Efficient Moving Object Segmentation Algorithm Using Background Registration Technique

Shao-Yi Chien, Shyh-Yih Ma, and Liang-Gee Chen, *Fellow, IEEE*

***Abstract*—An efficient moving object segmentation algorithm suitable for real-time content-based multimedia communication systems is proposed in this paper. First, a background registration technique is used to construct a reliable background image from the accumulated frame difference information. The moving object region is then separated from the background region by comparing the current frame with the constructed background image. Finally, a post-processing step is applied on the obtained object mask to remove noise regions and to smooth the object boundary. In situations where object shadows appear in the background region, a pre-processing gradient filter is applied on the input image to reduce the shadow effect. In order to meet the real-time requirement, no computationally intensive operation is included in this method. Moreover, the implementation is optimized using parallel processing and a processing speed of 25 QCIF fps can be achieved on a personal computer with a 450-MHz Pentium III processor. Good segmentation performance is demonstrated by the simulation results.**

***Index Terms*—Background registration, moving object segmentation, MPEG-4, video segmentation.**

I. Introduction

VIDEO segmentation, which extracts the shape information of moving object form the video sequence, is a key operation for content-based video coding [1], multimedia content description [2], [3], and intelligent signal processing. For example, the MPEG-4 multimedia communication standard enables the content-based functionalities by using the video object plane (VOP) as the basic coding element. Each VOP includes the shape and texture information of a semantically meaningful object in the scene. New functionalities like object manipulation and scene composition can be achieved because the video bitstream contains the object shape information.

However, the shape information of moving objects may not be available from the input video sequences; therefore, segmentation is an indispensable tool to benefit from this newly developed coding scheme. In addition, many multimedia communication applications have real-time requirement, and an efficient algorithm for automatic video segmentation is very desirable.

Conventional video segmentation algorithms can be roughly classified into two categories according to their primary segmentation criteria. Some promising results [4], [5] have been obtained by using spatial homogeneity as the primary segmentation criterion. The major steps of these algorithms can be summarized as follows. First, morphological filters are used to simplify the image and then, the watershed algorithm is applied for region boundary decision. After that, the motion vector of each region is calculated by motion estimation and regions with similar motion are merged together to form the final object region. The segmentation results of these algorithms tend to track the object boundary more precisely than other methods because of the watershed algorithm. However, the computation complexity is very high because both the watershed algorithm and the motion estimation are computationally intensive operations.

Other approaches [6]–[8] use change detection as their primary segmentation criterion. The position and shape of the moving object is detected from the frame difference of two consecutive frames, followed by a boundary fine-tuning process based on spatial or temporal information. We believe that these approaches is more efficient than the previous category because it is the motion that distinguishes a moving object from the background. Algorithms that deal with spatial domain processing first, without knowing the motion information, will waste much of the computing power in segmenting the background. Therefore, an efficient moving object segmentation algorithm should make the most use of the temporal information to achieve higher efficiency.

Several drawbacks exist in the conventional change-detection-based approaches. First, the primary criterion for change detection is frame difference. The value of frame difference depends on the speed of object motion, so the quality of the segmentation cannot be maintained consistently if the speed of the object changes significantly in the sequence.

Second, the uncovered background region will be detected as an object region from the frame difference information. The current solution [7] to remove the uncovered background region is applying motion estimation on regions with significant frame difference. An foreground object is defined as a region where good matching can be found between two consecutive frames; other areas are regarded as uncovered background regions and are discarded. However, motion estimation is a very time consuming operation, the processing speed will be significantly reduced.

Also, the object shadow in the background region can cause trouble in change detection based approach [9]. A simple yet effective solution to reduce the shadow effect should be developed for efficient segmentation algorithms.

In this paper, an efficient video segmentation algorithm that can handle situations with any object motion, uncovered background and shadow effect is proposed. The rest of this paper

Manuscript received May 9, 2000; revised April 16, 2002. This work was supported in part by the National Science Council, R.O.C., under Grant NSC89-2213-E-002-161 and in part by the SiS Education Foundation.

S.-Y. Chien and L.-G. Chen are with the Department of Electrical Engineering and Graduate Institute of Electronics Engineering, National Taiwan University, Taipei 106, Taiwan, R.O.C.

S.-Y. Ma is with Vivotek Inc., Taipei 235, Taiwan, R.O.C.

Publisher Item Identifier 10.1109/TCSVT.2002.800516.

is organized as follows. In the next section, the proposed algorithm is described in detail. Section III discusses the effect of object shadow and how to reduce it. Section IV describes an efficient implementation of the proposed algorithm and then, the experimental results are shown in Section V. Finally, Section VI concludes this paper.

II. Segmentation Algorithm

The basic idea of our segmentation algorithm is change detection. The moving object region is separated from other part of the scene by motion information. However, unlike other change-detection-based approaches [6]–[8], our judge criterion for motion does not come directly from the frame difference of two consecutive frames. Instead, we construct and maintain an up-to-date background information [10] from the video sequence and compare each frame with the background. Any pixel that is significantly different from the background is assumed to be in object region.

In other words, we are not trying to get the object shape information from the changing region of the scene because the characteristics of the changing part is very unpredictable. It depends on object motion, texture, and contrast information which cannot be obtained in advance. Our focus is on the stationary part. This information is more reliable, not very sensitive to object characteristics, and easier to obtain.

