A Hierarchical Clustering Analysis (HCA) in Automatic Driving Regarding to Vehicle-to-Vehicle Pedestrian Position Identification

Jie, Xue
Zhi, Huang
Jue, Zhou
Yaobin, Chen
Stanley, Chien
Transportation Active Safety Institute
Indiana University Purdue University, Indianapolis
USA

Paper Number 17-0176

ABSTRACT

Identifying the pedestrian position correctly is a challenging but crucial topic in the area of automatic driving. It is also an arising research focus that needs using the latest big data and data science techniques. In this paper, a hierarchical clustering (HCA) statistics learning algorithm has been applied to determine the location and amount of pedestrians detected by different vehicles. The vehicles have been equipped with a Pedestrian Autonomous Emergency Braking (PAEB) system. The inherent inaccuracy of the pedestrian sensing from these vehicles has been taken into consideration. It is found that the HCA method can generate robust results, since the proposed HCA structure also takes the vehicle ID information as additional block information between signals into the calculation. The HCA method determines the possible number of actual pedestrians by grouping the nearby pedestrians who are sensed and broadcasted by different vehicles. The simulation results have confirmed the effectiveness and applicability of the proposed HCA method. It is believed that the results using the HCA method can provide realistic information for vehicle PAEB systems to make better decisions to avoid crashing into pedestrians.

INTRODUCTION

As many automobile companies have announced the incorporation of Autonomous Emergency Braking (AEB) into their automobiles, pedestrian recognition systems based on onboard vehicle sensors, such as radar, camera, LiDAR, etc., have been installed on more vehicles. If a vehicle can send its sensor detected pedestrian information to nearby vehicles through the Vehicle-to-Vehicle (V2V) communication network, receiving vehicles may be able to use this information as early pedestrian detection and the chance of crashes will be reduced.

The V2V communication based on DSRC (Discrete Short Range Communication) technology has been studied extensively [1,3]. Many efforts have been made to use this technology to improve the road safety. Meanwhile, there also have been developments in Pedestrian Autonomous Emergency Braking (PAEB) technology, which can provide autonomous braking when there is an eminent frontal crash to a vehicle, pedestrian, or bicyclist if the driver fails to apply braking or applies insufficient braking [2,3]. The PAEB system uses radar, camera, and LiDAR sensors individually or in conjunction with one another to detect the presence and the location of the object in front of the vehicle [4,5]. For example, Premebida et al. [4] proposed a LiDAR and vision-based approach for pedestrian detection and tracking.

The performance of PAEB system has been improved significantly in recent years and been offered as standard equipment or an option on many vehicles. It is certain that all vehicles will be equipped with V2V communication capability and PAEB features in the future. There will also be a long period of time during which vehicles with and without the PAEB and V2V technology will coexist on the road.

If V2V works in conjunction with PAEB, this system is referred to V2V-PAEB system. One of the problems for this system is that when a subject vehicle receives many pedestrian position information messages from other vehicles, it does not know if each pedestrian reported by one vehicle is the same pedestrian reported by other vehicles. Therefore, it is necessary to create a method in order to accurately determine the actual amount of pedestrians. The main goal of this paper is to develop an efficient method for accurately identifying the exact positions and the amount of pedestrians from data provided by multiple vehicles equipped with PAEB systems in the V2V communication network environment.

There are significant safety benefits when the PAEB system is integrated into V2V communication systems. The benefits can be achieved by empowering every V2V enabled vehicle to make PAEB decisions based on the PAEB sensory data from other nearby vehicles. Figure 1 shows a scenario to demonstrate the usefulness of an integrated V2V and PAEB (V2V-PAEB) system. When the black car on the right lane is moving forward, a pedestrian is crossing the street. The pedestrian and the black car cannot see each other since their views are obscured by the truck in the middle lane. It is possible that the black car may collide with the pedestrian since it may be too late for the black car to brake after its PAEB system sees the pedestrian. In a V2V-PAEB environment, the position and the trajectory of the pedestrian can be detected by the truck and the car on the left lane. Then the pedestrian information can be transmitted through the V2V network to the black car on the right lane long before the black car can see the pedestrian. This enables the black car to use the received pedestrian information to make safety decisions earlier.

Figure 1. The truck obscures the right car and the pedestrian.

Figure 2 shows an example of V2V-PAEB environment in a busy intersection. Curved lines connecting cars represent the V2V communication. The key for the successful operation of the collaborated V2V-PAEB includes (1) each vehicle broadcasts its own PAEB detected pedestrians' information and receives pedestrian information from nearby vehicles through the V2V network, and (2) be able to extract location and trajectory information of pedestrians accurately from the V2V messages from many different sources.

