2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Estimating Pedestrian Crossing States Based on Single 2D Body Pose

Zixing Wang[1] and Nikolaos Papanikolopoulos[2]

Abstract— **The Crossing or Not-Crossing (C/NC) problem is important to autonomous vehicles (AVs) for safe vehicle/pedestrian interactions. However, this problem setup often ignores pedestrians walking along the direction of the vehicles' movement (LONG). To enhance the AVs' awareness of pedestrian behavior, we make the first step towards extending the C/NC to the C/NC/LONG problem and recognize them based on single body pose. In contrast, previous C/NC state classifiers depend on multiple poses or contextual information. Our proposed shallow neural network classifier aims to recognize these three states swiftly. We tested it on the JAAD dataset and reported an average 81.23% accuracy.**

I. Introduction

In 2017, there were 5,977 pedestrian deaths in USA that were reported by the United States National Highway Traffic Safety Administration (NHTSA). Among these fatalities, 5,890 pedestrians were killed by single or multiple motor vehicles. When the accidents happened, 84.4% (4,529) of victims were struck by the front of the vehicles [1]. According to these data, we believe that an approach to avoid hitting pedestrians by the front of vehicles, which accounts for the highest proportion of fatal vehicle-human accidents, is critical for AVs.

Simply obeying traffic rules is not enough for AVs to avoid vehicle-human accidents. Two mainstream classes of methods to remedy this problem include prediction-based and estimation-based methods. However, these methods have some flaws, which may impact their potential for deployment.

Previous prediction-based research computes the pedestrians' incoming moving trajectory or predicts their crossing intention based on previous trajectories [2] or velocities [3]. More recent research takes contextual elements [4] such as weather conditions, time in the day, etc., into consideration to improve performance. However, predicting pedestrians' intention remains a challenging problem due to their arbitrary upcoming motion [5]. They can decide to change moving direction, stop crossing, etc., within a second. Moreover, they are influenced by multiple internal and external factors. For instance, arbitrary actions such as failure to yield right of way, improper crossing of the roadway or the intersection, darting or running into the road and failure to obey traffic signs, signals, or officer commands led to 1,788 (29.9%), 1,268 (21.2%), 592 (9.9%), and 266 (4.5%) fatalities [1] in 2017. Thus, it is very difficult to predict pedestrian intention based on prior data (trajectory, velocity, etc.) or contextual information, which is unquantifiable and hard to identify, within a short time window. In addition, most of these approaches need significant computational power and work offline, which limit their deployment on AVs as a real-time safety mechanism.

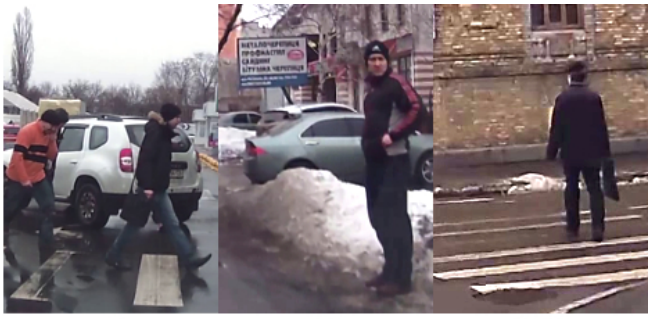

Fig. 1: Our focus: recognize crossing (C), not-crossing (NC) and walking along the direction of vehicles' movement (LONG) based on single body pose within a short time. Note that previous related works categorize both not-crossing and parallel walking as NC.

Earlier estimation-based research [6]–[8] simply classifies pedestrians' action to cross and not-cross (C/NC). Crossing indicates pedestrians' movement that is lateral to vehicle, and the rest belongs to not-crossing. This is a dangerous classification, especially when the vehicle turns at an intersection. For instance, as Fig. 2 shows, before the vehicle turns right, two pedestrians bounded by an orange box will be classified as NC. However, in the middle of turning, they would be classified in a C state by using the same criteria. This ambiguity can lead to severe traffic accidents. (We extend the categories from C/NC to C/NC/LONG in this work, which allows AVs to sense pedestrian behavior in both parallel and lateral directions to avoid this dilemma.) In addition, most of these works make the estimation based on the appearance of pedestrians. So, in low-light or severe weather conditions, the performance will drastically deteriorate.

