A REAL-TIME VIDEO OBJECT SEGMENTATION ALGORITHM BASED ON CHANGE DETECTION AND BACKGROUND UPDATING

Thou-Ho (Chao-Ho) Chen, Yung-Chuen Chiou, Hung-Shiuan Liau and Zhi-Xian Chen

Department of Electronic Engineering, National Kaohsiung University of Applied Sciences
Kaohsiung, Taiwan 807, R.O.C.

{thouho, a092315106, 1093320124, 1093320119}@cc.kuas.edu.tw

ABSTRACT

This paper proposes an efficient automatic video object segmentation algorithm by combining change detection with background updating. First, motion information is obtained from difference between previous frame and current frame instead of complicated motion estimation. Second, background subtraction is introduced to solve the problems of still object and uncovered background an ill-posed in conventional method. Moreover, the effect of shadow and residual background can be suppressed by a novel hierarchical segmentation in boundary refinement step. Finally, experimental results show that the proposed algorithm can maintain a high accuracy upper 95 percent with capturing situations of the fixed camera.

1. INTRODUCTION

Conventional video-coding standards, such as H.261/H.263 and MPEG-1/MPEG-2 that employ the frame-by-frame coding policy, can’t provide high-level feature of video contents. The MPEG-4 coding standard [1] has introduced the concept of a video-object plane (VOP) as the basic coding element for supporting visual multimedia communication and will be applied to many multimedia content descriptions [2], and intelligent signal processing, such as, digital video surveillance system and video conference. Each VOP contains the shape and texture information of semantically meaningful object in the scene. In order to encode video objects using object-by-object in video sequences rather than frame-by-frame and achieve the content-based manipulation for video content, automatic video segmentation will play an important role of deriving VOP from video sequences in MPEG-4 video part. For many practicable multimedia applications of real-time demand, a fast and efficient video segmentation algorithm will be very important.

Nowadays, video object segmentation algorithm can be coarsely classified into two types, semi-automatic or automatic. In semi-automatic algorithm [3][4], user must first define a high level semantic object of interest to be segmented and detect the object’s boundary in keyframe by way of manual. After then the extracted object of interest region is used to segment this object occurred in the video sequences. The user interaction based algorithm can give better segmentation results than automatic algorithm, but it may be unsuitable in real-time applications due to the fact that an interest object needs to be specified by the user before the algorithm begins to execute.

For this reason, automatic video object segmentation algorithm have been developed and they are roughly classified into three types: the edge-feature based segmentation, spatial-temporal based segmentation, change detection based segmentation. The edge-feature based segmentation algorithm [5][6] utilize Canny edge detector to find edge information of each frame, and it can obtain correct segmentation object for stable moving-object, but this approach must to acquire a absolute background from video sequence and suffering a computation-intensive processing. In [7], based on the spatial-temporal segmentation approach, a watershed transform is used to separate a frame into many homogeneous regions and then each region is checked with the motion information. It brings about over-segmentation due to sensitive to noise, but it can give a good result of segmentation with high accurate boundaries. This problem can be solved by smoothing texture of image content, but it will make the performance of algorithm reduced. In the change detection based segmentation algorithm [8][9][10], the position and shape of the moving object can be detected by thresholding the difference of two consecutive frames and them followed by a boundary fine-turning process with the spatial or temporal information to improve the result of segmentation.

Basically speaking, for the above three approaches, the segmentation algorithm based on the change detection technology will be more efficient than the other two types of segmentation algorithm which may be too computation-intensive to achieve the real-time purpose. The moving object obtained always suffers from the uncovered background situations, still object situation, light changing, shadow, residue background
problem and noise due to the motion information is obtained by frame difference. To overcome these problems and enhance the performance, this paper proposes an efficient video segmentation algorithm by using the change detection and background updating technique. The proposed algorithm can be shown in figure 1. The Gaussian smooth is focused on reducing the noise effect of input frame. Then, the change detection, object region detection, and background updating are combined to extract a rough moving object. Finally, the accuracy of segmented object will be improved through an efficient boundary refinement with a appropriate amount of computations.

