Pedestrian Crossing Time Prediction and Risk Identification

ECE408

Zhenyu Pan, Zhuo Liu, Yongchao Liu

ABSTRACT

This study aims to explore high-risk pedestrians attempting to cross the street at signalized intersections and predict the crossing time. First, the intersection and crosswalk zones were bounded to judge pedestrian movement status. A coordination transformation was conducted to deal with the distortion of raw trajectory data. Then, a Dynamic Time Warping (DTW) based hierarchical clustering model was adopted to classify pedestrians by evaluating the trajectory similarity. The model identified a group of oversized pedestrians who tended to cross the street at a slower speed than other pedestrians. Further, We improved the traditional CNN & LSTM model structure by adding a time-distributed layer, and we used this structure to predict pedestrian crossing time. This new structure can not only capture the historical feature information of the predicted object, but also consider the historical features of the top ten influential objects around it at the same time, and extract the features separately and share the weights. The results show that it achieves an accuracy of 83.3% for the oversized pedestrians, while a 94.02% accuracy for normal-sized pedestrians.

Keywords: Hierarchical Clustering Model, CNN & Time Distributed LSTM, Crossing Time Prediction

PROBLEM STATEMENT

This problem aims to explore high-risk pedestrians attempting to cross the street at signalized intersections. The proposals mentioned two tasks: (1) classify different pedestrians; (2) predict the time needed to cross the street and judge whether the pedestrian can safely cross the street.

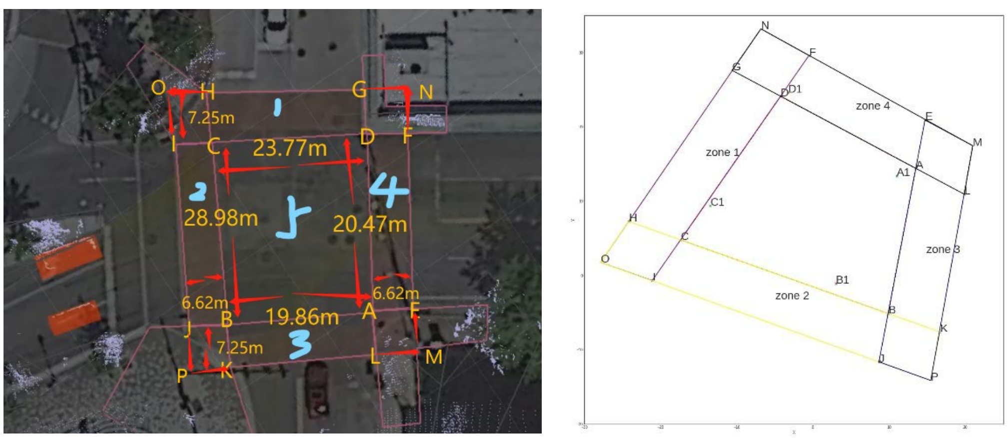
For the first task, the algorithm is supposed to extract features from pedestrian trajectories. Two intrinsic trajectory features must be accounted for, i.e. the trajectories are varied in length, and the trajectories are not aligned in time. In addition, the dataset provides the size of the bounding box, which means the body dimension can be considered.

For the second task, an algorithm that provides an end-to-end prediction may have better accuracy, given that the pedestrian movement can be affected by complicated exogenous and endogenous processes. An end-to-end based prediction means that the algorithm would directly output the predicted crossing time, without modeling the decision process of a pedestrian in an explanation way.

To support tasks 1 and 2, proper data reduction and pre-processing are necessary. Possible tasks related to quality screening and data cleaning would include noise reduction, trajectory transformation, trajectory filtering, etc.

