Pedestrian detection from traffic scenes based on probabilistic models of the contour fragments

Florin Florian, Ion Giosan, Sergiu Nedevschi
Computer Science Department
Technical University of Cluj-Napoca, Romania
Florian.Florin@student.utcluj.ro; Ion.Giosan, Sergiu.Nedevschi@cs.utcluj.ro

***Abstract*—Driving assistance systems usually have a pedestrian detection module for alerting the driver in case of a dangerous situation. In this paper we describe such a module that is used for obstacles classification in pedestrians and non-pedestrians. The obstacles are defined by their region of interest (ROI) in the grayscale scene image. Random size and location of pedestrians' contour-edge fragments are extracted and filtered. They are used for building a very large codebook of pedestrians' contour fragments. A novel multi-level clustering is introduced in order to sequentially group these fragments first on location, then on size and finally on the content. A new method is proposed for computing a set of probabilistic contour fragments models inside each individual cluster. It is used for characterizing the entire codebook in just few models, one for each cluster. These models are used in a fast matching process against the obstacles ROIs that should be classified. A SVM classifier is trained on the matching scores vector and applied for detecting the pedestrians.**

Keywords—pedestrian detection; contour fragments; multi-level clustering; probabilistic models; SVM classifier

I. INTRODUCTION

The high rate of road accidents worldwide motivated the implementation of driving assistance systems that assist the driver in order to reduce the number of fatalities. To illustrate the magnitude of this problem, consider the number of road accidents in European Union countries. The statistics provided by the European Comission [1] show that in 2010 there were a total of approx. 1.1 million accidents resulting in 30,000 deaths and 1.5 million injuries. It's a downward trend over the last 10 years, aiming at continuous and rapid reduction of these undesirable events.

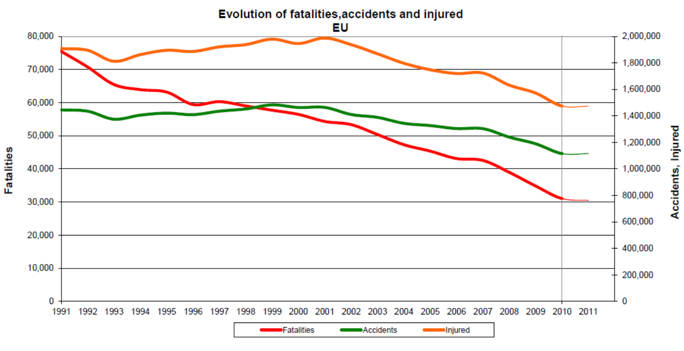


Figure 1. The number of accidents with deaths or injuries in the EU countries

This work was supported by the Romanian Ministry of National Education, UEFISCDI, project number PNII-PCCA 18/2012, SmartCoDrive

In urban traffic, pedestrians are the most important and vulnerable participants. In the last decade the problem of pedestrian detection has attracted high interest from national and international auto industry and the scientific community, all aimed at increasing their safety. Driving assistance systems that detect in advance potentially dangerous situations with pedestrians undoubtedly increase traffic safety for pedestrians. Their detection rate should be high enough to avoid both the false alarms sent to the driver and the dangerous situations with undetected pedestrians.

Although the pedestrian detection may be a simple problem for humans, it is a complex one for driving assistance systems that use artificial vision. This issue appears due to their large variations of body poses, different clothing and accessories they are carrying. Other variables that contribute as well are the variations of background scene, the environment conditions and the distance from the image acquisition cameras and their resolution, the vehicle vibrations and scene cluttering. For a driving assistance system, the most important aspect is that the pedestrian detection should achieve real time execution with high accuracy results. These aspects make the detection a very complex process which must involve multiple fast and efficient algorithms.

A lot of different technologies such as ultrasound sensors, piezo-electrical sensors, laser scanners [2], microwave radars and video cameras [3] are frequently used for pedestrian detection [4]. We use video cameras due to the fact that the acquired images contain a lot of information and it is a clean and passive acquisition way because it doesn't imply any type of pollution either to environment or people.

Our main contributions consist in developing an entire pedestrian detection system considering grayscale ROI images as input, based on novel contour-fragments probabilistic models formed within clusters, obtained from a new proposed schema for multi-level sequentially clustering of fragments on location, size and content.

