

RGB Motion Segmentation using Background Subtraction based on AMF

Yoon-Ho Kim*

ABSTRACT Motion segmentation is a fundamental technique for analysing image sequences of real scenes. A process of identifying moving objects from data is a typical task in many computer vision applications. In this paper, we propose motion segmentation that generally consists from background subtraction and foreground pixel segmentation. The Approximated Median Filter(AMF) was chosen to perform background modeling. Motion segmentation in this paper covers RGB video data.

Key Word : background subtraction, background modeling, approximated median filter (AFM), gray-scale and color motion segmentation

I . Introduction

In order to identify moving objects from video sequence we used motion segmentation procedure in static scenes that includes background subtraction and foreground pixel segmentation.

The process of separating background and object is a common problem in vision system applications like video surveillance, object tracking, object detection and so on.

In the paper a technique of background modelling based on the Approximated Median Filter (AMF) was used. Following motion segmentation step, moving objects are detected.

The organization of the paper is as follows: in section II a description of background research is given. In section III detection of moving objects in a video sequence is proposed. In section IV the result of experiment is presented and in section V test performance is done.

Finally section VI is devoted to conclusion and pointing on some further issues..

II . Introduction

Background subtraction algorithms can be divided into 4 steps: preprocessing, background modeling, background detection and data validation[1][2].

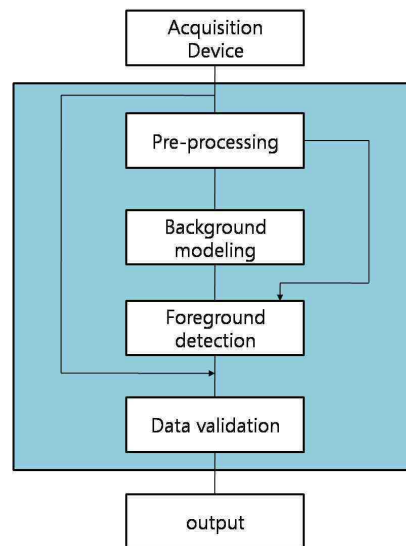


Fig 1. Subtraction algorithm

* Corresponding Author: Dvosopn of convergence computer & media fo Professor Mokwon University (yhkim@mokwon.ac.kr)
 Received : January 02, 2014 Revised : January 13, 2014 Accepted : January 27, 2014

In a pre-processing stage raw input video is changed in a form that can be used and conformed to an algorithm's goal and capability of acquisition device. Image processing tasks as frame, color, noise reduction and spatial operations are commonly used in most visual applications on the first stage. To make a real-time system some size and resolution processed can be done in order to suppress an amount of input information, also conditions of storing data should be considered properly. frame, color, noise reduction and spatial operations are commonly used in most visual applications on the first stage. To make a real-time system some size and resolution processed can be done in order to suppress an amount of input information, also conditions of storing data should be considered properly.

Background modeling is a description of the current background scene. The simplest way to model the background is to acquire a background image which doesn't include any moving objects. In some environments, the background isn't available and can always be changed under critical situations like illumination changes, objects being introduced or removed from the scene. To take into account these problems of robustness and adaptation, many background modeling methods have been developed recently. They can be categorized as follows: Basic Background Modeling, Statistical, Fuzzy and Background Estimation. Also classification can be made in term of recursion, prediction, adaptation, modality and so on[4].

Foreground detection is supposed to check whether the input pixels are foreground or background. The most commonly approach to

find all pixels that match next equation:

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

Another popular foreground detection scheme is to threshold based on the normalized statistics:

$$\left| \frac{I_t(x,y) - B_t(x,y) - \mu_d}{\delta_d} \right| > T_d \quad (2)$$

where μ_d and δ_d are the mean and the standard deviation of $I_t(x,y) - B_t(x,y)$ for all spatial locations (x,y). Most schemes determine the foreground threshold T or T_d experimentally. Of course an ideal threshold should be a function of the spatial location, that it could be scaled for different types regions. For example the threshold must be higher for regions with high contrast and smaller for low contrast regions.

Another approach to introduce spatial variability is to use two thresholds with hysteresis. The basic idea is that first threshold will identify pixels whose absolute differences with the background estimates exceeded a large threshold. Then, foreground regions are grown from strong foreground pixels by including neighboring pixels with absolute differences larger than a smaller threshold. So algorithm will perform foreground detection by using a two-pass foreground detection scheme[3].

Data validation process aims to improve output. Commonly a lot of different techniques can be used to preserve false-positive and false-negative result.

III. Proposed method of moving objects detection in a video sequence

Background modelling (also referred too as background subtraction) was based on Approximated Median filter (AMF). AMF had been previously used for classification system and urban traffic monitoring [5][6]. In this paper we focus on the problem of motion segmentation and though many algorithms have been proposed in the literature we will try to extend the AFM scheme for RGB color space.

The main disadvantage of Median filtering is that the previous frames of video are buffered and background modeled as the median of buffered frames. And though even if Median filtering is robust enough it'll require a lot of memory space. Due to the success of non-recursive median filtering, AMF was devised as efficient recursive approximation of the median filter by McFarlane and Schofield[7].

