Journal of Computer Science and Cybernetics, V.34, N.2 (2018), 113–125
DOI 10.15625/1813-9663/34/2/12655

PEDESTRIAN ACTIVITY PREDICTION BASED ON SEMANTIC SEGMENTATION AND HYBRID OF MACHINES

DIEM-PHUC TRAN[1], VAN-DUNG HOANG[2,a], TRI-CONG PHAM[3], CHI-MAI LUONG[3,4]

[1] *Duy Tan University*
[2] *Quang Binh University*
[3] *ICTLab, University of Science and Technology of Hanoi*
[4] *Institute of Information Technology, VAST*
[a] *dunghv@qbu.edu.vn*

Abstract. The article presents an advanced driver assistance system (ADAS) based on a situational recognition solution and provides alert levels in the context of actual traffic. The solution is a process in which a single image is segmented to detect pedestrians' position as well as extract features of pedestrian posture to predict the action. The main purpose of this process is to improve accuracy and provide warning levels, which supports autonomous vehicle navigation to avoid collisions. The process of the situation prediction and issuing of warning levels consists of two phases: (1) Segmenting in order to definite the located pedestrians and other objects in traffic environment, (2) Judging the situation according to the position and posture of pedestrians in traffic. The accuracy rate of the action prediction is 99.59% and the speed is 5 frames per second.

Keywords. Autonomous vehicle, deep learning, feature extraction, object detection, pedestrian recognition, semantic segmentation.

1. INTRODUCTION

Nowadays, recognition technology on autonomous vehicle (AV) is widely applied in real life. For AV, basic objects have been recognized with high accuracy and specific handling situations. However, of all subjects interacting with AVs in actual traffic, pedestrians are considered to be the most difficult to identify and handle. Consequently, the combination of multiple methods to improve the efficiency in predicting and conducting different levels of classification is absolutely necessary. When a pedestrian joins traffic on the road, there may be many situations of pedestrian behavior such as: crossing, waiting to cross, walking on the pavement, etc. According to the position and posture of pedestrian, different levels of warning is alerted for AV. The process of classifying and providing different levels of warning enables AVs to be active in moving, avoid unexpected accidents, and ensure the speed as well as the journey safety of the car.

2. RELATED WORKS

Recent studies have shown that all objects can be accurately identified using deep learning methods. Some original object recognition models such as: AlexNet [10], GoogleNet, etc. and current advanced solutions including RCNN, Fast-RCNN, Faster-RCNN, etc. are focused on improving the CNN network model to make predictions. However, in terms of robots, it is much more difficult to identify an action, especially to anticipate the object's unpredictable actions to appropriately handle the situation. Similarly, for AV, the pedestrian is considered to be the least accurate prediction object because of two basic factors: no limit of movement and unspecified moving trajectory. Therefore, pedestrian behavior prediction requires a combination of different approaches.

The identification of unmoving objects is considered the most diverse, including recognizing and identifying roadsides and curbs. A possible solution is the detection of roadside vegetation (DRV) [6], which uses a set of color features extracted from the camera image and the support vector machine (SVM) model to identify objects. Besides, in the urban road sections, there are solutions which identify road markers [1, 12] helping automatic vehicles determine the moving trajectory. These solutions focus on the use of Gaussian and Kalman filters in conjunction with the Hogh algorithm to identify the position of road markers serving the automatic direction. Some approaches use inductive devices [17, 18] installed along the curbs and line lanes of the road, allowing AVs to continuously transmit signals and determine the exact direction of the car.

Recently, high accuracy of solutions such as image segmentation [2, 3] color label assigning, and training and identifying on the pixel of the image has helped AVs to identify multiple objects interacting in the frame. In terms of computer vision, the image segmentation is a process in which a digital image is split into many different parts (a set of pixels, also known as super pixels). The target of image segmentation is to simplify or change the image expression into a direction which is more meaningful and easier to analyse. The image segmentation is usually used to identify the position of objects and borders (straight lines or curves).

Table 1. The color map

RGB Color	Objects
$\begin{bmatrix}0 & 255 & 0\end{bmatrix}$	Other objects: tree, building, sky,...
$\begin{bmatrix}255 & 0 & 0\end{bmatrix}$	Road
$\begin{bmatrix}0 & 0 & 255\end{bmatrix}$	Pavement
$\begin{bmatrix}255 & 255 & 0\end{bmatrix}$	Vehicle
$\begin{bmatrix}0 & 255 & 255\end{bmatrix}$	Pedestrian

In other words, image segmentation is a process in which every pixel in an image is assigned a label. Pixels in the same label share similar characteristics in terms of color, image intensity and texture. After the image segmentation, objects in the image are determined in size, location, shape, etc. and continue to be used to identify the objects, predict or train other identification models. Figure 6 simulates the image segmentation between the original image and the segmented one, consisting of five objects defined by the color code in Table 1.

