Transportation Research Part C 130 (2021) 103259

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Pedestrian intention prediction: A convolutional bottom-up multi-task approach

Haziq Razali *, Taylor Mordan, Alexandre Alahi

Visual Intelligence for Transportation Laboratory, EPFL, Switzerland

ARTICLE INFO

Keywords:
Traffic Management Systems
Advanced Driver Assistance Systems
Autonomous Vehicles
Pedestrian Intention Prediction
Human Pose Estimation
Human Behaviour Analysis

ABSTRACT

The ability to predict pedestrian behaviour is crucial for road safety, traffic management systems, Advanced Driver Assistance Systems (ADAS), and more broadly autonomous vehicles. We present a vision-based system that simultaneously locates where pedestrians are in the scene, estimates their body pose and predicts their intention to cross the road. Given a single image, our proposed neural network is designed using a bottom-up approach and thus runs at nearly constant time without relying on a pedestrian detector. Our method jointly detects human body poses and predicts their intention in a multitask framework. Experimental results show that the proposed model outperforms the precision scores of the state-of-the-art for the task of intention prediction by approximately 20% while running in real-time (5 fps). The source code is publicly available so that it can be easily integrated into an ADAS or into any traffic light management systems.

1. Introduction

Pedestrians exhibiting the intention to cross the road are normally identifiable through a certain set of cues. For instance, they turn their heads to look for incoming traffic as they approach the crosswalk. They also do not cross and instead wait at the sidewalk if there are nearby vehicles that have not come to a stop. The ability to learn these behaviours and use them to predict human motion in urban areas is extremely valuable in the transportation domain as it can be used for traffic management or in Advanced Driver Assistance Systems (ADAS) and autonomous vehicles for example.

In the context of traffic management, the emphasis is on minimizing pedestrian delay and traffic congestion by deciding when to bring oncoming traffic to a halt for pedestrian crossing, and when not to. For example, the traffic management system should trigger the red light for vehicles and the green for pedestrians if it senses a large number of pedestrians wanting to cross an empty road. If traffic is heavy however, then the system should not further congest vehicular traffic by triggering the red light for a single pedestrian. In the context of Advanced Driver Assistance Systems (ADAS) and autonomous vehicles, there is a much greater emphasis on early warnings as these systems need to predict pedestrian intention early enough to provide ample time for the driver to respond to in order to prevent a collision. Despite its importance, methods in use currently only focus on detecting pedestrians (Dalal and Triggs, 2005; Boudet and Midenet, 2009; Dollar et al., 2012; Seer et al., 2012; Alahi et al., 2014). However, a pedestrian who is currently on the sidewalk but is dashing towards the road may not be in the way of the vehicle and may therefore, not trigger the system to warn of an imminent collision until it is too late.

Recently, researchers have proposed two streams of work to forecast the future: either by predicting the pedestrian's trajectory

* Corresponding author.
E-mail addresses: muhammad.binrazali@alumni.epfl.ch (H. Razali), taylor.mordan@epfl.ch (T. Mordan), alexandre.alahi@epfl.ch (A. Alahi).

https://doi.org/10.1016/j.trc.2021.103259
Received 3 June 2020; Received in revised form 2 March 2021; Accepted 4 June 2021
Available online 15 July 2021
0968-090X/© 2021 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

(a) Input Image (b) Intention Map (c) Pose Map

Fig. 1. Given a single input image, our model generates an intention map and a pose map in a single feed-forward pass with a running time independent of the number of pedestrians. Pixels highlighted in blue constitute pedestrians that are about to or are currently crossing the road, and in green, pedestrians that are not about to, nor are currently crossing the road. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(Alahi et al., 2016; Fernandoa et al., 2017; Lee et al., 2017; Bartoli et al., 2017; Gupta et al., 2018; Sadeghian et al., 2018) or by predicting his intention to cross the road as a classification task (Rasouli et al., 2017; Saleh et al., 2019; Gujjar and Vaughan, 2019; Neogi et al., 2017). These works however, are built based on a top-down approach, requiring a person detector before forecasting the future for each detection. As such, they suffer from early commitment as there is no recourse to recovery should the detector fail. Furthermore, their run-time is proportional to the number of candidate boxes as the predictive model is run for each positive detection. For these reasons, there still exists an unmet need to improve the analysis of pedestrian behaviour at road crossings.

In this paper, we want to predict the pedestrians' intention to cross the road as early as possible given a single image. We utilize a single frame as opposed to multiple frames to eliminate the bias behind driving dynamics when collecting the data. For instance, the driver may break when seeing a pedestrian at the edge of the sidewalk and introduce a bias in the temporal model. We present a new neural network for the primary task of pedestrian intention prediction. Our model takes as input only a single RGB image and generates a map predicting the probability that each pixel constitutes a pedestrian who is either crossing or not (Fig. 1b), bypassing the need for a people detector and running at constant time. We additionally have our model output in parallel, the detailed human body pose for each pedestrian (Fig. 1c) c) to show that our network can be easily extended to perform a variety of other tasks with little overhead. The byproduct of the intention or pose map also allows the model to function as a generic people detector. Finally, because our model makes use of only RGB cameras, it can be easily integrated into an ADAS, autonomous vehicles or any traffic light management systems [1].

The remainder of this paper is structured as follows: we begin in Section 2 with a brief overview of existing methods that can be applied to the problem of intention prediction and introduce some existing multitask models. We then introduce our neural network in Section 3 where we discuss each of its components in detail. In Section 4, we provide a quick overview of the dataset and in Section 5, training details such as data pre-processing and optimization. Experiments are done in Section 6. Here we compare our method to several baselines, conduct several ablative studies and study the forward pass of our architecture. Finally, we conclude the paper and discuss possible future work.

2. Related Work

2.1. Pedestrian Intention Prediction

Methods to predict a pedestrian's intent can be grouped into two categories: (1) those that formulate the task as a problem of trajectory prediction where the eventual aim is to determine if the generated tracks cross the road (Rehder et al., 2018; Saleh et al., 2017; Sarkar et al., 2017; Batkovic et al., 2018; Amirian et al., 2019; Völz et al., 2019; Cheng et al., 2018; Saleh et al., 2018; Kooij et al., 2014; Keller and Gavrila, 2013; Quintero et al., 2014; Møgelmose et al., 2015; Quintero et al., 2014; Dominguez-Sanchez et al., 2017; Mínguez et al., 2018), and (2) those that treat it as a binary classification problem in which the output is the pedestrian's predicted intent (Köhler et al., 2015; Schulz and Stiefelhagen, 2015; Hashimoto et al., 2015; Hashimoto et al., 2015; Hashimoto et al., 2015; Zhao et al., 2019; Bieshaar et al., 2018; Neogi et al., 2017; Ghori et al., 2018; Hoy et al., 2018; Gujjar and Vaughan, 2019; Chaabane et al., 2020; Liu et al., 2020; Bonnin et al., 2014; Jeong et al., 2017; Fang et al., 2017; Fang and López, 2018; Quintero et al., 2015; Mínguez et al., 2018; Quintero et al., 2017; Koschi et al., 2018). We briefly review works built using deep learning and would encourage readers to refer to the vast literature for all other methods.

2.1.1. Trajectory Prediction

Neural networks built to predict an individual's trajectory typically assume the coordinates to have already been converted from image to real-world. They often model either the interactions between people (Alahi et al., 2016; Fernandoa et al., 2017; Gupta et al., 2018; Alahi et al., 2017; Sun et al., 2015) or between both people and environment (Bartoli et al., 2017; Sadeghian et al., 2018; Lee et al., 2017). For instance, Alahi et al. (2016) introduced a pooling module that aggregates the hidden states of all other people within a local neighbourhood. Fernandoa et al. (2017) extended this idea to include all agents in the environment using an attention mechanism

[1] The source code is available at https://github.com/HaziqRazali/Pedestrian-Intention-Prediction.

to assign a weight to each agent based on its proximity. In (Lee et al., 2017 and Bartoli et al., 2017), the authors incorporated scene information into their predictive models with the idea that the generated trajectories must remain within spaces that are navigable, *e.g.*, sidewalks for pedestrians and roads for vehicles. In a later work, (Gupta et al., 2018 and Sadeghian et al., 2018) showed how replacing the l_2 loss function, which computes the mean squared error between the predicted and ground truth coordinates at each timestep, with the generative adversarial networks prevents the model from learning the mean trajectory due to the averaging effect inherent in the l_2 loss.

A limitation shared by all these works however, is the need to obtain a top-down view of the scene from a moving camera, the process of which can induce errors that result in inaccurate trajectories. The absence of pose information presents another weakness considering how the pedestrian's pose can be used as a strong indicator of his intention to cross the road. Finally, the need for multiple frames delays the speed at which predictions are made.

2.1.2. Binary Classification

Architectures developed for intention prediction via binary classification differ in a variety of ways. Models that operate on an RGB input use either 2D or 3D convolutions, *i.e.*, filters that slide along the height and width or along the height, width and temporal depth, respectively. In the case of 2D Convolutional Neural Network (CNN), information is propagated across time either via LSTMs (Hochreiter and Schmidhuber, 1997) or feature aggregation over time (Karpathy et al., 2014). For instance, the authors in (Rasouli et al., 2017) proposed a two-stream architecture that takes as input a single crop of the pedestrian and the scene, sending them both through two separate CNNs to produce two feature vectors that are then concatenated for classification. An extension was later done in (Saleh et al., 2019) with the use of LSTMs and 3D CNNs and in (Gujjar and Vaughan, 2019) by generating several frames into the future and classifying using these frames. There also exist methods that operate directly on a skeleton of the individual (Shahroudy et al., 2016) where the main advantage is that the dimensions of the data is much lower (*e.g.*, 17 skeleton joints compared to the 2048 ResNet feature vector), resulting in models that are less likely to overfit. In this work, we will also experiment with intention prediction via keypoints. Lastly, there also exist works that utilize handcrafted features (Neogi et al., 2017) such as the distance of the pedestrian from the vehicle, his lateral motion, his surroundings as well as the vehicle's velocity as input to a conditional random field.

