Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2009, Article ID 797052, 13 pages
doi:10.1155/2009/797052

Research Article

Spatiotemporal Region Enhancement and Merging for Unsupervized Object Segmentation

K. Ryan,[1] A. Amer,[1] and L. Gagnon[2]

[1] *Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada H3G 1M8*
[2] *R&D Department, Computer Research Institute of Montreal (CRIM), Montreal, QC, Canada H3A 1B9*

Correspondence should be addressed to A. Amer, amer@ece.concordia.ca

Received 22 January 2009; Revised 29 April 2009; Accepted 25 May 2009

Recommended by Bulent Sankur

This paper proposes an unsupervized offline video object segmentation method that introduces a number of improvements to existing work in the area. It consists of the following steps. The initial segmentation utilizes object color and motion variance to more accurately classify image pixels in the first frame. Histogram-based merging is then employed to reduce oversegmentation of the first frame. During object tracking, segmentation quality measures based on object color and motion contrast are taken. These measures are then used to enhance video objects through selective pixel reclassification. After object enhancement, cumulative histogram-based merging, occlusion handling, and island detection are used to help group regions into meaningful objects. Compared to two reference methods, greater success and improved accuracy in segmenting video objects are first demonstrated by subjectively examining selected frames from a set of standard video sequences. Objective results are obtained through the use of a set of measures that aim at evaluating the accuracy of object boundaries and temporal stability through the use of color, motion, and histograms.

1. Introduction

This paper presents an unsupervized offline video object segmentation method based on features combination.

Generally, the goal of video object segmentation is to classify the pixels of a video into groups that represent the objects in that video. For example, in a video shot of a road intersection, we might classify each moving vehicle into its own group, and everything else into one group representing the background. However, this decision can vary depending on the context. A different segmentation method might identify each of the cars tires as a separate object. For this reason, there can be several interpretations of what is a correct segmentation for a particular video sequence. Our objective is to segment video clips into semantically meaningful objects, focusing on the main objects of a sequence. A correct segmentation would consist of video objects like person, automobile, and background, as apposed to dividing the video into smaller objects such as, head, hands, tires, and headlights.

An ideal video object segmentation algorithm should be unsupervized, effective in videos with or without global motion, effective with objects that are moving in some frames but stationary in others, and effective with objects which are nonhomogeneous in color or motion at the frame level. To achieve this, a video object segmentation method must make effective use of as much of the information in a video clip as possible. Two of the most commonly used cues are color and motion. The most basic pieces of information contained in each frame of a video are the color values of the pixels. Many segmentation methods rely on grouping pixels with similar color into the same object, often using algorithms developed for the segmentation of still images. While color is an important tool for segmentation, it is limited in its applicability. Clearly, real objects are not always homogeneous in color, and so segmentation techniques relying on color alone will not always yield satisfactory results. Another cue that can be used to segment video is motion. Motion in a video clip can be expressed as a set of motion vectors. Several methods of calculating this

displacement have been proposed. One commonly used method is block-based motion estimation, where the current frame is divided into blocks, and each block is matched with a block in the last frame by minimizing an error function. Another way to represent motion in a video clip is through the use of parameterized models. For example, the block motion vectors can be used to estimate the 6 affine or 8 bilinear parameters that model the camera motion in a video clip.

Good examples of multiple features-based video segmentation methods are the one proposed in [1, 2]. Those works are our main motivation. The method in [2] is a multiple feature segmentation with adaptive weighting that uses a Maximum A Posteriori (MAP) framework to combine motion and color to segment the first frame and uses the spatial probability density function (pdf) of the formed regions to track them through the remainder of the clip. It is not completely unsupervized, since the number of objects must be known prior to performing the segmentation. The algorithm in [1] is also a multiple feature segmentation one that combines numerous video features at both the frame and sequence levels. The algorithm starts with an initial segmentation using the K-means with connectivity constraint (KMCC) algorithm, over color, motion, and spatial information, followed by an enhancement of the segmented first frame. The next step is a tracking algorithm that uses a Bayes classifier and rule-based processing to reassign changed pixels to existing regions as well as detect newly appearing objects. Finally, a trajectory-based region merging procedure is used to group objects based on their long term motion.

Our work aims to use more video features than [1, 2], gathered through the entire video sequence to obtain improved object segmentation results compared with existing systems. We propose the following improvements compared to the above two related works.

(1) An improved initial segmentation: we include motion and color variances in the distance function of the KMCC algorithm and add histogram distance-based merging.

(2) Segmentation-quality-driven object enhancement: we take a set of segmentation measures while tracking objects to improve the accuracy of object boundaries.

(3) Posttracking merging: we merge regions based on cumulative histograms gathered over the entire clip.

(4) Trajectory-based merging: we handle partial occlusion and deal with isolated regions.

As we will show, our method is unsupervized and is effective under moving or stationary camera and segments objects that become stationary or that are nonhomogeneous with respect to color or motion at the frame level.

The paper is organized as follows. Section 2 gives a quick review of the various video object segmentation techniques. Section 3 describes the proposed segmentation method. Results are presented in Section 4 where we have selected the multiple feature methods in [1, 2] to implement and compare results with ours. Section 5 concludes the paper.

2. Literature Review

The goal of video object segmentation is to classify the pixels of a video into groups that represent the semantically meaningful objects in that video, focusing on the main objects of a sequence such as person, automobile, and background. Many approaches to video object segmentation have been proposed that use different cues to determine the best segmentation. Approaches that are unsupervized, off-line, and which use spatiotemporal information are a subset of them.

One can group many segmentation approaches as follows: layer-based (e.g., [3–5]), color-based (e.g., [6]), motion-based (e.g., [7–9]), edge-based (e.g., [10–12]), and multiple feature-based (e.g., [1, 2, 13, 14]). Other approaches make use of stereo information (e.g., [15, 16]), neural nets (e.g., [17]), graphes (e.g., [18, 19]), and active contours (e.g., [20]), but those are outside the scope of our work. The reader can refer to the review paper [21] and the book [22] for a more extensive references list.

The concept of segmenting video into layers was introduced by [3, 4]. These papers describe how different regions of an image are segmented and stored as layers, which contain information, such as an intensity map of the region and motion information. These layers correspond to the video object planes used in the MPEG verification model. The entire video clip can be represented by the segmented objects in each layer and the relative motion between layers. The authors of [4] use a robust estimation method to iteratively estimate the number of layers and the pixel assignments to each layer. In [3] an affine motion model is fitted to blocks of optical flow. Then, a K-means [5] approach is used to cluster the image points according to their affine parameters.

Color characteristics are sometimes used to segment video objects. One recent example of this is [6], where color-based deformable models are used to segment and track objects. This method uses color constant gradients, and a model is proposed estimating the sensor noise through these gradients. As a result, this method is robust when dealing with noisy data. As well, only color, and not intensity, is used so that the method can deal with illumination changes. However, this method is only effective when dealing with homogeneous objects and does not handle occlusion.

