Image sequence segmentation via heuristic texture analysis and region tracking

Yih-Haw Jan and David W. Lin

Dept. of Electronics Engineering and Center for Telecommunications Research
National Chiao Tung University, Hsinchu, Taiwan 30010, ROC

ABSTRACT

We develop a method for automatic segmentation of natural video sequences. The method is based on low-level spatial and temporal analyses. It features three designs to help facilitate good region segmentation while keeping the computational complexity at a reasonable level. Firstly, a preliminary seed-area identification and a final re-segmentation process are performed on each video frame to help region tracking. Secondly, a simple way to measure homogeneity of texture in a region is devised and the segmentation tries to locate object boundaries at where the texture shows significant changes. And thirdly, a reduced-complexity motion estimation technique is used, so that dense motion fields can be computed at a reasonable complexity. The overall method is organized into four tasks, namely, seed-area identification (for each frame), initial segmentation (only for the first frame in the sequence), motion-based segmentation (for all later frames), and region tracking and updating (also for all later frames). Some examples are provided to illustrate the performance of this method.

Keywords: Image sequence segmentation, object tracking

1. INTRODUCTION

Partly due to the MPEG-4 standards work,[1,2] image sequence segmentation has received much recent attention. A principal objective of such segmentation is to identify and track the motion of semantically meaningful (from a human perspective) video objects (VOs), such as people, houses, and automobiles, so as to facilitate object-based video coding or video content manipulation.[3,4] It is the experience of many researchers that current automatic segmentation methods are far from being perfect and that human intervention helps in achieving better video segmentation.[5,6] Nevertheless, automatic segmentation cannot be dispensed with when there is a large amount of video material to be analyzed.

If an automatic segmentation method is to approach the performance of human-assisted segmentation methods, sophisticated rule-based processing is perhaps unavoidable. In this work, we only resort to "low-level" spatial and temporal analyses. We present an automatic method, which employs heuristic texture analysis and region tracking, for segmentation of natural video sequences. In the following, the terms "video object" and "video region" will be used synonymously.

A primary difficulty in automatic video segmentation consists in maintaining accurate delineation of object boundaries across video frames as their shapes evolve over time and as probable mutual occlusions among the objects occur. In addition, the amount of computation should be kept at a reasonable level. In the method presented here, these issues are addressed, in part, by the three following features. First, a preliminary seed-area identification and a final re-segmentation process are performed on each frame to help facilitate good region tracking. Secondly, the segmentation tries to locate object boundaries at where the image texture shows significant changes. And thirdly, a reduced-complexity motion estimation technique is used, so that the computation of dense motion fields for the objects can be kept at a reasonable complexity.

The remainder of this paper is organized as follows. Section 2 describes our video segmentation method. Section 3 presents some experimental results. And Section 4 gives some concluding remarks.

Y.-H. Jan: yhjan.ee86g@nctu.edu.tw, D. W. Lin: dwlin@cc.nctu.edu.tw.

Visual Communications and Image Processing 2002,
C.-C. Jay Kuo, Editor, Proceedings of SPIE Vol. 4671 (2002) © 2002 SPIE · 0277-786X/02/$15.00

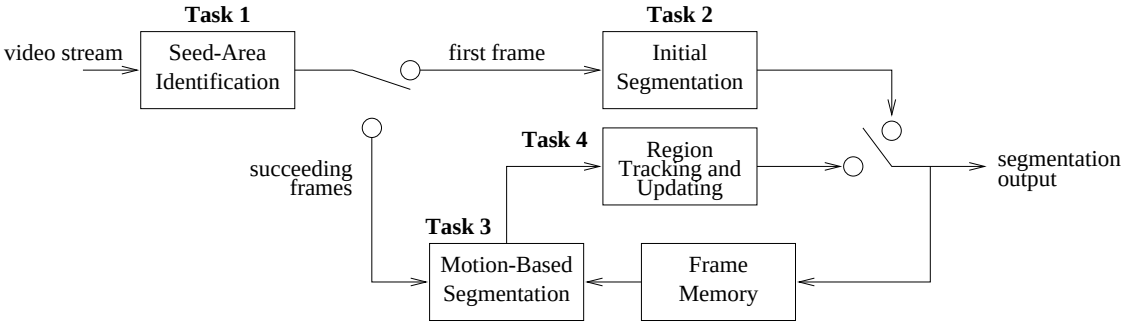

Figure 1: Structure of the video segmentation method.