In chapter II we briefly present the literature review regarding pedestrian detection methods. In chapter III we describe the entire architecture of the pedestrian detection module together with detailed description of its component parts. In chapter IV the experimental results are presented. Finally in chapter V the conclusions are drawn and possible future improvements are proposed.

II. RELATED WORK

In scientific literature, many pedestrian detection modules are developed by using different algorithms and technologies. They are integrated on driving assistance systems, having the goal of achieving a very robust detection even in very difficult traffic scenarios.

A pedestrian detection system based on local multi-scale oriented intensity differences, that uses the Haar wavelet transform and SVM for classification is presented in [5]. In [3] a complete and robust system that performs a stereo-based depth segmentation, a chamfer matching for shape, texture classification for verification using neural network, stereo-based verification, and tracking is described.

Feature extraction and tracking based on structural changes of shape and symmetry detection method using morphological operators are used for pedestrian detection and presented in [6]. Active contour models can be used for pedestrian intensity image segmentation and stereo-vision information for guiding the active contour location since they are very sensitive to initial position [7].

Scene objects motion computation is also very important [8] because it could separate the pedestrians from other moving objects. In case of pedestrians hypotheses a set of appearance models representing body-silhouettes can be used for their tracking [9]. A set of probabilistic appearance models could be used as well [10] for pedestrian detection and tracking. The probabilistic tracking methods [11] decompose the pedestrians' motion from video sequences in order recognize their attitudes. In [12], a tracking procedure based on temporal derivative and image template matching is used. It achieves high tracking performance and high classification results in case of objects that are partially occluded.

Although there are many features used for pedestrian detection, the shape contour is a robust feature that is widely used [7]. It eliminates the most of the issues specified in the introduction which could cause weak detection (both false positives and false negatives). The contour describes the shape of the pedestrian, being invariant to pedestrian clothing and environment illumination. Pedestrian hypotheses are usually matched against a set of pedestrian contour templates in order to determine if they are pedestrians or non-pedestrians. In [13], an approach for pedestrian detection using a hierarchy of contour templates and pattern matching using distance transform on edges is presented. A complete pedestrian detection system, using monocular vision, based on edges and shapes is described in [14].

Usually the acquired scene image is too crowded or there are obstacles parts that are partially occluded by other obstacles. The human detection system has a very high rate of identifying the obstacles even if they have a lot of occluded parts. In [15], an image annotation system extracts obstacles edges and keep just those belonging to the exterior contour. Like the human detection system, it achieves a good obstacle annotation just using the contour (edge) fragments (see Figure 2).

In [16],[17] and [18] the obstacles' contour is matched against an image edge map, but the important issues that affect the result are referring to the obstacle segmentation and edge detection. A good obstacle segmentation gives accurate very useful for further recognition.

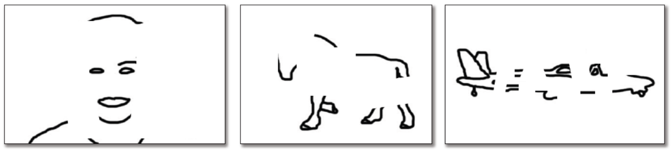


Figure 2. Examples of different objects' contour fragments [15]

Similar approaches, but using versions of improved models, where obstacle templates are divided into parts are widely used in vision applications [19], [20], [21]. Parts are individually likely to match a set of background artifacts, but it is proven that they are robust and able to describe both articulated and rigid obstacles.

A real time pedestrian detection system uses infrared images and probabilistic templates to capture the variations in pedestrians shape in the case where there is contrast is low and some body parts could be missing is implemented in [22]. The experiments are taken on infrared videos acquired from a moving vehicle in urban traffic scenarios.

In [23] the authors present a pedestrian detection system that uses probabilistic models in the infrared domain. Four different models are proposed in order to accurate determine the pose of the pedestrians' legs: open, almost open, almost closed and fully closed legs. It introduces two new approaches that overcome the drawbacks of template matching procedure in far infrared images.

A mixture of two probabilistic template based classifiers for pedestrian detection is described in [24]. The first is a binary probabilistic template based classifier (BPTC) used to reject the most of non-pedestrians by the features of binary image. The second is the gray probabilistic template based classifier (GPTC) used for achieving the final classification result by the gray probability.