Our method is closely related to the line of work on binary classification except that we do not employ a people detector but directly and jointly perform detection and intention prediction for each single pixel. As a result, we bypass the limitations inherent in said previous works that cannot recover from a missed detection and whose run-time become proportional to the number of detections in the image. In contrast, our method is able to generate predictions for where it thinks a pedestrian is with a running time that is independent of the total number of pedestrians in the image. Moreover, because our bottom-up method feeds the entire image through the network, we are, intuitively, letting the architecture make its decision based on the pedestrian's pose, where (s) he currently is in the scene as well as the surrounding objects in the context, *e.g.*, roads or buildings.

2.1.3. Human Pose and Actions

Human pose estimation consists in localizing all the joints and linking them together into a skeleton for each human in the images. There are mainly two categories of approaches: top-down and bottom-up. The first one starts with a human detection step, and then finds the skeleton within each detected bounding box. This category includes Mask R-CNN (He et al., 2017) and RMPE (Fang et al., 2017). On the other hand, bottom-up approaches first identify all the joints in the images, and link them to form instances and skeletons. PifPaf (Kreiss et al., 2019), OpenPose (Cao et al., 2019) and DeepCut (Pishchulin et al., 2016) are such methods. Human pose has been used as a relevant and meaningful intermediate representation used for action recognition (Wang et al., 2013; Luvizon et al., 2018; Agahian et al., 2020). In particular, this type of feature has received a lot of attention for crossing prediction (Fang et al., 2018; Quintero et al., 2015; Quintero et al., 2017; Fang et al., 2017; Wang and Papanikolopoulos, 2020). Indeed, pedestrians' poses should be highly correlated with their gaits, which we know carry significant information about their crossing intentions. In addition, pose features may be linked to other relevant properties, for example distance (Bertoni et al., 2019) or eye contact (Bertoni et al., 2020). In this paper, we use intermediate features learned through the pose estimation task to improve results on crossing intentions. Therefore, a better pose estimation model should lead to better learned features, and this would in return improve all tasks performed based on this representation.

2.2. Multitask Learning

Multitask Learning promises to increase generalization power by using the domain information contained in the training signals of related tasks as an inductive bias (Caruana, 1997; Nakamura et al., 2017). Tian et al. (2015) were able to improve the results of pedestrian detection by adding two auxiliary tasks related to learning attributes about the pedestrians and the scene. However, it can also lead to catastrophic forgetting of one task.

Multitask models typically have a number of shared layers termed as a base network followed by several task-specific layers that are also known as head networks. Their arrangement is both problem dependent and empirical, with some using a single shared representation (*e.g.*, the output of the 4th block of ResNet) as input to the head networks (Kreiss et al., 2019; He et al., 2017; Kendall et al., 2018; Kokkinos, 2017), and others, opting for a hierarchical approach in which increasingly complex tasks are predicted at successively deeper layers (Guo et al., 2018), *e.g.*, the 4^{th} block of ResNet for pedestrian detection and the 5^{th} for intention prediction.

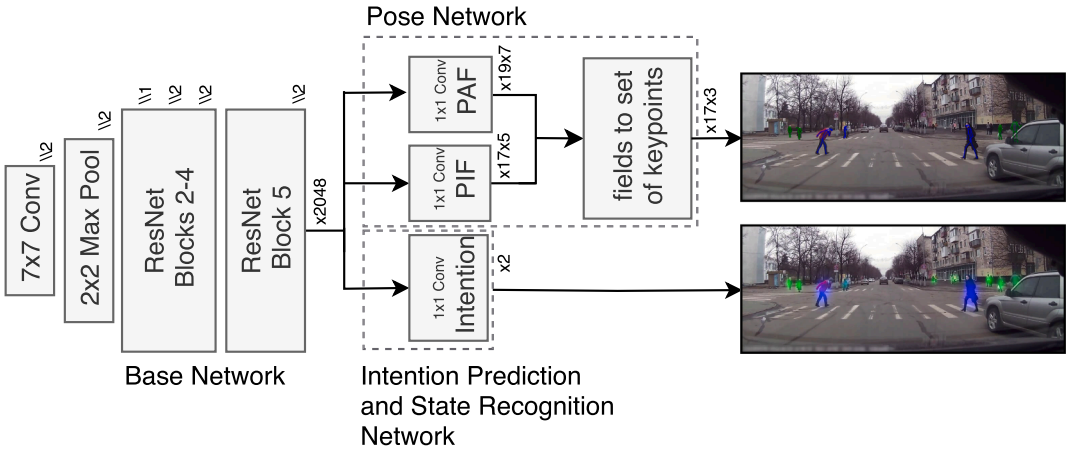


Fig. 2. Model architecture. The input is an image of size (H, W) with three color channels. An operation with stride two is indicated by "//2". The base network is a ResNet-50 with 5 blocks that produces a tensor of size $(H/16, W/16, 2048)$. The *pose network* produces Part-Intensity-Fields (PIF) and Part-Association-Fields (PAF) fields with 17×5 and 19×7 channels respectively that are then decoded to produce the pose estimates containing 17 joints each. The intention head network produces an activity map with 2 channels for the states"crossing" and "not crossing". The 1×1 convolution upsamples (Shi et al., 2016) the spatial resolution by a factor of 2 to produce an output of resolution of $(H/8, W/8)$. It is upscaled here for the sake of visualization.

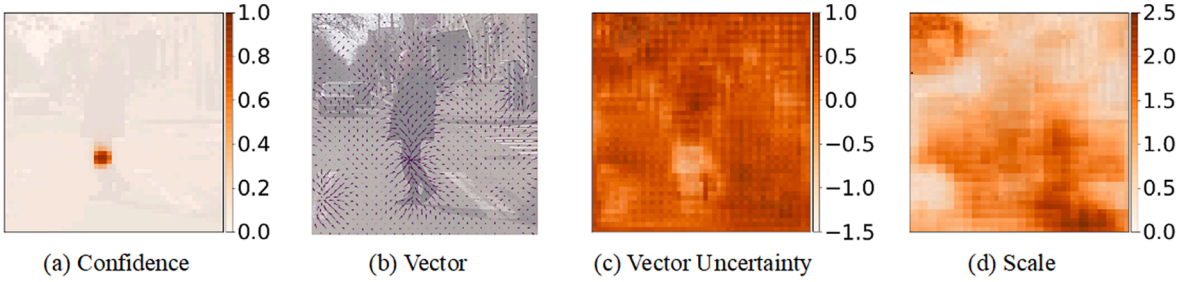


Fig. 3. Part-Intensity-Fields (PIF) for the left knee. There are a total of 17 sets of such maps; (a) predicts the probability of the knee at each pixel while (b) acts as a regressor for the confidence maps, (c) describes how uncertain the regressors are *i.e.,* regions that are whitish in color indicate that the vectors are indeed pointing to locations that contain the corresponding joint, and (d) predicts the scale or size of the knee.

3. Pedestrian Pose Estimation and Intention Prediction

3.1. Model Overview

The goal of our method is to estimate the pose and intention of each pedestrian given a single image. Our model is described in Fig. 2. It is a shared *base* network with two *head* networks: an intention prediction and state recognition head network that generates a map predicting the probability that each pixel belongs to a pedestrian who is either crossing or not, and a pose head network that estimates the pedestrian's pose. Our base network is a 5-block ResNet-50 and our head networks are 1x1 upsampling convolutional layers (Shi et al., 2016). Given an RGB image of size $(H, W, 3)$, the base network outputs a feature map of size $(H/16, W/16, 2048)$. This is then fed to the both head networks to perform their respective tasks. Note that both head networks run in parallel and make no use of a pedestrian detector. Our model achieves a frame rate of 5 fps for an image resolution of (378,960) on a GTX 1080 Ti. Finally, we point out that our design choice was mainly influenced by memory constraints. The ResNet-50 was the largest model that we could train on a single GTX 1080 Ti without compromising the batch size. Users with a larger memory limit would thus be able to benefit from the larger models for possible performance gains. We now briefly describe both head networks.

3.2. Pose Network

We use PifPaf (Kreiss et al., 2019) for our *pose network*. It is an encoder-decoder method that operates on the output of the base network for multi-person pose estimation. The encoders are two 1x1 convolutional network: a Part-Intensity-Field (PIF) network that outputs the location of *body joints* (left knee, left ankle, etc) and a Part-Association-Field (PAF) network that outputs the location of *body parts* connecting pairs of joints, *e.g.*, left knee to left ankle. The decoder is a greedy algorithm that converts these feature maps into sets of skeletons.

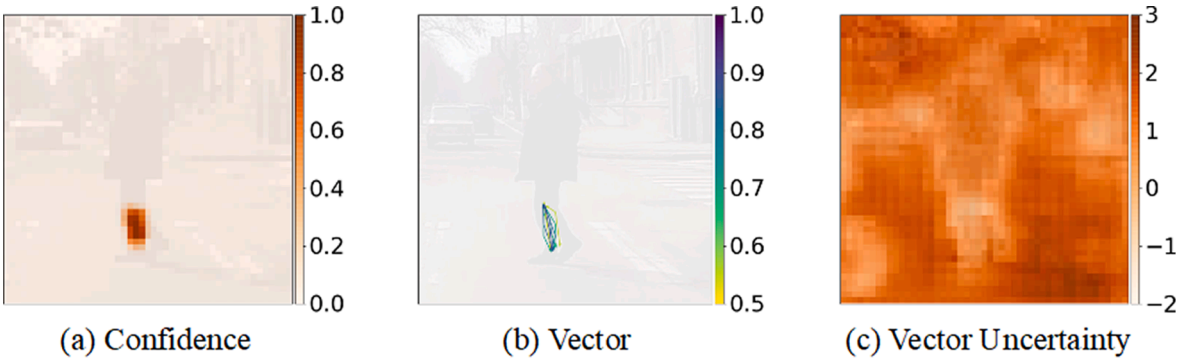


Fig. 4. Part-Association-Fields (PAF) for the body part connecting the left ankle to the left knee. There are a total of 19 sets of such maps; (a) predicts the probability of the body part connecting the left ankle joint to the left knee joint, (b) points to where the joints are, and (c) the uncertainty score for the vectors. Regions that are whitish in color thus indicate that the vectors are indeed pointing to locations that contain the corresponding joints.

Fig. 5. The input image shown in (a) is fed through the base network and subsequently the intention prediction and state recognition network to produce a map for the label "not crossing" shown in (b) and for the label "crossing" in (c).