2. THE SEGMENTATION METHOD

The proposed video segmentation method is illustrated in Fig. 1. It consists of four tasks, namely, seed-area identification, initial segmentation, motion-based segmentation, and region tracking and updating.

The task of seed-area identification is performed on every input frame. Based on simple intensity analysis, it identifies a number of relatively homogeneous *seed areas* in the frame for use in subsequent image segmentation and region tracking. The task of initial segmentation is conducted on the first frame of the sequence only. Starting with the seed areas, it arrives at a segmentation of the first frame by way of region growing and region merging, where the procedure for region growing tries to locate the region boundaries at where the local image texture shows significant changes. The other two tasks are performed on all subsequent frames. The task of motion-based segmentation extracts the moving regions by estimating the motion of each segmented region and integrates the regions showing similar motion. The task of region tracking and updating projects each moving region onto the next frame according to the dense motion vectors obtained in the last task, validates the mapping by examining the seed areas involved in the mapping, and re-segments the uncovered areas, the overlapped areas, and other areas in the next frame which show undesirable features.

As can be seen, the proposed method contains only one pass which operates in the forward direction. Without higher-level intelligence, a one-pass, forward-only method usually has difficulty making correct object segmentation in the first few frames an object appears. The mechanism built in tasks 3 and 4 can modify the segmentation by adjusting the object boundaries in later frames. In non-real-time applications or in applications where longer delays are permitted, a backward analysis or multiple analytical passes over the sequence may be performed based on the present method to attempt at an improved segmentation.

Below we explain each task in detail.

2.1. Task 1: Seed-Area Identification

As stated previousely, the purpose of this task is to divide a frame into a number of seed areas for use in subsequent image segmentation and region tracking. The rationale behind the particular approach is illustrated in Figs. 2 and 3.

Often, an image contains patches wherein the intensity (and color) values are relatively homogeneous. Still-image segmentation often capitalizes on such property and analyzes intensity gradients to determine boundaries. Incidentally, the watershed approach represents such thinking.[7,8] Since our segmentation method assumes no high-level knowledge concerning the image contents but employs only low-level signal processing, it is natural to base the initial segmentation on intensity analysis. And it is natural to consider thresholding the intensity gradients so that gradients above the threshold are considered to mark region boundaries. However, it may not be appropriate to use a single global threshold for the entire image, because the intensity structures of different areas of the image may be different. Using a single global threshold may miss some finer local intensity structures. Figure 2 illustrates the situation where areas A and B are both composed of two patches having relatively homogeneous intensities within each patch but significantly different intensities between patches. But a single-threshold gradient-based segmentation does not find this out. Therefore, the intensity analysis should be conducted more locally. On the other hand, since overly small objects are usually not considered appropriate,