An obvious assumption of this approach is stationary background. Since, in many video conferencing and remote surveillance applications, the camera is fixed, we will focus our algorithm on these situations. Some researchers [7], [11], [12] have shown that the background change due to camera motion can be compensated by global motion estimation and compensation. Therefore, we assumed that our input sequence has been properly compensated and the background region is stationary.

The proposed algorithm is divided into five major steps as shown in Fig. 1. The first step is to calculate the frame difference mask by thresholding the difference between two consecutive input frames.

Then, according to the frame difference mask of past several frames, pixels which are not moving for a long time are considered as reliable background in the background registration step. This step maintains an up-to-date background buffer as well as a background registration mask indicating whether the background information of a pixel is available or not.

By the third step, the background difference mask is generated by comparing the current input image and the background image stored in the background buffer. This background difference mask is our primary information for object shape generation.

In the fourth step, an initial object mask is constructed from the background difference mask and the frame difference mask. If the background registration mask indicates that the background information of an pixel is available, the background difference mask is used as the initial object mask. Otherwise, the value in the frame difference mask is copied to the object mask.

The initial object mask generated in the fourth step has some noise regions because of irregular object motion and camera noise. Also, the boundary region may not be very smooth. In the last step, these noise regions are removed and the initial object mask is filtered to obtain the final object mask.

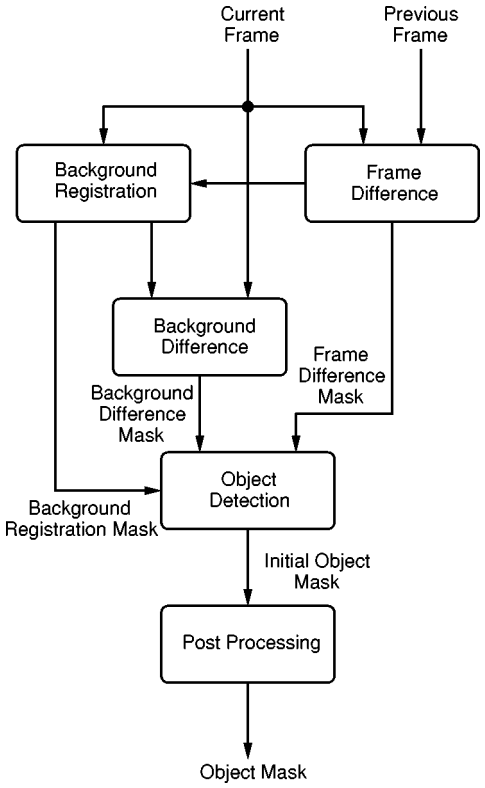

Fig. 1. Block diagram of proposed video segmentation algorithm.

The details of each step will be discussed in the following subsections.

A. Frame Difference

Thresholding the difference between two consecutive input frames is the basic concept of change detection based segmentation. However, since the behavior and characteristics of the moving objects differ significantly, the quality of segmentation result depends strongly on background noise, object motion, and the contrast between the object and the background. Reliable and consistent object information is very difficult to obtain.

Traditionally, a boundary relaxation [6] technique or higher order statistics [8] are used for thresholding to obtain more reliable object shape information by considering the boundary property of the object and the statistic characteristics of the changing part. However, only the motion information between two consecutive frames is used in these approaches, and therefore, object shape information may be lost in regions where the motion stops temporarily. Also, the assumption for the changing part characteristics may not be suitable for all situations.

We use a completely different approach. Instead of trying to get more information from the changing part of the scene, we focus on the stationary background where the characteristics is well known and more reliable. Also, we use the long-term behavior of the object motion accumulated from several frames instead of relying on frame difference of two consecutive frames only.

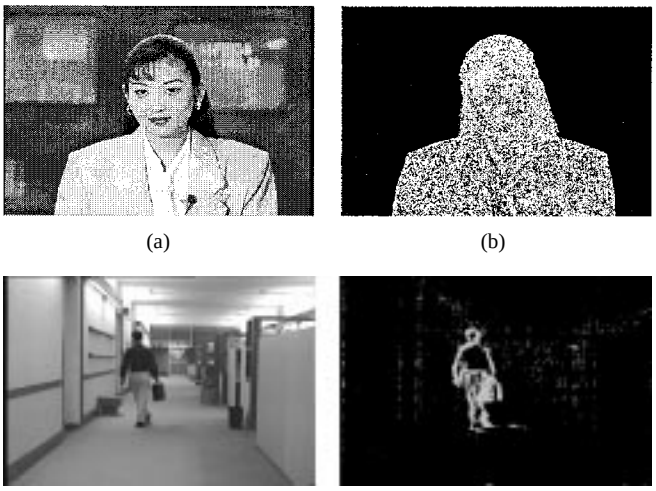

Fig. 2. Frame difference mask example. (a) and (c) The original image. (b) and (d) Frame difference mask.

The frame difference mask is generated simply by thresholding the frame difference. This information is sent to the background registration step where the reliable background is constructed from the accumulated information of several frame difference masks.

