Will Capsule Networks overcome Convolutional Neural Networks on Pedestrian Walking Direction ?*

Safaa Dafrallah, Aouatif Amine, Stephane Mousset, Abdelaziz Bensrhair

Will Capsule Networks overcome Convolutional Neural Networks on Pedestrian Walking Direction ?*

Safaâ Dafrallah[1], Aouatif Amine[1], Stéphane Mousset[2] and Abdelaziz Bensrhair[2]

***Abstract*— Thousands of people are dying every year due to road accidents; in fact 23% of world fatal accidents are pedestrians related, where 40% of them occur in Africa as reported by the World Health Organisation (WHO). Predicting the walking direction of a pedestrian could help to avoid an eventual accident. Existing studies can not handle pose and orientation transformations of the input object contrary to our proposed method. This paper describes a novel approach to determine the pedestrian orientation using Capsule Networks (CapsNet) based scheme. CapsNet are a new deep learning architecture that overcome some limitations of the existing studies, they are group of neurons invariant to rotation and affine transformations, which represent a specific interest to this work. Capsule Networks predicts the walking directions of pedestrians to prevent such mortal accidents, using four main walking directions (front, back, left and right).For this purpose, a new pedestrians dataset gathered from the most popular cities in Morocco is collected to be studied and used as a proof of the proposed approach. To enhance this proposed approach, we evaluated it using Daimler dataset and compared it to Convolutional Neural Networks (CNN) architectures.**

Experimental results reveal that the performance of the proposed approach reaches an accuracy of 97.60% on daimler dataset and 73.64% on our Moroccan collected dataset.

I. INTRODUCTION

Road traffic injuries are the first cause of death for those who are under 30 years. Approximately 1.35 million person dies annually around the world from road traffic accidents, where 23% of victims are pedestrians [1]. According to the World Health Organisation (WHO), 93% of the world fatalities on the road occur in low and middle income countries [2]. These countries have approximately 60% of the world's vehicles and a high population size.

In order to reduce fatal accidents involving pedestrians, multiple researches are implemented in Advanced Driver Assistance Systems (ADAS) specifically for Pedestrian Crash Avoidance Mitigation (PCAM) [3]–[6]. To that end, researches in PCAM systems focus more on detecting pedestrians than on predicting their walking direction, hence only few ones intend to include pedestrian orientation to those systems. Additionally, existing PCAM systems are designed for well structured areas containing road signs and floor markings, however in low and middle income countries roads are usually poorly structured.

In this paper, Morocco is chosen as a case study, where 28% of road fatal injuries are pedestrian related with 996 deaths in 2016, as reported by the Moroccan Ministry of Environment, Transport, Logistics and Water (METLE) [7].

In this context, we collected a new dataset of moroccan pedestrians from two cities (Rabat and Kenitra). After analyzing this data, we find that traffic laws are less respected by both drivers and pedestrians, where the common way of pedestrians crossing was arbitrarily, specially in poorly structured areas. As a result of this behavior, a high rate of pedestrian accidents is noticed in Morocco.

As a solution, we propose to include pedestrian walking direction to PCAM systems using the moroccan collected dataset. Note that this work is supported by the METLE, in collaboration with the National Center for the Scientific and Technical Research (CNRST).

The remainder of this paper is organized as follows: Section II discussed the Related Work. Section III describes the proposed approach and the collected dataset. Experimental results and discussion are in section IV, and finally the conclusion and future works in section V.

II. RELATED WORK

To predict the pedestrian walking direction the system has first to detect the pedestrian. Starting from hand crafted features to deep learning methods the pedestrian detection field has known remarkable improvements in the last decade.

Dalal et al. [8] proposed a human detection approach based on Histogram of Oriented Gradients (HOG) descriptor, that gives a performance up to 89%. Whereas, Viola et al. [9] used Haar-like Wavelets to detect pedestrians with a rate of 80% and a false positive rate of 1 for every 2 frames. Dollár et al. [10] proposed the Aggregated Channel Features (ACF), that aims to compute 10 channels from an input image, those channels represent the normalized gradient magnitude, Histogram of Oriented Gradients and LUV color. The average miss rate of ACF is about 41%.