The PIF network generates a map for each of the 17 body joints as defined in the COCO keypoints dataset. It outputs a 5-dimensional vector at every location of said map: (a) a score that predicts the probability that the location contains a joint, (b) a 2-dimensional vector that acts as a regressor, pointing to where the predicted joint is, (c) an uncertainty score for each vector and (d) a scale for the size of the joint. It is built using subpixel 1x1 convolutions (Shi et al., 2016), resulting in a final feature map of size ($H/8, W/8$, 17, 5). The maps for the joint left-knee are visualized in Fig. 3.

The PAF network generates a map for each of the 19 body parts. At each location, the map contains a 7-dimensional vector: a score predicting the probability of the body part, 2 vectors pointing to the joints associated to this body part (*e.g.*, left knee and left ankle) and 2 uncertainty scores for each vector pair. Similar to the PIF network, the PAF network is also built using subpixel 1x1 convolutions (Shi et al., 2016), resulting in a feature map of size ($H/4, W/4$, 19, 7). The maps for the body part that connects the left ankle to the left knee is shown in Fig. 4. The decoder starts by generating a seed at the location with the highest confidence as predicted by the PIF network. Connections to other joints are then added based on the PAF maps via a greedy decoding algorithm similar to what is used in (Papandreou et al., 2018). This process is repeated until the confidence map is exhausted.

For both of these networks, we use the independent binary cross entropy loss as the metric for the confidence maps, the l_1 loss for the scale and the Laplace loss for the vector maps. The component wise losses are expressed as:

$$L_{\mathrm{c}} = \sum_{p} p_i \log \widehat{p_i} + (1 - p_i) \log(1 - \widehat{p_i}), \tag{1}$$

$$L_{\mathrm{v}} = \sum_{x,y} \log b + b * \sqrt{((\hat{x} - x)^2 + (\hat{y} - y)^2)}, \tag{2}$$

$$L_{\mathrm{s}} = \sum_{p} ||\widehat{p_i} - p_i||_1, \tag{3}$$

where L_c, L_v, L_s represent the losses from the confidence, vector and scale maps respectively and p, x, y, b the predicted class, the x and y vectorial components and the predicted scale respectively. All in all, the loss for PIF-PAF can be expressed as:

$$L_{\mathrm{PIF-PAF}} = \lambda_a L_{\mathrm{PIF-c}} + \lambda_b L_{\mathrm{PIF-v}} + \lambda_c L_{\mathrm{PIF-s}} + \lambda_d L_{\mathrm{PAF-c}} + \lambda_e L_{\mathrm{PAF-v}}, \tag{4}$$

where the subscript PIF-c represent the confidence component of the PIF map and the lambdas used to balance the losses. In our experiments, we set $[\lambda_a, \lambda_b, \lambda_c, \lambda_d, \lambda_e] = [30, 2, 2, 50, 3]$.

Fig. 6. (a) Pedestrian lifetime from the moment they appear in the video to the moment they begin crossing the road. (b) Pedestrian lifetime from the moment they appear in the video to the moment they leave the field of view. (c) Pedestrian height at the point of crossing. Pedestrians with the state crossing and "not crossing" are shown in the figure with the bounding boxes colored in red and green respectively. Their heights in pixels are also shown for an input image of size (540,960). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.3. Intention Prediction and State Recognition Network

Our intention prediction and state recognition network generates a map with 2 channels for the labels "crossing" and "not crossing". In this map, each pedestrian is represented by a bivariate Gaussian distribution where the mean is located at the center of the pedestrian and the variances in x and y represent the pedestrian's width and height respectively (Fig. 5). Specifically, we consider the "crossing" and "not crossing" pedestrians as two classes with their respective channels. Each pixel in the first channel will have a value between 0 and 1 for the class "no pedestrian" and "crossing pedestrian" respectively. The values for each pixel in the second channel will also lie in a similar range but for the class "no pedestrian" and "not crossing pedestrian". Intuitively, each pixel in the map can be thought of as the probability that it constitutes a person and whether or not that person is crossing. We chose to represent the map in such a manner using 2 channels instead of only 1 to handle the scenario where a pixel belongs to both a crosser and a non-crossser. This can happen during partial occlusions or group crossings when the pedestrians are standing in close proximity to one another and the one standing closest to the sidewalk becomes the first to cross the road. At that instant, the Gaussians representing the crosser and the closest non-crosser would most likely overlap and a representation that uses only 1 channel would end up providing a training signal that is erroneous. With 2 channels, we provide the model the ability to handle such ambiguities and occlusions by giving it the freedom to concurrently generate a confident prediction for both crosser and non-crosser alike. We point out that this method of representation is fairly common in pose estimation where occlusions across multiple labels (joints) are highly common.

Our network is thus similar to the Convolutional Relational Machines in (Azar et al., 2019) but without the refinement stages. Additionally, while the authors in said work were using convolutions to model the spatial relations between volleyball players, we use it here as a means to encode the pedestrian and his surroundings for intention prediction. We would like to point out again that the main difference between our model and the existing classification-based ones (Rasouli et al., 2017; Saleh et al., 2019; Gujjar and Vaughan, 2019; Neogi et al., 2017) is that we do not send a cropped image of the pedestrian through a CNN, but the whole image. Therefore, our model requires no detector and makes its predictions for every pixel concurrently by running a series of convolutions and non-linear operations across the entire image. Each location (x, y) in the output map can thus be expressed as some non-linear function over a region of the input image. Intuitively, this can be interpreted as the model making its prediction based on the pedestrian's pose, where (s) he currently is in the scene as well as the surrounding objects in the context, *e.g.*, whether or not (s) he is currently standing at the sidewalk with buildings in the background or in the middle of the road with no other object in proximity. In other words, we believe that the model should be able to predict a pedestrian's intention by looking at the pedestrian and where (s) he

is in the scene as individuals about to cross the road are often facing the opposite sidewalk and in close proximity to the edge of the road. The network to generate the map is also a single 1x1 subpixel convolution that results in a map of size $(H/4, W/4, 2)$.

At test time, a pedestrian is declared a crosser if the average activation of the crossing map within the ground truth bounding box is larger than the average activation of the not-crossing map.

The head network is optimized for using the sum of squared errors between the ground truth and predicted maps and is expressed as:

$$L_{\text{intention}} = \lambda_f \sum_p ||\hat{p} - p||_2^2, \tag{5}$$

where again, the lambda is used to scale the loss coming from the intention network when trained jointly with the pose head network and is set to 0.2. Together, the loss is expressed as:

$$L = L_{\text{PIF-PAF}} + L_{\text{intention}}, \tag{6}$$

4. Dataset

We use the Joint Attention in Autonomous Driving (JAAD) dataset (Rasouli et al., 2017) that contains 346 videos, each 5–10 s long that were recorded by a camera mounted on a vehicle. The videos are recorded at a resolution of 1920x1080 at a frame rate of 30 fps. The dataset comes with ground truth bounding box annotations for the pedestrians as well as a behavioural tag that describes the state each pedestrian is currently in *e.g.*, whether or not (s) he is crossing. Note that the tag "crossing" is only assigned to pedestrians who cut across the vehicle. As such, a pedestrian who is crossing the road but is not in the way of the oncoming vehicle is not tagged as "crossing".

We plot in a bar graph, the pedestrian lifetime from the moment they appear in the video to the instant they begin crossing the road (Fig. 6a), and before leaving the frame (Fig. 6b). It can be seen that a majority of the pedestrians begin crossing only less than a second after appearing in the field of view. We also plot the distribution of pedestrian height in pixels to give us an idea of how close the pedestrians are to the vehicle the instant they leave the sidewalk (Fig. 6c) and and show some examples of their heights.

5. Training Details

Since the JAAD dataset does not include the ground truth keypoint annotations for the pedestrians, we augment our training set with the COCO keypoint dataset for pose estimation. During training, the model takes as input images from both datasets and outputs several sets of keypoints and an activity map for each image coming from the COCO and JAAD dataset respectively. Losses incurred are then backpropagated through the entire network for weight update.

5.1. Preprocessing

We resize images from the JAAD dataset to a resolution of (540,960) and crop the top 30% of the input frame to obtain a resolution of (376,960) before feeding it to the network to produce an activity map of size (48,120,2). The images were cropped in such a manner as the top 30% of the input frame contains only the sky and the upper sections of the trees and buildings and are thus not helpful in determining whether or not the pedestrians will cross the road. A smaller input image would also reduce the memory footprint, especially during training, at the cost of less visible details. Images from the COCO dataset are resized and padded to a resolution of (401,401) then sent through the network to produce PIF and PAF maps of sizes (51,51,17,5) and (51,51,19,7) respectively. We augment the JAAD dataset using horizontal flips and follow exactly the implementation as described in PifPaf (Kreiss et al., 2019) for the COCO dataset, either by randomly square-cropping the images or by adding bars to the shorter side to make them squares. We train on the first 300 videos of the JAAD dataset and validate on the remaining 46 and use the predefined train-val split for COCO. Overall, the JAAD training set contains 346 crossers with 65 k frames and 1854 non-crossers with 175 k frames and the validation set 79 crossers with 10 k frames and 276 non-crossers with 24 k frames.