Many obstacle classification algorithms including pedestrian/non-pedestrian classes are proposed in literature, but the issue of achieving high classification results for driving assistance systems in complex traffic scenarios is still far from being solved. A feed-forward neural network [25] is used for distinguishing between vehicles, pedestrians, and other background obstacles. Support Vector Machines [26] may also be used for pedestrians, vehicles and animals classification. Other classifier that use error correction output is described in [27] and used for obstacles classification of cars, trucks, bikes, pedestrians. In [8], a classifier that doesn't need to be trained with obstacles instances is used for classification.

In this paper, we describe the development of a pedestrian detection system. We propose a new method for computing a set of pedestrian probabilistic contour fragments models. A novel multi-level clustering is proposed in order to sequentially group the fragments. The models are used in a fast matching process in order to classify a ROI image of each pedestrian hypothesis. A SVM classifier is trained on the matching scores vector and then applied for real-time pedestrian detection.

III. PEDESTRIAN DETECTION BASED ON PROBABILISTIC MODELS OF THE CONTOUR FRAGMENTS

In this chapter we describe the system architecture with its component modules together with all the algorithms used for pedestrian detection. We emphasize the contour fragments extraction, the novel multi-level clustering and the new method for building the probabilistic models, and finally, the classification process used for pedestrian detection.

A. *System Architecture*

The modules and the data flow inside the pedestrian detection system architecture are depicted in Figure 3.

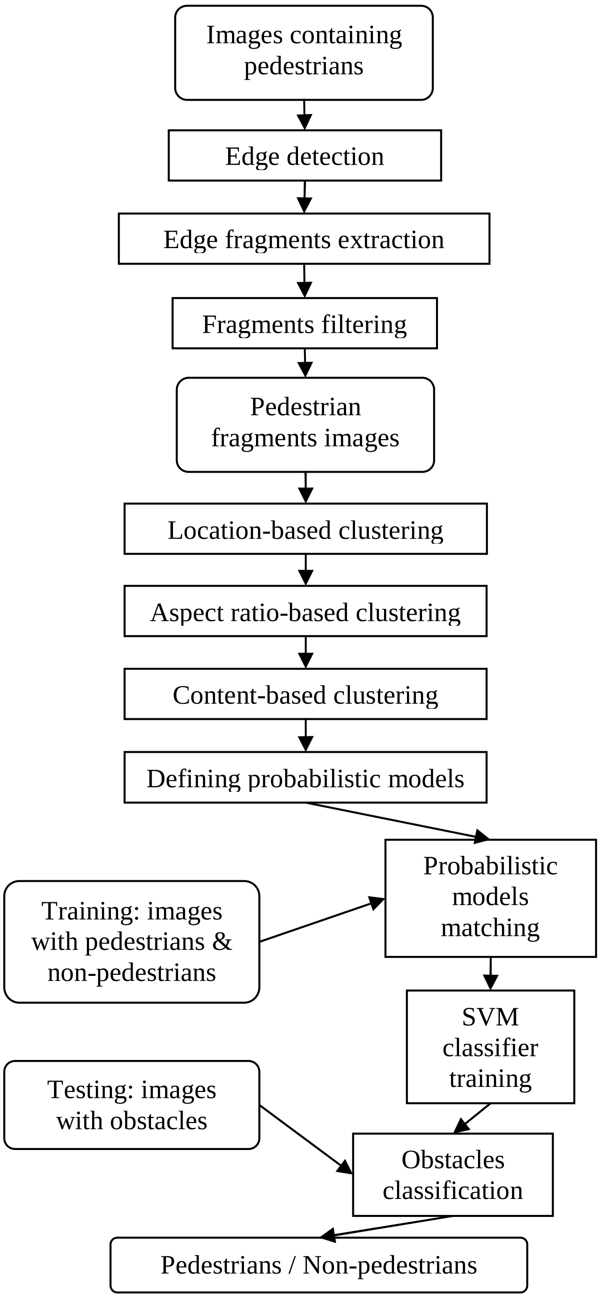


Figure 3. The pedestrian detection system architecture

B. *Input Images with Random Fragments Splitting*

The first step is to split the grayscale ROI input images (see Figure 4) containing pedestrians into small fragments. The location and the size of those fragments are randomly chosen. For generating those random values, we consider the same pseudo-random numbers generator.

