Building Pedestrian Contour Hierarchies for Improving Detection in Traffic Scenes

Ion Giosan, Sergiu Nedevschi

Tehnical University of Cluj-Napoca, Computer Science Department,
G. Baritiu 26-28, 400027 Cluj-Napoca, Romania
{Ion.Giosan,Sergiu.Nedevschi}@cs.utcluj.ro

Abstract. This paper presents a new method for extracting pedestrian contours from images using 2D and 3D information obtained from a stereo-vision acquisition system. Two pedestrian contour types are extracted. First is obtained from static pedestrian confidence images using fixed background scenes and second from general traffic scenes having variable background. A robust approach for building contour hierarchies of these contours is then presented. First hierarchy is built of "perfect" contours extracted from fixed background scenes and the second one is built of "imperfect" contours extracted from images with variable background. The objective is to evaluate the two hierarchies in order to identify the best one for real time pedestrian detection.

Key words: Pedestrian contours, Stereo-vision, Fixed background, Dynamic background, Contour hierarchies

1 Introduction

In the last 20 years, the high rate of traffic accidents from all world have determined the building of intelligent vehicles in a seriously way. Let's consider the number of traffic accidents only in Europe for illustrating the importance of this problem. In these traffic accidents a number of about 150.000 pedestrians are seriously injured and about 6.000 pedestrians are losing their lives [1]. A lot of technologies such as ultrasound sensors, piezo-electrical sensors, laser scanners [2], microwave radars and video cameras [3] are frequently used for pedestrian detection [4].

Although the pedestrian detection is a simple problem for humans, it is a complex one for artificial vision systems due to the large variations of human body poses, the clothes, the accessories which they are carrying and of course due to the variations of background scenes and the environment conditions. The most important thing is that the pedestrian detection should be done in real time and this converges to consider that the detection is a complex process which involves multiple efficient detection methods.

The usage of contour as a feature for pedestrian detection [5] is motivated because it eliminates most of the problems presented previously which could cause weak detection. The contour describes the shape of the pedestrians and it is invariant to the pedestrians' clothes which determine different texture aspects on their image. A classical approach for object detection using a hierarchy and pattern matching with distance transform is described in [6]. In [7] is presented a system of pedestrian detection based on edges and shapes but using only monocular vision. The main idea in the pattern matching is to have a collection with a lot of pedestrian contours with different attitudes and poses. The collection contains a set of perfect pedestrian contours obtained from ideal pedestrian images and another set of contours extracted from traffic stereo-reconstructed images which usually present imperfections.

The hierarchies of pedestrian contours will decrease the matching time between the ROI and the collection because of reducing the number of matches and offering the performance to the system of performing in real-time mode. The hierarchies of pedestrian contours consist in building a contour-templates tree. Similar systems of pedestrian detection using template trees are described in [8], [9] and [10].

2 Pedestrian Contour Extraction

In this section it is presented the main idea of the pedestrian contour extraction, the contour extraction algorithm with its input and output and the types of contour extraction obtained by using different types of input images.

A solution for real-time pixel matching is to consider a smaller number of pixels involved in the process [6]. Here it is the idea of using a pedestrian contour in pattern matching process rather than the entire intensity pedestrian image. Contour extraction is useful in building a pedestrian contours collection which contains entire body contours of pedestrians.

2.1 Types of contours

The collection contains a set of perfect pedestrian contours obtained from ideal pedestrian images and another set of contours extracted from traffic stereo-reconstructed images which usually present imperfections. This is useful for having a large contour collection to get a realistic valid score when matching the ROI contour in scene with the models from the collection.

In first case sequences of images having fixed background are used to obtain ideal confidence images. Then is applied the contour extraction and stored the obtained contour in the collection. In second case sequences of motion images when background is changing from one frame to another are used to obtain the approximated zones of the pedestrian from their intensity and range image. Then is applied the contour extraction on these zones, combined and formed a single contour and stored the contour in the collection.

2.2 Contour Extraction Algorithm

An object image which can contain several parts (connected components) represents the input of the algorithm. A continuous contour formed of a sequence of contour pixels which approximates the object is defining the output of the algorithm.

First step is to determine all the contours of the connected components. This image is a binary one containing white pixels for pedestrian and black pixels for the background. Objective is to draw only the contours of these pedestrian connected components. A border tracing is used to determine the contour of each connected component.

