Quadrilateral-based region segmentation for tracking

Nelson H. C. Yung
Hing Y. Chung
Paul Y. S. Cheung
University of Hong Kong
Department of Electrical & Electronic Engineering
Pokfulam Road
Hong Kong SAR, China
E-mail: hychung@eee.hku.hk

Abstract. We propose a novel quadrilateral based region segmentation method that is favorable for object tracking. Instead of using groups of pixels or regular blocks, it uses groups of connected quadrilaterals to represent regions. The proposed method derives the vertices of each quadrilateral from the edge map using the concept of center of masses. By merging the quadrilaterals, regions can be represented. The proposed method offers better data reduction than pixelwise region representation and better boundary approximation than block-based segmentation methods. Experimental results show that it generates a more reasonable region map, which is more suitable for object tracking, and a smaller number of regions than the seeded region growing, *K*-means clustering, and constrained gravitational clustering methods. *© 2002 Society of Photo-Optical Instrumentation Engineers.* [DOI: 10.1117/1.1511747]

Subject terms: segmentation; quadrilateral; region extraction; object tracking.

Paper 010479 received Dec. 27, 2001; revised manuscript received Apr. 1, 2002; accepted for publication Apr. 10, 2002.

1 Introduction

Image segmentation is a fundamental step in most image analysis applications.[1] It aims to partition the content of an image, according to specific features of interests, into distinct regions to make high-level tasks such as object tracking, recognition, and scene interpretation possible.[2] The elevation of segmentation techniques into prominence is partly due to the increasing demand for supporting a large number of object-based multimedia research and services,[3] and partly due to the need for more sophisticated computer vision applications in robotics, autonomous vehicles, and visual surveillance.[4] In essence, multimedia applications such as MPEG4 and MPEG7 require video images to be represented as objects rather than rectangular arrays of pixels.[5] To achieve the required interactivity and indexing functions in these standards, regions would have to be segmented from individual images from which objects are constructed. In computer vision applications, such as outdoor traffic surveillance, for example, traffic count and speed can be estimated if moving vehicles are segmented from the background and tracked over a number of frames. In general, it is argued that image segmentation should no longer remain at the stage of just extracting regions from a single image, but should be used to deduce objects over an image sequence.[6]

While considerable research and progress have been made in the area of image segmentation over an image sequence, the robustness and generality of these algorithms for an image sequence have not been fully established.[2] Many existing segmentation algorithms partition an image into groups of pixels that have similar features but these groups of pixels must be further interpreted before they can be used to produce object information. When an image sequence is concerned, the geometrical variations of an object due to its motion, the camera motion, and its relationship with other objects within view can be extremely complex. The interpretation of the groups of pixels into objects presents a challenge, and the robustness of the segmentation algorithm becomes critical.

In view of this, we are motivated to develop a robust image segmentation method that can extract complex objects in scenarios where none of the aforementioned geometrical variations are constrained. The concept of our proposed method is built on a network of quadrilaterals to represent regions. Each region is completely described by a set of quadrilaterals, which approximates the region boundary. These regions are then merged and/or split to generate objects under certain criteria. This approach has two merits: first, quadrilateral representation offers a large data reduction similar to polygon approximation techniques except that it is far more flexible with lower approximation error. Second, interframe tracking of regions such as geometric invariants[7,8] matching, which is capable of correlating regions under different perspective projection, is possible by matching of quadrilaterals. The performance of the proposed method is evaluated and compared with seeded region growing[9] (SRG), *K*-means clustering[10] (KMC) and constrained gravitational Clustering[11] (CGC). It is found that the average performance of the proposed method is better than that of the other three. Moreover, it appears that the segmented images generated by the proposed method are suitable for object tracking.

This paper is organized as follows. Section 2 gives an overview of existing segmentation methods, Sec. 3 details proposed segmentation method, and Sec. 4 presents the evaluation results.

2 Segmentation Overview

There are three main types of segmentation: spatial, temporal, and spatiotemporal methods.