DATA PREPARATION

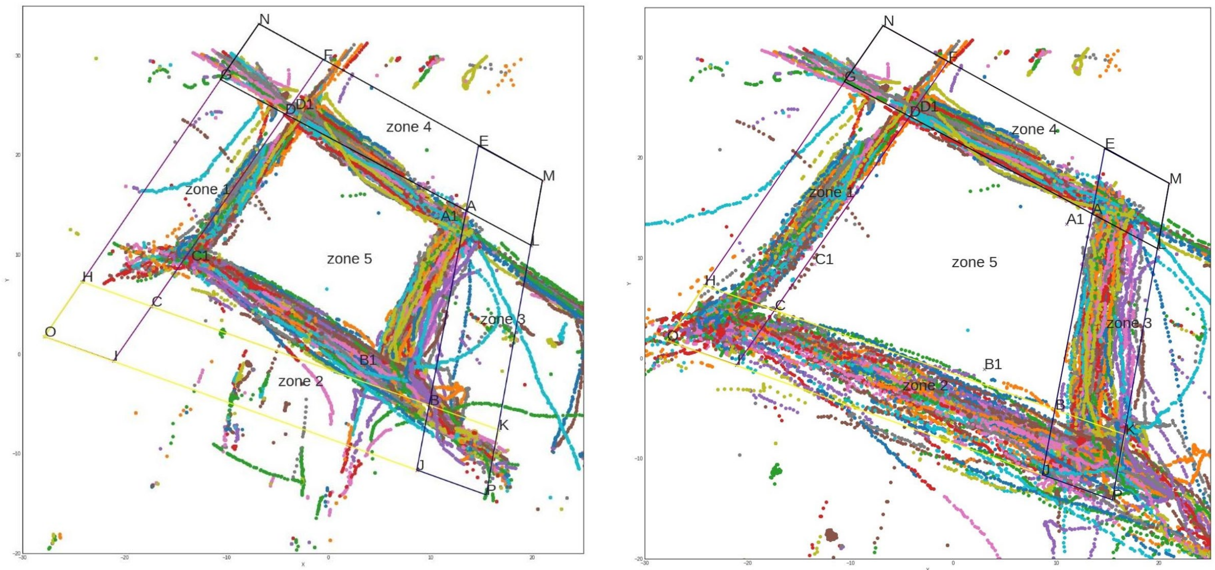
First, the intersection and crosswalk zones were bounded to justify whether a pedestrian crosses the intersection or not. Figure 1 shows the profile of the boundary. Zones 1 to 4 represent the crosswalk area, and zone 5 is the intersection area. Edge points were identified based on actual geometric designs in Google Map. Specifically, four nodes A, B, C, and D were given by the proposal, with A as the origin.



(a)bird view (b) zone profile

Figure 1 Boundary of intersection and zones

Figure 2-a shows the distribution of raw trajectories in the intersection area. The pedestrian trajectories are supposed to be distributed in the crosswalk areas, i.e. zones 1 to 4. However, a significant distortion was observed that most pedestrian trajectories were not in the crosswalk areas. This might be due to the coordination system was drifted between the intersection geometry (defined by A, B, C, D) and the LiDAR coordinates. For the LiDAR coordinate system, it was defined by A1, B1, C1, and D1. Therefore, a transformation matrix was calculated to map the "A1B1C1D1" coordination system to the actual "ABCD" coordination system. Accordingly, the raw pedestrian trajectories were transformed back to the correct positions. Figure 2-b shows that after the transformation, the pedestrian trajectories were mostly distributed in the crosswalk area.



(a)raw trajectories (b)transformed trajectories

Figure 2 Trajectory Transformation

Further data filtering process was adopted after the trajectory transformation. A trajectory was removed if it satisfied any one of the following conditions:

(1) the trajectory has over 95% of its points outside the boundary OPMN
(2) the trajectory length is less than 5 points
(3) distance between the trajectory start point and end point is less than 1.5m
(4) the trajectory was circling the start point
(5) the trajectory was failed to traverse the crosswalk boundary (using 1.5m as a buffer to account for the localization error), i.e. whether the pedestrian crossed the street is unclear.
(6) the trajectory has a point that belongs to the "invalidating" status

After the filtering process, 79 pedestrian trajectories were qualified for the analysis.