In pedestrian detection task, histograms of oriented gradients (HOG) method is an appropriate solution to be applied in practice [4, 9]. Input image is divided into a grid of small

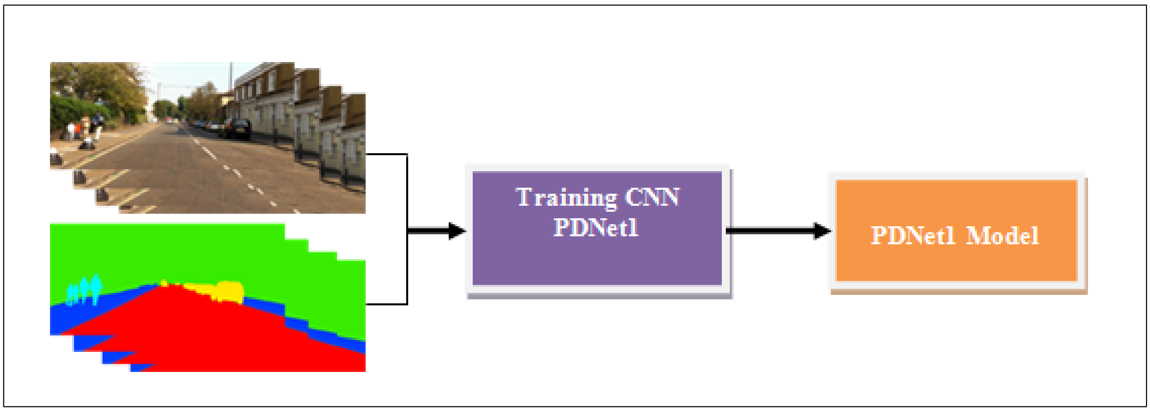


Figure 1. Flowchart of training network CNN PDNet1 to semantic segmentation

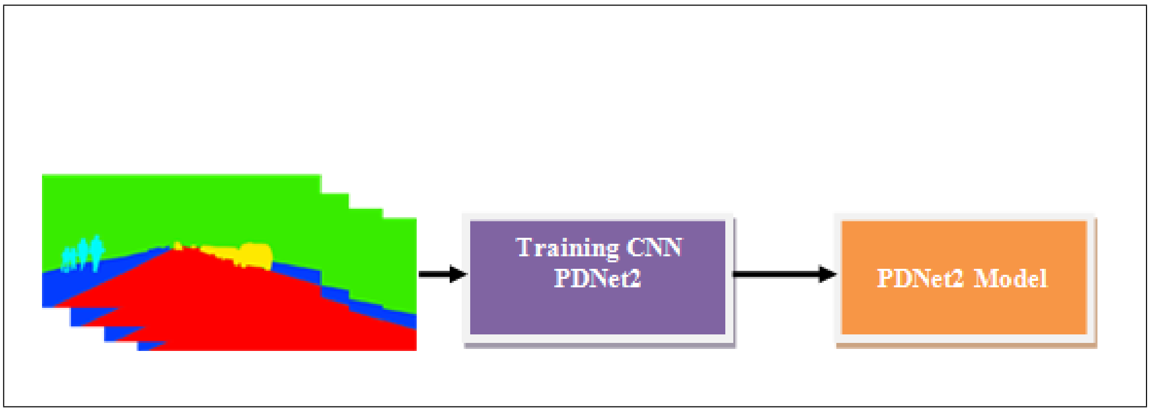


Figure 2. Flowchart of training network CNN PDNet2 to extract features

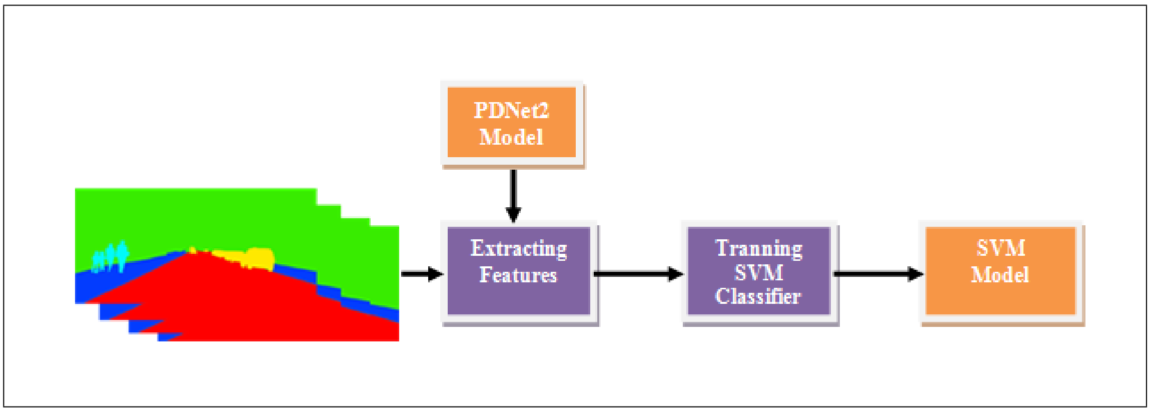


Figure 3. Flowchart of training network CNN PDNet2 to train SVM model

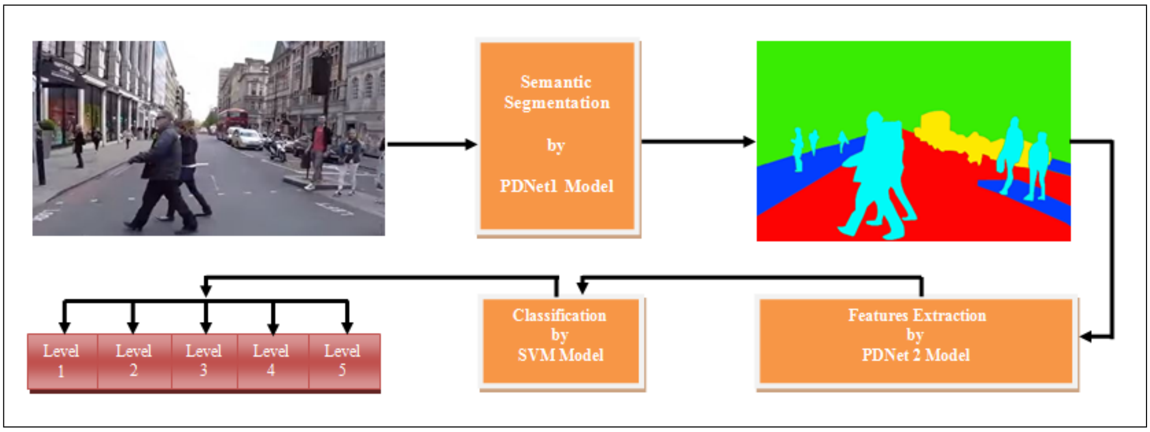

Semantic Segmentation by PDNet1 Model
Features Extraction by PDNet 2 Model
Classification by SVM Model
Level 1
Level 2
Level 3
Level 4
Level 5

Figure 4. Model of predicting action and issuing warning alerts from the actual captured image

regions called cells and HOG features are computed in each cell. Adjacent cells are considered to be grouped into block, which represented to spatial connected regions. The grouping of cells into a block for concatenated features for constructing block of HOG features then is normalized. The set of these features from blocks represents the descriptor known as vector of HOG features. The vector of HOG is fed to SVM machine to make the decision. Other approaches for object detection such as Kanade-Lucas-Tomasi (KLT) [11, 15], Latent SVM [5], etc., whose advantages are low computational power required, simple model and high processing speed. However, the loss of valuable information of the image such as color and sharpness while being processed results in low accuracy.

In the field of feature processing, there are various solutions to the training and extraction of identifying cases. However, the use of CNN to extract features has recently achieved significant results and become the state of the art approach. Widely-used CNN models are AlexNet, GoogleNet, Microsoft ResNet and Region based CNNs (R-CNN, Fast R-CNN, Faster R-CNN), each of which has its own features of specialization, processing speed and accuracy.