Motion-based approaches to video object segmentation are commonly employed as they often provide improved results on video clips for which color-based methods encounter problems. The authors of [7] present a number of region-based affine-parameter clustering methods using motion vector and intensity matching to align motion boundaries with real object boundaries. They then go on to use a specific combination of these methods to segment a number of video clips. A different motion-based approach to segmentation is presented in [23]. Here, the motion estimation error along occluding boundaries of moving objects is studied. The authors show how the nature of this error can be used as a depth cue. Their segmentation approach involves segmenting the image based on color and motion independently. Then, by examining the motion

estimation error at region boundaries, they are able to determine what are the occluding and occluded objects. In this way they are able to establish the relative depth of the image segments. The focus of [8] is on extracting objects with similar motion. The 2-step process consists of generating 3D watershed volumes followed by a Bayesian merging of these volumes. In the first frame, markers are extracted which provide reliable seed regions for segmentation in subsequent frames. One weakness of this method is that it is assumed that the number of video objects is previously known. In [9], deformable binary object models are used to segment and track objects. The models are updated from frame to frame and are therefore able to accommodate complex object motion as well as changes in shape. The models are updated using a modified watershed-based method. Like other methods, there is an initial detection/segmentation step followed by a tracking step. This method can handle moving backgrounds and partial occlusion. However, since the segmentation is based on motion and is done on the first frame, only objects that are present and moving in the first frame are detected. Newly appearing or stationary objects cannot be detected.

There has also been significant research into using edge detection to segment video. The extracted edges are used to determine the boundaries of the segmented objects. One major difficulty with this approach is deciding which edges represent object boundaries and which are the result of other image properties, such as textured surfaces. The authors of [10] try to deal with this problem by using a multiresolution approach to edge detection. A method of determining the optimal scale at each edge by examining edge strengths is presented. The edges at these optimal scales are then used to segment the image. Boundary completion techniques are used in [24] to complete contours that are smooth but have low contrast. However, this method is vulnerable to problems when dealing with textures. In [25], textures are dealt with explicitly by modeling them with textons. By combining texture cues with intervening boundary cues, this approach is able to deal with both textured and nontextured areas. A different approach to improving edge-based segmentation is taken by [11]. This algorithm uses information from edges at multiple scales. Instead of trying to select the optimum scale for each edge, and then segmenting the image on the selected edges, this approach collects edge information at multiple scales and then does a simultaneous segmentation over all the edges. This method can capture both large and small scale image properties as well as deal with textured areas.

Much of the recent work focuses on using multiple video features to aid in segmentation. The authors of [3] use color and motion to segment objects in the first frame, which are then tracked by using their estimated motion to predict their location in the next frame. This method can also segment new objects that appear after the first frame. In [2], a maximum a posteriori (MAP) framework is proposed. They assign weights to color and motion terms, which are adjusted at every pixel. They also model the spatial probability density function (pdf) of each region in order to impose temporal consistency. A slightly different approach is employed by [1]. Instead of segmenting based on motion at the frame level only, regions which have been divided based on color, motion, and position are tracked. The long-term trajectories of these regions are used to group them into an appropriate segmentation. Segmentation algorithms such as this one, which perform multiple passes through a video clip, are referred to as offline methods. Methods which only require knowledge of the current and previous frames being segmented are referred to as online methods.

3. Proposed Segmentation Method

A flow chart of the proposed method is depicted in Figure 1. It is divided in to five steps.

The first step is initial segmentation (Section 3.1), where we first apply a modified KMCC algorithm followed by a test for convergence. If the algorithm does not converge, indicating undersegmentation, we repeat initial segmentation but using the KMCC algorithm used in [1].

The second step of the proposed method is histogram and motion-variance-based region merging (Section 3.2) where regions are merged following an iterative process until convergence is achieved.

The third step is temporal tracking (Section 3.3), where existing regions are tracked and new objects detected. This step is followed by segmentation-quality-driven object enhancement (Section 3.4), where objects are selected for enhancement based on segmentation-quality measures. Key frames are then selected and used to reclassify pixels of the selected objects.

The fourth step of the proposed algorithm is a post-tracking region merging (Section 3.5), where cumulative histograms are used to iteratively merge regions until convergence.

The final step of our algorithm is a trajectory-based region merging (Section 3.6), where trajectories are used to merge regions in an iterative fashion. Upon convergence of the region merging, island regions are detected and merged into their surrounding regions.

3.1. Initial Segmentation. The initial segmentation of [1] uses the Euclidean distance of each pixel from each region's color and motion center-based to classify pixels. The motion estimation method is a block-based. This is effective for regions that have relatively simple color and motion distributions but can result in errors for more complex regions. In order to more accurately classify pixels, higher-order statistical information has to be taken into account.

We propose to include variance information about the color and motion distributions of each region in the KMCC distance function. After the initial centers are estimated, the feature variance of each region is calculated, and pixels are classified according to their distance from the center of each feature divided by the variance. So we propose the distance function

$$D_{\mathrm{KMCC}} = \frac{\left\|\mathbf{C}(\mathbf{p}) - \overline{\mathbf{C}}_{R_i}\right\|}{\sigma^2_{R_i,C}} + \lambda_1 \frac{\left\|\mathbf{M}(\mathbf{p}) - \overline{\mathbf{M}}_{R_i}\right\|}{\sigma^2_{R_i,M}} + \lambda_2 \frac{\overline{A}}{A_{R_i}} \left\|\mathbf{p} - \overline{\mathbf{S}}_{R_i}\right\|, \tag{1}$$