PEDESTRIAN CLASSIFICATION

Firstly, the trajectory data is time-series data, which means many traditional clustering methods such as the k-means clustering may not be applicable, due to their limitations to account for the time information. Secondly, as mentioned above, the trajectories are of different lengths and not aligned in time; therefore, a suitable algorithm must be able to account for this. In this study, the Hierarchical Clustering model was implemented to classify pedestrians based on trajectory features. Specifically, the Dynamic Time Warping (DTW) based hierarchical clustering model was adopted. The DTW is able to measure similarity between two temporal sequences that do not align exactly in time, speed, or length [1].

CROSSING TIME PREDICTION

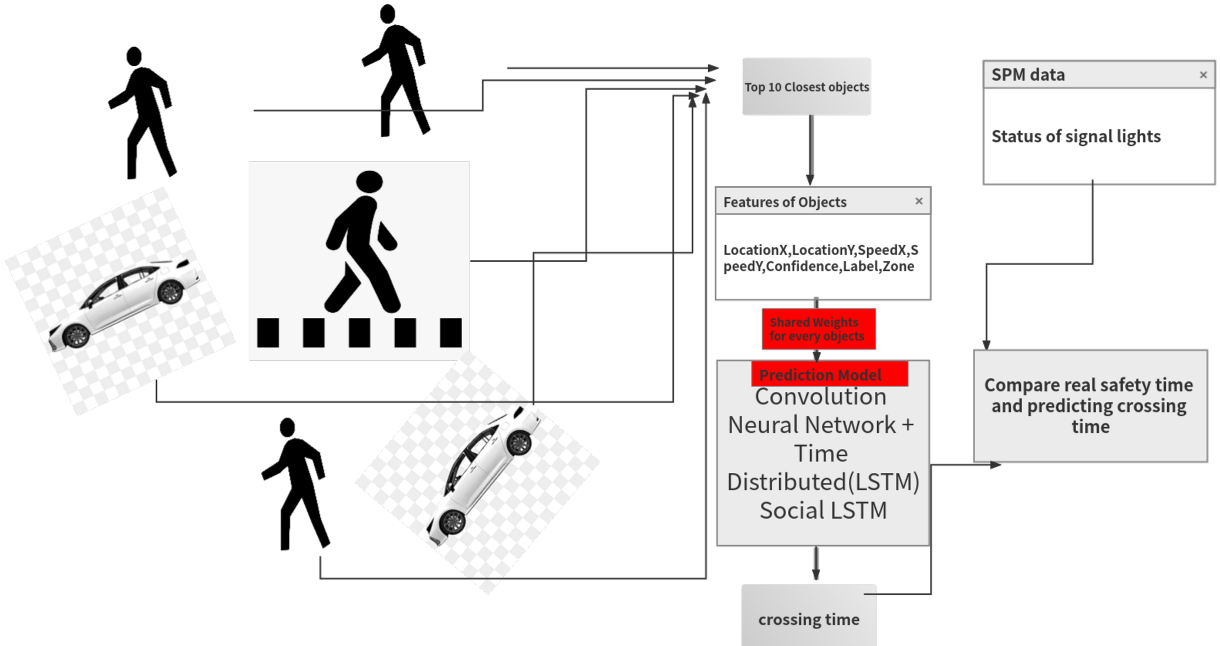


Figure 3 CNN & LSTM Algorithm for Crossing Time Prediction

Figure 3 shows the algorithm pipeline for the crossing time prediction. The shape of input data is (6751, 30, 10, 11). 6751 means the total number of samples.30 means the time dimension. We select 30 historical points (about equal 3s) from every object at every time as the individual sequence features. 10 means the number of closest objects related to the specific pedestrian. 11 means the total features

Then by capturing the other 10 most recent objects that exist in each time point of the pedestrian, and sequentially taking out the data of 30 points in the history of this object as input, if there are less than ten objects, it will automatically add 0. If it does not exceed, only 10 are selected. The same strategy is used to complete the historical time points. In this way, we finally got 6,540 sample data from 79 pedestrians.