In order to optimize the feature extraction, a new PDNet2 model is proposed (Figure 2) and training is conducted on this model. After being trained, PDNet2 model is used to extract features. After that, the features are used to train SVM classification model. Depending on the model and the problem requirements, possible proposed classification algorithms are: k-nearest neighbor (kNN), SVM, random forest, fully connected network, etc. For this article in specific, the proposed of Yichuan Tang [14] have shown that using CNN to extract features and then using the training features for the SVM model brings better performance and lower error rate compared to using the default classification model Fully connection of CNN.

In the other solution of pedestrian action prediction proposed [7, 8, 13], our most recent article [16] addresses the interaction between cars and pedestrians. However, in this proposal of paper, there are only 3 cases in which pedestrian action features are extracted, classified and predicted, including pedestrian crossing, pedestrian waiting and pedestrian walking.

Since the CNN model cannot extract the distinctive features of pedestrian positions and relative positions between pedestrians and AV, it cannot issue detailed warning levels. Despite rather high rate of prediction and high processing speed, CNN alerts are quite not detailed. This results in the fact that CNN model has not yet met the actual automatic requirements, affecting the journey safety and travel time of the vehicle.

In short, a general solution to the "complex" relationship between AVs and pedestrians is essential to ensure safety and mobility.

3. PROPOSED APPROACH

3.1. Generalized solution

Based on research and experimentation, we propose a pedestrian action prediction model and provide a two-step warning level:

(1) Training the CNN models for image semantic segmentation and to extract features:

   (a) Training the CNN PDNet1 model for image semantic segmentation identification (Figure 1);

   (b) Training the CNN PDNet2 model to extract features of labeled image dataset and applied training features to SVM classification model (Figure 2).

(2) Predicting pedestrian action, pedestrian situation and setting alert level (Figure 4), including:

   (a) Semantically segmenting the input image and identifying five objects in the image (road, pavement, cars, pedestrians, other objects);

   (b) Extracting features of the segmented image, applying the SVM classification model to predicting pedestrian actions and situations (Figure 3);

   (c) Issuing a warning level.

