

Foreground detection by using multi features

Dr. Emad Kadum Jabbar, Dr. Ahmed Tariq Sadiq, Nuha Jameel Ibrahim

Computer Sciences Department University of Technology/Baghdad

Abstract: Foreground detection is an essential task in many applications and a commonly used approach to obtain foreground objects from the background. Many techniques exist, but due to shadows and changes in illumination the detection of foreground objects from the background remains challenging. The most popular method for foreground detection is background subtraction where a background model needs to be maintained. In this paper, we propose a method for detecting objects based on texture, color and edge measure. The proposed method is characterized through robustness against illumination changes, shadows and little background changes, and it is validated with the experimental results.

Keywords: foreground detection, background subtraction, object segmentation.

I. Introduction

Background subtraction is a common method for detecting moving objects, and it has been widely used in many surveillance systems. Moving object detection using background subtraction compares an input image with a background model previously prepared, and picks up regions in an input image, which do not match a background mode. Moving object detection using background subtraction has the advantage of not requiring previous knowledge of moving objects such as shapes or movements. On the other hand, background subtraction has a problem in that it cannot discriminate moving objects from backgrounds when these backgrounds change significantly. There are many types of background changes, which we put into two categories; one is global changes in images such as transitions of sunlight or adjustments of ceiling light luminance, and the other is local changes in images such as waving of tree branches or fluctuations of water. Many kinds of methods have been proposed for generating a background model to cover changes in backgrounds and robustly detecting moving objects.

These different methods are classified following the model used:

1. Basic Background Modeling[1,2,4]: Background is modeled using the average, the median or the histogram analysis. The foreground detection is made as follows:

$$d(I_t(x, y, t) - B_t(x, y, t - 1)) > T \quad (1)$$

Otherwise, pixels are classified as background is a constant threshold, $I_t(x, y, t)$ and $B_t(x, y, t)$ are respectively the current and the background images.

2. Statistical Background Modeling[3,7]: Background is modeled using a single Gaussian, a Mixture of Gaussians or a Kernel Density Estimation. Statistical variables are used in the foreground detection to classify the pixels as foreground or background.
3. Background Estimation[5,6]: Background Representation is estimated using a filter. For the foreground detection. This filter may be a Wiener filter.

These different methods use the features texture, color and edge separately. We propose a method for detecting moving objects with a combination of many features. This method use color features obtained from RGB,HSV and YCbCr color space, texture and LOG filter to find similarity measures and threshold the result to find the objects.

In Section 2 we describe color space ,texture, edge features. In Section 3, we describe the proposed method for the foreground detection. In section 4, we describe the evaluation experiments of our proposed method. In section 5, we conclude this paper.

II. Features similarity

In this section, we describe the method for detecting moving objects by comparing current and background images. The way for comparison in defining a similarity measure between current and background images. The similarity measure is computed by using color, edge and texture features.

2.1 Color similarity

In color similarity; three color spaces are used for detecting moving objects RGB,HSV and YCbCr color space. Then the similarity measure is computed for each component

2.1.1 HSV color space

HSV color Space stands for hue (H),saturation (S) and value (V). The hue which is essentially the chromatic component of our perception may. Saturation is a measure of colorfulness or whiteness in the color perceived. The value (V) essentially provides a measure of the brightness of colors[8]. The HSV image may be computed from RGB using the following transformation:

$$\begin{aligned} H &= \tan[3(G-B)/(R-G)+(R-B)], \\ S &= 1 - (\min(R,G,B)/V), \\ V &= (R+G+B/3) \end{aligned} \tag{2}$$

The similarity measure $SH_n^C(x,y)$ for HSV color space is computed as in:

$$SH_n^C(x,y) = \begin{cases} H_n^C(x,y)/H_n^B(x,y) & \text{if } H_n^C(x,y) < H_n^B(x,y) \\ 1 & \text{if } H_n^C(x,y) = H_n^B(x,y) \\ H_n^B(x,y)/H_n^C(x,y) & \text{if } H_n^C(x,y) > H_n^B(x,y) \end{cases}$$

where $n=\{1,2,3\}$ is one of the three color features, Band C represent respectively the background and the current images at time t. B can be obtained using any of the background modeling method. Note that $SH_n^C(x,y)$ is between 0 and 1.

2.1.2 YCbCr color space

The color space in which luminance and chrominance are separately represented is the YCbCr, where Y, Cb, and Cr components are scaled and shifted versions of the Y, U, and V components. The Y component takes values from 16 to 235, while Cb and Cr take values from 16 to 240[9]. They are obtained from gamma-corrected R, G, and B values as follows:

$$\begin{matrix} Y \\ Cb \\ Cr \end{matrix} = \begin{matrix} \left. \begin{matrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{matrix} \right\} \begin{matrix} R \\ G \\ B \end{matrix} \end{matrix} \tag{3}$$

The similarity measure $SY_n^C(x,y)$ for YCbCr color space is computed as in:

$$SY_n^C(x,y) = \begin{cases} Y_n^C(x,y)/Y_n^B(x,y) & \text{if } Y_n^C(x,y) < Y_n^B(x,y) \\ 1 & \text{if } Y_n^C(x,y) = Y_n^B(x,y) \\ Y_n^B(x,y)/Y_n^C(x,y) & \text{if } Y_n^C(x,y) > Y_n^B(x,y) \end{cases}$$

where $n=\{1,2,3\}$ is one of the three color features, Band C represent respectively the background and the current images at time t. B can be obtained using any of the background modeling method. Note that $SY_n^C(x,y)$ is between 0 and 1.