Firstly, for the trajectory of each pedestrian, we can create more than one sample. We start with the 30th point of every pedestrian, and create num- 30 (num means number of points) samples for one person. When the pedestrian actually arrives on the opposite side of the road, and subtract it from the historical trajectory time to obtain the true mark value y.

After creating the data set, we divide all the data sets into three parts, 70% of the data is used as the training set, 20% of the data is used as the validation set, and 10% of the data is used as the test set. And import it into the model for training.

The model we choose is CNN as the input layer [2], the feature attributes of the sample as the channel dimension, and then through a layer of maximum pooling to extract the coarse features in the initial sample, and then through a layer of time distributed, in this layer, our purpose is to use LSTM for different objects to extract the features of a separate time series, and share the weights [3]. This can greatly take into account the influence of objects around pedestrians on pedestrians themselves. Then there is a layer of dropout layer, the purpose is to prevent the model from overfitting, reduce a certain calculation unit randomly, here the amount of reduction is set to 30%. Then there is a flatten layer, which pulls all the computing units apart, and directly outputs the final predicted value remaining time after passing through the dense layer of the two layers through the fully connected network.

For predicting whether pedestrians can cross the street safely, we predict the remaining time at all trajectory points in the entire path of the pedestrian [4] and compare it with the remaining time of the green light. If it is greater than the remaining time, it is judged as dangerous, if it is less, it is judged as safe. Specifically, it is divided into the following situations:

1. Find out the current traffic light conditions of the start track point and end track point (within 1.5 meters of the target area for the first time) of the track. If the end status is green or yellow, and the start status is green or yellow, then The real remaining safe crossing time in the middle is the time point of the first red light after the end minus the current time of each track point and the estimated time required. If it is positive, it is safe, if it is negative, it is dangerous.

2. If the starting state is a red light and the end state is a green light or a yellow light, find the time node that turns green for the first time after the starting point. All track points before this time node are judged to be dangerous, and all subsequent tracks Point, subtract the current time and predicted time of all track points from the time of the next red light appearance. If it is positive, it is safe, and if it is negative, it is dangerous [5].

3. If the end state is a red light, the time when the red light state appears for the first time after the starting point is found. Before that, subtract the current time of the trajectory point and the estimated time required. If it is positive, it is safe, if it is negative, then Danger, all points after the red light appears are dangerous.

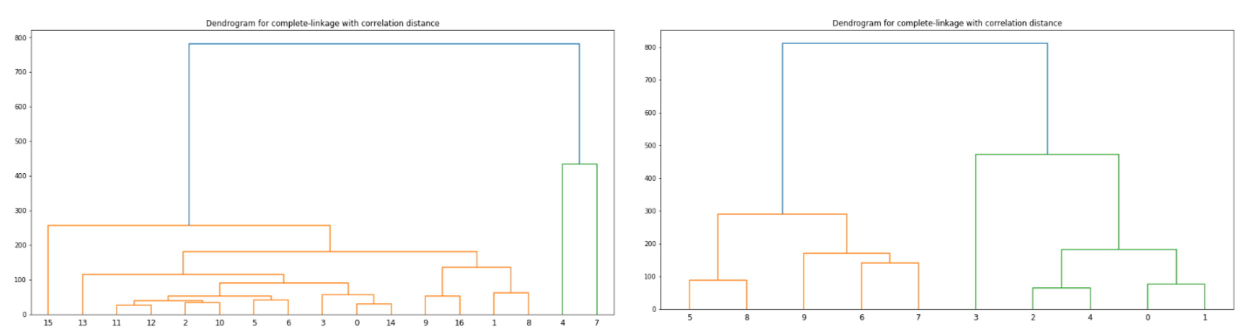
Finally, by selecting the bottom 20% of the trajectory points of each pedestrian as a feature, if more than 80% of the points are judged as safe, the pedestrian is judged as safe, otherwise it is dangerous.