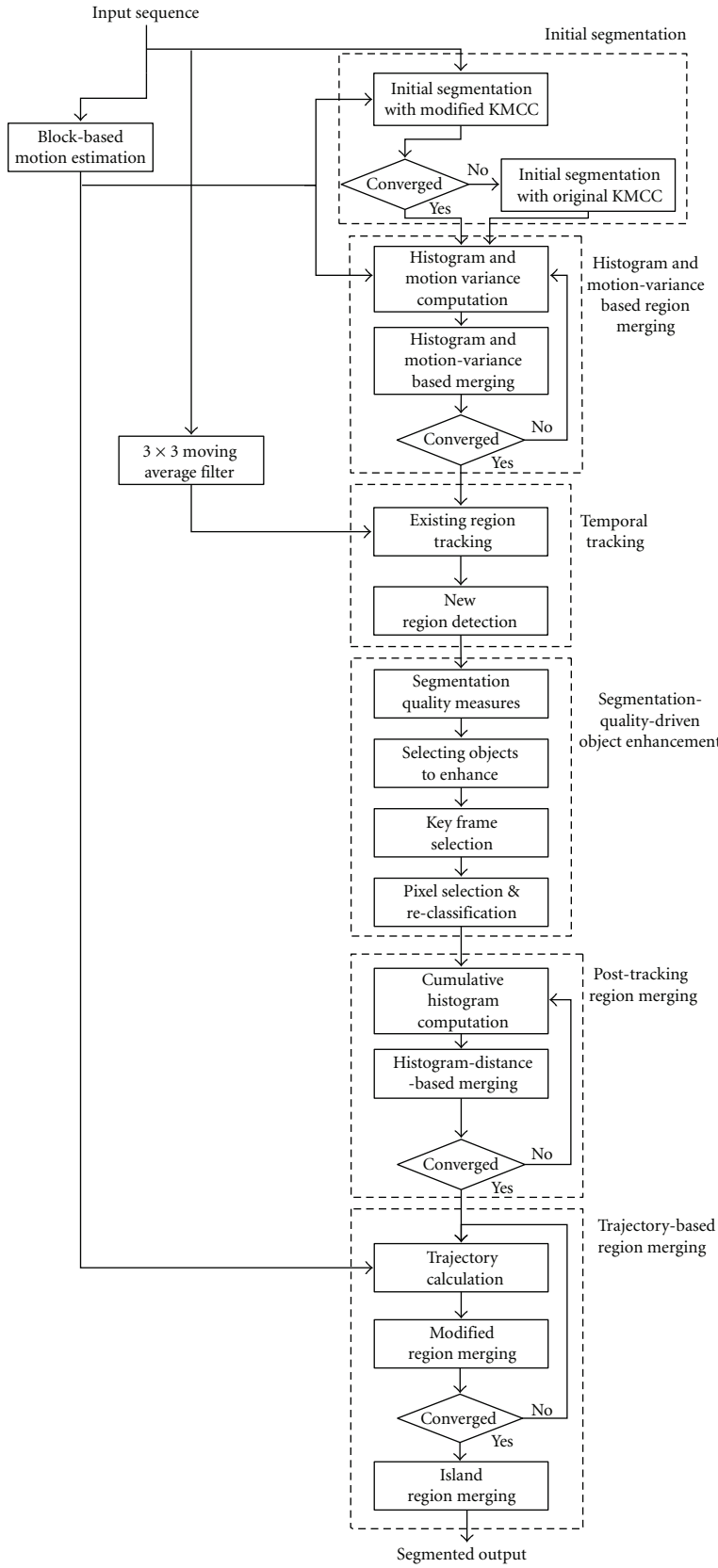


Figure 1: Flow chart of the proposed segmentation algorithm.

where $\overline{\mathbf{C}}_{R_i}$, $\overline{\mathbf{M}}_{R_i}$, and $\overline{\mathbf{S}}_{R_i}$ are the color, motion, and spatial centers of region R_i, respectively. $\mathbf{C}(\mathbf{p})$ and $\mathbf{M}(\mathbf{p})$ are the color and motion vector values for image point $\mathbf{p}$. A_{R_i} is the area of region R_i in pixels, and $\overline{A}$ is the average region area. $\sigma^2_{R_i,C}$ and $\sigma^2_{R_i,M}$ are the color and motion variances of region R_i, and λ_1 and λ_2 are regularization parameters defined in [1]. Classifying pixels in this way is more accurate than using only distances from region centers as in [1], since more information about the distribution of each region is being utilized. Also, this method divides the image into a smaller number of more complex regions, which reduces the oversegmentation normally associated with the KMCC algorithm. Reducing the oversegmentation of the first frame decreases the chances for error in later stages of the algorithm.

To improve the robustness of the initial segmentation, we examine the regions at the end of each iteration i of the KMCC algorithm. If the algorithm converges to less than two regions (indicating undersegmentation), R_i, R_j, that meet

$$A_{R_i} > \alpha \cdot X \cdot Y, \tag{2}$$

where A_{R_i} is the area of region R_i, X and Y are the image dimensions, and α set experimentally to 0.02, the entire process resets, and the original KMCC is used. The criteria in (2) is used to enforce the intuitive notion that we expect any sequence to have at least one nonbackground object that is of significant size. Through experimentation, we have found that setting this size threshold to 2% of the image area provides an effective test for under segmentation.

3.2. Histogram and Motion-Variance Based-Region Merging. The next stage of the initial segmentation is a histogram and motion-variance-based merging stage. The reference KMCC algorithm incorporates merging of neighboring regions whose color and motion centers are below a certain threshold. This merging process is another area that can be improved by using higher-order statistical information about the regions being examined. We accomplish this through the use of color histograms and the motion variance of each region.

First, color histograms are calculated for each region using the CIE L∗a∗b∗ color space. This is done by dividing the region into a 3-dimensional array of bins, where the value in each bin is the number of occurrences of that color in the region. This provides a more complete representation of a regions color distribution than using a color center or a simple statistical representation, such as a Gaussian.

Once color histograms have been calculated, the χ^2 histogram distance between each pair of neighboring regions is measured as

$$\forall R_i, R_j \in P_1, \quad \chi^2\left(H_{R_i}, H_{R_j}\right) = \sum_b \frac{\left(H_{R_i}(b) - H_{R_j}(b)\right)^2}{\left(H_{R_i}(b) + H_{R_j}(b)\right)}, \tag{3}$$

where P_1 is the set of all pairs of neighboring regions (R_i, R_j) in the first frame, H_{R_i} and H_{R_j} are the histograms of R_i and R_j, and b is the histogram bin. After the distances have been calculated, all neighboring regions satisfying (4) and (5) are merged:

$$\chi^2\left(H_{R_i}, H_{R_j}\right) < \beta \cdot S_{\text{hist}}, \tag{4}$$

$$\left|\left|\overline{\mathbf{M}}_{R_i} - \overline{\mathbf{M}}_{R_j}\right|\right| < \epsilon \cdot \max\left(\sigma^2_{R_i,M}, \sigma^2_{R_j,M}\right), \tag{5}$$

where β is experimentally set to1.3, and ϵ is experimentally set to 2. S_{hist} is the histogram size, defined as the sum of all bins in the histogram. Histograms are normalized so that both histograms being compared have the same size, S_{hist}.

After merging, we re-evaluate the region motion centers and histograms and redetermine neighbor relationships. The merging continues until no more regions meet (4) and (5).

3.3. Temporal Tracking. After the initial segmentation, we use the temporal tracking approach of [1] to track the segmented regions through the remainder of the clip. The temporal tracking begins with a frame difference and thresholding of the current and previous frame, where both frames have first been filtered with a moving average filter. Pixels with a color difference above the threshold are marked as disputed, and those with a difference below the threshold are marked as nondisputed. We then use a Bayes classifier to assign the pixels in each disputed region to one of its neighboring nondisputed regions, using the histograms for each region from the previous frame as the a priori probability. New regions are detected by first measuring how the homogeneity of each nondisputed region is affected by adding its neighboring disputed pixels. If a nondisputed region's homogeneity is significantly reduced by the addition of its neighboring disputed pixels, those disputed pixels are assigned to a new region.

3.4. Segmentation-Quality-Driven Object Enhancement. During object tracking, we measure the segmentation quality of each object in each frame. We use the following three measures to do this.

(1) Color homogeneity of the region [1]. This is defined as the average of the MAP probabilities of every pixel in the region and is determined from the color histograms for each region.

(2) Color contrast across the object boundary [26]. This measure was shown in [26] to be an effective objective measure of segmentation quality. The object contour is traced, and all along the object boundary pairs of blocks are chosen, with each pair consisting of one block inside and one block outside the object. The mean color value for each block is calculated, and the absolute difference between each pair of blocks is taken. The color contrast is the average of all these absolute differences along the object boundary.

(3) Motion contrast across the object boundary [26]. This is calculated in a similar manner to the color contrast, except that motion vectors are used instead of color values.

After objects have been tracked through the entire clip, we examine these segmentation measures and each object's movements to determine which objects we will enhance, and for which frames we will perform the enhancement.

For a given object, most variation in object segmentation quality between frames is due to movement. Therefore, we are here mainly interested in moving objects. To this end, we examine the trajectories of all objects in the entire video clip and choose which ones to enhance as follows.

