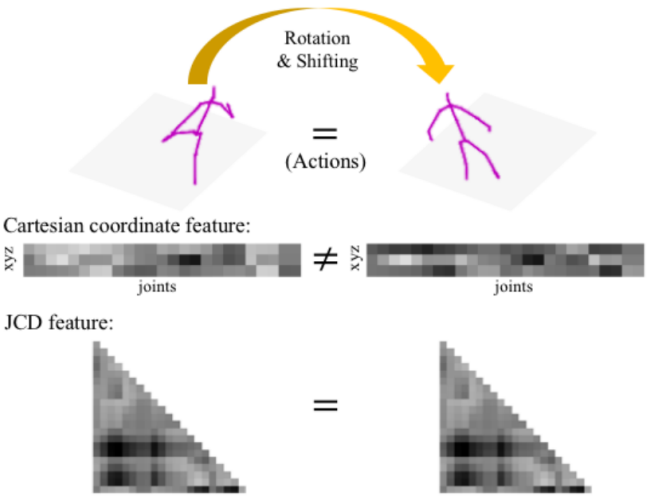


Figure 3.13: The Cartesian coordinate feature is highly dependent on locations and viewpoints. When body poses are rotated or shifted, the Cartesian coordinate feature can be significantly impacted representation-wise. Meanwhile, the geometric feature (e.g., angles/distances), is location-viewpoint invariant, and thereby stays the same. This compensates for the inabilities of the Cartesian coordinate feature branch in learning temporal patterns invariant to locations and viewpoints. Picture credits [Yang et al., 2019].

At frame k, the 2D Cartesian coordinates of joint n is represented as $J_i^k = (x, y)$. Put all of joints together, we obtain a joint collection $S^k = \{J_1^k, J_2^k, \ldots, J_N^k\}$. The formula for calculating the JCD feature of S^k is then defined as:

$$JCD^k = \begin{bmatrix} \left\|\overline{J_2^k J_1^k}\right\| & & & \\ \vdots & \ddots & & \\ \vdots & \cdots & \ddots & \\ \|\overline{J_N^k J_1^k}\| & \cdots & \cdots & \|\overline{J_N^k J_{N-1}^k}\| \end{bmatrix} \tag{3.3}$$

where $\left\|\overrightarrow{J_i^k J_j^k}\right\| (i \neq j)$ denotes the Euclidean distance between J_i^k and J_j^k. Calculating the Euclidean distances between a pair of collective joints gives us a symmetric matrix. To reduce the redundancy of information, the JCD feature is then defined as the lower triangular matrix without the diagonal.
The JCD feature is then flattened to be a one dimensional vector as our geometric's branch input of size equals to $T * \binom{nb_{keypoints}}{d}$ for each sequence. Where T is the sequence duration, $nb_{keypoints}$ is the count of key-points, and d is the dimension of each key-point.

We add to the reconstruction cost function of the auto-encoder a statistical supervised constraint specific to the separability of classes with a Linear Discriminant Analysis. This allows to condition the projection of the instances in the latent space upon their class. We then obtain, in addition to a reduced representation of the action, a first draft of the separability of the classes in the latent space[5]. Finally, we extract the encoder part of the trained auto-encoder and evaluate its classification ability as shown in Figure 3.12.

3.4.1.2 Cartesian Features Branch

As the Geometric branch only takes as input relative Euclidean distances between key-points, the Geometric branch is location-viewpoint invariant. Hence, it does not contain any global spatial motion information of the pedestrian. Solely using the Geometric feature branch is therefore unsubstantial as it does not take any information about the spatial information of the pedestrian in the scene. To overcome this issue, we develop a Cartesian Coordinates features branch that is made to retain such spatial information. Moreover, the Geometric features branch treats no explicit sequential modelling at all, but only treats the question of representation of an action in the embedding. Our Cartesian Coordinates features branch is therefore designed to extract both spatial and temporal features: features that are not explicitly learned in the Geometric branch.
Since we take the model inference speed as one of our priorities, we use a 2D-convolution-ready representation format[6] of the sequence to represent human pose sequences allowing us to extract spatio-temporal features using standard computer-vision deep-learning methods. Human pose sequences are converted to a 2D image-like spatio-temporal continuous representation based on a spatial joint reordering trick [Baradel et al., 2018, Liu et al., 2016] called Tree Structure Skeleton Image (TSSI) [Yang et al., 2018b]. Such representation preserves both spatial and temporal relationships by repeating the joints and re-indexing them[7]. Since a sequence is represented with a 3-dimensional $(T, nb_{keypoints}, d)$-shaped tensor, we can easily apply the TSSI normalization [Yang et al., 2018b] on the input and transform the original sequences into a multi-channel redundant image of shape $(300, 25, 2)$. A few sequences of pedestrian actions in the TSSI-format are plotted with their ground truth intentions in Figure 3.14 for illustration.

We then classify these images using standard computer vision deep-learning methods while preserving spatial and temporal relationships. Therefore, after the normalization of its input, the second branch corresponds to any other image classifier based on convolutions and pooling blocks for features extractions and fully-connected layers at later stages of the network. Similarly to the Geometric fea-

[5] We refer the reader to section 2.5.1 for a detailed explanation.
[6] We refer the reader to section 2.5.2 for a detailed explanation.
[7] TSSI is described in more details in Figure 2.21.

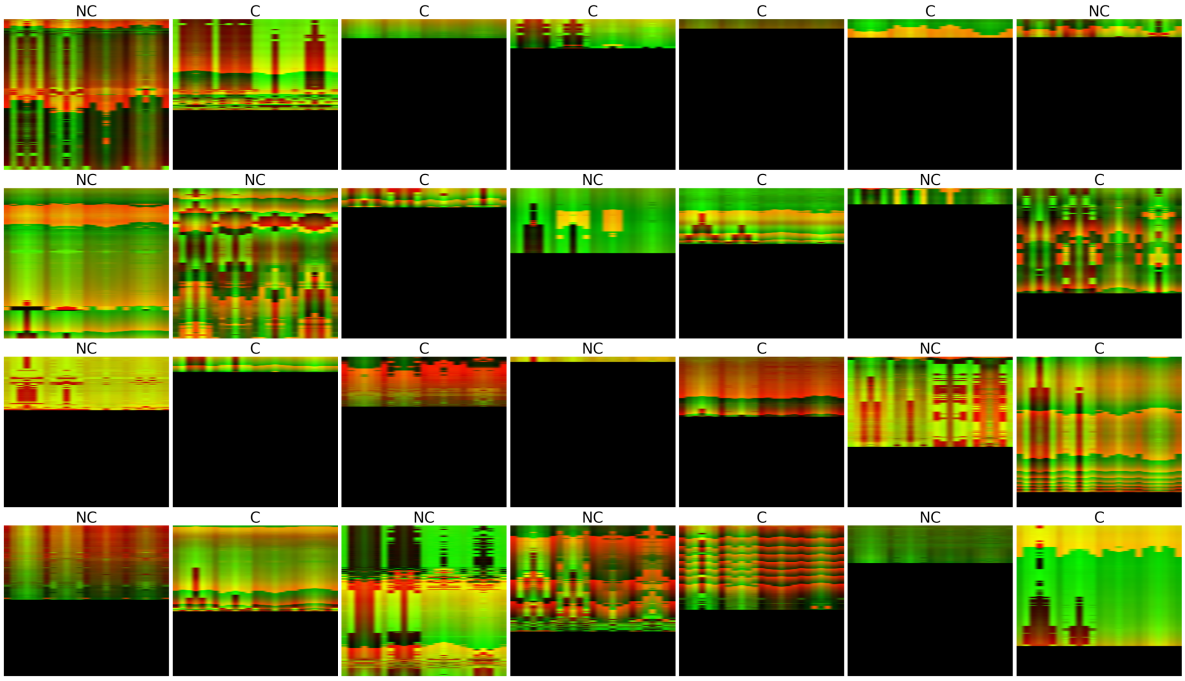


Figure 3.14: 28 different ground-truth sequences represented in a 3-dimensional (300,25,2)-shaped tensor after the TSSI normalization. The horizontal axis of each TSSI sequence is the keypoints axis. The vertical axis of each TSSI sequence is the time axis. The x, y dimensions are mapped to RG(B) channels for visualization. The axes are kept fixed and the aspect is adjusted so that the data fit in the axes. Ground truth labels C or NC represent the Crossing or not Crossing future action of the pedestrian.

tures branch, we evaluate the capability of discrimination of that branch alone for the Pedestrian Discrete Intention Prediction task and we then concatenate the two branches and evaluate the approach as its whole.

3.4.1.3 Experimental Dataset

Predicting whether or not a pedestrian is going to cross is addressed by the JAAD data set [Rasouli et al., 2017a, Rasouli et al., 2017b] which contains 346 videos. In each video, each pedestrian has its individual ID and its actions performed over time as presented in Fig 3.15. To extract the human key-points, we apply the Cascaded Pyramid Network (CPN [Chen et al., 2017]) algorithm to the ground truth spatial coordinates and individual IDs of each pedestrian provided by the data set. All video frames are normalized to 1280x1024 frame size. We then normalize each key-point $(x, y) \in \mathbb{R}^2$ individually, dividing each coordinate by 1280 and 1024 as shown in equation (3.4):

$$x' = \frac{x}{x_{\max}} \quad ; \quad y' = \frac{y}{y_{\max}} \tag{3.4}$$

Such normalization has two benefits: the first one is that data will be ready for neural networks whose weights initialization [He et al., 2015] expects such normalized input (variance ≤ 1), while retaining the spatial information of the pedestrian in the scene.

Subsequently, obtained pedestrian pose sequences are defined as: $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_T) \in \mathbf{R}^{T \times K \times d}$, where T is the sequence duration, K is the count of key-points, and d is the dimension of each key-

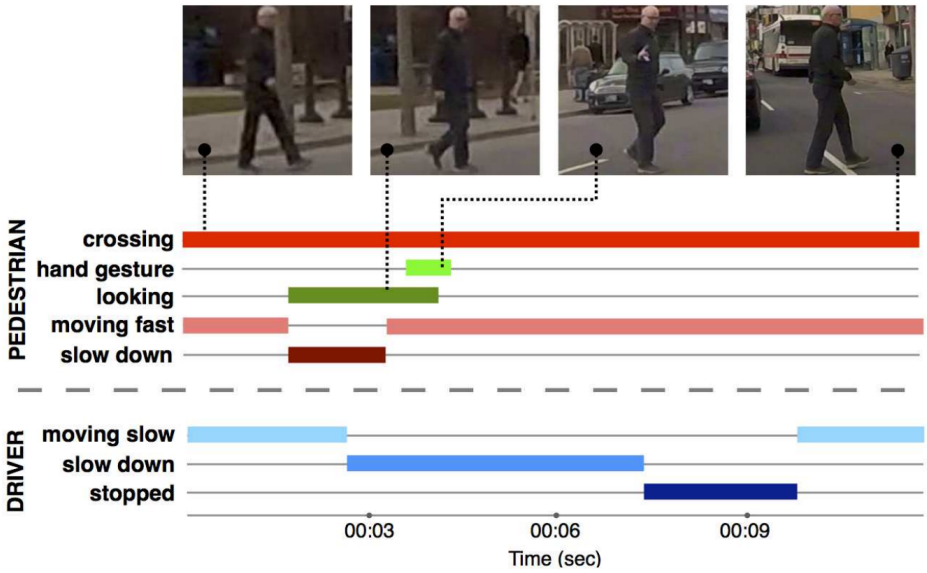


Figure 3.15: Behavioral Time line of a crossing pedestrian in the Joint Attention in Autonomous Driving (JAAD) data set.

point. All sequences of skeletons are then sampled by a sliding window to keep a fixed size in form of a 3-dimensional (T, K, d)-shaped tensor where $T = 300$, $K = 14$, and $d = 2$. The majority of the extracted sequences are smaller than the fixed T size of the sliding window, therefore sequences with less than T frames are padded with zeros. Finally, all processed data is introduced as a complete sequence to the SPI-network.

3.4.1.4 Evaluation Setup

We use the same methodology, splits and evaluation protocol as [Gantier et al., 2019] for the crossing prediction task on JAAD data set: to perform pedestrian crossing prediction, only crossing labels are used, other labels such as drivers information or context are omitted. Every pedestrian with a crossing marker along their timeline is taken as a positive sample, if not, it is taken as a negative sample. Afterwards, all positives samples are divided into two categories, the ones preceding the crossing stage and the ones taking action during the crossing stage. Only the ones preceding the crossing stage are considered. All frames with annotation are then taken from the starting time of the action to time n. They are then sampled with a sliding window of frame size $T = 300$. This procedure results in 927 crossing samples, 1855 non-crossing samples and 697 preceding the crossing samples. Only the remaining 697 prior to crossing positive samples and the 1855 negative samples are used. To avoid redundancy and bias in the data, only the last three steps of a single pedestrian sample are taken from the sliding window if the event is longer than the fixed T frames. It results in 322 positive and 182 negative samples being retained. All samples are then divided into training and test sets. According to [Fang and López, 2018] splits, we use the first 250 videos for training and the last 96 videos for testing. Since the number of positive examples is greater than the number of negative examples, some positive examples are discarded to maintain a balanced data set. The final data set consists of 240 examples equally distributed between crossing and not crossing labels in the training data set and 124 examples equally distributed in the test data set.

3.4.1.5 Implementation Details

As our SPI-Net implementation relies on multiple networks being trained independently and then concatenated for fine-tuning, we firstly here present our entire training setup to obtain SPI-Net:

- **Training the Geometric features branch:**
  - **Training the auto-encoder with a separability constraint term:** We use a standard feed-forward non-recurrent MLP whose dimensions are (3000) → (128) → (64) → (32) → (1) → (32) → (64) → (128) → (3000). We use a value of fixed $\lambda = 5$ for the LDA constraint term ponderation in the modified reconstruction cost function. To address the vanishing gradient problem, each perceptron in the given auto-encoder network uses the LeakyRelu [Maas, 2013] activation function. For regularization purposes, we use Dropout [Srivastava et al., 2014] ($p = 0.5$) , L_2 regularization with $\lambda = 10^{-1}$ and batch normalization [Ioffe and Szegedy, 2015] after each layer. We choose Adam ($\beta_1 = 0.9, \beta_2 = 0.999$) [Kingma and Ba, 2014] as the optimizer, with an annealing learning rate that drops from 10^{-3} to 10^{-8}. In order to obtain a good separability in the latent space with the LDA separability constraint, we choose to send all the training examples at once for the auto-encoder training and select a batch size of 240.

  - **Training the Encoder part for classification:** we recover the encoder part of the auto-encoder, then train a classifier with weights initialized via the auto-encoder. We use the same values of Adam optimizer for training. We however divide the training set into 30 batches of size 8. We use *ReduceLROnPlateau* with a factor of 0.2 and patience of 10.

- **Training the Cartesian features branch:** The Cartesian features branch is composed of four 2D-convolutions blocks composed of 2D-convolutions layers (kernel size= $3x3$). Similarly to the auto-encoder, we use the LeakyRelu activation function, L_2 regularization with $\lambda = 10^{-4}$ and a Dropout value of 0.5. Each convolution layer is then followed by a Batch Normalization layer and an Average Pooling layer. The fully connected layers following the spatio-temporal features extraction done by convolutions is then completely similar to any other Dense layer of the Geometric feature branch for hyper-parameters tuning. We choose Adam ($\beta_1 = 0.9, \beta_2 = 0.999$) with a learning rate that drops from 10^{-2} to 10^{-8} and *ReduceLROnPlateau* with a factor of 0.5, patience of 5, cooldown of 5 and a batch size of 8.

- **Concatenating the branches:** We then remove the classification layer of each branch and concatenate those two networks deprived of their last layer into a single one. It allows us to keep the previously learned weights of each network independently. We then add a classification layer whose weights are initialized randomly after the concatenated layer of the obtained network. Finally, we fine-tune the entire network, from pre-trained weights to the randomly initialized classification layer. We obtain SPI-Net: a late fusion and fine-tuned version of the Geometric and Cartesian features branches. As proposed in [Smith et al., 2017], we increase the batch size over time during the training and therefore fine-tune the approach with two different trainings on the same SPI-network with two different batch size. For the first training, we use Adam with a learning rate that drops from $9e{-}3$ to $5e{-}8$, *ReduceLROnPlateau* with a factor of 0.5, patience of 25

and a batch size of 8. For the second one, we use Adam with a learning rate that drops from $9e-8$ to $5e-18$ and *ReduceLROnPlateau* with a factor of 0.5, patience of 25 and a batch size of 240.

3.4.1.6 Results

In ablation studies, we first explore how each branch contributes to the intention prediction performance. We therefore explore how the LDA constraint for the Geometric branch or the spatial joint reordering trick impact the intention prediction performance on JAAD. Therefore, both Geometric and Cartesian branches results are presented in Table 3.4, Figure 3.16 and Table 3.5. The crossing prediction results of the overall approach on JAAD data set are then presented in Table 3.6. Finally, more details about each branch and SPI-Net are listed in their respective confusion matrices for the crossing or not crossing task in JAAD data set in Table 3.7.

Table 3.4: Intention prediction accuracies of the Geometric branch alone, for different encodings of the sequences of inter-keypoints distances.

Method	Accuracy
LDA on Geometric features branch input	51.6%
LDA on the classic Encoder ($\lambda = 0$)	53.2%
LDA on the regularized Encoder ($\lambda = 5$)	54.0%
Encoder (He initialization [He et al., 2015])	66.9%
Encoder with a classic auto-encoder ($\lambda = 0$)	68.5%
Encoder with a regularized auto-encoder ($\lambda = 5$)	69.4%

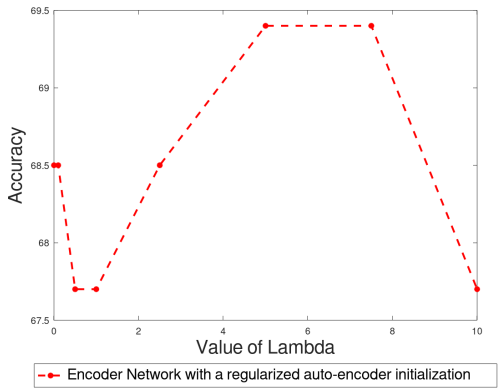


Figure 3.16: Intention prediction accuracy of the Geometric branch alone, as a function of its λ parameter.

Table 3.5: Ablation studies: classification accuracy of the Cartesian branch for pedestrian intention prediction for the crossing or not crossing task in JAAD.

Method	Accuracy
Cartesian feature branch without spatial joint reordering trick	83.1%
Cartesian feature branch with spatial joint reordering trick	88.7%

From Table 3.4, we figure that solely using the Geometric features branch alone cannot produce a satisfactory performance for the crossing or not crossing task: since most of the prior to crossing actions

are strongly correlated to global spatial motion of the pedestrian in the scene, the usage of only relative Euclidean distances between key-points is missing necessary information such as spatial dynamics or sequential modeling. However, the Geometric features branch still seems to capture some information only relative to the orientation and dynamics of the skeleton in the data without explicit temporal modeling or global spatial information. Table 3.4 shows that, by using the same binary classifier on the projected data in the bottleneck obtained from a classical auto-encoder, a simple LDA finds slightly more meaning in the data than the initial Geometric features input. Moreover, the latent space representation obtained by our regularized auto-encoder seems to be a little bit more informative than a regular auto-encoder latent space representation. In Figure 3.16, we evaluate the correspondence between the value of λ for the supervised separability constraint part and prediction accuracy. Afterward, we evaluate the necessity of using a pre-trained encoder network for classification initialized with an auto-encoder training. By comparing the results from the same network with He's weights initialization [He et al., 2015] prior to any auto-encoder training to the entire geometric branch approach, we show that using an auto-encoder to initialize the network's weights helps to a certain extent the network's accuracy. From Table 3.5, we can deduce that by taking into consideration both spatial and temporal features in the Cartesian coordinate system, we obtain better results than by only considering relative distances of given key-points of the pedestrian skeleton. We can also conclude that the usage of the Tree Structure Skeleton Image (TSSI) [Yang et al., 2018b] normalization improves the results of the Cartesian branch for the given task considerably. Such normalization is therefore relevant as it only changes the size of the image input and therefore does not change the network's architecture much while becoming better for the task it was designed for. Finally, Table 3.6, shows that by merging and fine-tuning both Geometric and Cartesian features branches into a single network, we can achieve better results for the crossing or not crossing task than by considering each branch independently.

Table 3.6: Classification accuracies for pedestrian intention prediction for the crossing or not crossing task in JAAD. CPN [Chen et al., 2017], Alphapose [Fang et al., 2016] and Openpose [Cao et al., 2017] stand for the use of human pose estimation algorithms used by the skeleton-based features method. We have also included the results reported in [Rasouli et al., 2017b, Varytimidis et al., 2018], where CNN features are based on a non-fine-tuned AlexNet [Krizhevsky et al., 2012] and Context refers to features of the environment, not of the pedestrian itself.

Method	**Accuracy**
Alexnet + Context [Rasouli et al., 2017b]	63.0%
Alexnet + SVM [Varytimidis et al., 2018]	74.4%
Alphapose + LSTM [Marginean et al., 2019]	78.0%
Res-EnDec [Gujjar and Vaughan, 2019]	81.0%
ST-DenseNet [Saleh et al., 2019]	84.76%
auto-encoder + Prediction[Chaabane et al., 2020]	86.7%
Openpose + Keypoints [Fang and López, 2018]	88.0%
Alexnet + SVM + Context [Varytimidis et al., 2018]	89.4%
CPN + GCN [Gantier et al., 2019]	91.9%
CPN + Geometric branch ($\lambda = 5$)	69.4%
CPN + Cartesian branch	88.7%
CPN + SPI-Net ($\lambda = 5$)	**94.4%**

Overall, although SPI-Net is not that complex in its architecture, Table 3.6 shows that it outperforms by more than 2.5% the previous state-of-the-art approach [Gantier et al., 2019] based on CPN [Chen et al., 2017] for pedestrian discrete intention prediction task on the JAAD dataset. The confusion matrices in Table 3.7 also shows that SPI-Net accuracy is similar on both action classes which demonstrates its ability to adapt to intra-class variation for skeleton-based dynamics.

Table 3.7: Confusion matrix of the JAAD data set obtained by each branch of SPI-Net and SPI-Net on JAAD for the crossing or not crossing task.

	Geometric Branch		**Cartesian Branch**		**SPI-Net**	
Ground Truth	**Crossing**	**Not Crossing**	**Crossing**	**Not Crossing**	**Crossing**	**Not Crossing**
Crossing	37	25	57	5	60	2
Not Crossing	16	46	9	53	5	57

3.4.1.7 Conclusion

In this work, we have introduced a new real-time representation-focused multi-branch deep-learning skeleton-based approach for the task of discrete intention prediction of pedestrians in urban traffic environments. We propose to go back to *"It is all about embedding and standardization in machine-learning"* and put great emphasis on finding a way to standardize and represent data in a more adequate way for 2D skeletal pose sequences based models. By normalizing the input data based on physical world constraints of the body structure, creating features in different coordinate systems allowing to capture different aspects of the data or enforcing certain constraints towards the data representation of designated hidden layers, we send informative-representation ready data to the classification network which allows us to rely on less hidden layers to learn informative representations of data. Our approach has achieved remarkable results: 94.4% accuracy *i.e.,* 2.5% more than the current state of the art for the Crossing or Not Crossing prediction task on JAAD data set while being completly invariant to context and road structure. Furthermore, since we choose to rely on a reduced number of hidden layers, we can focus on the inference time of our model, which is mandatory since we take the model speed as one of our priorities: SPI-Net speed can reach around one inference every 0.25 ms on one GPU (i.e., RTX 2080ti), or every 0.67 ms on one CPU (i.e., Intel Core i7 8700K), which makes it highly effective for the task of predicting discrete intentions of pedestrians and directly applicable to embedded devices with real-time constraints.

3.4.2 TrouSPI-Net: Spatio-temporal attention on parallel atrous convolutions

As stated in section 3.3.3, the lack of a common evaluation criterion, of normalized input modalities, of a common observation frames selection method, and common prediction horizons made the task of comparing each approach's robustness difficult if not impossible to realize. During the second part of the thesis, common evaluation protocols and modalities inputs [Kotseruba et al., 2021] were proposed to advance research on pedestrian action prediction further and obtain a fair comparison between all the upcoming methods. In order to propose the first and only pose-only based approach on the benchmark at this day, we proposed a new model for pedestrian action prediction based on 2D body poses: TrouSPI-Net, which is a largely modified and significantly improved version of the SPI-net architecture

(see section 3.4.1).

First, we introduce parallel processing branches to allow the architecture to access different time resolutions with atrous convolutions enhanced with self-attention mechanisms. Secondly, we apply U-GRUs (see section 3.4) on the evolution of relative Euclidean body distances over time, which acts as a regularizer of the first stream for both time and space. We then extend the U-GRUs approach as one baseline method to consider long-term temporal coherence and process each sequence of context features such as bounding box positions or ego-vehicle speed. The diagram of the model is shown in Figure 3.17 and the implementation details follow below.

3.4.2.1 Methodology

Extracting spatio-temporal features via parallel atrous convolutions on pseudo-images Pedestrian body poses sequences are defined as a vector:

$$\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m) \in \mathbf{R}^{m \times N \times d} \tag{3.5}$$

where m is the sequence duration, N is the count of key-points, and d is the dimension of each key-point. All sequences of skeletons are then sampled in the form of a 3-dimensional (m, N, d)-shaped tensor representing a 2D image-like spatio-temporal continuous representation of the sequence of poses. The horizontal axis of each pseudo-image represents the key-points axis while the vertical axis represents the time axis. (x, y) dimensions of each key-point are then mapped to $RG(B)$ channels.

By using a 2D-convolution-ready representation format, we extract multi-scale spatio-temporal features using standard computer-vision methods such as atrous convolutions and enhance the feature extraction modules by using Convolutional Block Attention Module (CBAM) [Woo et al., 2018] for self-attention mechanisms in each branch. Since sequences are represented as pseudo-images, CBAM blocks act as self-attention mechanisms for time and space conjointly. Each of the pseudo-images is directly fed to three parallel branches. All three branches present a similar architecture designed for single-scale spatio-temporal feature extraction. In each branch, the pseudo-image is passed to an atrous CBAM block, illustrated in Figure 3.17, followed by a pooling layer. This process is repeated two more times. The difference between the three atrous CBAM blocks resides in the value of the dilation rate fixed in each branch. Having three different dilation rates for the spatio-temporal convolution layers allows the network to directly work at different time resolutions while staying at the same spatial resolutions as shown in Fig 3.18. Moreover, compared to using different kernel sizes for each convolution, working with atrous convolution does not harm the model size. The outputs of the three branches extracting multi-scale spatio-temporal features are then summed into a single vector for later stages.

