System for Real Time Detection of Hands and Pedestrians Movements

Gina CRĂCIUN, Ionela RUSU, Ştefan-Gheorghe PENTIUC, Radu Daniel VATAVU
Stefan cel Mare University of Suceava
str.Universitatii nr.13, RO-720229 Suceava
ginac@eed.usv.ro, iorusu@stud.usv.ro, pentiuc@eed.usv.ro, vatavu@eed.usv.ro

***Abstract* — Interaction with virtual reality systems involves both navigation in virtual environments and virtual object manipulation. Hand gestures are used for direct interaction with virtual objects, while human body postures serves as navigation commands.**

The proposed system is focused on the human detection task in a video stream. Once a target is detected the system continues, in the obtained region of interest, the analysis of the motion history, the object tracking, and determines the convexity of the hand image.

The developed method is able to detect and track multiple targets. The hand convexity analysis presented in the paper may be useful in other gestures recognition systems.

Index Terms — hand posture, computer vision, human computer interaction (HCI), pedestrian detection

I. INTRODUCTION

Human-computer research has been confronted lately by a demanding provocation in what concerns the natural communication between human and computer. The solution to this challenge is to create an easy to use interface that exploits the communication-relevant characteristics of humans.

To achieve natural and immersive human-computer interaction, the hand posture and pedestrian posture could be viewed as interface devices. They both serve a dual propose as communication and manipulation devices. Hand gestures are a powerful human-to-human communication channel, which forms a major part of information transfer in our everyday life. In virtual environments (VE) the object manipulation interfaces uses the hand for selection and manipulation tasks and the pedestrian for navigation tasks.

This paper is focused on employing the human hand and pedestrian as interface devices to a computer. It presents a real-time system that detects the position of the hand and pedestrian and analyzes their movements.

Latest research in computer vision showed that the real-time detection and classification of human hands and pedestrians have some constraints, key issues that can be described in terms of the following major attributes:

Posture: its form is changing very quickly with the movement.

Orientation: the object can be seen from a variety of possible orientation regarding image camera. For example it may face the camera directly or be parallel to it.

Shape: significant differences arising from human to human, e.g. the hand/pedestrian posture of a child is very different from an adult.

Occlusion: occurs when some parts of the hand or pedestrian may be occluded by other objects in the scene. This regularly makes it impossible to observe the full state of the hand and this will get to some wrong configuration for our objects.

The paper is organized as follows. In Section 2 are summarized the Haar-like features and AdaBoost learning algorithm. Section 3 describes the proposed system along with data collection, detector construction, evaluation and performance improvements; this section is divided in two parts, each part details the posture detection and motions analyze process for hand and pedestrian. Conclusions are drawn in Section 4.

II. HAAR LIKE FEATURES AND ADABOOST

Real time hand detection and human pedestrian detection was performed using sets of Haar-like features that encoded information about objects to be detected. Haar-like features were chosen because they can easily encode different features of an image, they are specialized in encoding contrasts exhibited by the subject of interest and their spatial relationship.

Robustness to variations of light and noise (eliminated by subtraction operations) is another reason why it has been used Haar-like features.

Haar-like features are computed similar to the coefficients of transformations based on Haar wave. The features used are rectangular and of varying size, subdivided into white and black regions.

In Figure 1 are presented the three types of Haar-like features [1] used in learning the characteristics of hand and human posture.

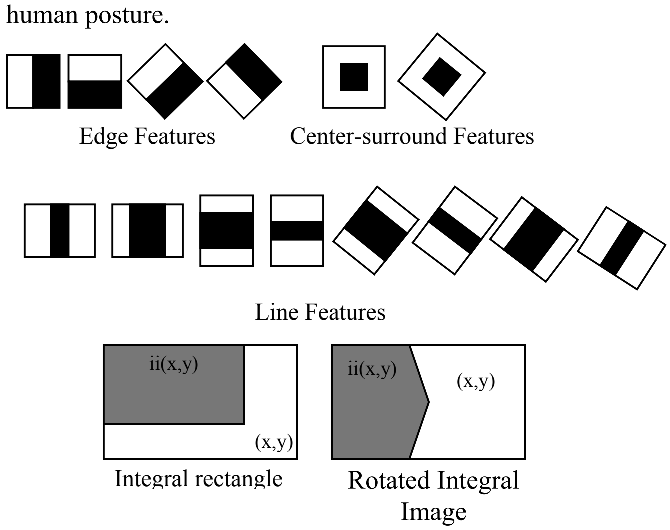

Figure 1. The prototype of Haar-like features used in object detection.