Figure 4. Samples of a pedestrian grayscale input images (ROIs)

The image fragments (see Figure 5) are having a width and a height that is variable between 20-40% from the entire ROI image original size to prevent having too big or too small fragments. The top-left corner coordinates (x, y) of each fragment are having a value between 0-80% from the width and height of the input ROI image.

The number of image fragments (see Figure 6) taken from an image is set to 120 fragments to ensure that the input image has maximum coverage without having unnecessary or redundant data (like fragments that overlap almost perfectly).

Figure 5. Extracting image fragments from random positions with random sizes (width and height)

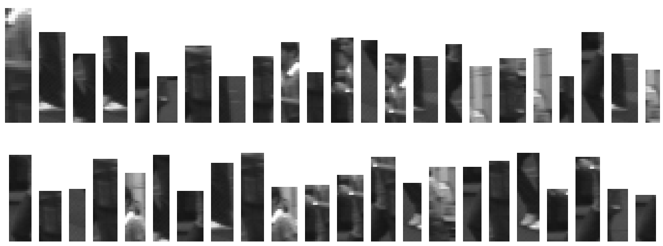


Figure 6. Samples of pedestrians image fragments

C. Fragments Edge Detection and Filtering

Each of the extracted fragments will proceed to a series of processes, with the final goal of probabilistic determining if the extracted fragment contains or not a part of a pedestrian. If the fragment doesn't contain a fragment of the pedestrian, the fragment is useless and it is eliminated.

In order to find if the image fragment contains a part of a pedestrian, the edges are firstly extracted from the fragment and then they are being analyzed.

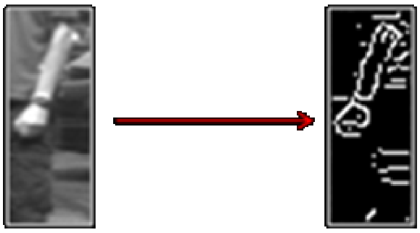


Figure 7. Edge detection of a fragment containing a pedestrian hand

A Sobel filter is used for gradient computation. It is a simple approach which offers the possibility of fast computing the gradient magnitude in each position due to its small convolution kernel size. A LoG approach with large σ would find better places of edges but it considers a wider area around the pixel leading to a higher processing time and malfunctioning at the curves and corners. The edges are obtained directly by thresholding the Sobel gradient values (see Figure 7).

This step just prepares the data for the next step which consists in the sorting of the fragments based on edge density.

The method of extracting random fragments from an image containing a pedestrian can provide fragments that contain background or fragments that contain too many details. That kind of fragments needs to be eliminated.

An edge density based filter is used for fragment filtering, keeping just those images that have a high probability to contain a pedestrian contour (see Figure 8). The inequality is defined in equation (1).

$$\begin{gathered} N_1 < EBD < N_2 \\ N_1 = 10, N_2 = 35 \end{gathered} \tag{1}$$

where:

N_1 is the minimum threshold value
N_2 is the maximum threshold value
EBD is edge density based value of the fragment

Figure 8. Sample of edge fragments that have been eliminated

D. Multi-level Clustering

The set of fragments obtained from the previous step still contain fragments that need to be eliminated because they don't contain valid data (do not contain a part of a pedestrian contour).

A clustering procedure is used in order to group the fragments. The clustering result contains some groups that have a low number of fragments. These groups are assumed that are formed by fragments that don't contain a pedestrian contour, so they need to be eliminated.

A multi-level clustering of the extracted fragments is proposed for increasing the chance to keep good fragments (pedestrian contour fragments) in our collection. A good quality set of fragments is essential because they have a major role in further matching and training of a classifier that is used for pedestrians' detection.

The multi-level fragment clustering has three sequential levels:

- fragment image location clustering
- fragment image aspect ratio clustering
- fragment image content clustering

Fragment Image Location Clustering

This kind of clustering is trying to create groups based on the relative position of the fragment in its original image, taking into account the coordinates of the top-left corner of the fragment (see Figure 9).