2.2 Texture similarity

The texture-based method for background subtraction is based on the Local Binary Pattern (LBP) texture measure. The LBP is a powerful means of texture description[10]. The operator labels the pixels of an image block by thresholding the neighborhood of each pixel with the center value and considering the result as a binary number:

$$LBP(x,y) = \sum_{i=0}^{N-1} s(g_i - g) 2^i \tag{4}$$

where g corresponds to the grey value of the center pixel (x; y) and gi to the grey values of the N neighborhood pixels. The function is defined as follows:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

The original LBP operator worked with the 3x3 neighborhood of a pixel. Then, the texture similarity measure $S^T(x,y)$ is computed as in:

$$S^T(x,y) = \begin{cases} M^C(x,y)/M^B(x,y) & \text{if } M^C(x,y) < M^B(x,y) \\ 1 & \text{if } M^C(x,y) = M^B(x,y) \\ M^B(x,y)/M^C(x,y) & \text{if } M^C(x,y) > M^B(x,y) \end{cases}$$

where $M^B(x,y)$ and $M^C(x,y)$ represent respectively the texture in the background and the current images at time t . Note that $S^T(x,y)$ is between 0 and 1.

2.3 Edge similarity

Laplacian operator is a second derivative operator often used in edge detection. Compared with the first derivative-based edge detectors such as Sobel operator, the Laplacian operator may yield better results in edge localization. Unfortunately, the Laplacian operator is very sensitive to noise[11].

$$S^E(x,y) = \begin{cases} E^C(x,y)/E^B(x,y) & \text{if } E^C(x,y) < E^B(x,y) \\ 1 & \text{if } E^C(x,y) = E^B(x,y) \\ E^B(x,y)/E^C(x,y) & \text{if } E^C(x,y) > E^B(x,y) \end{cases}$$

where $E^B(x,y)$ and $E^C(x,y)$ represent respectively the edge in the background and the current images at time t .

III. Proposed Method

In this section, we describe the algorithm for detecting moving objects in using a background model based on different features by computing similarity measure for each components features.

So, for each pixel, color, texture and edge similarity measures are computed as explained in section 2 from the background and the current image.

We define the set of criteria, color features of the chosen color space (RGB,HSV,YCbCr), texture feature and edge feature.

$$M(x,y) = SH_3(x,y)*0.2 + SY_1(x,y)*0.2 + SY_2(x,y)*0.3 + S^T(x,y)*0.25 + SR(x,y)*0.05$$

Where $SH_3(x,y)$ is the value component (v) in HSV color space, $SY_1(x,y)$ is the luminance component(Y) in YCbCr color space, $SY_2(x,y)$ is the chrominance component(Cb) in YCbCr color space, $SR(x,y)$ is the RGB color space, $S^T(x,y)$ is the texture and $E(x,y)$ the leblasian filter for edge detection.

The pixel at position (x,y) is considered as foreground if its value is less than a certain threshold and edge similarity measure is equal to zero..

if $M(x,y) < Th$ then (x,; y) is foreground,

Input: current frame, background frame.

Output: moving objects.

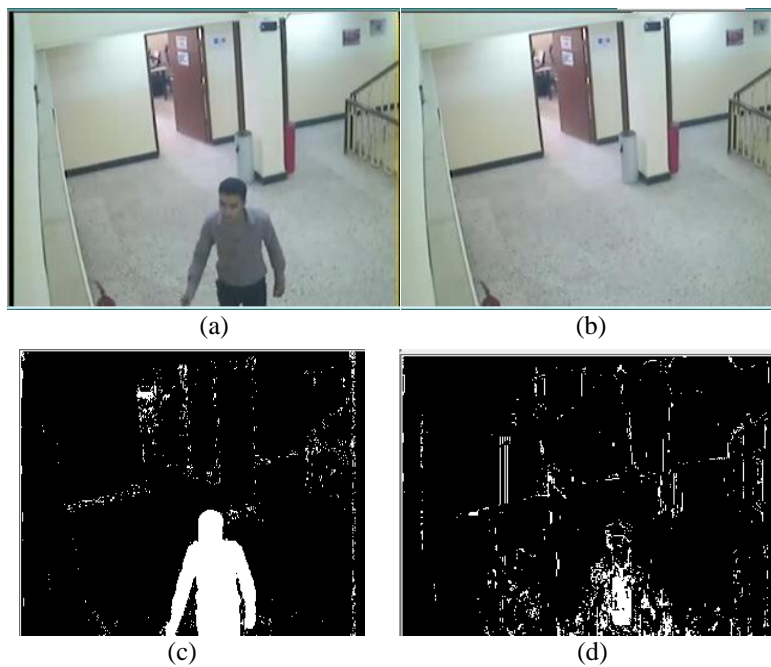
- 1: Start
- 2: Do Until not EOF(image file).
- 3: Convert RGB to HSV color space for both current image, background image and extract the value (V).
- 4: Convert RGB to YCbCr color space for both current image, background image and extract the luminance (Y) and chrominance (Cb) components.
- 5: Find Local Binary Pattern (LBP) texture measure for current image and background image.
- 6: Use Laplacian operator to detect edges in current image and background image.
- 7: Calculate similarity measure for all extracted features components.
- 8: Apply the equation:
 $M(x,y) = SH_3(x,y)*0.2 + SY_1(x,y)*0.2 + SY_2(x,y)*0.3 + S^T(x,y)*0.25 + SR(x,y)*0.05$
- 9: if $M(x,y) < Th$ and edge similarity measure is equal to zero then pixel (x,y) is foreground,
- 10: End.