Formally, let $h^{(l,\beta)}(m, n)$ represent the input of the l-th atrous CBAM block of the β branch, $K^{(l,\beta)}$ be the number of feature maps, $W_k^{(l,\beta)}(i, j)$ the k-th convolution filter of the l-th convolution in the β branch with the length and the width of m and n, $b_k^{(l,\beta)}$ the bias shared for the k-th filter map, $(r_1^{(l,\beta)}, r_2^{(l,\beta)})$ the dilation rates and σ an activation function. The intermediate feature map $F(m, n)$ obtained by atrous 2D

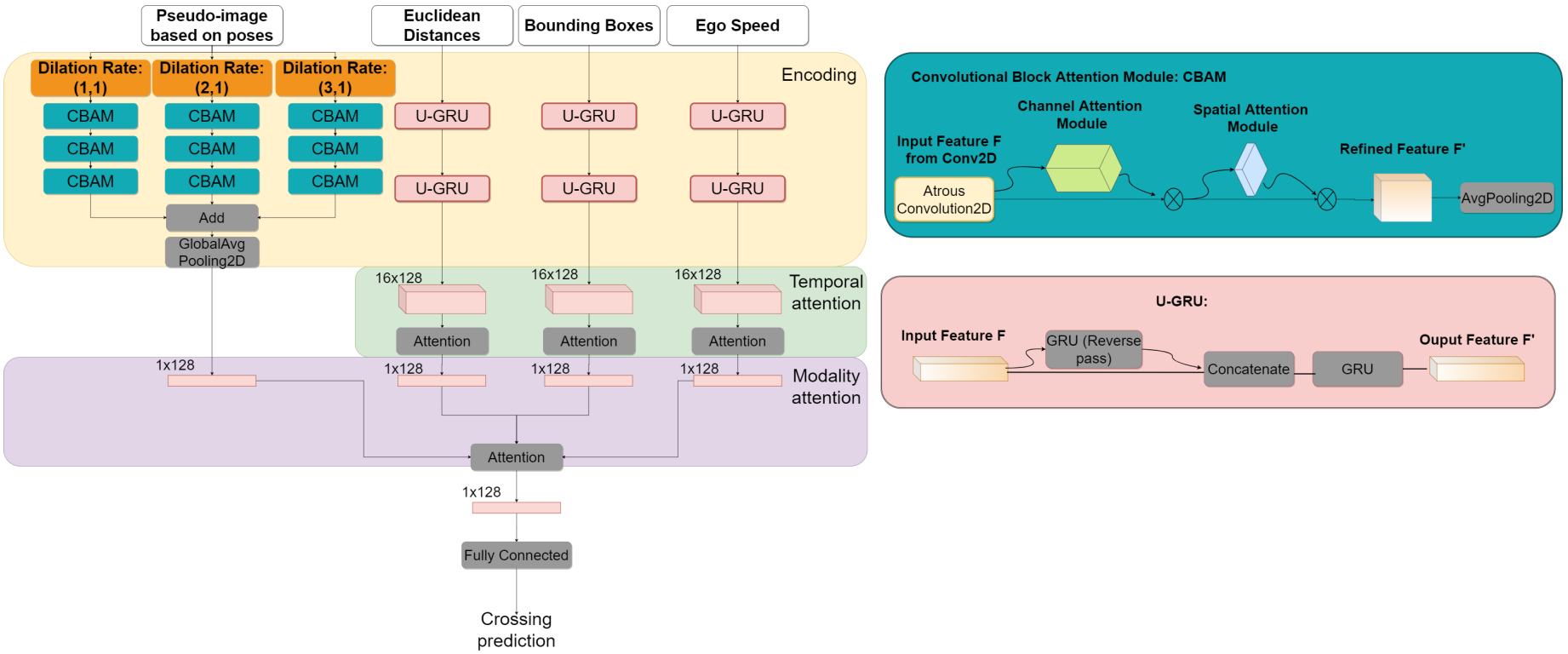


Figure 3.17: **The network architecture of TrouSPI-Net:** Its inputs consist of a sequence of 2D body poses transformed into a pseudo-image, relative pairwise distances of skeletal joints, bounding boxes, and ego-vehicle speed. U-GRUs encode every feature except pseudo-images, and each is fed into a temporal attention block. Pseudo-images are processed by parallel atrous CBAM [Woo et al., 2018] blocks with different dilation rates and then added into a single vector in order to make the size of the pseudo-images block equal to the size of the U-GRUs outputs. Modality attention is then applied to the outputs of each branch, and the weighted outputs are fed into the fully connected layer. **U-GRU blocks:** the first GRU layer does the reverse pass, we then concatenate its output with the input data and finally compute the second GRU layer's output with a forward pass. **CBAM blocks:** given an intermediate feature map extracted by atrous 2D convolutions, the module sequentially infers attention maps along two separate dimensions: channel and spatial.

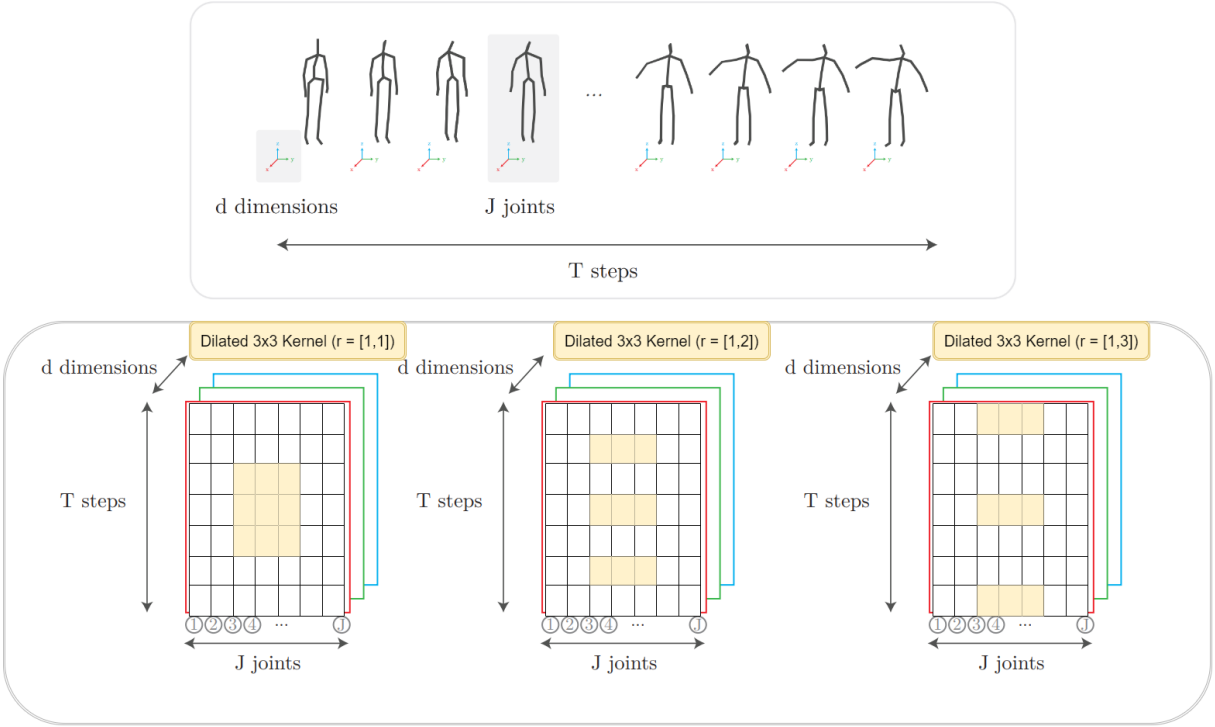


Figure 3.18: Atrous convolutions applied to our pseudo-images: compared to regular convolutions, it involves pixel skipping, so as to cover a larger area of the input in time while staying at the same spatial resolution. This could prove useful for two use cases: **(1)** the scale of pedestrians' actions patterns might extend through time and is not limited by a specific temporal resolution, relying on atrous convolution allows TrouSPI-Net to capture features for a given pedestrian action pattern for multiple temporal resolutions and could potentially improve generalization. **(2)** Pose estimation algorithms often reconstruct temporally noisy poses when given in-the-wild video data, combining three different action extraction feature protocols for three different time ranges could have a regularizing effect on the potential pose noise obtained at a given timestamp.

convolutions of the $l+1$-th CBAM block is calculated as:

$$F(m,n) = \sigma\left(\sum_{k=1}^{K}\sum_{i=1}^{M}\sum_{j=1}^{N} h^{(l,\beta)}(m+r_1\times i, n+r_2\times j)\times W(i,j)+b\right) \tag{3.6}$$

Where $K = K^{(l,\beta)}$, $(r_1,r_2) = (r_1^{(l,\beta)}, r_2^{(l,\beta)})$, $W = W_k^{(l,\beta)}$ and $b = b_k^{(l,\beta)}$. The ouput of the CBAM block $h^{(l+1,\beta)}(m,n)$ is then computed by sequentially inferring a 1D channel attention map $\mathrm{M_c}$ and a 2D spatial attention map $\mathrm{M_s}$ following the original recommendations of the CBAM paper [Woo et al., 2018] and as illustrated in Figure 3.17:

$$\begin{aligned} \mathbf{F}'(m,n) &= \mathbf{M_c}(\mathbf{F}(m,n)) \otimes \mathbf{F}(m,n) \\ h^{(l+1,\beta)}(m,n) &= \mathbf{M_s}(\mathbf{F}'(m,n)) \otimes \mathbf{F}'(m,n) \end{aligned} \tag{3.7}$$

where $\otimes$ denotes element-wise multiplication. Finally, the output $h^{(l+1,\beta)}(m,n)$ serves as the input of the batch normalization and pooling layer that directly follow the atrous CBAM block.

In our experiments, we have three branches: low resolution, medium resolution, high resolution branches

$r_1^{(l,\beta)} \in [1;3]$, $r_2^{(l,\beta)} = 1$, $\beta \in [1;3]$, three atrous CBAM blocks and pooling layers in each branch: $l \in [1;3]$. $K^{(l,\beta)} = 64$ feature maps for each layer. Each convolution uses 3x3 kernels and is followed by a batch normalization layer. All the neurons use the LeakyRelu activation function: $\sigma(x) = max(0.2\text{x}, \text{x})$, with the exception of the M_c and M_s neurons which use the same hyper-parameters settings than the original CBAM paper.

Modeling Location-viewpoint Invariant Features via U-GRUs To extract skeletal pose kinematic features invariant to locations and viewpoint, we represent a pose sequence as its evolution of skeletal joints relative Euclidean distances over time with the Joint Collection Distances (JCD) feature [Yang et al., 2019]. The JCD feature is then flattened to a vector of dimension $m * \binom{N}{2}$. Euclidean distances features are then processed with two U-GRUs blocks as illustrated in Figure 3.17. In contrast to SPI-Net, since we use recurrent neural networks and not a fully connected approach anymore, we do not handpick the keypoints of interest for the Euclidean distance matrix, we simply consider them all as it does not impact the speed of the approach much.

Compared to regular Bidirectional GRUs, where the output layer can get information from past and future states simultaneously but are most sensitive to the input values around time t, in U-GRUs, past and future interact but in a limited way. U-GRUs allow the model to accumulate information while knowing which part of the information will be useful in the future and therefore exploit long-term temporal patterns on invariant locations and viewpoint skeletal dynamics. This compensates for the inabilities of the first pseudo-images stream to learn long-range temporal patterns and therefore acts as a regularizer for time and space. Similarly, context features such as bounding box positions and ego-vehicle speed are processed in parallel through the same U-GRUs architecture.

Combining all the features branches Following the successful application of temporal attention and modality attention in multi-modal approaches for pedestrian action prediction, we finally apply the same temporal attention and modality attention mechanisms used in PCPA [Kotseruba et al., 2021] to all our features branches to fuse them effectively. Nonetheless, the nature of the inputs merged in TrouSPI-net is entirely different compared to the initial multi-modal PCPA [Kotseruba et al., 2021] architecture. While PCPA [Kotseruba et al., 2021] merges inputs such as sequences of RGB camera images processed by 3D convolution and poses processed via simple recurrent networks without spatio-temporal coherence of body actions, TrouSPI-Net was designed to operate without needing additional RGB scene-context and uses different body poses representations that were encoded to treat the spatial and the temporal information of body action for different time resolutions. For each feature extracted by U-GRUs: we apply temporal attention [Kotseruba et al., 2021] to weight the relative importance of frames in the observation relative to the last seen frame. We then apply modality attention [Kotseruba et al., 2021] to the weighted outputs of the U-GRUs features and the output of the pseudo-images stream. This fuses inputs from multiple modalities into a single representation by weighted summation of the information from individual modalities. The output of the modality attention block is finally passed to a dense layer for prediction.

3.4.2.2 Experiments

To evaluate the presented multi-branch approach and several variations of its architecture, we conducted experiments on two large public data-sets for studying pedestrian behaviors in traffic: JAAD [Rasouli et al., 2017b, Rasouli et al., 2017a] and PIE [Rasouli et al., 2019a]. JAAD contains 346 clips and focuses on pedestrians intending to cross, PIE contains 6 hours of continuous footage and provides annotations for all pedestrians sufficiently close to the road regardless of their intent to cross in front of the ego-vehicle and provides more diverse behaviors of pedestrians.

Evaluation Setup We base our experiments on the newly proposed evaluation criteria [Kotseruba et al., 2021] with common evaluation protocols, splits and normalized modalities inputs. As provided in the new benchmark, observation data for each pedestrian is sampled so that the last frame of observation is between 1s and 2s before the crossing event. We report the results using regular classification metrics: accuracy, AUC, precision, recall and F_1-score given by $F_1 = \frac{2\times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$.

In architecture variations and branch ablations studies, we explore how each TrouSPI-Net component contributes to the pedestrian action prediction performance by removing one component while keeping others unchanged. We also explore the performance of CBAM blocks in the pseudo-image stream by comparing them to similar self-attention blocks designed for 2D convolutions: Squeeze and Excitation method (SE blocks) [Hu et al., 2018]. Finally, we explore the impact of adding a second modality to TrouSPI-Net by using 3D convolutions [Tran et al., 2014] on the local box feature available in the data-set.

Implementation details We use U-GRUs with 64 hidden units for encoding all features, except the pseudo-image. L2 regularization of 0.001 is added to the final dense layer and a dropout of 0.5 is added after the attention block. The number of observation frames m is set to 16. Body poses extracted by OpenPose [Cao et al., 2017] and proposed in the benchmark [Kotseruba et al., 2021] are sampled in the form of a 3-dimensional (16,18,2)-shaped tensor for the pseudo-images stream and 2-dimensional (16,153)-shaped tensor for the U-GRUs stream. The ego-vehicle speed feature is used only in the PIE data-set and omitted in JAAD. To compensate for the significant class imbalance, we apply class weights inversely proportional to the percentage of samples of each class in each data-set. We train the model with Ranger Optimizer: a combination of Lookahead ($k = 6, \alpha = 0.5$) [Zhang et al., 2019] and Radam [Liu et al., 2019], binary cross-entropy loss and batch size set to 8. We train for 80 epochs with learning rate set to 5.0e-05 for PIE and 5.0e-06 for JAAD.

Discussion The results of the final TrouSPI-Net model are presented in Table 3.8. Results are most improved compared to State-of-the-Art on the PIE data-set, where accuracy is increased by 1%, AUC by 2% and F_1-score by 3% compared to PCPA [Kotseruba et al., 2021], a model with two perception modalities: RGB images and poses. On JAAD, our model performs comparably if not better with state-of-the-art across some metrics. This leads us to believe that approaches using only one additional network to compute perception modalities can be competitive with approaches that combine multiple. A comparison of F_1-scores between our approach and the best-performing methods that exist at this day shows that

Table 3.8: Evaluation results for baseline and state-of-the-art models and their variants on PIE and JAAD data-sets. Dashed lines separate different types of architectures. Modalities correspond to the type of networks used in the given approach, Model Params corresponds to the size of the network compiled on the benchmark [Kotseruba et al., 2021] with Additional Costs (Optical flow, Body Pose, RGB features) already extracted.

Model Name	Model Variants	Model Params (Additional Costs)	*PIE*					$JAAD_{behavior}$					$JAAD_{all}$				
			ACC	**AUC**	**F1**	**P**	**R**	**ACC**	**AUC**	**F1**	**P**	**R**	**ACC**	**AUC**	**F1**	**P**	**R**
Static	VGG16 [Simonyan and Zisserman, 2014]	14.7M	0.71	0.60	0.41	0.49	0.36	0.59	0.52	0.71	0.63	0.82	0.82	0.75	0.55	0.49	0.63
	Resnet50 [He et al., 2016]	23.6M	0.70	0.59	0.38	0.47	0.32	0.46	0.45	0.54	0.58	0.51	0.81	0.72	0.52	0.47	0.56
ATGC [Rasouli et al., 2017b]	AlexNet	58.3M	0.59	0.55	0.39	0.33	0.47	0.48	0.41	0.62	0.58	0.66	0.67	0.62	**0.76**	**0.72**	0.80
ConvLSTM [Shi et al., 2015]	VGG16	0.001M (VGG)	0.58	0.55	0.39	0.32	0.49	0.53	0.49	0.64	0.64	0.64	0.63	0.57	0.32	0.24	0.48
	ResNet50	0.001M (Resnet)	0.54	0.46	0.26	0.23	0.29	0.59	0.55	0.69	0.68	0.70	0.63	0.58	0.33	0.25	0.49
SPI-Net [Gesnouin et al., 2020]	CNN MLP	0.1M (OpenPose)	0.66	0.54	0.30	0.35	0.27	0.58	0.55	0.66	0.67	0.65	0.81	0.72	0.52	0.48	0.58
SingleRNN [Kotseruba et al., 2020]	LSTM	1.4M (2*VGG,OpenPose)	0.83	0.77	0.67	0.70	0.64	0.58	0.54	0.67	0.67	0.68	0.65	0.59	0.34	0.26	0.49
	GRU	1.0M (2*VGG,OpenPose)	0.81	0.75	0.64	0.67	0.61	0.51	0.48	0.61	0.63	0.59	0.78	0.75	0.54	0.44	0.70
MultiRNN [Bhattacharyya et al., 2018]	GRU	1.8M (2*VGG,OpenPose)	0.83	0.80	0.71	0.69	0.73	0.61	0.50	0.74	0.64	0.86	0.79	0.79	0.58	0.45	0.79
StackedRNN [Yue-Hei Ng et al., 2015]	GRU	2.6M (2*VGG,OpenPose)	0.82	0.78	0.67	0.67	0.68	0.6	**0.6**	0.66	**0.73**	0.61	0.79	0.79	0.58	0.46	0.79
HierarchicalRNN [Yong Du et al., 2015]	GRU	3M (2*VGG,OpenPose)	0.82	0.77	0.67	0.68	0.66	0.53	0.5	0.63	0.64	0.61	0.80	0.79	0.59	0.47	0.79
SFRNN [Rasouli et al., 2019b]	GRU	2.6M (2*VGG,OpenPose)	0.82	0.79	0.69	0.67	0.70	0.51	0.45	0.63	0.61	0.64	0.84	0.84	0.65	0.54	**0.84**
C3D [Tran et al., 2014]	RGB	78M	0.77	0.67	0.52	0.63	0.44	0.61	0.51	0.75	0.63	**0.91**	0.84	0.81	0.65	0.57	0.75
I3D [Carreira and Zisserman, 2017]	RGB	12.3M	0.80	0.73	0.62	0.67	0.58	0.62	0.56	0.73	0.68	0.79	0.81	0.74	0.63	0.66	0.61
	Optical flow	12.3M (FlowNet2)	0.81	0.83	0.72	0.60	**0.9**	0.62	0.51	0.75	0.65	0.88	0.84	0.80	0.63	0.55	0.73
TwoStream [Simonyan and Zisserman, 2014]	VGG16	134.3M (FlowNet2)	0.64	0.54	0.32	0.33	0.31	0.56	0.52	0.66	0.66	0.66	0.60	0.69	0.43	0.29	0.83
PCPA [Kotseruba et al., 2021]	Temp. +mod. attention	31.2M (C3D,OpenPose)	0.87	0.86	0.77	-	-	0.58	0.5	0.71	-	-	**0.85**	**0.86**	0.68	-	-
TrouSPI-Net (ours)	CBAM attention block	1.5M~ (OpenPose)	**0.88**	**0.88**	**0.80**	0.73	0.89	**0.64**	0.56	**0.76**	0.66	**0.91**	**0.85**	0.73	0.56	0.57	0.55
	SE attention block	1.5M (OpenPose)	**0.88**	0.87	**0.80**	**0.77**	0.84	**0.64**	0.55	**0.76**	0.65	**0.91**	0.82	0.77	0.58	0.49	0.70

Table 3.9: Architecture variations and Ablation studies for TrouSPI-Net on PIE data-set.

Model Variants (Additional Costs)	**Params**	**ACC**	**AUC**	**F1**
TrouSPI-Net without euclidean distances	1.4M	0.87	0.85	0.78
TrouSPI-Net without parallel atrous branches	0.8M	0.86	0.80	0.72
TrouSPI-Net without Ego-Vehicle Speed	1.4M	0.85	0.84	0.76
TrouSPI-Net GRUs	1.3M	0.85	0.80	0.72
TrouSPI-Net BiGRUs	1.6M	0.86	0.82	0.75
TrouSPI-Net without attention Block	1.4M	0.87	0.85	0.78
TrouSPI-Net with SE attention Block	1.5M	**0.88**	0.87	**0.80**
TrouSPI-Net	1.5M	**0.88**	**0.88**	**0.80**
TrouSPI-Net with two modalities (C3D)	30.2M	**0.88**	0.87	**0.80**

our approach offers better F_1-scores for two out of three benchmarks. It shows that TrouSPI-Net is more balanced than other approaches for the task of pedestrian crossing prediction. Finally, results obtained by TrouSPI-Net on $JAAD_{all}$ should be taken with a pinch of salt since the data-set considers all the visible pedestrians who are far away from the road and are not crossing. Since pose estimation algorithms are still struggling with scale to extract informative poses for people at the back of a scene, TrouSPI-Net does not manage to extract discriminating features because of the low quality of the poses extracted and relies mainly on other features to realize its inference. This explains its lower performance compared to the two other benchmarks. However, it should not be considered an issue since those pedestrians are not directly interacting with the vehicle in any way. If they were to become a danger in the future, they would have to step closer to it, and therefore pose estimation algorithms should be able to extract informative poses.

3.4.2.3 Architecture variations and branch ablations

Table 3.9 shows that removing the parallel atrous branches from the pseudo-image stream leads to a degradation of the performance indicators (Acc, AUC, F_1) on PIE data-set by respectively, 2%, 8% and 8%. Similarly, removing the stream acting as a regularizer with relative distances degrades the performance indicators by respectively 1%, 3% and 2%. Therefore, we can highlight the importance of the three parallel branches to extract spatio-temporal features for different time scales and the importance of the euclidean distances stream to act as a regularizer for the overall approach performance. Another interesting fact to mention is the performance drop of respectively 3%,4%,4% when the ego vehicle speed input is missing. This shows that of all the possible forms of communication by a pedestrian to announce that he or she wants to cross, pose kinematics could easily be supplemented with additional information such as vehicle/pedestrian communication forms provided by the ego-vehicle.

Secondly, we evaluate the importance of using a spatio-temporal attention module over the parallel pseudo-images extraction module. We first disregard spatio-temporal attention completely in the given pseudo-images stream and then replace CBAM blocks [Woo et al., 2018] with SE blocks [Hu et al., 2018]. Experimental results show that removing the attention-enhanced 2D atrous convolutions degrades the performance indicators by respectively 1%, 3%, 2%, whereas replacing CBAM blocks [Woo et al., 2018] by SE blocks [Hu et al., 2018] do not drastically impact TrouSPI-Net's performance and even increases it across some metrics according to Table 3.8. In conclusion, introducing a spatio-temporal attention

module over the parallel features extraction module seems to improve our model performance. Future studies could fruitfully explore this further by introducing a custom spatio-temporal attention module specifically designed for the parallel pseudo-images extraction module.

Finally, we evaluate the importance of U-GRUs by replacing them with GRUs and Bidirectional GRUs. Table 3.9 results show that both modified approaches lead to a degradation of the performance indicators by respectively 3%, 8%, 8% and 2%, 6%, 5%. Therefore, we can highlight the importance of U-GRUs to exploit the bidirectional temporal and long-term contexts compared to other state-of-the-art approaches designed to capture sequential features. It also leads us to believe that an effective pedestrian action prediction model should focus on both long-term dependencies and multi-scale short temporal features to be effective.

3.4.2.4 TrouSPI-net's comparison with other pose-only methods

By using the newly proposed evaluation procedures [Kotseruba et al., 2021] proposed by the authors of JAAD and PIE which was designed to ensure a fair comparison between all the pedestrian prediction approaches, we could not compare TrouSPI-Net to previous approaches that were not evaluated on the given benchmark including proposed pose-only based approaches [Fang and López, 2018, Ranga et al., 2020, Marginean et al., 2019, Ghori et al., 2018, Cadena et al., 2019, Gesnouin et al., 2020]. Even if their reported performance is evaluated on the same data-sets, they are reported under different experimental conditions and definitions.

To provide a fair comparison between TrouSPI-Net and another pose-based approach, we used the orginal implementation of SPI-Net and then extended it to evaluate it with the new benchmark in order to ensure that TrouSPI-Net provided better performance for all the data-sets than its previous version designed for short-term prediction (previously used for a prediction horizon of a single frame with an observation length of 300 frames on JAAD). Experimental results shown in Table 3.8 show that TrouSPI-Net always outperforms the extended SPI-Net [Gesnouin et al., 2020] by a large margin on PIE, and outperforms the SPI-net approach on JAAD by 6%, 1%, 10% and 4%,1%,4%. Despite the limitations of comparing TrouSPI-Net with only one existing pose-only method, our results demonstrate that TrouSPI-Net is way more reliable than SPI-net for the three benchmarks. Therefore, it should be considered as the first pose-only based approach proposed on the benchmark at this day and should become an interesting baseline to easily compare to for future works using the new evaluation procedures.

3.4.2.5 Using a second perception modality with TrouSPI-Net

One of the main advantages of using a scene-agnostic model using such sparse perception modality instead of aggregating multiple perception modalities is the smaller model size leading to an easier deployment into embedded devices, as table 3.10 shows. Moreover, when combined with 3D convolutions of cropped images including the pedestrians, TrouSPI-Net's computational costs dramatically grows without gaining any performance on PIE data-set as tables 3.9 and 3.10 show. This may be considered a further validation of pose-based only networks for Pedestrian Action Prediction as lightweight models designed for embedded devices with real-time constraints, which do not need additional context input to work effectively. While this affirmation is established for pedestrians where pose estimation inferences are possible and with limited occlusions, the question remains open for scenes with very high occlusions

between pedestrians, occlusions between pedestrians and scene objects, or abnormal behaviors such as crowd movement. For those cases, implementing a way to treat Static RGB images effectively as a context feature might still prove important.

Table 3.10: Architecture comparison of floating-point operations per second (FLOPS) in millions, Cuda Memory Usage (CMU) in Megabytes and Weights Memory Requirements (WMR) in Megabytes. RGB features extracted by CNNs are taken into consideration during computations.