A significance test technique [6] is use to obtain the threshold value. Since accumulated frame difference masks are used in the final decision for a reliable background, no filtering or boundary relaxation is applied on the frame difference. The test statistic is the absolute value of frame difference. Under the assumption that there is no change in the current pixel, the frame difference obeys a zero-mean Gaussian distribution and its probability density function is shown in the following equation:

$$p(\mathrm{FD} \mid \mathrm{H}_0) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\mathrm{FD}^2}{2\sigma^2}\right) \tag{1}$$

where FD is the frame difference and σ^2 is the variance of the frame difference and is equal to twice the camera noise variance σ_c^2. H_0 denotes the null hypothesis, i.e., the hypothesis that there is no change at the current pixel.

The threshold value is decided by required significance level. Their relation is shown as follows:

$$\alpha = \mathrm{Prob}\left(|\mathrm{FD}| > \mathrm{TH} \mid \mathrm{H}_0\right) \tag{2}$$

where α is the significance level and TH is the threshold value.

Fig. 2 shows two examples of the frame difference mask. If the camera noise σ_c^2 is small, it is easy to segment the foreground object from the background and we can set a higher significance level α, as shown in Fig. 2(a) and (b). The frame difference mask in Fig. 2(b) is almost ready to be a final object mask. On the other hand, if the camera noise σ_c^2 is large, it is hard to segment the foreground object and the significance level α should be lower. As shown in Fig. 2(c) and (d), although the changing part in the frame difference mask can only present a rough shape of the moving object, the accumulated information in the stationary part can give very reliable background information.

B. Background Registration

The goal of background registration step is to construct a reliable background information from the video sequence. Several approaches [13]–[15] have been proposed to construct and update the background information from the sequence. These approaches are developed for enhancing the coding efficiency in the uncovered background region, so the background information should be constructed as soon as possible. Therefore, complex operations are use to generate the background image.

In our application, we need a reliable background information for change detection. An approximated background information is not helpful for object detection, and even worse, it will cause error in the later segmentation result until the background information is corrected. Therefore, for information that we are not very sure to be background, we tend to reject and leave the corresponding area in the background buffer empty. Also, the operations used in background registration step should be very simple for higher processing speed.

In the background registration step, the history of frame difference mask is considered in constructing and updating the background buffer. A stationary map is maintained for this purpose. If a pixel is marked as changing in the frame difference mask, the corresponding value in the stationary map is cleared to zero, otherwise, if the pixel is stationary, the corresponding value is incremented by one. The values in the stationary map indicate that the corresponding pixel has been not changing for how many consecutive frames.

Our idea is that if a pixel is stationary for the past several frames, then the probability is high that it belongs to the background region. Therefore, if the value in the stationary map exceeds a predefined value, denoted by L, then the pixel value in the current frame is copied to the corresponding pixel in the background buffer.

A background registration mask is also changed in this process. The value in the background registration mask indicates that whether the background information of the corresponding pixel exists or not. If a new pixel value is added into the background buffer, the corresponding value in the background registration mask is changed from nonexisting to existing.

Fig. 3 shows the background registration results for the Weather sequence. The black area means that the background information in that area is not yet available. As shown in Fig. 3(b), background information can be correctly obtained except for the region covered by the reporter. After the reporter moves away from her original position, more background information can be constructed, as shown in Fig. 3(d).

C. Background Difference

This step generates a background difference mask by thresholding the difference between the current frame and the background information stored in the background buffer. This step is very similar to the generation of frame difference mask. The threshold value is also determined by the required significance level according to (2).

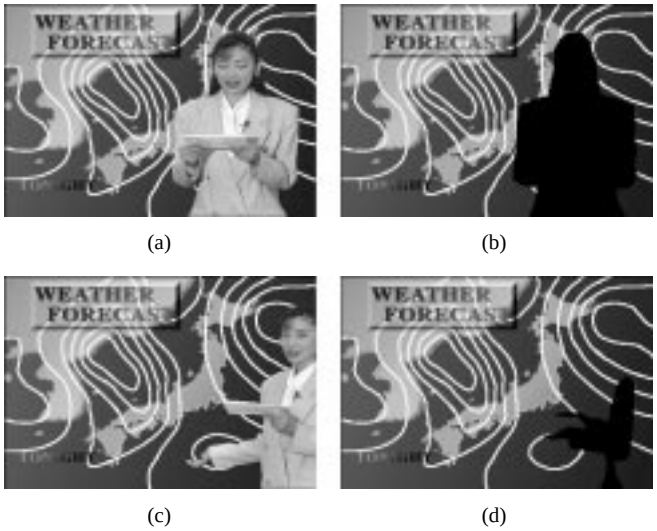

Fig. 3. Construction and updating of the background buffer. (a) Weather sequence frame #50. (b) Constructed background at frame #50. (c) Original image of frame #100. (d) Constructed background at frame #100.

TABLE I
OBJECT REGION DECISION

Index	Background Difference	Frame Difference	Region Description	OM
1	N/A	$\|FD\| > TH_{FD}$	Moving	Yes
2	N/A	$\|FD\| \leq TH_{FD}$	Stationary	No
3	$\|BD\| > TH_{BD}$	$\|FD\| > TH_{FD}$	Moving Object	Yes
4	$\|BD\| \leq TH_{BD}$	$\|FD\| \leq TH_{FD}$	Background	No
5	$\|BD\| > TH_{BD}$	$\|FD\| \leq TH_{FD}$	Still Object	Yes
6	$\|BD\| \leq TH_{BD}$	$\|FD\| > TH_{FD}$	Uncovered Background	No

D. Object Detection

The object detection step generates the initial object mask from the frame difference mask and the background difference mask. The background registration mask, frame difference mask, and background difference mask of each pixel are required information.