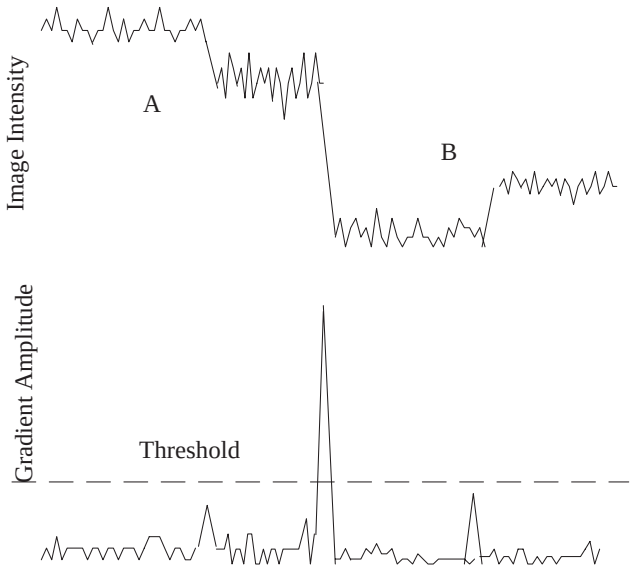

Figure 2. Illustration that a single global threshold may not capture local intensity structures properly. Above: a possible image intensity profile; below: corresponding gradient amplitudes.

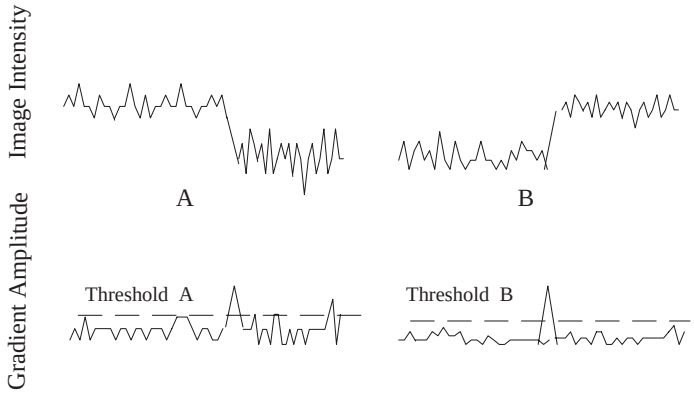

Figure 3. Illustration that a more local gradient intensity analysis may capture the more detailed structures in an image. Above: looking at two image regions separately; below: separate intensity analysis of the two image regions reveals detailed image structure in each region.

the intensity analysis should avoid producing overly fine segmentations. Figure 3 illustrates the idea that a more local gradient analysis may capture the more detailed structures in an image.

In the proposed method, we first calculate the mean intensity μ_I of the whole frame. The frame is called a bright image if more than 50% of the pixels have intensity values greater than μ_I; otherwise it is called a dark image. The pixels whose intensity values are above μ_I are called bright pixels, and those whose intensity values are below μ_I, dark pixels. The following describes the operations carried out on a bright image. The operations carried out on a dark image are

complementary.

For a bright image, each connected set of bright pixels is identified and rank-ordered according to descending sizes. Let R_i denote the ith such bright region. Its intensity mean μ_{R_i} and intensity variance $\sigma^2_{R_i}$ are calculated. Starting with the largest bright region, a region-growing process is performed. The neighboring points whose intensity values are greater than $\mu_{R_i} - \sigma_{R_i}$ are merged into the region. If these points fall in another bright region, then that region's size is reduced and its rank order may be affected. At the end of the process, some smaller bright regions may be merged completely into a larger one. The remaining, unmerged dark pixels may form multiple connected regions. These regions are identified.

For each region in the final set, bright or dark, we analyze the gradient structure and find one or more areas of low gradient values to form the desired seed areas. By this we achieve the localized gradient analysis illustrated in Fig. 3. Specifically, we calculate the gradient values at the interior pixels of each bright or dark region. The boundary pixels are disregarded, because their associated gradient values are expected to be larger on average. The mean gradient value in each region is computed. The pixels with gradients lower than the region mean are obtained. Each connected set of such pixels then constitute a seed area to be used in subsequent image segmentation and region tracking.

2.2. Task 2: Initial Segmentation

The task of initial segmentation is performed only on the starting frame of the video sequence. Its purpose is to effect an initial partition of the image content for later motion estimation and object tracking. The partitioning does not assume high-level knowledge concerning the information content of the frame, but only relies on pixel-domain analysis to divide the frame into regions of different textures.

The task is accomplished in two steps: region growing and region merging. In region growing, we grow the seed areas by examining the local variation and area-wide variation in pixel intensity values. Then in region merging, we merge the small regions until all the region are at least of a given size.