Model(Additional Costs)	**FLOPS (Mio.)**	**CMU (MB)**	**WMR (MB)**
VGG16 [Simonyan and Zisserman, 2014]	29.4	72.1	56.1
Resnet50 [He et al., 2016]	47.0	47.0	90.0
ConvLSTM [Shi et al., 2015] (VGG)	29.5	93.5	56.2
SingleRNN [Kotseruba et al., 2020] (2 VGG)	65.3	145.3	60.0
MultiRNN [Bhattacharyya et al., 2018] (2 VGG)	71.6	146.0	63.0
StackedRNN [Yue-Hei Ng et al., 2015] (2 VGG)	76.3	146.8	66.0
SFRNN [Rasouli et al., 2019b] (2 VGG)	73.6	146.5	64.5
C3D [Tran et al., 2014]	156.0	182.6	297.5
I3D [Carreira and Zisserman, 2017]	24.6	334.1	46.9
PCPA [Kotseruba et al., 2021] (C3D)	220	320.2	414.9
SPI-net [Gesnouin et al., 2020]	**0.3**	**2.5**	**0.3**
TrouSPI-Net (ours)	3.0	6.8	5.4
TrouSPI-Net with two modalities (C3D)	216.7	322.6	412.9

3.4.2.6 The drawbacks of relying on additional networks to extract perception modalities

Although other models also apply additional networks to extract multiple perception modalities such as pose, flow or background context and the proposed approach beats the state-of-the-art while being smaller in comparison according to Table 3.8, its application also relies on one additional algorithm to operate. If TrouSPI-Net was to be implemented outside of the JAAD and PIE benchmark, one would have to add to TrouSPI-Net's size the pose extraction model used to compute the pose information. In our case, OpenPose [Cao et al., 2017] was used to compute the inputs available in [Kotseruba et al., 2021]. Therefore, the overall approach is $\sim$ 53.5M parameters. However, it leads to a practical methodology as interchanging the additional approaches to extract poses does not jeopardize the TrouSPI-Net approach. Contrary to image-based approaches, if improvements such as inference time or average precision by key-points were made in the field of pose estimation, TrouSPI-Net could still be applicable without any modification. Moreover, the proposed benchmark [Kotseruba et al., 2021] currently omits a major issue for pedestrian intention prediction: temporal tracking of pedestrians to avoid mixing identities over time. Such questions are rarely raised and approaches mainly rely on the ground-truth IDs of each pedestrian. However, such concerns are mandatory to easily transpose the pedestrian action prediction approaches into real-life scenarios without pedestrians' ground-truth IDs. In TrouSPI-Net's case, to ensure a better tracking of the protagonists in the scene and avoid mixing the identities of two protagonists, one could for example replace OpenPose [Cao et al., 2017] by pose estimation networks sequentially based on pose matching for tracking [Xiu et al., 2018, Ning and Huang, 2019, Raaj et al., 2019]. Such a substitution would provide the TrouSPI-Net model every modality it needs to work in a non-controlled environment with only one additional network: body poses, handcrafted body poses features, bounding boxes positions of the pedestrians and their respective individual ID's.

3.4.2.7 Conclusion

We introduced a new lightweight multi-branch neural network to predict pedestrians' actions using only one additional network to extract perception modalities: 2D pedestrian body poses. The proposed TrouSPI-Net model largely extends and improves the SPI-Net approach in several ways. First, we introduce parallel processing branches to allow the architecture to access different time resolutions with atrous convolutions enhanced with self-attention mechanisms. Secondly, we apply U-GRUs on the evolution of relative Euclidean body distances over time, which acts as a regularizer of the first stream for both time and space. We then extend the U-GRUs approach as one baseline method to consider long-term temporal coherence and process each sequence of context features such as bounding box positions or ego-vehicle speed with U-GRUs. Finally, following the newly proposed evaluation procedures and benchmarks for JAAD and PIE (two challenging pedestrian action prediction data-sets), our experimental results show that TrouSPI-Net achieved 76% F1 score on JAAD and 80% F1 score on PIE, therefore outperforming current state-of-the-art. This shows that using only body poses can outperform approaches that combine multiple networks to extract different perception modalities. Subsequently, our model inherits interesting properties such as being completely invariant to any scene-background context, leading to a lightweight approach focusing only on the pedestrian's movement. Therefore, we believe that TrouSPI-Net could be an interesting baseline to easily compare to for future works aiming at developing a pose-only based model for pedestrian intention prediction and has the potential to improve many other human action recognition or prediction tasks.[8]

3.5 Summary

In this chapter, we first provide an overview of existing approaches for pedestrian action prediction. The majority of existing techniques for pedestrian action prediction are trajectory-based, which means they depend on previously observed pedestrian positions to anticipate pedestrian positions in the future. These methods are successful when pedestrians have already crossed or are going to cross, i.e., these algorithms react to an action that has already begun rather than predicting it. We first propose an asymmetrical bidirectional recurrent neural network architecture called U-RNN to encode pedestrian trajectories and evaluate its relevance to replace LSTMs for various trajectory-based models. Our results show that there is still room for improvement in coordinates-only approaches, and indicates that interactions are not the only aspect on which pedestrian trajectory prediction can progress. Thereafter, we address the problem of pedestrian discrete intention prediction: instead of focusing on continuous trajectories describing the expected future movement of the pedestrian and merely relying on scene dynamics to predict intentions, we define the intentions of a pedestrian as a combination of his/her high-level discrete behaviors such as his/her pose dynamics or head orientation... We then make the connection between the research questions addressed for human action recognition in chapter 2 and pedestrian discrete intention prediction. Considering the importance of crossing prediction algorithms to run efficiently for real-time usage while being robust to a multitude of complexities and conditions, we propose SPI-Net and TrouSPI-Net: two scene-agnostic, lightweight, multi-branch approaches that rely on pose kinematics to predict crossing behaviors. The proposed approaches could be applied following the application of any additional network

[8] We refer the reader to Appendix A

to compute pedestrian body poses and could be easily implemented in any embedded devices with real-time constraints and also in any neural hardware solution like Intel Movidius©, or FPGA since it only uses standard deep-learning operations in an euclidean grid space. Finally, We show that it is possible to make the link between the posture, the walking attitude and the future behaviours of the protagonists of a scene without using the contextual information of the scene (pedestrian crossing, traffic light...). This allowed us to divide by a factor of 20 the inference speed of existing approaches for pedestrian intention prediction while keeping the same prediction robustness.

Chapter 4

Assessing the Generalization of Pedestrian Crossing Predictors

Contents

4.1 Introduction **90**

4.2 The past, current and future state of pedestrian intention prediction benchmarks? **91**

4.2.1 Stone Age: prior to the release of the standardized evaluation procedures . . . 91

4.2.2 Bronze Age: one benchmark to rule them all 92

4.2.3 Iron Age: identifying the generalization capabilities of our models? 93

4.3 Sutor, ne ultra crepidam, or the necessity of uncertainty **94**

4.4 Generalization Capabilities **96**

4.4.1 Datasets and Implementation Details 96

4.4.2 Baselines and state-of-the-art models 96

4.5 New Evaluation Paradigm **98**

4.5.1 Cross-dataset Evaluation Results 98

4.5.2 Role of pre-training in uncertainty calibration 101

4.6 Improving Uncertainty Calibration **102**

4.6.1 Baselines from the probabilistic deep learning literature 104

4.6.2 Discussion 104

4.7 Summary **105**

4.1 Introduction

In a short novel entitled funes el memorioso, published in 1942, the argentine writer Jorge Luis Borges tells the story of a young man with a memory so prodigious that he was incapable of ignoring the many details invisible or insignificant to other humans. Far from being an advantage, his inability to ignore the variations he observed had quickly proved to be a great handicap. It was simply unthinkable for him to use the same term for different objects. Impossible to acknowledge that a dog seen in profile at a particular moment could have the same name as the dog seen from the front a minute later. "*To think is to forget a difference, to generalize, to abstract. In the overly replete world of Funes, there were nothing but details*." Unable to forget, Funes was unable to think. On the opposite side of the spectrum of generalization but still inspired by literature, popular for his apothegms and his taste for generalization at all costs, Orwell. "*Here are a couple of generalizations about England that would be accepted by almost all observers. One is that the English are not gifted artistically. [...] the English are not intellectual. They have a horror of abstract thought, they feel no need for any philosophy or systematic world-view.*"[1] Between this Orwellian way of presenting, as a self-evident truth, a fact or a personal experience that may have a general value, but which is not explained, supported or illustrated [2][3] and Funes' incapacity to generalize there should be a proper balance between both ways of thinking and abstracting.

In machine learning, generalization also plays an important role, we don't just want a model to learn to model the training data. We want it to generalize to data it has not seen yet. We want a model to acknowledge what Funes could not: that a rotated dog still is the same dog and that is without seeing dogs where none are present. In our research field, there is a convenient way of assessing such generalization capacity: we assess the performance of a held-out test set, which consists of cases that the model has not seen previously. Sampling bias is a kind of overfitting which is that in a real-world scenario, input distributions are frequently shifted from the training distribution. The network could therefore exploit accidental regularities available in both the training set and testing set but not available in a real-world scenario. The network would then, without understanding the true regularities, accurately classify all the training instances of a given class when evaluated via the dataset but would fail to work in real life. This is why over the last two decades, the computer vision paradigm has shifted. From a pure algorithm-based vision, we have given more and more interest to the data, until we arrived at a paradigm combining both: *"We started our search for a new approach with one key assumption: even the best algorithm would not generalize well if the data it learned from did not reflect the real world. In concrete terms, that meant that major advances in object recognition could occur only from access to a large quantity of diverse, high-quality training data."* [Fei-Fei and Krishna, 2022]. Even the best behavior prediction algorithm would not generalize well if the data it learned from did not reflect the real world. Yet, the full complexity of the real world cannot be encapsulated in the training data, no matter how big the dataset is.

In this chapter, we aim at showing that current evaluation protocols do not adequately represent the applicability of existing pedestrian prediction models for real-world scenarios. Comparable studies have

[1] The Lion and the Unicorn: Socialism and the English Genius - Orwell 1941
[2] "All art is propaganda."
[3] "All revolutions are failures, but they are not the same failures."

previously been conducted in computer vision, questioning whether recent progress on the ImageNet [Russakovsky et al., 2015] benchmark continues to represent meaningful generalization [Beyer et al., 2020] and identifying various sources of bias and noise [Stock and Cisse, 2018, Northcutt et al., 2021]. However, going beyond accuracy to evaluate a model for a high-risk application with a limited amount of training data, such as pedestrian crossing prediction, has never been properly investigated.

4.2 The past, current and future state of pedestrian intention prediction benchmarks?

4.2.1 Stone Age: prior to the release of the standardized evaluation procedures

As seen in section 3.3.2, pedestrian crossing prediction has been a topic of active research, resulting in many new algorithmic solutions. However, due to differences in their evaluation criteria, comparing them used to be a somewhat unsatisfactory practice to say the least, hardly possible to be perfectly honest. Table 4.1 lists all the pedestrian action prediction models trained and evaluated on JAAD and PIE datasets prior to the release of the standardized benchmarks and evaluation procedures. As previously

Table 4.1: Pedestrian action prediction models trained and evaluated on JAAD and PIE datasets prior to the standardized benchmarks and evaluation procedures.

Model	Dataset	Observation endpoint	Observation length (s)	Prediction Horizon (s)
[Rasouli et al., 2017b]	JAAD	before event	0.3-0.5	next frame
[Fang and López, 2018]	JAAD	all frames	0.46	next frame
[Varytimidis et al., 2018]	JAAD	before event	one frame	next frame
[Cadena et al., 2019]	JAAD	before event	10	next frame
[Gujjar and Vaughan, 2019]	JAAD	all frames	0.533	0.533
[Neogi et al., 2020]	JAAD	before event	-	1.33
[Pop et al., 2019]	JAAD	all frames	0.666	1.33
[Marginean et al., 2019]	JAAD	all frames	0.1	next frame
[Saleh et al., 2019]	JAAD	all frames	0.533	next frame
[Rasouli et al., 2019b]	PIE	before event	0.5	2
[Chaabane et al., 2020]	JAAD	all frames	0.533	0.533
[Kotseruba et al., 2020]	PIE	before event	0.5	0.3/0.5/1
[Gesnouin et al., 2020]	JAAD	before event	10	next frame
[Liu et al., 2020a]	JAAD	before event	-	1/2/3
[Piccoli et al., 2020]	JAAD	before event	0.533	0.533
[Ranga et al., 2020]	JAAD	all frames	0.5/1	1
[Singh and Suddamalla, 2021]	JAAD	before event	0.533	next frame
[Liu et al., 2020a]	JAAD	before event	1	next frame

stated in section 3.3.2, back in those days, anyone with a new interesting approach could then propose their evaluation protocols and therefore claim state-of-the-art without ensuring a proper comparison with others. Approaches using the entire sequence including the crossing section would claim high prediction accuracy whereas the term prediction would no longer apply since they inferred a behavior while the action had already begun. About half of the proposed approaches used sequences up to the frame preceding the crossing event for training and evaluation, while some had a prediction horizon up to three seconds later... To put it in a nutshell, the foundations of the field were thus more or less built on a comparison between apples and oranges. This particular research era where people started to get interested in discrete

intent prediction on the JAAD dataset lasted two years. Within those two years, we identified no less than 18 approaches that claimed state-of-the-art on JAAD which is roughly equivalent to a hot streak of improvements of 0.75 *SotA per month*. During this thesis, we also contributed to this problem, the idea is not to bite the hand that feeds you but to make the reader clearly understand the limitations that were quickly encountered in the domain to effectively sort out what was a real advance and what was not.

4.2.2 Bronze Age: one benchmark to rule them all

Intending to resolve some of the inconsistencies pointed out above, a standardized benchmark [Kotseruba et al., 2021] to evaluate pedestrian behavior prediction for three datasets was proposed to advance research further. In sixteen months of existence, we identified 8 approaches claiming state-of-the-art as Table 4.2 shows. This is roughly equivalent to 0.50 *SotA per month*. The validity of our metric to evaluate the speed of advances for the given research area is questionable. The fact is, having standardized evaluation protocols seems to smooth out some communications and the amount of noisy results. In a way, we believe that this brought a breath of fresh air to the field of pedestrian behavior prediction.

Table 4.2: List of all the pedestrian action prediction models trained and evaluated on the standardized benchmarks

Model Name	PIE			$JAAD_{behavior}$			$JAAD_{all}$		
	ACC	**AUC**	**F1**	**ACC**	**AUC**	**F1**	**ACC**	**AUC**	**F1**
PCPA baseline [Kotseruba et al., 2021]	0.87	0.86	0.77	0.58	0.5	0.71	0.85	0.86	0.68
Capformer [Lorenzo et al., 2021a]	-	0.85	0.79	-	0.55	0.74	-	0.70	0.51
Intformer [Lorenzo et al., 2021b]	0.89	0.92	0.81	0.59	0.54	0.69	0.86	0.78	0.62
TrouSPI-Net [Gesnouin et al., 2021]	0.88	0.88	0.80	0.64	0.56	0.76	0.85	0.73	0.56
TED [Achaji et al., 2021]	0.91	0.91	0.83	-	-	-	-	-	-
BiPed[Rasouli et al., 2021]	0.91	0.90	0.85	-	-	-	-	-	-
Mask PCPA [Yang et al., 2022]	-	-	-	0.62	0.54	0.74	0.83	0.82	0.63
PedGraph+ [Cadena et al., 2022]	0.89	0.90	0.81	0.70	0.70	0.76	0.86	0.88	0.65

However, proposed approaches evaluated on the benchmarks [Kotseruba et al., 2021] constantly report higher classification scores, giving the impression of clear improvements in pedestrian intention prediction. Usually, a new algorithm is proposed and the implicit hypothesis towards the proposed contribution is made such that it yields an improved performance over the existing state-of-the-art[4]. To confirm such hypothesis, an empirical evaluation of the given contribution is realized in a direct train-test sets evaluation and the quality of the model is evaluated by regular classification metrics: newly proposed methods are then claimed as the new state-of-the-art as soon as they outperform previous ones even by a small margin. However, as we saw in section 4.1, the ranking of the methods for a given task is currently only as good as the quality of the data used for comparison purposes, and the results obtained by one method on a given dataset do not always reflect its robustness in real-world applications. In addition, some approaches do not report their results on all three datasets of the standardized evaluation procedures. Although this may be due to an oversight, a desire not to burden the paper with numbers, it does not help in the comparison to the existing if the communications are only cherry-picked on the datasets where a given approach shines. This perspective encourages us even more to look at the

[4]which is the PCPA baseline in most cases.

field's structural failings, not simply the personal shortcomings of individual models, papers, or research groups.

4.2.3 Iron Age: identifying the generalization capabilities of our models?

We believe that there is still room for improvement to efficiently compare and rank the existing. For instance, knowing how well existing predictors react to unseen data remains an unanswered question. Nevertheless, this evaluation is imperative as serviceable crossing behavior predictors should be set to work in various scenarios without compromising pedestrian safety due to misprediction.

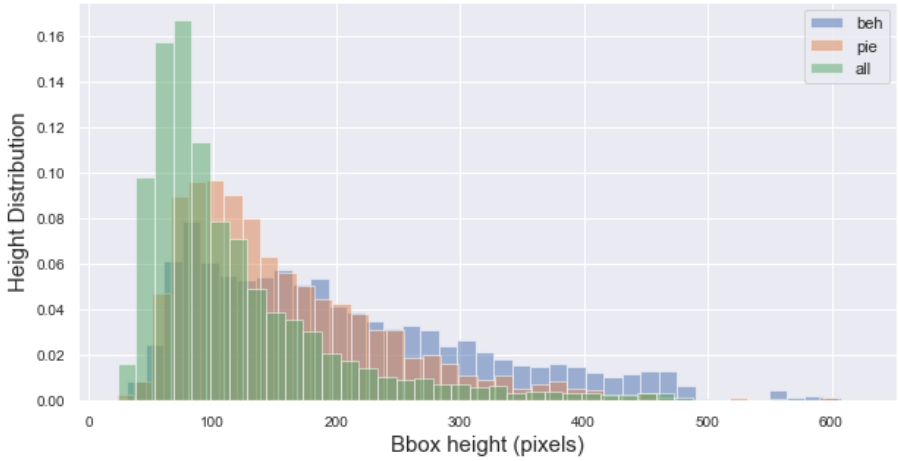


Figure 4.1: Distribution of pedestrian bounding box height in pixel for PIE, $JAAD_{behavior}$ and $JAAD_{all}$.

In this chapter, we assess how pedestrian action prediction approaches react to small domain shifts and evaluate their generalization capability outside a standard train-test evaluation protocol. We show that all the current pedestrian behavior predictors show signs of over-fitting when evaluated during a direct training-test sets evaluation setting on those standardized benchmarks. This problem leads to two major drawbacks for the field:

- The training source being generally not dense in variety of scenarios nor in the number of examples, the results of state-of-the-art approaches on each dataset might just come from noise: this noise effect should probably be further aggravated since the existing approaches are based on deep learning, depending heavily on the quantity and quality of data where the performance of approaches scales up with the amount of training data.
- It prevents pedestrian behavior predictors from scaling up to real-world applications, as they are not applicable in various scenarios with small domain shifts.

The above examples recap the general motivation of this work, encouraging us to rethink the evaluation methodology to rank current top-scoring behavior predictors from the perspective of uncertainty evaluation to small domain shifts. We argue that:

- The only empirical evaluation of models in a direct train-test sets evaluation offered by the original work introducing the method is not sufficient to effectively conclude anything about its applicabil-

ity in a real-world scenario. The result is often statistically non-significant during a cross-dataset evaluation scenario and leads to an ever-changing state-of-the-art.

- It would be more interesting to compare each method by evaluating how trustworthy are their uncertainty estimates under different domain shifts.

Figure 4.2: Examples of crossing and non-crossing pedestrians from $JAAD$ and PIE datasets. The conditions under which pedestrians act from one scenario to another can differ drastically concerning input format and domain shift: pedestrian size, pedestrian positioning in the scene, illumination conditions, occlusion...

To do so, we evaluate how pedestrian action prediction approaches react to small domain shifts by interchanging the training set of dataset A by the training set of dataset B and test it on the testing set of A. The given training routine is consistent across all experiments for all three datasets. This is referred throughout the manuscript as cross-dataset evaluation [Hasan et al., 2021, Chen et al., 2020, Guo et al., 2020]. By adopting cross-dataset evaluation, we test the generalization abilities of several state-of-the-art pedestrian crossing predictors to distributional shift such as pedestrian size, as shown in Fig 4.1, pedestrian positioning in the scene, illumination conditions or occlusion as shown in Fig 4.2.

4.3 Sutor, ne ultra crepidam, or the necessity of uncertainty

It is always disturbing to discover that we are always more foolish than we think ourselves to be. Embarrassing that a well-trained expert can make mistakes in any of his or her choices. Numerous studies have documented this in every possible and unexpected use case. From the classic example in medicine where the chance that a child will be recommended for tonsillectomy depends principally on the physician rather than on the child's health [Bakwin, 1945][5] to economics where Kahneman and Tversky were awarded the Nobel Prize to show that people were not as rational as economic models assume[6] [Kahneman and Tversky, 2013]. This cognitive bias does not only affect professionals: when it comes to

[5] Diagnostic mistakes becoming more widely acknowledged as a public health problem, The Institute of Medicine claimed that *"most people will experience at least one diagnostic error in their lifetime" [Balogh et al., 2015]*

[6] *"Overconfident professionals sincerely believe they have expertise, act as experts and look like experts. You will have to struggle to remind yourself that they may be in the grip of an illusion." Daniel Kahneman*

driving, most people think their skills are above average [Roy and Liersch, 2013]. The point is that we should not take for granted the reliability and accuracy of any judge, no matter how expert and whether it is human or algorithmic. In some circumstances it is necessary, in addition to giving an accurate statement, to be able to quantify the certainty of the statement. This sometimes allows us to disambiguate certain situations.

In real-world scenarios, quantifying uncertainty is crucial as the input distributions are frequently shifted from the training distribution due to a number of causes such as sampling bias. Evaluating the generalization abilities of models by using cross-dataset evaluation and classification metrics only is not sufficient. In high-risk applications such as pedestrian behavior prediction, the idea that a model's predicted probabilities of outcomes reflect true probabilities of those outcomes is mandatory for high-level decisions (*i.e.*, vehicle planning module in crowded urban traffic environments). Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) are standard uncertainty[7] metrics in this context [Naeini et al., 2015, Guo et al., 2017, Heo et al., 2018, Ovadia et al., 2019]. Predictions are divided into M interval bins according to a given binning strategy, we then calculate the accuracy of each bin to estimate the predicted accuracy from finite data. Let B_m denote the set of sample indices for which prediction confidence is inside one interval bin. The accuracy of B_m is defined as

$$\text{acc}\left(B_m\right) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbf{1}\left(\hat{y}_i = y_i\right) \tag{4.1}$$

where $\hat{y}_i$ and y_i are respectively the predicted and true class labels for sample i. The average confidence within one interval bin B_m is defined as:

$$\text{conf}\left(B_m\right) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i \tag{4.2}$$

where $\hat{p}_i$ is the model confidence for sample i. Throughout our experiments, the maximum softmax probability [Hendrycks and Gimpel, 2016] is used as the confidence score. We therefore compare each model output pseudo-probabilities to its accuracy. We obtain the following metrics to rank methods based on their calibration:

Expected Calibration Error (ECE): takes a weighted average of the absolute difference in accuracy and confidence.

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\text{acc}\left(B_m\right) - \text{conf}\left(B_m\right)\right| \tag{4.3}$$

Maximum Calibration Error (MCE): measures the maximum discrepancy between accuracy and confidence.

$$MCE = \max_{m} \left|\text{acc}\left(B_m\right) - \text{conf}\left(B_m\right)\right| \tag{4.4}$$

[7]Because confidence is the additive inverse of uncertainty with regard to 1, the terms are often interchanged.

Since the underlying binning approach has a significant impact on the accuracy and reliability of ECE and MCE, we use an adaptive binning strategy [Ding et al., 2020] instead of a uniform partition[8]: the number of samples in a bin is adaptive to the distribution of the samples in the confidence range.

4.4 Generalization Capabilities

4.4.1 Datasets and Implementation Details

For this evaluation, we use two large public naturalistic datasets for studying pedestrian behavior prediction: $JAAD$ [Rasouli et al., 2017b] and PIE [Rasouli et al., 2019a]. These datasets are typically obtained by a vehicle-mounted camera as it navigates through crowded urban traffic environments: $JAAD$ contains 346 clips and focuses on pedestrians intending to cross, PIE contains 6 hours of continuous footage and provides annotations for all pedestrians sufficiently close to the road regardless of their intent to cross in front of the ego-vehicle and provides more diverse behaviors of pedestrians. There are significant differences between JAAD and PIE dataset in terms of sensors: 3 different cameras are used in JAAD with narrow FOV while PIE continuous footage was recorded with with a single wide-angle lens camera. The $JAAD$ dataset is split into $JAAD_{behavior}$ and $JAAD_{all}$. $JAAD_{behavior}$ is biased towards pedestrians attempting to cross the street (402 crossing out of 648) and the smallest dataset available. $JAAD_{all}$ adds all visible pedestrians in $JAAD$, regardless of their position in the scene and contains more non-crossing pedestrians (490 crossing out of 2580). Similarly, PIE contains more non-crossing pedestrians (512 crossing out of 1842). All three datasets are heavily skewed towards one class. To compensate for such significant datasets shifts label-wise, we train all our models using class weights inversely proportional to the percentage of samples for each class. Following the existing evaluation procedures [Kotseruba et al., 2021], we use the same data sampling method, the same splits and the same inputs sets for our experiments[9]. However, we disregard the ego-vehicle speed input for all our models as the sensor data used for the ego-vehicle speed is only available for PIE and could not be used for cross-dataset evaluation purposes. The observation length for all models is fixed at 16 frames. In order to combine different models trained on different data sets, the sample overlap is set to 0.8 for both PIE and $JAAD$ trainings. We report the results using standard binary classification metrics: AUC and F1 Score and standard confidence calibration metrics: adaptive ECE and MCE.

4.4.2 Baselines and state-of-the-art models

We select a subset of methods from the pedestrian crossing prediction literature, and more broadly, action recognition literature for their prevalence, practical applicability and diversity in terms of architectures and input modalities. These include:

- **VGG16** [Simonyan and Zisserman, 2014] and **Resnet50** [He et al., 2016] : two baseline static models that use only the last frame in the observation sequence to predict the crossing behavior of a pedestrian.

