An Efficient Foreground Detection Algorithm for Visual Surveillance System

Sivabalakrishnan.M[+] and Dr.D.Manjula[++]

Research scholar , Assistant Professor
College of Engineering , Anna University

Abstract
In this paper we propose an efficient foreground detection algorithm based on a new color space model and morphological filtering. This model uses each pixel's color distortion and brightness distortion to detect the candidate foreground pixels. The color distortion considers the vector's position in color space so that it can assemble the color features effectively. Also shadow elimination process removes the moving shadow. By applying it to background subtraction, we get a comparative complete foreground object. For complex background, the post-processing and the filtering stage helps achieve better results. We have tested the presented algorithm with a variety of videos with various background and results clearly show the efficiency of the algorithm proposed.

Keywords
Forground detection,morphological filtering,shadow elimination,post prcessing

1. Introduction

A fundamental problem in surveillance and monitoring is the extraction of the interesting foreground regions or objects, given an incoming video of the scene. Several foreground-background segmentation algorithms have been proposed to solve this problem, including algorithms based on background subtraction, color distributions, motion, as well as range and stereo. Foreground Detection is often one of the first tasks in computer vision application with stationary camera. It is used in many emerging video applications, such as video surveillance, traffic monitoring, and gesture recognition for human-machine interfaces, to name a few. Foreground Detection is a region-based approach where the objective is to identify parts of the image plane that are significantly different to the background. The remaining of the paper is organized as follows. In section 2 discuss some of the previous work. In section 3 describe background .In section 4 discuss on system overviews. our proposed algorithm is introduced in section 5. experimental results on various video are presented in section 6.The paper is concluded in section 7.

2. Literature Review

Many methods exist for foreground detection, each with different strengths and weaknesses in terms of performance and computational requirements. However, the task becomes difficult when the background contain shadows and moving object. The thresholding method for change detection currently used in background subtraction are mostly adaptive global thresholding method[1].Li [2] proposed a foreground detection and segmentation algorithm which also adopts adaptive global thresholding method. As one crucial module of the Li's algorithm, the thresholding methods result directly determines the performance of the whole algorithm. The global thresholding algorithm used in [2] models each pixel in difference image as spatial distribution of noise, but it ignores the correlation of the neighboring pixels and three channels in RGB color space. In [3], a pixel is marked as foreground if the absolute of the difference between the foreground and the background is greater than a pre-defined threshold. Each pixel is modelled separately by a mixture of K Gaussians in [4, 5, 6]. Pfinder [7] uses a simple scheme, where background pixels are modeled by a single value, and foreground pixels are explicitly modeled by a mean and covariance, which are updated recursively. It requires an empty scene at start-up. In [8, 9, 10], a pixel is marked as foreground based on minimum, maximum, and largest inter-frame absolute difference observable in the background scene. These parameters are initially estimated from the first few seconds of video and are periodically updated for those parts of the scene not containing foreground objects. In [11], foreground detection is based on adaptive thresholding with hysteresis, with spatially varying thresholds. Of all the early methods we present a unique method based on the colorscape which has been tested with a variety of videos.

3. Background

A. RGB Colorspace

The RGB color model is an additive model in which red, green and blue (often used in additive light models) are combined in various ways to reproduce other colors. The name of the model and the abbreviation "RGB" come from the three primary colors, Red, Green and Blue. These three colors should not be confused with the

Manuscript received May 5, 2009
Manuscript revised May 20, 2009

primary pigments of red, blue and yellow, known in the art world as "primary colors". RGB is a convenient color model for computer graphics because the human visual system works in a way that is similar - though not quite identical - to an RGB color space. The most commonly used RGB color spaces are sRGB and Adobe RGB, which has a significantly larger gamut than sRGB. Adobe has recently developed another color space called Adobe Wide Gamut RGB, which is even larger, in detriment of gamut density. RGB spaces are generally specified by defining three primary colors and a white point. In the table below the three primary colors and white points for various RGB spaces are given. The primary colors are specified in terms of their CIE_1931_color_space chromaticity coordinates (x, y).