RESULT

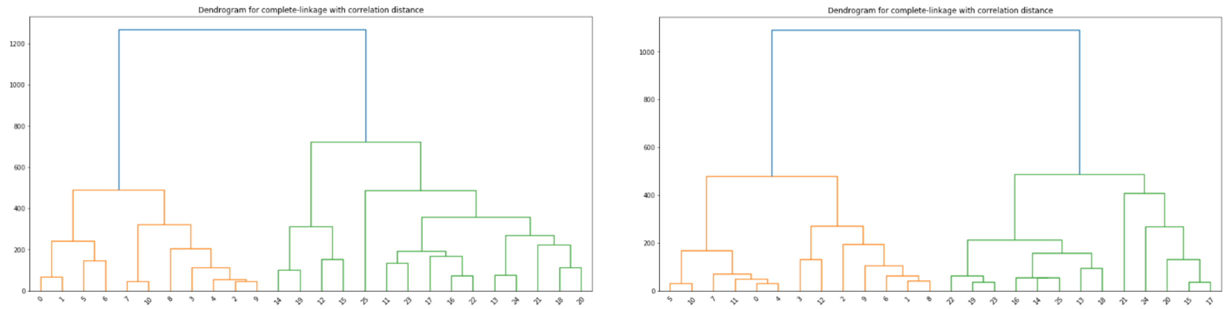
Pedestrian Classification

The clustering result is shown in Figure 4 and Table 1. The algorithm classified the pedestrian trajectories into four types for each zone respectively. The pedestrians in the same crosswalk area (i.e. zone area) were supposed to be affected by similar exogenous factors; therefore, a separate classification process on each zone would be more accurate to capture a pedestrian's intrinsic features.

It is expected that a pedestrian type that had a slow velocity may indicate a special pedestrian cluster, and these pedestrians would be more likely to be at risk when they cross the street. In zone 1, pedestrian type 4 had a significantly slow velocity (i.e. 0.44m/s); this pedestrian type was likely an old man. In zones 2 to zones 4, when looking at the slowest pedestrian type, it is very interesting that these pedestrians tended to have an oversized body dimension. Specifically, in zone 2, the pedestrian type 4 had the slowest velocity; for these pedestrians, both the rate of bounding box size Y over X, and the rate of bounding box size Z over X are the largest compared with other pedestrian types; in zone 3, the pedestrian type 3 had the slowest velocity; for these pedestrians, the rate of bounding box size Y over X is smallest; similarly, in zone 4, the pedestrian type 4 had the smallest rate of bounding box size Y over X, and this pedestrian type has the slowest velocity. The rate of bounding box dimension indicates the pedestrian body dimension to some extent; either a larger rate or a smaller rate means an oversized body dimension compared with the normal. The result indicates that oversized pedestrians tended to have a slower crossing speed.



(a)zone 1 (b)zone 2



(c)zone 3 (d)zone 4

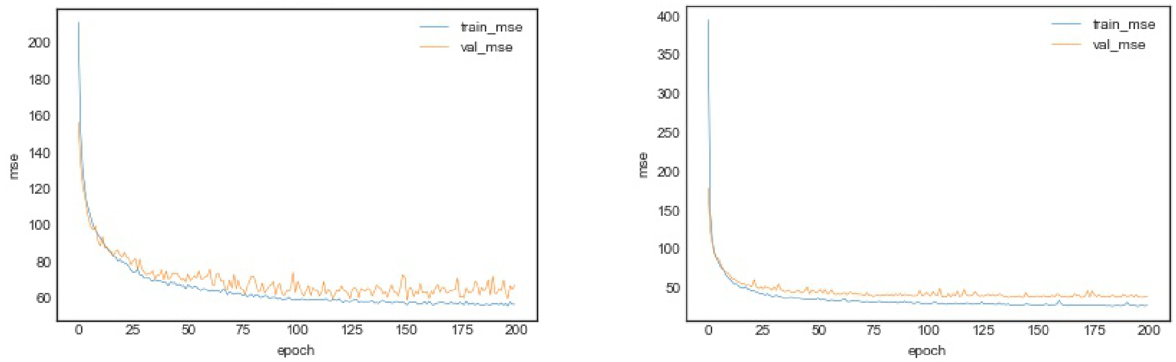
Figure 4 Pedestrian Trajectory Classification in Each Zone