3.2. Training CNN model for image semantic segmentation

Rather than focusing on semantic segmentation (step 1, Figure 1) this paper supposes that the results of semantic segmentation are acceptable with high accuracy. A classification machine is built and analyzed relationships of objects (pedestrians, road, pavement, etc.) are analyzed (step 2 - Figure 2). Recently, image fragmentation has been heavily researched and constructed with large data sets, resulting in very high accuracy rates with many different objects appearing in the video frame [2, 3]. However, to better understand the nature of the overall model, a model of our own has also been built and trained with relative precision. A CNN model of 25 layers (Figure 7) is proposed. This CNN model includes 5 convolution layers, 32 filters sized [7 × 7] and an input image sized [180 360 3]. The initial training data set consists of 3,000 input images, including an original image set and a labeled image set

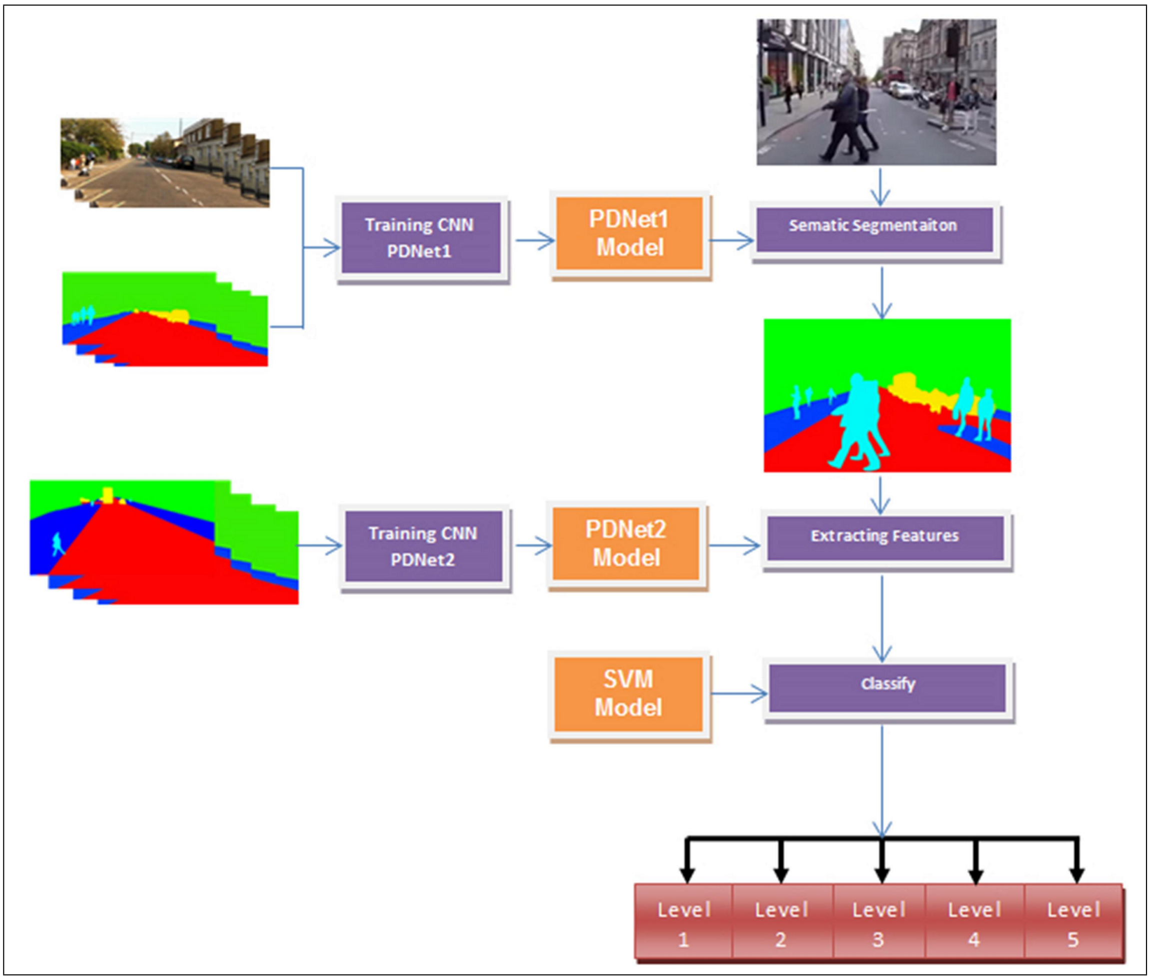
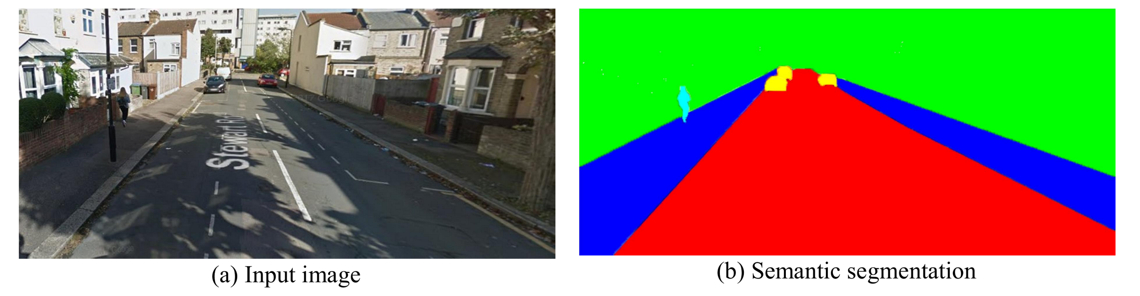


Figure 5. The general schema of algorithm

(Figure 6). Each labeled image is segmented into five basic objects: pedestrians, cars, road, pavement and other objects corresponding to five RGB color codes (Table 1). In order to speed up the training and identification, buildings, trees, sky, etc. are grouped into other objects. In addition, to improve the quality of identification, it is proposed that the number of images be increased based on data augmentation technique such as rotating the image horizontally (flip), adjusted tilt, added noise, etc. about 3 times. The total number of training samples is 3,000.