More specifically, in region growing, each seed area obtained in task 1 is assigned a distinct label if its area is greater than a certain threshold, say, N_r. Other points are merged into the seed areas one by one through the following process. Let R_k denote the kth seed area. We calculate its mean μ_{R_k} and variance $\sigma^2_{R_k}$ in intensity. We further calculate the local texture information at each pixel represented by the local mean and variance in intensity in a 3×3 window. For a pixel $p_{i,j}$ at location (i, j), the local mean and local variance are denoted $\mu(i, j)$ and $\sigma^2(i, j)$, respectively. For each seed area, identify all the pixels which are just outside its border. Define the "distance" between one such pixel $p_{i,j}$ and the seed area R_k as

$$d(R_k, p_{i,j}) = \alpha|I(i, j) - \mu_{R_k}| + \quad |\bar{\sigma}_{R_k} - \sigma(i, j)| + \quad |I(i, j) - \mu(i, j)|, \tag{1}$$

where $I(i, j)$ is the intensity of $p_{i,j}$, $\bar{\sigma}_{R_k}$ is the average of the local standard deviations of all the pixels in R_k, and α, , and are some weighting factors. From all the pixel-area pairs, find the one with the smallest distance. Merge that pixel into that seed area if the pixel intensity is within $\mu_{R_k} \pm 3\sigma_{R_k}$.

In the second step, namely, region merging, we make the size of each region at least as large as a predefined value N_r. This is accomplished recursively as follows. For each region that is smaller than N_r, find the nearest other region. From all the region pairs found above, obtain the nearest pair and merge the smaller region in the pair into the larger one. Check the areas of all the remaining regions. If at least one of them is smaller than N_r, then repeat the above process.

2.3. Task 3: Motion-Based Segmentation

This task is geared at estimating the motion of each region and extracting the moving regions. It first constructs a dense motion field for each region using a forward motion estimation approach. To reduce the computational load while obtaining reasonably accurate motion vectors, we first conduct motion estimation for each pixel on the region boundary. The motion estimation employs block-matching techniques. The resulting motion vectors are called *seed motion vectors*. Each pixel in a region is tested only with these seed motion vectors and the best is selected for that pixel. Regions with high percentage of moving pixels are considered moving regions. Connected moving regions are integrated into one big moving region if certain criteria are met. In line with MPEG-4 terminology, each such integrated region may be called a VO.

To test if two connected moving regions may be integrated, we first find the affine motion parameters of each region, where the affine motion model for pixel $p_{x,y}$ in region R_i is given by

$$\begin{aligned} u_{R_i}(x,y) &= a_{i1} + a_{i2}x + a_{i3}y, \\ v_{R_i}(x,y) &= a_{i4} + a_{i5}x + a_{i6}y, \end{aligned} \tag{2}$$

with $u_{R_i}(x,y)$ denoting horizontal displacement and $v_{R_i}(x,y)$ vertical displacement. The affine motion parameters a_{ik}, $k = 1, 2, \cdots, 6$, may be found with any appropriate error-minimization method. For convenience, let $(u_{R_i}^{R_j}(x,y), v_{R_i}^{R_j}(x,y))$ denote the *synthesized motion vector* at pixel $p_{x,y}$ in region R_i obtained by substituting the affine motion parameters for region R_j into the equations (2) for region R_i. Then we do the following.