The sum of pixels which lie within the white rectangles are subtracted from the sum of pixels in the black rectangles, according to the formula:

$$f(x)=W_{black}\cdot\sum_{black\_region}(pixel\_value)-W_{white}\cdot\sum_{white\_region}(pixel\_value) \quad (1)$$

where *black_region* and *white_region* represent the image pixels in the black and white regions; W_{black} and W_{white} are certain values which meet the condition of compensation:

$$W_{black}\cdot black\_region = W_{white}\cdot white\_region \quad (2)$$

A classifier based on Haar-like features is trained with several hundred models that contain object of interest, called positive examples, and negative patterns that represent arbitrary images, images that don't contain objects of interest. In the training process the classifiers which try to obtain better performance from stage to stage are called weak classifiers. Through "boosting", the new classifier models are encouraged to become "expert" models for those wrongly classified of previous models.

The cascade of classifiers is built using one of the four classification techniques for the re-weighted step: discrete AdaBoost, real AdaBoost, light AdaBoost and LogitBoost. With each step of the cascade the classifiers become more complex.

Classifier result is a multiple weak classifiers (cascade), which are applied subsequently to a region of interest; classification is complete when the form is rejected by one of series of classifiers and shape was evaluated by all classifiers. The final classifier is a strong one, composed by a linear combination of the selected weak classifiers. The cascade of classifiers C1, C2... Ck is built like in Figure 2.

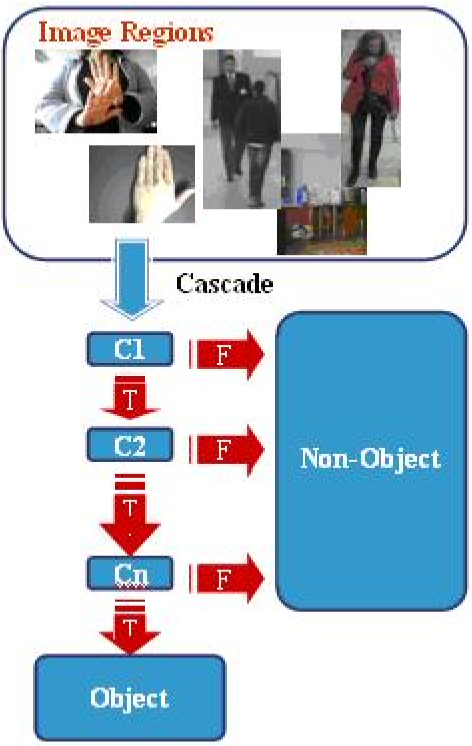

Figure 2. AdaBoost Cascade Classifiers.

When the classifier is complete, it will be applied to a region of interest of an input image. Finding an object in the input image is accomplished by moving the search window on the entire image area, checking to each location. The classifier is designed so that it can find objects of interest to different search window sizes, more efficient than resizing the image itself [10].

III. HAND AND PEDESTRIAN DETECTION SYSTEM OVERVIEW

The architecture for the detection system developed is the same as the architecture proposed by Viola-Jones [1] for face detection. Figure 3 shows the system architecture for hand and pedestrian detection.