2. THE PROPOSED SEGMENTATION ALGORITHM

The block diagram of the proposed video segmentation algorithm is shown in figure 1, which is dedicated to separate the moving-object regions from other parts of the scene by using of motion information. We construct and maintain an updating background reference from the previous still object mask, which is used due to a static object or a region where moving object shift out their position are must to be renewed. Therefore, our approach, is aimed at the stationary object of a scene since the information of stationary object of a scene is more stable to make the constructed background to reliable and conform human visual perception. Because the background change owing to camera motion can be compensated by global motion estimation and compensation [8], the input sequence to our algorithm will be assumed to have been correctly compensated and thus the background is stationary. The details of each module in the proposed algorithm, as shown in figure 1, will be discussed in the following subsections.

2.1. Change detection

Frame difference technique [12][13] is often used in change detection based segmentation algorithm, that gives difference between two successive input frames. After, the pixel on difference frame will be classified to change parts or unchanged parts by considering the value of pixel. Figure 2 shows the proposed block diagram of change detection, which adopts histogram-based statistic method. First, the parameter of background model is estimated from difference frame. Second, the pixel will be classified according to statistic parameter obtained from previous phase. The detail of this process will be described in the following subsection.

2.1.1. Histogram analysis

The histogram is constructed from difference frame, which can provide information of the gray-level distribution to analyze the characteristic of the difference frame. Let $\text{His}(p')$ be the maximum gray of bin at $p'$ position of the histogram. To extract more informations about the background, both $\text{His}(p')$ and $p'$ need to be obtained and them such two values will be used in the following estimation of the background model.

2.1.2. Parameters estimation

Generally, the model of background region of the difference frame can be regarded as Gaussian distribution due to the noise effect existed between inter-frames. Hence, the mean and standard deviation for the background model within a window, denoted as $\mu_b$ and $\sigma_b$, respectively, need to be acquired. Firstly, the value of mean $\mu_b$ and standard deviation $\sigma_b$ will be calculated at the site of $p'$ with window size $N$. Then, $\mu_b$ and $\sigma_b$ will be obtained by averaging those
values of $\mu_w$ and $\text{std}_w$, which can be described in the following equation:

$$\mu_w(p') = \frac{1}{N} \sum_{j=1}^{N} w_i(j)$$

$$\text{std}_w(p') = \sqrt{\frac{1}{N} \sum_{j=1}^{N} [w_i(j) - \mu_w(p')]^2}$$

for $i = 1, 2, \ldots, \text{His}(p')$

$$\mu_b = \frac{1}{\text{his}(p')} \sum_{i=1}^{\text{his}(p')} \mu_w(p')$$

$$\sigma_b = \frac{1}{\text{his}(p')} \sum_{i=1}^{\text{his}(p')} \text{std}_w(p')$$

(1)

### 2.1.3. Pixel classification

The pixel classification will classify the pixel on the difference frame into the change region or unchanged region according to the parameter of background model, which can be described as

if $(|DF(i, j) - \mu_b| > c \cdot \text{std}_b)$

then foreground pixel

else

background pixel

(2)

where $DF(i, j)$ is difference frame, and $c$ is a constant.

Based on prior description, the threshold curves for Hall Monitor and Lerry sequence are shown in figure 3 and results of pixel classification are shown in figure 4, when constant value is 10.