Table I lists the criteria for object detection, where $|BD|$ means the absolute value of difference between the current frame and the background information stored in the background buffer, $|FD|$ is the absolute value of frame difference, and the OM field indicates that whether or not the pixel is included in the object mask. TH_{BD} and TH_{FD} are the threshold values for generating the background difference mask and frame difference mask, respectively.

For the first two cases listed in Table I, the background information is not yet available, so the frame difference information is used as the criterion for separating object from background. Although only change detection is used in these situations, the background registration keeps accumulating the background information so the number of pixels without background information reduce rapidly.

For cases 3 to 6 in the decision table, the criteria is background difference because the background information exists. If both the frame difference and the background difference are significant, the pixel is part of a moving object. On the other hand, if both the frame difference and the background difference are insignificant, the pixel should not be included in the object mask. Therefore, for the third and fourth cases in Table I, our result is the same as the result of using only the frame difference for change detection.

Cases 5 and 6 are situations that frame difference based change detection cannot handle properly, but our approach works. One of the problems that confuse the conventional change detector is that the object may stop moving temporarily or move very slowly. In these cases, the motion information disappears if we check the frame difference only. However, if we have background difference information, we can see very clearly that these pixels belong to the object region and should be included in the object mask.

The uncovered background (case 6) is another region where the proposed algorithm outperform the traditional change detection algorithms. Since both the uncovered background region and the moving object region have significant luminance change, distinguishing the uncovered background from the object is not very easy if only the frame difference is available.

Motion estimation has been proposed [7] to solve this problem. If both ends of a motion vector are inside the frame difference mask, then the corresponding area is part of the object. Otherwise, that area is assumed to be background. This approach has several drawbacks. First, the motion estimation is not very accurate near the object boundary where highest accuracy is required. Second, motion estimation can deal with the translation type of motion only; if other forms of movement are involved, motion vectors may fail to track the object motion. Also, motion estimation is a computationally intensive operation; this process will dramatically increase the complexity of the segmentation system.

In our algorithm, the uncovered background region is handled correctly because we recognize that this region matches the background information even though frame difference suggests significant motion. As shown in the sixth case of Table I, pixels in the uncovered background will not be included in the object mask. By eliminating the use of motion estimation in the uncovered background removal process, the computation for object detection can be greatly reduced.

Note that this operation is applied on each pixel independently; that is, pixels in a frame may belong to several difference cases and will be manipulated in difference way as shown in Table I.

E. Post Processing

After the object detection step, an initial object mask is generated. However, due to the camera noise and irregular object motion, there exist some noise regions in the initial object mask. Fig. 4(a) shows an example of initial object mask. We can see from the figure that some noise areas exist in both the background and object region. Also, the object boundary is not very smooth. Therefore, a post-processing step to eliminate these noise regions and to filter out the ragged boundary is necessary.

A traditional way to remove the noise regions is using the morphological operations to filter out smaller regions. The close operation is effective for eliminating the background noise and

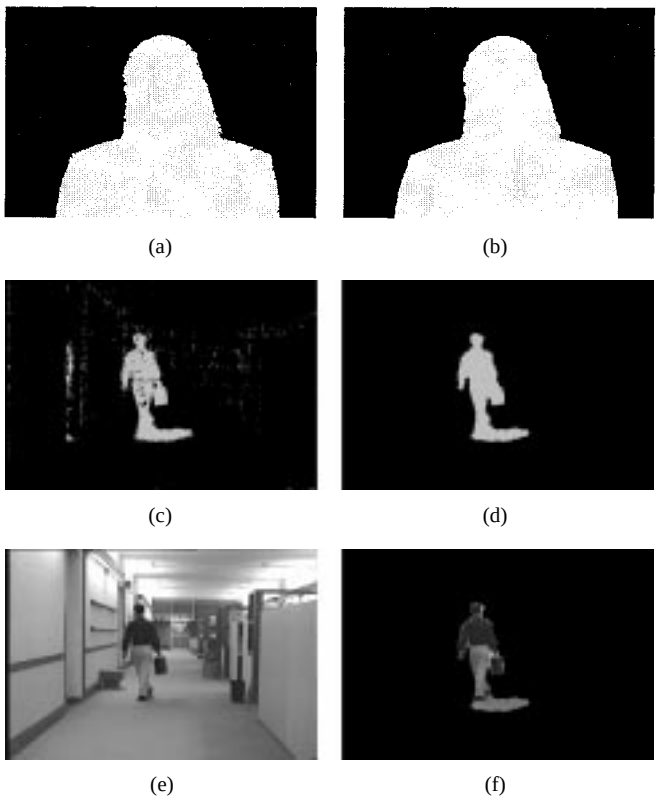

Fig. 4. Effect of noise region elimination. (a) Mask after small-region filtering. (b) Final object mask after close-open operation. (c) Initial object mask. (d) Final object mask after the noise region elimination step. (e) Original image. (f) Segmented object.

the open operation is effective for removing noise within the object region. However, noise regions whose areas are larger than the structuring element cannot be removed by the close or open operations. In order to remove noise regions with large area, larger structuring element should be used. This will not only increase the computation complexity, but also degrade the precision of the object boundary.