Nowadays, automated feature extraction also known by deep learning methods, have shown competitive results for pedestrian detection. As an instance, Bunel et al. [11] used the Convolutional Neural Network (CNN) to detect small scale pedestrians that are at far distance from the camera (30 pixels or less). This method reachs a miss rate lower than 10%. Faster Region-based Convolutional Neural Network (Faster R-CNN) was used by [12] for pedestrian detection and achieves an accuracy of 92.7%. While, Redmon et al. [13] proposed a new approach for real-time object detection called You Only Look Once(YOLO), which gives an accuracy of 57.9% on COCO dataset but still 1000x faster than R-CNN and 100x faster than Fast R-CNN. Tian et

*This work was supported by Moroccan METLE and CNRST

[1] Systems Engineering Laboratory (LGS), National School of Applied Sciences (ENSA), Ibn Tofail University, Kenitra, Morocco.

[2] Computer, Information Processing and Systems Laboratory (LITIS), National Institute of Applied Sciences (INSA), Normandie University, Rouen, France.

al. proposed in [14] a multi-task deep model for pedestrian detection that achieves a log-average miss rate of 34.99%.

Once the detection is done, bounding boxes are extracted and used as an input to the pedestrian orientation prediction network. In this point, Gandhi et al. [15] proposed a Hidden Markov Model (HMM) to model transitions between pedestrian orientations over time and applied the Support Vector Machine (SVM) to estimate the discrete probability distribution of the orientation using 8 orientation bins. This method achives an accuracy of 49.7% for images that fall in the same bins as the ground truth and 81.3% for images that fall in both same or adjacent bins. Shimizu et al. [16] proposed as well an SVM based approach to classify the pedestrians walking directions into 16 directions with a difference of 22.5 degrees. This approach classifies more than 90% of test images into the correct direction. The use of Deep Convolutional Neural Network (DCNN) by Hara et al. [17] gives an accuracy of 70.6% for a predicted orientation within 22.5 degrees from the ground truth and of 86.1% within 45 degrees. In [18], Sanchez et al. used various state of the art CNN networks such as AlexNet, GoogleNet and ResNet for pedestrian movement direction recognition. This approach first compute the optical flow of the input image to estimate the direction of the pedestrian as one of the three predefined directions (left, right and front), and use afterwards the output image as an input to the CNN architecture. As a result, ResNet achieves the best accuracy of 94% on the validation set.

In spite of the good performing results in pedestrian detection and orientation prediction, Convolutional Neural Network has as well some limitations. In fact, the use of the pooling layer makes CNN losing valuable information about the relative position and the orientation of the object, which makes CNN invariant for object translation and rotation as reported in [19] by Lecun et al. As a solution, Capsule Networks was proposed by the team of G. Hinton in [20] achieving a test error of 0.25% on MNIST dataset, they were applied in several research fields such as Traffic Signs detection [21] with an accuracy of 97.6%. To the best of our knowledge, the CapsNet was never used on pedestrian orientation classification.

In this paper, we proposed a novel pedestrian walking direction prediction approach, based on Capsule Networks that takes as an input pedestrian bounding boxes detected using YOLO algorithm since it produces fast and good results on pedestrians detection.

III. PROPOSED APPROACH

The aim of this work is to decrease pedestrian fatal accidents, by making a new ADAS system taking into consideration pedestrians orientation. This system detects pedestrians when moving around the danger zone of the vehicle in poorly structured areas, and predict thereafter the pedestrian direction. In this paper, the effective pedestrian movement has not a proper interest due to moving vehicles, which may give a stable status to the pedestrian and makes his orientation intention more important.

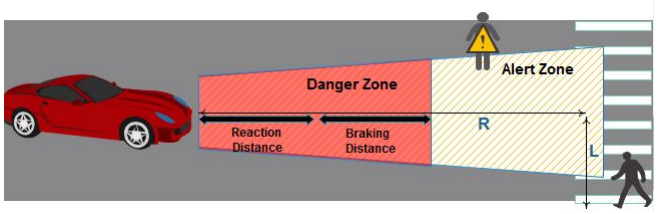


Fig. 1: General scheme of a vehicle and a pedestrian in a danger and alert zone

The danger zone is defined as an area where the vehicle cannot stop in time. In other words, it is the area that is less than or equal to the stopping distance of the vehicle, where:

$$Stopping\ Distance = Reaction\ Distance + Braking\ Distance \quad (1)$$

However, the focus of this paper is on the outer zone, which will be designated as alert zone (see Fig. 1).