The relative position of its top-left corner and its relative width and height are stored for extracted fragment. These values are combined using the Euclidian metric to define the distance function between fragments. The clustering is made using the BSAS algorithm for creating groups of fragments.

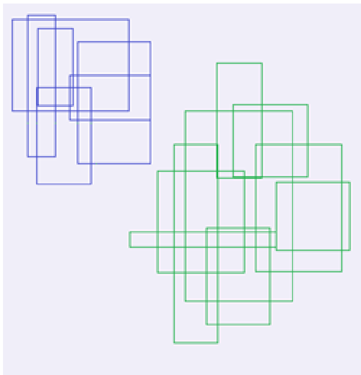


Figure 9. Sample of two groups of fragments obtained from the location based clustering together with their position in the image

The variables from BSAS algorithm have the values:

O = 0.2 - the threshold of dissimilarity
Q = 10 - the maximum number of clusters
M = 5 - the value of the matching coefficient

A relative location is calculated for each cluster, representing the average value of the contained fragments location, value that is needed to generate the test benchmark for the SVM classifier used for the pedestrian detection.

Fragment Image Aspect Ratio Clustering

The extracted fragments may have different aspect ratios (see Figure 10), so a clustering is needed to group the fragments based on aspect ratio similarity in order to obtain more precise clusters (see Figure 11). The input data is the clusters from the previous step, which will be split into subgroups that contain fragments with similar aspect ratio. The distance function is computed using the Euclidian distance.

The variables from BSAS algorithm have the values:

SC = 5 - the maximum sub clusters created for each cluster from input data
O = 2 - the threshold of dissimilarity
M = 0.07 - the value of the matching coefficient

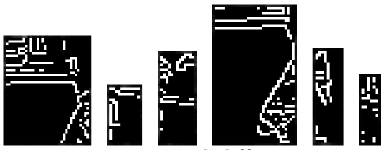


Figure 10. Fragments with different aspect ratio

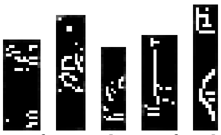


Figure 11. Fragments from a cluster after the fragment image aspect ratio clustering

The clusters that have few fragments are eliminated after completing the aspect ratio clustering, because they have a very low probability of containing a pedestrian contour.

Fragment Image Content Clustering

The final step is to create groups of fragments that contain a pedestrian contour considering their content similarity. Every cluster obtained from the previous step is separately analyzed and divided in subgroups that contain fragments that are similar. Two fragments are similar if they contain similar contour shapes. The fragments are analyzed by overlapping the fragments two by two and calculating their similarity value.

After completing this step and eliminating the clusters with few fragments we obtain groups of fragments for key zones (representative zones) of a pedestrian. The function used to compare two fragments uses the distance transform (*DT*) algorithm to generate the matching value used by the BSAS algorithm. The distance between two fragments is computed using the equation (2).

$$D(F_1,F_2) = Sum\left\{ \begin{array}{l} Max\left\{ \begin{array}{l} RF_1\ over\ DT(F_2), \\ RF_2\ over\ DT(F_1) \end{array} \right\}, \\ Max\left\{ \begin{array}{l} F_1\ over\ DT(RF_2), \\ F_2\ over\ DT(RF_1) \end{array} \right\} \end{array} \right\} \quad (2)$$

where:

$D(F_1,F_2)$ = the distance between fragments F_1 and F_2
RF_k = the fragment k resized to the other fragment that is compared to
$DT(X)$ = the distance transform image of fragment X

The variables from BSAS algorithm have the values:

SC = 5 - the maximum sub clusters created for each cluster from input
O = 2 - the threshold of dissimilarity
M = 5 - the value of the matching coefficient

Similar to the previous step, the clusters that have few fragments are eliminated after completing the clustering operation. For each resulted cluster, we compute its relative location and relative size. The relative size is obtained from the average of the entire cluster's fragments size.

E. *Building the Probabilistic Models*

The probabilistic models are created by using the resulted clusters. A single model is computed for each cluster. The probabilistic model of a cluster is obtained by overlapping and averaging all the contour fragments from that cluster over a white image with 50x100 pixels (see Figure 12). In the probabilistic model map, a darker intensity pixel represents a point where many contour fragments overlapped on that position (they have a edge-contour pixel in that location).