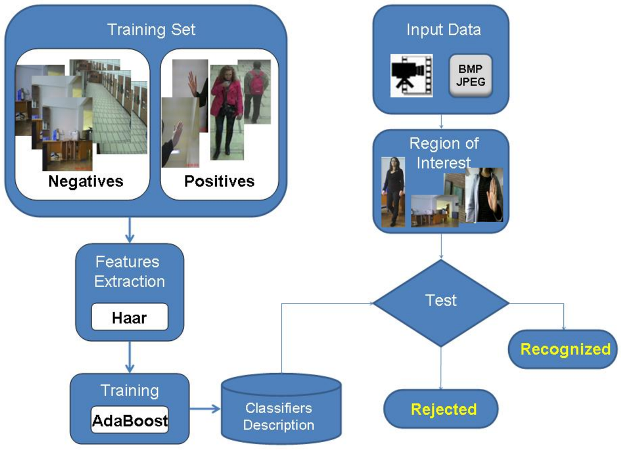

Figure 3. Detection System Architecture.

Detection process starts from building a training set of images. Learning set is a database with a large number of examples that have the same size, called positive images because they contain objects of interest, and a number of images that do not contain objects of interest, called negative images. **Error! Reference source not found.**Figure 4 shows the training set used by the system.

Figure 4. Positive (up) / negative (down) images subset.

Before building the classifier the most important things is the selection of some expressive features and their ability to be applied in real world scenes.

Evaluation of thousands of features associated with each image is inefficient in time computing, so it is assumed that only a number of these windows can be classified. The problem is to find these features and finally to obtain the training classifier. Each region of the image will last through the entire chain of filters to determine if it is rejected or recognized.

The trained classifier is used to scan an input image and extract the interest region. The scanning procedure supposes to match the classifier's features on the image search box. Looking for the interest region, the image search box is resized to determine the interest region at different scales. The procedure repeats several times for objects of unknown sizes.

A. Hand Posture Detection, Tracking and Motion Analysis

A hand posture is defined as a static hand pose; making a fist and holding it in a certain position is considered a hand posture. A hand gesture specifies a dynamic movement, such as waving goodbye. The dynamic movement of hand gestures includes two aspects: global hand motions and local finger motions. Global hand motions change the position or orientation of the hand while local finger motions involve only moving the fingers in some way but without changing the position and orientation of the hand. Hand gestures can be seen as a complex set of hand actions constructed by global hand motions, and a series of hand postures that act as transition states [2][5] [11].

Hand posture detection and tracking represent the low level of this architecture and motion analysis represents the high level [3].

To achieve the real-time requirement, a series of cascades of classifiers are trained for each hand posture based on Haar-like features and the AdaBoost learning algorithm.

A parallel cascades structure is implemented to identify the selected hand postures from the camera's live input. For each image achieved or frame taken it is chose a rectangular area to establish whether that region is an object or non-object. Parameters are transmitted via an xml file; with the information gathered in this file it can be detect the presence of the hand at a certain location in the image. The scanning window size changes so that it can detect hands at various distances from the input camera.

The results of detection process are shown in Figure 5:

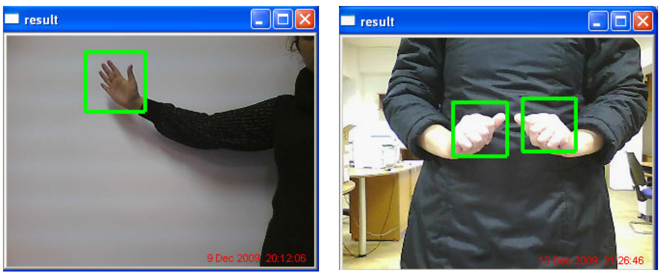

Figure 5. Hand posture detection.

Motion history (motion history image- MHI) represents a new approach to visual perception models based on the representation of action. Approach assumes that a human can easily distinguish action scenes in a very short time, almost immediately, even for a very low resolution, without knowledge or information about the structure of scene. Human representations of the action are implied descriptions of models based on non-visual perception of image motion. The advantage is that the movement of several frames can be summarized into a single image and using this model is independent by time. To isolate a moving object it is necessary to extract the background from two successive frames [4].

History user's hand movement and the tracking step, which uses Camshift, are illustrated in Figure 6.