B. Median Filter

The median filter is also a spatial filter, but it replaces the center value in the window with the median of all the pixel values in the window. The kernel is usually square but can be any shape. Median filtering is a non-linear signal enhancement technique for the smoothing of signals, the suppression of impulse noise, and preserving of edges. In the one-dimensional case it consists of sliding a window of an odd number of elements along the signal, replacing the centre sample by the median of the samples in the window.

$$Y(n) = med(y_{n-k}, y_{n-k}, \ldots\ldots y_{n-1}, x_n, \ldots. x_{n+k}) \qquad (1)$$

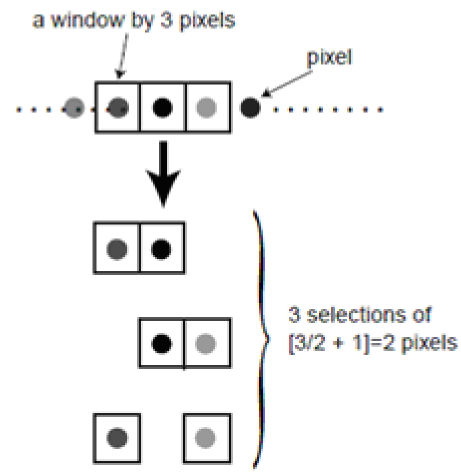
Properties of Median

1. For odd number of points only pre-existing values can be chosen
2. For even number of points average of two middle values are chosen
3. Preserves discontinuity in one dimension (depends on window size).
4. It is a one dimensional operator and so does NOT preserve two dimensional discontinuity

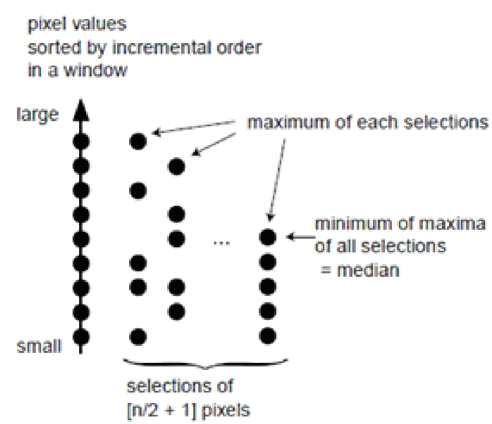
C. Morphological Filter

A morphological filter is performed by passing a structuring element over the image in a convolution-like operation. The output image is determined by the size and shape of the structuring element, as well as the operation chosen. Morphological Filter is a very important tool widely used in suppressing noise, image segmentation, etc.

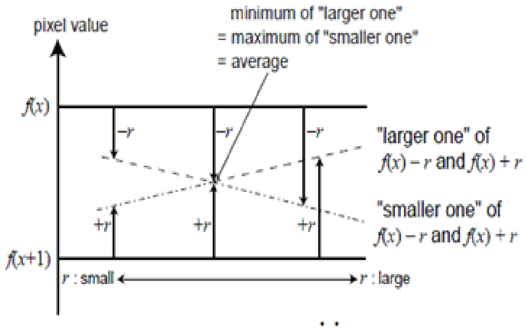
In morphological sense, the term filter is restricted to all image-image transformations that are translation-invariant and increasing. The definition of morphological filter or M-filter sometime requires idem potency in addition to the above requirement of filter. The class of anti-extensive morphological filters is called t-opening, and the class of extensive morphological filters is called t-closing. The ordinary opening and closing are the most basic operations of t-opening and t-closing, respectively. Fig 1 illustrates some basics behind the morphological operations such window selection and calculation of median and average values using the maxima and the minima. Extensively and anti-extensively are not always useful for practical situations. For example, the median filter is not extensive or anti-extensive but it handles "white" objects and "black" backgrounds equivalently. The cascades of opening and closing, open-closing and close-opening, are used for this purpose.