2.1 *Spatial Segmentation*

Segmentation techniques in this category utilize only spatial information to partition the image into distinct regions of interests. The pixels within a region usually share similar features such as pixel intensity, color values, texture, gradient, edges, histogram modes, or certain other statistical qualities. To extract the regions, one or more of these features are first evaluated at each pixel location. The point at which an abrupt change of feature occurs is then considered as a region boundary point.

Classical segmentation techniques like K-means,[9] SRG (Ref. 10) belong to this category. Although state of the art spatial segmentation algorithms can generate reasonably good object boundaries in a subjective sense, the computation requirement is usually high, relying on algorithms such as edge thinning, gap filling, contour fitting, or polygonal approximation. However, as exact object boundaries are usually not required in tracking, the computation invested thus may not be worthwhile. Therefore, in applications involving an image sequence, spatial segmentation techniques that generate just enough information for tracking is preferred.

In spite of the previously noted drawbacks, tracking algorithms[12,13] that utilize classical spatial segmentation methods have been proposed. Park et al.[12,13] proposed an image to be segmented into homogenous regions using the block-based maximum *a posteriori* (MAP) image segmentation,[14] where user intervention is required to cope with overmerging or undermerging.

2.2 *Temporal Segmentation*

Segmentation techniques in this category utilize only temporal information across frames to extract regions. Similar to spatial segmentation, pixels having common features are grouped into regions. In this case, interframe features such as difference in pixel values, difference in histograms, and difference in motion information are used rather than intraframe features. As a result, only moving regions are extracted, which is not the case in spatial segmentation. The drawback is that objects with slow motion may be treated as background and are not segmented, and occluded moving objects may be treated as one.

Temporal segmentation has been mainly used for tracking in the past.[15–20] In the Cretual et al.[15] work, successive images were first filtered by a local spatial average of image intensities to remove noise, from which moving region locations were then located by taking the thresholded difference between successive images. The merit of this approach is that computation requirement involved is relatively low. However, detection of regions may become unreliable when illumination changes occurred. On the other hand, Oron[20] proposed to obtain the image difference in the frequency domain using the phase of successive frames to equalize the lighting conditions of the images while McCandless[19] proposed to segment an image from the optical flow. Nine sequential frames are involved in one segmentation process for camera motion compensation, spatiotemporal smoothing of images, and optical flow computations. The region with optical flow vectors having small magnitudes are regarded as background and discarded. Those regions with optical flow vectors that pass a consistency check and predictive test are treated as moving regions. In theory, these regions may be merged to form objects, but further interpretation is required.

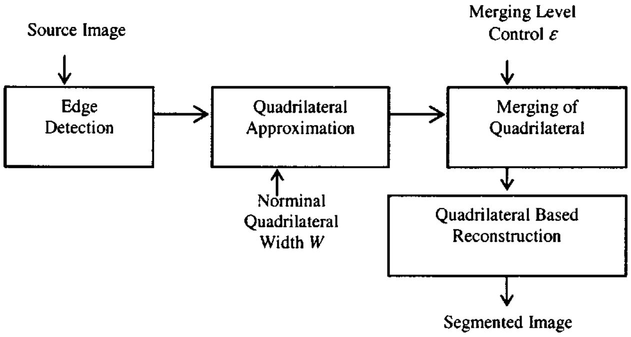

Source Image
Merging Level Control ε
Edge Detection
Quadrilateral Approximation
Merging of Quadrilateral
Norminal Quadrilateral Width W
Quadrilateral Based Reconstruction
Segmented Image

Fig. 1 Proposed algorithm.

2.3 *Spatiotemporal Segmentation*

Spatiotemporal segmentation is a hybrid of the two classes of techniques discussed in the preceding sections. It uses both intra- and interframe features to extract regions. Spatial segmentation can be carried out first, followed by temporal segmentation to refine the regions extracted. By doing so, overmerging or undermerging problems found in spatial segmentation may be alleviated and slowly moving regions may also be extracted. However, a good balance between the two techniques must be found to obtain satisfactory segmentation results.