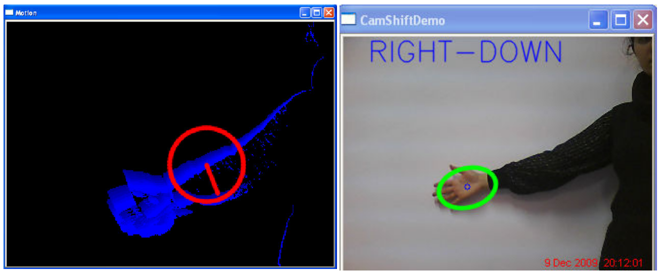

Figure 6. Motion History / Hand Motion Analysis.

When the silhouette of a moving object is determined, all pixels of zero value are decremented by one. The pixels with values below a certain threshold are also zero. The pixels from the most recent difference frames are initialized to a maximum value. Therefore, the latest difference image due to the latest moves is the highest value and older movements are encoded by successive decreasing values [8]. Time allocation motion can be based on selected key frames, from time to time depending on activation or objects movement in the scene pursued.

The results from hand posture detection are used to make a short analysis for the object detected and to develop other concepts that are closely related with our problem [14]. The convex polygon, that encloses the hand, is detected to highlight the hand's fingers. The whole experiment is based on the methods described above. After the detection process of a region of interest in a particular image, the result is subjected to a process of edge detection and find contour. A way of comprehending the shape of an object or contour is to compute a convex hull for the object and then compute its convexity defects. The shapes of many complex objects are well characterized by such defects. These convexity defects offer a means of characterizing not only the hand itself but also the state of the hand [8].

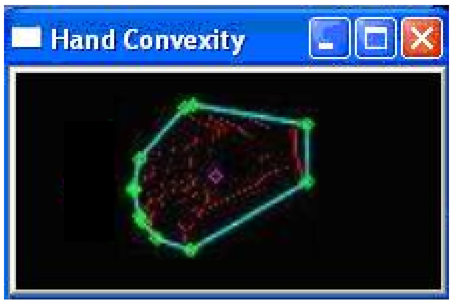

Figure 7. Hand convexity.

Figure 7 illustrates the concept of hand convexity. The convex hull is pictured as a blue line around the hand and the green points highlights the possible fingers locations. As it can see there are problems that appear because of convexity defects and need to be solved.

B. Pedestrian Movements

In this part we present the other module from our system – the pedestrian detection module. The main idea is to build a classifier based on features extracted from images containing models of human posture.

The robust segmentation and tracking of pedestrians under unconstrained conditions introduces a multitude of complicating factors that have been mention above. These complicating factors have to be acknowledged and addressed by computer vision systems if robust pedestrian detection is to become possible in real-world scenarios [12].

The first thing to do for tracking pedestrian's motion is to detect it. The pedestrian detection process is described in [13]. Figure 8 and Figure **9** present scenes with one and two persons detected.

Figure 8. Single Posture Detected.

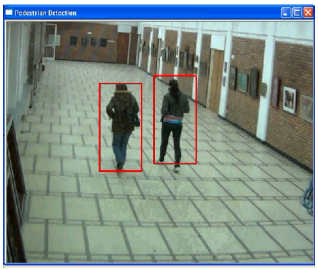

Figure 9. Multiple postures detected.

After human posture is detected, proceed to follow it with Camshift algorithm [8]. The CamShift algorithm finds the center and the size of the object on a color probability image frame. The probability is created via a histogram model of a specific color. The tracker moves and resizes the search window until its center converges with the center of mass. Compared with regular mean shift that uses static distributions, CamShift can effectively track dynamically changing probability distributions in a visual scene [15]. It will return the image region where it's possible to be a person. Thus, the CamShift window will follow AdaBoost detection window (the CamShift window is returned less than AdaBoost search box). If in a frame is detected more people, the tracking algorithm will be applied for each target separately.

Figure 10 and Figure 11 present the results of pedestrian detection and tracking system in scenes with more persons.

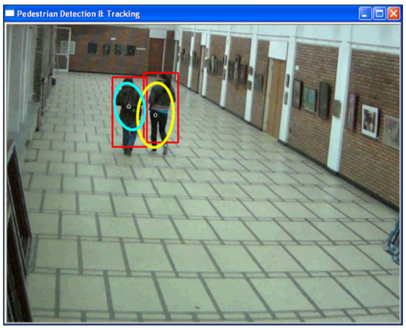

Figure 10. Detection and tracking of multiple pedestrian.