Our approach to eliminate the noise region relies on an observation that the area of noise regions tend to be smaller than the area of the object. First, the classic connected components algorithm [16] is applied on the initial object mask to mark each isolated region. Then, the area of each region is calculated. Regions with area smaller than a threshold value are removed from the object mask. In this way, the object shape information is preserved while smaller noise regions are removed.

Since there are two kinds of noise, noise in the background region and noise in the foreground region, two passes are included in this step. The first pass removes small black regions (background regions), which are noise regions in foreground or holes in the change detection mask. The second pass removes small white regions (foreground regions), which are noise regions in background or false alarm regions in change detection mask. After small region filtering, the initial mask in Fig. 2(b) is refined as shown in Fig. 4(a).

After removing noise regions, a close and an open operations with a 3×3 structuring element are applied on the object mask. The small structuring element is chosen to smooth the object boundary without affecting the details of the shape information.

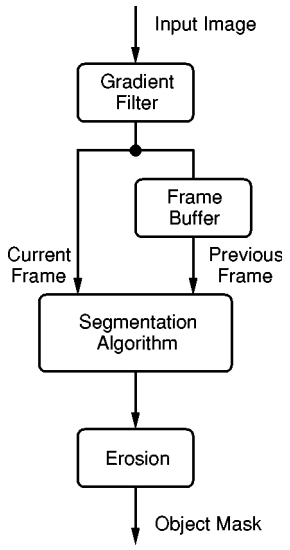

Fig. 5. Block diagram for shadow effect reduction.

After close–open operation, the final object mask is shown in Fig. 4(b).

Another example of this noise region elimination step is also shown in Fig. 4. Fig. 4(c) is the initial object mask and Fig. 4(d) shows the result of this noise region elimination step. The noise in the background region and within the object region are removed. Also, the boundary is smoothed while preserving the shape details.

Fig. 4(e) is the original image and Fig. 4(f) shows the segmented moving object, it can be seen that the object shape is obtained correctly in spite of the presence of noise. The shape boundary tracks the moving object quite well, except in the shadow region, which will be discussed in more details in the next section.

III. Shadow Effects

In many real applications, moving cast shadows may appear in the background region of the scene. Since the shadow changes when the object moves, it is very difficult to distinguish moving object from the shadow region. In our algorithm, the shadow region will be detected as significantly different form the background and marked as object region. Therefore, we need more processing to deal with scenes where shadows are involved.

Stauder *et al.* [9] has done an extensive analysis on the behavior of shadows and has proposed an approach to avoid the moving shadows to affect the segmentation result. However, their approach is very complex and many computational intensive operations are used to achieve good results. For a real-time multimedia communication system, simpler solution is needed.

We propose a simple method to reduce the shadow effects. The block diagram is shown in Fig. 5. The input images are filtered by a gradient filter and then feed into our segmentation algorithm described in the previous section. We use the morphological gradient operation in the gradient filter for its simplicity. The operation is described in the following equations:

$$G = (I \oplus B) - (I \ominus B) \tag{3}$$

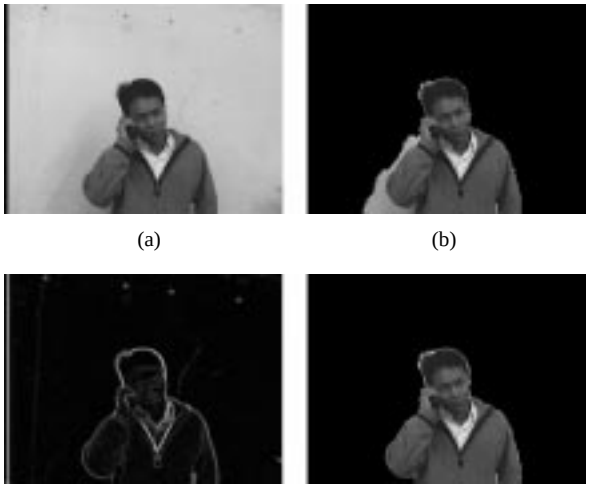

Fig. 6. Effect of gradient filter. (a) Original image. (b) Segmentation result of the original image. (c) Gradient image after applying the morphological gradient operation. (d) Segmentation result of the gradient image.

where I is the input image, G is the gradient image, and B is a 3×3 structuring element of morphological operation.

The reason for using the gradient filter is based on an observation that in normal conditions, shadow area tends to have a gradual change in luminance value. Therefore, after taking the gradient, the values in the shadow region tend to be very small while the edges have large gradient value. The effect of the shadow can be reduced significantly. Another benefit of the gradient filter is that if the illumination or the camera gain change within the sequence, the effect is small in the gradient domain. We can still get very good segmentation results even though the luminance values have changed significantly.

Note that an erosion operation is applied on the output object mask because the dilation operation in (3) will expand the object boundary. This effect can be compensated by applying the erosion operation on the final object mask.

Fig. 6 demonstrates the effect of gradient filter on segmentation results. Fig. 6(a) is the original image—note that shadows appear at the background region due to indoor illumination. Fig. 6(b) shows the segmentation result for the original image, the shadow area is included in the object region. After applying the morphological gradient operation on the original image, the gradient image is shown in Fig. 6(c). We can see from this figure that the shadow area has a low gradient value, while the gradient is large at the object boundary. Fig. 6(d) shows the segmentation result for the gradient image. The shadow region has been successfully removed.