We believe that the state estimation is more promising and applicable than intention prediction. Our reasons are the following. Firstly, when vehicles move on roads, information about the surroundings including pedestrians updates very frequently and includes a lot of recent content. State estimation can capture this varying information and can guide the decision making more effectively. Moreover, intention prediction is using a lot of subtle cues that often could not

[1]Zixing Wang is with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55414, USA. wang7923@umn.edu

[2]Nikolaos Papanikolopoulos is with the Faculty of the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55414, USA. npapas@cs.umn.edu

Fig. 2: The pedestrians inside the orange bounding boxes are classified as NC and C in the left and right figures.

capture the real actions or intentions of pedestrians. Finally, intention is often inferred from limited in time information that may not be representative of what the actual state of the pedestrian movement is.

To make AVs interact with pedestrians more safely and address the problems of the estimation type research (stated before), we propose a neural network classifier performing the C/NC/LONG task based on a single 2D body pose. The idea is illustrated in Fig. 1. Since our network works with AVs and is designed to avoid collisions, it is very sensitive to running speed and computational resources. In fact, our 2D pose contains 36 floating point numbers. Due to these two reasons, the network is very shallow and has a limited number of parameters. By extending the previous approach to C/NC/LONG problems, the AVs will have better awareness of pedestrians' moving/crossing states. Moreover, having more classes enables a larger variety of vehicle control and evasive actions. Dilemmas such as what is illustrated in Fig. 2 could be addressed in a more comprehensive way. For instance, as pedestrians in the orange bounding boxes are labeled as LONG, the AVs can stop or decelerate immediately to avoid collision once a right turn control is initiated. However, with NC labels the vehicle will not stop until the pedestrians move almost perpendicular to the front of the vehicle if there is no other pre-collision sensing device. Moreover, we replace the pedestrians' appearance by the 2D body pose since we believe it is a strong indicator of pedestrians motion state and current state-of-the-art algorithms and sensors integrated on AVs are able to provide reliable pedestrian detection, tracking, and pose estimation [9], [10] data in various weather and light conditions to enable all-time onboard operation. We believe the crossing state can provide more detailed references for AVs to enable more complex controls and reactions. For instance, pose estimators provide tracking (e.g., Bounding Boxes (BB)) and 2D pose data. If a BB is near or at the sidewalk with a C tag, then the AV should stop to avoid potential collisions. But if the label is NC or LONG, the AV can start or keep moving, which will not affect traffic efficiency. In contrast, if a BB is at the roadways, then no matter what label is associated with it, the AV should immediately stop.

II. Related Work

In this section, we mainly review prior research in the area of predicting pedestrian behavior based on body pose and C/NC classification.

The research in [11], [12] leverages the contour of pedestrians to predict their intentions. Moreover, posture [13], [14] and body language [15] were also studied for the purpose of predicting pedestrian intentions. The research in [16]–[18] tends to approximate head and body orientation to estimate pedestrian intentions. However, in [18] the experiments showed that head detection did not provide useful data for the C/NC task. The work in [19] combined lateral speed, orientation, pose, and abstract scene information to feed them to a neural network, which was able to predict impending actions.

Most studies mentioned above focus on partial features of the body pose. The work in [20] suggested that without information about the pedestrians' posture and body motion, the detection of the pedestrians' intention changes will be delayed. The baseline evaluation of the JAAD dataset [4] supported this conclusion. They compared C/NC task performance between approaches with full body features and appearance. They specifically focused on sub-appearance of partial feature sets. The results suggested that the latter would not help to improve C/NC task performance. In [21], instead of appearance, they accumulate and update the full body pose (i.e., skeleton key-points) and features over time by a sliding time-window as input to their support vector machine (SVM) classifier. They obtained best C/NC task performance on the Daimler's dataset [22]. Later they improved their method and tested the approach on the JAAD dataset.

III. Method

In this section, we introduce the way to pre-process data in III-A, the definition of C, NC and LONG states in refsubsec:def and the structure of our neural network in III-C.

A. Data Processing

The typical COCO format body pose has 18 key-points. However, when pedestrians cross in front of the vehicle, one of their arms is likely partially or fully invisible to the ego-camera on the vehicle, so as key-points on the face. To resolve this issue, we follow a similar pose preprocessing procedure to the work described in [6]. A pruned pose only contains the 9 most stable key-points[1], which represent shoulders and legs (Fig. 3 shows relative positions of these points on the human body). These key-points indicate

[1]Right & left shoulder, neck, right & left hip, right & left knee, right & left ankle.