The (x, y) coordinates of each object's center in each frame are used to calculate the maximum displacement of every object in the clip. The displacement is taken with respect to the first frame. Objects whose maximum displacement is above a certain threshold are considered to have undergone significant motion and are candidates for enhancement as in

$$\begin{aligned} \Delta\mathbf{D}_{R_i,\max} > t &: \text{enhance } R_i, \\ \Delta\mathbf{D}_{R_i,\max} \le t &: \text{keep } R_i, \\ \forall R_i \in I, \quad t &= \frac{\sqrt{\overline{A}_{R_i}/\pi}}{2}, \end{aligned} \tag{6}$$

where $\Delta\mathbf{D}_{R_i,\max}$ is the maximum displacement of region R_i over the entire clip I and $\overline{A}_{R_i}$ is the size of R_i averaged over I. The maximum displacement $\Delta\mathbf{D}_{R_i,\max}$ is calculated by recording the spatial center of the object in each frame and by finding the maximum distance from the object's initial position. The calculation in (6) is based on the criteria that a circular object would need a displacement of greater than half its radius to be considered a good candidate for enhancement. However, for arbitrarily shaped objects, this calculation still serves to provide a rough measure of how far the object has moved in relation to its size.

Once we have chosen which objects to enhance, we examine their segmentation quality measures for each frame and enhance objects according to the following rules.

(1) If an object's color homogeneity in a given frame is below that same object's average color homogeneity for all frames, this indicates that pixels belonging outside the object have been classified inside the object in this frame. In this case, pixels within the object and close to the boundary will be marked as disputed and reclassified.

(2) High color homogeneity with below average color contrast indicates that pixels belonging inside the object have been classified outside. In this case, pixels close to the boundary but outside the object will be reclassified.

(3) High color homogeneity with high color contrast indicates a good segmentation. Nothing will be done.

We reclassify pixels through a Bayesian approach [1] using histograms from key frames of the clip to determine the MAP probability of each disputed pixel. Out of every five frames, the frame with the highest homogeneity and contrast is a key frame. The disputed pixels in each frame are reassigned based on each object's nearest key frame histogram.

(a) (b)

Figure 2: Effect of segmentation-quality-driven enhancement for frame I_{15} of tennis sequence: (a) without enhancement, (b) with enhancement. Improved accuracy of the tennis ball boundary can be seen.

After reassigning pixels, we perform an error check based on the assumption that object enhancements should not result in drastic changes in object size, and that motion contrast should not decrease. The error check fails if either of these conditions occur, as in

$$A_{R_i} > 2.0 \cdot A'_{R_i} \quad \text{or} \quad A_{R_i} < 0.7 \cdot A'_{R_i} \quad \text{or} \quad C_{M,R_i} < C'_{M,R_i}, \tag{7}$$

where A'_{R_i} and A_{R_i} are the object sizes before and after the enhancement, and C'_{M,R_i} and C_{M,R_i} are the motion contrast before and after the enhancement. Due to the use of block-based motion estimation, motion contrast is not effective for locating small inaccuracies in object boundaries, and so it was not used in selecting the frames needing improvement or the key frames. However, a decrease in motion contrast does indicate a significant reduction in boundary accuracy, making motion contrast an effective measure for error checking. If the enhanced object fails either of the error checks, the enhancement is rejected; otherwise it is accepted.

This enhancement stage improves the boundaries of tracked objects over that of [1]. This also allows more accurate motion parameters to be estimated for each object, improving the performance of the trajectory-based merging stage. Figure 2 shows results for a selected frame of the tennis sequence when the proposed method is run with and without the segmentation-quality-driven object enhancement. The table tennis ball was selected for enhancement, and significant improvement of the object's boundary can be seen.

3.5. Posttracking Region Merging. Posttracking region merging simplifies the trajectory-based merging stage (Section 3.6). This is desirable, because trajectory-based merging can fail when an object's motion is too complicated (deformation or articulated motion), or when accurate motion vectors are not available (e.g., when an object is highly uniform in color).

Color histograms are used to merge regions which are spatiotemporal neighbors. We use the same spatiotemporal neighborhouod definition as [1], that is, "two regions are spatiotemporal neighbors if they coexist in at least one segmentation mask and they are spatial neighbors in all segmentation masks that they coexist in" (see Figure 5 in [1]). Here we use cumulative histograms calculated from an object's pixels taken over all frames in the clip.

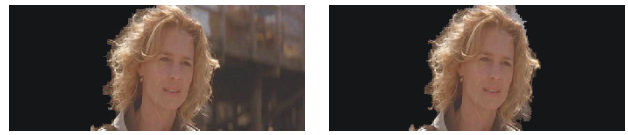
(a) (b)

Figure 3: Effect of histogram-based merging for frame I_1 of foreman sequence: (a) without histogram-based merging, (b) with histogram-based merging. Improved segmentation of the background region can be seen.

Compared with histograms computed for an object in a single frame, cumulative histograms are less sensitive to noise, inaccurate object boundaries for particular frames, changing illumination, and occlusion. For example, an object with lighting that varies across its surface in the first frame could be segmented into two regions, but as the object moves these illumination differences could even out, and the two halves of the object can be merged. As with the first frame histogram-based merging (Section 3.2), the χ^2 histogram distance (3) is used to select regions to merge. This stage improves the segmentation of objects with complex motion that present problems for [1].

Figure 3 shows results for a selected frame of the foreman sequence with and without the histogram-based merging enabled, demonstrating that parts of the background are misclassified when the histogram-based merging is not employed. The histogram-based merging causes these regions to be merged into the background, preventing them from being incorrectly assigned to the foreground during the subsequent trajectory-based merging stage.

3.6. Trajectory-Based Region Merging. We propose a trajectory-based merging that accounts for high occlusion of the background. We use the same trajectory-based merging of [1] but with an extended definition of spatiotemporal neighbor criteria. The trajectory-based merging stage of [1] only examines regions which are spatiotemporal neighbors. However, since region connectivity is enforced during the initial segmentation with the KMCC algorithm, it is possible for the background to be initially segmented into multiple regions that are not spatiotemporal neighbors. One example is when there is a large object, extending from top to bottom in the middle of a frame. In these cases, the video cannot be segmented correctly without merging these nonneighboring background regions. To account for this, any region that contains a corner point, $(0,0)$, $(X-1,0)$, $(0,Y-1)$, $(X-1,Y-1)$, of a frame is considered to be a potential background region and will be treated as a spatiotemporal neighbor of all other potential background regions in the clip for the purposes of trajectory-based merging. With this change of the spatiotemporal neighbor criteria, we are able to correctly segment the disconnected pieces of the background, while still enforcing connectivity of all other objects. Furthermore, after the trajectory-based merging is finished, any island regions (those with only one spatiotemporal neighbor which is not a potential background region) are merged into their surrounding object.



(a) (b)

Figure 4: Effect of trajectory-based merging for I_1 of Harbor sequence: (a) without enhancement, (b) with enhancement. Background occlusion is taken into account which greatly improve segmentation.

Figure 4 shows results for the proposed method when the trajectory-based merging does not account for background occlusion. It can clearly be seen that in this case the background is not correctly segmented but is instead merged with the actor as part of the foreground.

4. Results

4.1. Algorithm Parameters. The proposed segmentation algorithm utilizes the following parameter values for all video sequences.

α: First frame undersegmentation threshold (2), set to 0.02. This parameter is set so that if there is not at least one object in the first frame with an area greater than 2% of the image size, it is assumed that we have under segmented and the initial segmentation resets.

β: Histogram merging threshold (4), set to 1.3. The value of this parameter is chosen to provide an effective histogram-based merging stage, while preventing the merging of regions which do not belong to the same object.