Some limitations exist in this approach. First, the elimination of of shadow effect relies on smooth change in the shadow region. If a shadow appears in regions with strong texture, the benefit of gradient filter will reduce. Also, since the gradient filter removes some image information from the original image, the segmentation result may degrade if the object has a weak edge and low texture. Therefore, this gradient filter is optional in our algorithm and is only turned on when shadows are involved in the input sequence.

TABLE II
RUN TIME ANALYSIS

Function Name	Original	Optimized
File I/O	6	5.5
Object Detection	3.4	3.2
Frame Difference	2	1.4
Background	4.1	0.6
Noise Region	22	16.7
Open-Close	64	11.6
Total	101.5ms	39ms

IV. IMPLEMENTATION

Our goal is to construct an efficient moving object segmentation system. In order to achieve this goal, we avoid the use of computation intensive operations in our algorithm. Also, we have tried to optimize the implementation to achieve faster processing time without compromising the quality of segmentation results.

Table II shows a run-time analysis of our algorithm. Our test platform is a personal computer with a 450-MHz Pentium III processor. The listed numbers are the run time of each function for one QCIF (176×144) image frame in units of milliseconds. The column labeled "Original" shows the run time of a straightforward implementation of our algorithm. The processing time for each frame of the original implementation is 101.5 ms, which corresponds to 9.8 fps.

A basic idea for faster implementation is to compute in parallel, therefore, we tried to exploit parallelism as much as we could. For image processing, most data is 8-bits long and cannot fully utilize the 32-bit or 64-bit datapath of the processor. Many processors have provided multimedia instructions to segment their datapath to perform several 8-bit processing in parallel.

In our algorithm, the absolute difference operations for frame difference step and background registration step use only 8-bit data, and therefore can be calculated in parallel. The Intel MMX instructions [17] are utilized for these functions and the processing speed is three times faster than the original implementations including the overheads. However, the overall performance is improved for only 4% because these operations are not the most time-consuming part of our algorithm.

As shown in Table II, 84% of the run time is spent on the post-processing step so this step is our main focus for optimization. An important property of post-processing operations is that they operate on binary images. Since most microprocessors have 32- or 64-bit datapaths, it is not efficient to store and process a pixel value in bytes or word. Therefore, we use a 32-bit word to represent the binary values of 32 pixels so the morphological operations for these 32 pixels are computed in parallel [18]. The pseudo code of such an implementation is shown in Table III. This bit-parallel is very effective in reducing the computation time and the optimized morphological operations run 5.5 times faster than the original implementation.

Table II shows that our implementation for binary morphological operations is very efficient. However, the connected component algorithm used to remove noise regions now

TABLE III
EFFICIENT IMPLEMENTATION FOR BINARY MORPHOLOGICAL OPERATION

```
Dilate(I, width, height)
{
  // I: input image; T: frame buffer
  // row operation
  for(y = 0; y < height; y++)
     for(x = 0; x < width/32; x++)
       T[x][y] = I[x][y]
               | (I[x][y]<<1 + I[x+1][y]>>31)
               | (I[x][y]>>1 + I[x-1][y]<<31);
  // column operation
  for(y = 0; y < height; y++)
    for(x = 0; x < width/32; x++)
      I[x][y] = T[x][y] | T[x][y-1] | T[x][y+1];
}
```

becomes the speed bottleneck because it is a sequential process and does not benefit from parallelism.

There are two passes of the connected component algorithm in the noise region elimination step, one for removing the noise in the background region and the other for removing the noise in the object region. We observed that the area of noise regions in the background region tend to be small but the area of noise region within the object may be large. Therefore, we replace the connected component algorithm for background noise region elimination with a morphological open operation. In this way, the performance is almost the same but the processing time can be significantly reduced, as shown in Table II.

After these optimization steps, the processing time for a QCIF frame now becomes 39 ms, which is 2.6 times faster than the original implementation. Therefore, we can achieve a processing speed of 25 fps for QCIF format. Compared with the processing speed of other change detection based approach, e.g., 5–10 s per QCIF frame on a 200-MHz Sun Ultra workstation [7] or a watershed based approach, or 2.5 s per CIF frame on a Sun workstation [4], our algorithm runs much faster.

V. EXPERIMENTAL RESULTS

Simulation have been carried out on the standard MPEG-4 test sequences as well as video sequences captured in our laboratory. Both the objective and subjective quality evaluations are applied on our algorithm.

A. 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:

$$\text{Error Rate} = \frac{\text{Error Pixel Count}}{\text{Frame Size}} \tag{4}$$

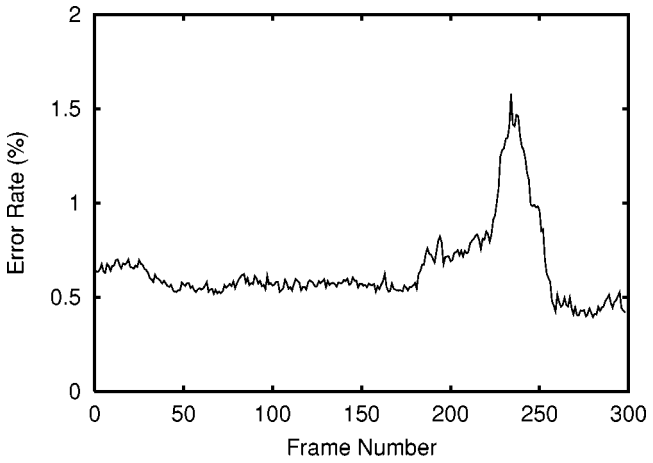

Fig. 7. Error rate in each frame of the Weather sequence (CIF).

where the error pixel count is the number of pixels from which the obtained object mask is different from the reference alpha plane.