(a)



(b)



(c)

Fig. 1 (a) Selection pixels in a window (b) Illustration of expressing median by minimum of maxima (c) Illustration of expressing average by max of mins or min of maxs

4. System Overview

The System has been designed for a fixed background and the input given will be the current frame and the background frame. Initially the candidate foreground pixels are identified using the color space. Then the shadow regions are identified and eliminated. After which a series of post processing is performed using the median filtering and the morphological filtering.Fig2 shows you the flowchart of all the steps involved in our algorithm from acquired background and foreground frame which is the input to the detected foreground taken as the output.

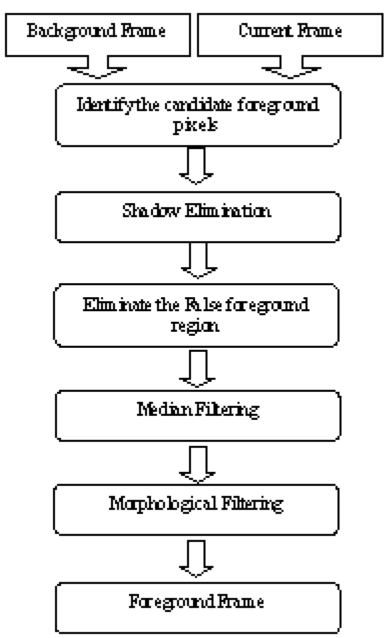


Fig. 2 Flowchart of the Proposed System

5. Proposed Algorithm

While performing in real time the current frame captures initially and the background frame is also loaded along. The input frames will be a color frame and represented in RGB Colorspace. An example of the input frame and the background frame is shown in Fig3.where (a) and (b) are the input frame and the background frame respectively in color format and (c) and (d) are the input and background grayscale frames respectively.

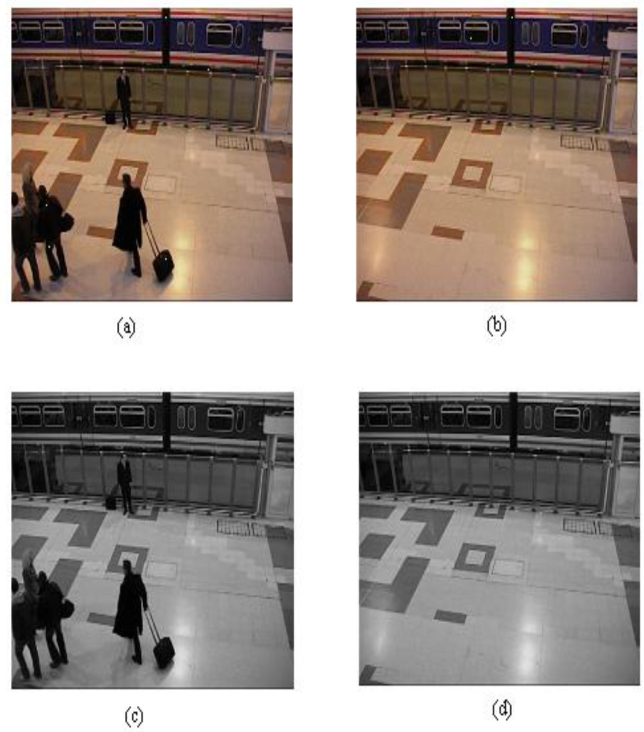


Fig. 3. (a) Input Color Frame (b) Background Color Frame (c) Equivalent Grayscale Current Frame (d) Equivalent Background Grayscale Frame

A. Identification of Candidate Foreground Pixels

Considering the correlations between RGB channels, this paper proposes a thresholding method based on a new color space. Our method originates from the idea of codeword model and color space illustrated in [12]. Kim [12] revealed that most of the pixel value’s variation appears to be in an extended shape along with a axis through the origin point. In another word, variation of illumination is mainly caused by distortion of intensity. Based on this point we propose a thresholding approach based on color distortion and brightness distortion. Fig 4. Illustrates thresholds chosen based on the color space.