ϵ: Motion variance merging threshold (5), set to 2. It is used along with the histogram merging threshold during the histogram and motion-variance-based region merging stage.

t: Threshold used to determine whether or not to enhance a given object (6). The value is set to $\sqrt{(\overline{A}_{R_i}/\pi)}/2$. This provides means of measuring an objects motion relative to its size. Larger objects would require a larger absolute displacement to be considered for enhancement.

4.2. Subjective Results. Simulations were done for a number of standard video test sequences listed in Table 1. We present visual results for seven of them that are representative of the whole set. Figures 5 to 11 show sample results where results of the method in [1] are labeled reference method 1, and results for the method in [2] are labeled reference method 2.

Figure 5 presents results for the Gameshow test sequence. The main object in this clip consists of multiple colors and motion that is difficult to model accurately (there is some movement of the neck and head, while the body remains mostly stationary). Improvement over reference method 1

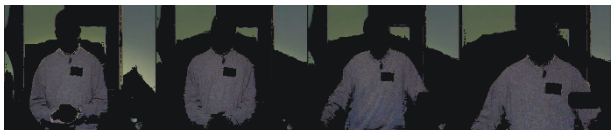
Table 1: Test sequences (total of 1855 frames) used in simulations.

Sequence	Dimensions	Frames	GM
Coastguard	352 × 288 (CIF)	300	Pan
Gameshow	352 × 288 (CIF)	600	Zoom
Mobile	352 × 288 (CIF)	100	Pan
Foreman	352 × 288 (CIF)	300	Pan
Basket ball	352 × 288 (CIF)	20	Pan
Harbor	564 × 240	50	Pan
Tennis	352 × 288 (CIF)	60	Zoom
Miss	176 × 144 (QCIF)	150	None
Suzie	176 × 144 (QCIF)	150	None
Road1	352 × 288 (CIF)	30	None
Carphone	176 × 144 (QCIF)	95	None

(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]



(d) Reference method 2 [2]

Figure 5: Frames 1, 120, 360, and 600 of the Gameshow sequence (some GM).

(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 6: Frames 1, 20, 30, and 40 of the Harbor sequence (with GM).

(a) Original clip

(b) Proposed method

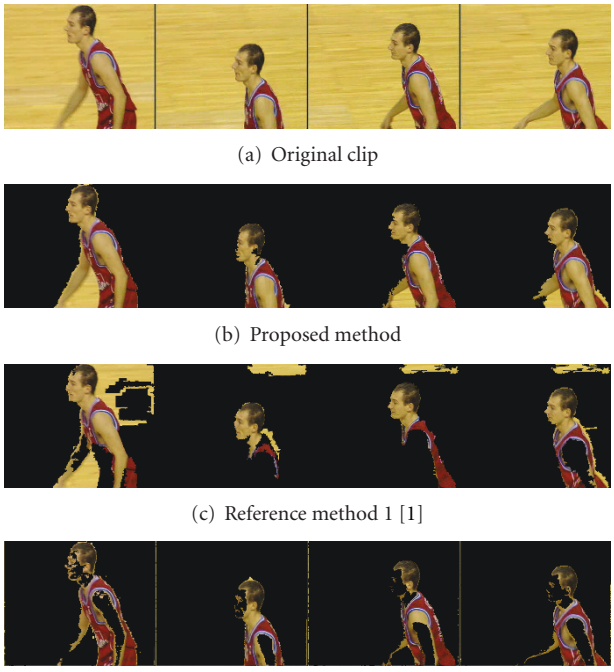
(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 7: Frames 1, 40, 80, and 100 of the Mobile sequence (with GM).

is due mainly to our improved first frame segmentation. This sequence is initially segmented into 6 regions, with 1 region corresponding to the actor, and the background divided into several regions, which are all correctly merged in the histogram- and trajectory-based merging stages. In comparison, reference method 1 initially segments the actor of this clip into several regions, corresponding to the head, shoulders, and torso. Due to the inconsistency of motion between the head and torso of the actor, the reference method's trajectory-based merging stage is unable to correctly merge all of the initially segmented regions.

Figures 6 and 7 present results for the Harbor and Mobile test sequences. These sequences contain complex backgrounds with a moving camera, which present difficulties for both reference methods. The complex background of the Mobile sequence (the background is the wall paper, e.g., everything except the train, the wagon, the ball, and the rails), which contains the same colors as the foreground objects (particularly in the region directly behind the ball) makes it difficult to segment out and track the ball. However, for both sequences, the proposed method exhibits significantly improved performance over both reference methods. The Harbor sequence also demonstrates the improved effectiveness of our trajectory-based merging stage, obtained by accounting for background occlusion. Since the main object in the Harbor sequence is large enough to divide the background into 2 disconnected regions, it is important to account for this when performing region merging (4).



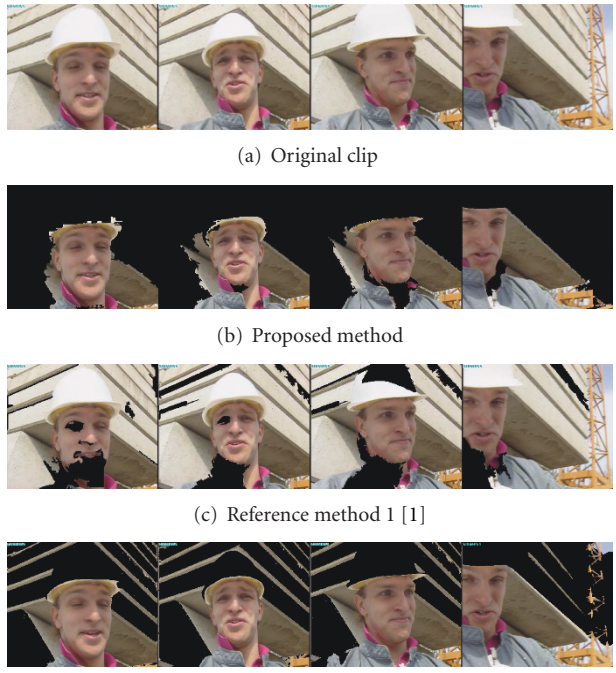
(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 8: Frames 1, 8, 16, and 20 of the Basketball sequence (with GM).



(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 9: Frames 1, 60, 120, and 180 of the Foreman sequence (with GM).

Figure 8 presents results for the Basketball test sequence. This sequence contains rapid object motion along with a fast moving camera. Figure 8 also shows that the proposed method's histogram- and trajectory-based merging stages can still be effective when histograms and trajectories are taken over a relatively short period.

Figures 9 and 10 present results for the Foreman and Carphone sequences. These sequences consist of moving faces with complex backgrounds. Significant improvement over both reference methods can be seen. Due to the complexity of the backgrounds, these clips are initially segmented into many regions. The posttracking region merging stage is important for these clips since they begin the merging process, reducing the chance for error in the final trajectory-based merging stage (3).

Figure 11 presents results for the Miss America sequence. This clip has a simple background and little movement of the foreground object. The Miss America sequence is initially segmented into 2 regions, one corresponding to the actor, and one correspond-ing to the background, so that the merging stages consist of simply distinguishing a single object from the background. Due to the improved initial segmentation, the proposed method performs significantly better than reference method 1 and comparable to reference method 2.

