An efficient Video Segmentation Algorithm with Real time Adaptive Threshold Technique

Yasira Beevi C P[1] and Dr. S. Natarajan[2]

[1] *MTech. CS&E Student, PES Institute of Technology, Bangalore, India*
[2]*Professor, Dept. of IS&E, PES Institute of Technology, Bangalore, India*
[1]*yasihere@gmail.com*, [2] *natarajan@pes.edu*

Abstract

Automatic video segmentation plays an important role in real-time MPEG-4 encoding systems. This paper presents a video segmentation algorithm for MPEG-4 camera system with change detection, background registration techniques and real time adaptive threshold techniques. This algorithm can give satisfying segmentation results with low computation load. Besides, it has shadow cancellation mode, which can deal with light changing effect and shadow effect. Furthermore, this algorithm also implemented real time adaptive threshold techniques by which the parameters can be decided automatically.

Keywords: *Adaptive threshold, background registration, object extraction, shadow cancellation, video segmentation.*

1. Introduction

The MPEG-4 standard has been taken as the most important standard for multimedia and visual communication and will be applied to many real-time applications, such as video phones, video conference systems and smart camera systems. The most important function of MPEG-4 video part is content-based coding, which can support content-based manipulation and representation of video signal and random access of video objects (VO). Automatic video segmentation is the technique to generate shape information of video objects from video sequences. It is very important in a real-time MPEG-4 camera system with content-based coding scheme, since the shape information is required for shape coding, motion estimation, motion compensation, and texture coding.

Several video segmentation algorithms have been proposed. They can be classified into three types: edge information based video segmentation, image segmentation based video segmentation and change detection based video segmentation. Edge information based algorithms, first apply Canny edge detector to find edge information of each frame and then keep tracking these edges. A morphology motion filter is also applied to find edges belonging to foreground objects. Next, a filling technique can connect edge information to generate final object masks. This method can deal with both still camera and moving camera situations; however, the computation load is very large. Image segmentation based algorithms first apply image segmentation algorithms, such as watershed transform and colour segmentation on each frame to separate a frame into many homogeneous regions. By combining motion information derived with motion estimation, optical flow, or frame difference, regions with motion vectors different from the global motion are merged as foreground regions. These algorithms often can give segmentation results with accurate boundaries, but the computation load for image segmentation and motion information calculation is also high, and the region merging process often has many parameters to set. Both these two kinds of algorithms are too complex to be integrated into a real-time system. Change detection based segmentation algorithms [1], threshold the frame difference to form change detection mask. Then the change detection masks are further processed to generate final object masks. The processing speed is high, but it is often not robust. The segmentation results are suffered from the uncovered background situations; still object situations, light changing, shadow, and noise. The robustness can be promoted by a lot of post-processing algorithms; however, complex post-processing will make the efficiency of less computation lost. The threshold of change detection is very critical and cannot be automatically decided. These reasons make this kind of algorithms not practical for real applications.

In this paper, a fast video segmentation algorithm for MPEG-4 camera systems is proposed. The algorithm has three modes: baseline mode, shadow cancellation mode and adaptive threshold mode. It is based on work using change detection [6]. With background registration technique, this algorithm can deal with uncovered background and still object situations. An efficient post-processing algorithm can improve segmentation results without large computation overhead. Moreover, it has a shadow cancellation mode, in which light changing effect and shadow effect can be suppressed. Furthermore, an adaptive threshold mode is also proposed to decide the threshold automatically.

This paper is organized as follows. Sections 2–4 describe baseline mode, shadow cancellation mode (SC mode) and adaptive threshold mode (AT mode), respectively. The experimental results are shown in Section 5. Finally, Section 6 gives a conclusion of this paper.

2. Baseline mode

Baseline mode is designed for stable situations. That is, the camera is still, and there is no light changing and no shadows. It is based on change detection and background registration technique. Unlike other change detection algorithms, the change detection mask here is not only generated from the frame difference of current frame and previous frame but also from the frame difference between current frame and background frame, which can be produced by background registration technique. Since the background is stationary, it is well-behaved and more reliable than previous frame. Besides, still objects and uncovered background problems can be easily solved under this scheme. The block diagram of baseline mode is shown in Figure 1. There are five parts in baseline mode: *Frame Difference*, *Background Registration*, *Background Difference*, *Object Detection*, and *Post processing*.

Frame Difference

In *Frame Difference*, the frame difference between current frame and previous frame, which is stored in *Frame Buffer*, is calculated and thresholded. It can be presented as

$$FD(x, y, t) = | I(x, y, t) - I(x, y, t-1) | \quad \textbf{(1)}$$

$$FDM(x, y, t) = \begin{cases} 1 & \text{if } FD \geq Th \\ 0 & \text{if } FD < Th \end{cases} \quad \textbf{(2)}$$

where I is frame data, FD is frame difference, and FDM is Frame Difference Mask. Pixels belonging to FDM are moving pixels. Note that there is a parameter *Th* needed to be set in advance. The method to decide the optimal *Th* is shown in Section 4.