For instance, Deng and Manjunath[21] proposed that an image sequence be processed in consecutive groups of frames, in which the first frame in each group is spatially segmented first. The regions in the other frames within the group are then obtained through a six-parameter 2-D affine transformation to project each region in the first frame. This is not a strong spatiotemporal segmentation scheme since spatial and temporal information are used together within a group but not within a frame. However, this scheme does simplify the intragroup region tracking as it is inherently done during the segmentation and the tracking problem is reduced to intergroup region tracking. In Kompatsiaris and Strintz's work,[22] a stronger spatiotemporal segmentation scheme was applied. The segmentation is based on the K-means algorithm. The authors introduced the concept of connectivity constraint, which takes the spatial and motion information in a number of consecutive frames to resolves the problem of unconnected regions resulted in conventional K-means algorithm.

In this paper, we emphasized spatial segmentation. Both spatial and temporal information are useful for segmentation in image sequences. However, we believe spatial segmentation should be considered first and temporal segmentation should be introduced as refinement for the results of spatial segmentation and good segmentation results cannot be obtained without a good spatial segmentation as the first step.

3 Proposed Segmentation Method

Our proposed method is built on a network of quadrilaterals to represent regions. Each region is completely described by a set of quadrilaterals, which approximates the region boundary. Figure 1 presents a block diagram of the pro-

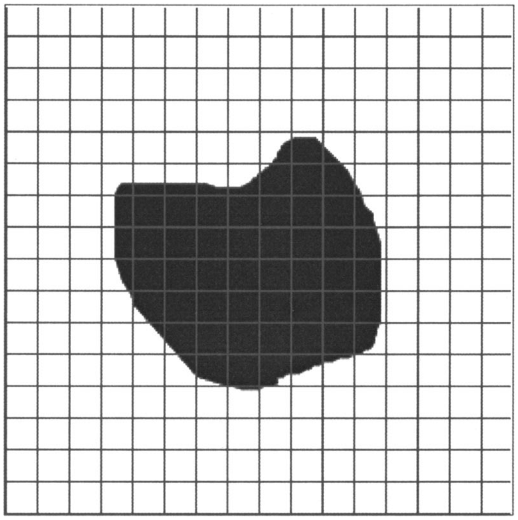


Fig. 2 Image frame divided into square blocks.

posed segmentation method. To build a network of quadrilaterals, edge detection is first used to construct an edge map. There are plenty of edge detection methods that can be used for this purpose. Gradient operation is simple and efficient but more susceptible to noise, while the Canny operator[23] can generate good and precise edges but requires convolution with a Gaussian filter, which demands more computation. In our case, we employ the Sobel edge detector to derive the edge map for its simplicity and reasonable edge map. Note that we do not require a very precise edge map when it is used in the quadrilateral approximation. In quadrilateral approximation, the concept of center of masses is adopted. The center of masses is chosen because it is less sensitive to the noise that appears in the edge map. Quadrilateral approximation is followed by quadrilateral merging. In quadrilateral merging, quadrilaterals are merged based on a criterion to form different regions. The segmented image is finally constructed from the regions described by merged quadrilaterals.

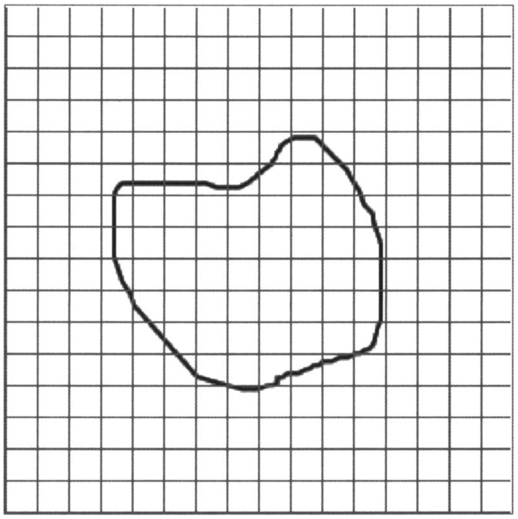


Fig. 3 Edge map overlapped with divided image frame.