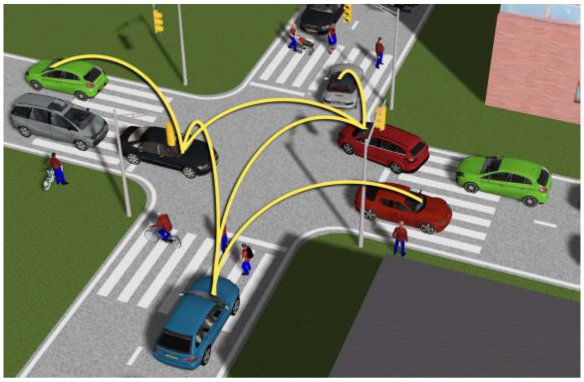

Figure 2. V2V-PAEB environment in a busy intersection.

Figure 3 shows an information extraction process of the V2V-PAEB system. The goal of the pedestrian identification is to develop an algorithm that enables each V2V enabled vehicle to construct pedestrians' locations and trajectory information accurately from the pedestrian information sent from several nearby V2V-PAEB enabled vehicles.

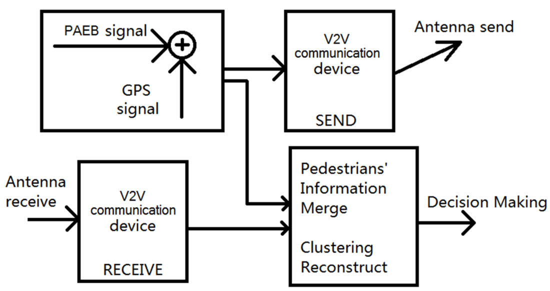

Figure 3. V2V-PAEB pedestrian safety decision-making process.

The work described in this paper is built on the prior V2V-PAEB research effort described in [6]. Figure 4 shows the architecture of the V2V-PAEB system described in [6]. The architecture assumes that V2V enabled vehicles can broadcast their PAEB system detected pedestrian position information as a V2V message, and can receive pedestrian position V2V messages broadcasted from other nearby vehicles. Each vehicle makes safety decisions (warning/braking) by predicting potential collisions based on the pedestrians' locations obtained from its own PAEB system and received V2V messages.

The flowchart in Figure 4 shows the necessary subtasks to make the V2V-PAEB system work. Each block in Figure 4 represents a specific problem that needs to be addressed in order to make the V2V-PAEB system function properly. One specific block,

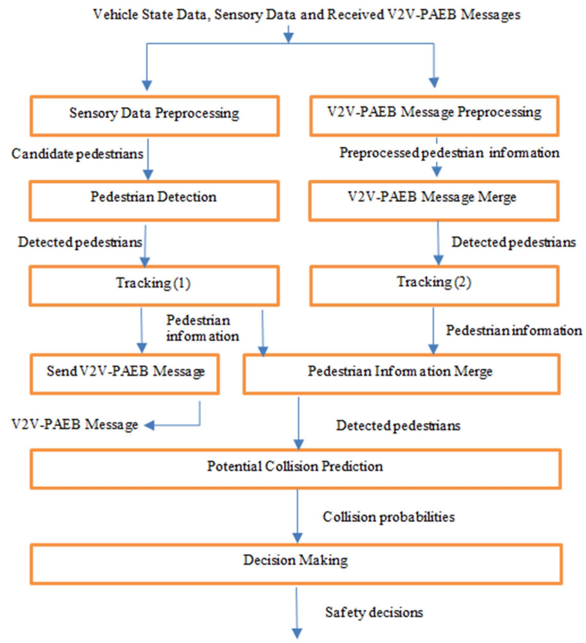

Figure 4. V2V-PAEB system proposed in [6].

"Pedestrian Information Merge", presents an interesting problem. When n pedestrians and m vehicles are in a small area, each vehicle can potentially see 0 to n pedestrians and can broadcast detected pedestrian positions through the V2V network. Due to the errors introduced by the inaccuracy of a vehicle's GPS and PAEB sensors, different vehicles may generate different pedestrian locations for the same pedestrian. There is a high possibility that $n \times m$ pedestrian positions are broadcasted in the V2V network. Assuming that each pedestrian is seen by at least one vehicle, and each vehicle does not necessarily see all pedestrians, how to determine the location of n pedestrians from V2V messages by m vehicles is a major issue raised but not solved [6]. This paper describes a method for the block "Pedestrian Information merge". The method enables each V2V-PAEB enabled vehicle to construct pedestrians' location information accurately from the pedestrians' information received from nearby vehicles.