### 2.2. Object region detection

Based on the frame difference mask and background subtraction mask obtained in the above subsections, the situation of still object and uncovered background can be solved. However, the background information can’t be obtained completely yet in the previous steps. For this reason, an object region detection rule, as described in Table 1, is used to distinguish various regions in a scene into four kinds in order to generate a coarse object. In case 1, a pixel is recognized as the background region if there is “0” in the corresponding pixel position of both background subtraction mask and frame difference mask. This case implies that there will be no any moving object existed in the region. If only the frame difference mask-bit is detected as change, but unchange for the background subtraction mask-bit, the pixel is classified into the uncovered background region, as shown in case 2. Oppositely, in case 3, the pixel is classified into the still region resulting from overlapping of moving-objects on consecutive frames if only the background subtraction mask-bit is detected as change. The final case describes the moving region identification with “1” for the both frame i.e., a pixel is included in moving region if the pixel is detected as change by both frame difference mask and background subtraction mask checking processes.

### 2.3. Background Updating

Basically, the major function of background updating is to renew a scene when moving-object shift out their original positions or its state is transformed to standstill. In the reported researches [14]-[16], a frame difference mask is fed into a kalman-filter to estimate the background needing to be updated. But this causes a large burden of computations since the kalman gain matrix requires to be updated frequently. Besides, this approach can’t overcome the situation that the moving object occurs on the first frame of the video sequence captured. To reduce computation for a real-time purpose, our method use previous still region mask to construct and update background information. The block diagram of background updating can be illustrated in figure 5.

<table>
<thead>
<tr>
<th>Case</th>
<th>Background Subtraction Mask</th>
<th>Frame Difference Mask</th>
<th>Region Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Background</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>Uncovered-background</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>Still region</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>Moving region</td>
</tr>
</tbody>
</table>
2.3.1. Non-static region elimination
Our scheme of background updating performs with an input of the still region mask from the object region detection phase. In general, the still region mask may contain three types: real static-region, light change, and static part region of moving-object. Actually, we hope the still region mask that it only involves real static-object and light change. In order to realize this purpose, the concept of region adjacent graphic can be used to solve this problem. We can find a fact that if a region is not real static-object, it will connect to region of moving region mask. Hence, this discovery can be utilized to decide the region whether real static-region or not. Figure 6 shows this situation.

2.3.2. Updating selector
The non-static region elimination described previously which will be fail when a moving object existed at beginning of video frames due to the moving object region connect with a region where moving objects shift out their original positions. Hence, the entire connecting region will be removed so that no information used to renew a scene. This problem can be solved by using absolute background detection to decide whether the moving object existed in beginning of video frames or not, for each pixel site \((i, j)\), the four-order moments \(m_4'(i, j)\) is evaluated on a moving window \(w(i, j)\) of \(N_w = 9\) elements of the inter-frame difference \(d_{i,c}(r, c)\) between first frame and \(n\)th frame.

\[
m'_4(i, j) = \frac{1}{N_w} \sum_{(r,c) \in w(i,j)} (d_{i,c}(r,c) - m_\mu)^4
\]

\[
m_c(i, j) = \frac{1}{N_w} \sum_{(r,c) \in w(i,j)} d_{i,c}(r,c)
\]

Finally, the output of updating selector can be decided in following procedure, where \(\sigma^2_n\) denotes variance of background of \(d_{i,c}(r,c)\).

\[
RS = \begin{cases} 
SR, & \text{if } \rho > S_{th} \\
NSRE(sr), & \text{otherwise}
\end{cases}
\]

\[
\rho = \begin{cases} 
\rho + 1 & \text{if } (m'_4 > 75 * \sigma^2_n) \\
\rho & \text{otherwise}
\end{cases}
\]

2.3.3 Background Updating
The basic strategy of background updating is that if a pixel is marked as change on \(RS\) map; the corresponding value in the stationary map is increased by one; otherwise, the corresponding pixel is cleared as zero. Thus, each pixel value in the stationary map indicates the change number of the corresponding pixel in the previous consecutive frame. This strategy of background updating is described as

\[
SM_{i,j} = \begin{cases} 
SM_{i,j} + 1, & \text{if } RS_{i,j} \text{ is change pixel} \\
0, & \text{otherwise}
\end{cases}
\]