[8]https://github.com/yding5/AdaptiveBinning
[9]https://github.com/ykotseruba/PedestrianActionBenchmark

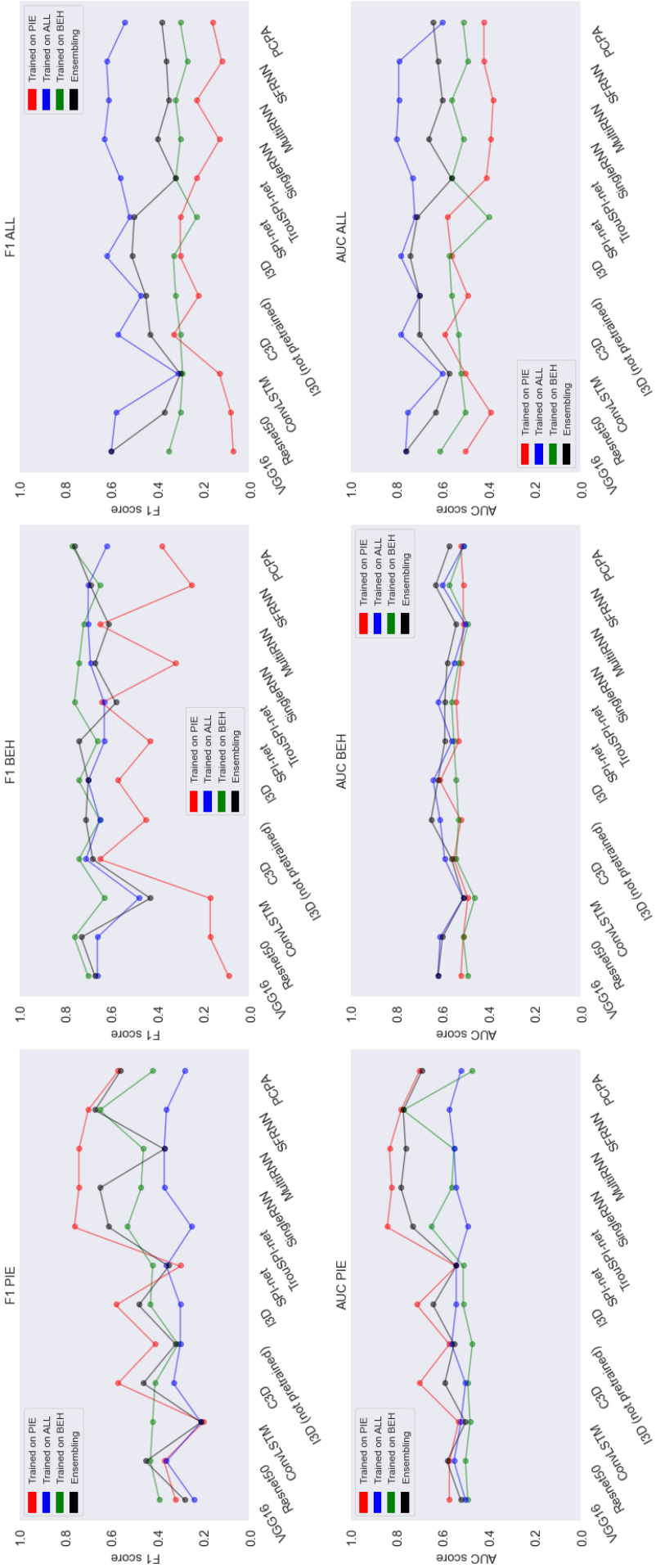


Figure 4.3: Pedestrian crossing prediction performance for PIE, $JAAD_{behavior}$ and $JAAD_{all}$. We show a comparison between traditional single-dataset train and test evaluation on each dataset compared to cross-dataset evaluation for eleven methods representing the diversity of architectures and modalities usually used for pedestrian crossing prediction. Ensembling denotes the average prediction given by the three models trained on each dataset for one given test set.

- **ConvLSTM** [Shi et al., 2015]: A model using a stack of images as input, pre-process those images with pre-trained CNN and apply ConvLSTM on those features.

- **Convolutional-3D** (*C3D*) [Tran et al., 2014] and **Inflated-3D** (*I3D*) [Carreira and Zisserman, 2017]: two models pretrained on Sports1M [Karpathy et al., 2014] using a stack of images as input and applying 3D convolutions to extract features.

- **SPI-net** [Gesnouin et al., 2020] and **TrouSPI-net** [Gesnouin et al., 2021]: two multi-modal models relying on pedestrians' pose kinematics extracted by OpenPose [Cao et al., 2017], relative euclidean distance of key-points and evolution of the pedestrian spatial positioning. Poses sequences are converted into 2D image-like spatio-temporal representations and self-spatio-temporal attention is applied via CNN-based models for multiple time resolutions. Each remaining feature is independently processed via either U-GRUs [Rozenberg et al., 2021] or feed forward neural network and fused by either applying temporal and modality attention or sent to a *fc* layer to predict crossing behaviors.

- **SingleRNN** [Kotseruba et al., 2020], **Multi-stream RNN** (*MultiRNN*) [Bhattacharyya et al., 2018] and **Stacked with multilevel Fusion RNN** (*SFRNN*) [Rasouli et al., 2019b]: Three multi-modal models relying on RGB Images extracted by VGG16 [Simonyan and Zisserman, 2014], pose kinematics extracted by OpenPose [Cao et al., 2017] and evolution of the pedestrian spatial positioning. Input features are either concatenated into a single vector and sent to a recurrent network followed by a *fc* layer for crossing prediction, either processed independently by GRUs [Chung et al., 2014] and the hidden state of GRUs are then concatenated and sent into a *fc* layer for crossing prediction or either processed by GRUs [Chung et al., 2014] and fused gradually at different levels of processing and complexity.

- **Pedestrian Crossing Prediction with Attention** (*PCPA*) [Kotseruba et al., 2021]: A multi-modal model relying on RGB images extracted by C3D [Tran et al., 2014], pose kinematics extracted by OpenPose [Cao et al., 2017] and evolution of the pedestrian spatial positioning. Non-images features are independently encoded by GRUs [Chung et al., 2014] and each is fed to a temporal attention block. 3D Convoluted features are flattened and fed into a *fc* layer. Modality attention is then applied to all the branches to fuse them into a single representation by weighted summation of the information from individual modalities.

4.5 New Evaluation Paradigm

4.5.1 Cross-dataset Evaluation Results

We present the coarse results of our cross-dataset evaluation in Fig 4.3. For readability purposes, the corresponding critical difference diagram is reported on Fig 4.5 and the average distribution of performance of the selected approaches is reported on Fig 4.4. The results of the average prediction given by the three models trained on each training set for one given test set are reported in Table 4.3. As expected, all methods, regardless of their architecture or input modalities, suffer a consequent performance drop

	AUC (↑)			F1(↑)			ECE(↓)			MCE (↓)		
Method	**pie** (±0.02)	**beh** (±0.02)	**all** (±0.01)	**pie** (±0.03)	**beh** (±0.01)	**all** (±0.02)	**pie** (±0.01)	**beh** (±0.02)	**all** (±0.02)	**pie** (±0.02)	**beh** (±0.03)	**all** (±0.03)
VGG16 [Simonyan and Zisserman, 2014]	0.52	**0.62**	**0.76**	0.28	0.67	**0.60**	0.07	0.06	0.20	0.24	**0.13**	0.25
Resnet [He et al., 2016]	0.58	0.60	0.63	0.45	0.68	0.37	0.09	0.05	**0.04**	0.37	**0.15**	0.44
ConvLSTM [Shi et al., 2015]	0.50	0.51	0.57	0.21	0.43	0.30	0.09	0.14	0.10	0.22	0.25	0.41
C3D [Tran et al., 2014]	0.59	0.56	0.70	0.46	0.73	0.43	0.17	0.08	**0.03**	0.43	**0.12**	**0.11**
I3D [Carreira and Zisserman, 2017]	0.64	**0.62**	**0.74**	0.48	0.71	0.51	**0.05**	0.08	0.09	**0.13**	**0.15**	**0.16**
PCPA [Kotseruba et al., 2021]	0.69	0.57	0.64	0.56	0.67	0.38	0.12	0.04	0.12	0.36	**0.13**	0.28
SingleRNN [Kotseruba et al., 2020]	**0.78**	0.58	0.66	**0.65**	0.69	0.40	0.09	**0.02**	0.09	0.16	**0.15**	**0.14**
MultiRNN [Bhattacharyya et al., 2018]	**0.76**	0.54	0.60	**0.64**	**0.74**	0.35	**0.06**	0.08	0.19	**0.13**	**0.17**	0.378
SFRNN [Rasouli et al., 2019b]	**0.77**	**0.63**	0.62	**0.67**	0.58	0.36	0.07	0.08	0.11	**0.11**	0.29	**0.16**
Spi-Net [Gesnouin et al., 2020]	0.54	0.59	0.71	0.35	0.61	0.50	0.10	0.07	0.22	0.30	**0.15**	0.33
TrouSPI-net [Gesnouin et al., 2021]	0.73	0.59	0.56	0.61	**0.76**	0.32	0.07	0.05	0.24	**0.12**	**0.13**	0.41

Table 4.3: Average prediction given by the three models trained on each training sets for one given test-set (Ensembling). In addition to classification metrics (we use arrows to indicate which direction is better), we compare models with predictive uncertainty metrics such as Expected Calibration Error (ECE) and Maximum Calibration Error (MCE). Dashed lines separate different types of architectures

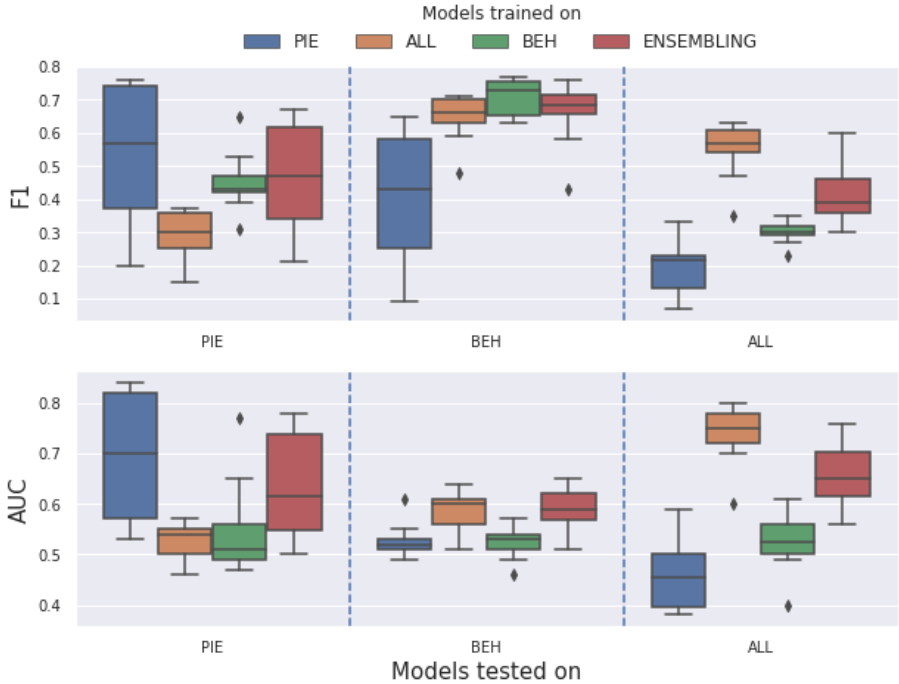


Figure 4.4: Distribution of the performance of the eleven selected approaches when evaluated in a direct train-test scenario and when evaluated in cross-dataset scenarios.

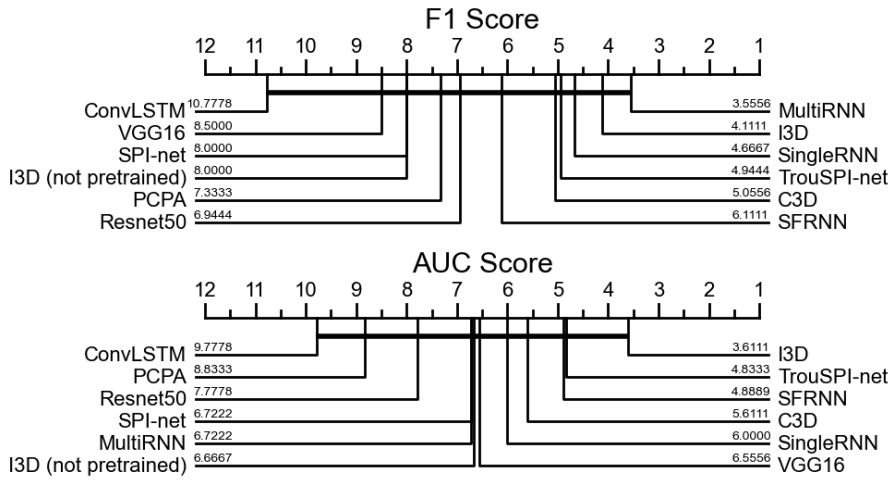


Figure 4.5: Critical Difference Diagram [Demšar, 2006]: first a Friedman test is performed to reject the null hypothesis, we then proceed with a post-hoc analysis based on the Wilcoxon-Holm method. We compare the robustness of classifiers over multiple training and testing sets shifts. We can see how each method ranks on average. A thick horizontal line groups a set of classifiers that are not significantly different ($\alpha = 0.1$).

when trained on PIE and tested on $JAAD$ or vice versa. Fig 4.4 shows that however robust the individual classifier is, there is a general trend for classifiers to decline when exposed to a different test set than the expected one. This is consistent with all our experiments with the exception of $JAAD_{behavior}$. $JAAD_{all}$ being an extension to the set of samples with behavioral annotations, $JAAD_{all}$ "generalizes" well on $JAAD_{behavior}$ but unsurprisingly, the converse is far from true. Even when trained on a relatively diverse dataset (PIE) and inferred on a smaller one in comparison ($JAAD_{behavior}$), selected

methods barely show signs of generalization. More alarming, some methods even under-performed a random binary guess based on class distribution when exposed to a different testing set than the expected one. While the task, standardized inputs and observation length are the same across all three datasets, none of the tested models reaches a satisfactory level of generalization across any other testing set. When it comes to comparing performance towards small domain-shift at the granular level of individuals methods, the critical difference diagram reported in Fig 4.5, shows that none of the selected methods arise as a clear winner when it comes to cross-dataset ranking. More importantly, the obtained ranks of each method when evaluated under cross-dataset evaluation are far from the ones we usually consider when developing pedestrian crossing behavior predictors: some general methods such as I3D or C3D are on par with multi-modal methods specifically designed to tackle the problem of pedestrian crossing prediction. While part of this could be due to the removal of ego-vehicle data which is an important source of information exploited by many multi-modal approaches, this still confirms the importance of rethinking the evaluation methodology of our approaches. The ensembling provided in Table 4.3, is the closest plausible approximation of the selected models' robustness for real-world application as it integrates all available conditions and training instances while removing the sampling biases of each specific training set. It shows that the only empirical evaluation of models in a direct train-test sets evaluation is not sufficient to effectively conclude anything about its applicability in a real-world scenario. This also demonstrates that the use of classification metrics alone is not representative of the overall capacity of the models. For two given models which are equivalent with respect to classification metrics (AUC or F1 score), their calibration (ECE and MCE) can differ drastically. This supports our argument that the usage of uncertainty metrics should complement the metrics conventionally used in order to obtain a comprehensive view of the robustness of existing approaches.

4.5.2 Role of pre-training in uncertainty calibration

Table 4.3 illustrates that generic baseline methods (*i.e.* VGG16, C3D, I3D) pre-trained on well diverse and dense datasets further away from the target domain, benefit in terms of generalization and uncertainty calibration as they are on par with the methods specifically designed to tackle the problem of pedestrian crossing prediction, which was not the case in a simple train-test evaluation setting[10].

To better isolate the effects of pre-training with larger datasets we consider two I3D [Carreira and Zisserman, 2017] but trained with different configuration: the first one being randomly initialized and the second one being pre-trained on Sports1M [Karpathy et al., 2014]. We assess their performance on the same datasets and report our findings in Fig 4.6. We show that pre-trained models significantly outperform randomly initialized models across all three datasets in terms of calibration. As far as robustness aspects towards small domain shifts are concerned, this may become an important factor to consider when designing pedestrian crossing behavior approaches for real-world scenarios. The training source being generally not dense in variety of conditions nor in the number of examples, the results provided on each dataset might just come from noise on testing sets. Pre-training well-established models on diverse and dense datasets further away from the target domain before fine-tunning to our target task might prove efficient and mandatory for the next step of pedestrian crossing behavior prediction: generalization and vehicle implementation.

[10]We refer the reader to section B for a detailed list of calibrations for each of the models.

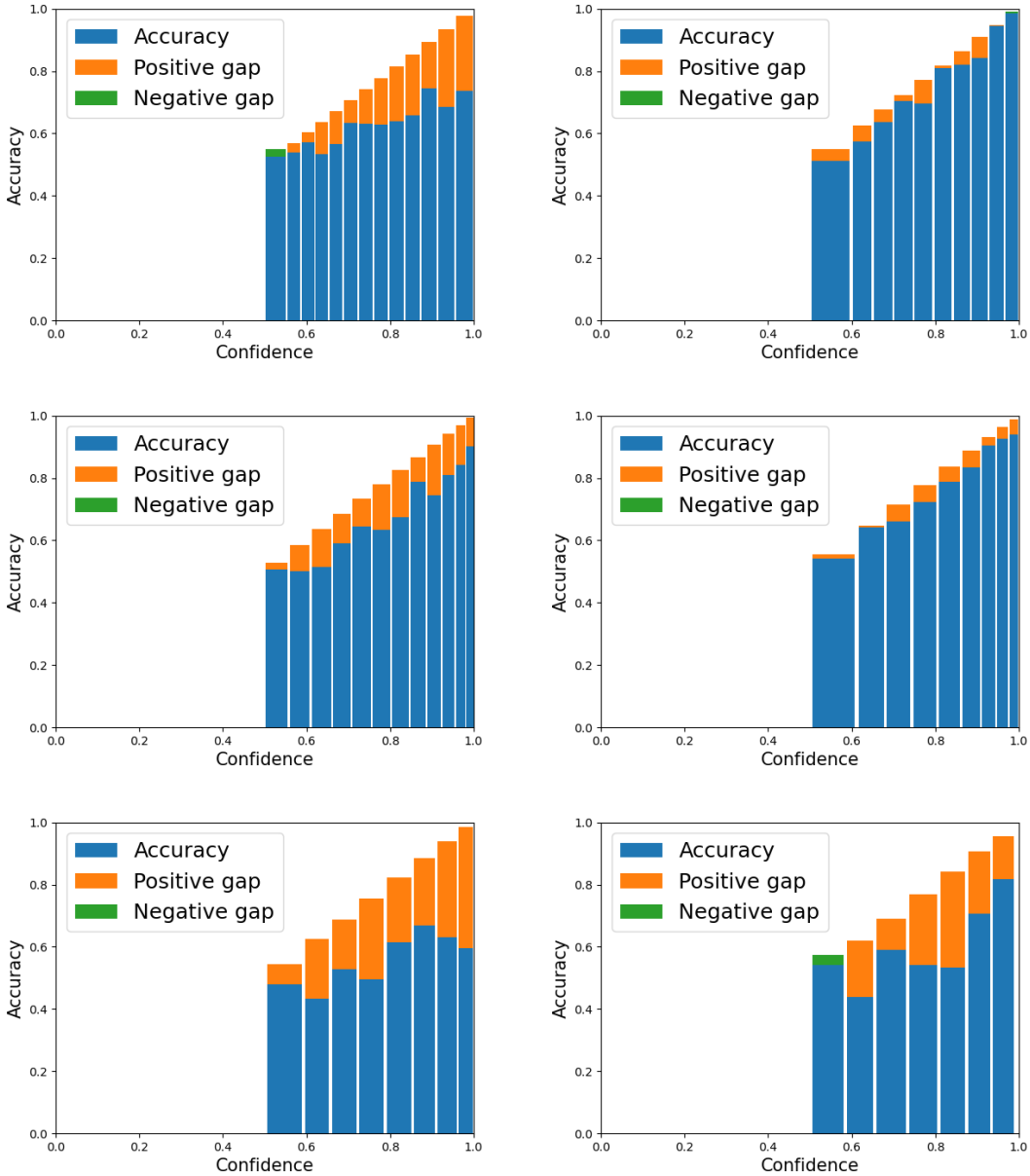


Figure 4.6: Reliability Diagrams between I3D [Carreira and Zisserman, 2017] randomly initialized (left) and pre-trained on Sports1M [Karpathy et al., 2014](right) on PIE, $JAAD_{all}$ and $JAAD_{behavior}$ datasets. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green).

4.6 Improving Uncertainty Calibration

For the very same approach, there is a significant discrepancy between traditional train-test and cross-dataset evaluation results. This calls into question the reliability of current methods in regard to their capacity to generalize. In addition, we have shown that the standard classification metrics are not sufficient to reliably evaluate an approach since the use of uncertainty metrics raises additional issues that are not reflected otherwise. We are confident that the future breakthroughs in the area will not occur by

	AUC (↑)			F1 (↑)			ECE (↓)			MCE (↓)		
Method	**pie**	**beh**	**all**	**pie**	**beh**	**all**	**pie**	**beh**	**all**	**pie**	**beh**	**all**
Non-pretrained	0.55 (±0.04)	0.50 (±0.02)	0.69 (± 0.05)	0.34 (±0.09)	0.65 (±0.08)	0.54 (±0.06)	0.205 (±0.062)	0.184 (± 0.089)	0.111 (± 0.069)	0.290 (±0.039)	0.338 (±0.191)	0.162 (±0.075)
Ens-Nonpretrained	0.59 (±0.06)	0.58 (±0.05)	0.64 (±0.04)	0.37 (±0.10)	0.55 (±0.10)	0.39 (±0.05)	0.065 (±0.022)	0.131 (±0.058)	0.091 (±0.026)	0.248 (±0.122)	0.644 (±0.152)	0.280 (±0.172)
Deterministic	0.72 (±0.01)	0.56 (±0.03)	0.76 (±0.03)	0.60 (±0.02)	0.74 (±0.02)	0.61 (±0.02)	0.026 (±0.007)	0.143 (± 0.020)	0.054 (±0.010)	0.063 (±0.007)	0.239 (± 0.030)	0.118 (±0.020)
Ens-Deterministic	0.64 (±0.01)	0.62 (±0.01)	0.73 (± 0.1)	0.49 (±0.02)	0.70 (±0.01)	0.51 (±0.01)	0.053 (±0.003)	0.080 (±0.001)	0.097 (±0.016)	0.120 (±0.013)	0.138 (±0.024)	0.172 (±0.022)
MC Dropout	0.73 (±0.01)	0.55 (±0.01)	**0.78** (±0.01)	0.61 (±0.01)	0.67 (±0.01)	0.60 (± 0.01)	0.064 (± 0.003)	**0.063** (±0.005)	0.040 (±0.002)	0.106 (±0.004)	**0.134** (±0.012)	**0.059** (±0.009)
Ens-MC Dropout	0.61 (±0.01)	0.61 (±0.01)	0.73 (±0.01)	0.42 (±0.01)	0.49 (±0.02)	0.53 (±0.01)	0.053 (±0.002)	0.053 (±0.003)	0.129 (±0.002)	0.096 (±0.005)	0.120 (±0.013)	0.181 (±0.007)
TempScaling	0.70 (±0.02)	**0.58** (±0.02)	0.76 (±0.01)	0.57 (±0.03)	0.72 (±0.01)	0.61 (± 0.02)	**0.020** (±0.005)	0.070 (± 0.020)	0.037 (±0.08)	**0.050** (±0.010)	0.300 (±0.130)	0.146 (±0.035)
Ens-TempScaling	0.61 (± 0.01)	**0.66** (±0.01)	**0.75** (±0.01)	0.41 (±0.02)	0.69 (±0.01)	**0.56** (±0.01)	0.058 (±0.004)	0.054 (±0.008)	0.142 (±0.015)	0.127 (±0.012)	0.237 (±0.068)	0.215 (±0.016)
LL Dropout	0.71 (±0.01)	0.54 (±0.003)	**0.78** (±0.001)	0.59 (±0.01)	0.74 (± 0.001)	**0.62** (±0.003)	**0.020** (± 0.003)	0.155 (±0.003)	0.061 (± 0.001)	0.063 (±0.012)	0.247 (±0.01)	0.107 (±0.016)
Ens-LL Dropout	0.65 (± 0.002)	0.62 (±0.003)	0.73 (±0.002)	0.49 (±0.003)	0.70 (±0.001)	0.50 (±0.003)	0.052 (±0.003)	0.081 (±0.002)	0.098 (±0.002)	**0.105** (±0.002)	0.145 (±0.023)	0.176 (±0.005)
LL SVI	**0.74** (±0.01)	0.53 (±0.01)	0.77 (± 0.003)	**0.62** (±0.01)	**0.76** (±0.01)	0.57 (±0.004)	**0.021** (±0.002)	0.162 (±0.004)	**0.026** (±0.003)	0.059 (±0.009)	0.214 (±0.006)	**0.054** (±0.009)
Ens-LL SVI	**0.68** (±0.003)	0.61 (±0.003)	0.69 (±0.003)	**0.55** (±0.003)	**0.73** (±0.002)	0.43 (±0.004)	**0.045** (±0.003)	**0.036** (±0.003)	**0.046** (±0.005)	0.146 (±0.027)	**0.075** (±0.009)	**0.133** (±0.016)

Table 4.4: Average Pedestrian Crossing Prediction performance for PIE, $JAAD_{behavior}$ and $JAAD_{all}$ (5 runs). Dashed lines separate each probabilistic deep learning baseline. Each baseline is tested twice: first, in a classical train-test evaluation protocol and then tested by ensembling all three models trained on each training set to evaluate its robustness to small domain shift. We highlight the highest scores for each metric and for both evaluation protocols: train-test or ensembling.

outperforming current state-of-the-art by a small margin on conventionally used evaluation protocols as they currently fail to provide the big picture of pedestrian crossing behavior prediction.

As we encourage the community to change the direction in which we are taking the research field, we investigate how additional baselines from the probabilistic deep learning literature improve the generalization ability of pedestrian behavior predictors towards small domain shifts. We believe that those methods could prove useful for the next generation of predictors and present our results with the intention that they will serve as a baseline for future work addressing our prescriptions.

4.6.1 Baselines from the probabilistic deep learning literature

Below, we present the selected methods from the probabilistic deep learning literature applied on top of an I3D [Carreira and Zisserman, 2017] model:

- **Non-pretrained and Deterministic**: Maximum softmax probability [Hendrycks and Gimpel, 2016] of N networks trained independently on each dataset using either random initialization or pre-trained weights from Sports1M [Karpathy et al., 2014]. (We set $N = 5$ for each method below.)
- **Monte-Carlo Dropout** (*MC Dropout*): Dropout activated at test time as an approximate bayesian inference in deep Gaussian processes [Gal and Ghahramani, 2016].
- **Temperature Scaling**[11] (*TempScaling*): Post-hoc calibration of softmax probability by temperature scaling using a validation set [Guo et al., 2017].
- **Last Layer Dropout** (*LL Dropout*): Bayesian inference for the parameters of the last layer only: Dropout activated at test time on the activations before the last layer.
- **Last Layer Stochastic Variational Bayesian Inference** (*LL SVI*): Mean field stochastic variational inference on the last layer using Flipout [Wen et al., 2018].
- **Ensembling** (*Ens*): Average prediction of three networks trained independently on each training set using pre-trained weights [Lakshminarayanan et al., 2017]. Similarly to Table 4.3, we use ensembling as a plausible approximation of one model's robustness for real-world scenarios.