Fig. 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 behavior 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.

B. Subjective Evaluation

Fig. 8 shows the segmentation results for several benchmark sequences. The sequences are Akiyo, Weather, Hall Monitor, and Silent Voice in CIF format at 10 fps. Segmentation results at frame #25, #50, and #75 of each sequence are shown in the figure. The Akiyo and Weather sequences do not have background noise so their segmentation results tend to be better than that of other sequences. Background noise does exist in the Hall Monitor sequence; also, shadows cause by indoor illumination appear in the background region. Therefore, gradient filter is used in segmenting this sequence. Our segmentation results track the object shape quite well and are subjectively better than previous results [7], [8],[10].

The Silent Voice is a more difficult sequence to be segmented because of the fast moving shadows on a textured background. A gradient filter is applied on this sequence, but not all of the shadow region can be eliminated. However, the motion of the object can still be tracked and a rough object shape information can be obtained. Note that some background regions are included in the segmentation result of frame #25, since the background information is not ready in those regions.

Fig. 9 shows the segmentation results for the Frank and Shaoyi sequences captured in our laboratory. Shadows are involved in both sequences, but after applying the gradient filter, very good object shape information can be obtained.

Some effects of the region size threshold value described in Section II-D are shown in Fig. 10. The segmentation results of Hall Monitor frame #36 and frame #47 are shown in Fig. 10(a) and (b). The suitcase is not included in Fig. 10(b), since the

(a) (b) (c)

Fig. 8. Segmentation results for four benchmark sequences: Akiyo, Weather, Hall Monitor, and Silent Voice.

(a) (b) (c)

(d) (e) (f)

Fig. 9. Segmentation result for Frank and Shaoyi sequences.

size is smaller than the preset threshold value and regarded as a noise region. This threshold can be set to different values for different applications: if only large objects are important, the threshold value should be higher; otherwise, if small objects are

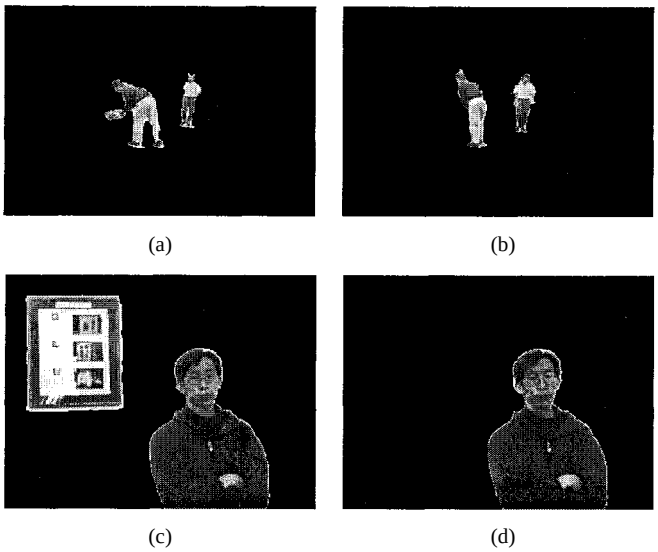

Fig. 10. (a) Hall Monitor #36 and (b) Hall Monitor #47 show the suitcase is not included in the segmentation results because of its small size. (c), (d) Shaoyi # 100 with different region size threshold values.

also important, the threshold value should be lower and some noise regions may not be eliminated in this situation. The segmentation results for Shaoyi frame #100 with different region size threshold values are shown in Fig. 10(c) and (d).

VI. Conclusions

In this paper, we proposed an efficient moving segmentation algorithm. A background registration technique is used to construct reliable background information from the video sequence. Then, each incoming frame is compared with the background image. If the luminance value of a pixel differ significantly from the background image, the pixel is marked as moving object; otherwise, the pixel is regarded as background. Finally, a post-processing step is used to remove noise regions and produce a more smooth shape boundary. In this way, many situations which may cause trouble in conventional approaches can be handled properly without using complicated operations. Shadow effect is a problem in many change detection based segmentation algorithms. In the proposed algorithm, a morphological gradient operation is used to filter out the shadow area while preserving the object shape. In order to achieve the real-time requirement for many multimedia communication systems, our algorithm avoids the use of computation intensive operations. In addition, we optimize the implementation of the algorithm to achieve a even faster processing time. When running on a personal computer with a 450-MHz Pentium III processor, our program can process 25 QCIF (176×144) fps. The experimental results demonstrate that good segmentation quality can be obtained efficiently; therefore, this algorithm is very suitable for the real-time VOP generation in MPEG-4 multimedia communication systems.

References

[1] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 19–31, Feb. 1997.

[2] F. Nack and A. T. Lindsay, "Everything you wanted to know about MPEG-7: Part 2," *IEEE MultiMedia*, vol. 6, pp. 64–73, Dec. 1999.