Background Registration can extract background information from video sequences. According to *FDM*, pixels not moving for a long time are considered as reliable background pixels. The procedure of *Background Registration* can be shown as

$$SI(x, y, t) = \begin{cases} SI(x, y, t-1)+1, & \text{if } FDM=0 \\ 0, & \text{if } FDM=1 \end{cases} \quad \textbf{(3)}$$

$$BG(x, y, t) = \begin{cases} I(x, y, t), & \text{if } SI(x, y, t)=Fth \\ BG(x, y, t-1), & \text{else} \end{cases} \quad \textbf{(4)}$$

$$BI(x, y, t) = \begin{cases} 1, & \text{if } SI(x, y, t)=Fth \\ BI(x, y, t-1), & \text{else} \end{cases} \quad \textbf{(5)}$$

where SI is Stationary Index, BI is Background Indicator, and BG is the background information. The initial values of SI, BI and BG are all set to “0.” Stationary Index records the possibility if a pixel is in background region. If a pixel is “not moving” for many consecutive frames, the possibility should be high, which is the main concept of (3). When the possibility is high enough, the current pixel information of the position is registered into the background buffer *BG* , which is shown as (4). Besides, *Background Indicator* is used to indicate whether the background information of current position exists or not, which is shown as (5). Note that (3)–(5) also imply that a background updating ability is also included in *Background Registration*, that is, if background changes, new background information will be updated in to the background buffer. The parameter *Fth* needed to set in advance, which indicates the number of consecutive frames pixel is not moving.

Background difference mask: After Background Difference, another change detection mask named Background Difference Mask (*FDM*) is generated the operations of Background Difference, can be shown as.

$$BD(x, y, t) = | I(x, y, t) - BG(x, y, t-1) | \quad \textbf{(6)}$$

$$BDM(x, y, t) = \begin{cases} 1, & \text{if } BD \geq Th \\ 0, & \text{if } BD < Th \end{cases} \quad \textbf{(7)}$$

where BD is background difference, BG is background frame, and BDM is Background Difference Mask, respectively.

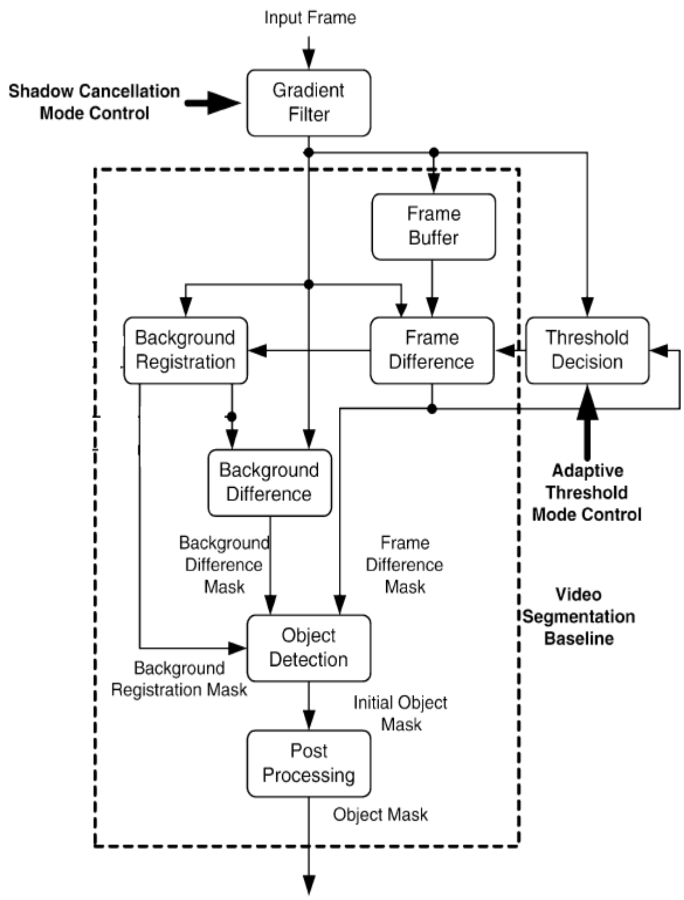

Figure. 1. Block diagram of video segmentation Algorithm Using change detection

Table 1. Situations of object detection

Situation	FDM	BDM	BI	IOM
Stationary	0	-	0	0
Moving	1	-	0	1
Background	0	0	1	0
Moving Object	1	1	1	1
Still Object	0	1	1	1
Uncovered Background	1	0	1	0

Object Detection

Both of FDM and BDM are input into Object Detection to produce Initial Object Mask (IOM_k). The procedure of Object Detection can be presented as the following equation.

$$IOM(x, y, t) = \begin{cases} BDM(x, y, t), & if\ BI(x, y, t) = 1 \\ FDM(x. y, t), & else \end{cases} \quad \textbf{(8)}$$

This process can deal with the six situations shown in Table I, where "-" means "not available." Note that the last two situations are easily misclassified by other change detection based segmentation algorithms, where *BDM* information is not available. In other algorithms, the still objects are often taken as background objects because they are not included in *FDM*, and the uncovered background is often taken as foreground object because it is included in *FDM*. Both of these two situations need complex post-processing algorithms to compensate the mis-classification, which are not needed in the proposed algorithm.