According to Massachusetts driver's manual, the two-second rule is a minimum safe distance for good road conditions and moderate traffic. Whereas, for more safety three to four seconds must be added. Thus, the alert zone of four seconds is used for the system which gives a distance of 64m for a vehicle speed of 60km/h.

An accident between a vehicle and a pedestrian is more likely to occur when the pedestrian walks towards the vehicle within the alert zone. The general scheme of a vehicle and a pedestrian in a danger and alert zone is defined based on the vehicle speed as well as the distance between the vehicle and the pedestrian, which is outlined by the lateral distance called L and the longitudinal distance named R as shown in Fig. 1.

In this study, the system is considered as three major factors: Firstly, the number of people in the nearby zone is considered to be one of the main attributes of the road. Therefore, the areas are going to be divided in high an low density areas. Another major factor is the direction of the vehicle itself, it must deduce the surrounding danger regarding the actual direction of the vehicle whether it is going straight or taking a turn. Finally, the orientation of the pedestrian is regarded as the third major factor. In fact, this work shrinks the pedestrians' moving directions to four major ones, that are moving forward and backward inside the pavement, and crossing the street left to right and right to left. Actually, this first part of the approach mainly emphasises the movement of pedestrians in all the described directions within an environment of low density area and up-front moving vehicle.

This system regards pedestrians walking either within the sidewalks or crossing the road. The first case is considered to be safe; therefore, no intervention is needed. On the other hand, pedestrians crossing the road are more likely to get involved in mortal accidents. Hence, the system should alert the driver to pay more attention to the upcoming danger. Thus, not only the detection of pedestrians is a major requirement, but also the walking direction, which is the main concentration of this approach, is fundamental for this

system to work properly. In order to select only pedestrians in risky situations, a pedestrian orientation prediction algorithm based on Capsule Networks is proposed.

In this current section, we will firstly introduce Capsule Networks, then we will describe our moroccan collected dataset used as an input to the network, and finally we will present our proposed architecture for pedestrians' walking direction.

A. Capsule Networks

A capsule is a group of neurons whose activity vectors represent the pose parameters of an entity, and the vectors length represents the existence probability of that entity. Unlike the convolutional network, capsules conserve detailed information about the location and the pose of the entity. A slight rotation of the image involved a slight change in the activation vector. The capsule network as represented in [20], contains two convolutional layers and one fully connected layer. The first convolutional layer extracts feature maps from the input image. The result is then resized to an array of vectors with components, which constitute the input to the second convolutional layer called primary capsule layer. The vector length must be between 0 and 1 since it represents the existence probability of an entity. Hence, a squash function is applied to shrunk the long vectors to nearly one and the short vectors to almost zero.

The main role of the third layer, which is the fully connected layer, is classification. This latter contains a capsule per class, where each capsule has a belonging probability for all classes and the class having the highest probability, is assigned to the capsule. Capsules from the primary capsule layer (second layer) predict the output vectors of the fully connected layer (third layer). The prediction $\hat{u}_{j/i}$ is produced by multiplying the output vector u_i of a capsule in the second layer with a transformation matrix W_{ij}:

$$\hat{u}_{j/i} = W_{ij} u_i \tag{2}$$

The transformation matrix is learned by the network gradually using backpropagation in the learning process of the primary capsule layer. Thereafter the agreement a_{ij} between the prediction value made by the capsule *i* of the second layer $\hat{u}_{j/i}$ and the current output vector of the capsule *j* in the third layer v_j, is calculated using a dot product as represented in(3)

$$a_{ij} = \hat{u}_{j/i}.v_j \tag{3}$$

For each predicted vector, a routing weight is used called b_{ij} and initialized by zero for all capsules in the both layers. Then a softmax function c_{ij} is applied to that routing weight for each capsule in the primary capsule layer. The weighted sum s_j of all prediction vectors, is after that calculated for each capsule in the fully connected layer:

$$s_j = \sum_i c_{ij} \hat{u}_{j/i} \tag{4}$$

Then a squash function is applied to that weighted sum, giving as a result the real outputs v_j of capsules of the third layer:

$$v_j = \frac{||s_j||^2}{1 + ||s_j||^2} \frac{s_j}{||s_j||} \tag{5}$$

Subsequently, the routing weight b_{ij} is updated by adding to it the agreement between the predicted and the real vector.