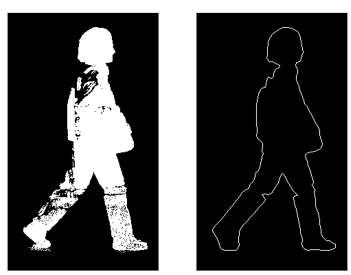
Next step is to calculate the minimum distance between all valid contours obtained from previous steps. This distance is calculated as a Euclidian distance between all pairs of contours' points. The minimum distance between two contours is set as being the smallest value of the distances between their contour points.

A merging process between all connected components is then applied. The order of merging components is given by the ascending order of distance values between components. Merging is done between contour points of all pairs of different connected components that are satisfying a maximum distance constraint. The result of this step is one connected component which approximates the image of the input object.

Final step is to trace the contour of this connected component which will represent the output of the algorithm. The contour trace method and the considered contour points which approximate the object are the same as we previously described.

2.3 Contour Extraction from Sequences of Fixed Background Images

In this case contour is extracted from confidence images (Fig. 1). Confidence images are formed of connected components of a pedestrian image. After confidence image is obtained then the previously described algorithm is applied to obtain the pedestrian's contour.



a) b)

Fig. 1. Pedestrian contour extracted from images with fix background: a) Confidence image. b) The extracted contour

2.4 Contour Extraction from Sequences of Dynamic Background Images

First step is to mark in the intensity image the region that is representing a pedestrian. This is done with mouse dragging and the selection rectangle should be as narrow as possible to select a small amount of background in comparison with the amount of the pedestrian pixels (Fig. 2).

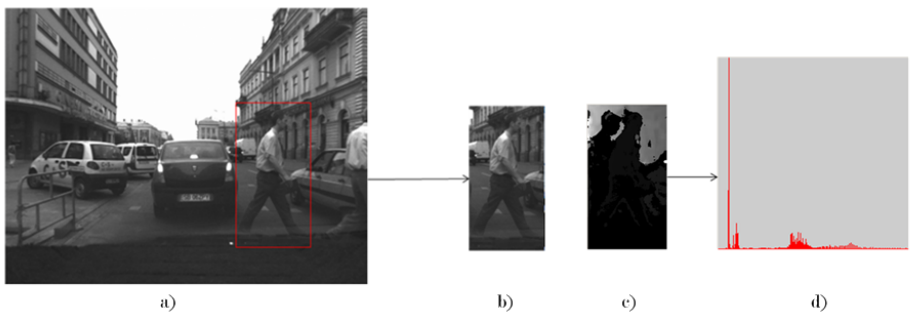


Fig. 2. Selection of pedestrian area and depth histogram computation: a) Scene image from a sequence with pedestrian selected area (red). b) Pedestrian selected area. c) Depth image of the pedestrian selected area (red). d) The computed depth histogram.

Next step is to compute a depth histogram in range image only in the selected pedestrian area. This is used to separate the pedestrian from the background in the selected area. In Fig. 2 is presented a depth histogram of the pedestrian selected area from the scene image.

Then in the previously calculated histogram is searching for the maximum value. A binary image containing white pixels for pedestrian and black pixels for background is then constructed considering the previous peak. Next step is to extract the contour (Fig. 3a)) using the algorithm described in previous sections.

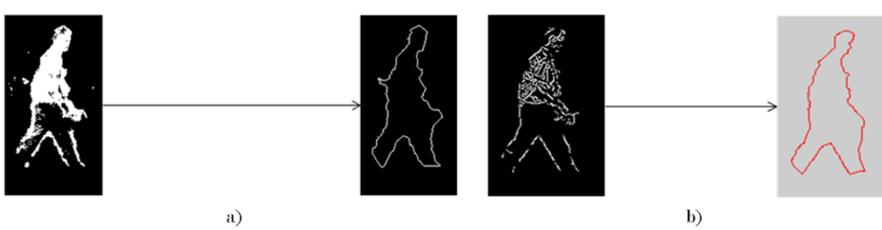


Fig. 3. a) Contour extraction (right) on binary image of pedestrian connected components (left). b) Edge image of pedestrian (left) and smoothed contour on edge image using snake (right)

Final step in contour extraction from sequences of changing background images is to refine previously extracted contour. In smoothing process of this contour is used an active contour (snake). Active contour models [11] may be used in image segmentation and understanding and are also suitable for analysis of dynamic image data or 3D image data. The snake is used to best fit the pedestrian contour as near as possible to the real edges.