When AVs move on the road, pedestrian detection in the validation view commences the process of the system (Figure 4). Therefore, accurate detection of pedestrians becomes very important. In the previous article [16], pedestrian detection has been specifically analyzed using aggregate channel features (ACF).

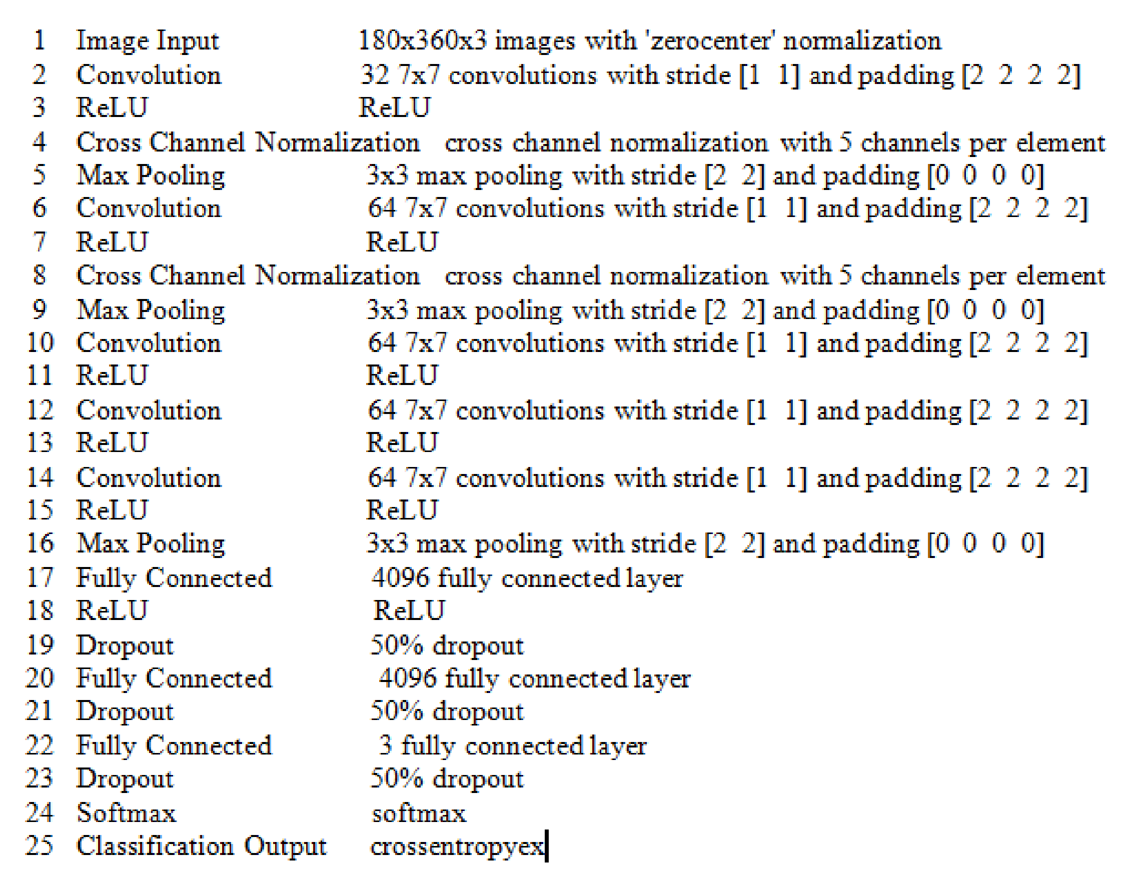
However, the experimental process has shown that the ACF algorithm ignores some of the pedestrian detection cases. Therefore, the proposed solution is to use the segmented image to determine the presence of pedestrians. Pedestrians are considered to appear in the frame



(a) Input image

(b) Semantic segmentation

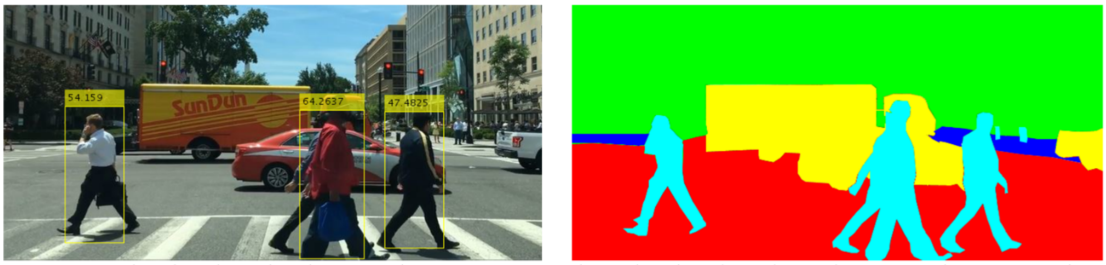
Figure 6. Simulation of the original dataset and the labeled set