4.3. Objective Results. We use three objective measures [26] (see also [27] for other measures): color contrast, histogram distance, and motion contrast. The color and motion contrast measures trace the object boundary and compare color values and motion vectors inside and outside the object. The histogram distance measures calculate the stability of object histograms throughout the clip.

The color and motion contrast measures are calculated by first tracing the object contour and then drawing a set of normal lines at equally spaced locations across the object boundary. The points on either side of these normal lines are selected to be the centers of blocks inside and outside of the object. In this way, a set of sample blocks containing pixels on either side of the object boundary are constructed. The pixels inside these blocks are used to calculate the object's color contrast:

$$0 \le d_{\text{color}}(t) = 1 - \frac{1}{K_t} \cdot \sum_{i=1}^{K_t} \delta_{\text{color}}(t \mid i) \le 1, \tag{8}$$

where

$$\delta_{\text{color}}(t \mid i) = \frac{\left\| C_O^i(t) - C_I^i(t) \right\|}{\sqrt{3 \cdot 255^2}}. \tag{9}$$

K_t is the total number of normal lines used to calculate the blocks inside and outside the object in frame t. $C_o^i(t)$ and $C_I^i(t)$ are the average color values for the 3 by 3 blocks on the outside and inside of each normal line.

The motion contrast across the object boundary is calculated using the same set of sample blocks inside and outside of the object:

$$0 \le d_{\text{motion}}(t) = 1 - \frac{\sum_{i=1}^{K_t} \delta_{\text{motion}}(ti)}{\sum_{i=1}^{K_t} w_i} \le 1, \tag{10}$$

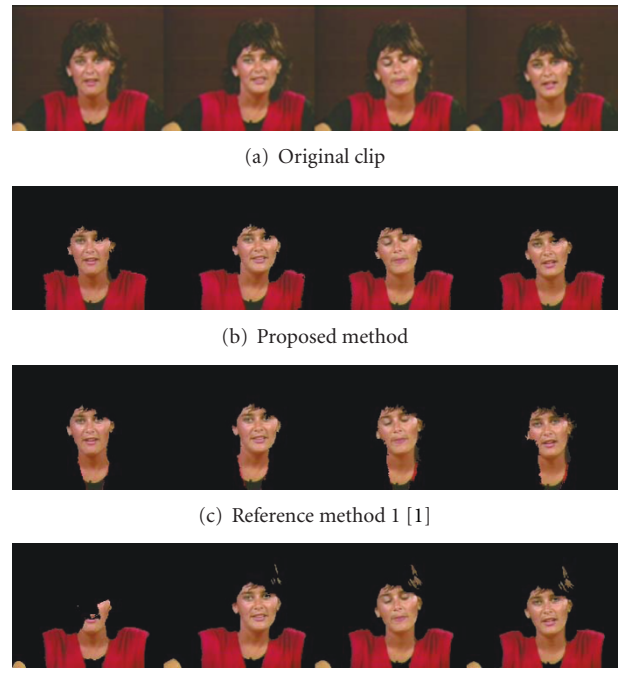
(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 10: Frames 19, 38, 57, and 95 of the Carphone sequence (without GM).



(a) Original clip

(b) Proposed method

(c) Reference method 1 [1]

(d) Reference method 2 [2]

Figure 11: Frames 1, 90, 120, and 150 of Miss America sequence (no GM).

where

$$\delta_{\text{motion}}(t \mid i) = \left(1 - \exp\left(-\frac{\left|\mathbf{v}_O^i(t) - \mathbf{v}_I^i(t)\right|}{\sigma^2}\right)\right) \cdot w_i. \tag{11}$$

$\mathbf{v}_O^i(t)$ and $\mathbf{v}_I^i(t)$ are the average values of the motion vectors in the sample blocks outside and inside the object boundary. The weighting term w_i is calculated as

$$0 \le w_i = R\left(\mathbf{v}_O^i(t)\right) \cdot R\left(\mathbf{v}_I^i(t)\right) \le 1, \tag{12}$$

where

$$R\left(\mathbf{v}^i(t)\right) = \exp\left(-\frac{\left|\mathbf{v}^i(t) - \mathbf{b}^i(t+1)\right|^2}{2\sigma_m^2}\right) \cdot \exp\left(-\frac{\left|c(p^i \mid t) - c(p^i + \mathbf{v}^i(t \mid t+1)\right|^2}{\sigma_c^2}\right). \tag{13}$$

The term $\mathbf{b}^i(t+1)$ is the backwards motion vector in frame $t+1$ at the location $c(p^i + \mathbf{v}^i(t))$. In this way, the motion reliability term $R(\mathbf{v}^i(t))$ estimates the reliability of the motion vectors at each point by measuring the similarity of the backward and forward motion vectors and the difference in pixel intensities at the estimated displacements.

The histogram distance measure for each object is determined by calculating the object's χ^2 histogram distance between each frame and the first frame:

$$\begin{aligned} 0 \le w_i &= \chi^2(H_t, H_{\text{ref}}) \\ &= \frac{1}{(N_{H_t} + N_{H_{\text{ref}}})} \cdot \sum_b \frac{\left(r_1 \cdot H_{R_i}(b) - r_1 \cdot H_{R_j}(b)\right)^2}{\left(H_{R_i}(b) + H_{R_j}(b)\right)} \\ &\le 1, \end{aligned} \tag{14}$$

where H_t is the histogram for the current frame, and H_{ref} is the histogram for the first frame. The normalization factors r_1, r_2, N_{H_t}, and $N_{H_{\text{ref}}}$ are defined as

$$\begin{aligned} r_1 &= \sqrt{\frac{N_{H_{\text{ref}}}}{N_{H_t}}}, \\ r_2 &= \frac{1}{r_1}, \\ N_{H_t} &= \sum_b H_t(j), \\ N_{H_{\text{ref}}} &= \sum_b H_{\text{ref}}(j), \end{aligned} \tag{15}$$

where N_{H_t} and $N_{H_{\text{ref}}}$ are the sizes of the current and reference histograms.

Sample objective measures are presented in Figures 12 and 13. In these graphs, lower normalized values of the color and motion contrast measures indicate more accurate