Figure 12. Building a probabilistic model for a pedestrian hand

The probabilistic model has different content for different parts of a pedestrian, and contains information about the relative location and the relative size of the cluster that it represents (see Figure 13).

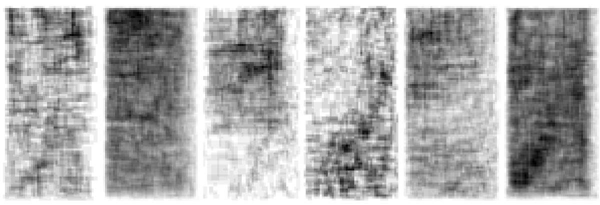


Figure 13. Samples of different probabilistic models

F. *Training the SVM Classifier*

All the computed probabilistic models will be used to generate the training data that will be loaded into the SVM classifier. The input data used by the SVM classifier contains the class of the analyzed object (pedestrian or non-pedestrian) together with the matching scores obtained after overlapping all the probabilistic models over the image that is analyzed. In order to overlap the probabilistic models, firstly it is necessary to detect the edges from the input image. The training set contains images with pedestrians and non-pedestrians. In case of pedestrians, the *class* identifier is 1 and in case of non-pedestrians its value is 2.

An array of matching scores will be generated for each set of images (3).

$$Arr[i] = \{class, v_1, v_2, ..., v_n\} \quad (3)$$

where v_k is the matching score resulted from overlapping the probabilistic model number k over the input image i.

The probabilistic models are overlapped over the test images (see Figure 14) after completing two preliminary steps: calculating the top-left corner location over the image and resizing the probabilistic model according the image size (for every model, its relative size is known).

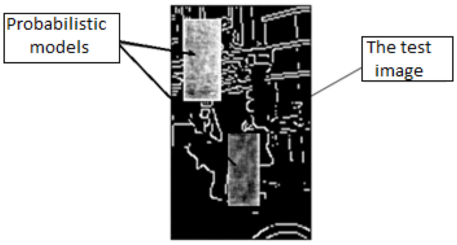


Figure 14. Overlapping the probabilistic models over a test image

The SVM classifier will store the generated support vectors in an XML file that will be used for further classifications. The training data used by the classifier needs to be computed just once.

IV. Experimental Results

In this chapter we present our pedestrian detection results obtained using the probabilistic models in order to distinguish pedestrians from a set of obstacles. The proposed methods and algorithms were tested on a large set of obstacles images acquired from different traffic scenarios.

In the training phase we use about 1000 grayscale images with pedestrians and non-pedestrians (other obstacles).

The test data set and the detection results with the performance metrics can be summarized as:

- Number of test images : 1100
- Correct detection: 825
- Wrong detection: 275
- Accuracy: 75%
- Error rate: 25%

TABLE I. The confusion matrix for 2 sets of images, each with 550 images containing pedestrians and nonpedestrians

		Predicted class	
		Pedestrian	**Non-pedestrian**
Real class	**Pedestrian**	440	110
	Non-pedestrian	165	385

An optimization has been applied on the way the probabilistic model is overlapped over the image, in order to obtain the matching score. The model is slid on each horizontal and vertical direction with an offset (number of pixels) that varies between 1-5 pixels. In each location, a matching score is computed and it is passed to the SVM classifier. This step is introducing a lot of computations and is slowing down the detection, but the system still performs in real time.

The test data set together with the optimized detection results and the performance evaluation are summarized as:

- Number of test images : 1100
- Correct detection: 890
- Wrong detection: 210
- Accuracy: 81%
- Error rate: 19%

TABLE II. The confusion matrix for 2 sets of images, each with 550 images containing pedestrians and nonpedestrians using the probabilistic model translation approach

		Predicted class	
		Pedestrian	**Non-pedestrian**
Real class	**Pedestrian**	470	80
	Non-pedestrian	130	420

The ROC curve of the pedestrian detector is depicted in Figure 15. We achieved the best results when we split the input images in a number of N=120 fragments (a TP rate of 81% and a FP rate of 20%).