2.4. Object boundary refinement technology
A coarse moving object has been extracted through union of moving region mask and still region mask, called as initial object mask. However, in practically, the characteristics and behavior of moving object are not reliable, which makes the accuracy of the obtained boundary of object on temporal to be decreased. Thereby, the procedure of temporal coherence compensation and hierarchical boundary segmentation will be applied to repair the boundary, then, the post-processing filter is used to eliminate noise and smooth the boundary. The block diagram of boundary refinement is shown in figure 8.
2.4.1 Temporal coherence compensation

This function is used to compensate the problem of the fragmentary object which caused by that the motion information is not valid on object region. Hence, we take several frames of the previous initial object mask into count to compensate current frame due to video signal has high correlation between inter-frame. The example can be shown in Figure 9.

2.4.2 Hierarchical segmentation

The proposed hierarchical segmentation which is process of block based so that initial object mask must first be separate into block whose size is $16 \times 16$ and fill them before this functions executed, which shown in Figure 10. We calculate the variance of the boundary block, denoted as $V_{\text{block}}$, if the variance is smaller than a threshold which obtained by difference of gradient of previous frame and current frame plus a constant $\lambda$, the block is regarded as non-moving, and eliminated from the object region, this procedure of hierarchical segmentation can be shown as

\[
\begin{align*}
\text{if} \ (V_{\text{block}} < V_{\text{block}}(G_{f(t)}, G_{f(t+1)}) + \lambda) \\
\text{the block is removed}
\end{align*}
\]

\[
\begin{align*}
\text{else} \\
\text{the block is kept}
\end{align*}
\]

(6)

\[
\begin{align*}
\text{if} \ (\text{blocksize} = 16) \\
\lambda = 2
\end{align*}
\]

\[
\begin{align*}
\text{elseif} \ (\text{blocksize} = 8) \\
\lambda = 1.5
\end{align*}
\]

\[
\begin{align*}
\text{elseif} \ (\text{blocksize} = 4) \\
\lambda = 1
\end{align*}
\]

\[
\begin{align*}
\text{elseif} \ (\text{blocksize} = 2) \\
\lambda = 0.5
\end{align*}
\]

Repeating the procedure for all block along the boundary until the non-moving blocks are all removed. We can further divide the block into $8 \times 8$, $4 \times 4$, and then $2 \times 2$, then more compact boundary can be obtained consequently, see Figure 11.

2.4.3 Object boundary refinement

By way of the above step, a stable boundary of object can be obtained. However, interior of object may be eroded due to effect of hierarchical segmentation introduced and the thick background region still exited around the sharp of object. For first question, we can find the region where is non-union between initial object mask and hierarchical segmentation mask, then, if the value of gradient within this region is larger than a threshold, the corresponding pixel will be compensated to hierarchical segmentation mask, it can be shown in Figure 12. In addition, the problem of thick background which closes to sharp of object will make the compression ratio of object-based compression to be reduced. Therefore, we use strategy of region growing to remove this area, and the seed obtained through

![Figure 9](image1.png) Temporal coherence compensation for Akyio. (a) initial object mask at frame # 29; (b) After using temporal coherence compensation for (a).

![Figure 10](image2.png) Block-froming for boundary of the segmented object in the Thor sequence. (a) Initial object mask of frame #70. (b) Object mask fill and block division of (a).

![Figure 11](image3.png) Hierarchical segmentation for Weather sequence. (a) Object mask fill and block division at frame #22; (b) The block size is $16 \times 16$; (c) The block size is $8 \times 8$; (d) The block size is $4 \times 4$; (e) The block size is $2 \times 2$.
applying the Laplacian operator on hierarchical segmentation mask. This result is illustrated in figure 13.

Figure 12. Remove effect of hierarchical segmentation. (a) The effect of hierarchical segmentation at frame #111; (b) After compensating the effect for (a).