Post processing

The *Initial Object Mask (IOM)* generated by *Object Detection* has some noise regions because of irregular object motion and camera noise. Also, the boundary may not be very smooth. Therefore, there are two parts in *Post processing*: noise region elimination and boundary smoothing.

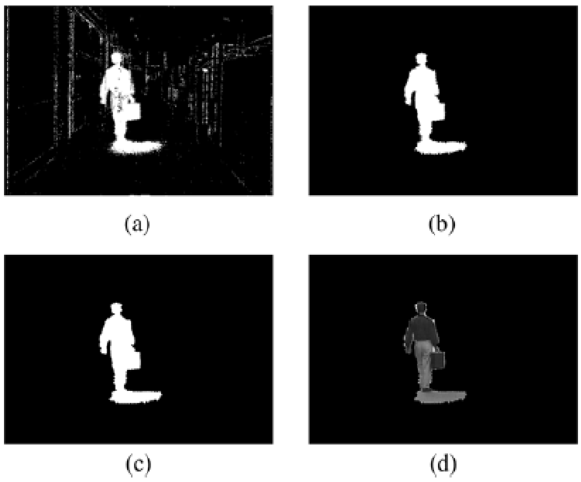

Figure. 2. Illustration of post-processing. (a) Initial object mask; (b) after noise elimination; (c) after morphological closing operation; (d) generated VOP.

The connected component algorithm [8] can mark each connected region with a special label. Then we can filter these regions by their area. If the area of a region is small, it may be a noise region and can be eliminated. Background regions, which are indicated by "0" in *IOM*, are first filtered, that is, background regions with small area are eliminated. This process eliminates holes in the change detection mask, which often occur especially when the texture of foreground objects is insignificant. Then foreground regions, which are indicated by "1" in *IOM*, are then filtered. This process removes noise regions. Next, the morphological close–open operations are applied to smooth the boundary of object mask. In addition, Stationary Index is further revised with *IOM* by

$$SI(x,y,t) = 0, \text{ if } IOM(x,y,t) = 1 \tag{9}$$

This process can avoid still objects to be registered into the background buffer. Figure 2 shows the effect of *Postprocessing*. Figure 2(a) is, where the white parts are those indicated by "1" in, and the black parts are those indicated by "0" in. After noise region elimination, the mask can be improved as shown in Figure 2(b). After boundary smoothing, the improved mask is shown in Figure 2(c). Finally, the generated VOP is shown in Figure 2(d).

3. Shadow Cancellation Mode

Conventional change detection algorithms usually cannot give acceptable segmentation results when light changes, or shadows exist. Shadow regions are always falsely detected as parts of foreground regions, and the whole frame will be regarded as foreground object if light changes. In these situations, shadow cancellation mode (SC mode) [8] is preferred.

3.1 Proposed Shadow Cancellation Algorithm:

Based on the analysis, a shadow cancellation algorithm is proposed and is employed in shadow cancellation mode (SC mode). There are two different parts in SC mode: *Gradient Filter* and *Post-processing*.

1) Gradient Filter: The morphological gradient is chosen because of its simple operations. It can be described by given equation,

$$GRA = (I \oplus B) - (I \ominus B) \tag{10}$$

Where I is the original image, B is the structuring element of morphological operations, $\oplus$ is morphological dilation operation, $\ominus$ is morphological erosion operation, and GRA is the gradient image. After the gradient operation, the shadow effect will be reduced, and the motion information is still kept.

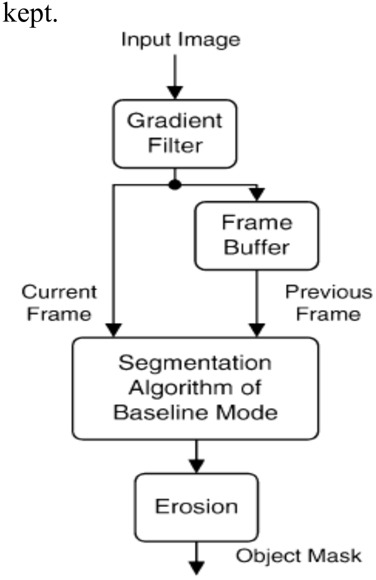

Figure. 3. Block diagram of shadow cancellation mode

2) Post-processing: After morphological gradient operation, the edges are thickened. The edge thickening effect is illustrated in Figure 4. In Fig. 4, compared with ideal edge image, the edge of gradient image is thickened, which may make the segmentation results not accurate at boundaries. To eliminate this effect, a morphological erosion operation should be added at the end of *Post-Processing*.

