

Moving Object Detection Unaffected by Cast Shadows, Highlights and Ghosts

Duarte Duque, Henrique Santos and Paulo Cortez
Information Systems Department
University of Minho
4800-058 Guimarães, Portugal
{duarteduque, hsantos, pcortez}@dsi.uminho.pt

Abstract—This paper describes a new approach to perform segmentation of moving objects in real-time from images acquired by a fixed color video camera and is the first tool of a major project that aspires to recognize abnormal human behavior in public areas. The moving objects detection is based on background subtraction and it is unaffected by changes in illumination, i.e., cast shadows and highlights. Furthermore it does not require a special attention during the initialization process, due to its ability to detect and rectify ghosts. The results show that with image resolutions of 380x280 at 24 bits per pixel, the time spent in the segmentation process is around 80ms, in a 32 bits 3GHz processor based computer.

I. INTRODUCTION

The new generation of surveillance systems aims the automatic monitoring, recognition and prediction of human activity. To successfully accomplish this task, a surveillance system has to perform an accurate segmentation of moving objects in the scene, free from cast shadows and reflections. However, despite the importance of the segmentation algorithms quality, these systems have also severe time restrictions to accomplish. Therefore, the computational cost of each technique used in the system should be considered. The key of the success is to achieve an optimal balance between efficiency and computational cost of each algorithm.

In this paper, we propose a novel algorithm to segment moving objects in real-time from images acquired by a fixed color video camera. The moving objects detection is based on background subtraction and it is unaffected by changes in illumination due to cast shadows and highlights.

Our approach does not require special attention during the initialization process, i.e., the system can be initialized with moving objects in the scene. This can be done, due to its ability in detection and rectification of ghosts.

The paper is organized as follows. In Section II we briefly review some related work. Then, in Section III the shadows and highlights detection techniques are described. In Section IV we present an algorithm to extract the moving edges of objects in the scene. Next, Section V describes the segmentation process completely, based on background subtraction, which includes the ghost detection algorithm. Experimental results are presented in Section VI and, in Section VII, we draw some conclusions.

This work was supported by the FCT under the grant SFRH/BD/17259/2004.

II. RELATED WORK

The segmentation of moving objects is an important research area having impact in a vast range of applications, such as content-based video retrieval, traffic and human surveillance. The most popular approaches for segmenting moving objects include temporal differencing [1], optical flow [2] and background subtraction [3].

Temporal differencing based methods produce poor results because of the existence of large regions of false positives and false negatives. The false positives are generated in regions opposed to the directions of the object motion. False negatives are produced when the object inside colour is uniform and the distance covered by the object is less than the uniform colour area.

Optical flow based methods allows good detection of moving objects, free of ghosts, in the presence of camera motion. Yet this technique is computationally complex and therefore is not suitable for use in real-time surveillance systems.

The background subtraction consists in the subtraction of the current image from a reference one, and is commonly used mainly due to its simplicity and low computational cost.

Despite the amount of work developed in the field of moving objects segmentation based on background subtraction techniques, there is a question that remains without a consensual answer from the research community: What should contain a background image?

Some [4] state that “a robust detection system should continue to ‘see’ objects that have stopped, and disambiguate between overlapping objects in the image.” It seems a good definition for a detection system, although it should not be applied directly to the background actualization. With this approach the background will not be updated in regions of pixels belonging to objects that stop their motion in the scene. This constraint will prevent the actualization in the background image of ghosts generated by the motion of objects that are in the scene in the system initialization phase.

Others researchers [3] [5] model the background by pure statistical measures of the images captured over the time. Examples of these approaches are median filters, infinite impulse response filters [5] and statistical probability distributions like Gaussians and Mixture of Gaussians [3]. The drawback of this solution is that it will fail if there is a large

amount of motion in the scene for a wide period of time. When this occurs, the background will include the colours of the objects that more frequently appear in the scene, originating errors in the segmentation process.

From our point of view, the background image should contain all the objects observed in the scene that are not in motion at the present time. This should be done in such a way that once a moving object stops, it should be instantaneously included as part of the background. In the same mode, when a still object starts moving it should be immediately excluded from the background image.

To achieve this goal, we developed a new segmentation algorithm that is described in this paper.

III. SHADOW AND HIGHLIGHT DETECTION

The presence of cast shadows (projected by objects in the direction of the light source) and highlights in an image can modify the perceived object shape, causing its incorrect segmentation.

Here we present the process used for the segmentation of image areas affected by cast shadows and highlights, based in the previous work from [6].

For this purpose, a background image B_n and a current image I_n , both in the HSV color space, are used. From those images we generate a shadow mask SM_n and a highlight mask LM_n , which will define the image areas where shadows and highlights are detected.