In [28] some pedestrian detection systems from monocular images were tested on different datasets. The evaluation is done by plotting the miss rate against the false positives per image. Our system takes as input just the grayscale ROI of the traffic scene obstacles (windows) that need to be classified in pedestrians and non-pedestrians. Unfortunately we cannot make a perfect comparison between our system and other similar systems (that considers the whole scene image as input). However our results are comparable with other pedestrian detection systems results from literature.

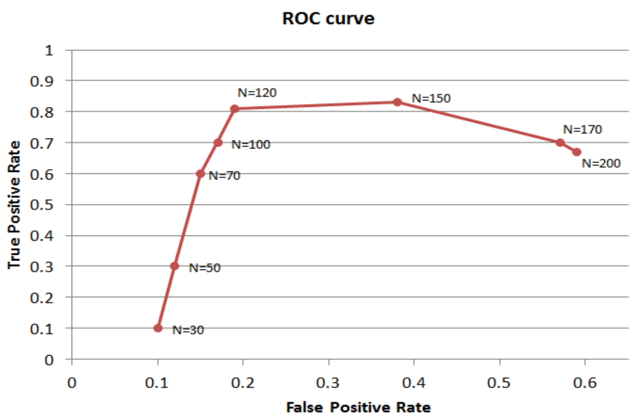


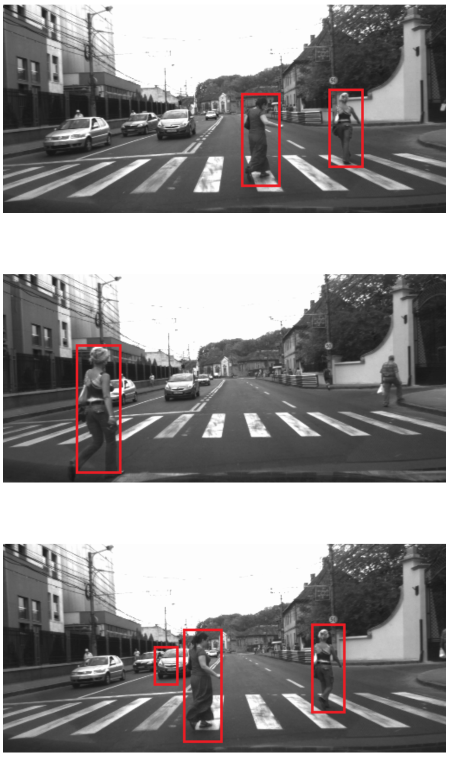
Figure 15. Pedestrian detector ROC curve (performance measurements with different number of fragments N)

V. CONCLUSIONS

We have built a pedestrian detection system from monocular grayscale images. The obstacles ROIs from the traffic scene images were supposed to be known (they were considered as being the output from an obstacle detection module). Our approach consists in extracting random size and location samples of pedestrians' contour-edge fragments. A codebook is built upon these pedestrians' contour fragments.

We proposed a novel multi-level clustering in order to sequentially group the fragments on location, size and content. A new method is described for computing a set of probabilistic contour fragments models inside each individual cluster. A probabilistic model characterizes the codebook within one cluster. We define one probabilistic model for each cluster. A fast matching procedure with probabilistic models is applied in order to find if an obstacle ROI is similar to a pedestrian. A SVM classifier is finally trained (positive and negative samples) on the matching vector scores. The classifier is applied in each sequence frame for classifying the unknown obstacles ROIs in pedestrians and non-pedestrians. In Figure 16 we present some detection results. The pedestrian detection is running in real-time mode, achieving above 30 fps when running the optimized version on a computer with an Intel Core i3 processor at 2.52 GHz.

Future work will involve the increase of the number of training images, combining our probabilistic method with other complex pedestrian detection approaches in order to increase the accuracy and parallelizing the implementation to maintain the real-time execution.



a) correct detection of both pedestrians

b) an example of a false-negative: the pedestrian in the right side of the image is miss-classified

c) an example of a false-positive: the car from the center-left of the image

Figure 16. Pedestrian detection results (with red bounding boxes)

REFERENCES

[1] European Comission, "Road Safety evolution in the EU," in *http://ec.europa.eu/transport/road_safety/specialist/statistics/*, ed, 2012.