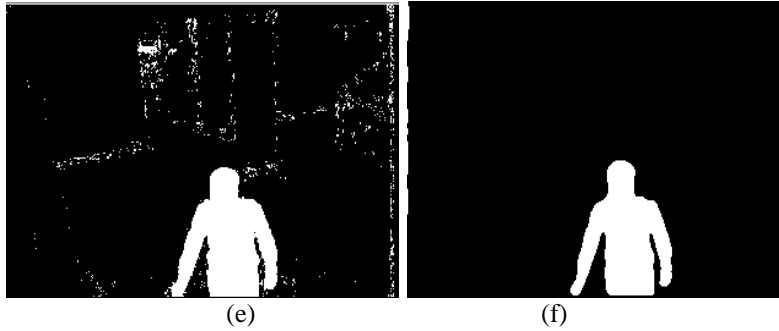
IV. Experiments

To evaluate the performance of the proposed moving object detection method, many image sequences from public domain are adopted as benchmark, which are taken from various types of scenes, to demonstrate the robustness of the proposed method. The foreground objects detected by our method are more accurate than previous methods. The proposed method is more robust against the noise due to light actuation. Cast shadows caused by the foreground objects are also detected well in all test sequences. Figure(1) shows the results for the test sequence.



Figure 1: (a) represent current image. (b) represent background image. (c) represent the moving object.





Figure(2) (a) current image. (b) background image. (c) object detected by YCbCr. (d) object detected by HSV. (e) object detected by RGB. (f) object detected by multi features

Table 1 shows the similarity value obtained for the previous experiments. It is well identified that better results are obtained by the proposed method.

	YCbCr feature	HSV feature	RGB feature	multi features
Similarity value	70 %	25%	60 %	90%

V. Conclusion

In this paper, we have presented a moving object detection method based on color, textures and edges features. Experiments in multimedia and video surveillance datasets show that this method is robust against some changes in background than previous methods such as HSV,YCbCr color feature. The results confirm that optimum results are obtained using multi features than the results obtained by using other color spaces, i.e the RGB, HSV and YCrCb

References

- [1]. S.Swathikiran, P. Sajith Sethu,2. **Human Detection and Tracking System for Automatic Video Surveillance**.*International Journal of Engineering Inventions*,e-ISSN: 2278-7461, p-ISSN: 2319-6491,Volume 2, Issue 10 (June 2013) PP: 58-61
- [2]. **Martínez-Martín, Ester, Pobil, Ángel P. del,**"Robust Motion Detection in Real-Life Scenarios", Series: [SpringerBriefs in Computer Science](#),2012.
- [3]. C.Stauffer, W.Grimson. "**Adaptive background mixture models for real-time tracking**". In Proc. CVPR, 1999.
- [4]. Piccardi, M., "**Background subtraction techniques: a review**", Comput. Vision Group, Univ. of Technol., Sydney, NSW, Australia.
- [5]. **K. Toyama, J. Krumm.** "Wallflower: Principles and practice of background maintenance". **International Conference on Computer Vision, September 1999.**
- [6]. Christof Ridder, Olaf Munkelt, and Harald Kirchner, " **Adaptive Background Estimation and Foreground Detection using Kalman-Filtering**", Bavarian Research Center for Knowledge-Based Systems Orleansstr. 34, D-81667 Munchen, Germany.
- [7]. Zoran Zivkovic, "**Improved Adaptive Gaussian Mixture Model for BackgroundSubtraction**", University of Amsterdam, The Netherlands, In Proc.ICPR,2004.
- [8]. Tinku A., Ajoy K.Ray, (2005), "**Image Processing, Principles and Applications Book**", ISBN-13 978-0-471-71998-4.
- [9]. Rastislav L., Konstantions N.Plataniotis, (2007), "**Color Image Processing Book**", CRC Press, ISBN-13 978-0-8493-9774-5.
- [10]. M. Heikkila, M. Pietikinen. "**A texture-based method for modeling the background and detecting moving objects**". IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI, 28(4):657-662, 2006.
- [11]. Xin Wang," **Laplacian Operator-Based Edge Detectors**" Sch. of Inf. Sci. & Eng., Shandong Univ., Jinan.