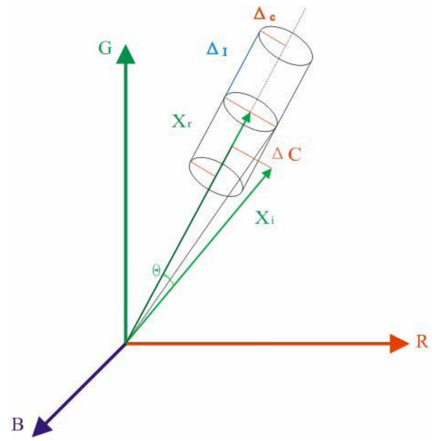


Fig. 4 The thresholding approach based on color space

depicts this thresholding method. When we have an input pixel *xi*= (*Ri*,*Gi*,*Bi*) and the reference (previous frame or reference background)
pixel *xr*= (*Rr*,*Gr*,*Br*), then,

$$||x_i||^2 = R_i^2+G_i^2+B_i^2 \quad (2)$$

$$||X_r||^2 = R_r^2+G_r^2+B_r^2 \quad (3)$$

$$< x_i, X_r>^2 = (R_i R_r + G_i G_r + B_i B_r)^2 \quad (4)$$

And the color distortion ΔC can be calculated by

$$a^2 = ||x_i||^2 \cos^2\theta = \frac{< x_i, X_r>^2}{||X_r||^2} \quad (5)$$

$$colordist(x_i,x_r) = \Delta c = \sqrt{||x_i||^2 - a^2} \quad (6)$$

where θ is the angle between *i x* and *r x* . The brightness distortion ΔI is defined as the absolute value of the difference between the input brightness *Ii* and the reference brightness *Ir*.

$$brightness(I_i, I_r) = \Delta I = | Ii - Ir | = | ||x_i|| - ||X_r|| \quad (7)$$

(a)

(b)

Fig. 5 (a) Color Distortion (b) Brightness Distortion

This paper assumes that brightness distortion detection threshold value *I* Δ is proportionate to the brightness radius *r I* . If color distortion detection threshold *c* Δ 's corresponding distortion angle θ Δ is constant, then color distortion detection threshold value *c* Δ is as well proportional to the brightness radius *r I* , which means that both color and brightness distortion detection threshold varies as brightness varies. Define brightness distortion detection threshold *I* Δ and color distortion detection threshold *c* Δ as:

$$\Delta_I = \alpha . I_r \quad (8)$$

$$\Delta_c = \tan\Delta_\theta \approx \Delta_\theta . I_r \quad (9)$$

where 0<α <1, 0 < Δ θ < 1. For a better sensitivity, we usually choose 0 < Δ θ <α < 0.1. To sum up, define the thresholding formula as follows:

colordist (x i, x r) = ΔC ≤ Δ (10)

brightness (I i, I r) = ΔI ≤ Δ (11)

If one pixel's vector satisfies both of the color and brightness distortion formula above, then this pixel is regarded as a static pixel, otherwise it is a dynamic pixel. If the reference pixel's brightness is relatively small, then color distortion is relatively less significant, so we can ignore the color distortion and only take care of the brightness distortion.

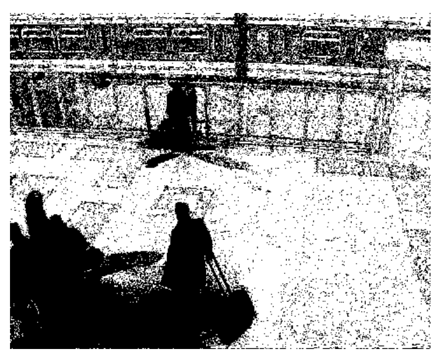
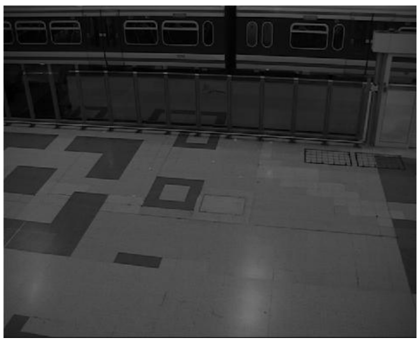