In order to implement AMF we use next classification. If a pixel value in the current frame has a value larger than the corresponding background pixel, the background pixel is incremented by 1. Otherwise the background is decremented by one. In this way input data comes to a state when half of pixels are greater then the background and half are less than the background. The AMF does good job in separating moving object across video. The pseudocode written below gives a brief introduction to AMF.

```

Algorithm for AMF
1) B and F are sets of RGB describing
background and foreground pixels
Initialization
2) X is a test sequence
3) F is output foreground sequence
4) Foreground detection
   FOR x = 1 to w
     FOR y = 1 to h
       IF all pixels of test sequence for
all color spaces (R,G,B respectively) are
within the distance  $\tau$  of corresponding
background value B(x,y) THEN
         F(x, y) = 0
       ELSE
         F(x, y) = X(x,y)
       ENDIF
5) The background model B={ }
   IF a pixel value in the current
frame has a value larger than the
corresponding background pixel THEN
     B(x, y, z) = B(x,y,z) + 1 //z is
color space index
   ELSE
     B(x,y,z) = B(x,y,z) - 1;
   ENDIF
ENDFOR

```

Foreground pixel are detected by calculating the Euclidean norm at time t:

$$\| I_t(x, y) - B_t(x, y) \| > T_e \quad (3)$$

where I_t is the pixel intensity value, B_t is the background intensity value at time t and T_e is the foreground threshold.

$$I_t = [I_{1,t} \dots I_{c,t}]^T$$

$$B_t = [B_{1,t} \dots B_{c,t}]^T \quad (4)$$

where c is the number of image channles[5].

The foreground threshold T_e determined experimentally.

IV. Experimental Result

Motion segmentation was conducted by

using Matlab Image Acquisition toolbox. Single static 1.3 mega pixel PC "Wezel" webcam was used to acquire video sequence, test was performed on Intel Processor 2.83 HGz, 1.98 Gb of RAM.

We acquired ideo sequence with RGB human in-doors motion. The result is presented in figure 3.



Fig 2. Original RGB frames.



Fig 3. Result image with shadows

As shown above result image (Figure 3) contains some noise and in some frames even trails behind the object or shadows.

Shadows are caused by illumination effects so common remedial is convertingan RGB color space to normalized RGB that should remove the effect of any intensity variations. We normalized RGB as follows:

1) Every frame from video sequence was splitted into three separated red, green and blue component images. For every pixel (5) in

component images eq.6 was calculated.

$$Z = \begin{bmatrix} Z_R \\ Z_G \\ Z_B \end{bmatrix} \quad (5)$$

where Z is a color pixel, ZR, ZG, ZB are all color components respectively

$$Z = \begin{bmatrix} Z_r \\ Z_g \\ Z_b \end{bmatrix} = > \begin{bmatrix} r = \frac{R}{\sqrt{R^2 + G^2 + B^2}} \\ g = \frac{G}{\sqrt{R^2 + G^2 + B^2}} \\ b = \frac{B}{\sqrt{R^2 + G^2 + B^2}} \end{bmatrix} \quad (6)$$

where z is normalized color pixel, color components defined as follows.

Normalized planes were restacked again to form normalized RGB image as presented in Figure 4.

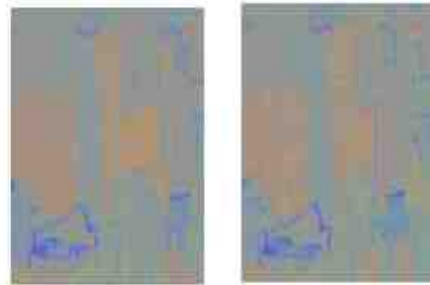


Fig 4. Normalized frames

Figure 5 shows motion segmentation performed on video sequence that was preprocessed as described above, but because some particularly important pixels were removed too and preprocessing step (RGB normalization) distorted color information (black parts of his shorts became blue),

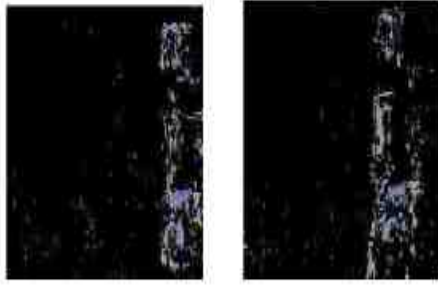


Fig 5. Output image (normalized)

I decided to keep shadows but safe color information as though my application is color based.

Commonly for background subtraction algorithm and color classifier as well, binary mask is used in order to mark all matched components. However I wanted to keep color information for some future work. So I focused on RGB space and implemented RGB normalization on first video sequence to check whether output is better then not normalized RGB motion segmentation. The test didn't satisfy my color criteria (almost all colors were distorted as you can see in Figure 5) and I preferred not normalized output though it is more corrupted with noise (Figure 3) compare with normalized one.

Though the AMF has a good performance it adapts very slowly to a large change in background and sensitives to environmental noise. So low frame rate can also improve the result.