4.6.2 Discussion

We present the results obtained by probabilistic methods for both evaluation protocols: train-test and ensembling in Table 4.4. This allows us to report the effect of dataset shift on accuracy and calibration for the probabilistic deep learning methods. Naturally, we would like to obtain a model, that is well-calibrated on the training and testing distributions of each dataset and remains calibrated with ensembling. We observe that, similarly to the deterministic methods, the quality of predictions consistently degrades with dataset shift regardless of the selected probabilistic method for both PIE and $JAAD_{all}$. However, overall robustness degrades more significantly for some methods. For instance, TempScaling, *e.g.* post-hoc calibration of softmax probability, seems to be one of the best train-test probabilistic methods in regards to expected calibration error (ECE) when evaluated in a standard train-test procedure but falls behind when evaluated under dataset shift. In fact, when evaluated under dataset shift,

[11] https://github.com/gpleiss/temperature_scaling

all the methods except Non-pretrained ones outperform TempScaling in regards to ECE. Similarly, we report that better calibration and accuracy on each test set does not correlate[12] with better calibration under ensembling: the average ECE of the methods when evaluated with classical train-test scenario is [0.166, 0.074, 0.056, 0.042, 0.079, 0.070] and the average ECE of the same methods under dataset shift are [0.096, 0.077, 0.078, 0.085, 0.077, 0.042]. Interestingly, most of the selected probabilistic methods perform better on average than the deterministic I3D under train-test evaluation protocols but fail to generalize when exposed to dataset shift. The exception to the rule is LL SVI, which looks very promising in terms of generalization to small domain shifts. As our experiments required pre-trained weights from I3D, we could not replace each convolutional layer with mean-field variational Flipout layers, we only changed the last layer of the given model to obtain a variational bayesian inference for a quick baseline. Nevertheless, we believe that this could be a future research to consider. We should explore the effects of transferring initially learned features on large bases further away from the target task and explore how probabilistic methods react to transfer-learning and domain-shift.

4.7 Summary

In this chapter, we show that the classical train-test sets evaluation for pedestrian crossing prediction, *i.e.*, models being trained and tested on the same dataset, is not sufficient to efficiently compare nor conclude anything about their applicability in a real-world scenario: the benchmarks being either too small or too loose in variety of scenarios, it is easy for a given model to over-fit on a specific target dataset. In order to evaluate the generalization capacity of the approaches, we conduct a study based on direct cross-dataset evaluation for eleven methods representing the diversity of architectures and modalities used for pedestrian crossing prediction. We found a huge lack of generalization and robustness for all selected approaches. This led us to a ranking of existing approaches that is much more complex and less absolute than the standard one. We secondly discuss the importance of quantifying a model's uncertainty. Although this is currently completely disregarded, it is common sense to use it in our field of application. We discover two interesting properties: pre-training well-established models on diverse and dense datasets further away from the target domain before fine-tuning to our target task improves calibration and, two models with equivalent classification scores do not necessarily have equivalent calibration scores. This may prove interesting to consider when comparing their usefulness in real-world scenarios with inputs distribution frequently shifted from the training distribution. Finally, we enforce the importance of evaluating the robustness of pedestrian crossing behavior models by evaluating how trustworthy are their uncertainty estimates under domain shifts with cross-dataset evaluation. We encourage the community to consider those new protocols and metrics in order to reach the end-goal of pedestrian crossing behavior predictors: vehicle implementation.

[12] Pearson's Correlation coefficient: 0.4203, p-value: 0.4067.

Chapter 5

Conclusion

D'abord, la science n'est pas : elle se fait. Le savant du jour n'est que l'ignorant du lendemain.

Elisée Reclus

Contents

5.1 Summary 108

5.2 Future Works 110

In this thesis, we explored deep learning approaches as a way to efficiently leverage spatial and temporal components of pedestrian poses kinematics to efficiently detect their intention of crossing in urban traffic environment.

5.1 Summary

This thesis aimed to answer the three research questions presented in section 1.2. In the following, we detail what we have learned and try to provide response elements for each one of them.

Question 1 *Inductive biases are the set of assumptions a learner uses to predict results given inputs it has not yet encountered. When training deep learning architectures with little available data, should we only rely on the very composition of layers to impose relational inductive biases on the learner? Does enforcing certain constraints towards the data representation of designated hidden layers, sending informative-representation ready data to the classification network help the performance of deep learning networks for action classification?*

No, we should not only rely on the very composition of layers to impose relational inductive biases on the learner when faced with problems with little available data. Many modern deep learning methods follow an "end-to-end" design philosophy that emphasizes minimal a priori representational and computational assumptions, which explains why they tend to be so data-intensive. However, we have seen in chapter 2 that since anything that imposes constraints on the learning trajectory is considered as an inductive bias, and given all the possible combinations that can only be evaluated empirically, no trend is easily distinguishable in terms of identifying the best architecture to model sequence for skeletal action recognition. Deep learning being a science that is constantly confronted to the risk of confirmation bias, we questioned the importance of representations, inductive biases and their roles in skeletal action recognition. Firstly, we evaluated the importance of explicit temporal modeling for gesture recognition. We proposed a fully-connected autoencoder, that does not benefit from any relational inductive bias and enforces the mapping from inputs to outputs in the embedding via statistical regularizations. We showed that the proposed approach reaches the performances of classic sequence-focused architectures on action classification tasks with little available data. Secondly, we investigated the importance of sending informative-representation ready data to a deep learning architecture in a 1D-2D grid space. Neural networks are designed to extract temporal features from gestures, and then merge them hierarchically depending on their sequence-focused design in order to perform the final classification. Intermediate representations of the gestures are entirely learned by the model and its corresponding inductive biases, without any manual intervention. However, since model representations are based on the input data representation, finding an appropriate input representation is crucial to leverage the full potential of the network. By transforming the input data based on physical world constraints of the body structure prior to the learning of multiple layers of feature hierarchies that automatically build high-level representations of the raw input, we showed that finding an appropriate input representation is crucial to leverage the full potential of a deep learning network for action recognition.

Question 2 *Visual skeletal representations are known to be sufficient for both humans and machines to describe and recognize biological motion, including human motion. Can pose kinematics be sufficient to serve as the only input when modeling non-trivial and non-periodic tasks related to pedestrian intention prediction?*

Nuanced yes. Many approaches have been proposed that report interesting results on pedestrian crossing prediction. However, most of them may suffer from a large model size and slow inference speed by aggregating multiple forms of perception modalities extracted by additional networks such as background context, optical flow, or pose estimation information. In this manuscript, we specifically focused on the simple yet informative skeleton modality, as it has been proven to be sufficient to describe and understand the motion of a given action without any background context. We first list the pedestrian and environmental factors involved in pedestrian decision-making process in accordance with the perceptive modality selected. By only considering the kinematics of a pedestrian's pose, we only capture certain factors impacting the decision to cross, mainly, attention of the pedestrian towards its environment, walking pattern and certain forms of communication such as eye contact with the driver. We proposed SPI-Net and TrouSPI-Net: two scene-agnostic, lightweight, multi-branch approaches that rely on pose kinematics to predict crossing behaviors. Then, we showed that it is possible to make the link between the posture, the walking attitude and the future behaviours of the protagonists of a scene without using the contextual information of the scene (pedestrian crossing, traffic light...). Still, for a crossing prediction algorithm to be as efficient as the approaches using multiple perception modalities, it is necessary to include additional information to the pose kinematics itself. For instance, spatial positioning of the pedestrian based on 2D bounding box locations can then be used to infer his trajectory and velocity, ego-vehicle speed allows the model to incorporate an additional form of non-gestural communication between the pedestrian and the driver. These two inputs are not strictly speaking visual perception modalities as they can be directly derived from a pose estimation algorithm or retrieved directly by the vehicle data. Nevertheless, this is what leads us to be cautious when answering our second research question.

Question 3 *Does recent progress on pedestrian intention prediction benchmarks continue to represent meaningful generalization? What evaluation protocol and metrics should be used to go beyond accuracy in order to evaluate a model for a high-risk application with a limited amount of training data?*

No, we showed that the classical train-test sets evaluation protocol for pedestrian crossing prediction, *i.e.*, models being trained and tested on the same dataset, is not sufficient to efficiently compare nor conclude anything about their applicability in a real-world scenario: the benchmarks being either too small or too loose in variety of scenarios, it is easy for a given model to over-fit on a specific target dataset. The classical performance metrics for classification being no longer sufficient to compare the existing methods with the new evaluation paradigm, we looked at a complementary category of metrics to compare pedestrian intention prediction models and discussed the importance of quantifying a model's uncertainty. Although uncertainty is currently completely disregarded in the current state of the benchmark, it is common sense to use it in our field of application. In order to build the foundation on which future work should be based on, and, in addition to the eleven deterministic baselines evaluated under domain shift to demonstrate that the current evaluation protocol will also reach its limits, we report the results

of multiple baselines from the probabilistic deep learning literature, designed to tackle the problem of improving model uncertainty. Given all of the above, we advise the community to change the direction in which we are taking the research field: with so little existing data, non-existent generalization of models, and inconclusive ranking of them, we need to agree to properly evaluate our approaches in order to minimize the noise of our productions and thus, make the research field more sustainable and representative of the real advances to come.

5.2 Future Works

Our work can be extended in several directions. We suggest a few prospective ideas that possibly might be relevant for future research:

Action Recognition with little available data

- **The development of a sample-efficient "AI"** for small-data problems that arise in many domains related to human motion. From preserving the knowledge for future generations of expert gestures in niche fields such as glassblowing, blacksmithing or stonemasonry, to tracking the posture of an athlete for a specific sport such as fencing, boxing or gymnastics, we will need to efficiently represent those expert gestures. We are mostly moving towards this industrial transfer learning paradigm in which big foundation models that emphasize minimal a priori representational assumptions are fine-tuned on downstream tasks. Academia should persist with orthogonal goals and create new ways of representing gestures. This may involve, manually integrating domain knowledge into the network, prior to the application of the network or inside the cost function of the network.

Pedestrian Discrete Intention Prediction

- **Temporal tracking of pedestrians:** In the real world, there are usually more pedestrians on the streets passing and occluding each other, which requires sophisticated mechanisms not only for their detection but for their temporal tracking without mixing their identity over time. The literature completely omits such issues and relies on the ground truth spatial coordinates and individual IDs of each pedestrian provided by each dataset. To address a better follow-up of the protagonists in the scene and to avoid mixing the dynamics of two protagonists due to a change of camera angle, future research should focus on building an end-to-end framework based on unlabeled coordinates of pedestrians, temporal tracking of pedestrians and any pose-based model for pedestrian intention prediction. New research questions will then arise, it might be necessary to quantify the robustness to tracking errors for each new contribution. That new robustness to tracking error metric may even be the missing piece to having a more representative ranking of methods.

- **Improving pose estimation methods:** necessary step of an intention prediction model of which the analysis of the posture is an essential component. One major drawback of our work is to rely on off-the-shelf pose estimation algorithms without trying to improve the existing ones to make them fit the application field. However, similarly to the OSI[1] model, our approaches rely on inde-

[1] *Open Systems Interconnection*, enables interoperability between different products and software.

pendent implementations of methods for each specific task. It leads to a practical methodology: interchanging the pose estimation algorithms does not compromise the entire approach. Currently, one of the main limitations of a 2D pose estimation is the ability to deal with pedestrian occlusions in a two-dimensional space. Therefore, in order to improve the pose detection, the question of adding a third dimension may arise. Currently, the methods for estimating 3D poses are much less mature than those for 2D pose estimation. One of the main reasons, to this day, has been the lack of reliable data sets available. However, our pipeline makes it easy to keep up with the state-of-the-art in this field without completely disrupting the approach for intention prediction. If major advances were made in the field of pose estimation, our approach might still be relevant. A concrete example would be the release of cheap RBG-D cameras that can be easily deployed on the ego-vehicle: the depth modality would then make it easier to estimate 3D poses and thus potentially improve the robustness of our approaches without any architectural modifications.

The future of pedestrian crossing prediction benchmarks

- **We need to properly deal with the fact that the world is not completely predictable:** There is no such thing as truly random but sometimes, we simply lack the information to make a sound judgement: if you know the entire wave function of the universe in a cubic kilometer surrounding the tosser and have a very powerful computer, coin flipping is almost completely deterministic. If we combine all the available perception modalities, if we take into account the age, sex and religion or any other human factor perspectives influencing pedestrians' behaviors, will the prediction be any less noisy? In this thesis, we came up with the idea of using uncertainty calibration to show that sometimes, a judgement is not as enlightened as it seems, despite looking accurate at first glance. If anyone was to continue my research, I would advise him/her not to put too much stress on beating the current state-of-the-art with the current evaluation protocols. I would rather advise him/her to answer these two questions which I believe are essential for the future of the research field: "*Given two models, one more accurate and the other better calibrated, which should a practitioner choose*?" and *"Is there a way to ensure good system performance at integration-time?"*

Appendix A

Appendix

A Assessing TrouSPI-Net performance for skeletal action recognition datasets

Table A.1: Results obtained via TrouSPI-Net on SHREC [De Smedt et al., 2017]. We change the model size by modifying the filters parameter for each convolution block.

Methods	Parameters	Accuracy on *SHREC* 14	Accuracy on *SHREC* 28
[Nunez et al., 2018](CNN+LSTM)	8-9 M	89.8%	86.3%
[Devineau et al., 2018] (Skelnet)	13.83 M	91.3%	84.4%
[Hou et al., 2018](STA-Res-TCN)	5-6 M	93.6%	90.7%
[Yang et al., 2019] (DD-Net)	1.82 M	94.6%	91.9%
[Min et al., 2020](PointLSTM)	1.2 M	95.9%	**94.7%**
[Avola et al., 2018] (LM controller)	-	**97.6%**	91.4%
TrouSPI-net (filters=64)	2.2 M	96.3%	93.8%
TrouSPI-net (filters=32)	0.57 M	95.9%	92.6%
TrouSPI-net (filters=16)	0.15 M	95.6%	91.2%
TrouSPI-net (filters=8)	**0.04 M**	95.3%	90.4%

Table A.2: Results obtained via TrouSPI-Net on JHMDB [Jhuang et al., 2013]. We change the model size by modifying the filters parameter for each convolution block.

Methods	Parameters	Accuracy on 3 splits of *JHMDB*
[Zolfaghari et al., 2017](Chained Net)	17.50 M	56.8%
[Ludl et al., 2019] (EHPI)	1.22 M	65.5%
[Choutas et al., 2018] (Potion)	4.87 M	67.9%
[Yang et al., 2019] (DD-Net (filters=64))	1.82 M	**77.2%**
[Yang et al., 2019] (DD-Net (filters=16))	0.15 M	65.7%
TrouSPI-net (filters=64)	2.2 M	**74.5%**
TrouSPI-net (filters=32)	0.56 M	72.4%
TrouSPI-net (filters=16)	0.14 M	71.8%
TrouSPI-net (filters=8)	**0.04 M**	72.2%

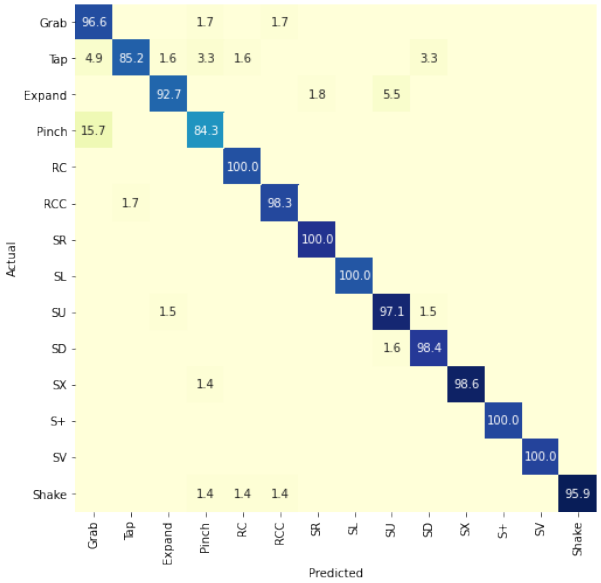


Figure A.1: Confusion matrix obtained on SHREC 14 with TrouSPI-Net

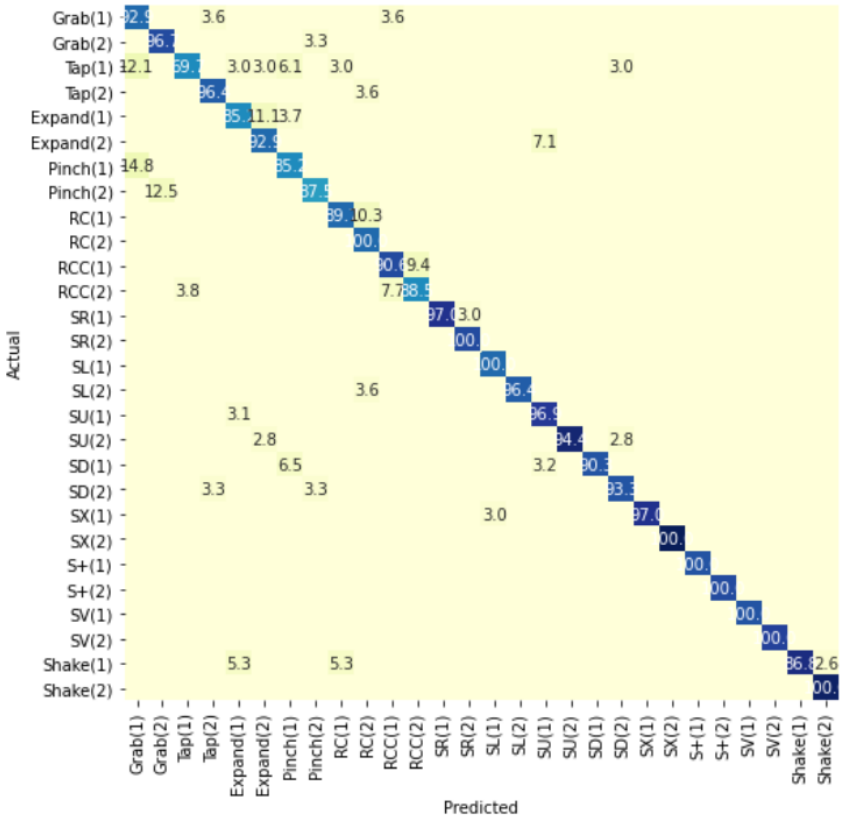


Figure A.2: Confusion matrix obtained on SHREC 28 with TrouSPI-Net

B Additional reliability diagrams for eleven baselines on three dataset for pedestrian discrete intention prediction

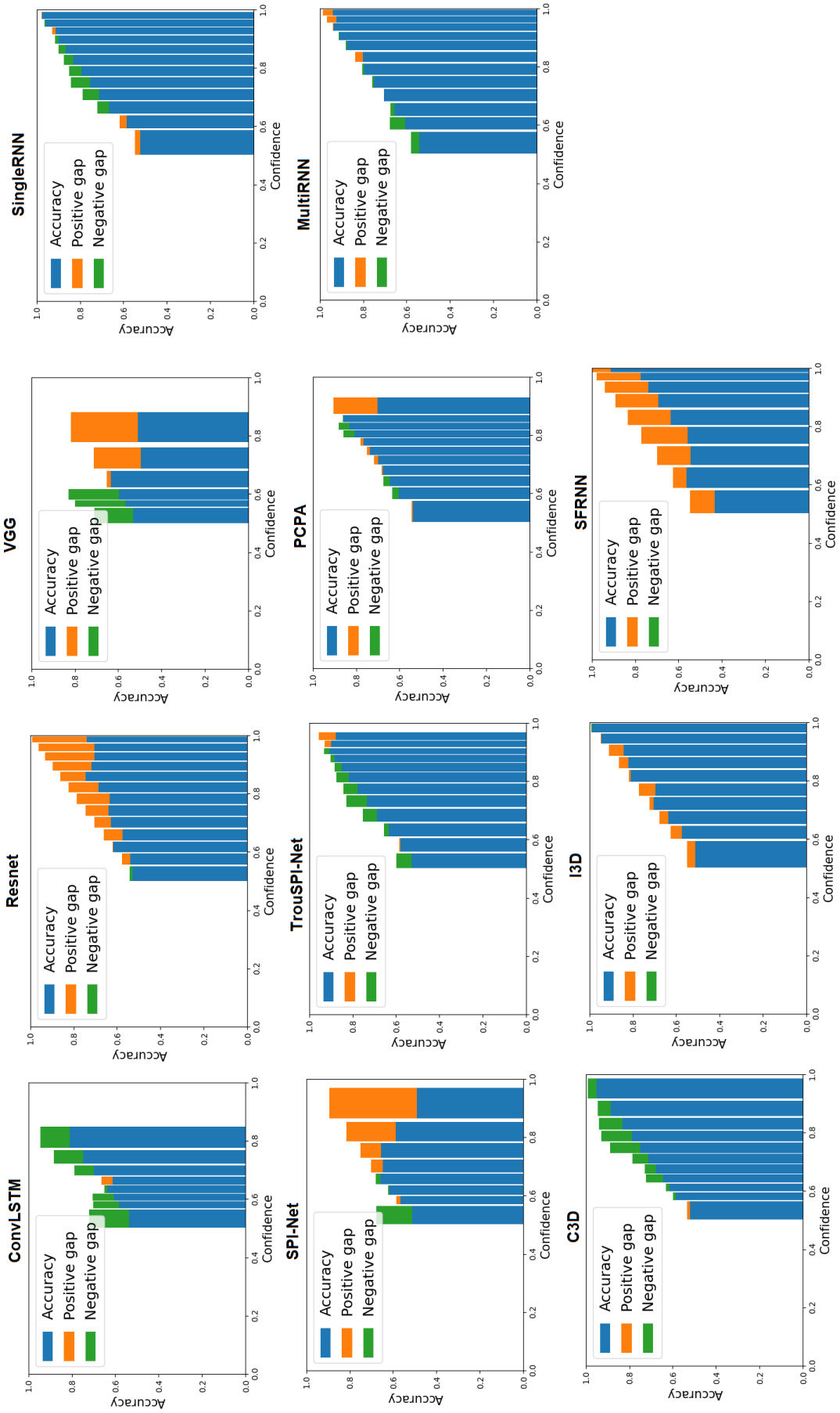


Figure A.3: Reliability Diagrams for the eleven selected methods on *PIE* data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green).

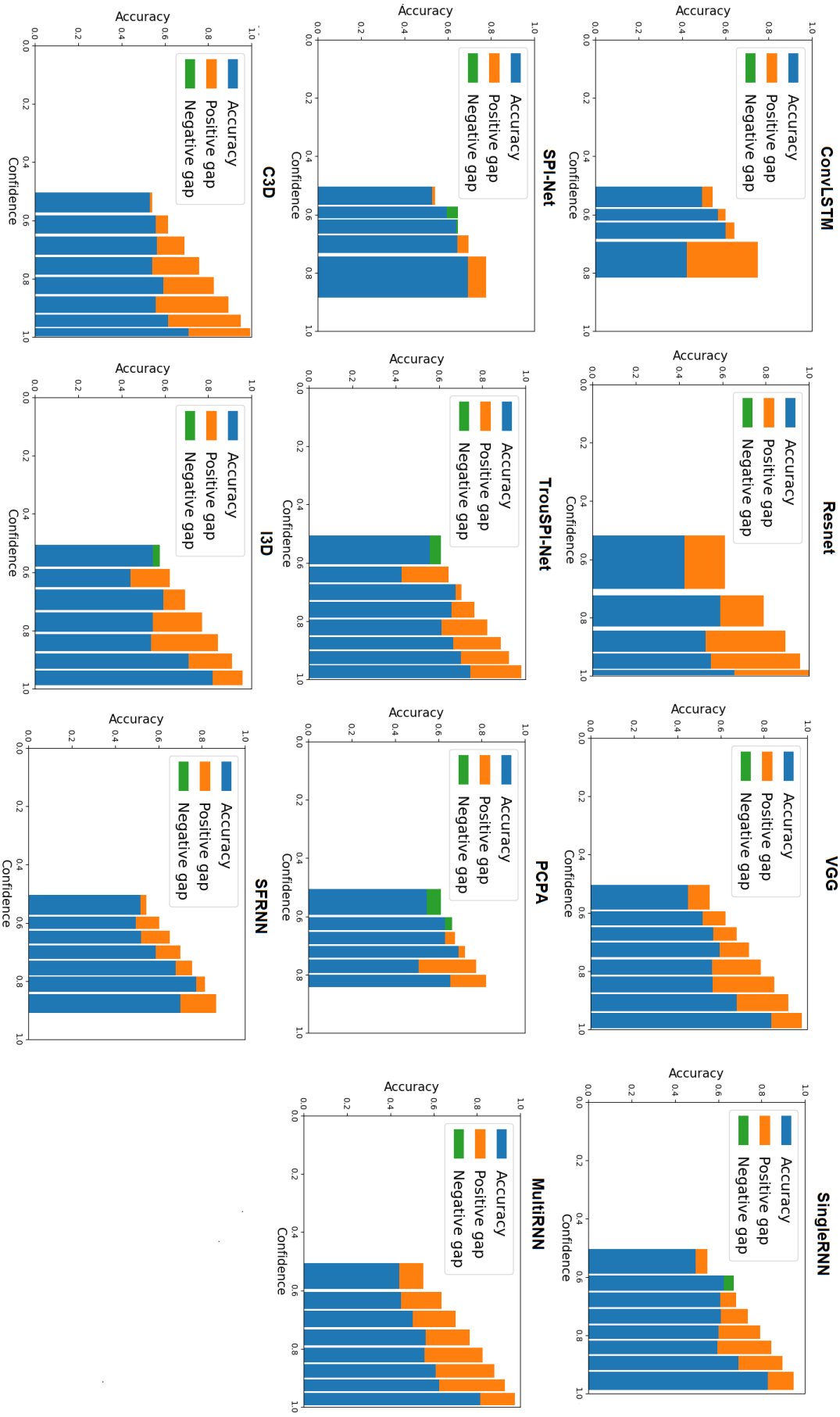


Figure A.4: Reliability Diagrams for the eleven selected methods on $JAAD_{behavior}$ data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green).