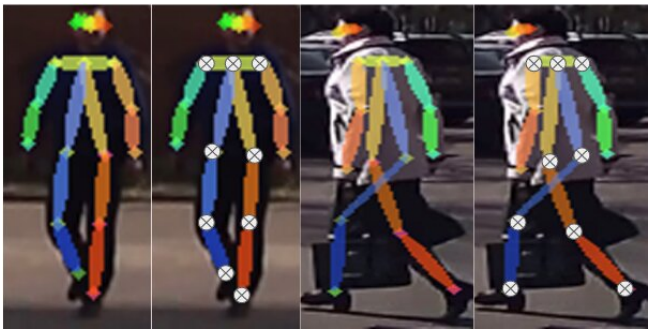

Fig. 3: 9 most stable key-points.

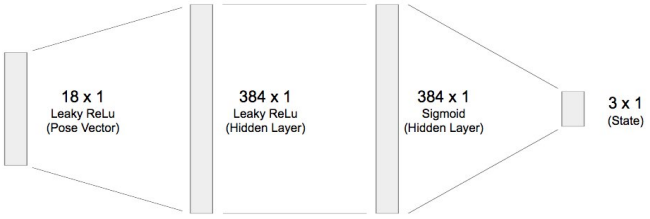

Fig. 4: Structure of the C/NC/LONG Classifier.

essential action information including motion state (start walking/ keep walking/ stop walking/ stand) and movement orientation. To eliminate the negative influence caused by the different pose scales, we translate and normalize them. First, we set the pose center at the neck (assign (0, 0) to the neck key point). Then, for each pose we respectively compute the normalization factor n: $n = max(pose.y) - min(pose.y)$ and normalize the x, y coordinates. The *pose.x* represents all x coordinates while the *pose.y* represents all the y coordinates associated with the pose. In Fang's work [6], they computed 396 features including distances, angle, etc. for their Support Vector Machine (SVM) classifier. In contrast, we have the neural network to extract and interpret features.

B. State Definition

Explicit definitions were made to divide the data into three categories. A pedestrian will be classified as C state if the pedestrian is crossing the road that the vehicles are using regardless of the collision chance. The NC state includes all the instances that the pedestrian's position with respect to the ground doesn't change or change due to some in-place actions such as loitering. For the LONG state, the pedestrian and the vehicle move parallel within ± 5 degrees deviation.

C. Neural Network

As the first method perform C/NC/LONG classification task, we provide a baseline method for future work to compare. As we mentioned in the introduction section, the real-time C/NC/LONG classification task is extremely sensitive to time constraints and mobile platforms generally have limited computational power. Thus, the baseline approach only contains 2 hidden fully connected layers. The network takes the x and y coordinates of the body pose as input data and outputs the probability for each state. Considering the very limited size of the input data (18 floating point numbers), we have trust in the ability of extracting and understanding features of our neural network despite the limited number of parameters. The details of our network are illustrated in Fig. 4.

IV. Experiments

In order to evaluate the performance of our classifier and create a pose annotation of the JAAD dataset, we use the AlphaPose [10] as pose estimator, which is briefly introduced in IV-A. Moreover, we describe the overall organization of our dataset and training procedure in Sections IV-B and IV-C. Then, the general performance including numeric information and sample images is reported in Section IV-D.

A. Pose Estimation

AlphaPose is an advanced real-time pose estimator which firstly achieved 70+ mAP (72.3 mAP) on the COCO dataset and 80+ mAP (82.1 mAP) on the MPII dataset. We use it to generate the COCO model body poses. We run the Windows version PyTorch implementation of AlphaPose with parameters: –conf 0.05, –nms 0.8, which are the confidence thresholds for human detection.

B. Dataset

The JAAD (Joint Attention for Autonomous Driving) [4] is a dataset that provides data including but not limited to BBs. AlphaPose generates poses for qualified images (width is greater than 60 pixels) cropped from the original JAAD frames. We remove poses that have average confidence scores lower than 0.6 and with key-points' scores lower than 0.5. Note that the average confidence score only takes 9 body key-points (Fig. 3) into account. In the end, after inspection of them, we have 12,756 manual annotated and auto-generated poses in total. Furthermore, since the JAAD does not directly provide labels of C, NC, and LONG, we map the behavioral label to them as follows. We term as C the behavioral labels of walking, crossing, moving fast, moving slow, slow down, speed up, and clear path when the corresponding subject possesses an LAT label, which means the pedestrian is crossing in front of the car. In contrast, if the subject is labeled as LONG in JAAD, which indicates the same behavior as our LONG label does, we term it to this label set. And, stopped and standing labels belong to NC. We also manually inspect and correct the label of each frame following the definition mentioned in III-B. After mapping and correction, we have 4,805 C, 4,096 LONG, and 3,855 NC labels in our pose dataset.