In order to extract real pedestrian information in a large set of PAEB messages in the V2V network, the nature of the errors in the data need to be investigated. Wang, T. et al. [7] described human tracking using Delphi ESR-Vision Fusion in complex environments. A radar-vision fusion system has been built utilizing a 77GHz 2D Delphi Electronically Scanning Radar (ESR) and a CCD camera. The radar error distribution results has been explained. A simple uniform error distribution will be taken into consideration in the experiments of this paper.

Based on our best knowledge, there is no published work on data fusion (reconstructing pedestrians from PAEB information) in a V2V network provided from multiple vehicles. This paper attempts to develop a data fusion (pedestrians' signal reconstruction /clustering) algorithm to address this problem.

This paper is organized in three parts. Section II describes the problem formulation of pedestrian position detection and broadcast between vehicles. In Section III, an algorithm is proposed to cluster pedestrian information from different vehicles, find the approximate number of pedestrians, and draw the safe region. Simulation and conclusions are given in Section IV.

PROBLEM FORMULATION

Figure 5 shows a scenario at a road intersection, while vehicles and pedestrains are going across the intersection at the same time. In this scenario, there exist four vehicles (A, B, C, and D) and five pedestrians (1, 2, 3, 4, and 5).

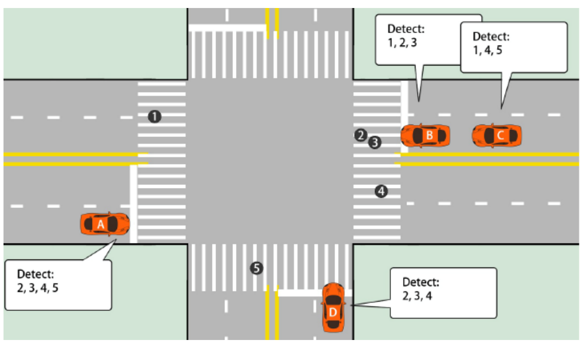

Figure 5. A road intersection scenario.

In our study, a sensor (camera, radar or Lidar) is installed in front of the vehicle. Due to the limitation of sensor's viewing angle, only the pedestrian within the detention area could be detected.

The pedestrians detection results are shown in Table 1. Pedestrian 1 is inside of the sensing area of vehicle A and B. As a result, it is detected by these two vehicles. Pedestrian 2 is detected four times by vehicles A, B, C and D. Pedestrian 3 is blocked from vehicle C's view by vehicle B. Therefore, pedestrian 3 is detected three times by vehicle A, B and D. Pedestrian 4 is outside of the sensing area of vehicle B, it can be detected three times by vehicle A, C and D. Pedestrian 5 is outside of the sensing area of vehicle B and D, it can be detected twice by vehicle A and C.

Table 1.
Pedestrians detected by Vehicles

Vehicle ID	Pedestrian ID
A	2 ,3 ,4 ,5
B	1 ,2 ,3
C	1 ,4 ,5
D	2, 3, 4

Because of the inevitable error of GPS and Detection sensor, the position signal for the same person varies. One pedestrian may cause multiple signal during the detecting process.

Figure 6 represents the pedestrian positions from the view of vehicle A, which means we take the position of vehicle A as the original point (0, 0). The red spot indicates the pedestrian position signal. The numbers 1 to 5 indicate the pedestrian's true position. Although there are only five pedestrians, each pedestrian is detected multiple times. As a result, there exists a total number of 13 points in Figure 6 instead of 5. The available information including the location of each signal in X and Y axis and also the vehicle ID information indicateing where the signal comes from. The signal information is presented in Table 2.

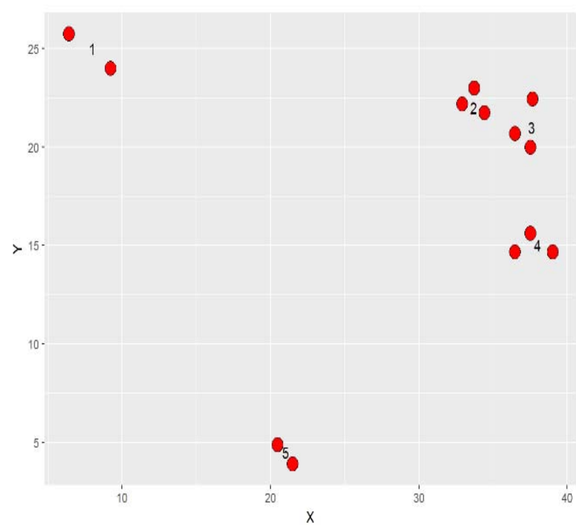

Figure 6. Pedestrian position signal.