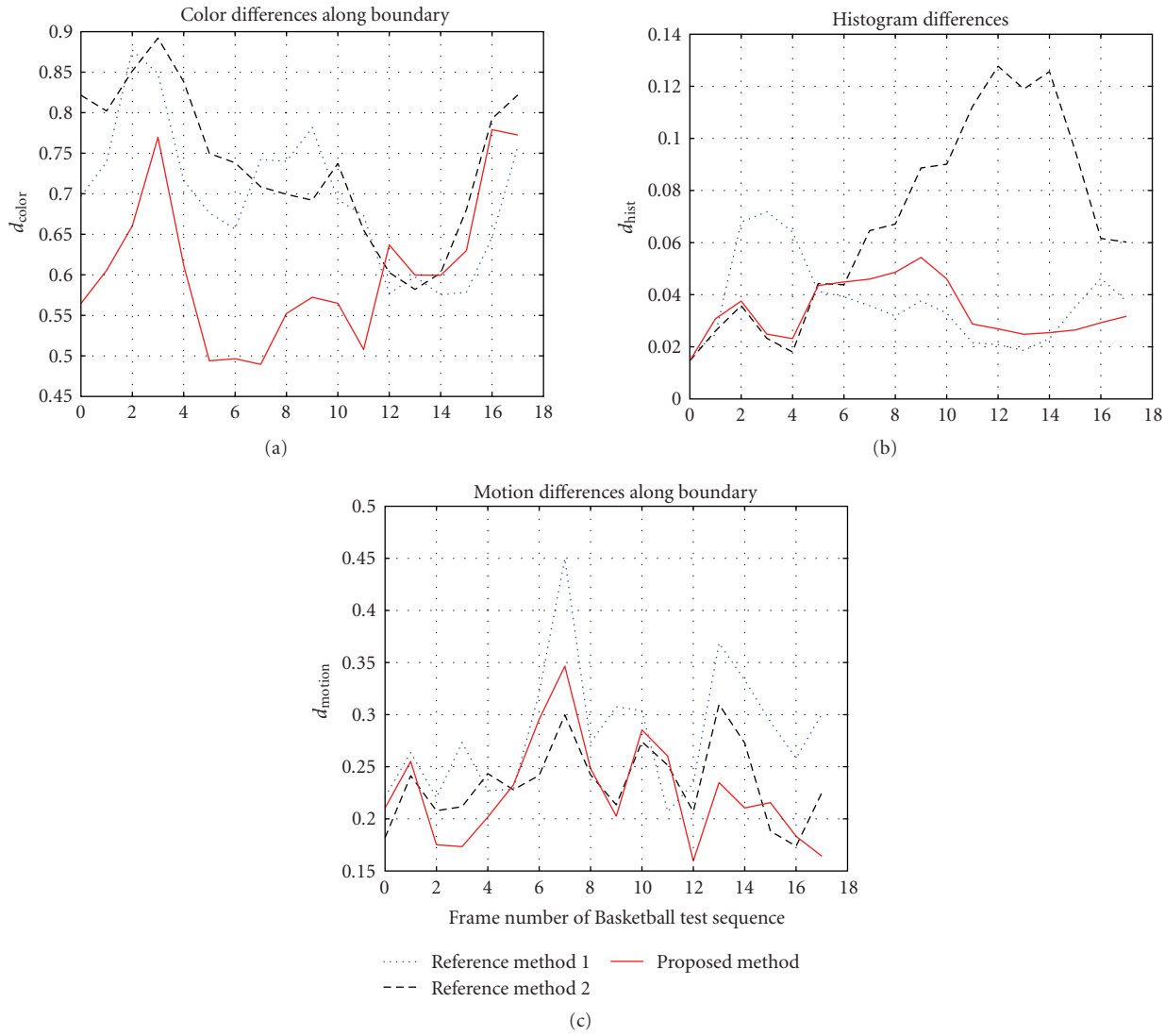


Figure 12: Sample objective results for the basketball sequence. "Reference method 1" is [1], and "Reference method 2" is [2].

segmentation, and lower values of the histogram distance measure means that the object histogram is more stable over the clip, indicating better object tracking. Note that objective measures can be misleading without the context of the subjective results. For example, a poorly segmented object that happens to line up with sharp motion boundaries can score better on the motion objective measure. Similarly, a poorly segmented region that is well tracked can score well on the histogram difference measure. In the case of the Basketball sequence it can clearly be seen that the objective measures confirm the improved subjective performance of the proposed method over both reference methods. Improvement (lower values) can be seen in each of the color, histogram, and motion measures, indicating improved accuracy and stability of the proposed segmentation method. In the case of the Harbor sequence, the proposed method shows better (lower) histogram and color difference. The motion difference measure shows that, for some frames, the proposed method has higher values, but when we compare subjective results to the objective results (Figure 13), we note that subjectively it gives superior results. Significantly better segmentation of moving regions is achieved.

5. Conclusion

The proposed method combines initial segmentation, object tracking, histogram-based object enhancement, and region merging and introduces a number of improvement in using them concurrently. These include reducing oversegmentation of the first frame, using segmentation quality measures to enhance object accuracy, merging tracked regions based on histograms, and accounting for background occlusion.

Our approach proceeds through the following steps: (a) an initial segmentation that incorporates color and motion variance into an existing region clustering scheme; (b) the addition of a histogram distance and motion-variance-based merging stage to reduce oversegmentation of the first frame; (c) segmentation-quality-driven object

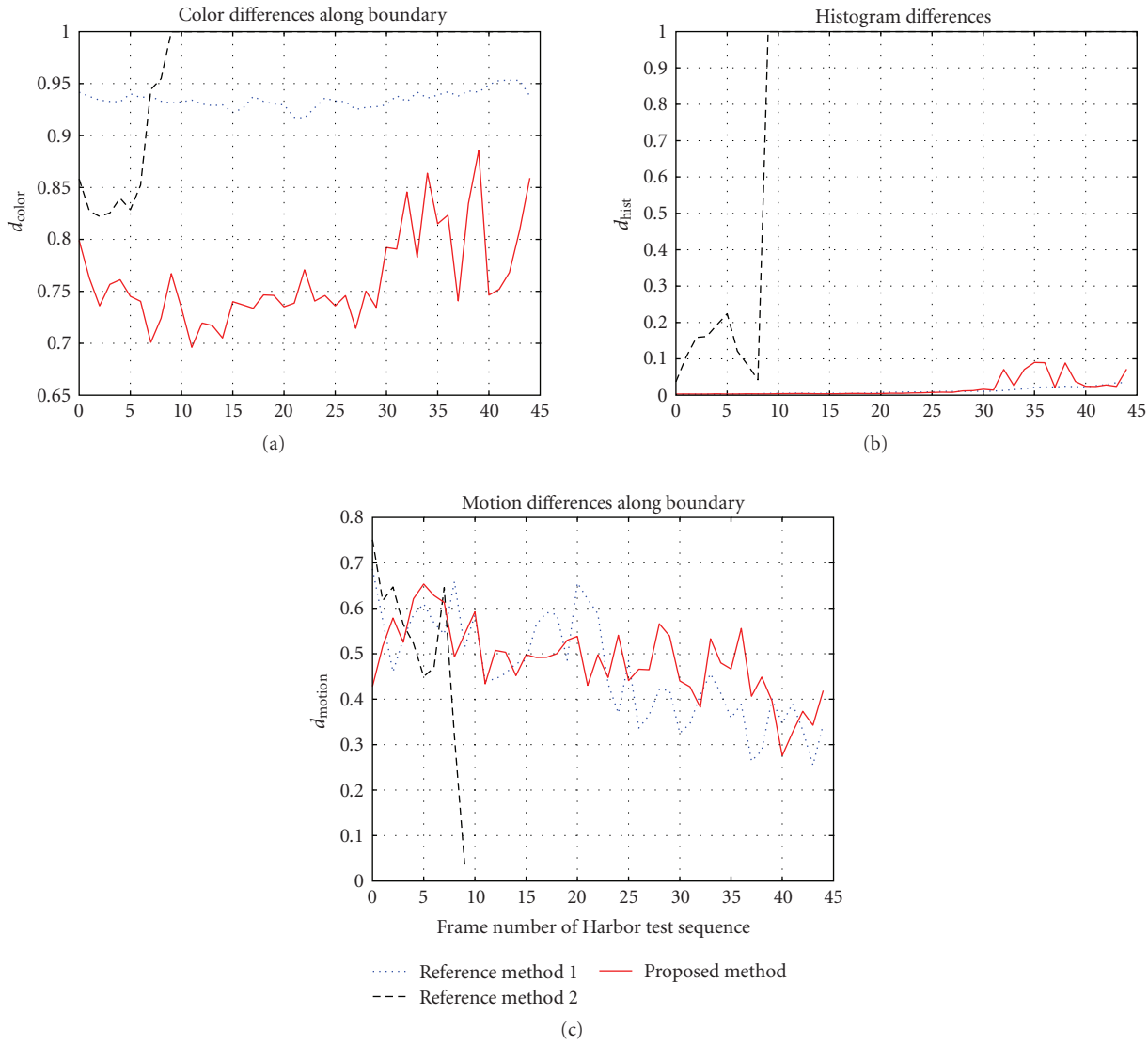


Figure 13: Sample objective results for the Harbor sequence. "Reference method 1" is [1], and "Reference method 2" is [2].

enhancement, where a set of segmentation measures taken while tracking objects are used to improve the accuracy of object boundaries; (d) merging tracked objects based on cumulative histograms gathered throughout the video clip; (e) trajectory-based merging that has been extended to handle partial occlusion and isolated regions.