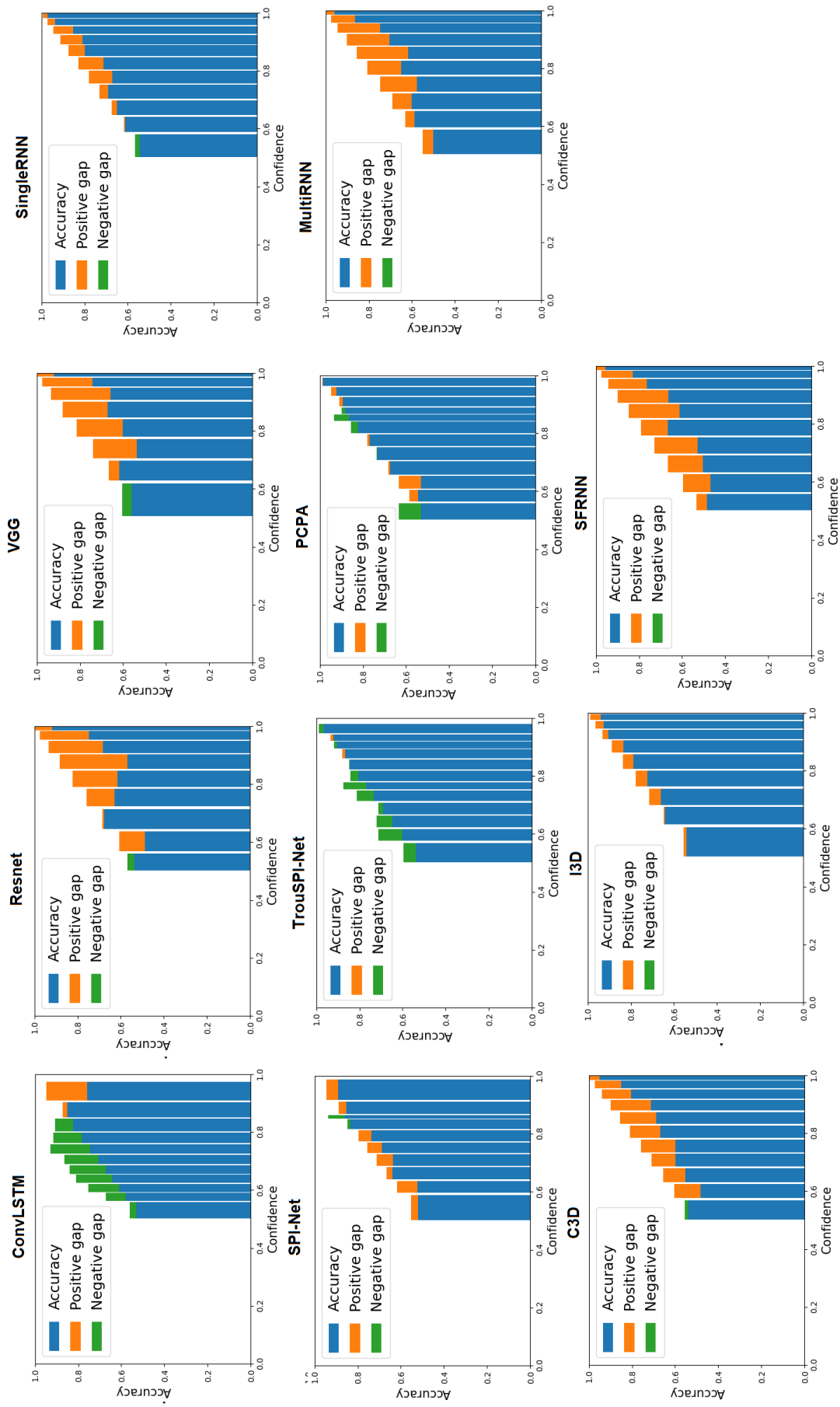


Figure A.5: Reliability Diagrams for the eleven selected methods on $JAAD_{all}$ data set. If the model is perfectly calibrated, then the diagram plots the identity function. Any deviation from a perfect diagonal represents miscalibration: the model is either overconfident (orange) or subconfident (green).

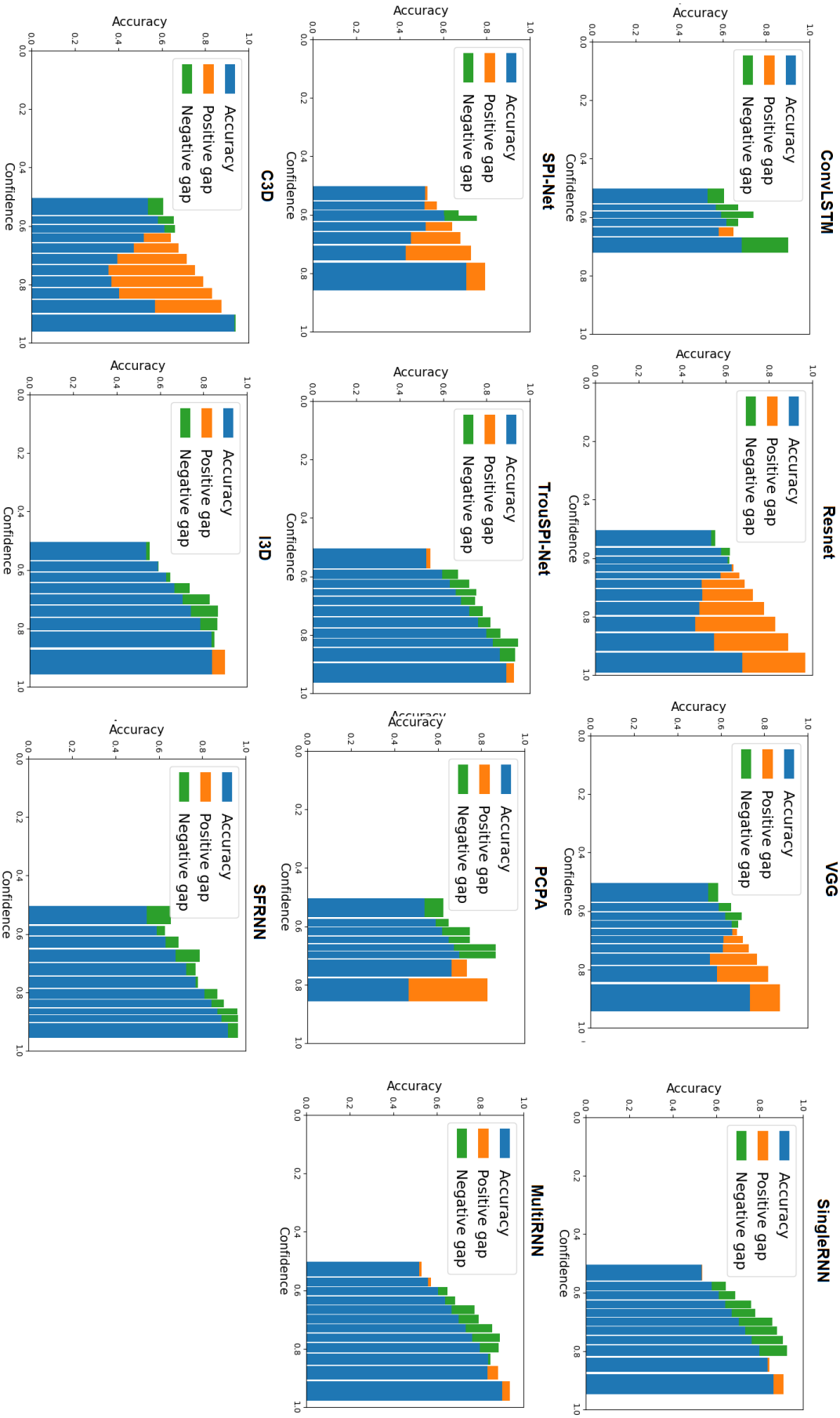


Figure A.6: Reliability Diagram of the Average prediction given by three individual models and their respective outputs for PIE (Ensembling), Each individual model is either trained on $PIE, JAAD_{behavior}$ or $JAAD_{all}$.

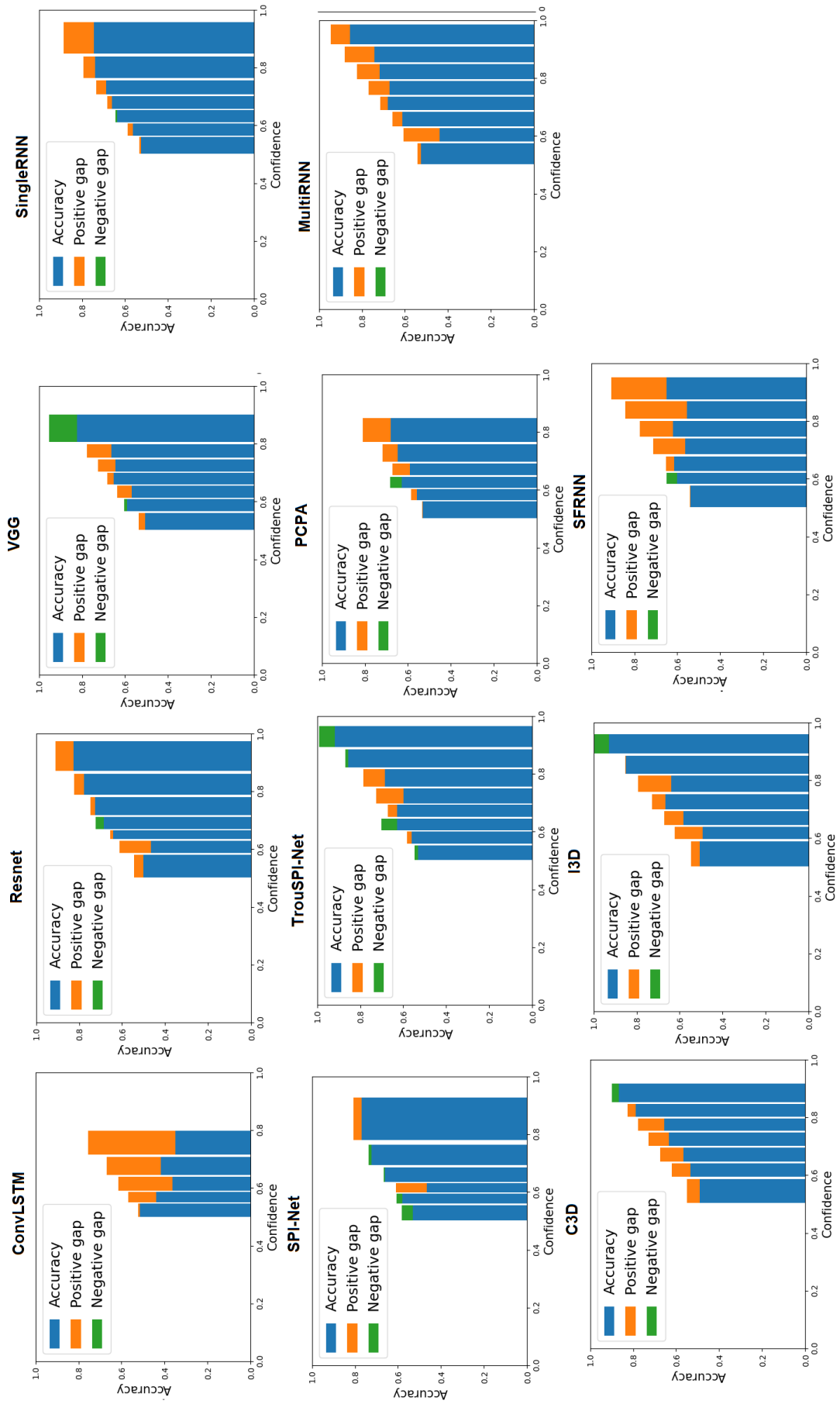


Figure A.7: Reliability Diagram of the Average prediction given by three individual models and their respective outputs for $JAAD_{behavior}$ (Ensembling), Each individual model is either trained on $PIE, JAAD_{behavior}$ or $JAAD_{all}$.

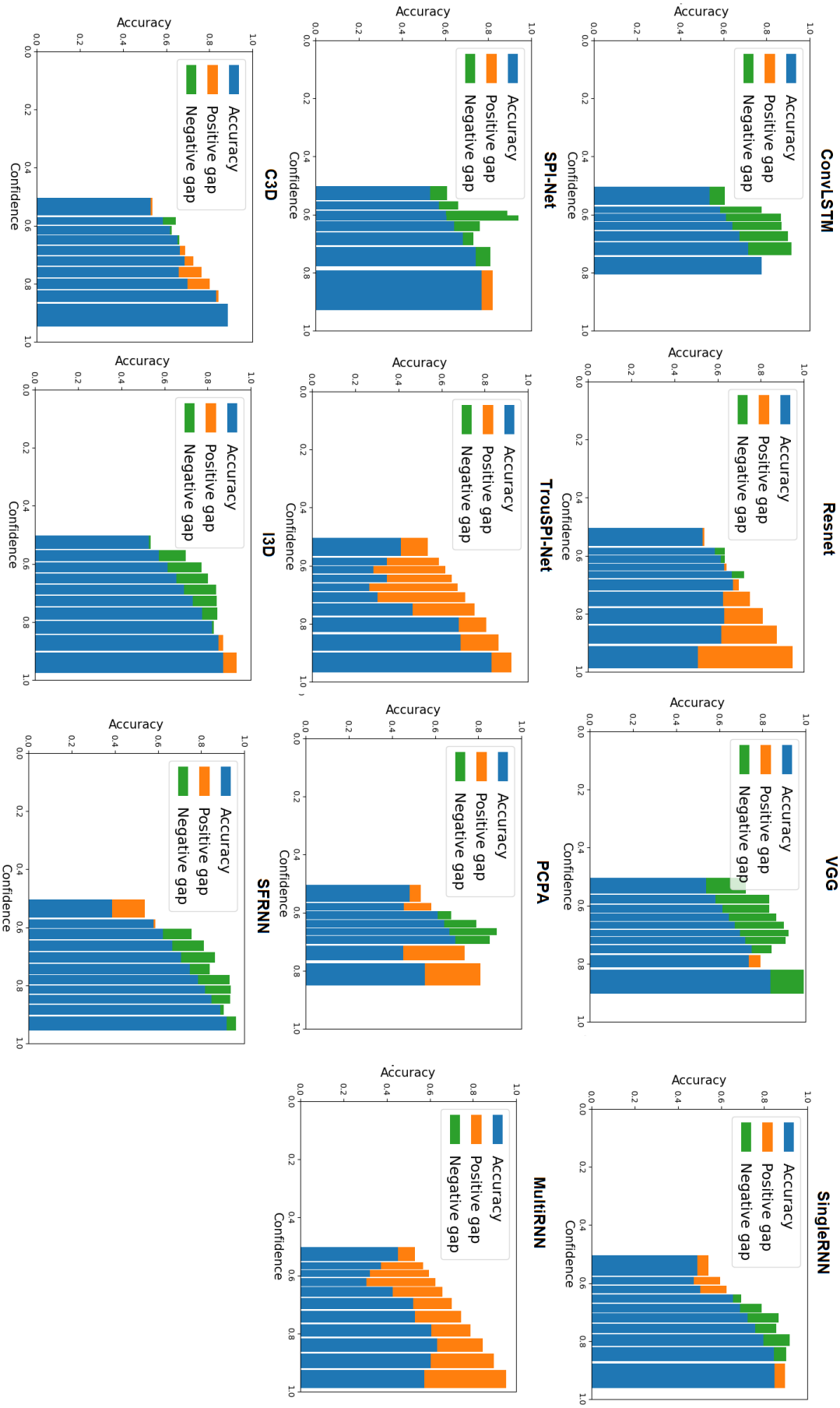


Figure A.8: Reliability Diagram of the Average prediction given by three individual models and their respective outputs for $JAAD_{all}$ (Ensembling), Each individual model is either trained on PIE,$JAAD_{behavior}$ or $JAAD_{all}$.

Appendix B

Résumé en français

Introduction

Ce chapitre constitue l'introduction de la thèse. Nous décrivons brièvement le contexte de cette thèse, les problèmes qu'elle aborde et les principales contributions de cette thèse.

Reconnaissance de l'activité humaine avec modèles d'apprentissage profond basés sur la cinématique de la posture humaine

Dans ce chapitre, nous présentons quelques repères historiques pour la compréhension des actions humaines ainsi qu'un aperçu des modalités actuelles de vision par ordinateur pour la reconnaissance d'actions. Nous décrivons ensuite les différentes familles d'apprentissage profond pour la modélisation de séquences squelettiques ainsi que leurs biais inductifs respectifs. Les approches existantes se répartissent en quatre grandes catégories: les réseaux neuronaux récurrents, les réseaux neuronaux convolutifs, les réseaux neuronaux à mémoire associative basés sur l'attention et les réseaux de type graphes neuronaux. Par la suite, nous nous interrogeons sur l'importance des représentations, des biais inductifs et de leurs rôles pour la reconnaissance d'actions squelettiques. Tout d'abord, nous évaluons l'importance d'une modélisation temporelle explicite pour la reconnaissance de gestes : alors que les gestes sont des phénomènes temporels, de nombreux gestes et actions peuvent en réalité être déduits sur la base de poses spatiales uniquement. Nous proposons un auto-encodeur, qui ne bénéficie d'aucun biais inductif et qui renforce la correspondance entre les entrées et les sorties dans l'espace latent via des régularisations statistiques. Nous montrons que l'approche proposée atteint les performances des architectures classiques de modélisation de séquences sur des tâches de classification d'actions avec peu de données disponibles. Deuxièmement, nous étudions l'importance d'envoyer des données porteuses d'information à une architecture d'apprentissage profond, et cela, avant l'apprentissage automatique de caractéristiques de haut niveau de l'entrée brute. En normalisant les données d'entrée sur la base des contraintes du monde physique comme la structure du corps humain, nous montrons que pour les tâches de classification d'actions avec peu de données, les réseaux de neurones bénéficient de ces caractéristiques fabriquées à la main et pourraient s'appuyer sur moins de couches cachées pour apprendre des représentations informatives des données.

De la reconnaissance de l'activité humaine à la prédiction discrete d'intention des piétons

Dans ce chapitre, nous proposons d'abord un aperçu des approches existantes pour la prédiction d'action des piétons dans un contexte urbain. La majorité des techniques existantes de prédiction d'action des piétons sont basées sur la trajectoire, ce qui signifie qu'elles dépendent des positions des piétons observées précédemment afin d'anticiper les positions des mêmes piétons dans le futur. Ces méthodes sont efficaces seulement lorsque les piétons ont déjà traversé ou sont sur le point de traverser, c'est-à-dire que ces algorithmes réagissent à une action qui a déjà commencé plutôt que de la prédire. Nous proposons d'abord une architecture de réseau neuronal récurrent bidirectionnel asymétrique appelée U-RNN pour encoder les trajectoires des piétons et nous évaluons sa pertinence pour remplacer les LSTM pour une variété d'approches et de modules d'interaction différents. Nous montrons alors qu'il y a encore de la marge d'amélioration pour les approches basées sur les coordonnées et concluons que les interactions ne sont pas le seul aspect sur lequel la prédiction de la trajectoire des piétons peut progresser. Nous abordons ensuite le problème de la prédiction des intentions discrètes des piétons : au lieu de se concentrer sur les trajectoires continues décrivant le mouvement futur attendu du piéton et de se fier uniquement à la dynamique d'une scène pour prédire les intentions des protagonistes, nous définissons les intentions d'un piéton comme une combinaison de ses comportements discrets de haut niveau tels que la dynamique de sa pose, l'orientation de sa tête, etc. Nous montrons alors qu'il est possible de faire le lien entre la posture, l'attitude de marche et les comportements futurs des protagonistes d'une scène sans utiliser les informations contextuelles de celle-ci (passage piéton, feu de circulation...). Cela nous permet alors de diviser par un facteur 20 la vitesse d'inférence des approches existantes pour la prédiction de l'intention des piétons tout en gardant la même robustesse de prédiction.

Évaluation de la capacité de généralisation des algorithmes de prédiction discrete d'intention des piétons

Ce dernier chapitre est délibérément plus exploratoire. La prédiction d'intention des piétons a fait l'objet de recherches actives, ce qui a donné lieu à de nombreuses nouvelles solutions algorithmiques. Bien que la mesure de la progression globale de ces solutions dans le temps tende à être de plus en plus établie grâce aux nouveaux jeux de données accessibles au public et aux procédures d'évaluation standardisées, savoir à quel point les prédicteurs existants réagissent aux données non rencontrées reste une question sans réponse. Cette évaluation est impérative, car nos algorithmes doivent pouvoir fonctionner dans divers scénarios sans compromettre la sécurité des piétons en raison d'une mauvaise prédiction. À cette fin, nous menons une étude basée sur l'évaluation croisée d'ensembles de données. Nos expériences montrent que les approches actuellement considérées comme à la pointe pour la prédiction d'intention des piétons généralisent mal lorsqu'elles sont évaluées lors de scénarios d'évaluation croisée, et ce, indépendamment de leur robustesse dans un cadre d'évaluation dit classique avec un ensemble d'apprentissage et de test. À la lumière de ce que nous observons, nous soutenons que l'avenir de notre domaine de recherche, c'est à dire des implémentations fiables et généralisables, ne devrait pas consister à adapter des modèles, entraînés avec très peu de données disponibles, et testés dans un scénario classique d'évaluation avec la volonté de déduire quoi que ce soit sur leur comportement dans la vie réelle. Il s'agirait plutôt d'évaluer les modèles à venir dans un contexte d'évaluation croisée tout en tenant compte

des estimations d'incertitude de ces modèles pour des cas peu connus.

Conclusion

Nous résumons cette thèse et identifions les orientations futures potentielles de notre recherche.

Appendix C

Bibliography

[Achaji et al., 2021] Achaji, L., Moreau, J., Fouqueray, T., Aioun, F., and Charpillet, F. (2021). Is attention to bounding boxes all you need for pedestrian action prediction? *arXiv preprint arXiv:2107.08031*.

[Ahmet, 2020] Ahmet, E. (2020). 6th place solution: Very custom gru. `https://www.kaggle.com/c/riiid-test-answer-prediction/discussion/209581`.

[Alahi et al., 2016] Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., and Savarese, S. (2016). Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971. ISSN: 1063-6919.

[Alahi et al., 2014] Alahi, A., Ramanathan, V., and Fei-Fei, L. (2014). Socially-aware large-scale crowd forecasting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2203–2210.

[Avola et al., 2018] Avola, D., Bernardi, M., Cinque, L., Foresti, G. L., and Massaroni, C. (2018). Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures. *IEEE Transactions on Multimedia*, 21(1):234–245.

[Babu, 2019] Babu, S. C. (2019). A 2019 guide to human pose estimation with deep learning. `https://nanonets.com/blog/human-pose-estimation-2d-guide/`.

[Baccouche et al., 2011] Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., and Baskurt, A. (2011). Sequential deep learning for human action recognition. In *International workshop on human behavior understanding*, pages 29–39. Springer.

[Bai et al., 2018] Bai, S., Kolter, J. Z., and Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

[Bakwin, 1945] Bakwin, H. (1945). Pseudodoxia pediatrica. *New England journal of medicine*, 232(24):691–697.

[Balogh et al., 2015] Balogh, E., Miller, B., and Ball, J. (2015). Committee on diagnostic error in health care; board on health care services; institute of medicine. *The National Academy of Sciences, Engineering and Medicine, Improving Diagnosis in Health Care, The National Academic Press, Washington DC*.

[Banerjee et al., 2020] Banerjee, A., Singh, P. K., and Sarkar, R. (2020). Fuzzy integral-based cnn classifier fusion for 3d skeleton action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(6):2206–2216.

[Baradel et al., 2018] Baradel, F., Wolf, C., and Mille, J. (2018). Human Activity Recognition with Pose-driven Attention to RGB. In *BMVC 2018 - 29th British Machine Vision Conference*, pages 1–14, Newcastle, United Kingdom.

[Barnachon, 2013] Barnachon, M. (2013). *Reconnaissance d'actions en temps réel à partir d'exemples*. Theses, Université Claude Bernard - Lyon I.

[Bartoli et al., 2018] Bartoli, F., Lisanti, G., Ballan, L., and Del Bimbo, A. (2018). Context-aware trajectory prediction. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 1941–1946. IEEE.

[Battaglia et al., 2018] Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.

[Becker et al., 2019] Becker, S., Hug, R., Hübner, W., and Arens, M. (2019). RED: A Simple but Effective Baseline Predictor for the TrajNet Benchmark. In Leal-Taixé, L. and Roth, S., editors, *Computer Vision – ECCV 2018 Workshops*, volume 11131, pages 138–153. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

[Benzine et al., 2020] Benzine, A., Chabot, F., Luvison, B., Pham, Q. C., and Achard, C. (2020). Pandanet: Anchor-based single-shot multi-person 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Benzine et al., 2019] Benzine, A., Luvison, B., Pham, Q. C., and Achard, C. (2019). Deep, robust and single shot 3d multi-person human pose estimation from monocular images. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 584–588. IEEE.

[Beyer et al., 2020] Beyer, L., Hénaff, O. J., Kolesnikov, A., Zhai, X., and Oord, A. v. d. (2020). Are we done with imagenet? *arXiv preprint arXiv:2006.07159*.

[BeyondMinds, 2020] BeyondMinds, T. (2020). An overview of human pose estimation with deep learning. `https://beyondminds.ai/blog/an-overview-of-human-pose-estimation-with-deep-learning/`.

[Bhat et al., 2021] Bhat, S. F., Alhashim, I., and Wonka, P. (2021). Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4009–4018.

[Bhattacharyya et al., 2018] Bhattacharyya, A., Fritz, M., and Schiele, B. (2018). Long-term on-board prediction of people in traffic scenes under uncertainty. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4194–4202.

[Björklund and Åberg, 2005] Björklund, G. M. and Åberg, L. (2005). Driver behaviour in intersections: Formal and informal traffic rules. *Transportation Research Part F: Traffic Psychology and Behaviour*, 8(3):239–253.

[Boulahia et al., 2017] Boulahia, S. Y., Anquetil, E., Multon, F., and Kulpa, R. (2017). Dynamic hand gesture recognition based on 3d pattern assembled trajectories. In *2017 seventh international conference on image processing theory, tools and applications (IPTA)*, pages 1–6. IEEE.

[Brigato and Iocchi, 2021] Brigato, L. and Iocchi, L. (2021). A close look at deep learning with small data. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2490–2497. IEEE.

[Bronstein et al., 2017] Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. (2017). Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42.

[Bujalance Martin and Moutarde, 2019] Bujalance Martin, J. and Moutarde, F. (2019). Real-time gestural control of robot manipulator through deep learning human-pose inference. In *International Conference on Computer Vision Systems*, pages 565–572. Springer.

[Cadena et al., 2022] Cadena, P. R. G., Qian, Y., Wang, C., and Yang, M. (2022). Pedestrian graph+: A fast pedestrian crossing prediction model based on graph convolutional networks. *IEEE Transactions on Intelligent Transportation Systems*.

[Cadena et al., 2019] Cadena, P. R. G., Yang, M., Qian, Y., and Wang, C. (2019). Pedestrian graph: Pedestrian crossing prediction based on 2d pose estimation and graph convolutional networks. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 2000–2005.

[Caird and Hancock, 1994] Caird, J. K. and Hancock, P. A. (1994). The perception of arrival time for different oncoming vehicles at an intersection. *Ecological Psychology*, 6(2):83–109.

[Cao et al., 2018] Cao, C., Lan, C., Zhang, Y., Zeng, W., Lu, H., and Zhang, Y. (2018). Skeleton-based action recognition with gated convolutional neural networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3247–3257.

[Cao et al., 2017] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7291–7299.

[Caputo et al., 2018] Caputo, F. M., Prebianca, P., Carcangiu, A., Spano, L. D., and Giachetti, A. (2018). Comparing 3d trajectories for simple mid-air gesture recognition. *Computers & Graphics*, 73:17–25.

[Carreira and Zisserman, 2017] Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733.