5.2. Curriculum Learning

We initialize both the base and pose head networks with weights from the model that was trained in PifPaf. The intention prediction and state recognition head is initialized using PyTorch's default initializer. We found in our experiments that it was extremely crucial to initialize the architecture such that the model is already performant on one task, which in our case, was human pose estimation. We noted that our model was neither able to perform intention prediction nor pose estimation if it was initialized from scratch. Our technique of starting with a pre-trained network for human pose estimation is thus similar to the curriculum learning framework of (Bengio et al., 2009; Graves et al., 2017; Guo et al., 2018) that starts the training with an easy objective function containing a single task before ramping it up to multiple tasks and also datasets through data slicing. In our case, it would be similar to training the model on human pose estimation before increasing the difficulty to perform both human pose estimation and pedestrian intention prediction. We manually tune the weights for each head to ensure that the losses coming from any one of them does not dominate. In the end, we found it best to scale the loss coming from the intention head network by a factor of 0.2 and to use the same weights of [30,2,2,50,3] selected by the authors in PifPaf for the pose head network as previously mentioned in Section 3. Finally, optimization was performed

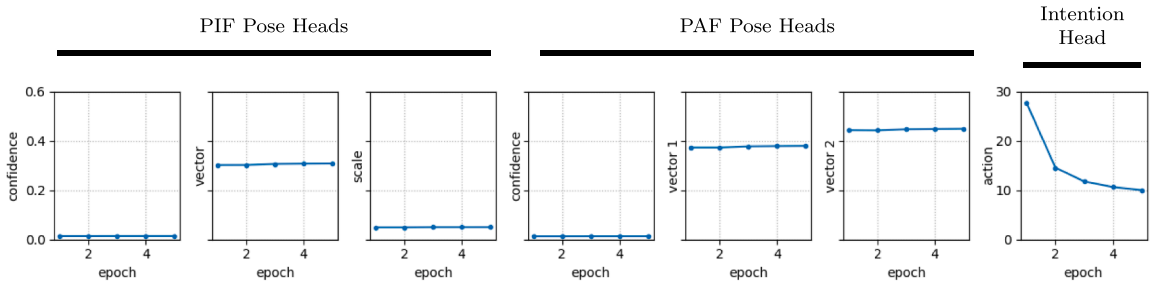
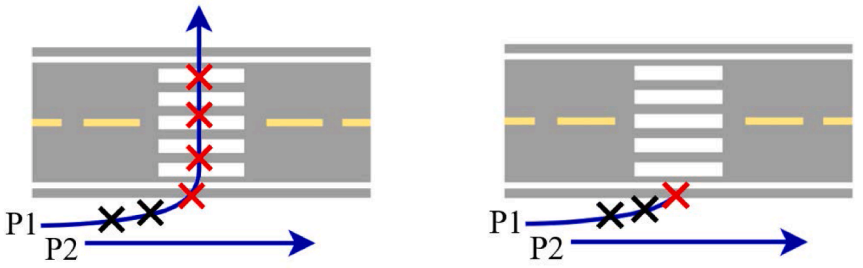


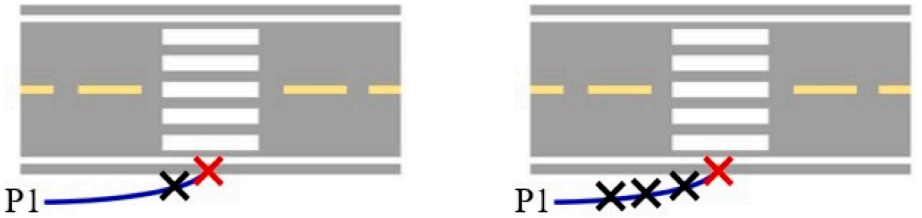
Fig. 7. Validation losses for the head networks. Note that the losses from the pose head networks are much lower than the intention head network as both the base and pose head nets were initialized with weights from PifPaf. The challenge was thus in reducing the loss coming from the intention head while maintaining losses from the pose heads with curriculum learning as described in Section 5.2.



(a) Complete ground truth used for training and also for state recognition

(b) Truncated ground truth used for intent prediction

Fig. 8. Ground truth pre-processing. The red X marks the timesteps where the pedestrian is in the state "crossing". The black X indicates the labels that have been inverted from "not crossing" to crossing. Note that the videos were recorded at a frame rate of 30 fps and that the ground truth contains the annotations for every single timestep.



(a) $t_{pi} - 1$

(b) $t_{pi} - 3$

Fig. 9. Similar to Fig. 8a, the red X marks the timesteps where the pedestrian is in the state "crossing" and the black, the timesteps that have been inverted from "not crossing" to crossing. If (a) depicts the setup for $t_{pi} - 1$ and (b) for $t_{pi} - 3$, then a model that tends to predict early would have a lower precision score for (a) than for (b).

using stochastic gradient descent with a learning rate of 10^{-5}, momentum of 0.95, COCO batch size of 8, JAAD batch size of 4 and no weight decay. The validation losses for the networks are shown in Fig. 7.

6. Experiments

6.1. Evaluation Protocol

Models tasked to predict a pedestrian's intent must be evaluated based on their ability to output the correct prediction as early as possible. Ideally, this output needs to be made before the pedestrian steps onto the road. They should also be able to recognize the state (action) a pedestrian is currently in as it would not make any sense for either an ADAS or a traffic management system for example, to keep predicting if a pedestrian is about to cross or not, even as that pedestrian is already halfway across the road. In short, these models need to (1) *predict* a pedestrian's intention as well as (2) *recognize* the state (action) that (s) he currently is in.

Table 1
Models evaluated over the entire validation set. Pedestrians that have either not been detected by our model or have no keypoints are simply assigned the label "not crossing". Each entry in the table denotes the model's precision for the ground truth with the labels T seconds before the crossing inverted (Fig. 9). Also note that our time-series variant that uses a 3D ResNet-50 does not contain the *pose network*. Finally, it can be seen that our model improves the state of the art by roughly 20%.

	State Recognition					Intention Prediction			
$T =$	0s	−1s	−2s	−3s	−4s	−1s	−2s	−3s	−4s
Ours	**81.7**	**83.6**	**83.5**	**83.0**	**82.7**	**42.6**	**46.1**	**46.3**	**46.0**
Keypoints MLP	59.2	63.4	65.0	65.2	65.2	15.2	21.2	22.4	22.9
CNN (ResNet-50) Rasouli et al. (2017)	72.3	75.0	75.3	75.1	75.1	21.9	26.8	27.5	28.6
CNN (AlexNet) Rasouli et al. (2017)	67.6	72.2	74.2	74.6	74.6	22.3	31.2	33.3	33.7
JAAD (ResNet-50) Rasouli et al. (2017)	68.3	71.5	72.4	72.3	72.3	18.3	23.7	24.9	25.6
JAAD (AlexNet) Rasouli et al. (2017)	63.0	66.1	67.1	67.1	67.0	15.0	20.0	21.1	21.3
Fork-Normalization Mordan et al. (2020)	54.8	61.3	66.1	68.6	69.3	14.9	25.5	31.1	32.5
Ours (3D ResNet-50)	**75.2**	**73.9**	73.0	72.6	72.4	**35.2**	35.8	37.8	37.5
Keypoints LSTM	55.6	65.5	71.4	74.1	74.8	26.5	33.8	34.7	34.5
CNN LSTM Saleh et al. (2019)	53.3	60.8	65.1	67.7	68.4	20.9	24.9	24.8	24.7
3D ResNet-50 Saleh et al. (2019)	56.1	63.3	67.6	70.4	70.8	18.2	26.0	27.9	28.6
3D DenseNet-121 Saleh et al. (2019)	62.4	71.1	**76.0**	**78.7**	**79.1**	25.2	**37.8**	**38.2**	**38.2**

6.1.1. Intention Prediction

We assess the model's ability to predict a pedestrian's intent by computing its precision on a variant of the validation set where the sequence of each pedestrian is truncated up until the moment (s) he begins to cross, thereby keeping the entire sequence intact for pedestrians that did not cross. We also invert the labels several timesteps before the pedestrian begins to cross from "not crossing" to "crossing" as illustrated in Fig. 8. More specifically, if each pedestrian p_i begins to cross at time t_{pi}, we modify the ground truth in the validation set such that they now begin crossing at time $t_{pi} - T$ where T is arbitrarily chosen to be 1 s in the first experiment and increases by an additional 1 s for each additional experiment, for a total of 4 experiments, *i.e.*, $t_{pi} - 1, t_{pi} - 2, t_{pi} - 3$ and $t_{pi} - 4$. This lets us study how early the model can predict the pedestrian's intention because if the precision scores for the experiment where the crossing time of $t_{pi} - 3$ is higher than the precision scores for another experiment where the crossing time is $t_{pi} - 1$, then this probably indicates that the model performs better when trained to predict earlier in time (Fig. 9). Note that the above preprocessing is only performed for the sake of evaluation. We train our model on the training set where the eventual crossers have all their labels across time set to "crossing".

We then divide each video into small video clips as input to the model using a fixed sliding window with length w and stride 1. If we denote a video sequence with t frames as $X = [x_1, x_2, \ldots x_t]$, the problem can then be described as generating for each subsequence $x_{i:i+w}$, $i \in [1, t-w]$, an intention map and a set of poses for the final item x_{i+w} where w is the window length and is equals to 1 for models that operate on a single frame, *i.e.*, ours. We do it this way to ensure that the validation set is reflective of the most important task the model needs to perform: trying to determine if a pedestrian at the sidewalk is about to cross *before* (s) he steps onto the road. This way, a low precision indicates that the model either lacks the capacity to predict the pedestrian's intention moments before the point of crossing or that the model incorrectly assigns the class crossing to a significant portion of the samples.

6.1.2. State Recognition

For recognizing a pedestrian's state, we simply evaluate the model over the entire sequence. In our experiments, we train on the complete sequences and validate on both the complete and truncated sequences. We found in our experiments that training on the truncated sequences as opposed to the complete sequences degrades the model's performance to do either tasks.

6.2. Comparison to Baselines for Intention Prediction and State Recognition

We compare our method to several baselines following the protocol described in Section 6.1. We train them all using the unaltered ground truth and validate them on both the truncated and complete sequences where we invert the labels T seconds before the crossing. We now describe each baseline.

- **CNN (**Rasouli et al., 2017**).** This method takes as input a cropped image of the pedestrian and sends it through a Convolutional Neural Network (CNN) for classification. We use AlexNet and ResNet-50.
- **JAAD (**Rasouli et al., 2017**).** This method takes as input a cropped image of the pedestrian and the entire scene and feeds them through 2 separate CNNs. The vectors produced by these networks are then concatenated and sent through a linear layer for classification. Similar to the CNN method, we use AlexNet and ResNet-50.
- **CNN-LSTM (**Saleh et al., 2019**).** This time-series model takes as input a sequence of 16 frames (0.5 s), sending each of them through a CNN for feature extraction then through the LSTM for sequence learning. We use ResNet-50 for the CNN.
- **3D CNN (**Saleh et al., 2019**).** Another time-series model that uses 3D convolutions over the 16 frames. We experiment with a 3D ResNet-50 and a 3D DenseNet-121.

Table 2
Models evaluated on the subset that were detected by PifPaf and by our model. Each entry in the table denotes the model's precision on the ground truth with the labels T seconds before the crossing inverted. (Fig. 9). Also note that our time-series variant that uses a 3D ResNet-50 does not contain the *pose network*. Finally, it can be seen that our model improves the state of the art by roughly 20%.