In these masks each pixel will have value "1" if it is marked as shadow or highlights, and value "0" otherwise. These masks will be used later in conjunction with the motion segmentation algorithm, in order to avoid false positives. The following equations define the process of computing the shadow and highlight masks:

$$SM_n(x, y) = \begin{cases} 1 & \text{if } \alpha \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \beta \wedge \left| I_n^S(x, y) - B_n^S(x, y) \right| \leq \tau_S \wedge \\ & \left| I_n^H(x, y) - B_n^H(x, y) \right| \leq \tau_H \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$LM_n(x, y) = \begin{cases} 1 & \text{if } \frac{1}{\beta} \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \frac{1}{\alpha} \wedge \left| I_n^S(x, y) - B_n^S(x, y) \right| \leq \tau_S \wedge \\ & \left| I_n^H(x, y) - B_n^H(x, y) \right| \leq \tau_H \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The $I_n^H(x, y)$, $I_n^S(x, y)$ and $I_n^V(x, y)$ represent respectively the hue, saturation and value components at coordinate (x, y) of the input image I at time n . The same notation is applied to the background image B .

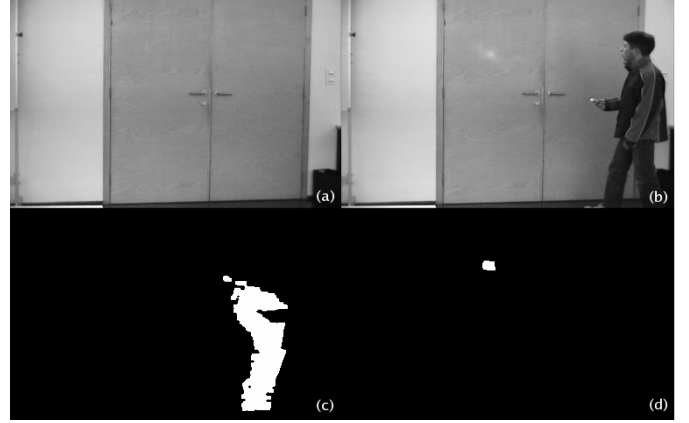


Fig. 1. (a) Background image B_n ; (b) current image I_n ; (c) shadow mask SM_n ; (d) highlight mask LM_n .

The α parameter depends on the light source and the reflectance and irradiance properties of the objects in the scene. High intensive light sources and high reflective or irradiant objects imply lower α values. Usually α ranges from 0.65 to 0.80 (this reference values were obtained by lab experiences taking as base the values proposed by [6]). On the other hand, the β parameter is used to prevent the incorrect classification of shadows as part of a moving object, points where the background was slightly changed by noise. The β typically ranges from 0.90 to 0.95.

Parameters τ_S and τ_H are the maximal variation allowed for the saturation and hue components, respectively. We define τ_S as 15% of the digitizer's saturation range. The variation of hue should not pass the 60 degrees. This value is obtained through the division of the hue range (360°) by the six colors (red, yellow, green, cyan, blue and magenta). Fig. 1 shows the output of shadow and highlight masks generation algorithm in a scene with a person pointing a light to the door.

The use of separated masks for shadow and highlight detection allows a better result concerning noise removal, than was obtained with a common mask. The noise is removed from each mask by a morphologic open operator followed by a close operator, using a 3×3 structuring element.

IV. EXTRACTION OF MOVING EDGES

In this section a method to extract the moving object edges in a scene is described. The basic idea is to get those edges by the interception of the edges extracted from the result of the absolute difference between the current image I_n and the previous one I_{n-1} , with the edges computed with the absolute difference between the background image B_n and the actual image.

The choice of the technique to be used in the edge detection is critical. It must both present good results and low response times. For this purpose the Canny, Sobel and Prewitt algorithms were tested. Despite the result quality, the computational requirements for those algorithms exceed significantly the time requirements for our application. To overcome this drawback, a faster and simple yet efficient edge detection algorithm is used.

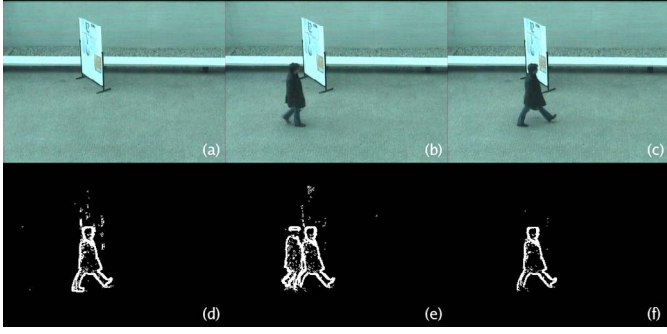


Fig. 2. (a) Background image B_n ; (b) previous image I_{n-1} ; (c) current image I_n ; (d) edges of $|I_n - B_n|$; (e) edges of $|I_n - I_{n-1}|$; (f) moving edges ME_n .