Figure 13. Remove the background region of the boundary by region growing. (a) The result of hierarchical segmentation with block 2 × 2; (b) The seed pixels of region-growing; (c) The grown region using seeds of (b); (d) The result of eliminating (c) from (a).

2.4.4. Post-processing filter
After hierarchical segmentation, the object mask may contain noises and the boundary of object may be not smoothing. Thus, we employ the classic connected components technique [17] on the object mask to mark each isolated region, followed by a threshold to remove these isolated regions. After removing the noise regions, the close and open operations of morphological processing [18] are need to smooth the boundary of object with a small structuring element. Based on such a post-processing, the object mask is so refined that a better object mask with the more complete boundary and without noises will be generated. Figure 14 describes the post-processing, in which the subfigure (b) shows an object mask after noise-eliminating on subfigure (a) which obtained from object boundary refinement step and then the morphological close-open operation is employed on the mask of (b) to yield a final object mask, as shown in (c).

Figure 14. Post-processing: (a) the object mask after hierarchical segmentation; (b) the object mask after noise elimination; (c) the object mask after close-open operation.

3. EXPERIMENTAL RESULTS

In this section, we will demonstrate the experimental results of the proposed video segmentation algorithm. The simulation results show that this algorithm can give accurate video segmentation results for the tested video sequences, and also solve both problems of shadow and residual background region which existed in conventional video segmentation algorithm. A reliable objective criterion, called spatial accuracy, is presented in section 3.1 to evaluate the segmentation result. Finally, in section 3.2, the segmentation results are presented to give subjective evaluation this algorithm, and objective evaluation in spatial accuracy is also shown.

3.1. Spatial accuracy evaluation

A reliable objective criterion for spatial accuracy evaluation [19] is presented, which is to measure the distortion of the algorithm by accumulating the distortion area, then, subtracted by one. Spatial accuracy evaluation can be described as following equation:

\[
\text{Spatial accuracy} = \left(1 - \frac{\sum_{(r,c)} M_{\text{ref}}(r,c) \oplus M_{\text{seg}}(r,c)}{\sum_{(r,c)} M_{\text{ref}}(r,c)}\right) \times 100% \tag{7}
\]

where \(M_{\text{ref}}(r,c)\) is ideal alpha map, \(M_{\text{seg}}(r,c)\) is final object mask generated with this algorithm and \(\oplus\) is exclusive OR.

3.2. Simulation

This algorithm used to simulate: Weather, Thor and Akyio sequences. In figure 15, the segmentation results in initial object region for them are shown. For Weather, which has large motion and residual background region exited, and for sequence Akyio, which has only slight motion so that can not give satisfied results of segmented object region, as shown in figure 15(a)-(d). In figure 15(e)(f), the shadow affects the sequence Thor. In figure 16, the segmentation results corresponding to figure 15 after object boundary process applied. The phenomenon of residual background region has been removed in Weather, as shown in figure 16(a)(b). In figure 16(c)(d) illustrates the problem of fragmentary object can be overcome, and the shadow effect for Thor sequence is also eliminated, as shown in figure 16(e)(f).

Spatial accuracy chart for sequence Weather, Thor, and Akyio are shown in figure 17. From figure 17, we can find out that the spatial accuracy for them can be hold above 95%, which means the accuracy is quite high, and human eye will not pay attention to such small error.
4. CONCLUSIONS

In this paper, an efficient automatic moving-object segmentation algorithm is proposed. By combining change detection with background updating, both frame difference mask and background subtraction mask are used to generate an initial moving object mask. The inherent problems of residual background and shadow phenomenon can be solved by the boundary refinement technique, and spatial accuracy also can be hold above 95%. Finally, a post-processing with noise-filtering and morphological operations is used to remove the noises existed on the object mask and smooth the boundary of object. Although, the proposed algorithm can still achieve the real-time segmentation because most of operations involved are not computation-intensive. Experimental results reveal that the proposed segmentation algorithm will be more attractive the other algorithms.

5. REFERENCES