We take the first 10,544 (84%) images for training and the rest (16%) for testing (the images are organized in the same order as the original JAAD dataset). The dataset was split to 84:16 in order to maintain the balance of the samples from each class in the training and testing sets. In addition, this split ratio is also for generating a maximally independent testing set. The two sets are disjoint in terms of video sequences. The testing set contains zero frames from video sequences with frames in the training set, and vice versa.

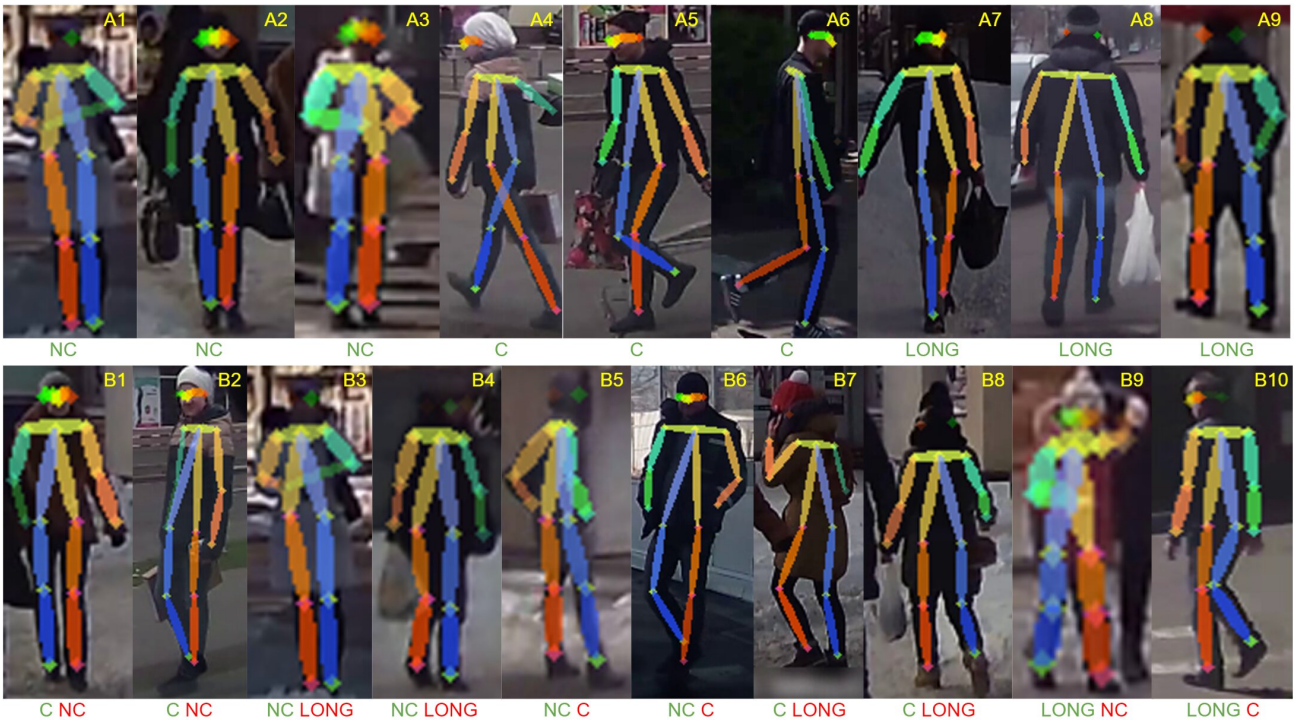

Fig. 5: The first and second row show success and failure cases. Green text indicates the ground truth for each image and red text indicates an erroneous prediction. If there is only green text under an image, it indicates a correct prediction.

In the testing set, there are 626 (28.3%) C, 862 (39.0%) NC, and 724 (32.7%) LONG instances. Furthermore, in the training set there are 4,179 (39.6%) C, 2,993 (28.4%) NC, and 3,372 (32.0%) LONG instances.