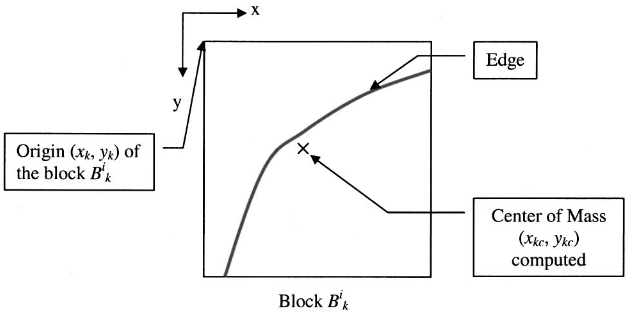

x
y
Edge
Origin (x_k, y_k) of the block B^i_k
Center of Mass (x_{kc}, y_{kc}) computed
Block B^i_k

Fig. 4 Center of mass extracted from the edge pixels in a block.

3.1 *Quadrilateral Approximation*

Assume that each object or region can be represented by a group of four-connected (left, right, top, and bottom) quadrilaterals. To approximate the regions by quadrilaterals, we first divide the image frame into square blocks of arbitrary width W, as shown in Fig. 2, where the object is shaded and the block division is shown as a grid overlay on the object. For each block, a representative point is computed by considering the edge map, as shown in Fig. 3. Those edge values within the block are treated as masses and the center of mass of the block is considered as the representative point.

In essence, for the k'th block B_k^i in the i'th image frame I_i, let (x_k, y_k) denote the origin of B_k^i. Let (x_{kc}, y_{kc}) denotes the location of the center of mass of B_k^i with respect to B_k^i's origin, then we have

$$(x_{kc},y_{kc})=\left[x_k+\frac{\Sigma_{(p,q)\in B_k^i} p\times G(p,q)}{\Sigma_{(p,q)\in B_k^i} G(p,q)},\; y_k+\frac{\Sigma_{(p,q)\in B_k^i} q\times G(p,q)}{\Sigma_{(p,q)\in B_k^i} G(p,q)}\right], \tag{1}$$

where $G(p,q)$ represents the magnitude of the Sobel detector output. In this equation, only pixels with nonzero $G(p,q)$ contribute to the calculation of the center of mass.

After a center of mass is determined for each block, as shown in Fig. 4, we construct the quadrilaterals by connecting up the center of masses, in four-connected directions. Each center of mass in a block would only connect to the center of masses in the left, top, bottom, and right neighboring blocks. Figure 5 shows an example of the quadrilaterals approximated and represented by the thick lines.

3.1.1 *Further consideration on quadrilateral approximation*

It is observed that if the edge points in a block approximate a straight line, the center of mass is usually sufficiently close to the edge points. However, that might not be the case when corner points exist in a block. As shown in Fig. 6, one center of mass is not sufficient to approximate the edge feature of the block in the center and refinement is therefore necessary. To do this, a criterion must be deter-

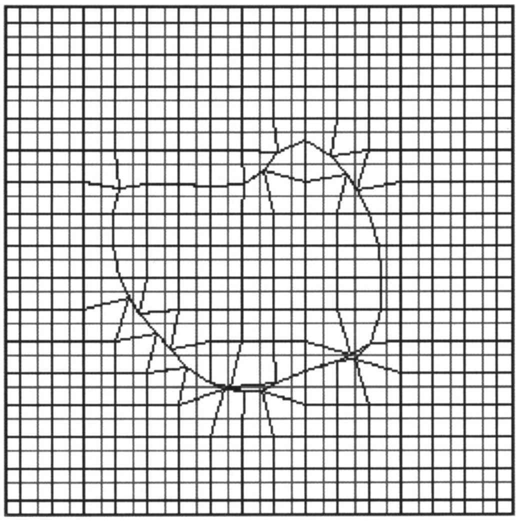


Fig. 5 Quadrilaterals approximated for the edge map depicted in Fig. 3.