[3] P. Salembier and F. Marqués, "Region-based representations of image and video: Segmentation tools for multimedia services," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1147–1169, Dec. 1999.

[4] D. Wang, "Unsupervised video segmentation based on watersheds and temporal tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 539–546, Sept. 1998.

[5] J. C. Choi, S.-W. Lee, and S.-D. Kim, "Spatio-temporal video segmentation using a joint similarity measure," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 279–286, Apr. 1997.

[6] T. Aach, A. Kaup, and R. Mester, "Statistical model-based change detection in moving video," *Signal Processing*, vol. 31, pp. 165–180, Mar. 1993.

[7] R. Mech and M. Wollborn, "A noise robust method for 2D shape estimation of moving objects in video sequences considering a moving camera," *Signal Processing*, vol. 66, pp. 203–217, Apr. 1998.

[8] A. Neri, S. Colonnese, G. Russo, and P. Talone, "Automatic moving object and background separation," *Signal Processing*, vol. 66, pp. 219–232, Apr. 1998.

[9] J. Stauder, R. Mech, and J. Ostermann, "Detection of moving cast shadows for object segmentation," *IEEE Trans. Multimedia*, vol. 1, pp. 65–76, Mar. 1999.

[10] T. Meier and K. N. Ngan, "Automatic segmentation of moving objects for video object plane generation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 525–538, Sept. 1998.

[11] H. Nicolas and C. Labit, "Global motion identification for image sequence analysis and coding," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 1991, pp. 2825–2828.

[12] A. Smolić, T. Sikora, and J.-R. Ohm, "Long-term global motion estimation and its application for sprite coding, content description and segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1227–1242, Dec. 1999.

[13] N. Mukawa and H. Kuroda, "Uncovered background prediction in interframe coding," *IEEE Trans. Commun.*, vol. COM-33, pp. 1227–1231, Nov. 1985.

[14] D. Hepper, "Efficiency analysis and application of uncovered background prediction in a low bit rate image coder," *IEEE Transactions Commun.*, vol. 38, pp. 1578–1584, Sept. 1990.

[15] K. Zhang and J. Kittler, "Using background memory for efficient video coding," in *Proc. IEEE Int. Conf. Image Processing*, 1998, pp. 944–947.

[16] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley, 1992, pp. 28–48.

[17] A. Peleg and U. Weiser, "MMX technology extension to the intel architecture," *IEEE Micro*, pp. 42–50, Aug. 1996.

[18] R. V. D. Boomgaard and R. V. Balen, "Methods for fast morphological image transforms using bitmapped binary images," *CVGIP: Graphical Models and Image Processing*, vol. 54, pp. 252–258, May 1992.

Shao-Yi Chien was born in Taipei, Taiwan, R.O.C., in 1977. He received the B.S. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C., in 1999. He currently is working toward the Ph.D. degree at the Graduate Institute of Electrical Engineering, National Taiwan University.

His research interests include video segmentation algorithms, intelligent video-coding technology, and associated VLSI architectures.

Shyh-Yih Ma received the B.S.E.E, M.S.E.E, and Ph.D. degrees from National Taiwan University, Taipei, Taiwan, R.O.C., in 1992, 1994, and 2001, respectively.

He joined Vivotek Inc., Taiwan, R.O.C., in 2000, where he developed multimedia communication systems on DSPs. His research interests include video processing algorithm design, algorithm optimization for DSP architecture, and embedded system design.

Liang-Gee Chen (F'01) was born in Yun-Lin, Taiwan, R.O.C., in 1956. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, R.O.C., in 1979, 1981, and 1986, respectively.

He was an Instructor (1981–1986) and an Associate Professor (1986–1988) in the the Department of Electrical Engineering, National Cheng Kung University. In the military service during 1987–1988, he was an Associate Professor in the Institute of Resource Management, Defense Management College. In 1988, he joined the Department of Electrical Engineering, National Taiwan University, Taiwan, R.O.C. During 1993–1994, he was a Visiting Consultant in the DSP Research Department, AT&T Bell LabS, Murray Hill, NJ. In 1997, he was a Visiting Scholar of the Department of Electrical Engineering, University of Washington at Seattle. Currently, he is a Professor at National Taiwan University. His current research interests are DSP architecture design, video processor design, and video coding system.

Dr. Chen has served as Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since 1996, Associate Editor of IEEE TRANSACTIONS ON VLSI SYSTEMS since January 1999, and Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING. He was the Associate Editor of the *Journal of Circuits, Systems and Signal Processing* in 1999, served as the Guest Editor of *The Journal of VLSI Signal Processing-Systems for Signal, Image and Video Technology*, and was the General Chairman of the 7th VLSI Design/CAD Symposium and the 1999 IEEE Workshop on Signal Processing Systems: Design and Implementation. He received the Best Paper Award from the ROC Computer Society in 1990 and 1994, the Long-Term (Acer) Paper Awards annually from 1991 to 1999, the Best Paper Award of the Asia-Pacific Conference on Circuits and Systems in the VLSI design track in 1992, the Annual Paper Award of Chinese Engineer Society in 1993, and the Outstanding Research Award from the National Science Council and the Dragon Excellence Award frome Acer, both in 1996. He is currently the elected IEEE Circuits and Systems Distinguished Lecturer for 2001–2002. He is a member of Phi Tan Phi.