C. Training

We use PyTorch [23] to build, train, and test our neural network with an NVIDIA GTX 1080 Ti. We use the stochastic gradient descent (SGD) optimizer with an initial learning rate of 2e-1 and a momentum of 0.5, a PyTorch ReduceLROnPlateau scheduler with a 0.5 decay rate and 3 patience and a batch size of 128. In addition, we use cross-entropy to evaluate loss. We trained our neural network in 50 epochs and it converged within 10 epochs. Note that our model needs a high initial learning rate to converge and the Adam optimizer is less likely to lead to convergence in our experience. In addition, the size of this neural network is 2,378KB as a python pickle file without a computational graph, which is easy to be deployed.

D. Classifier

We make independent and sequential tests to evaluate the absolute and realistic performance of the classifier. However, to the best of our information and knowledge (at the moment) our classifier is the only work designed for the C/NC/LONG task. Although it is possible to pool the classes of interest to do the comparisons, we cannot simply pool C/NC sections from our C/NC/LONG results. This is due to the fact that the NC category of earlier works contains the LONG category of our work (previous NC = our NC + LONG) as Fig. 1 indicates. So, even if we take all the LONG instances out of the NC category of previous works, they are still incomparable with our work since they are designed for totally different tasks. In addition, although we reported acceptable results, our neural network is a first step towards a new set of algorithms that are applicable to this domain and have much space for improvement (as expected). We believe it is a good baseline experiment in this field and a step for future broader investigations. Due to these reasons, in this section we focus on reporting results and analyzing potential reasons for the failure cases.

TABLE I: Performance of the Classifier on the Testing Set.

Class	Total	Correct	Accuracy
C	626	439	70.13%
NC	862	797	92.46%
LONG	724	561	77.48%
ALL	2212	1797	81.23%

For the independent evaluation, Table I reports the overall and category accuracy and Fig. 5 shows samples of success and failure cases. We calculate the accuracy according to the widely used definition: $AC = C/T * 100\%$, where AC, C, and T stands for accuracy rate, total correct prediction, and total poses tested, respectively. We achieved a 81.23% general accuracy rate.

According to Table I, we can see that the model has better performance on recognizing NC state poses. The reason