1	Image Input	180x360x3 images with 'zerocenter' normalization
2	Convolution	32 7x7 convolutions with stride [1 1] and padding [2 2 2 2]
3	ReLU	ReLU
4	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
6	Convolution	64 7x7 convolutions with stride [1 1] and padding [2 2 2 2]
7	ReLU	ReLU
8	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10	Convolution	64 7x7 convolutions with stride [1 1] and padding [2 2 2 2]
11	ReLU	ReLU
12	Convolution	64 7x7 convolutions with stride [1 1] and padding [2 2 2 2]
13	ReLU	ReLU
14	Convolution	64 7x7 convolutions with stride [1 1] and padding [2 2 2 2]
15	ReLU	ReLU
16	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17	Fully Connected	4096 fully connected layer
18	ReLU	ReLU
19	Dropout	50% dropout
20	Fully Connected	4096 fully connected layer
21	Dropout	50% dropout
22	Fully Connected	3 fully connected layer
23	Dropout	50% dropout
24	Softmax	softmax
25	Classification Output	crossentropyex

Figure 7. CNN PDNet1 network structure of image semantic segmentation

when the coating or the number of constant pixels in specified colors [0, 255, 255] (Table 1) appears at a certain rate compared to the color of the road and pavement. Experimentation has illustrated that 100% of pedestrians are correctly detected when they appear in AVs' moving frame (Figure 8).



a. Pedestrian detection by ACF detector

b. Pedestrian detection using semantic segmentation

Figure 8. Comparison between pedestrian detection using ACF and semantic segmentation

3.3. Training PDNet2 network and extracting postures and positions of pedestrians

The output of PDNet1 is the input data of PDNet2. The purpose of using the PDNet1 model is for semantic segmentation, which indicates objects' location and area such as pedestrian, vehicle, pavement, road and other objects (tree, building, sky,...). The result will then be used as input of PDNet2 model to analyze the relation between them and make predictions about the situation, ensuring traffic safety. In order to create the training data for PDNet2 network learning, segmented images are divided into three situations of pedestrians crossing, pedestrians walking on pavement and pedestrians waiting to cross the road. With the "pedestrian crossing" and "pedestrian waiting" case, pedestrians are divided into two cases: pedestrians close to the vehicle and pedestrians far away from the vehicle. Thus, there are five datasets labeled corresponding to five warning situations, which include: Alert 1: Pedestrian crossing 2 - pedestrian is crossing in the near-front of the AV; Alert 2: Pedestrian crossing 2 - pedestrian is crossing in far-front the AV; Alert 3: Pedestrian waiting 1 - pedestrian is waiting near the AV; Alert 4: Pedestrian waiting 2 - pedestrian is waiting far the AV; Alert 5: Pedestrian walking. The detection (far or near AV) is based on the location and number of pixels identified through the PDNet1 model.

Our experiment pointed out that the network with the fully connection layer for classifying achieved low accuracy, which is inappropriate for practical application. Therefore, the PDNet2 model is only used to extract features, which is fed to SVM for alert situation prediction. The set of data for SVM training consists of five classes, which includes 5000 images (Alert 1, Alert 2, Alert 3, Alert 4 and Alert 5), as shown in Table 2. In this system, alert levels are predicted with expectation based on the relative position between the pedestrian and the road, pavement, and pedestrian posture when moving on the street as illustrated in Figure 9.

The pedestrian location is determined by percentage of occupied area on road and pavement, indicating the distance between a pedestrian and the AV. In addition, the location of the pedestrian pixels also illustrates the pedestrian's state. If the pixels appear on the ground of the roadway, the pedestrian is crossing the road. Otherwise, the pedestrian is waiting to cross the road or walking along the sidewalk.