TABLE 1 Pedestrian Classifications

Crosswalk Area	Type	BBox_Size_XY_rate	BBox_Size_XZ_rate	Velocity
zone1	1	1.089	2.911	1.096
	2	0.947	1.909	2.264
	3	1.080	2.471	1.237
	4	1.030	2.468	0.445
zone2	1	0.867	2.246	1.357
	2	0.945	1.758	1.033
	3	0.943	2.214	1.699
	4	0.987	2.527	0.799
zone3	1	1.109	2.863	0.669
	2	1.274	2.518	1.008
	3	1.005	2.600	0.498
	4	1.144	2.742	0.697
zone4	1	1.318	2.528	1.315
	2	1.082	2.465	0.814
	3	1.301	2.326	1.228
	4	1.120	2.370	0.884

Pedestrian Crossing Time prediction

After using the CNN to extract features from the object, we used multiple LSTMs to analyze each dimension of the object, and let the LSMTs share the weights, finally we combined the features from LTSMs and fed it into the remained network. The model achieved a MSE of 7.2 on the training dataset, 11.4 on the validation set, and 17.6 on the test dataset (Figure 5-b). Figure 5-a shows the results from traditional CNN, which is significantly worse compared with our improved algorithm.



(a)traditional CNN (b)CNN & LSTM with time-distributed layer

Figure 5 CNN & LSTM Algorithm for Crossing Time Prediction

DISCUSSION AND CONCLUSION

The study used a Dynamic Time Warping (DTW) based Hierarchical Clustering model to classify pedestrians, by evaluating the trajectory similarity. The advantage of this clustering algorithm is that it is suitable for the time-series data, and it is able to account for trajectories that is either of different length or not aligned in time. By using this algorithm, a type of oversized pedestrians who tended to cross the street at a slower velocity was identified.

Further, the study used an improved CNN & LSTM network to predict the pedestrian crossing time. Traditional CNN & LSTM network can only capture the historical feature of the object itself, while it can ignore the influence from other objects in the scene. Therefore, we added a time-distributed layer, and we embedded the LSTM in it, so that it can generate multiple LSTM to extract object features.

Finally, we applied the crossing time prediction model to two types of pedestrians that we classified, i.e. the oversized pedestrians and the normal sized pedestrians; the results show that the model achieved an accuracy of 83.3% and 94.02% respectively.

EFERENCES

1. Bruce P, Bruce A, Gedeck P. Practical statistics for data scientists: 50+ essential concepts using R and Python[M]. O'Reilly Media, 2020.

2. A. Breuer, S. Elflein, T. Joseph, J. Bolte, S. Homoceanu and T. Fingscheidt, "Analysis of the Effect of Various Input Representations for LSTM-Based Trajectory Prediction," 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019

3. Taeyeop Lee, Min Kyung Sung, Seulkee Lee, Woojin Yang, Jaeho Oh, Jeong Yeon Kim, Seongwon Hwang, Hyo-Jeong Ban, Jung Kyoon Choi, Convolutional neural network model to predict causal risk factors that share complex regulatory features, Nucleic Acids Research, Volume 47, Issue 22, 16 December 2019

4. Muhamad Nazri Borhan, Ahmad Nazrul Hakimi Ibrahim, Affan Aziz, Muhamad Razuhanfi Mat Yazid, The relationship between the demographic, personal, and social factors of Malaysian motorcyclists and risk taking behavior at signalized intersections,Accident Analysis & Prevention,Volume 121, 2018

5. Per Gårder, Pedestrian safety at traffic signals: A study carried out with the help of a traffic conflicts technique, Accident Analysis & Prevention, Volume 21, Issue 5, 1989