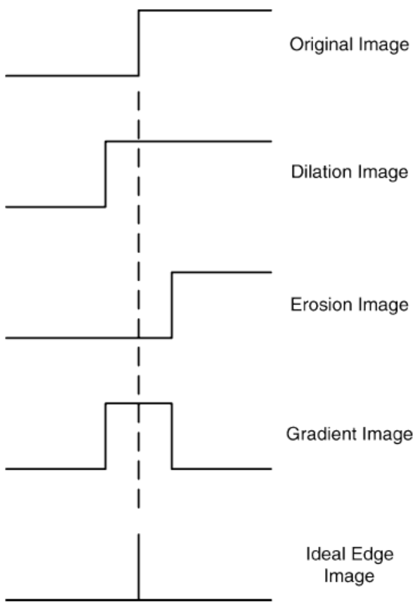

Figure. 4. Edge thickening effect of morphological gradient filter.

4. Adaptive Threshold Mode

The threshold is a very critical parameter for change detection based algorithms. If the optimal threshold cannot be decided automatically, these kinds of video segmentation algorithms are hardly used in real applications. Therefore, the automatic threshold decision is very important in our video segmentation system.

Many thresholding methods have been proposed in literatures; however, few of them are specific to change detection. Thresholding methods can be classified into gray-level distribution based [10, 11] and spatial properties based [7, 13]. After evaluating many thresholding methods for change detection, Rosin *et. al.* [7, 12] recommends three thresholding methods: Euler-number [13], Poisson-noise modelling [7], and Kapur method [10]. The Euler-number thresholding is based on the assumption that the number of regions of change in a difference image will tend to be stable over a wide range of threshold values. The Poisson-noise model thresholding is based on the assumption that observations (number of pixels over a specific threshold) in an image usually follow a Poisson distribution. The Kapur thresholding is entropy based. These three thresholding methods perform well for change detection. However, the loads of computation of the Euler number thresholding and Poisson-noise modeling thresholding are high and not suitable in real-time. In addition, the Poisson-noise modeling thresholding is sensitive to its parameter, the window size. The Euler method tends to under-threshold some images. The Kapur thresholding is sensitive to the noise level and under-threshold a difference image.

The proposed non-parametric algorithm [2] computes a threshold of each block of an image adaptively based on the scatter of regions of change (ROC) and averages all thresholds for image blocks to obtain the global threshold. First, the output D_n of change detection at time instant n is divided into K equal-sized blocks. Then a ROC scatter estimation algorithm is applied, where each image block W_k, k = {1, 2, …. ,K}, is marked either as containing ROC, denoted W_k^r, or not containing ROC, denoted $W_k^{\boldsymbol{b}}$. The threshold T_k^b of a $W_k^{\boldsymbol{b}}$ is computed by a noise statistical-testing algorithm. The threshold T_k^r of a W_k^r is computed by a noise-robust thresholding method. That is, the threshold T_k of a W_k in D_n is defined as

$$T_k = \begin{cases} T_k^r & \text{of} \quad W_k^r \\ T_k^b & \text{of} \quad W_k^b. \end{cases} \tag{11}$$

Finally, the global threshold T_n of a difference image D_n is

$$T_n = \frac{1}{K}\sum_{k=1}^{K} T_k \tag{12}$$

Since size and velocity of objects, noise, local changes in videos may affect the histogram of a W_k, the first moment of histogram is used to estimate the scatter of ROC making the estimation adaptive to these characteristics. This also account for noise and local changes.

The ROC in D_n are, in general, scattered over the K image blocks. Let *i* be a pixel in D_n that varies between 0 and 255. *i* is high in ROC which is caused by strong changes such as motion or significant illumination changes and is low in non ROC which is caused by slight changes such as noise or slight illumination changes. We use the first moment, m_k, of the histogram of each image block W_k as a measure for determining if an image block contains ROC. If m_k of W_k is greater than a threshold T_m, the image block is regarded as a block containing ROC, and marked as W_k^r, otherwise, it is marked as W_k^b, i.e.,

$$W_k = \begin{cases} W_k^b & : \quad m_k \leq T_m \\ W_k^r & : \quad m_k > T_m \end{cases} \tag{13}$$

To find T_m, we first compute the m_k of each block and then descending sorts the m_k values. A straight line between the first bin and the last filled bin is then drawn. T_m is selected to maximize the perpendicular distance between the line and the sorted first moment curve. Figure 5 shows an example of adaptive thresholding, where we used relative m_k

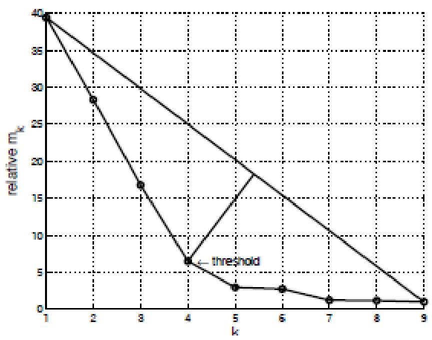

Figure. 5. An example of adaptive m_k thresholding (K = 9).