[2] B. Fardi, U. Schuenert, and G. Wanielik, "Shape and motion-based pedestrian detection in infrared images: a multi sensor approach," in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, 2005, pp. 18-23.

[3] D. M. Gavrila, J. Giebel, and S. Munder, "Vision-based pedestrian detection: the PROTECTOR system," in *Intelligent Vehicles Symposium, 2004 IEEE*, 2004, pp. 13-18.

[4] 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*, 2006, pp. 976-981.

[5] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *International Journal of Computer Vision*, vol. 38, pp. 15-33, 2000.

[6] L. H. Z. S. T. Sziranyi, "Pedestrian Detection Using Derived Third-Order Symmetry of Legs," *Proceedings of the IEEE International Conference on Computer Vision and Graphics*, pp. 11-17, 2004.

[7] C. Hilario, J. M. Collado, J. M. Armingol, and A. d. l. Escalera, "Pedestrian detection for intelligent vehicles based on active contour models and stereo vision," *Proceedings of the 10th international conference on Computer Aided Systems Theory*, pp. 537-542, 2005.

[8] O. Javed and M. Shah, "Tracking and Object Classification for Automated Surveillance," *Proceedings of the 7th European Conference on Computer Vision-Part IV*, pp. 343-357, 2002.

[9] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: real-time surveillance of people and their activities," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, pp. 809-830, 2000.

[10] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proceedings of the IEEE*, vol. 90, pp. 1151-1163, 2002.

[11] C. Bregler, "Learning and recognizing human dynamics in video sequences," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, 1997, pp. 568-574.

[12] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target classification and tracking from real-time video," in *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on*, 1998, pp. 8-14.

[13] D. M. Gavrila and V. Philomin, "Real-time object detection for “smart” vehicles," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, 1999, pp. 87-93 vol.1.

[14] A. Broggi, M. Bertozzi, A. Fascioli, and M. Sechi, "Shape-based pedestrian detection," in *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, 2000, pp. 215-220.

[15] J. Shotton, *Contour and Texture for Visual Recognition of Object Categories: Automatic Object Recognition Using Learned Patterns of Contour and Texture*: Lightning Source Incorporated, 2008.

[16] Y. Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2001, pp. 105-112 vol.1.

[17] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, pp. 309-314, 2004.

[18] P. Dollar, T. Zhuowen, and S. Belongie, "Supervised Learning of Edges and Object Boundaries," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, 2006, pp. 1964-1971.

[19] M. Weber, M. Welling, and P. Perona, "Unsupervised Learning of Models for Recognition," presented at the Proceedings of the 6th European Conference on Computer Vision-Part I, 2000.

[20] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2003, pp. II-264-II-271 vol.2.

[21] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial Structures for Object Recognition," *Int. J. Comput. Vision*, vol. 61, pp. 55-79, 2005.

[22] H. Nanda and L. Davis, "Probabilistic template based pedestrian detection in infrared videos," in *Intelligent Vehicle Symposium, 2002. IEEE*, 2002, pp. 15-20 vol.1.

[23] M. Bertozzi, A. Broggi, C. H. Gomez, R. I. Fedriga, G. Vezzoni, and M. Del Rose, "Pedestrian Detection in Far Infrared Images based on the use of Probabilistic Templates," in *Intelligent Vehicles Symposium, 2007 IEEE*, 2007, pp. 327-332.

[24] H. Mei, "Multiple Probabilistic Templates Based Pedestrian Detection in Night Driving with a Normal Camera," in *Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on*, 2006, pp. 574-577.

[25] D. Toth and T. Aach, "Detection and recognition of moving objects using statistical motion detection and Fourier descriptors," in *Image Analysis and Processing, 2003.Proceedings. 12th International Conference on*, 2003, pp. 430-435.

[26] E. Rivlin, M. Rudzsky, R. Goldenberg, U. Bogomolov, and S. Lepchev, "A real-time system for classification of moving objects," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, 2002, pp. 688-691 vol.3.

[27] Z. Lun, S. Z. Li, Y. Xiaotong, and X. Shiming, "Real-time Object Classification in Video Surveillance Based on Appearance Learning," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 2007, pp. 1-8.

[28] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, pp. 743-761, 2012.