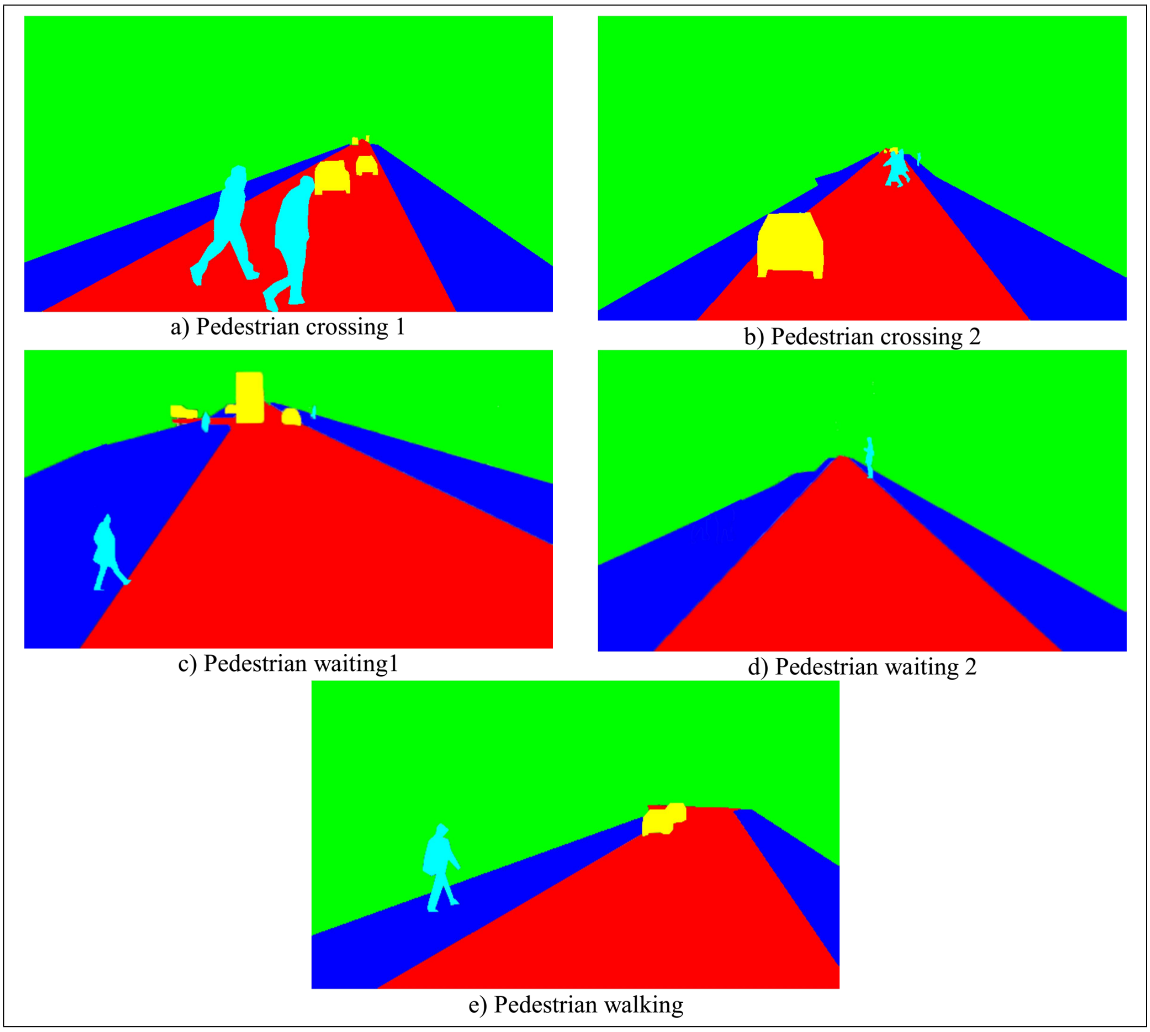


Figure 9. Simulation of training data sets of PDNet2

3.4. Training SVM classification model

After extraction, these features continue to be extracted at the 20th layer (fully connected - Fc2) of the PDNet2 model and to train the support vector machine (SVM) classification model. The aim of combining PDNet2 and SVM is to improve the accuracy of recognition and warning systems for drivers. In particular, PDNet2 is used for features extraction purpose at the last layer of PDNet2 while SVM is used for classification of alert levels. Following the traditional approach, deep learning network is used for both specific features extraction and sample classification, in which accuracy reaches 78%-83%. In order to improve accuracy, we propose an approach combining two machine learning techniques and PDNet2 for features extraction and SVM to classify alert levels. In this way, accuracy increases to 99% when evaluated on the same dataset.

Table 2. The case and order of alert

Alert	Pedestrian Status	Description
Alert 1	Pedestrian crossing 1 (PC1)	Pedestrian crossing and distance between pedestrian and small vehicle.
Alert 2	Pedestrian crossing 2 (PC2)	Pedestrian crossing and distance between pedestrian and big vehicle.
Alert 3	Pedestrian waiting 1 (PW1)	Large currents pedestrian waiting to cross and Distance between the pedestrian and small vehicle.
Alert 4	Pedestrian waiting 2 (PW2)	Pedestrian waiting to cross and distance between the pedestrian and big vehicle.
Alert 5	Pedestrian walking (PW)	Pedestrian walking along the pavement.

Table 3. Images and labels dataset to train PDNet1

Class	Quantity
Original image	3,000
Segmented image	3,000

3.5. Deciding a warning level

Generally, as shown in Figure 4, each input image received is semantically segmented by PDNet1-trained CNN model. After being segmented, the input image is processed into five basic RGB colors according to objects (Table 1). In the image segmentation process, in case of pedestrian appearing in frame of AV, the system starts the process of predicting actions and recognizing the pedestrian situation. The image is continued to be extracted using the PDNet2 pre-trained CNN network model and then be used to predict action and situation based on the SVM classification model.

The results of the situation prediction include five alert levels in Table 2, representing the five datasets that have been extracted features and trained for the SVM classification.

4. EXPERIMENTAL RESULTS

4.1. Training the PDNet1

The initial data set for PDNet1 network model training includes 1,000 original and 1,000 semantic segmented images. However, in order to improve the quality of image identification and segmentation, data Augmentation solutions, which uses flip and rotation methods, is proposed. The total number of trained photos is 3,000 (Table 3).

To check the accuracy of the PDNet1 training process, 90% of the data set is used for training and the remaining is used for testing, the result of which is illustrated in Table 4.

Table 4. Test result of image segmentation training

Class	Accuracy	IoU	MeanBFScores
Other Object	0.93502	0.92343	0.72522
Road	0.95232	0.92174	0.80267
Pavement	0.91091	0.62024	0.62024
Vehicle	0.97841	0.437	0.29594
Pedestrian	0.76942	0.111	0.2067

Table 5. Images and labels dataset for train PDNet2 and extract features

Alert	Class	Quantity
Alert 1	Pedestrian crossing 1	1,000
Alert 2	Pedestrian crossing 2	1,000
Alert 3	Pedestrian waiting 1	1,000
Alert 4	Pedestrian waiting 2	1,000
Alert 5	Pedestrian walking	1,000

4.2. Training the PDNet2

The original PDNet2 training dataset consists of 5,000 images, which are divided equally into five cases (Table 5).