[Chaabane et al., 2020] Chaabane, M., Trabelsi, A., Blanchard, N., and Beveridge, R. (2020). Looking ahead: Anticipating pedestrians crossing with future frames prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2297–2306.

[Chang et al., 2011] Chang, Y.-J., Chen, S.-F., and Huang, J.-D. (2011). A kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. *Research in developmental disabilities*, 32(6):2566–2570.

[Chavdarova et al., 2018] Chavdarova, T., Baqué, P., Bouquet, S., Maksai, A., Jose, C., Bagautdinov, T., Lettry, L., Fua, P., Van Gool, L., and Fleuret, F. (2018). Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5030–5039.

[Chen and Ramanan, 2016] Chen, C.-H. and Ramanan, D. (2016). 3D Human Pose Estimation = 2D Pose Estimation + Matching. *arXiv e-prints*, page arXiv:1612.06524.

[Chen and Ye, 2019] Chen, H. and Ye, W. (2019). Classification of human activity based on radar signal using 1-d convolutional neural network. *IEEE Geoscience and Remote Sensing Letters*, 17(7):1178–1182.

[Chen et al., 2019] Chen, X., Wang, G., Guo, H., Zhang, C., Wang, H., and Zhang, L. (2019). Mfa-net: Motion feature augmented network for dynamic hand gesture recognition from skeletal data. *Sensors*, 19(2):239.

[Chen et al., 2020] Chen, Y., Liu, P., Zhong, M., Dou, Z.-Y., Wang, D., Qiu, X., and Huang, X. (2020). Cdevalsumm: An empirical study of cross-dataset evaluation for neural summarization systems. *arXiv preprint arXiv:2010.05139*.

[Chen et al., 2017] Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., and Sun, J. (2017). Cascaded Pyramid Network for Multi-Person Pose Estimation. *arXiv e-prints*, page arXiv:1711.07319.

[Cheng et al., 2021] Cheng, H., Liao, W., Yang, M. Y., Rosenhahn, B., and Sester, M. (2021). Amenet: Attentive maps encoder network for trajectory prediction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 172:253–266.

[Cheng et al., 2020] Cheng, K., Zhang, Y., He, X., Chen, W., Cheng, J., and Lu, H. (2020). Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 183–192.

[Cho et al., 2014] Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

[Cho et al., 2014] Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv e-prints*, page arXiv:1406.1078.

[Choi and Dariush, 2019] Choi, C. and Dariush, B. (2019). Looking to relations for future trajectory forecast. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 921–930.

[Choutas et al., 2018] Choutas, V., Weinzaepfel, P., Revaud, J., and Schmid, C. (2018). Potion: Pose motion representation for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7024–7033.

[Chung et al., 2014] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

[Clay, 1995] Clay, D. (1995). Driver attitude and attribution: implications for accident prevention.

[Das et al., 2005] Das, S., Manski, C. F., and Manuszak, M. D. (2005). Walk or wait? an empirical analysis of street crossing decisions. *Journal of applied econometrics*, 20(4):529–548.

[De Smedt et al., 2016] De Smedt, Q., Wannous, H., and Vandeborre, J.-P. (2016). Skeleton-based dynamic hand gesture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–9.

[De Smedt et al., 2017] De Smedt, Q., Wannous, H., Vandeborre, J.-P., Guerry, J., Le Saux, B., and Filliat, D. (2017). Shrec'17 track: 3d hand gesture recognition using a depth and skeletal dataset.

[Dehghani et al., 2018] Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., and Kaiser, Ł. (2018). Universal transformers. *arXiv preprint arXiv:1807.03819*.

[Demšar, 2006] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.

[Devanne et al., 2014] Devanne, M., Wannous, H., Berretti, S., Pala, P., Daoudi, M., and Del Bimbo, A. (2014). 3-d human action recognition by shape analysis of motion trajectories on riemannian manifold. *IEEE transactions on cybernetics*, 45(7):1340–1352.

[Devineau et al., 2018] Devineau, G., Moutarde, F., Xi, W., and Yang, J. (2018). Deep learning for hand gesture recognition on skeletal data. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 106–113. IEEE.

[Devlin et al., 2018] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[Dey and Terken, 2017] Dey, D. and Terken, J. (2017). Pedestrian interaction with vehicles: roles of explicit and implicit communication. In *Proceedings of the 9th international conference on automotive user interfaces and interactive vehicular applications*, pages 109–113.

[Diba et al., 2017a] Diba, A., Fayyaz, M., Sharma, V., Karami, A. H., Arzani, M. M., Yousefzadeh, R., and Van Gool, L. (2017a). Temporal 3d convnets: New architecture and transfer learning for video classification. *arXiv preprint arXiv:1711.08200*.

[Diba et al., 2017b] Diba, A., Sharma, V., and Van Gool, L. (2017b). Deep temporal linear encoding networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2329–2338.

[Ding et al., 2020] Ding, Y., Liu, J., Xiong, J., and Shi, Y. (2020). Revisiting the evaluation of uncertainty estimation and its application to explore model complexity-uncertainty trade-off. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 4–5.

[DiPietro and King, 1970] DiPietro, C. M. and King, L. E. (1970). Pedestrian gap-acceptance. *Highway Research Record*, (308).

[Donahue et al., 2015] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634.

[Du et al., 2015a] Du, Y., Fu, Y., and Wang, L. (2015a). Skeleton based action recognition with convolutional neural network. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 579–583. IEEE.

[Du et al., 2015b] Du, Y., Wang, W., and Wang, L. (2015b). Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118.

[Duan et al., 2020] Duan, H., Zhao, Y., Xiong, Y., Liu, W., and Lin, D. (2020). Omni-sourced webly-supervised learning for video recognition. In *European Conference on Computer Vision*, pages 670–688. Springer.

[Dwibedi et al., 2018] Dwibedi, D., Sermanet, P., and Tompson, J. (2018). Temporal reasoning in videos using convolutional gated recurrent units. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1111–1116.

[Elias et al., 2015] Elias, P., Sedmidubsky, J., and Zezula, P. (2015). Motion images: An effective representation of motion capture data for similarity search. In Amato, G., Connor, R., Falchi, F., and Gennaro, C., editors, *Similarity Search and Applications*, pages 250–255, Cham. Springer International Publishing.

[Fan et al., 2021] Fan, H., Yang, Y., and Kankanhalli, M. (2021). Point 4d transformer networks for spatio-temporal modeling in point cloud videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14204–14213.

[Fan et al., 2022] Fan, H., Yu, X., Ding, Y., Yang, Y., and Kankanhalli, M. (2022). Pstnet: Point spatio-temporal convolution on point cloud sequences. *arXiv preprint arXiv:2205.13713*.

[Fan et al., 2019] Fan, Z., Zhao, X., Lin, T., and Su, H. (2019). Attention-based multiview re-observation fusion network for skeletal action recognition. *IEEE Transactions on Multimedia*, 21:363–374.

[Fang et al., 2016] Fang, H.-S., Xie, S., Tai, Y.-W., and Lu, C. (2016). RMPE: Regional Multi-person Pose Estimation. *arXiv e-prints*, page arXiv:1612.00137.

[Fang and López, 2018] Fang, Z. and López, A. M. (2018). Is the pedestrian going to cross? answering by 2d pose estimation. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1271–1276. IEEE.

[Fang and López, 2020] Fang, Z. and López, A. M. (2020). Intention recognition of pedestrians and cyclists by 2d pose estimation. *IEEE Transactions on Intelligent Transportation Systems*, 21(11):4773–4783.

[Färber, 2016] Färber, B. (2016). Communication and communication problems between autonomous vehicles and human drivers. In *Autonomous driving*, pages 125–144. Springer.

[Favarò et al., 2017] Favarò, F. M., Nader, N., Eurich, S. O., Tripp, M., and Varadaraju, N. (2017). Examining accident reports involving autonomous vehicles in california. *PLoS one*, 12(9):e0184952.

[Fei-Fei and Krishna, 2022] Fei-Fei, L. and Krishna, R. (2022). Searching for computer vision north stars. *Daedalus*, 151(2):85–99.

[Fernando et al., 2019] Fernando, T., Denman, S., Sridharan, S., and Fookes, C. (2019). Neighbourhood context embeddings in deep inverse reinforcement learning for predicting pedestrian motion over long time horizons. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1179–1187.

[Flohr et al., 2015] Flohr, F., Dumitru-Guzu, M., Kooij, J. F., and Gavrila, D. M. (2015). A probabilistic framework for joint pedestrian head and body orientation estimation. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1872–1882.

[Gal and Ghahramani, 2016] Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

[Gantier et al., 2019] Gantier, R., YANG, M., Qian, Y., and Wang, C. (2019). Pedestrian graph: Pedestrian crossing prediction based on 2d pose estimation and graph convolutional networks. pages 2000–2005.

[Gao et al., 2019] Gao, M., Jiang, J., Zou, G., John, V., and Liu, Z. (2019). Rgb-d-based object recognition using multimodal convolutional neural networks: A survey. *IEEE access*, 7:43110–43136.

[Gao et al., 2020] Gao, R., Oh, T.-H., Grauman, K., and Torresani, L. (2020). Listen to look: Action recognition by previewing audio. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10457–10467.

[Geruschat et al., 2003] Geruschat, D. R., Hassan, S. E., and Turano, K. A. (2003). Gaze behavior while crossing complex intersections. *Optometry and vision science*, 80(7):515–528.

[Gesnouin et al., 2020] Gesnouin, J., Pechberti, S., Bresson, G., Stanciulescu, B., and Moutarde, F. (2020). Predicting intentions of pedestrians from 2d skeletal pose sequences with a representation-focused multi-branch deep learning network. *Algorithms*, 13(12):331.

[Gesnouin et al., 2021] Gesnouin, J., Pechberti, S., Stanciulcscu, B., and Moutarde, F. (2021). Trouspinet: Spatio-temporal attention on parallel atrous convolutions and u-grus for skeletal pedestrian crossing prediction. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pages 01–07. IEEE.

[Ghadiyaram et al., 2019] Ghadiyaram, D., Tran, D., and Mahajan, D. (2019). Large-scale weakly-supervised pre-training for video action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12046–12055.

[Ghori et al., 2018] Ghori, O., Mackowiak, R., Bautista, M., Beuter, N., Drumond, L., Diego, F., and Ommer, B. (2018). Learning to forecast pedestrian intention from pose dynamics. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1277–1284.

[Giuliari et al., 2021] Giuliari, F., Hasan, I., Cristani, M., and Galasso, F. (2021). Transformer networks for trajectory forecasting. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10335–10342. IEEE.

[Goldhammer et al., 2014] Goldhammer, M., Hubert, A., Koehler, S., Zindler, K., Brunsmann, U., Doll, K., and Sick, B. (2014). Analysis on termination of pedestrians' gait at urban intersections. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1758–1763. IEEE.

[Gori et al., 2005] Gori, M., Monfardini, G., and Scarselli, F. (2005). A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE.

[Guéguen et al., 2015] Guéguen, N., Meineri, S., and Eyssartier, C. (2015). A pedestrian's stare and drivers' stopping behavior: A field experiment at the pedestrian crossing. *Safety science*, 75:87–89.

[Guerry et al., 2017] Guerry, J., Le Saux, B., and Filliat, D. (2017). " Look At This One " Detection sharing between modality-independent classifiers for robotic discovery of people. In *ECMR 2017 - European Conference on Mobile Robotics*, pages 1–6, Paris, France.

[Gujjar and Vaughan, 2019] Gujjar, P. and Vaughan, R. (2019). Classifying pedestrian actions in advance using predicted video of urban driving scenes. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2097–2103.

[Guo et al., 2017] Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR.

[Guo et al., 2020] Guo, P., Xue, Z., Long, L. R., and Antani, S. (2020). Cross-dataset evaluation of deep learning networks for uterine cervix segmentation. *Diagnostics*, 10(1):44.

[Gupta et al., 2018] Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., and Alahi, A. (2018). Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2255–2264. ISSN: 2575-7075.

[Hara et al., 2017] Hara, K., Kataoka, H., and Satoh, Y. (2017). Learning spatio-temporal features with 3d residual networks for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 3154–3160.

[Harrell, 1991] Harrell, W. A. (1991). Factors influencing pedestrian cautiousness in crossing streets. *The Journal of Social Psychology*, 131(3):367–372.

[Hasan et al., 2021] Hasan, I., Liao, S., Li, J., Akram, S. U., and Shao, L. (2021). Generalizable pedestrian detection: The elephant in the room. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11328–11337.

[He et al., 2021] He, J.-Y., Wu, X., Cheng, Z.-Q., Yuan, Z., and Jiang, Y.-G. (2021). Db-lstm: Densely-connected bi-directional lstm for human action recognition. *Neurocomputing*, 444:319–331.

[He et al., 2017] He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask R-CNN. *arXiv e-prints*, page arXiv:1703.06870.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *arXiv e-prints*, page arXiv:1502.01852.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Heimstra et al., 1969] Heimstra, N. W., Nichols, J., and Martin, G. (1969). An experimental methodology for analysis of child pedestrian behavior. *Pediatrics*, 44(5):832–838.

[Hendrycks and Gimpel, 2016] Hendrycks, D. and Gimpel, K. (2016). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

[Heo et al., 2018] Heo, J., Lee, H. B., Kim, S., Lee, J., Kim, K. J., Yang, E., and Hwang, S. J. (2018). Uncertainty-aware attention for reliable interpretation and prediction. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 917–926.

[Hermann et al., 2020] Hermann, K., Chen, T., and Kornblith, S. (2020). The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:19000–19015.

[Hinton, 2007] Hinton, G. E. (2007). To recognize shapes, first learn to generate images. *Progress in brain research*, 165:535–547.

[Hinton and Salakhutdinov, 2006] Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Holland and Hill, 2007] Holland, C. and Hill, R. (2007). The effect of age, gender and driver status on pedestrians' intentions to cross the road in risky situations. *Accident Analysis & Prevention*, 39(2):224–237.

[Hong et al., 2019] Hong, J., Sapp, B., and Philbin, J. (2019). Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8454–8462.

[Hornik, 1991] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257.

[Hou et al., 2018] Hou, J., Wang, G., Chen, X., Xue, J.-H., Zhu, R., and Yang, H. (2018). Spatial-temporal attention res-tcn for skeleton-based dynamic hand gesture recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0.

[Hu et al., 2018] Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141.

[Huang et al., 2017] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

[Imran and Kumar, 2016] Imran, J. and Kumar, P. (2016). Human action recognition using rgb-d sensor and deep convolutional neural networks. In *2016 international conference on advances in computing, communications and informatics (ICACCI)*, pages 144–148. IEEE.

[Imran and Raman, 2019] Imran, J. and Raman, B. (2019). Deep residual infrared action recognition by integrating local and global spatio-temporal cues. *Infrared Physics & Technology*, 102:103014.

[Insafutdinov et al., 2016] Insafutdinov, E., Andriluka, M., Pishchulin, L., Tang, S., Levinkov, E., Andres, B., and Schiele, B. (2016). Articulated multi-person tracking in the wild. *CoRR*, abs/1612.01465.

[Insafutdinov et al., 2016] Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., and Schiele, B. (2016). DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model. *arXiv e-prints*, page arXiv:1605.03170.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv e-prints*, page arXiv:1502.03167.

[Iqbal and Gall, 2016] Iqbal, U. and Gall, J. (2016). Multi-Person Pose Estimation with Local Joint-to-Person Associations. *arXiv e-prints*, page arXiv:1608.08526.

[Ishaque and Noland, 2008] Ishaque, M. M. and Noland, R. B. (2008). Behavioural issues in pedestrian speed choice and street crossing behaviour: a review. *Transport Reviews*, 28(1):61–85.

[Jhuang et al., 2013] Jhuang, H., Gall, J., Zuffi, S., Schmid, C., and Black, M. J. (2013). Towards understanding action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3192–3199.

[Ji et al., 2012] Ji, S., Xu, W., Yang, M., and Yu, K. (2012). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231.

[Johansson, 1973] Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2):201–211.

[Johansson, 1976] Johansson, G. (1976). Spatio-temporal differentiation and integration in visual motion perception. *Psychological research*, 38(4):379–393.

[Kahneman and Tversky, 2013] Kahneman, D. and Tversky, A. (2013). Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific.

[Kalman, 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.

[Karpathy et al., 2014] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

[Kawashima et al., 2017] Kawashima, T., Kawanishi, Y., Ide, I., Murase, H., Deguchi, D., Aizawa, T., and Kawade, M. (2017). Action recognition from extremely low-resolution thermal image sequence. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE.

[Ke et al., 2017] Ke, Q., Bennamoun, M., An, S., Sohel, F., and Boussaid, F. (2017). A new representation of skeleton sequences for 3d action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3288–3297.

[Keskar et al., 2019] Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., and Socher, R. (2019). Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv e-prints*, page arXiv:1412.6980.

[Klette and Tee, 2008] Klette, R. and Tee, G. (2008). Understanding human motion: A historic review. *http://citr.auckland.ac.nz/techreports/2007/CITR-TR-192.pdf*, 36.

[Köhler et al., 2012] Köhler, S., Goldhammer, M., Bauer, S., Doll, K., Brunsmann, U., and Dietmayer, K. (2012). Early detection of the pedestrian's intention to cross the street. In *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pages 1759–1764. IEEE.

[Köhler et al., 2015] Köhler, S., Goldhammer, M., Zindler, K., Doll, K., and Dietmeyer, K. (2015). Stereo-vision-based pedestrian's intention detection in a moving vehicle. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 2317–2322. IEEE.

[Köhler et al., 2013] Köhler, S., Schreiner, B., Ronalter, S., Doll, K., Brunsmann, U., and Zindler, K. (2013). Autonomous evasive maneuvers triggered by infrastructure-based detection of pedestrian intentions. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 519–526. IEEE.

[Kopuklu et al., 2019] Kopuklu, O., Kose, N., Gunduz, A., and Rigoll, G. (2019). Resource efficient 3d convolutional neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*.

[Kothari et al., 2021] Kothari, P., Kreiss, S., and Alahi, A. (2021). Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–15.

[Kotseruba et al., 2020] Kotseruba, I., Rasouli, A., and Tsotsos, J. K. (2020). Do they want to cross? understanding pedestrian intention for behavior prediction. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1688–1693.

[Kotseruba et al., 2021] Kotseruba, I., Rasouli, A., and Tsotsos, J. K. (2021). Benchmark for evaluating pedestrian action prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1258–1268.

[Kreiss et al., 2019] Kreiss, S., Bertoni, L., and Alahi, A. (2019). Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11977–11986.

[Kreiss et al., 2021] Kreiss, S., Bertoni, L., and Alahi, A. (2021). Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

[Lakshminarayanan et al., 2017] Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30.

[Lam et al., 1995] Lam, W. H., Morrall, J. F., and Ho, H. (1995). Pedestrian flow characteristics in hong kong. *Transportation research record*, 1487:56–62.

[Leal-Taixé et al., 2014] Leal-Taixé, L., Fenzi, M., Kuznetsova, A., Rosenhahn, B., and Savarese, S. (2014). Learning an image-based motion context for multiple people tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3542–3549.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[Li et al., 2019a] Li, B., Li, X., Zhang, Z., and Wu, F. (2019a). Spatio-temporal graph routing for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8561–8568.

[Li et al., 2018] Li, C., Cui, Z., Zheng, W., Xu, C., and Yang, J. (2018). Spatio-Temporal Graph Convolution for Skeleton Based Action Recognition. *arXiv e-prints*, page arXiv:1802.09834.

[Li et al., 2017a] Li, C., Hou, Y., Wang, P., and Li, W. (2017a). Joint distance maps based action recognition with convolutional neural networks. *IEEE Signal Processing Letters*, 24(5):624–628.

[Li et al., 2017b] Li, C., Wang, P., Wang, S., Hou, Y., and Li, W. (2017b). Skeleton-based action recognition using lstm and cnn. In *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 585–590. IEEE.

[Li et al., 2019b] Li, C., Zhang, X., Liao, L., Jin, L., and Yang, W. (2019b). Skeleton-based gesture recognition using several fully connected layers with path signature features and temporal transformer module. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8585–8593.

[Li and Chan, 2014] Li, S. and Chan, A. B. (2014). 3d human pose estimation from monocular images with deep convolutional neural network. In *Asian Conference on Computer Vision*, pages 332–347. Springer.

[Li et al., 2018] Li, S., Li, W., Cook, C., Zhu, C., and Gao, Y. (2018). Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5457–5466.

[Li et al., 2019c] Li, T., Fan, L., Zhao, M., Liu, Y., and Katabi, D. (2019c). Making the invisible visible: Action recognition through walls and occlusions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 872–881.

[Lindgren et al., 2008] Lindgren, A., Chen, F., Jordan, P. W., and Zhang, H. (2008). Requirements for the design of advanced driver assistance systems-the differences between swedish and chinese drivers. *International Journal of Design*, 2(2).

[Liu et al., 2020a] Liu, B., Adeli, E., Cao, Z., Lee, K.-H., Shenoi, A., Gaidon, A., and Niebles, J. C. (2020a). Spatiotemporal relationship reasoning for pedestrian intent prediction. *IEEE Robotics and Automation Letters*, 5(2):3485–3492.

[Liu et al., 2018] Liu, J., Gu, Y., and Kamijo, S. (2018). Integral customer pose estimation using body orientation and visibility mask. *Multimedia Tools and Applications*, 77.

[Liu et al., 2016] Liu, J., Shahroudy, A., Xu, D., and Wang, G. (2016). Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*, pages 816–833. Springer.

[Liu et al., 2019] Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J. (2019). On the Variance of the Adaptive Learning Rate and Beyond. *arXiv e-prints*, page arXiv:1908.03265.

[Liu et al., 2019] Liu, X., Yan, M., and Bohg, J. (2019). Meteornet: Deep learning on dynamic 3d point cloud sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9246–9255.

[Liu et al., 2020b] Liu, Y., Yan, Q., and Alahi, A. (2020b). Social NCE: Contrastive Learning of Socially-aware Motion Representations. *arXiv:2012.11717 [cs]*.

[Londe, 1893] Londe, A. (1893). *La photographie médicale: Application aux sciences médicales et physiologiques*.

[Lorenzo et al., 2021a] Lorenzo, J., Alonso, I. P., Izquierdo, R., Ballardini, A. L., Saz, Á. H., Llorca, D. F., and Sotelo, M. Á. (2021a). Capformer: Pedestrian crossing action prediction using transformer. *Sensors*, 21(17):5694.

[Lorenzo et al., 2021b] Lorenzo, J., Parra, I., and Sotelo, M. (2021b). Intformer: Predicting pedestrian intention with the aid of the transformer architecture. *arXiv preprint arXiv:2105.08647*.

[Ludl et al., 2019] Ludl, D., Gulde, T., and Curio, C. (2019). Simple yet efficient real-time pose-based action recognition. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 581–588. IEEE.

[Ma et al., 2016] Ma, Y., Lee, E. W. M., and Yuen, R. K. K. (2016). An artificial intelligence-based approach for simulating pedestrian movement. *IEEE Transactions on Intelligent Transportation Systems*, 17(11):3159–3170.

[Maas, 2013] Maas, A. L. (2013). Rectifier nonlinearities improve neural network acoustic models.

[Maghoumi and LaViola Jr, 2019] Maghoumi, M. and LaViola Jr, J. J. (2019). Deepgru: Deep gesture recognition utility. In *International Symposium on Visual Computing*, pages 16–31. Springer.

[Mahajan et al., 2018] Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., and Van Der Maaten, L. (2018). Exploring the limits of weakly supervised pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, pages 181–196.

[Malla et al., 2020] Malla, S., Dariush, B., and Choi, C. (2020). Titan: Future forecast using action priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11186–11196.

[Mangalam et al., 2020] Mangalam, K., Girase, H., Agarwal, S., Lee, K.-H., Adeli, E., Malik, J., and Gaidon, A. (2020). It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, pages 759–776. Springer.

[Marginean et al., 2019] Marginean, A., Brehar, R., and Negru, M. (2019). Understanding pedestrian behaviour with pose estimation and recurrent networks. In *2019 6th International Symposium on Electrical and Electronics Engineering (ISEEE)*, pages 1–6.

[Martinez et al., 2017] Martinez, J., Hossain, R., Romero, J., and Little, J. J. (2017). A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649.

[Mazhar et al., 2018] Mazhar, O., Ramdani, S., Navarro, B., Passama, R., and Cherubini, A. (2018). Towards Real-Time Physical Human-Robot Interaction Using Skeleton Information and Hand Gestures. In *IROS: Intelligent Robots and Systems*, pages 1–6, Madrid, Spain.

[Mehta et al., 2018] Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Sridhar, S., Pons-Moll, G., and Theobalt, C. (2018). Single-shot multi-person 3d pose estimation from monocular rgb. In *3D Vision (3DV), 2018 Sixth International Conference on*. IEEE.

[Mehta et al., 2021] Mehta, V., Dhall, A., Pal, S., and Khan, S. S. (2021). Motion and region aware adversarial learning for fall detection with thermal imaging. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6321–6328. IEEE.

[Min et al., 2020] Min, Y., Zhang, Y., Chai, X., and Chen, X. (2020). An efficient pointlstm for point clouds based gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5761–5770.

[Moore, 1953] Moore, R. L. (1953). Pedestrian choice and judgment. *Journal of the Operational Research Society*, 4(1):3–10.

[Mousavi Hondori and Khademi, 2014] Mousavi Hondori, H. and Khademi, M. (2014). A review on technical and clinical impact of microsoft kinect on physical therapy and rehabilitation. *Journal of medical engineering*, 2014.

[Naeini et al., 2015] Naeini, M. P., Cooper, G., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[Neogi et al., 2020] Neogi, S., Hoy, M., Dang, K., Yu, H., and Dauwels, J. (2020). Context model for pedestrian intention prediction using factored latent-dynamic conditional random fields. *IEEE Transactions on Intelligent Transportation Systems*, 22(11):6821–6832.

[Newell et al., 2017] Newell, A., Huang, Z., and Deng, J. (2017). Associative embedding: End-to-end learning for joint detection and grouping. *Advances in Neural Information Processing Systems*, 2017-December:2278–2288. 31st Annual Conference on Neural Information Processing Systems, NIPS 2017 ; Conference date: 04-12-2017 Through 09-12-2017.

[Nguyen et al., 2019] Nguyen, X. S., Brun, L., Lézoray, O., and Bougleux, S. (2019). A neural network based on spd manifold learning for skeleton-based hand gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12036–12045.

[Nie et al., 2017] Nie, B. X., Wei, P., and Zhu, S.-C. (2017). Monocular 3d human pose estimation by predicting depth on joints. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3467–3475. IEEE.

[Nikhil and Tran Morris, 2018] Nikhil, N. and Tran Morris, B. (2018). Convolutional neural network for trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*.

[Ning and Huang, 2019] Ning, G. and Huang, H. (2019). LightTrack: A Generic Framework for Online Top-Down Human Pose Tracking. *arXiv e-prints*, page arXiv:1905.02822.

[Northcutt et al., 2021] Northcutt, C., Jiang, L., and Chuang, I. (2021). Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411.

[Nunez et al., 2018] Nunez, J. C., Cabido, R., Pantrigo, J. J., Montemayor, A. S., and Velez, J. F. (2018). Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. *Pattern Recognition*, 76:80–94.