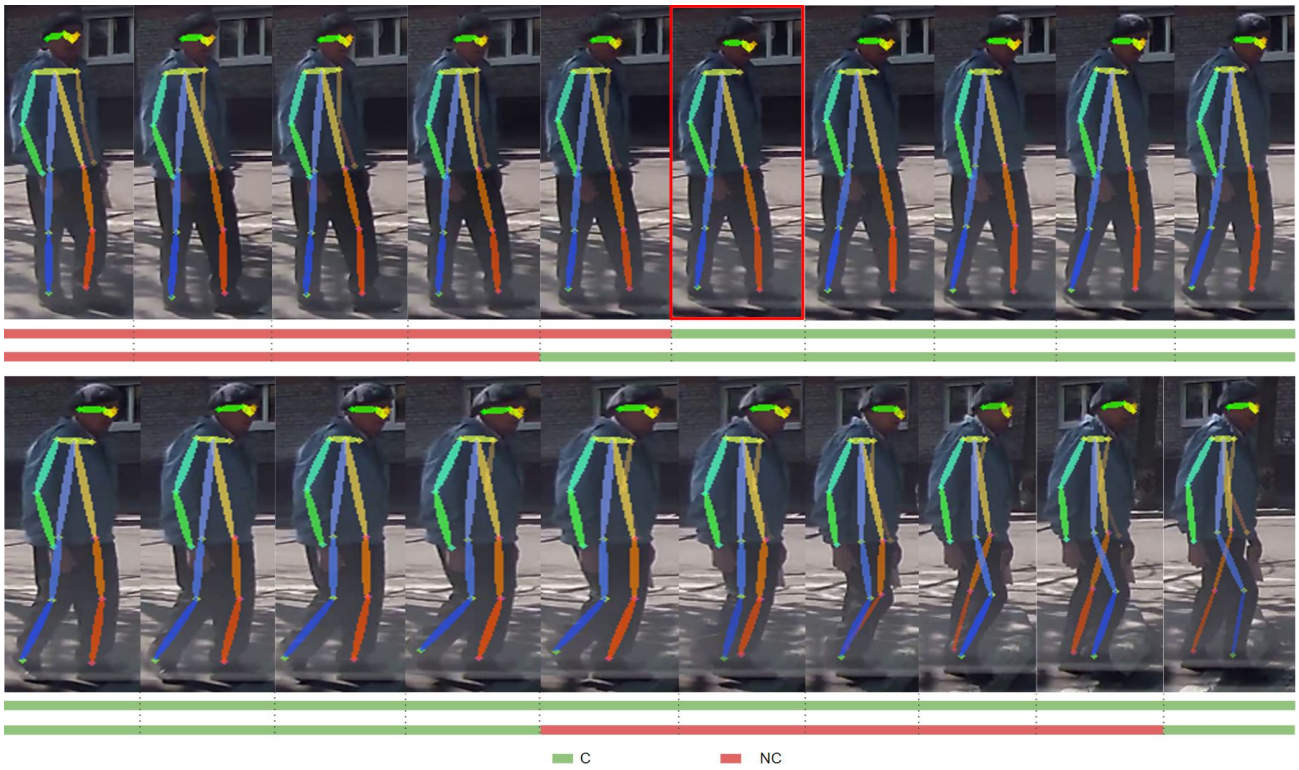

Fig. 6: A typical TTE test sequence. The highlighted frame is the Time-To-Event (TTE) frame. The first and second line of the bar under each frame represent ground-truth and estimation. Bars in green and red represent the C and NC states.

could be that the features of a standing pose are more stable and obvious than C and LONG state poses since the former are from static but the latter are from dynamic actions. Furthermore, through observation of the NC failure cases we notice that while standing, particularly close to curbside, some of the subjects would pace in-situ or move legs without exact purpose, such as in Fig. 5 B6, which makes their pose similar to C or LONG state. Moreover, people's unbalanced and ready-to-walk standing poses (Fig. 5 B5 can also lead to failed predictions). According to the error test results, we find that a large portion of failed predictions are incorrectly recognized C or LONG poses as NC states. The essential reason could be that the pose is similar to the NC state when the distance of legs is very close while walking as shown in Figs. 5 (B1, B2, and B9).

In addition, we report the Time-To-Event (TTE) test. It evaluates the classifier's response time to a state change of pedestrians, which partially reflects how early an AV starts maneuvering to avoid collision. To simplify our description, we introduce some related concepts and notation. The Time-To-Event frame (TTEF) is the first frame after the change of the pedestrian's crossing state. For instance, in Fig. 6, the subject's state changes from NC to C. The first five frames are in NC state and the last fifteen are in C state. So the sixth frame is the TTE frame. The Response Frame (RF) corresponds to the number of frames before ($RF < 0$) or after ($RF > 0$) the classifier starts responding to state changes. When the classifier responds to a state change and generates no less than nine correct estimations, the result is considered as a confident estimation (CE). Moreover, because there is no previous research report on the TTE test, we indirectly compare the response time of the classifier (CRT) with the human driver's response time (DRT). CRT is defined as $CRT = RF * 1/FrameRate$. The unit of CRT and DRT is the second (s).

We apply the TTE test on 87 sequences, which are in chronological order. As Fig. 6 shows, each sequence contains 20 frames (5 and 15 frames of them are before and after the TTEF (TTEF belongs to the later states)). The corresponding CRT space is from -1/6 to 1/2 s. Fig. 7 shows the general result. In total, the classifier successfully recognized state changes no later than 15 frames after the TTEF in 85 sequences. Another two late CEs happened in the 19 and 22 RF. The average and median RF (Events for which $RF > 15$ are ignored) is 4.238 and 1, which is 0.141267 s and 0.03333 s in the CRT. As the chart shows, the classifier is quite sensitive to state changes and even the latest CE has a CRT of 0.3667 s.

It is non-negligible that this TTE evaluation is an offline test since it does not take into account the latency introduced by the pose estimator. During the process of creating the 2D pose for the subjects, we process 15.4 frames per second on average using AlphaPose with an GTX 1080 card. As the JAAD dataset is described, its camera takes 30 or 60 frames per second for different videos. So, the advantage of processing speed and timeliness of our method is heavily dependent on and limited by the pose estimator.