For each pair of adjacent regions, say R_i and R_j, compute the sum of root-mean-square (RMS) motion vector errors with synthesized motion vectors as

$$\begin{aligned} C(R_i,R_j) &= \sqrt{\frac{1}{N_{R_i}} \sum_{(x,y)\in R_i} \{[u_{R_i}^{R_i}(x,y) - u_{R_i}^{R_j}(x,y)]^2 + [v_{R_i}^{R_i}(x,y) - v_{R_i}^{R_j}(x,y)]^2\}} \\ &+ \sqrt{\frac{1}{N_{R_j}} \sum_{(x,y)\in R_j} \{[u_{R_j}^{R_j}(x,y) - u_{R_j}^{R_i}(x,y)]^2 + [v_{R_j}^{R_j}(x,y) - v_{R_j}^{R_i}(x,y)]^2\}}, \end{aligned} \tag{3}$$

where N_{R_k} denotes the number of pixels in region R_k. Find the pair (i^*, j^*) with lowest error, that is,

$$(i^*, j^*) = \arg\min_{(i,j)} C(R_i, R_j). \tag{4}$$

For it, if both the RMS errors in (3) are lower than a predefined value, then the two regions are integration into one. The better set of affine motion parameter is used for the integrated region, and the same procedure is iterated until no more region integration can be made.

2.4. Task 4: Region Tracking and Updating

In image sequence segmentation for video compression and content-based functionalities, of key importance is proper object tracking over video frames, including proper adjustment of object shapes with time to deal with possible object deformation, possible object occlusions from motion, and possible segmentation errors in earlier frames. By monitoring the progression of each object over the frames, a human viewer may, for example, select some objects of interest and observe their life over the video sequence.

In the task of region tracking and updating, we first project each moving region into the next video frame using the forward motion information obtained in task 3. Some covered and uncovered areas may appear. We look for seed areas (found in task 1) which have nonempty intersection with the footprint of the projection. For each such seed area, if over a certain percentage of its size (for example, 50%) is in the footprint and the area of intersection is no smaller than a predefined threshold (for example, 20 pixels), then the intersection is regarded a valid new seed area. In addition, pixels must have low percertage prediction errors (for example, under 0.05) to be included. The remaining pixels are considered "uncertain areas" to be re-segmented. They include covered areas (areas with overlapped projection from multiple regions), uncovered areas (areas not in the footprint of projection of any region), areas showing relatively large motion-compensated prediction errors, and areas which are isolated and small. The re-segmentation employs region growing from the (modified) seed areas and region merging, as in the task of initial segmentation.

The above procedure allows splitting, merging, and deformation of regions, as are needed. In tracking, one also needs to maintain the correspondence of regions in two successive frames. To deal with splitting and merging of regions, the newly segmented regions in the second frame which fall in the footprint of a region in the previous frame may be considered a valid child of the region. The idea is illustrated in Fig. 4 with the point marked *R*. The correspondence between regions in two successive frames are then affirmed, completing the work of region tracking and updating.

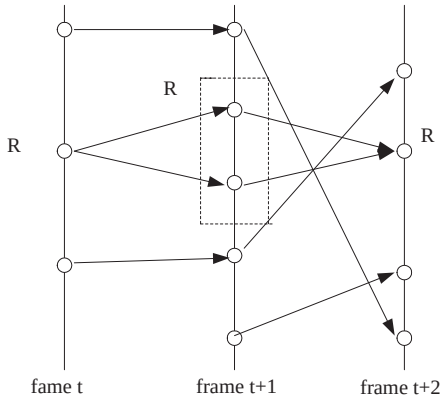

Figure 4: Illustrative temporal progression of different regions.