4.1 Threshold in Non ROC

To account for video noise, we model the noise as a Gaussian distribution with zero mean $N(0,\sigma_v^2)$ with σ_v^2 as the noise the noise variance. In such a case, the noise in the difference image obtained by image differencing followed by taking absolute value can be modelled as $2N(0,2\sigma_v^2)$ [7]. Since a linear function of a Gaussian random variable is also a Gaussian random variable [9]. Let $\sigma_D = \sqrt{2}\sigma_v$, the noise of a difference image can be modeled as a new Gaussian distribution $N(\mu_D, \sigma_D^2)$, where μ_D and σ_D^2 are the mean and variance of the noise in a difference image, respectively, and μ_D is usually zero. For image blocks without ROC, non-zero pixels are usually caused by noise. So based on the noise model above, for each W_k^b we can find a T_k^b, where the pixels in W_k^b are highly-probable lower than T_k^b. The gray-level T_k^b is a reasonable threshold for W_k^b, i.e., the probability that the pixels in an image block W_k^b are lower than T_k^b should satisfy

$$P[X \leq T_k^b] > p_h \tag{14}$$

where X = $W_k^b(i, j)$ is the gray-level of a pixel at location (i, j) in W_k^b and p_h is a high probability value. Then

$$P[X \leq T_k^b] = \int_{-\infty}^{T_k^b} \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}}\, dx \tag{15}$$

where μ_k and σ_k^2 are the mean and variance of gray-level in image block W_k^b, respectively. Let t = $x-\mu_k/\sigma_k$, we get

$$P[X \leq T_k^b] = \Phi\left(\frac{T_k^b - \mu_k}{\sigma_k}\right) > p_h \tag{16}$$

where Φ(.) is the cumulative distribution function (cdf) of standard normal distribution. To estimate p_h, using the standard normal distribution table,we have Φ(2.8) = 0.9975.Because the cdf of a random variable is a non-decreasing function, P[X ≤ T_k^b] > 0.9975 means $(T_k^b-\mu_k)/\sigma_k > 2.8$,i.e., $T_k^b > \mu_k + 2.8\sigma_k$. The threshold of an image block without ROC is, therefore defined as

$$T_k^b = \mu_k + c \cdot \sigma_k \tag{17}$$

where c is a varying coefficient to improve the robustness to noise, illumination changes, or background movement. The coefficient c is determined adaptively based on the min-max ratio of m_k, $r_m = m_{min}/m_{max}$, where m_{min} and m_{max} are minimum and maximum of all m_k in D_n. Because noise, illumination changes and background movement in an image can significantly increase the probability of the occurrence of a high gray-level in W_k^b, m_{min} for such an image increases greatly. This leads the ratio r_m increasing too. So r_m is a good measure to adaptively estimate c as follows

$$\begin{aligned} c_r &= \frac{1}{r_m} e^{r_m} - \alpha \\ c &= \begin{cases} c_{max} & : \; c_r > c_{max} \\ c_r & : \; \text{otherwise} \end{cases} \end{aligned} \tag{18}$$

where c_{max} is the maximum value of c determined by Eq.14, and α is a constant experimentally set to 0.9.

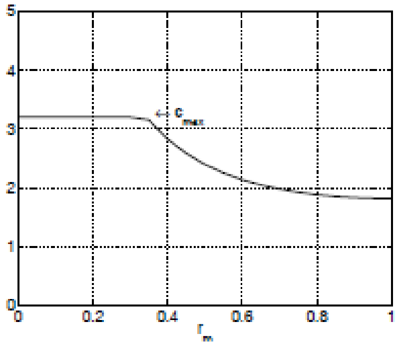

Figure. 6. Adjusting rate function of c vs. rm

4.2 Thresholding in ROC

For each image block W_k^r containing ROC, we proposed the local-change adaptive thresholding. First, the histogram of each W_k^r is computed and divided into L equal partitions, and the most frequent gray-level g_{fl}, l = {1, 2, …L}, in each histogram partition is fixed .Then the average gray level μ_k of W_k^r is computed and the threshold T_k^r of W_k^r is obtained by averaging the sum of all g_{fl} and μ_k, i.e.

$$T_k^r = \frac{\sum_{l=1}^{L} g_{f_l} + \mu_k}{L+1} \tag{19}$$

5. Experimental Results

The performance of the proposed video segmentation algorithm is tested with many video sequences. Both the objective and subjective quality evaluations are applied on our algorithm.