	State Recognition					Intention Prediction			
$T =$	0s	−1s	−2s	−3s	−4s	−1s	−2s	−3s	−4s
Ours	**82.2**	**84.0**	**83.7**	**83.1**	**82.7**	**43.4**	**47.0**	**47.2**	**46.8**
Keypoints MLP	63.2	67.1	68.7	68.8	68.7	16.0	22.5	23.7	24.1
CNN (ResNet-50) Rasouli et al. (2017)	75.0	77.5	77.7	77.4	77.5	23.4	28.6	29.3	30.6
CNN (AlexNet) Rasouli et al. (2017)	70.5	74.5	76.3	76.8	76.8	23.6	32.9	35.4	36.1
JAAD (ResNet-50) Rasouli et al. (2017)	70.2	73.3	74.0	73.9	74.0	19.6	25.1	26.3	27.2
JAAD (AlexNet) Rasouli et al. (2017)	65.9	68.9	69.8	69.8	69.6	16.5	21.5	22.9	23.2
Ours (3D ResNet-50)	**73.9**	**72.4**	71.2	70.8	70.6	**34.7**	35.0	36.9	36.6
Keypoints LSTM	59.9	69.5	75.5	78.2	78.7	28.3	36.2	37.3	37.4
CNN LSTM Saleh et al. (2019)	55.0	62.8	67.1	69.9	70.7	22.4	26.6	26.7	26.8
3D ResNet-50 Saleh et al. (2019)	58.6	66.2	70.1	73.7	74.1	20.1	29.9	32.2	33.2
3D DenseNet-121 Saleh et al. (2019)	63.0	72.1	**77.3**	**80.0**	**80.6**	26.1	**39.6**	**40.5**	**40.9**

Table 3
Models evaluated over the entire validation set. Pedestrians that have either not been detected by our model or have no keypoints are simply assigned the label "not crossing". Each entry in the table denotes the model's recall for the ground truth with the labels T seconds before the crossing inverted (Fig. 9).

	State Recognition					Intention Prediction			
$T =$	0s	−1s	−2s	−3s	−4s	−1s	−2s	−3s	−4s
Ours	71.0	65.8	61.4	59.0	57.8	26.6	19.3	16.1	14.8
Fork-Normalization Mordan et al. (2020)	**87.0**	**87.3**	**87.1**	**86.6**	**85.5**	**89.1**	**87.1**	**84.8**	**80.2**

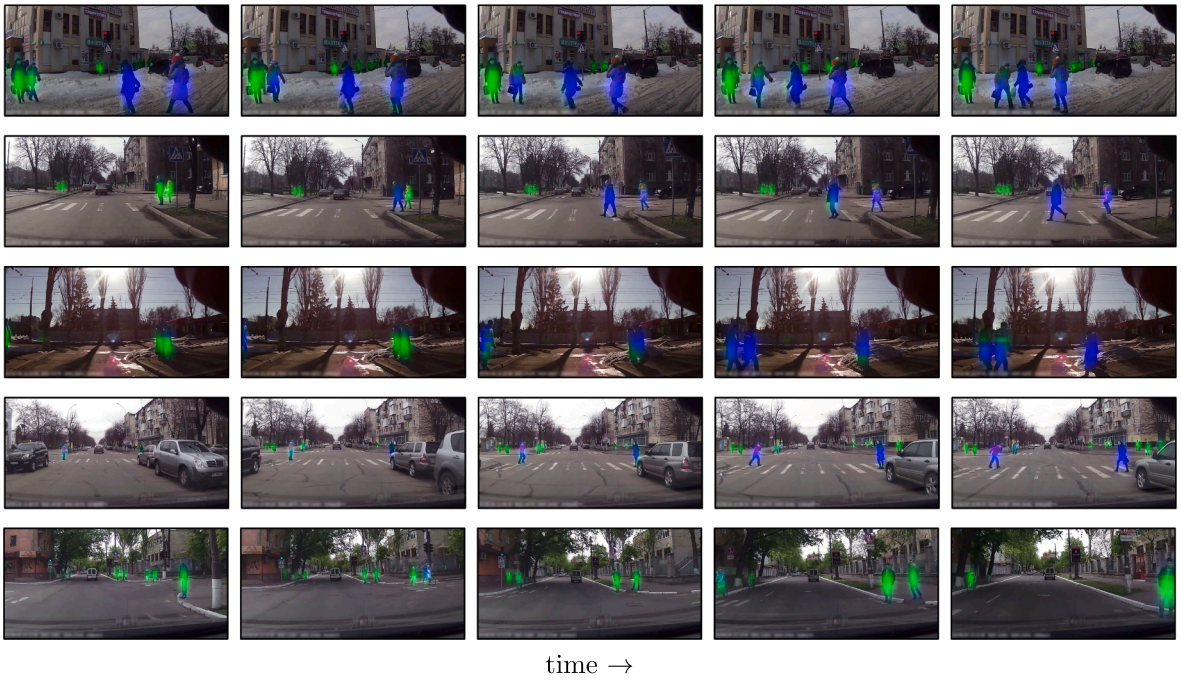


Fig. 10. Output of the intention head network. Our model can differentiate crossers and non-crossers even though they share the same pose. Pixels highlighted in blue constitute pedestrians that are about to or are currently crossing the road, and in green, pedestrians that are not about to, nor are currently crossing the road. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

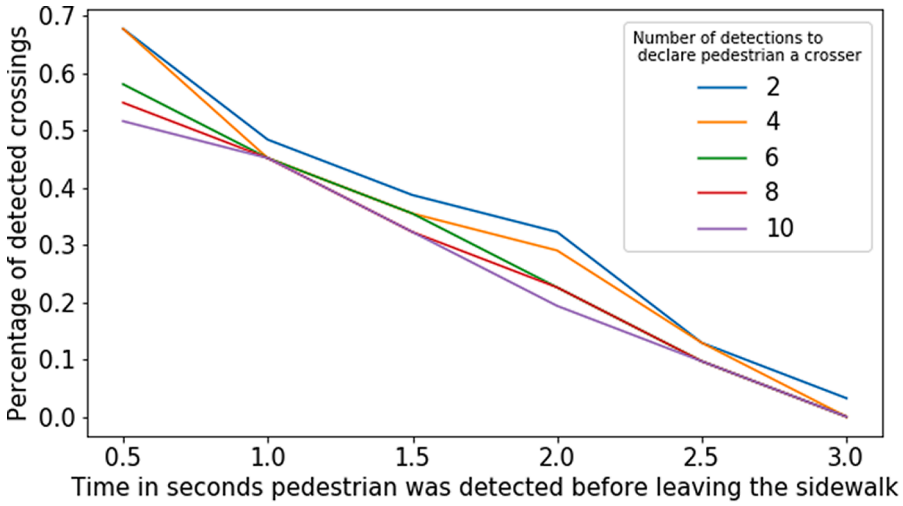


Fig. 11. The number of crossers that are correctly predicted to cross seconds before leaving the sidewalk. Each line represents the results given the minimum number of positive detections.

- **Keypoints.** Instead of operating on the images, this method takes as input the coordinates of the keypoints produced by PifPaf (17 joints x vector of size 2) and sends them through a 3-layer MLP for classification.
- **Keypoints-LSTM.** Instead of using a CNN to extract the pedestrian features, this network directly feeds a sequence of 16 keypoints at each timestep into an LSTM.
- **Fork-Normalization (**Mordan et al., 2020**).** This method takes the whole image as input and learns a heatmap corresponding to the crossing attribute.

Unlike our method and the keypoint-based ones, the baselines presented above operate directly on a crop of the pedestrian and thus can produce a classification score for the entire validation set. PifPaf produces no keypoints for pedestrians that are too far out in the scene and the intention head network of our model does not have perfect recall. When a pedestrian is not detected, we simply assign that pedestrian the label 'not crossing". We also run another evaluation but only on the subset of pedestrians that were detected both by PifPaf and our network.

Tables 1 and 2 present our quantitative results. First, it can be seen that our single-frame variant outperforms all other baselines, even against the time-series models. We believe this is because our model has access to contextual information as it models both the pedestrian and the scene during its forward pass. Intuitively, a sufficient number of convolutional layers stacked together has an effective receptive field that is similar to the size of the scene, enabling the model to look at the pedestrian's pose and his surroundings when generating the activity map. This lends our model an advantage over the other models that operate only the cropped pedestrian images. It is difficult to ascertain the state the pedestrian is in simply through his pose as evidenced by the precision scores attained by the other models as there are a notable number of pedestrians having a similar orientation but with a different label. For example, a pedestrian who has completed the crossing would be in the same orientation as another pedestrian who is currently crossing. Although JAAD jointly models both pedestrian and scene, it does this in a two-stream approach, sending the scene and a crop of the pedestrian through two separate CNNs, thereby losing spatial context of what the pedestrian's surrounding is. Second, it can be observed that the temporal models that contain more parameters tend to perform worse than their single frame counterparts. We suspect this to be the result of an ill posed problem - trying to predict the pedestrian's intent purely via a sequence of crops - made even more severe with an overfitted model. Finally, we point out that our bottom-up temporal model performs worse than the single frame variant due to a suboptimal setup. We had to resize the sequence of frames to a resolution of (189,480) and train it with batch size of 1 in order to keep the memory footprint below 11 GB on the GTX 1080 Ti.

Table 3 shows recall values for our method and Fork-Normalization (Mordan et al., 2020) competing approach. Although the latter has better recall scores, we have better precision scores, which are usually exclusive performance. This means that no method is dominating the other one. When applying to real problems, precision and recall values should be balanced and tailored for any specific need.

Fig. 10 shows the output of the *intention network* where the prediction has been color coded to represent the intention: green for "not crossing" and blue for crossing. It can be observed from the figure that our model is able to correctly locate the pedestrian and predict his intention at the sidewalk: it fills pixels in the image with the color blue for where it thinks are pedestrians that are currently crossing or are about to cross the road, and the color green for bystanders. Visually, there are no false positives. More importantly, it is able to predict the pedestrian's intention early enough, before (s) he steps onto the road. This can be fundamental in developing safe self-driving cars. The figures also show that we are able to distinguish between crossers and non-crossers even though they share the same orientation. The results here lend to our hypothesis that the model looks at both the pedestrian's pose and his surroundings during the forward pass.