$$b_{ij} = b_{ij} + \hat{u}_{j/i}.v_j \tag{6}$$

This process represents one iteration of the routing algorithm. In case of a right prediction, the routing weight increases which increased the length of the output vector in the next iteration; thus, the existence probability of the entity represented by the vector. According to [20], the recommended number of routing iterations is three.

The length of the output vector is subsequently used to measure the probability that the entity exists by calculating the margin loss L_k. A separate margin loss is used for each class k as shown in (7).

$$L_k = T_k max(0, m^+ - ||v_k||)^2 + \lambda(1 - T_k) max(0, ||V_k|| - m^-)^2 \tag{7}$$

Taking into account that:

- $T_k = 1$ if the entity is present.
- m^+ and m^- are the hyperparameters which equal respectively 0.9 and 0.1.
- $\lambda = 0.5$.

B. Used datasets

1) Daimler dataset: We first train the network on Daimler Monocular Pedestrian detection images of 48x96px that we downsampled to 48x48 for computational resource purpose. 12000 samples are used for the training step and 1000 for testing. Daimler dataset doesn't provide ground truth information about the four pedestrian walking directions (front, back, left and right). To fit our purpose, we manually annotated the samples to the four pedestrian orientation classes described above. Despite the high image quality and the neat dataset that daimler provides, the samples are taken from well structured areas of a european city which represents the perfect case for a classification system. To evaluate the proposed architecture on a less structured area, we collected our own dataset gathered from Moroccan cities.

2) Moroccan collected dataset: The dataset contains images obtained from natural scenes of pedestrians walking and crossing the road, gathered from two Moroccan cities, Rabat and Kenitra. The acquisition was done in well structured avenues containing floor marking and pavements, in addition to poor structured ones where the place is crowded and pedestrians walk arbitrarily as seen in Fig. 2. To have a variant database the collection was during different lighting and weather conditions. In poor structured areas and due to the absence of road signs, pedestrians act randomly and cross between moving vehicles, specially in a crowded place. Therefore, in well structured areas known by a fast traffic flow, road signs are less respected by road users. This results in more serious accidents endangering pedestrians lifes.

Our collected data contain approximately 2580 cropped pedestrian images, that are extracted from a three hours video

Fig. 2: sample of acquisitions from Kenitra and Rabat captured using a camera on board a moving vehicle

Fig. 3: Samples of the four orientations (right, left, front and back) taken from the collected dataset

recording of one minute each, captured by a monochrome industrial camera with CMOS sensor and a resolution of 2.3 MP for a maximum of 60 fps. A picture of the camera on board the testing vehicle is shown in Fig. 2. To augment our dataset, images are mirrored using horizontal flip, which doubled the size of the dataset to 5160 pedestrian samples, that are resized afterwards to 48x48 pixels. The dataset contains four classes, each one represents one of the four orientations (front, back, left and right) of the pedestrian walking direction as represented in Fig. 3. For training, 4160 images among the 5160 images are used, where each class contains 1040 samples. While, the testing was done using 250 samples for each class, so a total of 1000 samples for test.

C. Capsnet Architecture for Pedestrian Walking Direction

We build a pedestrian orientation classification system based on capsule networks. The system contains the encoder and decoder part as illustrated in Fig. 4. While the encoder part classifies the input image into one of the four classes, the decoder part reconstructs the input image basing on the result of the classification.

The encoder part contains four layers:

- The two first ones are convolutional layers with 64x5x5 filters of stride 1 for the first one, it takes as an input a 48x48 grayscale image which gives an output of 44x44x64 tensor. While the second one contains 128x5x5 filters with stride of 1 and outputs a tensor of 40x40x128.
- The third layer represents the primary capsule layer, it contains 16 channels of 8 dimensions, thus each capsule receives as an input features extracted from the first layer, so a total input of 40x40x128x16.
- The final layer which we named PedCaps (inspired from capsules for pedestrians) consists of 4 capsules of 16 dimension, each capsule refers to one class among the four orientation classes. This layer aims to classify the input image and assign it to one of the classes aforementioned.