All others motion segmentation details are described in a section II and III.

V. Test performance

In order to have a quantitative evaluation of the performance, we have selected sixteen

frames with moving objects in test sequence, and manually highlighted all the moving objects in them. Manually highlighted pixels are regarded as "ground-truth" information and include only object's movements. The shadows weren't been included since we do not use any shadow suppression scheme in our comparison, all shadows would be cast as noise and false positives.

We used ROC (receiver operating characteristics) curve to quantify algorithm[14].

The true positive rate (also called as hit rate and recall) is estimated as

$$TPR = \frac{TP}{TP + FN} \quad (7)$$

where TPR- true positive rate, TP - true positives, FN- true negatives.

The false positive rate (also called false alarm rate) is

$$FPR = \frac{FP}{FP + TN} \quad (8)$$

where FPR - false positives rate, FP -false positives, TN - true negatives.

Tab. 1. The operating characteristics

cutoff	true positives	false positives
4	0.62	0.4
8	0.65	0.5
12	0.66	0.5
16	0.7	0.5

The measures in Table 1. can be represented graphically as shown below.

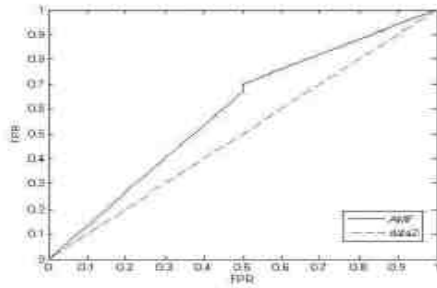


Fig 6. ROC curve of test performance

In Figure 9. TPR is plotted on the Y axis and FPR is plotted on the X axis. Test accuracy is characterized by the plot: the closer curve comes to the 45-degree diagonal of the ROC space, the less accurate the test. Also area under the curve can be a measure of test accuracy.

Test accuracy and other performance characteristics can be improved by using special shadows suppression, applying to some data validation procedures and some pre-processing that not necessary for my future purpose so I was satisfied with test's result.

VI. Conclusion and Future Work

In this paper, we have developed a motion segmentation technique using background subtraction and tested it both for gray-scale and RGB color space. Background subtraction procedure was based on the AMF. Future work will include next aims:

- 1) Elaborating data validation procedures, especially RGB shadow removal techniques
- 2) Also I want to implement some ideas about color detection system Similar that mentioned in reference list

References

- [1] Gary Brodski and Adrian Kaehler, "Learning OpenCV", O'Reilly, 2008.
- [2] Thanarat Horprasert, David Harwood, Larry S. Davis, "A statistical approach for real-time robust background subtraction and shadow detection", 1999.
- [3] Sen-Ching S. Cheung, Chandrika Kamath, "Robust techniques for background subtraction in urban traffic video", proceedings of the SPIE, Vol. 5308, pp. 881-892, 2004.
- [4] T.Bouwman, F. E Baf, B.Vachon, "Background Modeling using Mixture of Gaussians for Foreground Detection - A Survey", Recent Patents on Computer Science, Vol. 1, no. 3, 219-237, 2008.
- [5] Swantje Johnsen and Ashkey Tews, "Real-Time Object Tracking and Classification Using a Static Camera", Proceedings of the IEEE ICRA 2009, Workshop on People Detection and Tracking, Kobe, Japan, May, 2009.
- [6] P.Remagnino et al., "An integrated traffic and pedestrian model-based vision system", Proceedings of BMVC97, Vol. 2, Colchester, 8-11 th September, University of Essex, UK, pp380-389, 1997.
- [7] N.McFarlane and C. Shofield, "Segmentation and tracking of piglets in images", Machine Vision and Applications, Springer, Vol. 8, no.3, 1995.
- [8] Seth Benton, "Background subtraction, Matlab Models", 2008.
- [10] Rastislav Lukac, Konstantinos N. Plataniotis, "Color image processing: methods and applications", Published by CRC Press, ISBN 084939774X,

9780849397745, 2006.

- [11] Turgay Celik, Hasan Demirel, Huseyin Ozkaramanli, Mustafa Uyguroglu, "Fire detection using statistical color model in video sequences", Journal of Visual Communication and Image Representation, Vol. 18, Issue 2, 2007.

Acknowledgements

The authors would like to thank to the PoDIT in Mokwon University. Support for this work was provided in part by Prevention of Disaster with Information Technology RIC.

저자약력

김 윤 호(Yoon-Ho Kim)

중신회원



1992~현재 :
 목원대학교 컴퓨터공학부 교수
 2005~2006 :
 University of Auckland,
 CITR Lab.
 NZ. Research Fellow.
 2008~현재 :
 ISO/TC 223
 KOREA Delegate.
 IEEE Computer science member,
 대한전자공학회,
 한국통신학회 정회원.
 한국모바일학회, 한국항공학회,
 사회안전학회,
 한국정보전자통신기술학회
 중신회원.

<관심분야> 영상처리, 컴퓨터비전, 뉴로퍼지,
 IT 정책, 방재정보통신 등.