5.1. Objective Evaluation

The error rate of the object mask is adopted to present the effectiveness of our algorithm. The error rate is defined as the following equation:

$$\text{Error rate} = \text{Error Pixel Count} / \text{Frame Size} \tag{20}$$

where the error pixel count is the number of pixels from which the obtained object mask is different from the reference alpha plane. Figure 7 shows the error rate of the Weather sequence. The format of the test sequence is 360 × 243 at 30 fps. The error rate is lower than 0.8% most of the time, with an exception that a sudden rise of error rate start at frame #180. This behaviour corresponds to a large motion of the object with a large area of newly uncovered background. After the background information is stored in the background region, the error rate drops to its normal value.

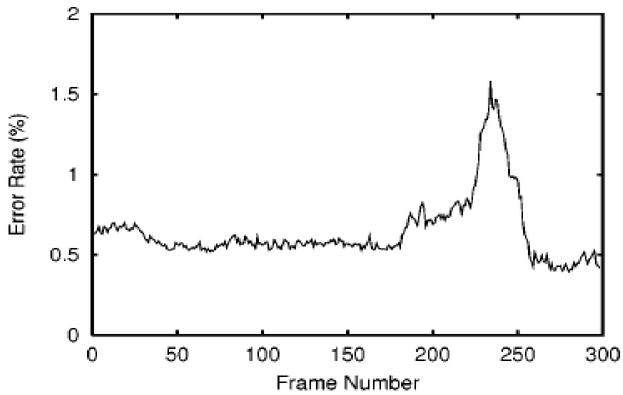

Figure. 7. Error rate in each frame of the Weather sequence (QCIF)

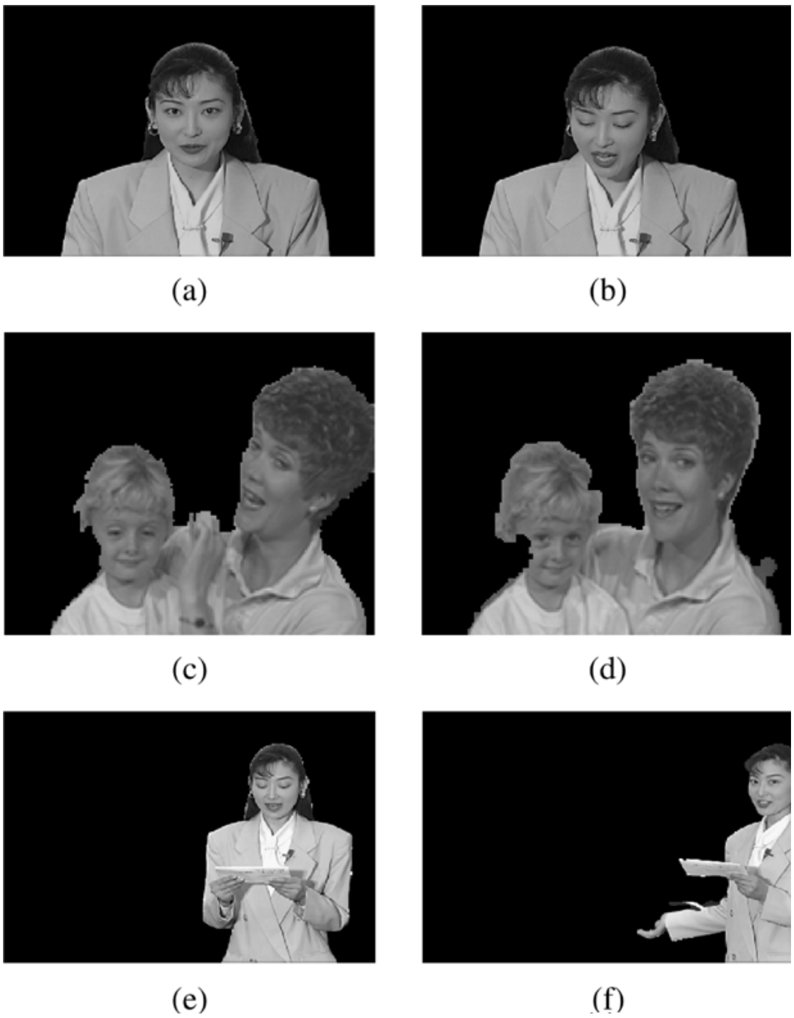

Figure. 8. Segmentation results of baseline mode. (a) *Akiyo* #50; (b) *Akiyo* #100; (c) *Mother and Daughter* #50;(d) *Mother and Daughter* #100; (e)*Weather* #50;(f) *Weather* #100.

5.2. Subjective Evaluation

A fair evaluation method named "fix frame number test" is used here. That is, the segmentation results of fixed frames of each sequence are picked, rather than chosen by the algorithm developers themselves. The quality of segmentation results is then evaluated subjectively. In these experiments, frame 50 and frame 100 are chosen in every sequence.