Knowing the position and vehicle ID, our goal is to figure out which signal belongs to which person and draw a safe region that is large enough to cover the person's true position.

Table 2.
Pedestrian Signals

Signal ID	Vehicle ID	Position in X	Position in Y
1	A	33.719	23
2	A	36.469	20.688
3	A	37.5	15.625
4	A	20.469	4.875
5	B	6.406	25.75
6	B	32.906	22.188
7	B	37.656	22.438
8	C	9.219	24
9	C	36.469	14.688
10	C	21.469	3.906
11	D	34.406	21.75
12	D	37.5	20
13	D	39	14.688

Retrieving the pedestrian ID could be considered as clustering points into distinct groups. Finding the safe region for each group could be formulated as drawing the confidence region for each cluster. Hence, the pedestrian detection problem could be studied as a combination of clustering and confidence region.

PEDESTRIAN DETECTION APPROACH

Clustering Algorithm

Cluster algorithm is an important part of unsupervised learning. It is widely applied in machine learning and data mining areas. As the algorithm is unsupervised, which means no label is assigned to the data. Clustering means the separate data into groups or clusters that the data are similar inside the group but dissimilar between each group. There are different types of clustering algorithms, including K-mean clustering algorithm, Gaussian mixture model, Latent Dirichlet allocation, and hierarchical clustering [8].

In this paper, hierarchical clustering is chosen for implementation mainly for two reasons. First of all, among other clustering algorithms, such as K-mean clustering algorithm or Gaussian mixture model, the number of clusters has to be picked beforehand. However, for our problem the true number of pedestrians is not known. In the contrast, it is the answer we would like to find out by using the signal data. Thus, these types of algorithm do not fit our problem well. The second reason is comparing with K-mean or Gaussian mixture model, the hierarchical clustering is more powerful in cope with more complex shapes. In this study, no presumption is assigned for the shape. So hierarchical could be an algorithm fit for our problem.

The hierarchical clustering algorithm is an algorithm that produces a sequence of nested clusters. There are two general ways to implement hierarchical clustering algorithms, one is the agglomerative method, and the other is the divisive method. The divisive method is a top-down approach. It starts with all data in one big group, then recursively splits until each point lies in one individual cluster or a stop criterion is met. The agglomerative method, on the other hand, is a bottom-up approach. In this approach, each data point is initialized in its own cluster, and then clusters are merged until all points fall into one large cluster .

The key step of the either divisive approach or the agglomerative approach is how to calculate the proximity between two clusters. The most common way of defining proximity including the single link, complete link, average link, ward's method, and centroid method. Single link method uses the minimum of the distance between the points in the different clusters. Complete link method uses the maximum of the distance between the points in the different clusters. The average link proximity between two clusters is determined by the average of the pairwise proximities between all pairs of points. The proximity for ward's method is defined as the increase in the squared error that results when two clusters are merged. In this paper, the ward's algorithm is chosen because the ward's proximity provides the approach of cluster analysis that focuses on the analysis of variance. For this type of proximity measurement, the agglomerative approach is applied for implementing the hierarchical clustering algorithm [9].

There are multiple approaches for measuring the distance of quantitative variables. In this paper, Euclidean distance is chosen to measure the similarity between each signal as it is the most common way to measure the physical distance in the Cartesian coordinates.

For distance between signal s_i and s_j with position (x_i, y_i) and (x_j, y_j), the Euclidean distance is calculated as

$$d_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \qquad \text{(Equation 1)}$$

Vehicle ID Information Implementation

Besides of the position information, we also have the vehicle ID information within the signal. The vehicle ID information tells us where the signal comes from. This type of information could be used to prevent the clustering algorithm from grouping points from same vehicle ID. For example, s_i^A and s_j^A cannot be merged as one group. Because they should be signals from different persons. However, s_i^A and s_j^B may put into the same group as they might be signals from the same person but detected by multiple vehicles. In order to avoid signals with same vehicle ID grouping together, a relative large distance is assigned so that they would be less likely to merge.

Safe Region

The concept of multivariate analysis is applied to draw the safe region. After cutting the dendrogram, points are assigned into a number of groups. Assuming in group k, we have n points X_1, X_2, ..., X_n. These points are considered to be a random sample from a $N_p(\mu,\Sigma)$ population. The confidence region for the mean, μ, of a p-dimensional normal population is available from

$$\alpha = P\left[n\left(\bar{X}-\mu\right)' S^{-1}\left(\bar{X}-\mu\right) > \frac{(n-1)p}{(n-p)} F_{p,n-p}(\alpha) \right]$$

(Equation 2) [10]

where

$$\bar{X} = \frac{1}{n}\sum_{j=1}^{n} X_j \quad \text{(Equation 3) [10]}$$

$$S = \frac{1}{n-1}\sum_{j=1}^{n}\left(X_j-\bar{X}\right)\left(X_j-\bar{X}\right)'$$

(Equation 4) [10]

In our case, we have $p=2$ as we are dealing with signal in x and y direction [10].