[Ohn-Bar and Trivedi, 2013] Ohn-Bar, E. and Trivedi, M. (2013). Joint angles similarities and hog2 for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 465–470.

[Olsen et al., 2018] Olsen, N. L., Markussen, B., and Raket, L. L. (2018). Simultaneous inference for misaligned multivariate functional data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 67(5):1147–1176.

[Oreifej and Liu, 2013] Oreifej, O. and Liu, Z. (2013). Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Oudejans et al., 1996] Oudejans, R. R., Michaels, C. F., Van Dort, B., and Frissen, E. J. (1996). To cross or not to cross: The effect of locomotion on street-crossing behavior. *Ecological psychology*, 8(3):259–267.

[Ovadia et al., 2019] Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., and Snoek, J. (2019). Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32:13991–14002.

[Papandreou et al., 2017] Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., and Murphy, K. (2017). Towards Accurate Multi-person Pose Estimation in the Wild. *arXiv e-prints*, page arXiv:1701.01779.

[Patil et al., 2022] Patil, V., Sakaridis, C., Liniger, A., and Van Gool, L. (2022). P3depth: Monocular depth estimation with a piecewise planarity prior. *arXiv preprint arXiv:2204.02091*.

[Pellegrini et al., 2010] Pellegrini, S., Ess, A., and Van Gool, L. (2010). Improving data association by joint modeling of pedestrian trajectories and groupings. In *European conference on computer vision*, pages 452–465. Springer.

[Pfeiffer et al., 2018] Pfeiffer, M., Paolo, G., Sommer, H., Nieto, J., Siegwart, R., and Cadena, C. (2018). A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5921–5928. IEEE.

[Pham et al., 2018] Pham, H.-H., Khoudour, L., Crouzil, A., Zegers, P., and Velastin, S. A. (2018). Learning to recognise 3d human action from a new skeleton-based representation using deep convolutional neural networks. *IET Computer Vision*, 13(3):319–328.

[Picard, 2011] Picard, F. (2011). *Contextualisation*
Capture de Gestuelles Utilisateur : Contributions à l'Adaptativité des Applications Interactives Scénarisées. Theses, Université de La Rochelle.

[Piccoli et al., 2020] Piccoli, F., Balakrishnan, R., Perez, M. J., Sachdeo, M., Nunez, C., Tang, M., Andreasson, K., Bjurek, K., Dass Raj, R., Davidsson, E., Eriksson, C., Hagman, V., Sjoberg, J., Li, Y., Srikar Muppirisetty, L., and Roychowdhury, S. (2020). FuSSI-Net: Fusion of Spatio-temporal Skeletons for Intention Prediction Network. *arXiv e-prints*, page arXiv:2005.07796.

[Piergiovanni and Ryoo, 2019] Piergiovanni, A. and Ryoo, M. S. (2019). Representation flow for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9945–9953.

[Pishchulin et al., 2015] Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., and Schiele, B. (2015). DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation. *arXiv e-prints*, page arXiv:1511.06645.

[Pop et al., 2019] Pop, D. O., Rogozan, A., Chatelain, C., Nashashibi, F., and Bensrhair, A. (2019). Multi-task deep learning for pedestrian detection, action recognition and time to cross prediction. *IEEE Access*, 7:149318–149327.

[Qi et al., 2017] Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*.

[Qiu et al., 2017] Qiu, Z., Yao, T., and Mei, T. (2017). Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541.

[Raaj et al., 2019] Raaj, Y., Idrees, H., Hidalgo, G., and Sheikh, Y. (2019). Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Ranga et al., 2020] Ranga, A., Giruzzi, F., Bhanushali, J., Wirbel, E., Pérez, P., Vu, T.-H., and Perotton, X. (2020). Vrunet: Multi-task learning model for intent prediction of vulnerable road users. *Electronic Imaging*, 2020(16):109–1.

[Ranzato et al., 2006] Ranzato, M., Poultney, C., Chopra, S., and Cun, Y. (2006). Efficient learning of sparse representations with an energy-based model. *Advances in neural information processing systems*, 19.

[Rasouli et al., 2019a] Rasouli, A., Kotseruba, I., Kunic, T., and Tsotsos, J. K. (2019a). Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction. In *ICCV*.

[Rasouli et al., 2017a] Rasouli, A., Kotseruba, I., and Tsotsos, J. K. (2017a). Agreeing to cross: How drivers and pedestrians communicate. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 264–269.

[Rasouli et al., 2017b] Rasouli, A., Kotseruba, I., and Tsotsos, J. K. (2017b). Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 206–213.

[Rasouli et al., 2018] Rasouli, A., Kotseruba, I., and Tsotsos, J. K. (2018). Towards social autonomous vehicles: Understanding pedestrian-driver interactions. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 729–734.

[Rasouli et al., 2019b] Rasouli, A., Kotseruba, I., and Tsotsos, J. K. (2019b). Pedestrian action anticipation using contextual feature fusion in stacked rnns. In *BMVC*.

[Rasouli et al., 2021] Rasouli, A., Rohani, M., and Luo, J. (2021). Bifold and semantic reasoning for pedestrian behavior prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15600–15610.

[Rasouli and Tsotsos, 2019] Rasouli, A. and Tsotsos, J. K. (2019). Autonomous vehicles that interact with pedestrians: A survey of theory and practice. *IEEE transactions on intelligent transportation systems*, 21(3):900–918.

[Rehder et al., 2014] Rehder, E., Kloeden, H., and Stiller, C. (2014). Head detection and orientation estimation for pedestrian safety. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2292–2297. IEEE.

[Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

[Rhinehart et al., 2018] Rhinehart, N., Kitani, K. M., and Vernaza, P. (2018). R2p2: A reparameterized pushforward policy for diverse, precise generative path forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 772–788.

[Rhinehart et al., 2019] Rhinehart, N., McAllister, R., Kitani, K., and Levine, S. (2019). Precog: Prediction conditioned on goals in visual multi-agent settings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2821–2830.

[Rogez et al., 2019] Rogez, G., Weinzaepfel, P., and Schmid, C. (2019). Lcr-net++: Multi-person 2d and 3d pose detection in natural images. *IEEE transactions on pattern analysis and machine intelligence*.

[Rosenbloom et al., 2004] Rosenbloom, T., Nemrodov, D., and Barkan, H. (2004). For heaven's sake follow the rules: pedestrians' behavior in an ultra-orthodox and a non-orthodox city. *Transportation Research Part F: Traffic Psychology and Behaviour*, 7(6):395–404.

[Roy and Liersch, 2013] Roy, M. M. and Liersch, M. J. (2013). I am a better driver than you think: Examining self-enhancement for driving ability. *Journal of applied social psychology*, 43(8):1648–1659.

[Rozenberg et al., 2021] Rozenberg, R., Gesnouin, J., and Moutarde, F. (2021). Asymmetrical bi-rnn for pedestrian trajectory encoding. *arXiv preprint arXiv:2106.04419*.

[Rudenko et al., 2020] Rudenko, A., Palmieri, L., Herman, M., Kitani, K. M., Gavrila, D. M., and Arras, K. O. (2020). Human motion trajectory prediction: a survey. *The International Journal of Robotics Research*, 39(8):895–935. Publisher: SAGE Publications Ltd STM.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

[Saleh et al., 2019] Saleh, K., Hossny, M., and Nahavandi, S. (2019). Real-time intent prediction of pedestrians for autonomous ground vehicles via spatio-temporal densenet. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9704–9710. IEEE.

[Scarselli et al., 2008] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2008). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.

[Schmidt and Faerber, 2009] Schmidt, S. and Faerber, B. (2009). Pedestrians at the kerb–recognising the action intentions of humans. *Transportation research part F: traffic psychology and behaviour*, 12(4):300–310.

[Schneemann and Heinemann, 2016] Schneemann, F. and Heinemann, P. (2016). Context-based detection of pedestrian crossing intention for autonomous driving in urban environments. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2243–2248.

[Schneider and Gavrila, 2013] Schneider, N. and Gavrila, D. (2013). Pedestrian path prediction with recursive bayesian filters: A comparative study. In *GCPR*.

[Schulz and Stiefelhagen, 2015] Schulz, A. T. and Stiefelhagen, R. (2015). Pedestrian intention recognition using latent-dynamic conditional random fields. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 622–627. IEEE.

[Schuster and Paliwal, 1997] Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

[Schöller et al., 2020] Schöller, C., Aravantinos, V., Lay, F., and Knoll, A. (2020). What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robotics and Automation Letters*, 5(2):1696–1703.

[Shah et al., 2018] Shah, A. K., Ghosh, R., and Akula, A. (2018). A spatio-temporal deep learning approach for human action recognition in infrared videos. In *Optics and Photonics for Information Processing XII*, volume 10751, page 1075111. International Society for Optics and Photonics.

[Shahroudy et al., 2016] Shahroudy, A., Liu, J., Ng, T.-T., and Wang, G. (2016). Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019.

[Shi et al., 2019] Shi, L., Zhang, Y., Cheng, J., and Lu, H. (2019). Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12026–12035.

[Shi et al., 2015] Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., and Woo, W. C. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 2015:802–810.

[Shi et al., 2017] Shi, Y., Tian, Y., Wang, Y., Zeng, W., and Huang, T. (2017). Learning long-term dependencies for action recognition with a biologically-inspired deep network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 716–725.

[Shukla et al., 2017] Shukla, P., Biswas, K. K., and Kalra, P. K. (2017). Recurrent neural network based action recognition from 3d skeleton data. In *2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 339–345. IEEE.

[Si et al., 2019] Si, C., Chen, W., Wang, W., Wang, L., and Tan, T. (2019). An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Simo-Serra et al., 2013] Simo-Serra, E., Quattoni, A., Torras, C., and Moreno-Noguer, F. (2013). A joint model for 2d and 3d pose estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3634–3641.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 27, pages 568–576. Curran Associates, Inc.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv e-prints*, page arXiv:1409.1556.

[Singh and Suddamalla, 2021] Singh, A. and Suddamalla, U. (2021). Multi-input fusion for practical pedestrian intention prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2304–2311.

[Sisiopiku and Akin, 2003] Sisiopiku, V. P. and Akin, D. (2003). Pedestrian behaviors at and perceptions towards various pedestrian facilities: an examination based on observation and survey data. *Transportation research part f: traffic psychology and behaviour*, 6(4):249–274.

[Smith et al., 2017] Smith, S. L., Kindermans, P.-J., Ying, C., and Le, Q. V. (2017). Don't Decay the Learning Rate, Increase the Batch Size. *arXiv e-prints*, page arXiv:1711.00489.

[Song et al., 2017] Song, S., Lan, C., Xing, J., Zeng, W., and Liu, J. (2017). An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *Thirty-first AAAI conference on artificial intelligence*.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

[Stock and Cisse, 2018] Stock, P. and Cisse, M. (2018). Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 498–512.

[Sun et al., 2003] Sun, D., Ukkusuri, S., Benekohal, R. F., and Waller, S. T. (2003). Modeling of motorist-pedestrian interaction at uncontrolled mid-block crosswalks. In *Transportation Research Record, TRB Annual Meeting CD-ROM, Washington, DC*.

[Sun et al., 2018] Sun, L., Yan, Z., Mellado, S. M., Hanheide, M., and Duckett, T. (2018). 3dof pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5942–5948. IEEE.

[Sun et al., 2015] Sun, R., Zhuang, X., Wu, C., Zhao, G., and Zhang, K. (2015). The estimation of vehicle speed and stopping distance by pedestrians crossing streets in a naturalistic traffic environment. *Transportation research part F: traffic psychology and behaviour*, 30:97–106.

[Sun et al., 2017] Sun, X., Shang, J., Liang, S., and Wei, Y. (2017). Compositional human pose regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2602–2611.

[Sun et al., 2020] Sun, Z., Ke, Q., Rahmani, H., Bennamoun, M., Wang, G., and Liu, J. (2020). Human action recognition from various data modalities: A review. *arXiv preprint arXiv:2012.11866*.

[Tan and Le, 2019] Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.

[Tekin et al., 2016] Tekin, B., Katircioglu, I., Salzmann, M., Lepetit, V., and Fua, P. (2016). Structured prediction of 3d human pose with deep neural networks. *arXiv preprint arXiv:1605.05180*.

[Tian et al., 2013] Tian, R., Du, E. Y., Yang, K., Jiang, P., Jiang, F., Chen, Y., Sherony, R., and Takahashi, H. (2013). Pilot study on pedestrian step frequency in naturalistic driving environment. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1215–1220. IEEE.

[Tom and Granié, 2011] Tom, A. and Granié, M.-A. (2011). Gender differences in pedestrian rule compliance and visual search at signalized and unsignalized crossroads. *Accident Analysis & Prevention*, 43(5):1794–1801.

[Tome et al., 2017] Tome, D., Russell, C., and Agapito, L. (2017). Lifting from the deep: Convolutional 3d pose estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2500–2509.

[Toshev and Szegedy, 2014] Toshev, A. and Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660.

[Tran et al., 2014] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2014). Learning Spatiotemporal Features with 3D Convolutional Networks. *arXiv e-prints*, page arXiv:1412.0767.

[Tran et al., 2018] Tran, K., Bisazza, A., and Monz, C. (2018). The importance of being recurrent for modeling hierarchical structure. *arXiv preprint arXiv:1803.03585*.

[Uber, 2020] Uber (2020). Uber atg safety report 2020. `https://uber.app.box.com/v/uberatgsafetyreport`,.

[Ullah et al., 2017] Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., and Baik, S. W. (2017). Action recognition in video sequences using deep bi-directional lstm with cnn features. *IEEE Access*, 6:1155–1166.

[van den Berg et al., 2011] van den Berg, J., Guy, S. J., Lin, M., and Manocha, D. (2011). Reciprocal n-body collision avoidance. In Pradalier, C., Siegwart, R., and Hirzinger, G., editors, *Robotics Research*, pages 3–19, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Varytimidis et al., 2018] Varytimidis, D., Alonso-Fernandez, F., Duran, B., and Englund, C. (2018). Action and intention recognition of pedestrians in urban traffic. In *2018 14th International conference on signal-image technology & internet-based systems (SITIS)*, pages 676–682. IEEE.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Vemula et al., 2018] Vemula, A., Muelling, K., and Oh, J. (2018). Social attention: Modeling attention in human crowds. In *2018 IEEE international Conference on Robotics and Automation (ICRA)*, pages 4601–4607. IEEE.

[Voillemin et al., 2021] Voillemin, T., Wannous, H., and Vandeborre, J.-P. (2021). 2d deep video capsule network with temporal shift for action recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 3513–3519. IEEE.

[Wang et al., 2020a] Wang, H., Song, Z., Li, W., and Wang, P. (2020a). A hybrid network for large-scale action recognition from rgb and depth modalities. *Sensors*, 20(11):3305.

[Wang and Wang, 2017] Wang, H. and Wang, L. (2017). Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 499–508.

[Wang et al., 2019] Wang, J., Qiu, K., Peng, H., Fu, J., and Zhu, J. (2019). Ai coach: Deep human pose estimation and analysis for personalized athletic training assistance. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 374–382.

[Wang et al., 2016a] Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., and Gool, L. V. (2016a). Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer.

[Wang et al., 2018] Wang, P., Li, W., Wan, J., Ogunbona, P., and Liu, X. (2018). Cooperative training of deep aggregation networks for rgb-d action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Wang et al., 2016b] Wang, P., Li, Z., Hou, Y., and Li, W. (2016b). Action recognition based on joint trajectory maps using convolutional neural networks. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 102–106.

[Wang et al., 2010] Wang, T., Wu, J., Zheng, P., and McDonald, M. (2010). Study of pedestrians' gap acceptance behavior when they jaywalk outside crossing facilities. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1295–1300. IEEE.

[Wang et al., 2017] Wang, X., Gao, L., Wang, P., Sun, X., and Liu, X. (2017). Two-stream 3-d convnet fusion for action recognition in videos with arbitrary size and length. *IEEE Transactions on Multimedia*, 20(3):634–644.

[Wang et al., 2020b] Wang, Y., Xiao, Y., Xiong, F., Jiang, W., Cao, Z., Zhou, J. T., and Yuan, J. (2020b). 3dv: 3d dynamic voxel for action recognition in depth video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 511–520.

[Waymo, 2021] Waymo (2021). Waymo safety report 2021. `https://downloads.ctfassets.net/sv23gofxcuiz/4gZ7ZUxd4SRj1D1W6z3rpR/2ea16814cdb42f9e8eb34cae4f30b35d/2021-03-waymo-safety-report.pdf`,.

[Wen et al., 2018] Wen, Y., Vicol, P., Ba, J., Tran, D., and Grosse, R. (2018). Flipout: Efficient pseudo-independent weight perturbations on mini-batches. *arXiv preprint arXiv:1803.04386*.

[Weng et al., 2018] Weng, J., Liu, M., Jiang, X., and Yuan, J. (2018). Deformable pose traversal convolution for 3d action and gesture recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 136–152.

[Wilde, 1980] Wilde, G. S. (1980). Immediate and delayed social interaction in road user behaviour. *Applied Psychology*, 29(4):439–460.

[Willis et al., 2004] Willis, A., Gjersoe, N., Havard, C., Kerridge, J., and Kukla, R. (2004). Human movement behaviour in urban spaces: Implications for the design and modelling of effective pedestrian environments. *Environment and Planning B: Planning and Design*, 31(6):805–828.

[Woo et al., 2018] Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19.

[Wu et al., 2019a] Wu, C., Wu, X.-J., and Kittler, J. (2019a). Spatial residual layer and dense connection block enhanced spatial temporal graph convolutional network for skeleton-based action recognition. In *proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0.

[Wu et al., 2019b] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. (2019b). A comprehensive survey on graph neural networks. *arXiv preprint arXiv:1901.00596*.

[Xiao et al., 2018] Xiao, B., Wu, H., and Wei, Y. (2018). Simple Baselines for Human Pose Estimation and Tracking. *arXiv e-prints*, page arXiv:1804.06208.

[Xie et al., 2018] Xie, S., Sun, C., Huang, J., Tu, Z., and Murphy, K. (2018). Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 305–321.

[Xiu et al., 2018] Xiu, Y., Li, J., Wang, H., Fang, Y., and Lu, C. (2018). Pose Flow: Efficient Online Pose Tracking. *arXiv e-prints*, page arXiv:1802.00977.

[Xue et al., 2017] Xue, H., Huynh, D. Q., and Reynolds, M. (2017). Bi-prediction: pedestrian trajectory prediction based on bidirectional lstm classification. In *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE.

[Yan et al., 2018] Yan, S., Xiong, Y., and Lin, D. (2018). Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*.

[Yang et al., 2022] Yang, D., Zhang, H., Yurtsever, E., Redmill, K., and Ozguner, U. (2022). Predicting pedestrian crossing intention with feature fusion and spatio-temporal attention. *IEEE Transactions on Intelligent Vehicles*.

[Yang et al., 2019] Yang, F., Sakti, S., Wu, Y., and Nakamura, S. (2019). Make Skeleton-based Action Recognition Model Smaller, Faster and Better. *arXiv e-prints*, page arXiv:1907.09658.

[Yang et al., 2019] Yang, F., Wu, Y., Sakti, S., and Nakamura, S. (2019). Make skeleton-based action recognition model smaller, faster and better. In *Proceedings of the ACM Multimedia Asia*, pages 1–6.

[Yang et al., 2021] Yang, G., Tang, H., Ding, M., Sebe, N., and Ricci, E. (2021). Transformers solve the limited receptive field for monocular depth prediction. *arXiv e-prints*, pages arXiv–2103.

[Yang et al., 2018a] Yang, W., Ouyang, W., Wang, X., Ren, J., Li, H., and Wang, X. (2018a). 3D Human Pose Estimation in the Wild by Adversarial Learning. *arXiv e-prints*, page arXiv:1803.09722.

[Yang et al., 2018b] Yang, Z., Li, Y., Yang, J., and Luo, J. (2018b). Action Recognition with Spatio-Temporal Visual Attention on Skeleton Image Sequences. *arXiv e-prints*, page arXiv:1801.10304.

[Yao et al., 2021] Yao, Y., Atkins, E., Johnson-Roberson, M., Vasudevan, R., and Du, X. (2021). Bitrap: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation. *IEEE Robotics and Automation Letters*, 6(2):1463–1470.

[Ye et al., 2020] Ye, F., Pu, S., Zhong, Q., Li, C., Xie, D., and Tang, H. (2020). Dynamic gcn: Context-enriched topology learning for skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 55–63.

[Yong Du et al., 2015] Yong Du, Wang, W., and Wang, L. (2015). Hierarchical recurrent neural network for skeleton based action recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118.

[Yue-Hei Ng et al., 2015] Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., and Toderici, G. (2015). Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702.

[Zecha et al., 2018] Zecha, D., Einfalt, M., Eggert, C., and Lienhart, R. (2018). Kinematic pose rectification for performance analysis and retrieval in sports. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1791–1799.

[Zhang et al., 2016] Zhang, B., Wang, L., Wang, Z., Qiao, Y., and Wang, H. (2016). Real-time action recognition with enhanced motion vector cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2718–2726.

[Zhang et al., 2019] Zhang, M. R., Lucas, J., Hinton, G., and Ba, J. (2019). Lookahead Optimizer: k steps forward, 1 step back. *arXiv e-prints*, page arXiv:1907.08610.

[Zhang et al., 2017a] Zhang, P., Lan, C., Xing, J., Zeng, W., Xue, J., and Zheng, N. (2017a). View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2117–2126.

[Zhang et al., 2019] Zhang, P., Ouyang, W., Zhang, P., Xue, J., and Zheng, N. (2019). Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12085–12094.

[Zhang et al., 2017b] Zhang, S., Liu, X., and Xiao, J. (2017b). On geometric features for skeleton-based action recognition using multilayer lstm networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 148–157. IEEE.

[Zhang et al., 2018] Zhang, X., Xu, C., Tian, X., and Tao, D. (2018). Graph Edge Convolutional Neural Networks for Skeleton Based Action Recognition. *arXiv e-prints*, page arXiv:1805.06184.

[Zhang, 2012] Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10.

[Zhao et al., 2019] Zhao, T., Xu, Y., Monfort, M., Choi, W., Baker, C., Zhao, Y., Wang, Y., and Wu, Y. N. (2019). Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134.

[Zhu et al., 2020] Zhu, Y., Ren, D., Fan, M., Qian, D., Li, X., and Xia, H. (2020). Robust trajectory forecasting for multiple intelligent agents in dynamic scene. *arXiv preprint arXiv:2005.13133*.

[Zolfaghari et al., 2017] Zolfaghari, M., Oliveira, G. L., Sedaghat, N., and Brox, T. (2017). Chained Multi-stream Networks Exploiting Pose, Motion, and Appearance for Action Classification and Detection. *arXiv e-prints*, page arXiv:1704.00616.

[Zong et al., 2021] Zong, M., Wang, R., Chen, X., Chen, Z., and Gong, Y. (2021). Motion saliency based multi-stream multiplier resnets for action recognition. *Image and Vision Computing*, 107:104108.

RÉSUMÉ

Le véhicule autonome est un défi majeur pour la mobilité de demain. Des progrès sont réalisés chaque jour pour y parvenir ; cependant, de nombreux problèmes restent à résoudre pour obtenir un résultat sûr pour les usagers de la route les plus vulnérables. L'un des principaux défis auxquels sont confrontés les véhicules autonomes est la capacité à conduire efficacement en milieu urbain. Une telle tâche nécessite la gestion des interactions entre les véhicules et les usagers vulnérables de la route afin de résoudre les ambiguïtés du trafic. Afin d'interagir avec ces usagers, les véhicules doivent être capables de comprendre leurs intentions et de prédire leurs actions à venir. Dans cette thèse, notre travail s'articule autour de la technologie d'apprentissage automatique comme moyen de comprendre et de prédire le comportement humain à partir de signaux visuels et plus particulièrement de la cinématique de pose. Notre objectif est de proposer un système d'assistance au véhicule qui soit léger, agnostique à la scène et qui puisse être facilement implémenté dans n'importe quel dispositif embarqué avec des contraintes temps réel. Premièrement, dans le domaine de la reconnaissance de gestes et d'actions, nous étudions et introduisons différentes représentations de la cinématique de pose, basées sur des modèles d'apprentissage profond afin d'exploiter efficacement leurs composantes spatiales et temporelles tout en restant dans un espace euclidien. Deuxièmement, dans le domaine de la conduite autonome, nous montrons qu'il est possible de lier la posture, l'attitude de marche et les comportements futurs des protagonistes d'une scène sans utiliser les informations contextuelles de la scène. Cela nous permet de diviser par un facteur 20 le temps d'inférence des approches existantes pour la prédiction de l'intention des piétons tout en gardant la même robustesse de prédiction. Finalement, nous évaluons la capacité de généralisation des approches de prédiction d'intention de piétons et montrons que le mode d'évaluation classique des approches pour la prédiction de traversée de piétons, n'est pas suffisante pour comparer ni conclure efficacement sur leur applicabilité lors d'un scénario réel. Nous proposons de nouveaux protocoles et de nouvelles mesures basés sur l'estimations d'incertitude afin de rendre le domaine de recherche plus durable et plus représentatif des réelles avancées à venir.

MOTS CLÉS

Prédiction d'action basée sur le squelette, Apprentissage des représentations spatio-temporelles, estimation de l'incertitude prédictive

ABSTRACT

The autonomous vehicle (AV) is a major challenge for the mobility of tomorrow. Progress is being made every day to achieve it; however, many problems remain to be solved to achieve a safe outcome for the most vulnerable road users (VRUs). One of the major challenge faced by AVs is the ability to efficiently drive in urban environments. Such a task requires interactions between autonomous vehicles and VRUs to resolve traffic ambiguities. In order to interact with VRUs, AVs must be able to understand their intentions and predict their incoming actions. In this dissertation, our work revolves around machine learning technology as a way to understand and predict human behaviour from visual signals and more specifically pose kinematics. Our goal is to propose an assistance system to the AV that is lightweight, scene-agnostic that could be easily implemented in any embedded devices with real-time constraints. Firstly, in the gesture and action recognition domain, we study and introduce different representations for pose kinematics, based on deep learning models as a way to efficiently leverage their spatial and temporal components while staying in an euclidean grid-space. Secondly, in the autonomous driving domain, we show that it is possible to link the posture, the walking attitude and the future behaviours of the protagonists of a scene without using the contextual information of the scene (zebra crossing, traffic light...). This allowed us to divide by a factor of 20 the inference speed of existing approaches for pedestrian intention prediction while keeping the same prediction robustness. Finally, we assess the generalization capabilities of pedestrian crossing predictors and show that the classical train-test sets evaluation for pedestrian crossing prediction, *i.e.*, models being trained and tested on the same dataset, is not sufficient to efficiently compare nor conclude anything about their applicability in a real-world scenario. To make the research field more sustainable and representative of the real advances to come. We propose new protocols and metrics based on uncertainty estimates under domain-shift in order to reach the end-goal of pedestrian crossing behavior predictors: vehicle implementation.

KEYWORDS

Skeleton-based action prediction, Learning spatio-temporal representations, Predictive uncertainty estimation