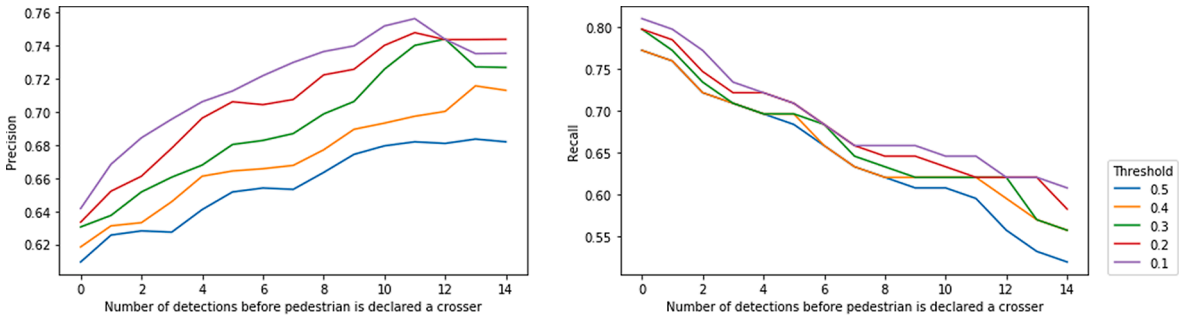


Fig. 12. Precision and recall scores using all unique pedestrians. Each detection is declared a positive (crossing) if its IoU threshold is above some value and each pedestrian, a crosser if the number of detections meet the requirement.

6.3. How early can we predict a crossing?

The ground truth of a pedestrian switches from "not crossing" to "crossing" the instant (s) he leaves the sidewalk. If we label this time instant as *t*, then a model that can predict a crossing intention early must be able to output the label "crossing" *C* seconds before *t*. We study how early the model is able to anticipate a crosser. For this experiment, we ignore all bystanders and look at only each *unique* crosser and declare an individual to be a crosser only if (s) he is positively detected by the model *N* times where *N* is varied from 2 to 10. It can be seen from Fig. 11 that most of the crossers are detected between 0.5 to 1 s before they cross followed by a gradual decrease and that naturally, the time taken to detect a crossing increases with *N*. Note that this does not indicate the inefficacy of our model or that it can only predict a crossing 0.5 s before the actual crossing takes place. The results occur as so simply because the dataset is highly skewed in the first place. Recall from Fig. 4a that a large majority of the pedestrians in the entire train and test set begin crossing only less than a second after appearing in the field of view. As such, taking that into consideration, it would be reasonable to say that our model is performing well.

6.4. Precision and Recall for all unique pedestrians.

We next study the stability of our model by computing its precision and recall scores using *all unique pedestrians* and not all pedestrians as was done in Tables 1 and 2. We vary the IoU threshold required to declare a detection a positive as well as the number of detections needed to declare the pedestrian a crosser. This is primarily due to the fact that our model is trained to output gaussians. The results are presented in Fig. 12. It can be seen that our model naturally performs worse given stricter requirements. This can be primarily attributed to the fact that our model outputs gaussians and not bounding boxes where the intensities are only stronger at the centers. As we require more detections for a pedestrian to be consider a crosser, the precision will naturally increase as we have more stable positive predictions, but the recall will decrease because detections happening when the threshold is not reached will be counted as false negatives. Also note that the numbers are lower than what was presented in Tables 1 and 2 because the result in said tables were computed using every pedestrian whereas the ones in Fig. 12 are computed using every unique pedestrian. Small changes in classification metrics such as true positives, false positives etc., would thus have a greater impact on both precision and recall.

6.5. Pedestrian Detection

Although the focus of this paper is on intention prediction, we also show its performance on detection because as previously mentioned, both the *intention* and the *pose networks* can act as generic pedestrian detectors. The outputs are shown in Fig. 13. Our *pose network* performs well at localizing the human parts. There are no false positives in the outputs and keypoints that are critical such as the eyes have been detected by model. However, we also see here that the intention head-net outperforms the pose head-net on the task of detection: it is able to locate pedestrians that are farther out the scene. We believe this to be the consequence of not having the ground truth keypoints for the JAAD pedestrians. Although the model was jointly trained to minimize the errors across all heads, losses arising from keypoint false negatives come only from the COCO dataset, the images of which might not be representative of those in JAAD, *i.e.*, very different human sizes. Regardless, we see that the missing detections are for pedestrians that are too far off to be in any real danger from the vehicle. We have summarized the bounding box recall rates for both head networks in Table 4.

It should be noted here that both head nets in our model have not been trained to regress on the pedestrian ground truth boxes. The pose head network generates the skeleton whereas the intention head network computes the probability that each pixel makes up a person who is either crossing or not. We therefore evaluate the recall of each detector independently. For the boxes generated via keypoints, we simply declare it a true positive if its IoU with the ground truth box is non-zero. The IoU of a predicted and ground truth bounding box is defined as:

$$\text{IoU} = \frac{\text{AreaofOverlap}}{\text{AreaofUnion}} \tag{7}$$

(a) Intention Head Network (b) Pose Head Network

Fig. 13. Output of the intention and pose head networks. Notice how the intention head network is able to localize pedestrians that are farther out in the scene. Pixels highlighted in blue constitute pedestrians that are about to or are currently crossing the road, and in green, pedestrians that are not about to, nor are currently crossing the road. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Pedestrian detection recall on the JAAD dataset for each head network. R^S, R^M and R^L denote the recall scores for pedestrians of heights below 75 pixels, between 75 to 175 pixels and above 175 pixels respectively. Refer to Fig. 6 to get a rough gauge of the heights for an input image of (540,960). Finally, it can be seen that the recall of the *pose* degrades by less than 1% when adding the *intention network* while the recall of the *intention network* improves by 7% with the inclusion of the *pose network*.

	R	$R^{h<75px}$	$R^{75px\leqslant h<175px}$	$R^{h\geqslant 175px}$
Pose Network (Multitask)	61.3	**44.4**	92.3	**97.7**
Pose Network (Single task)	**61.7**	44.3	**94.1**	97.2
Intention Network (Multitask)	**89.5**	**80.0**	**91.9**	98.9
Intention Network (Single task)	82.3	68.6	91.8	98.9

Table 5
Running times of the various methods. Note that our running time is per-image and includes detection whereas the running times of the other methods excludes detection (which is an additional 0.2 s for all the keypoint-based methods and even more for the others). They would thus scale according to the number of detections returned.

	Running time in seconds	
	Per image	Per batch of 16 detections
Ours	0.217	-
Keypoints MLP	-	0.001
CNN (AlexNet)	-	0.371
JAAD (AlexNet)	-	1.225
Keypoints LSTM	-	0.003
CNN LSTM	-	0.524
3D ResNet-50	-	0.874
3D DenseNet-121	-	0.712

Table 6
Comparing our model to PifPaf on the COCO dataset for the task of keypoint detection. Note that PifPaf is equivalent to our model but without the intention head network. The results are based on the evaluation described in Kreiss et al. (2019) for low resolution images with the long side equal to 321 pixels. The table shows that the addition of the intention head network reduces the performance of our model by approximately 1%.

	AP	$AP^{0.50}$	$AP^{0.75}$	AP^{M}	AP^{L}	AR	$AR^{0.50}$	$AR^{0.75}$	AR^{M}	AR
Pose Network (Multitask)	51.6	76.2	55.1	37.9	70.4	56.0	78.1	59.1	41.4	76.1
Pose Network (Single task)	**52.7**	**76.9**	**56.5**	**39.3**	**71.2**	**57.6**	**79.8**	**61.1**	**43.2**	**77.4**

Table 7
Comparing our model to its variant without the pose head-net on the JAAD dataset for the task of state recognition and intention prediction. The models were evaluated over the entire validation set similar to Table 1

	State Recognition					Intention Prediction			
	0s	−1s	−2s	−3s	−4s	−1s	−2s	−3s	−4s
Intention Network (Multitask)	**81.7**	**83.6**	**83.5**	**83.0**	**82.7**	42.6	46.1	46.3	46.0
Intention Network (Single task)	76.1	78.5	78.4	77.9	77.5	**49.6**	**51.6**	**51.5**	**51.1**

which returns a score close to 1 for bounding box predictions that closely matches the ground truth and conversely, a score close to 0 for inaccurate predictions with very little overlap. For the activity map, we threshold the output at 0.2 and declare a ground-truth box to have been detected if the sum of the activity-map pixels within it are non-zero. Based on how little false positives there are, we believe that this way of computing the recall scores to not be unreasonable.

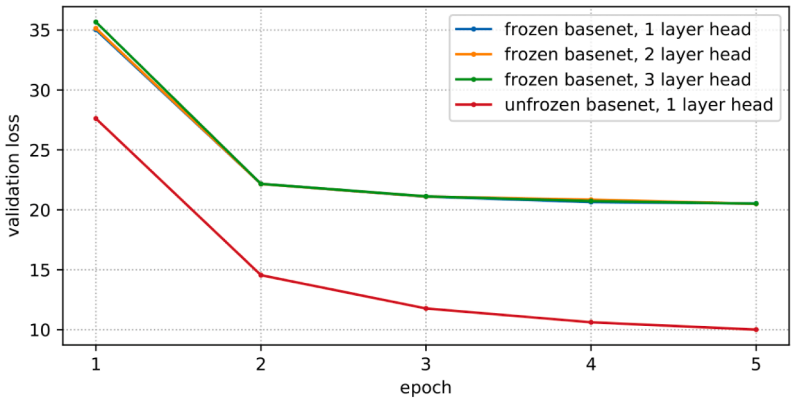
6.6. Running Time

Lastly, we present the running time of the various methods in Table 5. Note that our method processes the entire image unlike the baselines that operate on a crop of the pedestrian. Our method, running at approximately 0.2s per image is only outperformed by the keypoint based models. Furthermore, the time taken for the baselines would scale in proportion to the number of detections returned. Our method in contrast scales lightly to the number of keypoints returned.

6.7. Ablation studies

6.7.1. Comparison to the single head variants.

We study the effects various design decisions have on our model. We first look at the influence one head has on another *i.e.,* if the inclusion of the pose head network has any effect on the performance of the intention head network and vice versa. To do this, we trained a single-head (single task) variant of our network and compared their performance on the COCO dataset for keypoint detection and on JAAD for intention prediction and state recognition against our original multitask model. The results presented in Tables 6 and 7 show that the introduction of a second head can negatively impact the model's performance: we perform better on the JAAD dataset but worse on COCO. It is then interesting to note that learning the model on a related task has not boosted its performance on either, unlike the results presented in the literature. We believe this is because our model was not trained in the same fashion as is typically done in the literature, using a single dataset with multiple annotations for the different tasks. Our problem required the use of multiple datasets through data slicing.