TABLE I
PEDESTRIAN ORIENTATION CLASSIFICATION ACCURACY BASING ON DIFFERENT CAPSNET ARCHITECTURES

Architecture	Nb of conv layers	Nb of filters	Nb of primary capsules	Loss	Accuracy
1	1	256	32	0.07	90.62%
2	1	128	16	0.016	96.87%
3	1	64	8	0.02	96.66%
4	2	conv1:256 and conv2:128	16	0.06	95.20%
5	2	conv1:64 and conv2:128	16	0.014	97.60%

In what concerns the decoder part, this latter contains 3 fully connected layers of 512, 1024 and 2304 filters respectively, it uses the true label from the PedCaps layer to reconstruct the input image.

IV. Results and Discussion

The proposed approach is tested using both Daimler and our collected dataset giving an accuracy of 97.60% and 73.64% respectively. The gap of the accuracy rate between the two datasets can be merely interpreted by the difference of the pedestrian crossing way for the both datasets. In well structured areas represented by Daimler dataset, pedestrians cross horizontally which makes the classification much easier than in our collected dataset where pedestrians cross obliquely.

We tested the network using different capsule architectures as represented in Table I. To train the network we used 50 epochs of 64 batch sizes. According to the results in Table I, reducing the number of feature maps from 256 to 128 and the number of capsule channels from 32 to 16 using 2 routing iterations, leads to a higher accuracy from 90.62% to 96.87% and a loss of 0.016. However, we choose the last architecture having the best accuracy of 97.60% for Daimler dataset by using two convolutional layers with 64 filters in the first layer and 128 in the second one with 16 primary capsules. Figure 5 illustrates the loss obtained while training the last architecture. The total loss of the algorithm is calculated using the margin loss and the reconstruction loss, with a regularization scale (8).

$$Total\,loss = Margin\,loss + \lambda(Reconstruction\,loss) \quad (8)$$

With $\lambda = 0.0005$ represent the regularization scale.

The reconstruction loss is calculated using mean squared error between the input and the reconstructed image. Samples of the reconstructed images are shown in Fig. 6.

We compare the proposed approach to some well known CNN architectures. Table II shows the test accuracy of Alexnet and Resnet on Daimler dataset that achieved an accuracy of 95.52% and 96.45% respectively, while the

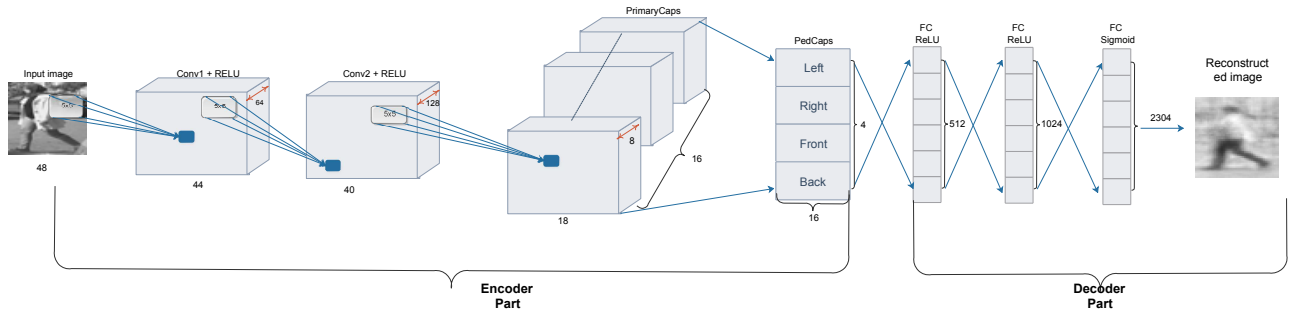
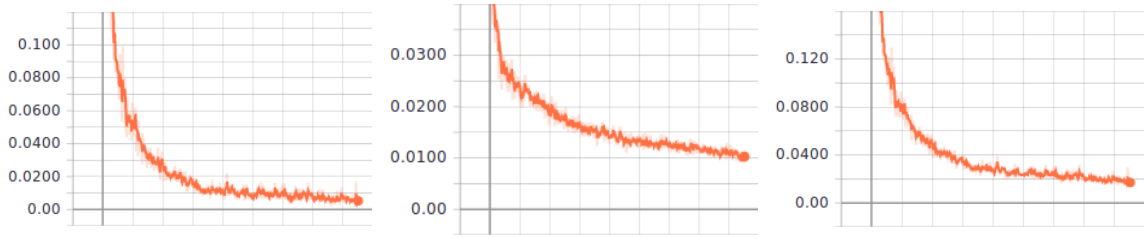