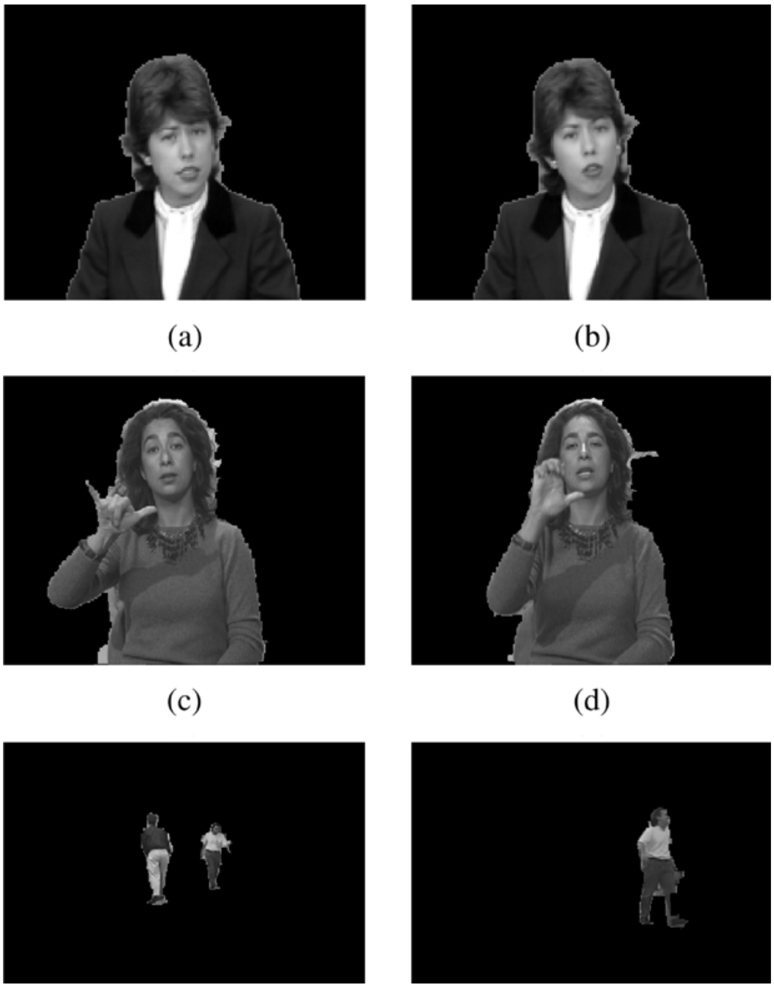

Figure. 9. Segmentation results of SC mode. (a) *Claire* #50; (b) *Claire* #100; (c) *Silent Voice* #50; (d) *Silent Voice* #100; (e) *Hall Monitor* #50; (f) *Hall Monitor* #100.

Baseline Mode: The segmentation results of sequence *Akiyo*, *Mother and Daughter*, and *Weather* are shown in Figure 8. These sequences are not influenced by shadow and light changing effects. The segmentation results of sequence *Akiyo* and *Mother and Daughter* are shown in Figure 8(a)–(d) respectively, where the still object problem can be correctly solved. In Figure 8(e) and (f), the sequence *Weather*, in which the foreground object has large motion, can also be correctly segmented. Note that in Figure 9(f), some segmentation errors occur near the hand of the reporter. That is because the background information of that region is not yet available.

Shadow Cancellation Mode: The experimental results of shadow cancellation mode are presented in Figures. 9. The sequence *Claire*, which is influenced by light changing effect, can be correctly manipulated in SC mode, as shown in Figure 9(a) and (b). In Figure, 9(c) and (d), the segmentation results of sequence *Silent Voice* are shown. It shows that the shadow effect in *Silent Voice* can be reduced, and good object mask can be generated. The segmentation results of sequence *Hall Monitor* are presented in Figure 9(e) and (f). Both of the light changing and shadow effects occur in this sequence. The experimental results show that this sequence can also be correctly segmented in SC mode. The shadows of the foreground objects can be cancelled.

Adaptive Threshold Mode: The experimental results of AT mode are shown in all the figures mentioned above because the threshold in all experiments is decided by the automatic threshold decision algorithm. The segmentation results show the proposed threshold decision algorithm is suitable for change detection and background registration based video segmentation algorithms.

6. Conclusion

This paper proposes, background registration and change detection based video segmentation algorithm with shadow cancellation to analyzes light changing and shadow effects in indoor environments and a real time adaptive threshold techniques to decide the parameters automatically, This algorithm can generate segmentation results with low computation complexity and high efficiency compare to other change detection based video segmentation algorithm.