So the $100(1-\alpha)\%$ confidence region for the mean of a 2-dimensional normal distribution is the ellipsoid determined by all μ such that

$$n\left(\bar{X}-\mu\right)' S^{-1}\left(\bar{X}-\mu\right) \le c^2 = \frac{(n-1)p}{(n-p)} F_{p,n-p}(\alpha)$$

(Equation 5) [10]

From the symmetric 2 by 2 matrix S, we have the eigenvalue λ_1 and λ_2 with its corresponding eigenvector e_1 and e_2. With the center of $\bar{X}$ the axis of the confidence region are

$$\sqrt{\lambda_1}\sqrt{\frac{2(n-1)}{n(n-2)}F_{2,n-2}(\alpha)}e_1 \text{ and}$$

$$\sqrt{\lambda_2}\sqrt{\frac{2(n-1)}{n(n-2)}F_{2,n-2}(\alpha)}e_2.$$

By choosing the significant level α, there are $100(1-\alpha)\%$ that the true mean lies in confidence region. Usually, it is chosen to be 5% [11].

Proposed Pedestrian Detection Algorithm

The flowchart of pedestrian detection algorithm is shown in Figure 8.

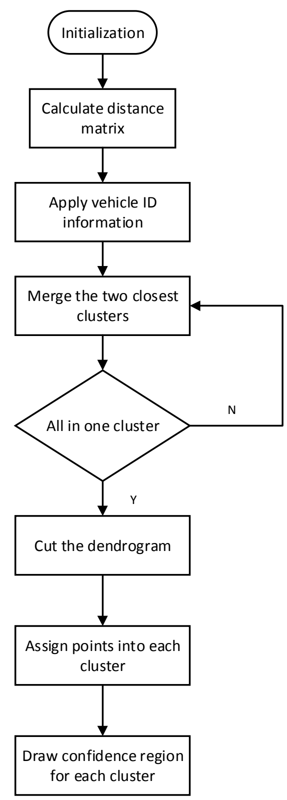

Figure 8. Flowchart of pedestrian detection algorithm.

The algorhm has 8 steps as follows:

1. Initialize each point to be its own cluster
2. Calculate Euclidean distance between each cluster
3. Assign large distance between same vehicle ID by using the maximum distance
4. Merge the two closest clusters by using ward's method
5. Repeat step 4 until all points are inside one cluster
6. Choose a distance D to cut the dendrogram
7. Group points into each cluster, by using the cutting result from step 6
8. Draw confidence region for each cluster

SIMULATION AND DISCUSSIONS

One Case Simulation

In the one case simulation, we have 8 pedestrians with pedestrian ID a, b, c, d, e, f, g, h and 10 vehicles with vehicle ID A, B, C, D, E, F, G, H, I, J. The maximum spread range from the detected position to true position is set to 2.

Figure 9 shows a sample case of the simulation with pedestrian ID information. The large red spots indicate the true pedestrian position. The colored small spots indicate the Pedestrian ID. As can be seen from the plot, the shape of each point group varies for each pedestrian. Some positions may overlap with each other. For example, the position signals of pedestrian 'a' and 'e' are overlapped. This type of overlapping usually brings extra difficulty in distinguishes each group.

Figure 10 represents the same simulation case as Figure 9 but with the vehicle ID information. The large red spots still indicate the true pedestrian position. The different shapes indicate the Pedestrian ID. It is found that each pedestrian is detected 10 times by 10 vehicles.

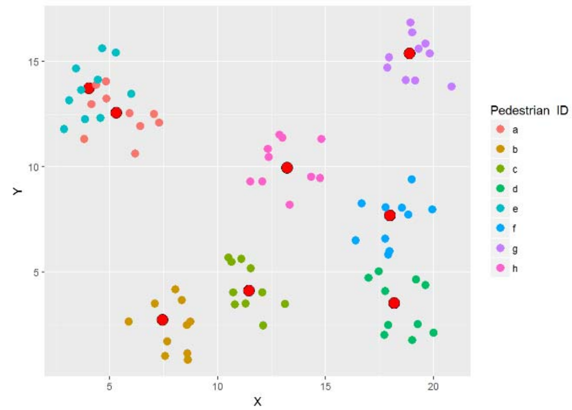

Figure 9. Pedestrian position plot with pedestrian ID.