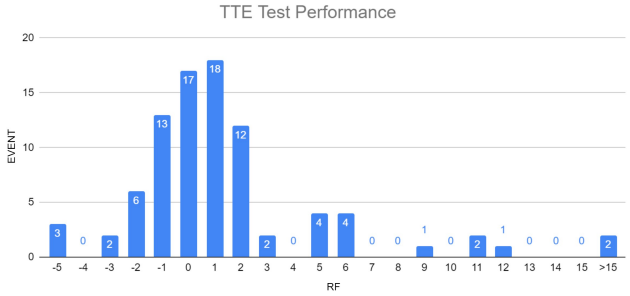

Fig. 7: The x-axis is the sample space of RF, the height of each bar represents the number of occurrences of each RF. On the x-axis in this graph the unit is the frame.

It is worth noting that the classifier even recognized the state change before TTEF in 41 sequences. For instance, one should look at the sequence in Fig. 6. We think the reason is that the neural network is able to detect features or preparatory poses, which are almost unperceivable to the human (these appear when pedestrians decide to change crossing states). It supports our opinion that the 2D pose contains enough information to be a strong indicator of the human's crossing state. Moreover, our neural network has also been proved to be powerful to extract and understand features with limited numbers of parameters and shallow layers.

V. Conclusions

In this paper, we extend the C/NC method for AVs to the C/NC/LONG problem and propose a fast shallow neural network classifier for this task. This paper contains extensive validations of our method and reports independent and sequential performance. Promising future work in this area could involve resolving occasional error poses generated by the pose estimator, extracting more robust features from the human pose to improve the classifier's performance, and integrating this with other reliable traffic information to make collision-avoidance systems more robust. Finally, we also hope that future work can compare this type of an approach with human-based systems in well-designed experiments.

Acknowledgment

The authors would like to thank all the members of the Center for Distributed Robotics Laboratory for their help. This material is based upon work partially supported by the Minnesota Robotics Institute (MnRI) and the National Science Foundation through grants #CNS-1439728 and #CNS-1939033.

References

[1] N. C. for Statistics and Analysis, *Pedestrians: 2017 data. (Traffic Safety Facts. Report No. DOT HS 812 681)*, Washington, DC: National Highway Traffic Safety Administration, 2019.

[2] W. Choi and S. Savarese, "Understanding collective activitiesof people from videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1242–1257, 2014.

[3] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 261–268.

[4] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017, pp. 206–213.

[5] D. Ridel, E. Rehder, M. Lauer, C. Stiller, and D. Wolf, "A literature review on the prediction of pedestrian behavior in urban scenarios," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 3105–3112.

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

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

[8] S. Koehler, M. Goldhammer, S. Bauer, S. Zecha, K. Doll, U. Brunsmann, and K. Dietmayer, "Stationary detection of the pedestrian?s intention at intersections," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 4, pp. 87–99, 2013.

[9] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.

[10] H. Fang, S. Xie, Y. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2353–2362.

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

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

[13] R. Furuhashi and K. Yamada, "Estimation of street crossing intention from a pedestrian's posture on a sidewalk using multiple image frames," in *The First Asian Conference on Pattern Recognition*, 2011, pp. 17–21.

[14] J. Hariyono and K.-H. Jo, "Detection of pedestrian crossing road a study on pedestrian pose recognition," *Neurocomputing*, vol. 234, 12 2016.

[15] R. Quintero, I. Parra, D. F. Llorca, and M. A. Sotelo, "Pedestrian path prediction based on body language and action classification," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 679–684.

[16] E. Rehder, H. Kloeden, and C. Stiller, "Head detection and orientation estimation for pedestrian safety," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 2292–2297.

[17] R. Quintero, I. Parra, D. F. Llorca, and M. A. Sotelo, "Pedestrian intention and pose prediction through dynamical models and behaviour classification," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 83–88.

[18] A. T. Schulz and R. Stiefelhagen, "Pedestrian intention recognition using latent-dynamic conditional random fields," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 622–627.

[19] J. Hariyono and K. Jo, "Pedestrian action recognition using motion type classification," in *2015 IEEE 2nd International Conference on Cybernetics (CYBCONF)*, 2015, pp. 129–132.

[20] F. Schneemann and P. Heinemann, "Context-based detection of pedestrian crossing intention for autonomous driving in urban environments," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 2243–2248.

[21] Z. Fang, D. Vázquez, and A. López, "On-board detection of pedestrian intentions," *Sensors*, vol. 17, p. 2193, 09 2017.

[22] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive bayesian filters: A comparative study," in *Pattern Recognition*, J. Weickert, M. Hein, and B. Schiele, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 174–183.

[23] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017. [Online]. Available: https://openreview.net/pdf?id=BJJsrmfCZ