There are still limitations in the proposed segmentation system. The shadow cancellation cannot deal with parallel and strong light sources and may cause errors when the texture of background is significant. Furthermore, the decision to turn on and off each mode of the proposed algorithm is not automatic. Finally, this algorithm is designed for moving objects segmentation; therefore, for stable and accurate results, the foreground object should not be still for a long time.

References

[1] R.J.Radke, S.Andra,O.Al-Kofahi, and B. Roysam "Image Change Detection Algorithms: A Systematic Survey", IEEE Trans. Image Processing, Vol.14, No. 3, pp.294–303,March. 2005.

[2] ChangSu and AishyAmer "A Real Time Adaptive Thresholding for video change detection" .IEEE International Conference on Image Processing (ICIP), pp 157-160, Oct-2006

[3] S. Y. Chien, S. Y. Ma, and L. G. Chen "Efficient Moving Object Segmentation Algorithm Using Background Registration Technique", IEEE Trans. on circuits and system for video technology, Vol. 12, No. 7, pp. 577–586, July 2002.

[4] Shao-Yi Chien, Yu-Wen Huang, Bing-Yu Hsieh, Shyh-Yih Ma, and Liang-Gee Chen,"Fast Video Segmentation Algorithm with Shadow Cancellation, Global Motion Compensation, and Adaptive Threshold Techniques," *IEEE Trans. on Circuits and System for Video Technol.*, Vol. 6, pp. 732-748, No. 5, October. 2004.

[5] Rafael C.Gonzalez, Richard E. Woods and Steven L.Eddins," Digital Image Processing Using Matlab", Pearson Education, 2nd Edition 2006.

[6] Shao-Yi Chien, Shyh-Yih Ma, and Lieng-Gee Chen, "An efficient video segmentation algorithm for real-time MPEG-4 camera system," in *Proc. Visual Communication and Image Processing*, pp. 1087–1098,2000.(SPIE, Vol 4067, 1067).
[7]P.L.Rosin,"Thresholding for change detection," *Computer Vision and Image Understanding*, vol. 86, pp. 79–95, 2002.
[8] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley, 1992.
[9] A. Leon-Garcia, *Probability and Random Processes For Electrical Engineering*, Addison-Wesley Publishing Company Inc., second edition, 1994.
[10] J. Kapur, P. Sahoo, and A. Wong, "A new method for gray-level picture thresholding using the entropy of histogram," *Computer Vision, Graphics Image Process*, vol. 29, no. 3, pp. 273–285, 1985.
[11] N. Otsu, "A threshold selection method from gray-level histogram," *IEEE Trans. System, Man and Cybernetics*, vol. 19, pp. 62–66, 1979.
[12] P. L. Rosin and E. Ioannidis, "Evaluation of global image thresholding for change detection," *Pattern Recognition Letters*, vol. 24, pp. 2345–2356, 2003.
[13] P. L. Rosin and T. Ellis, "Image difference threshold strategies and shadow detection," *In Proc. British Machine Vision Conference*, pp. 347–356, 1995.

Authors

Yasira Beevi C P was born in Kerala, India in 1981.She received her Bachelor of Technology in Electronics and Communication Engineering from University of Kannur in 2003.She has started her career as Lecturer in Department of Electronics and Communication Engineering and served for 3 years. Currently, she is pursuing Master of Technology in Computer Science and Engineering from PES Institute of Technology, Bangalore,India. Her research interest includes Image processing, Video Segmentation and moving object detection and tracking.

Dr. S. Natarajan has initially worked in Defence Research and Development Laboratory (DRDL) for five years in the area of software development in defence missions. He then worked for twenty eight years in National Remote Sensing Agency (NRSA) in the areas pertaining to Digital Image Processing and Geographical Information Systems for several remote sensing missions like IRS-1A, IRS-1B, IRS-1C, IKONOS and LANDSAT. He was the Deputy Project Director of Large Scale Mapping (LSM) of Department of Space and was responsible for the generation of maps for utility mapping within the country, done for the first time in India. He successfully implemented the base map generation for Class I towns of Andhra Pradesh under DFID funded Project for Andhra Pradesh Urban Services for the Poor (APUSP). Using aerial data of Bangalore, the digital database for Bangalore covering 99 wards for Bangalore Mahanagar Palike (BMP) has been created under his guidance. He had undergone training in Digital Image Processing and allied fields at DLR, Germany for one year during 1978-1979. He had also undergone training in Automated Cartography for three months at IFAG, Germany during 1980. He had taught courses in Object Oriented Technology, GIS, Cartography, Artificial Intelligence and Neural Networks for M.Tech students at JNTU for during 1994 – 2003. He had visited Singapore, Philippines, Germany and United States of America during his tenure at NRSA He has presented about ten papers in international conferences and published three Journal articles. Dr Natarajan has published about thirty papers in National/ International Conferences and Journals. After retiring from NRSA, he took up full time teaching assignment as Professor and is serving for the past five years.