As a result, a total of 80 signals are obtained. By calculating the Euclidean distance between each signal, we will have an 80 by 80 distance matrix. Table 3 shows a part of the original distance matrix with column 1 to 10 and column 11, 21, 31, 41, 51,

Table 3.
Original distance matrix

1	2	3	4	5	6	7	8	9	10	11	21	31	41	51	61	71
0.00	1.84	3.68	1.33	2.07	3.10	1.94	2.47	3.72	2.94	10.11	8.01	14.83	2.84	12.98	13.41	7.55
1.84	0.00	3.13	0.87	0.46	3.24	1.43	3.55	3.40	2.68	11.36	8.78	14.57	3.43	12.36	12.02	7.21
3.68	3.13	0.00	2.65	2.71	1.27	1.86	2.92	0.48	0.81	13.76	11.63	17.68	2.05	15.37	14.31	10.32
1.33	0.87	2.65	0.00	0.85	2.49	0.79	2.68	2.82	2.04	11.33	9.00	15.22	2.58	13.11	12.90	7.86
2.07	0.46	2.71	0.85	0.00	2.93	1.13	3.45	3.01	2.32	11.79	9.24	14.97	3.20	12.72	12.19	7.62
3.10	3.24	1.27	2.49	2.93	0.00	1.81	1.69	0.94	0.75	12.94	11.10	17.70	0.78	15.58	15.02	10.34
1.94	1.43	1.86	0.79	1.13	1.81	0.00	2.44	2.04	1.27	12.03	9.78	15.96	2.08	13.79	13.31	8.60
2.47	3.55	2.92	2.68	3.45	1.69	2.44	0.00	2.63	2.19	11.55	10.05	17.27	0.93	15.44	15.57	10.02
3.72	3.40	0.48	2.82	3.01	0.94	2.04	2.63	0.00	0.79	13.73	11.72	17.97	1.72	15.71	14.76	10.61
2.94	2.68	0.81	2.04	2.32	0.75	1.27	2.19	0.79	0.00	12.98	10.93	17.23	1.41	15.03	14.31	9.87

61 and 71. The numbers in this table is the original computed pairwise distance.

For each vehicle, it detected all eight pedestrians. As for vehicle A, it generates eight signals from Aa to Ah. The pairwise distance between Aa and other seven signals from vehicle A matched the column 11, 21, 31, 41, 51, 61, and 71 in the distance matrix. Vehicle ID information is used for preventing Aa of merge with signal Ab to Ah. A proportion of maximum distance is assigned to the distance from Aa to Ab, Ac, …, and Ah. The updated distance matrix is shown in Table 4. The red part is assign larger values than previous. We keep assigning large distance through the entire matrix for all ten vehicles. The proportion is decided by a block coefficient which will be discussed in the next section.

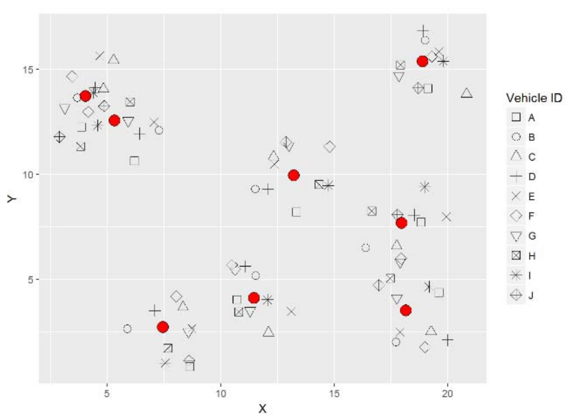

Figure 10. Pedestrian position plot with vehicle ID.

By combining each point from bottom to up, a dendrogram is created as shown in Figure 11.

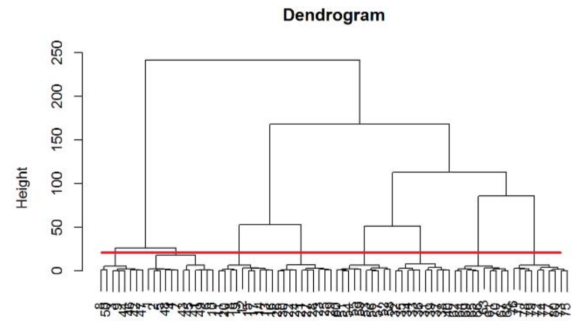

Figure 11. Dendrogram by hierarchal clustering.

Within the dendrogram, the X axis shows all the data points, the Y axis shows distance between each pair of clusters. After cutting the tree along the red line, we will have 8 clusters. The cut height is decided by the proportion of the maximum distance. The scale rate is given by the cut parameter. The grouping result is shown in Table 5.