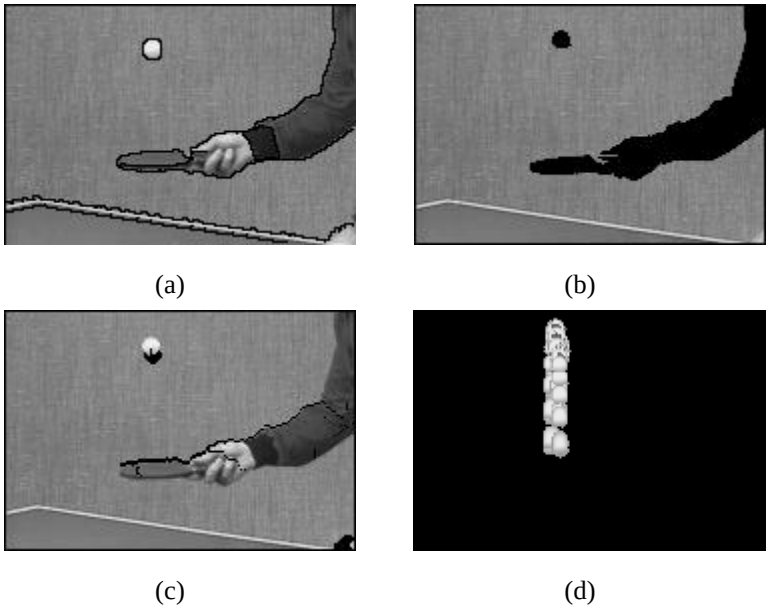

Figure 5. Segmentation performance of the QSIF Table Tennis sequence. (a) Initial segmentation of the first frame. (b) Background portion found in the second frame (except for the small area in the lower-right corner showing the left hand of the player). (c) Uncovered areas in frame 2 (dark points). (d) Tracking of a region (the ball) from frame 1 to frame 35.

3. EXPERIMENTAL RESULTS

To illustrate the performance of the proposed method, we now give some example results from segmenting some common test sequences.

Consider first the Table Tennis sequence, where for convenience the picture resolution used is QSIF (176×120). Figure 5(a) shows the result of initial segmentation of the first frame. Figure 5(b) shows the background portion of the second frame in the sequence. The dark-filled areas mark the locations of two connected moving regions in the foreground. Not dark-filled is a small, third moving region in the lower-right corner of the picture which shows the left hand of the table tennis player. The uncovered areas found in the second frame are shown in dark in Fig. 5(c). The segmented table tennis ball, tracked from frame 1 to frame 35, is shown in Fig. 5(d). The segmentation result every fifth frame of the ball-playing arm is shown in Fig. 6.

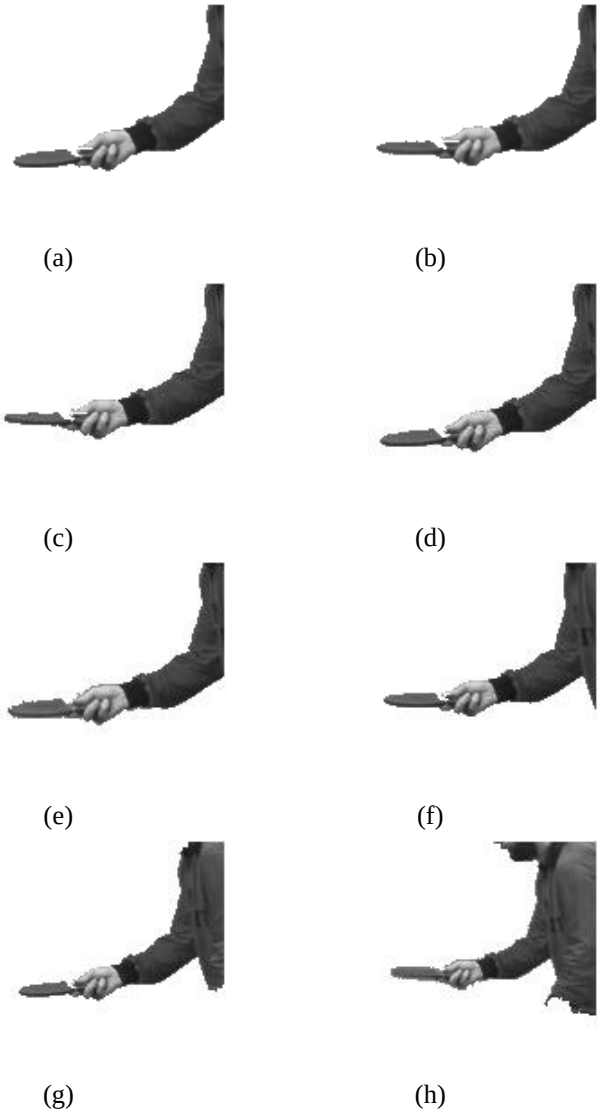

Figure 6. VO tracking result of the ball-playing arm. (a) Frame 5. (b) Frame 10. (c) Frame 15. (d) Frame 20. (c) Frame 25. (d) Frame 30. (c) Frame 35. (d) Frame 40.