It compares each pixel value (V component of the HSV color space) with its 4 connected neighbor pixels. If the difference between the pixel and one of its neighbors is higher than a given threshold, the pixel is marked as an edge. Based in this description, the following operator will then produce a binary image with all edges detected.

$$\Phi(D(x, y)) = \begin{cases} 1 & \text{if } |D(x, y) - D(x-1, y-1)| > T \quad \checkmark \\ & |D(x, y) - D(x-1, y+1)| > T \quad \checkmark \\ & |D(x, y) - D(x+1, y-1)| > T \quad \checkmark \\ & |D(x, y) - D(x+1, y+1)| > T \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

As stated before, the Φ operator is applied to both the difference between the current image and the background image, and the difference between the current image and the last image. The intersection of the two binary images obtained will contain the moving object edges. The following expression describes the overall process:

$$ME_n = \Phi(I_n^V - B_n^V) \cap \Phi(I_n^V - I_{n-1}^V) \quad (4)$$

V. THE SEGMENTATION PROCESS

For each new image a binary representation of pixel motion (called PMM_n - primary motion mask) is calculated based on the absolute difference, for each pixel, between the V components of the HSV color space, of the current image I_n and the background image B_n . This is given by:

$$PMM_n(x, y) = \begin{cases} 1 & \text{if } |I_n^V(x, y) - B_n^V(x, y)| > T \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where T is a constant threshold.

Next, the detection of cast shadows and highlights are accomplished as defined in the Section III, creating the binary masks SM_n and LM_n , which represent the shadow and highlight regions respectively.

Those masks are then removed from the PMM_n with a simple intersection operation. The final result is stored in the motion mask (MM), and is defined by the following expression:

$$MM_n = PMM_n \cap \neg SM_n \cap \neg LM_n \quad (6)$$

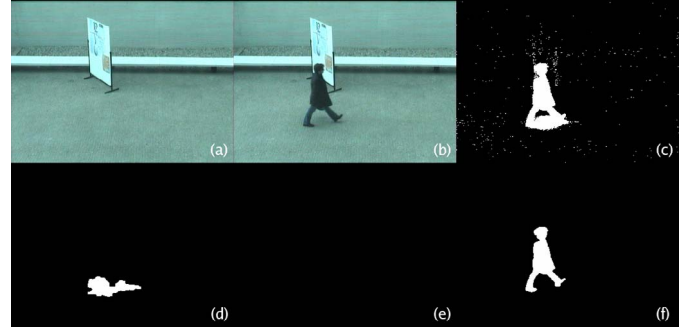


Fig. 3. (a) Background image B_n ; (b) current image I_n ; (c) primary motion mask PMM_n ; (d) shadow mask SM_n ; (e) highlight mask LM_n ; (f) resulting motion mask MM_n after shadows, highlight and noise removal.

The MM_n is now free from false positives produced by shadows and highlights, but it is still possibly affected by ghosts, i.e., false positives generated by the motion of background objects.

In the next phase, the extraction of moving edges is performed as described in Section IV, producing a ME_n mask.

Afterward, it is necessary to count and label the blobs defined in the motion mask MM_n , and compute their main characteristics, such as the center of mass, area, perimeter and bounding box (these are the minimum attribute set required for analysis purpose). Blobs with area less than a predetermined value are considered as noise and are removed from the motion mask.

After this we compute the blob edges mask (BEM). The BEM_n is a binary mask that contains edges of the blobs detected so far, which, as remarked before, also includes undesirable ghosts. To eliminate them, the perimeter of a blob is compared with the perimeter obtained by the intersection of the ME_n and BEM_n masks. If the difference between those values is above 90%, the blob is marked as a potential ghost.

Next, the clustering of the blobs is done, in order to define objects. The clustering process is based on two rules: blobs with distances between their centers of mass less than a predefined value (depending on object types) are connected; blobs remain connected if there is a uniform color path between them.

An object, i.e., group of connected blobs, is classified as a ghost if all of its blobs are marked as potential ghosts. In that case it is excluded from the segmentation, in the MM_n mask.

Finally, the background image is updated in the HSV color space. Pixels belonging to shadows, highlights and objects identified as moving ones remain unchanged. Background pixels of segmented regions classified as ghosts are set with the actual image value. The remaining pixels are updated by an infinite impulse response filter in order to adapt the reference image to slight luminosity changes. The background updating can therefore be expressed by:

$$B_{n+1}(x, y) = \begin{cases} B_n(x, y) & \text{if } motion \vee shadow \vee highlight \\ I_n(x, y) & \text{if } ghost \\ \alpha B_n(x, y) + (1 - \alpha) I_n(x, y) & \text{otherwise} \end{cases} \quad (7)$$

where α controls how fast are the changes introduced.