Before starting the snake it is calculated an edge image of the pedestrian selected area on intensity image using Canny edge detector and then the result is masked by depth range (Fig. 3b)). In the edge image are also removed the small edges as being considered noise. Snake is starting from contour points previously calculated on the edge image and tries to fit better on the pedestrian shape.

Smoothing with active contour removes small deviations introduced by depth masking (Fig. 4).

Fig. 4. Final pedestrian contour viewed over the intensity image

3 Building the Contour Templates Hierarchies

The contour templates hierarchy is very important in the process of pedestrian detection based on shape matching because it reduces the number of matches between the region of interest (ROI) from the scene image which represents an unknown object and the entire collection of pedestrian contour templates.

Contour templates hierarchy is built as a multipath tree structure, each node representing a pedestrian contour template obtained using the methods already presented in previous chapter.

An unsupervised clustering algorithm based on a modified version of classical "K-means" algorithm is used in building of the multipath tree. This algorithm is applied to build each level in the tree hierarchy starting from leafs level to root level.

At each level the input of the algorithm is a set of pedestrian contour templates, leafs level containing the entire set of templates. The result of applying the algorithm at every level is a number of template groups, each group containing similar templates and a prototype template for each group.

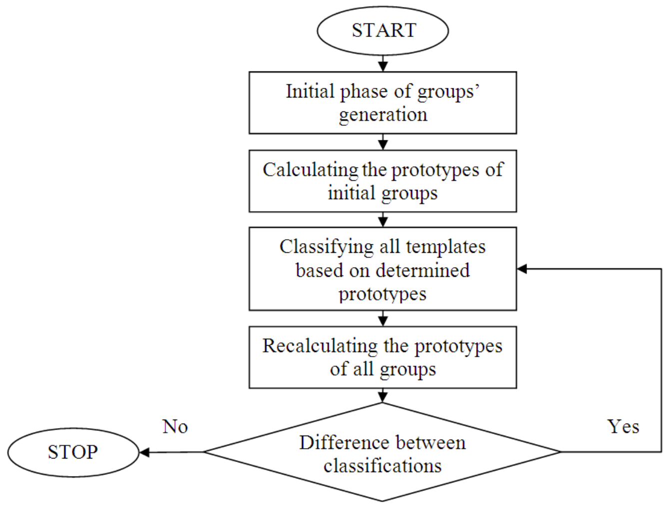


Fig. 5. Template tree hierarchy building algorithm at each level of the tree

The algorithm is following a number of well defined steps presented in next paragraphs (Fig. 5).

Before applying the clustering algorithm it is necessary to calculate the dissimilarity matrix between all templates. Considering N contour templates of pedestrians, the elements of the matrix are calculated as evaluating the minimum Euclidian distance between that pair of templates.

First step of the algorithm is the initial phase of groups' generation. In this step are generated first groups of templates starting from a heuristic. First of all it is considered that neither of all templates belongs to a group. Then all distances from distance matrix are ordered ascending in an array. Next procedure is an iterative one: starting from the first distance to the last one, in the ordered array, are connected as belonging to the same group or to a new group (if neither of them belongs to a group) both templates which form that distance only if at least one of them is not a member of an existing group. The result of this step is a number of group templates.

Second step is calculating the prototypes for each group previously generated. The prototype is that model from the group that has the minimum of the maximal distances between it and all other templates from group.

Next two steps consist in assigning each template to a prototype previously calculated. A template is assigned to that prototype that has the minimum distance from it. After this step, new groups are formed, each one with a prototype. The prototype is recalculated for each newly formed group. These two steps are repeated until there are no differences between last two classifications.

The result of applying this algorithm at current level is a set of groups of templates, each one with a calculated prototype. The prototypes from current level are considered input templates for the upper level. Then the algorithm is applied at upper level and so on until a single group with a single template which represents the root of the multipath tree is reached.