Figure 7: Tracking result of the foreground object in the Claire sequence. (a) Frame 1. (b) Frame 60. (c) Frame 70. (d) Frame 80.

Figure 7 shows some tracking result of the foreground object in the Claire sequence. Note that the initial segmentation at frame 1 does not yield the full outline of the talker but only her head. This is not surprising since we did not endow high-level intelligence into the segmentation algorithm. As time progresses, the motion information is picked up and the segmentation becomes more in line with human perception.

Figure 8 shows some tracking result of the foreground object in the Akiyo sequence. Here the outline of the talker appears better identified in frame 1. However, the upper-right part of the talker's head suffers continued mis-segmentation due to likeness in intensity between the hair and the background. The left ear also caused some mis-segmentation at the beginning, but the problem disappeared in frame 80.

These examples confirm the effectiveness of the proposed method to some extent. They also indicate some areas for improvement.

4. CONCLUSION

We presented a method for automatic segmentation of natural video sequences. The method is based on low-level spatial and temporal analyses. It features three designs to help facilitate good region segmentation while keeping the computational complexity at a reasonable level. Firstly, a preliminary seed-area identification and a final re-segmentation process are performed on each video frame to help region tracking. Secondly, a simple way to measure homogeneity of texture in a region is devised and the segmentation tries to locate object boundaries at where the texture shows significant changes. And thirdly, a reduced-complexity motion estimation technique is used, so that dense motion fields can be computed at a reasonable complexity.

The overall method is organized into four tasks. The task of seed-area identification identifies a number of relatively homogeneous areas in each video frame for subsequent image segmentation and region tracking. The task of initial segmentation segments the first frame in the sequence by a procedure which takes the homogeneity of texture in a region into consideration. The task of motion-based segmentation estimates the motion of each segmented region and integrates the regions showing similar motion. And the task of region tracking and updating tracks the regions into the next frame and deals with object occlusion and object deformation.

Figure 8: Tracking result of the foreground object in the Akiyo sequence. (a) Frame 1. (b) Frame 60. (c) Frame 70. (d) Frame 80.

The experimental results show reasonably good segmentation performance, but also indicate some areas for further improvement. For example, a temporally bidirectional or a multiple-pass segmentation procedure may yield better results. Tracking with a longer memory may also help when objects show significant shape changes over time. And a motion estimation method with an even lower complexity than the one used is desirable.

ACKNOWLEDGMENTS

This work was supported in part by National Science Council of R.O.C. under grant NSC 89-2213-E-009-233 and by Lee and MTI Center for Networking Research at National Chiao Tung University.

REFERENCES

1. ISO/IEC JTC1/SC29/WG11, *MPEG-4 Proposal Package Description (PPD).* July 1995.
2. T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 19–31, Feb. 1997.
3. P. Salembier and F. Marqués, "Region-based representation of image and video: segmentation tools for multimedia services," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1147–1169, Dec. 1999.
4. T. Meier and K. N. Ngan, "Automatic segmentation of moving objects for video object plane generation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 525–538, Sep. 1998.
5. R. Castagno, T. Ebrahimi, and M. Kunt, "Video segmentation based on multiple features for interactive multimedia applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 562–571, Sep. 1998.
6. C. Gu and M.-C. Lee, "Semiautomatic segmentation and tracking of semantic video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 572–584, Sep. 1998.
7. D. Wang, "Unsupervised video segmentation based on watersheds and temporal tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 539–546, Sep. 1998.
8. D. Gatica-Perez, C. Gu, and M.-T. Sun, "Semantic video object extraction using four-band watershed and partition lattice operators," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 5, pp. 603–618, May 2001.