Fig. 6 Output with candidate foreground pixels marked 0's (black)

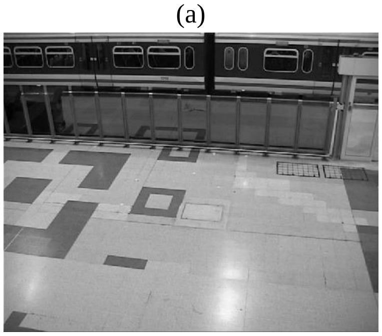
B. Shadow Elimination

To remove the impact of shadow, we multiply the brightness detection threshold a lower limit with a coefficient β bigger than α and thus obtain the brightness variation range to be

[*Ilow*,*Ihigh*]
Where, $Ilow = (1-\beta)*Ir$, $Ihigh = (1+\alpha)*Ir$.



(a)



(b)

Fig. 7 (a) Shadow Elimination Low Threshold (b) Shadow Elimination High Threshold

Then the brightness distortion detection formula would be $Ilow \le Ii \le Ihigh$.

(a)

(b)

Fig. 8 (a) White pixels identified as shadow and unwanted pixels in foreground (b) Output after shadow elimination

C. Post Processing

The difference image is obtained by subtracting the current frame from the background frame. The absolute value is taken as the different image. Set an appropriate threshold. Obtain logical image which satisfies the following conditions

- Pixel should be static pixel
- Different image should be less that t_1

The complement of logical image is subtracted with the candidate foreground image which is obtained from the previous process. Again taking complement of the result and image eliminates few falsely identify static pixel.

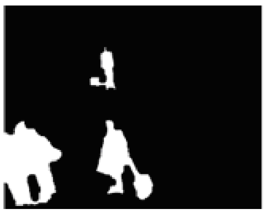
(a)

(b)

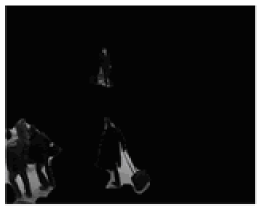
Fig. 9 (a) Difference image for post-processing (b) Output after post-processing

D .Filtering

In some cases spot type noise will be present at the processed stage. Apply median filter with appropriate window size to eliminate if any small noise present. Morphological close operation is performed with an appropriate structure element so that any holes or incomplete regions present in the foreground area will be filled. Perform morphological open operation to remove than small unnecessary objects if present.



(a)



(b)

(c)

Fig. 10 (a) Final binary images with foreground identified as white (1's) (b) Foreground Grayscale Image (c) Foreground RGB Image

6. Experimental Results

To check the efficiency of the algorithm proposed we tested our algorithm on various videos taken from PETS2006 [13] and ten sequences (MPEG files) of test videos and results, including the examples presented in this paper [14] of different complex scenes with moving objects. Tests are executed on 2 Gigahertz P4 PC and 1 Gigabyte memory and can reach 20 frames per second for videos sized 240*384. Figure11 shows four frames taken from different video sequence, in which some parts of the image are illuminated by direct sunlight, while in other parts there is only indirect light incidence. As a consequence, different kinds of shadows are produced (weak and strong shadows). It can be noticed that shadows were effectively detected and removed in both weak and strong shadow regions. Apart from the shadow the background has also been identified and subtracted to get a detected foreground frame . In the experiments, in the Identification of Candidate Foreground Pixels part there are few parameters used $\alpha = 0.1, \beta=0.5$ and $\Delta_{\theta} = 0.12$.In Post processing different threshold is 30.In Median filter window size is 3.In morphological filter close disk radius is 20,opening remove object less than 200 pixels.