Our approach (a) is unsupervized, (b) is applicable to a variety of video clips, (c) segments videos with or without global motion, (d) segments objects which are moving in some frames, but stationary in others, (e) segments objects which are nonhomogeneous in color or motion at the frame level, and (f) segments the video into regions that correspond to the main objects being captured in the video. Experimental results demonstrate that our algorithm shows significantly improved performance over two well-recognized reference methods [1, 2].

Video object segmentation remains a challenging topic in video processing. Possible extensions to the proposed approach include improved handling of heavy object occlusion as well as mechanisms to deal with object splitting and merging. These two issues present problems for many segmentation methods, and designing methods to deal with them is an active area of research in video processing. Another area where improvement is possible is in the execution speed of the algorithm. For example, by reducing the number of frames used in the trajectory-based merging stage, significant improvement in execution speed may be attainable. As can be seen in some figures (e.g., Figure 8), an accurate segmentation can sometimes be achieved by examining object trajectories over a relatively small number of frames. By selectively applying the trajectory-based merging to subsegments of longer clips, accurate segmentations may be attainable at reduced computational complexity. The challenge to this approach is determining when to apply it and which subsegments to choose. Finally, improvement may also be obtained by making algorithm parameters more

adaptable to video content. For example, clips with global motion might have different optimal settings for certain thresholds than clips without.

Acknowledgments

This work was supported in part by the Natural Science and Engineering Research Council (NSERC) of Canada and the "Ministere du Developpement Economique de l'Innovation et de l'Exportation (MDEIE) of Gouvernement du Quebec." The authors also wish to express their gratitude to the reviewers for their helpful and constructive comments.

References

[1] V. Mezaris, I. Kompatsiaris, and M. G. Strintzis, "Video object segmentation using bayes-based temporal tracking and trajectory-based region merging," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 782–795, 2004.

[2] S. Khan and M. Shah, "Object based segmentation of video using color, motion and spatial information," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 2, pp. 746–751, Kauai, Hawaii, USA, December 2001.

[3] J. Y. A. Wang and E. H. Adelson, "Representing moving hands with layers," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 625–638, 1994.

[4] T. Darrell and A. P. Pentland, "Cooperative robust estimation using layers of support," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 474–487, 1995.

[5] J. McQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematics Statistics and Probability*, vol. 1, pp. 281–296, 1967.

[6] T. Gevers, "Robust segmentation and tracking of colored objects in video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 776–781, 2004.

[7] P. E. Eren, Y. Altunbasak, and A. M. Tekalp, "Region-based affine motion segmentation using color information," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '97)*, vol. 4, pp. 3005–3008, Munich, Germany, April 1997.

[8] Y.-P. Tsai, C.-C. Lai, Y.-P. Hung, and Z.-C. Shih, "A Bayesian approach to video object segmentation via merging 3-D watershed volumes," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 1, pp. 175–180, 2005.

[9] H. Xu, A. A. Younis, and M. R. Kabuka, "Automatic moving object extraction for content-based applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 796–812, 2004.

[10] T. Lindeberg, "Scale-space for discrete signals," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 3, pp. 234–254, 1990.

[11] S. X. Yu, "Segmentation using multiscale cues," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 1, pp. 247–254, Washington, DC, USA, June 2004.

[12] B.-G. Kim and D.-J. Park, "Unsupervised video object segmentation and tracking based on new edge features," *Pattern Recognition Letters*, vol. 25, no. 15, pp. 1731–1742, 2004.

[13] D. Wang, "Unsupervised video segmentation based on watersheds and temporal tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 539–546, 1998.

[14] K. Ryan, A. Amer, and L. Gagnon, "Video object segmentation based on object enhancement and region merging," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 273–276, Toronto, Canada, July 2006.

[15] S. Wang, X. Wang, and H. Chen, "A stereo video segmentation algorithm combining disparity map and frame difference," in *Proceedings of the 3rd International Conference on Intelligent System and Knowledge Engineering (ISKE '08)*, pp. 1121–1124, Xiamen, China, November 2008.

[16] Q. Wu, P. Boulanger, and W. F. Bischof, "Robust real-time bilayer video segmentation using infrared video," in *Proceedings of the 5th Canadian Conference on Computer and Robot Vision (CRV '08)*, pp. 87–94, Windsor, Canada, May 2008.

[17] E. D. R. M. Luque, D. Lopez-Rodriguez, and E. Palomo, "A dipolar competitive neural network for video segmentation," in *Proceedings of the 11th Ibero-American Conference on Artificial Intelligence (IBERAMIA '08)*, vol. 5290, pp. 103–112, Lisbon, Portugal, October 2008.

[18] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[19] Z. Cernekova, N. Nikolaidis, and I. Pitas, "Temporal video segmentation by graph partitioning," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '06)*, vol. 2, pp. 209–212, Toulouse, France, May 2006.

[20] C. E. Erdem, "Video object segmentation and tracking using region-based statistics," *Signal Processing: Image Communication*, vol. 22, no. 10, pp. 891–905, 2007.

[21] S. Lefèvre, J. Holler, and N. Vincent, "A review of real-time segmentation of uncompressed video sequences for content-based search and retrieval," *Real-Time Imaging*, vol. 9, no. 1, pp. 73–98, 2003.

[22] Y.-J. Zhang, *Advances in Image and Video Segmentation*, IRM Press, Hershey, Pa, USA, 2006.

[23] L. Bergen and F. Meyer, "A novel approach to depth ordering in monocular image sequences," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 2, pp. 536–541, Hilton Head Island, SC, USA, June 2000.

[24] L. R. Williams and D. W. Jacobs, "Local parallel computation of stochastic completion fields," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '96)*, pp. 161–168, San Francisco, Calif, USA, June 1996.

[25] J. Malik, S. Belongie, J. Shi, and T. Leung, "Textons, contours and regions: cue integration in image segmentation," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, vol. 2, pp. 918–925, September 1999.

[26] Ç. E. Erdem, B. Sankur, and A. M. Tekalp, "Performance measures for video object segmentation and tracking," *IEEE Transactions on Image Processing*, vol. 13, no. 7, pp. 937–951, 2004.

[27] C. E. Erdem, A. M. Tekalp, and B. Sankur, "Video object tracking with feedback of performance measures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 4, pp. 310–324, 2003.