The number of templates from each group isn't a constant. There can be groups with different number of templates but at least two models in a group. Only the root of the tree is a group with one template.

In Fig. 6 it is represented a small example of a tree hierarchy of contour templates.

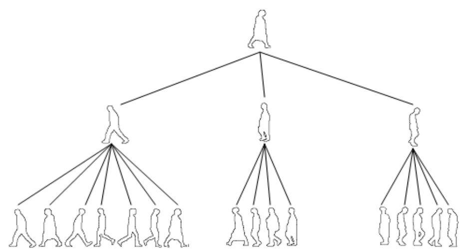


Fig. 6. Example of a pedestrian contour template hierarchy

4 Experimental Results

In this section we present the results obtained on contour extraction methods and then the results of constructing two template trees hierarchies of contour (perfect and imperfect) pedestrian models. All the results have been obtained by testing the proposed methods and algorithms on thousands of images in different environments. We subjectively evaluated the good quality of the results (the shape of the extracted contours and the building of the hierarchies).

4.1 Contour Extraction Results from Sequences of Fixed Background Images

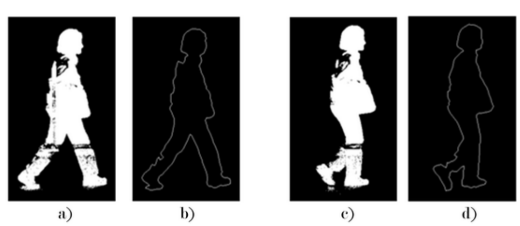


Fig. 7. Pedestrian contour extracted from images with fix background: a), c) Confidence image. b), d) The extracted contour of the corresponding confidence image.

In Fig. 7 is shown the pedestrian contours extracted from confidence images. The important aspect is that the contour extraction algorithm succeeded in extracting the perfect contours of the pedestrians eliminating the holes and merging the distances between the unconnected components in the pedestrian confidence image.

4.2 Contour Extraction Results from Sequences of Dynamic Background Images

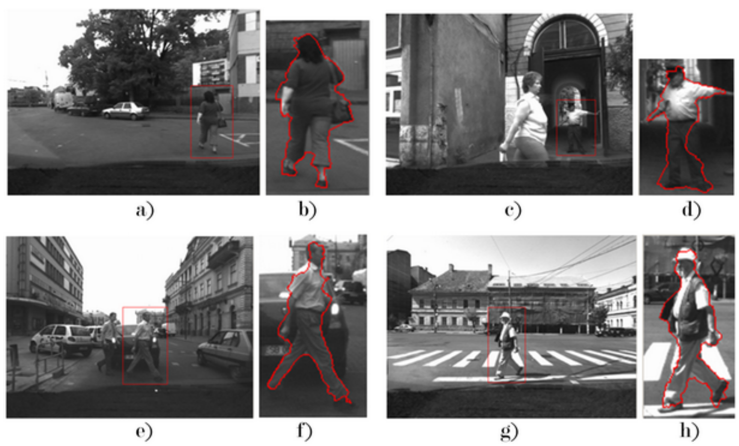


Fig. 8. Pedestrian contour extracted from scene image with dynamic background: a), c), e), g) Scene image and pedestrian selected area. b), d), f), h) The extracted contour of the corresponding pedestrian.

In Fig. 8 is shown the pedestrian contours extracted from traffic images where the background is changing from frame to frame. The contour extraction and refinement algorithm obtained a good result which approximates the contour of the pedestrian in conditions that we have available only grayscale images of the scenes.

4.3 Hierarchy of Pedestrian's Contours Results

Here we present the results of building a pedestrian's contours hierarchy containing "perfect" templates of contours. These templates were obtained by extracting contours with previously presented algorithm using scenes with fixed background. The hierarchy is built of 420 templates of contours previously extracted, obtaining a 4 level-height multi-path tree.

In Fig. 9 is presented an example of a group from the hierarchy having 6 members (contour templates) and one prototype (chosen from those members).