Fig. 11 few more results with other videos (a) Input current frame (b) Background Frame (c) Foreground detected frame

7. Conclusion

In this paper, a unique and improved algorithm has been proposed and presented. The approach is based on a new color space model for identifying the foreground pixels. It not only detects the complete foreground object, but also removes the shadows of the moving object.Experiments have been performed, the improved algorithm achieves detection of foreground objects to be higher precision for the simple scene and removes the shadows of the foreground object. For the relatively complex background the post-processing and filtering helps in improving the efficiency of the algorithm. Finally we test our proposed algorithm with a variety of images having various background and lighting conditions. Further work can be done to design and adaptive background model according to this method. By testing the algorithm with a larger variety of video sequences, the parameters can be optimized which will further increase the efficiency.

References

[1] P. Rosin. Thresholding for Change Detection. In Proc. Of IEEE ICCV, pages 274–279, 1998.

[2] L Li, W Huang, IYH Gu and Q Tian. Foreground Object Detection from Videos Containing Complex Background. In Proc. of the 11th ACM Multimedia: 2– 10, 2003.

[3] J. Heikkila and O. Silven: A real-time system for monitoring of cyclists and pedestrians in: Second IEEE Workshop on Visual Surveillance Fort Collins, Colorado (Jun.1999) pp. 74-81.

[4] W. E. L. Grimson, C. Stau_er, R. Romano and L. Lee: Using adaptive tracking to classify and monitor activities in a site in: Computer Vision and Pattern Recognition Santa Barbara, California (Jun. 1998) pp. 1 8.

[5] Y. Ivanov, C. Stau_er, A. Bobick and W. E. L. Grimson: Video surveillance of interactions in: Second IEEE Workshop on Visual Surveillance Fort Collins, Colorado (Jun. 1999) pp. 82-90.

[6] C. Stau_er and W. E. L. Grimson: Adaptive background mixture models for real-time tracking in: Computer Vision and Pattern Recognition Fort Collins, Colorado (Jun. 1999) pp. 246-252.

[7] C. Wren, A. Azabayejani, T. Darrell and A. Pentland: P_nder: Real-time tracking of the human body IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (1997) 780-785.

[8] I. Haritaoglu, R. Cutler, D. Harwood and L. S. Davis: Backpack: Detection of people carrying objects using silhouettes in: International Conference on Computer Vision (1999) pp. 102-107.

[9] I. Haritaoglu, D. Harwood and L. S. Davis: W4: Who? when? where? what? a real time system for detecting and tracking people in: Third Face and Gesture Recognition Conference (Apr. 1998) pp. 222-227.

[10] I. Haritaoglu, D. Harwood and L. S. Davis: Hydra: Multiple people detection and tracking using silhouettes in: Second IEEE Workshop on Visual Surveillance Fort Collins, Colorado (Jun. 1999) pp. 6-13.

[11] T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin and A. Erkan: Framerate omnidirectional surveillance and tracking of camouaged and occluded targets in: Second IEEE Workshop on Visual Surveillance Fort Collins, Colorado (Jun. 1999) pp. 48-55.

[12] T Parag, A Elgammal, A Mittal. A Framework for Feature Selection for Background Subtraction. Proc. IEEE CVPR, 2:1916-1923, 2006

[13] WWW.cvg.rdg.ac.uk/ PETS 2006

[14] http://perception.i2r.a-star.edu.sg

M.Sivabalakrishnan received the Master's degree from P.S.N.A College of Engineering, Anna university, Chennai., in 2004 .He is presently doing his Ph.d at College of Engineering, Anna University, Chennai, India. His main research interests include multimedia and image processing.

Dr.D.Manjula received the Master's and Ph.D degree from College of Engineering, Anna University, Chennai, India in 1987and 2004 respectively. She is currently an Assistant Professor in the Department of Computer Science and Engineering, Anna University, Chennai. Her research interests include NLP, Text Mining, Parallel processing ,Grid computing, Database and multimedia.