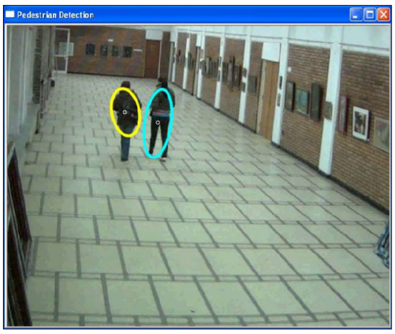

Figure 11. Multiple person tracked.

IV. CONCLUSION

Hand and pedestrian detection and tracking system is good but not perfect. Sometimes detection module fails to detect human hand and pedestrian in each frame or at the same time, therefore the motion tracking algorithm will not be so efficient. A solution may be modeling the shapes of movements [9].

V. ACKNOWLEDGEMENTS

This paper was supported by the project "Knowledge provocation and development through doctoral research PRO-DOCT - Contract no. POSDRU/88/1.5/S/52946 ", project co-funded from European Social Fund through Sectoral Operational Program Human Resources 2007-2013

REFERENCES

[1] P. Viola and M. J. Jones, "Robust real-time face detection", in International Journal of Computer Vision, 2004.

[2] Prateem C., Prashant S., Ankit M., "Hand Gesture Recognition – A Comparative Study", in Proceedings of the International Multi Conference of Engineers and Computer Scientists 2008 Vol I IMECS 2008, 19-21 March, 2008, Hong Kong.

[3] A Erol, G Bebis, M Nicolescu, RD Boyle, X Twombly, "Vision-based hand pose estimation: A review", Computer Vision and Image Understanding Volume 108, Issues 1-2, October-November 2007, Pages 52-73 Special Issue on Vision for Human-Computer Interaction.

[4] M.K. Bhuyan, P.K. Bora, and D. Ghosh, "Trajectory Guided Recognition of Hand Gestures having only Global Motions", International Journal of Computer Science, Fall 2008.

[5] Q. Chen, F. Malric, Yi Zhang, M. Abid, "Interacting with Digital Signage Using Hand Gestures", Institute of Computer Graphics and Image Processing, Tianjin University, China, 2008.

[6] JS. Munder and D.M. Gavrila, "An Experimental Study on Pedestrian Classification", IEEE Transactions on Pattern Analysis and Macjine Intelligence, vol. 28, no. 11, november 2006

[7] G. Monteiro, P. Peixoto, U. Nunes, "Vision based pedestrian detection using haar-like features", in Robotica- Festival nacional de robotica, Guimaraes 2006

[8] Gary Bradski, Adrian Kaehler, "Learning OpenCV", 2008, Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

[9] Radu-Daniel Vatavu, Ştefan Gheorghe Pentiuc, Laurent Grisoni, Christophe Chaillou, "Modeling Shapes for Pattern Recognition: A Simple Low-Cost Spline-based Approach", Advances in Electrical and Computer Engineering, Volume 8 (15), Number 1 (29), 2008, University „Stefan cel Mare" of Suceava, ISSN 1582-7445, pp. 67-71

[10] http://opencv.willowgarage.com/wiki/FaceDetection

[11] Pragati Garg, Naveen Aggarwal, Sanjeev Sofat, "Vision Based Hand Gesture Recognition", World Academy of Science, Engineering and Technology 49 2009

[12] Philip Kelly, "Pedestrian detection and tracking using stereo vision techniques", December, 2007

[13] Ionela Rusu, "Pedestrian Detection Using Vision-Based Techniques", Sisteme Distribuite (Suceava - online), p. 85-89, ISSN 2067 – 5259, Suceava, 2010.

[14] Elena-Gina Craciun, Ştefan-Gheorghe Pentiuc, "Hand Detection using a Set Of Classifiers Based on Haar-Like features", Sisteme Distribuite (Suceava - online), p. 90-93, ISSN 2067 – 5259, Suceava, 2010.

[15] Ovidiu Ungurean, "Cercetari privind comunicarea gestuala cu sistemele de calcul", Suceava 2009.