(a) Validation losses for the models with a frozen and unfrozen basenet.

(b) Sample output of a frozen base-net with a 3-layer head-net. Pixels highlighted in green constitute pedestrians that are about to or are currently crossing the road, and in green, pedestrians that are not about to, nor are currently crossing the road.

(c) Another sample output of a frozen base-net with a 3-layer head-net.

Fig. 14. The plot in (a) shows that the unfrozen base-net can achieve a much lower loss even with only a 1-layer head-net. The sample outputs in figures (b) and (c) indicate that the model with a frozen base-net is unable to differentiate between crossers and non-crossers as the pedestrians are being painted with both labels.

6.7.2. Freezing the base network

Next, we studied the effects of initializing the base network with weights from PifPaf and freezing them during training. It can be seen in Fig. 14 that this approach is not a good idea: the validation loss converges at a higher value even when adding additional layers to the intention head net. The figures also show that networks trained in this way are able to localize the pedestrians but not able to assign the correct label to them. This shows that the most important features are learned in the shared base network and although these features pertain to pedestrians, they are not sufficient for the intention head to perform its task.

6.8. Saliency Visualization

We visualize pixels that positively affect the output class through a method called guided back-propagation (Springenberg et al., 2014). This technique of analysis works by computing the derivative of the output pixel with respect to the input image, but suppresses gradients through units that do not have a positive contribution or activation. This derivative image tells us how much a small change

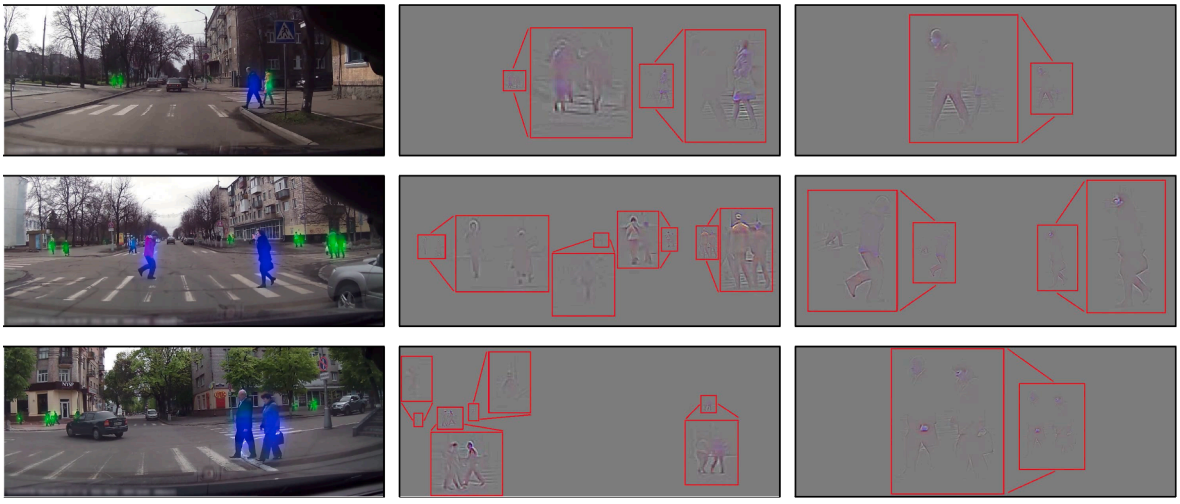


Fig. 15. The figures in column (a) show the output of the model on 3 separate frames where pixels highlighted in blue correspond to predcitions of pedestrians that are about to or are currently crossing the road, and in green, to pedestrians that are not about to, nor are currently crossing the road. The figures in columns (b) and (c) illustrate the output of the guided backpropagation algorithm on the 3 frames for non-crossers and crossers respectively with pixels that are brighter or darker than the color of grayscale (*i.e.*, the l_1 distance from [127,127,127]) having a greater impact on the output. The red boxes show a magnified view of each pedestrian or group of pedestrians.

to each pixel would affect the prediction, with pixels that are brighter or darker than the color of grayscale (*i.e.*, the l_1 distance from [127,127,127]) having a greater impact on the output. It can thus be thought of as a saliency map that tells us how important each pixel is for classification. We display some results in Fig. 15.

On first inspection, it can be observed that the architecture is using information pertaining to the pedestrians due to the silhouettes that can be seen on the image. This suggests that our model makes its prediction based on the pedestrian's pose. Also note how the textures on the clothing play little to no part in classification. Although they are not exactly gray, their intensities are neither as bright nor as dark as the silhouettes. The areas around heads and legs have a bigger impact on the predictions. This makes sense as a person's attire should have no relation to his intention. The gradient image also indicates that the regions surrounding the pedestrians have a positive impact on classification as they are not 'greyed-out'. Lastly, we can see that the roads are not a strong indicator of the pedestrian's state. It will thus be useful to investigate if learning the model for semantic segmentation, *i.e.*, adding an additional task head, will result in any change.

7. Conclusions

We have presented a method for joint detection, pose estimation and intention prediction of pedestrians. The model is a 5-block ResNet-50 base network followed by two parallel single-layered convolutional task heads for pose estimation and intention prediction and has a frame rate of 5 fps for an image resolution of (378,960) on a GTX 1080 Ti. Our task heads are single-layered 1x1 convolutional layers that run in parallel and thus do not add much overhead. Experiments on the JAAD dataset clearly demonstrate the benefits of our architecture, improving the precision scores of the state-of-the-art for the task of intention prediction by approximately 20% (Tables 1 and 2). However, we also noted how appending additional task heads does not always boost the performance of the model on each individual task when training over multiple datasets using data slicing. For instance, we saw in Tables 4 and 6 that the *pose network* suffers a 1% drop in its recall for pedestrian detection and a 1% drop in its average precision for keypoint detection when going from single task to multitask but the converse for the *intention network*, that it gains a 7% increase in its detection recall when going from single to multitask. We believe this to simply be attributed to unoptimal data pre-processing *e.g.*, different human sizes as mentioned in Section 6.5 which we plan to resolve in the future. Lastly, our saliency maps in Section 6.8 show that our fully convolutional architecture encodes the pedestrian's pose as well as his surroundings during the forward pass, giving it the edge needed to outperform all other models that operate directly on a crop of the pedestrian.

As future work, one can investigate the amount of information the base network can hold by adding in more single-layer heads to perform a variety of other tasks such as semantic segmentation and depth estimation and to record changes in the model's performance. Finally, we would also like to mention that the JAAD dataset does not annotate how far the pedestrians are from the sidewalk and that there are a number of scenes in the dataset where the pedestrian stands at the edge of the sidewalk for some seconds before crossing. As such, being able to predict early for such scenarios is not informative of the performance of the model. A better problem would be to predict the pedestrian's intention as far from the edge of the sidewalk as possible instead of trying to predict as early as possible before (s) he leaves the sidewalk. It would thus be very useful to augment the existing dataset with annotations that state how far the crossers are from the edge of the sidewalk.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.trc.2021.103259.

References

Agahian, S., Negin, F., Köse, C., 2020. An efficient human action recognition framework with pose-based spatiotemporal features. Engineering Science and Technology, an International Journal 23 (1), 196–203.

Alahi, A., Bierlaire, M., Vandergheynst, P., 2014. Robust real-time pedestrians detection in urban environments with low-resolution cameras. Transportation Research Part C: Emerging Technologies.

A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, S. Savarese, Social lstm: Human trajectory prediction in crowded spaces, in: Computer Vision and Pattern Recognition, 2016.

A. Alahi, V. Ramanathan, K. Goel, A. Robicquet, A.A. Sadeghian, L. Fei-Fei, S. Savarese, Learning to predict human behavior in crowded scenes, in: Group and Crowd Behavior for Computer Vision, Elsevier, 2017, pp. 183–207.

Amirian, J., Hayet, J.-B., Pettre, J., 2019. Social ways: Learning multi-modal distributions of pedestrian trajectories with gans. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.

S.M. Azar, M.G. Atigh, A. Nickabadi, A. Alahi, Convolutional relational machine for group activity recognition, in: Computer Vision and Pattern Recognition, 2019.

F. Bartoli, G. Lisanti, L. Ballan, A.D. Bimbo, Context-aware trajectory prediction, in: arXiv, 2017.

Batkovic, I., Zanon, M., Lubbe, N., Falcone, P., 2018. A computationally efficient model for pedestrian motion prediction. European Control Conference (ECC) 2018, 374–379.

Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: International Conference on Machine Learning, 2009.

Bertoni, L., Kreiss, S., Alahi, A., 2019. Monoloco: Monocular 3d pedestrian localization and uncertainty estimation, in. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6861–6871.

L. Bertoni, S. Kreiss, A. Alahi, Perceiving humans: from monocular 3d localization to social distancing, IEEE Transactions on Intelligent Transportation Systems.

M. Bieshaar, G. Reitberger, S. Zernetsch, B. Sick, E. Fuchs, K. Doll, Detecting intentions of vulnerable road users based on collective intelligence, arXiv preprint arXiv: 1809.03916.

Bonnin, S., Weisswange, T.H., Kummert, F., Schmuedderich, J., 2014. Pedestrian crossing prediction using multiple context-based models. In: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 378–385.

Boudet, L., Midenet, S., 2009. Pedestrian crossing detection based on evidential fusion of video-sensors. Transportation Research Part C: Emerging Technologies.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., Sheikh, Y., 2019. Openpose: realtime multi-person 2d pose estimation using part affinity fields. IEEE transactions on pattern analysis and machine intelligence 43 (1), 172–186.

Caruana, R., 1997. sMultitask learning. Machine Learning 28 (1), 41–75. https://doi.org/10.1023/A:1007379606734.

Chaabane, M., Trabelsi, A., Blanchard, N., Beveridge, R., 2020. Looking ahead: Anticipating pedestrians crossing with future frames prediction. In: The IEEE Winter Conference on Applications of Computer Vision, pp. 2297–2306.

Cheng, B., Xu, X., Zeng, Y., Ren, J., Jung, S., 2018. Pedestrian trajectory prediction via the social-grid lstm model. The Journal of Engineering 2018 (16), 1468–1474.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition.