Table 5.
Grouping result

Cluster number	1	2	3	4	5	6	7	8
Number of points	8	12	10	10	10	10	10	10

From the grouping result, it is noticed that group 3 to group 8 perfectly retrieve the pedestrian signals.

Table 4.

Updated	2	3	4	5	6	7	8	9	10	11	21	31	41	51	61	71
0.00	1.84	3.68	1.33	2.07	3.10	1.94	2.47	3.72	2.94	31.18	31.18	31.18	31.18	31.18	31.18	31.18
1.84	0.00	3.13	0.87	0.46	3.24	1.43	3.55	3.40	2.68	11.36	8.78	14.57	3.43	12.36	12.02	7.21
3.68	3.13	0.00	2.65	2.71	1.27	1.86	2.92	0.48	0.81	13.76	11.63	17.68	2.05	15.37	14.31	10.32
1.33	0.87	2.65	0.00	0.85	2.49	0.79	2.68	2.82	2.04	11.33	9.00	15.22	2.58	13.11	12.90	7.86
2.07	0.46	2.71	0.85	0.00	2.93	1.13	3.45	3.01	2.32	11.79	9.24	14.97	3.20	12.72	12.19	7.62
3.10	3.24	1.27	2.49	2.93	0.00	1.81	1.69	0.94	0.75	12.94	11.10	17.70	0.78	15.58	15.02	10.34
1.94	1.43	1.86	0.79	1.13	1.81	0.00	2.44	2.04	1.27	12.03	9.78	15.96	2.08	13.79	13.31	8.60
2.47	3.55	2.92	2.68	3.45	1.69	2.44	0.00	2.63	2.19	11.55	10.05	17.27	0.93	15.44	15.57	10.02
3.72	3.40	0.48	2.82	3.01	0.94	2.04	2.63	0.00	0.79	13.73	11.72	17.97	1.72	15.71	14.76	10.61
2.94	2.68	0.81	2.04	2.32	0.75	1.27	2.19	0.79	0.00	12.98	10.93	17.23	1.41	15.03	14.31	9.87

However, for group 1 and group 2, 2 points from pedestrian 1 are mistakenly assigned to pedestrian 2 (group 2). This type of error is tolerable, as we are more focusing on whether or not the safe region will cover the person's true position.

The safe region is present in Figure 12. In the figure, it can been seen that even though group 1 and group 2 have signals misassignment, the safe region still covers the pedestrian's true position. The centers of the group 1 and 2 are quite close to their real position. For group 3 in the left bottom, though we exactly retrieve all its signals, the group center is relative far away from the pedestrian's true position, due to the high variance of the signals. However, the confidence region is still able to cover the original position.

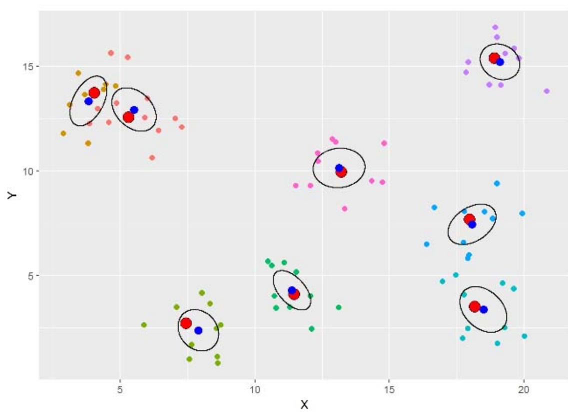

Figure 12. Safe region plot.

Multiple Cases Simulation

By choosing different proportion of block parameter and cut parameter, the clustering algorithm would have different accuracy rate. In this simulation, each case is simulated 1000 times to test the accuracy rate. True position inside the ellipsoid is considered to be passing. If true position lies out of the safe region, the case is considered to be failure. The simulation results with different parameters are presented in Figure 13. The best average accuracy we get within all 441 cases 1000 simulation is 99.4%.

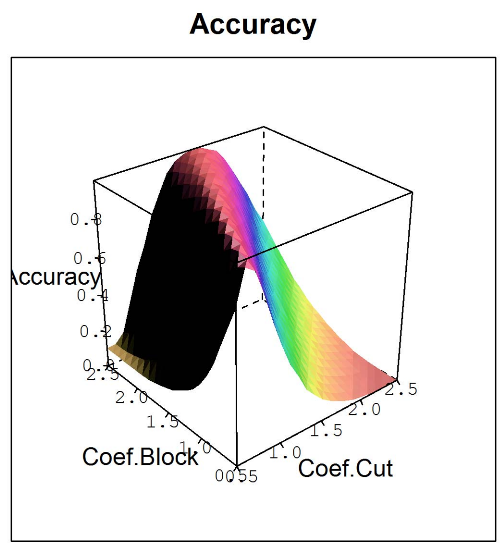

Figure 13. Three dimensional plot Accuracy Rate.