Fig. 4: Pedestrian Capsnet Architecture containing the encoder part that classify the input image into one of the four walking directions, and the decoder part that reconstructs the input image using the result of the encoder part



(a) Margin loss **(b)** Reconstruction loss **(c)** Total loss

Fig. 5: Representation of the losses of the second architecture where the abscissa axis represents the epoch number of training process and the ordinate one represents the rate of the loss: a) margin loss that equals to 0.004 and measured using the length of the output vector, b) reconstruction loss calculated using mean squared error between the input and the reconstructed image with a value of 0.01, finally the c) total loss which is the sum of the both losses and equal to 0.014.

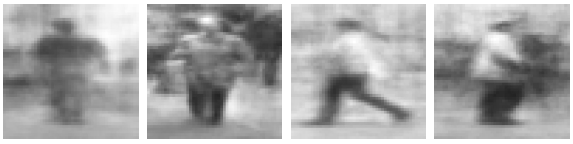


Fig. 6: Samples of reconstructed images

TABLE II
ALEXNET, RESNET AND CAPSNET ACCURACY COMPARAISON ON DAIMLER DATASET

Architecture	Accuracy
Alexnet	95.52%
Resnet	96.45%
Capsnet	97.60%

proposed Capsule Network architecture performed the best result of 97.60%.

Table III represents confusion matrices of the architectures presented above on Daimler dataset, we can observe that the best performer network is Capsule networks.

In Table IV, we present the confusion matrix of the capsnet architecture in Daimler dataset and we compared it with the one published in [18] (see Table V) using 3 orientation classes only. According to the results we can deduct that the use of 4 orientations gives more precision as well as a good classification accuracy comparing to 3 orientations.

Comparing between Daimler and our collected dataset using Capsule Network architecture, we can easily observe

TABLE III
CONFUSION MATRICES FOR DIFFERENT ARCHITECTURES: FROM TOP TO BOTTOM: CAPSNET, ALEXNET, RESNET

	Front	Back	Left	Right
Front	1	0	0	0
Back	0.004	0.98	0	0.008
Left	0.016	0.004	0.97	0
Right	0.008	0.04	0	0.93

	Front	Back	Left	Right
Front	0.99	0	0	0.008
Back	0.13	0.97	0.008	0.008
Left	0.016	0.012	0.95	0.02
Right	0.025	0.05	0.029	0.88

	Front	Back	Left	Right
Front	1	0	0	0
Back	0.008	0.983	0	0.008
Left	0.020	0.020	0.950	0.008
Right	0.016	0.045	0.012	0.925

TABLE IV
CONFUSION MATRICES OF OUR METHOD

	Front	Back	Left	Right
Front	1	0	0	0
Back	0.004	0.98	0	0.008
Left	0.016	0.004	0.97	0
Right	0.008	0.04	0	0.93

TABLE V
CONFUSION MATRICES IN [18]

	Front	Left	Right
Front	0.980	0.011	0.008
Left	0.058	0.841	0.100
Right	0.081	0.264	0.652

TABLE VI
CONFUSION MATRICES OF DATASETS USING CAPSNET

Daimler dataset

	Front	Back	Left	Right
Front	1	0	0	0
Back	0.004	0.98	0	0.008
Left	0.016	0.004	0.97	0
Right	0.008	0.04	0	0.93

Morrocan dataset

	Front	Back	Left	Right
Front	0.758	0.158	0.062	0.020
Back	0.095	0.779	0.058	0.066
Left	0.116	0.120	0.683	0.079
Right	0.070	0.125	0.083	0.720

Fig. 7: Samples of false predicted images

from Table VI that the collected dataset contains important misclassification rate of 26%, while false predictions on Daimler dataset are barely existant. The main four pedestrian directions used in this work are not well respected in poor structured areas where pedestrians cross in an oblique way, which makes the classification to the right orientation a rough task for the network, since false predictions are often noticed when the pedestrian's walking direction is between two orientations. Fig. 7 shown samples from those false predictions where we have a front direction classified as right, back as front, and left as front respectively.