Experimental results of PDNet1 model training show that the accuracy rate is approximately 69%-71% when using fully connected to classify.

4.3. Extracting features and training SVM model

After being trained, the PDNet2 model continues to be used to extract features on the dataset in Table 5. The extracted features continue to be trained for the SVM subclass model. 90% of the images are used for training and the remaining 10% for accuracy. The result is illustrated in Table 6. Processing speed for each pedestrian case from being detected to action prediction reaches 5 frames per second with the device configuration as in Table 7.

Table 6. The confusion matrix for pedestrian action prediction

	PC1	PC2	PW1	PW2	PW
PC1	**0.9959**	0	0.0010	0.0010	0.0020
PC2	0.0030	**0.9767**	0.0010	0.0142	0.0051
PW1	0.0010	0	**0.9746**	0.0030	0.0213
PW2	0	0.0061	0	**0.9473**	0.0467
PW	0	0	0.0325	0.2049	**0.7627**

Table 7. The configuration of device to test the speed of process

Device	Description
CPU	I3 3.6Ghz
GPU	Geforce 1060 6GB
RAM	16GB
HDD	SSD 160GB

5. CONCLUSION

In general, this solution brings high accuracy in pedestrian detection as well as pedestrian location and the relative distance between pedestrians and AVs. The combination of image semantic segmentation and pedestrian extraction provides the possibility of accurately predicting the situation, assisting in issuing alerts to AVs.

Therefore, our proposed solution has certain advances and innovations:

(1) Description use only single images received to conduct the pedestrian action prediction, situation recognition and warnings (do not use video to track pedestrians, thus speeding up processing yet maintaining high accuracy).

(2) High accuracy (with maximum accuracy ∼ 100% and minimum accuracy ∼76%) thanks to the application of image semantic segmentation before extraction pedestrian action prediction and situation prediction.

Accurate identification with an accuracy rate of ∼76% corresponding to Pedestrian walking is consistent with the actual training data. As pedestrians move along the pavement, PDNet2 model extracts pedestrian posture characteristics. However, they are easily confused with Pedestrian waiting 1 and Pedestrian waiting 2, when pedestrians are waiting or about to cross.

In the near future, it is possible to apply the image segmentation process to many objects appearing and moving on the road in detail, such as: cars (cars, trucks, container cars, etc.), motorcycles, bicycles, etc. in setting different alert levels in combination with pedestrian alerts. Although this may not be the best solution, the article's recommendations may be a guide for AVs under limited conditions of present hardware's speed and size.

ACKNOWLEDGMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.05-2015.09.

REFERENCES

[1] M. Aly, "Real time detection of lane markers in urban streets," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 7–12.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *arXiv preprint arXiv:1511.00561*, 2015.

[3] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[4] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[5] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[6] I. Harbas and M. Subasic, “Detection of roadside vegetation using features from the visible spectrum,” in *37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).* IEEE, 2014, pp. 1204–1209.

[7] J. Hariyono and K.-H. Jo, “Detection of pedestrian crossing road: A study on pedestrian pose recognition,” *Neurocomputing*, vol. 234, pp. 144–153, 2017.

[8] V.-D. Hoang, “Multiple classifier-based spatiotemporal features for living activity prediction,” *Journal of Information and Telecommunication*, vol. 1, no. 1, pp. 100–112, 2017.

[9] V.-D. Hoang, M.-H. Le, and K.-H. Jo, “Hybrid cascade boosting machine using variant scale blocks based hog features for pedestrian detection,” *Neurocomputing*, vol. 135, pp. 357–366, 2014.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[11] B. D. Lucas, T. Kanade *et al.*, “An iterative image registration technique with an application to stereo vision,” 1981.

[12] R. Satzoda and M. Trivedi, “Vision-based lane analysis: Exploration of issues and approaches for embedded realization,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 604–609.

[13] R. Stewart, M. Andriluka, and A. Y. Ng, “End-to-end people detection in crowded scenes,” in *IEEE conference on computer vision and pattern recognition*, 2016, pp. 2325–2333.

[14] Y. Tang, “Deep learning using linear support vector machines,” *arXiv preprint arXiv:1306.0239*, 2013.

[15] C. Tomasi and T. Kanade, “Detection and tracking of point features,” 1991.

[16] D.-P. Tran, N. G. Nhu, and V.-D. Hoang, “Pedestrian action prediction based on deep features extraction of human posture and traffic scene,” in *Asian Conference on Intelligent Information and Database Systems.* Springer, 2018, pp. 563–572.

[17] H. Wang, W. Quan, Y. Wang, and G. R. Miller, “Dual roadside seismic sensor for moving road vehicle detection and characterization,” *Sensors*, vol. 14, no. 2, pp. 2892–2910, 2014.

[18] Q. Wang, J. Zheng, H. Xu, B. Xu, and R. Chen, “Roadside magnetic sensor system for vehicle detection in urban environments,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1365–1374, 2018.

Received on June 10, 2018
Revised on July 30, 2018