P. Dollar, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: An evaluation of the state of the art, in: Pattern Analysis and Machine Intelligence, 2012.

Dominguez-Sanchez, A., Cazorla, M., Orts-Escolano, S., 2017. Pedestrian movement direction recognition using convolutional neural networks. IEEE transactions on intelligent transportation systems 18 (12), 3540–3548.

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

Fang, Z., Vázquez, D., López, A.M., 2017. On-board detection of pedestrian intentions. Sensors 17 (10), 2193.

Fang, H.-S., Xie, S., Tai, Y.-W., Lu, C., 2017. Rmpe: Regional multi-person pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2334–2343.

Z. Fang, A.M. López, Is the pedestrian going to cross? answering by 2d pose estimation, in: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2018, pp. 1271–1276.

T. Fernandoa, S. Denmana, S. Sridharana, C. Fookesa, Soft+hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection, in: arXiv, 2017.

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

A. Graves, M.G. Bellemare, J. Menick, R. Munos, K. Kavukcuoglu, Automated curriculum learning for neural networks, in: Machine Learning Research, 2017.

Gujjar, P., Vaughan, R., 2019. Classifying pedestrian actions in advance using predicted video of urban driving scenes. In: International Conference on Robotics and Automation.

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

Guo, M., Haque, A., Huang, D.-A., Yeung, S., Fei-Fei, L., 2018. Dynamic task prioritization for multitask learning. In: European Conference on Computer Vision.

Guo, M., Haque, A., Huang, D.-A., Yeung, S., Fei-Fei, L., 2018. Dynamic task prioritization for multitask learning. In: European Conference on Computer Vision.

A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, A. Alahi, Social gan: Socially acceptable trajectories with generative adversarial networks, in: Computer Vision and Pattern Recognition, 2018.

Hashimoto, Y., Yanlei, G., Hsu, L., Shunsuke, K., 2015. A probabilistic model for the estimation of pedestrian crossing behavior at signalized intersections. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 1520–1526.

Hashimoto, Y., Gu, Y., Hsu, L., Kamijo, S., 2015. Probability estimation for pedestrian crossing intention at signalized crosswalks. IEEE International Conference on Vehicular Electronics and Safety (ICVES) 2015, 114–119.

K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask r-cnn, in: IEEE International Conference on Computer Vision, 2017.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. In: Neural Comput..

Hoy, M., Tu, Z., Dang, K., Dauwels, J., 2018. Learning to predict pedestrian intention via variational tracking networks. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 3132–3137.

Jeong, M., Ko, B.C., Nam, J., 2017. Early detection of sudden pedestrian crossing for safe driving during summer nights. IEEE Trans. Circuits Syst. Video Technol. 27 (6), 1368–1380.

A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in: Computer Vision and Pattern Recognition, 2014.

Keller, C.G., Gavrila, D.M., 2013. Will the pedestrian cross? a study on pedestrian path prediction. IEEE Trans. Intell. Transp. Syst. 15 (2), 494–506.

A. Kendall, Y. Gal, R. Cipolla, Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, in: Computer Vision and Pattern Recognition, 2018.

Köhler, S., Goldhammer, M., Zindler, K., Doll, K., Dietmeyer, K., 2015. Stereo-vision-based pedestrian's intention detection in a moving vehicle. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 2317–2322.

I. Kokkinos, Ubernet - training a 'universal' convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory, in: Computer Vision and Pattern Recognition, 2017.
Kooij, J.F.P., Schneider, N., Flohr, F., Gavrila, D.M., 2014. Context-based pedestrian path prediction. In: European Conference on Computer Vision. Springer, pp. 618–633.
Koschi, M., Pek, C., Beikirch, M., Althoff, M., 2018. Set-based prediction of pedestrians in urban environments considering formalized traffic rules. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 2704–2711.
Kreiss, S., Bertoni, L., Alahi, A., 2019. Pifpaf: Composite fields for human pose estimation, in. In: Computer Vision and Pattern Recognition.
N. Lee, W. Choi, P. Vernaza, C.B. Choy, P.H.S. Torr, M. Chandraker, Desire: Distant future prediction in dynamic scenes with interacting agents, in: Computer Vision and Pattern Recognition, 2017.
Liu, B., Adeli, E., Cao, Z., Lee, K., Shenoi, A., Gaidon, A., Niebles, J.C., 2020. Spatiotemporal relationship reasoning for pedestrian intent prediction. IEEE Robotics and Automation Letters 5 (2), 3485–3492.
Luvizon, D.C., Picard, D., Tabia, H., 2018. 2d/3d pose estimation and action recognition using multitask deep learning, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5137–5146.
Mínguez, R.Q., Alonso, I.P., Fernández-Llorca, D., Sotelo, M.Á., 2018. Pedestrian path, pose, and intention prediction through gaussian process dynamical models and pedestrian activity recognition. IEEE Trans. Intell. Transp. Syst. 20 (5), 1803–1814.
A. Møgelmose, M.M. Trivedi, T.B. Moeslund, Trajectory analysis and prediction for improved pedestrian safety: Integrated framework and evaluations, in: 2015 IEEE Intelligent Vehicles Symposium (IV), 2015, pp. 330–335.
T. Mordan, M. Cord, P. Pérez, A. Alahi, Detecting 32 pedestrian attributes for autonomous vehicles, arXiv preprint arXiv:2012.02647.
Nakamura, K., Yeung, S., Alahi, A., Fei-Fei, L., 2017. Jointly learning energy expenditures and activities using egocentric multimodal signals, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1868–1877.
S. Neogi, M. Hoy, W. Chaoqun, J. Dauwels, Context based pedestrian intention prediction using factored latent dynamic conditional random fields, in: SSCI, 2017.
S. Neogi, M. Hoy, W. Chaoqun, J. Dauwels, Context based pedestrian intention prediction using factored latent dynamic conditional random fields, in: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2017, pp. 1–8.
Papandreou, G., Zhu, T., Chen, L.-C., Gidaris, S., Tompson, J., Murphy, K., 2018. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In: European Conference on Computer Vision.
Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P.V., Schiele, B., 2016. Deepcut: Joint subset partition and labeling for multi person pose estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4929–4937.
Quintero, R., Almeida, J., Llorca, D.F., Sotelo, M.A., 2014. Pedestrian path prediction using body language traits. IEEE Intelligent Vehicles Symposium Proceedings 2014, 317–323.
Quintero, R., Parra, I., Llorca, D.F., Sotelo, M.A., 2014. Pedestrian path prediction based on body language and action classification. In: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 679–684.
Quintero, R., Parra, I., Llorca, D.F., Sotelo, M.A., 2015. Pedestrian intention and pose prediction through dynamical models and behaviour classification. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 83–88.
Quintero, R., Parra, I., Llorca, D.F., Sotelo, M., 2015. Pedestrian intention and pose prediction through dynamical models and behaviour classification. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems. IEEE, pp. 83–88.
Quintero, R., Parra, I., Lorenzo, J., Fernández-Llorca, D., Sotelo, M.A., 2017. Pedestrian intention recognition by means of a hidden markov model and body language. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–7.
R. Quintero, I. Parra, J. Lorenzo, D. Fernández-Llorca, M. Sotelo, Pedestrian intention recognition by means of a hidden markov model and body language, in: 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), IEEE, 2017, pp. 1–7.
A. Rasouli, I. Kotseruba, J.K. Tsotsos, Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior, in: International Conference on Computer Vision, 2017.
Rehder, E., Wirth, F., Lauer, M., Stiller, C., 2018. Pedestrian prediction by planning using deep neural networks. IEEE International Conference on Robotics and Automation (ICRA) 2018, 5903–5908.
Sadeghian, A., Kosaraju, V., Sadeghian, A., Hirose, N., Rezatofighi, S.H., Savarese, S., 2018. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In: Computer Vision and Pattern Recognition.
Saleh, K., Hossny, M., Nahavandi, S., 2017. Intent prediction of vulnerable road users from motion trajectories using stacked lstm network. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 327–332.
Saleh, K., Hossny, M., Nahavandi, S., 2018. Long-term recurrent predictive model for intent prediction of pedestrians via inverse reinforcement learning. Digital Image Computing: Techniques and Applications (DICTA) 2018, 1–8.
Saleh, K., Hossny, M., Nahavandi, S., 2019. Real-time intent prediction of pedestrians for autonomous ground vehicles via spatio-temporal densenet. In: International Conference on Robotics and Automation.
Sarkar, A., Czarnecki, K., Angus, M., Li, C., Waslander, S., 2017. Trajectory prediction of traffic agents at urban intersections through learned interactions. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–8.
Schulz, A.T., Stiefelhagen, R., 2015. A controlled interactive multiple model filter for combined pedestrian intention recognition and path prediction. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pp. 173–178.
Seer, S., Brändle, N., Ratti, C., 2012. Kinects and human kinetics: A new approach for studying pedestrian behavior. Transportation Research Part C: Emerging Technologies.
A. Shahroudy, J. Liu, T.-T. Ng, G. Wang, Ntu rgb+d: A large scale dataset for 3d human activity analysis, in: Computer Vision and Pattern Recognition, 2016.
W. Shi, J. Caballero, F. Huszár, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, in: Computer Vision and Pattern Recognition, 2016.
J.T. Springenberg, A. Dosovitskiy, T. Brox, M. Riedmiller, Striving for simplicity: The all convolutional net, in: ArXiv, 2014.
Sun, L., Jia, K., Yeung, D.-Y., Shi, B.E., 2015. Human action recognition using factorized spatio-temporal convolutional networks. In: IEEE International Conference on Computer Vision.
Tian, Y., Luo, P., Wang, X., Tang, X., 2015. Pedestrian detection aided by deep learning semantic tasks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5079–5087.
Völz, B., Mielenz, H., Gilitschenski, I., Siegwart, R., Nieto, J., 2019. Inferring pedestrian motions at urban crosswalks. IEEE Trans. Intell. Transp. Syst. 20 (2), 544–555.
Z. Wang, N. Papanikolopoulos, Estimating pedestrian crossing states based on single 2d body pose, in: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), Vol. 2, 2020.
Wang, C., Wang, Y., Yuille, A.L., 2013. An approach to pose-based action recognition, in. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 915–922.
Zhao, J., Li, Y., Xu, H., Liu, H., 2019. Probabilistic prediction of pedestrian crossing intention using roadside lidar data. IEEE Access 7, 93781–93790.