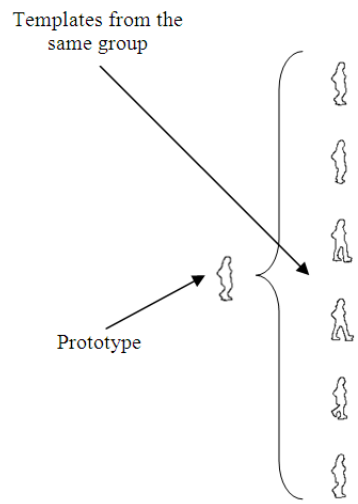


Fig. 9. Example of a group from the hierarchy having 6 contour templates and one prototype

All the members from the group have the same attitude and orientation and we can say that the algorithm worked correctly in building the entire hierarchy.

We have applied the same tree generation algorithm for obtaining the pedestrian's contours hierarchy containing imperfect templates of contours (from images with variable background).

5 Conclusions

We have presented first a pedestrian contour extraction method applied on both fixed and changing background sequences of frames. Scenes of 2D images were acquisitioned from a stereo system camera. We also exploited the 3D information in the contour extraction algorithm to compute the distance points from the camera to the pedestrian area and to filter out the edges that are outside the area of interest when tracing the pedestrian contour. Here we have the contribution of developing the contour extraction algorithm based on depth histogram, edge tracing and active contour refinement for obtaining the pedestrian contours from scenes with variable background.

After extracting the pedestrian contours we have built two pedestrian hierarchies of contour templates extracted from fixed and variable background scenes. The contribution in hierarchy building algorithm consists in the heuristic of initial groups' generation which offers the possibility to automatically choose the initial groups. Another contribution is that we don't fix a number of templates that should contain all the groups formed in the hierarchy tree offering a flexibility and possibility of grouping any number of contours if they have almost the same attitudes and orientations. This new approach permits an automatic run of the hierarchy builder without setting any initial parameters and provides better results.

The constructed hierarchies are very useful in improving the detection of pedestrians using pattern matching technique.

Future work will consist first in refinement of the pedestrians contours extracted from sequences with dynamic background by using at least one color camera. The refinement will consist in extracting a set of textural features which could be exploited for separating the points belonging to the pedestrian surface from those belonging to the background. As future work we also propose the evaluation of the hierarchies in order to identify the best one for obtaining higher performances in real-time pedestrian detection.

References

1. L. Zhenjiang, W. Kunfeng, L. Li, and W. Fei-Yue, "A review of vision-based pedestrian detection for intelligent vehicles," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, December 13-15 2006, pp. 57–62.
2. B. Fardi, U. Schuenert, and G. Wanielik, "Shape and motion-based pedestrian detection in infrared images: a multi sensor approach," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, June 6-8 2005, pp. 18–23.
3. D. Gavrila, J. Giebel, and S. Munder, "Vision-based pedestrian detection: the protector system," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, June 14-17 2004, pp. 13–18.
4. T. Gandhi and M. Trivedi, "Pedestrian collision avoidance systems: A survey of computer vision based recent studies," in *Proceedings of the IEEE Intelligent Transportation Systems Conference*, 2006, pp. 976–981.
5. C. Hilario, J. Collado, J. Armingol, and A. de la Escalera, "Pedestrian detection for intelligent vehicles based on active contour models and stereo vision," in *Proceedings of the IEEE Intelligent Vehicular Sytems*, October 19 2005, pp. 537–542.
6. D. Gavrila and V. Philomin, "Real-time object detection for "smart" vehicles," in *Proceedings of the IEEE International Conference on Computer Vision*, 1999, pp. 87–93.
7. A. Broggi, M. Bertozzi, A. Fascioli, and M. Sechi, "Shape-based pedestrian detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, October 3-5 2000, pp. 215–220.
8. D. Gavrila, "Pedestrian detection from a moving vehicle," in *Proceedings of the European Conference on Computer Vision*, January 01 2000, pp. 37–49.
9. D. Gavrila and S. Munder, "Multi-cue pedestrian detection and tracking from a moving vehicle," in *International Journal of Computer Vision*, 2007, pp. 41–59.
10. S. Nedevschi, C. Tomiuc, and S. Bota, "Stereo based pedestrian detection for collision avoidance applications," in *Proceedings of ICRA Workshop: Planning, Perception and Navigation for Intelligent Vehicles*, 2007, pp. 39–44.
11. D. T. M. Kass, A. Witkin, "Snakes: Active contour models," in *International Journal of Computer Vision*, 1988, pp. 321–331.