As this approach is intended to be involved in a pedestrian collision prediction system, it has to be suitable for real life. Thereby, the use of more than 4 orientation bins is highly recommended, and will be the intent of future works.

V. CONCLUSIONS

This paper tackles one of the most crucial issues, which is pedestrians' walking directions, by presenting a novel approach based on Capsule Networks. This approach aims to classify pedestrians' orientations into one of the main four directions (front, back, left and right), the proposed CapsNet architecture performed better results for pedestrian orientation classification with an accuracy of 97.60% on Daimler dataset, compared to the tested CNN architectures. Additionally, we proposed in this paper a new dataset captured from Moroccan cities using a fixed camera on-board a moving vehicle. For future work, we aim to predict the pedestrian direction in video sequences to be applied in the Moroccan area. The integration of this method on PCAM systems is also planned to estimate collision probability and generate appropriate warnings to the Moroccan driver.

ACKNOWLEDGMENT

This research work is supported by METLE and CNRST under title: "SAFEROAD Meta-plateforme pour la Sécurité Routière (MSR)" project with contract No. 24/2017, and by LITIS laboratory in INSA Rouen.

REFERENCES

[1] WHO, *Global status report on road safety 2018*. World Health Organisation, 2018.
[2] WHO, *Road Traffic Injuries*. World Health Organisation, 2018.
[3] T. Gandhi and M. M. Trivedi, "Pedestrian collision avoidance systems: A survey of computer vision based recent studies," in *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pp. 976–981, IEEE, 2006.
[4] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *IEEE Transactions on intelligent Transportation systems*, vol. 8, no. 3, pp. 413–430, 2007.
[5] H. Hamdane, T. Serre, C. Masson, and R. Anderson, "Issues and challenges for pedestrian active safety systems based on real world accidents," *Accident Analysis & Prevention*, vol. 82, pp. 53–60, 2015.
[6] Z. Chen, C. Wu, N. Lyu, G. Liu, and Y. He, "Pedestrian-vehicular collision avoidance based on vision system," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pp. 11–15, IEEE, 2014.
[7] METLE, *Recueil des Statistiques des Accidents Corporels de la Circulation de l'année 2016*. Ministry of Environment, Transport, Logistics and Water, 2017.
[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.
[9] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," in *null*, p. 734, IEEE, 2003.
[10] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.
[11] R. Bunel, F. Davoine, and P. Xu, "Detection of pedestrians at far distance," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 2326–2331, IEEE, 2016.
[12] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster r-cnn doing well for pedestrian detection?," in *European Conference on Computer Vision*, pp. 443–457, Springer, 2016.
[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.
[14] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5079–5087, 2015.
[15] T. Gandhi and M. M. Trivedi, "Image based estimation of pedestrian orientation for improving path prediction," in *Intelligent Vehicles Symposium, 2008 IEEE*, pp. 506–511, IEEE, 2008.
[16] H. Shimizu and T. Poggio, "Direction estimation of pedestrian from multiple still images," in *Intelligent Vehicles Symposium, 2004 IEEE*, pp. 596–600, IEEE, 2004.
[17] K. Hara, R. Vemulapalli, and R. Chellappa, "Designing deep convolutional neural networks for continuous object orientation estimation," *arXiv preprint arXiv:1702.01499*, 2017.
[18] A. Dominguez-Sanchez, S. Orts-Escolano, and M. Cazorla, "Recognizing pedestrian direction using convolutional neural networks," in *International Work-Conference on Artificial Neural Networks*, pp. 235–245, Springer, 2017.
[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.
[20] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems*, pp. 3856–3866, 2017.
[21] A. D. Kumar, "Novel deep learning model for traffic sign detection using capsule networks," *arXiv preprint arXiv:1805.04424*, 2018.