VI. EXPERIMENTAL RESULTS

The described process was implemented in a standard computer, based in a 32 bits 3GHz processor, running a GNU/linux operating system. The video capture board is a BT878 based one, allowing a maximal resolution of 720x576.

Several experiments were made using image resolutions of 380x280 at 24 bits per pixel, with both indoor and outdoor scenarios. Fig. 4 shows the results obtained with an outdoor scene. In this figure, the first row presents the frames 0, 8, 216 and 959 from the acquired test sequence; the second row displays the correspondent's background images; and the third row shows the segmentation results.

In order to show how fast the background ghost correction performs, a momentarily stopped person is shown in the first acquired frame (top left image) and, despite the short time interval (8 frames), he/she was removed from the background. The results demonstrate that after the person leaved the area occupied by the ghost, it is immediately removed from the background, as illustrated by the images in the second column of fig. 4.

The fourth column shows again a person included in the background, since he/she has stopped moving. As expected, the algorithm includes instantaneously in the background non-moving objects.

Tests were made in scenes with a number of persons ranging from 1 to 10. Under the referred conditions, the overall segmentation process takes about 80ms. This response time is low enough for typical surveillance systems where a frame rate of 12.5 is acceptable. Nevertheless, taking in account other high-level functions required by our projected system, some optimization of both the algorithms and the architecture should be necessary.

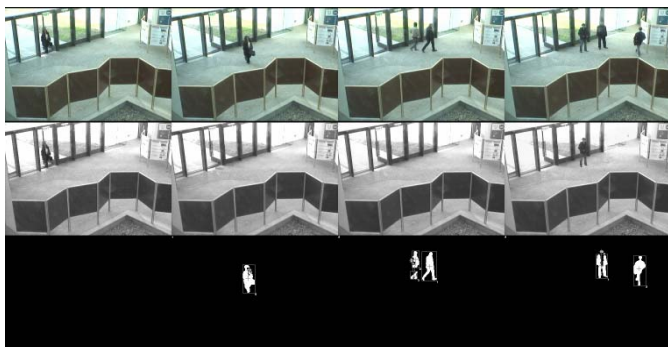


Fig. 4. Example of moving objects segmentation.

VII. CONCLUSIONS

The proposed method for moving objects segmentation has shown to be accurate, robust, while accomplishing the real-time constraint. The system behavior is in conformity with our definition of background. In particular, ghosts are removed from the segmentation results immediately after an object leave the ghost area, and stopping objects are instantly included as part of the background. While our method performs well in detection of human motion in a typical surveillance scenario, when dealing with large background objects that start to move, the time that it takes to rectify the reference image can be a drawback.

The use of techniques to exclude cast shadows and highlights from the segmentation allow a precise detection of moving objects.

In future work we will try to improve the performance of the shadow and highlight detector by applying automatic tuning to the α and β parameters, in order to adapt to the changes in illumination that take place during the period of the day.

ACKNOWLEDGMENT

Special thanks to all department colleagues that help in paper revising.

REFERENCES

- [1] R.T. Collins, A.J. Lipton, T. Kanade, H. Fujiyoshi D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto and O. Hasegawa, "A System for Video Surveillance and Monitoring," *VSAM Final Report*, Carnegie Mellon University, Pittsburgh, 2000.
- [2] G. Havelly and D. Weinshall, "Motion of Disturbances: Detection and Tracking of Multi-Body Non-Rigid Motion," *Computer Vision and Pattern Recognition*, IEEE Computer Society, Puerto Rico, pp. 897-902, 1997.
- [3] C. Stauffer and W.E.L. Grimson, "Learning Patterns of Activity Using Real-Time Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, pp. 747-757, 2000.
- [4] H. Fujiyoshi and T. Kanade, "Layered Detection for Multiple Overlapping Objects," *16th International Conference on Pattern Recognition*, IEEE Computer Society, Québec, pp. 156-161, 2002.
- [5] A. Lipton, H. Fujiyoshi and R. Patil, "Moving Target Classification and Tracking from Real-Time Video," *Workshop on Applications of Computer Vision*, IEEE Computer Society, 1998.
- [6] R. Cucchiara, C. Grana, G. Neri, M. Piccardi and A. Prati, "The Sakbot System for Moving Object Detection and Tracking," *10th ACM International Conference on Multimedia*, ACM Press, France, pp. 223-226, 2002.