Figure 14 provides the contour plot of the accuracy rate with the combination of cut coefficient and block coefficient. It projects the 3 dimensional information of Figure 13 into a 2 dimensional space. The brighter color represents the high accuracy. The white part covers the accuracy beyond 95%. The green color represents relative low accuracy. Low accuracy region is in the up-left corner and lower-right corner of the Figure 14. From Figure 14, it is also found that the pedestrian detection algorithm is able to achieve a high accuracy above 95 % (the white part) by choosing a proper block coefficient. As we can see that the white region becomes wider as block coefficient grows. We conclude that larger block coefficient leads to higher likelihood of accuracy if we choose the range between 0.48 and 0.68.

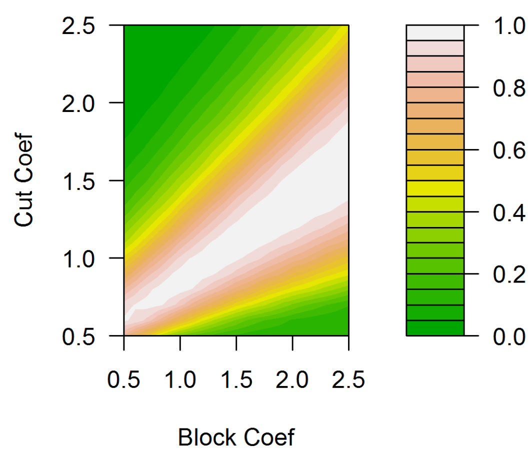

Figure 14. Contour plot Accuracy Rate.

Conclusions

As PAEB and V2V technologies are becoming mature, sending PAEB detected pedestrian information to the V2V network provides a potential benefit to make safety decisions earlier and more effective. This paper has provided a solution for a specific pedestrian data fusion problem in the V2V-PAEB system by using hierarchical clustering algorithm to provide the safe region. A hierarchical clustering algorithm is proposed by using position and vehicle ID information. This method provides an approach of cutting the dendorgram by using a proportion of the maximum Euclidean distance. The results ensure a subject vehicle to approximate the number of pedestrians and their estimated locations from a large number of pedestrian alert messages by many nearby vehicles through the V2V network and the subject vehicle itself. The simulation results have demonstrated the effectiveness and applicability of the proposed method.

This result can be useful for PAEB system to make the warning/braking decisions earlier and hence, improve its pedestrian safety performance. The same idea can be applied to other objects (such as bicyclists) on the road.

REFERENCES

[1] Xu, Qing, et al. "Vehicle-to-vehicle safety messaging in DSRC." Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks. ACM, 2004.

[2] Rosen, Erik. "Autonomous emergency braking for vulnerable road users." IRCOBI Conference. 2013.

[3] Coelingh, Erik, Andreas Eidehall, and Mattias Bengtsson. "Collision warning with full auto brake and pedestrian detection-a practical example of automatic emergency braking." Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on. IEEE, 2010.

[4] Premebida, Cristiano, et al. "A lidar and vision-based approach for pedestrian and vehicle detection and tracking." Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE. IEEE, 2007.

[5] Szarvas, Mate, Utsushi Sakai, and Jun Ogata. "Real-time pedestrian detection using LIDAR and convolutional neural networks." Intelligent Vehicles Symposium, 2006 IEEE. IEEE, 2006.

[6] Tang, Bo. "Pedestrian Protection Using the Integration of V2V Communication and Pedestrian Automatic Emergency Braking System." Thesis. Purdue University, Indianapolis, 2015. Print.

[7] Wang, T., R. Aggarwal, and A. K. Somani. "Human Tracking Using Delphi ESR-Vision Fusion in Complex Environments." Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV). The Steering Committee of The World Congress in Computer Science, 2015.

[8] James, Gareth, et al. An introduction to statistical learning. Vol. 6. New York: springer, 2013.

[9] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The elements of statistical learning. Vol. 1. Springer, Berlin: Springer series in statistics, 2001.

[10] Anderson, Rolph E., Ronald L. Tatham, and William C. Black. "Multivariate data analysis." (1998).

[11] Hair, Joseph F., et al. Multivariate data analysis: A global perspective. Vol. 7. Upper Saddle River, NJ: Pearson, 2010.