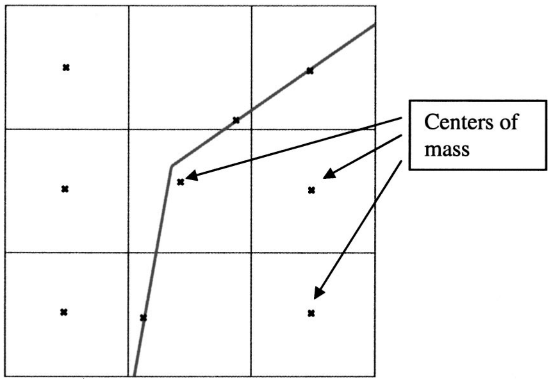


Fig. 6 Center of mass not on the edge.

mined to test whether a block needs further refinement. In this case, we employ the concept of correlation to see whether or not the edge inside a block is a straight line.

Let X and Y be random variables representing the x coordinate and y coordinate of the edge points in B_k^i, respectively. Let $\mathrm{var}(B_k^i)_X = E(X^2) - E(X)^2$ denote the variance of the random variable X; $\mathrm{var}(B_k^i)_{XY} = E(XY) - E(X)E(Y)$ denote the covariance of the random variables X and Y, and $\mathrm{corr}(B_k^i)_{XY} = \mathrm{var}_{XY}/[(\mathrm{var}_X \cdot \mathrm{var}_Y)]^{1/2}$ denote the correlation of the random variables X and Y. By treating the x and y coordinates of the edge points within the block B_k^i as the random variables X and Y, respectively, the correlation of the edge points within B_k^i can be obtained as follows:

$$\mathrm{corr}(B_k^i)_{XY} = \frac{\dfrac{\Sigma_{(p,q)\in B_k^i} p\times q\times G(p,q)}{\Sigma_{(p,q)\in B_k^i} G(p,q)} - \dfrac{\Sigma_{(p,q)\in B_k^i} p\times G(p,q)\Sigma_{(p,q)\in B_k^i} q\times G(p,q)}{[\Sigma_{(p,q)\in B_k^i} G(p,q)]^2}}{\left\{\left[\dfrac{\Sigma_{(p,q)\in B_k^i} p\times p\times G(p,q)}{\Sigma_{(p,q)\in B_k^i} G(p,q)} - \dfrac{\Sigma_{(p,q)\in B_k^i} p\times G(p,q)\Sigma_{(p,q)\in B_k^i} p\times G(p,q)}{[\Sigma_{(p,q)\in B_k^i} G(p,q)]^2}\right]\left[\dfrac{\Sigma_{(p,q)\in B_k^i} q\times q\times G(p,q)}{\Sigma_{(p,q)\in B_k^i} G(p,q)} - \dfrac{\Sigma_{(p,q)\in B_k^i} q\times G(p,q)\Sigma_{(p,q)\in B_k^i} q\times G(p,q)}{[\Sigma_{(p,q)\in B_k^i} G(p,q)]^2}\right]\right\}^{1/2}}. \tag{2}$$

From Eq. (2), the following condition can be tested for each block:

Condition. If $|\mathrm{corr}(B_k^i)|$ is close to 1, then most of edge points lie on a straight line, otherwise the edge points do not lie on a straight line.

Consider Fig. 6 again, except for the block in the center, each block contains either no edge points or edge points that form a straight line. For the block in the center, the second part of the condition is satisfied, and therefore, refinement is required. To perform the refinement, we first divide the block into four equal subblocks, as shown in Fig. 7. The center of mass of each subblock is then calculated. Second, Eq. (2) is applied to each subblock to see if further refinement is necessary. This process of subblock division and recalculation of center of mass is repeated until the subblocks have correlation values sufficiently close to one or no further subdivision is possible (zero width and height).

After the refinement, quadrilaterals can be constructed by interconnecting the centers of mass. We adopt the four-connectivity scheme, i.e., a quadrilateral is made up of the four nearest centers of mass. To avoid a one-to-many mapping, it is further conditioned that a line between two centers of mass can only be constructed from a lower level subblock to a higher level subblock, but not the other way round. For example, in Fig. 8 a line can be drawn from B_1^0 to B_2^0 as they are on the same hierarchical level. However, line can be drawn from B_{51}^0 to B_2^0 but not from B_2^0 to B_{51}^0.

Compared with the case without subdivision, subblock refinement results in more accurate centers of mass for representing the edge points, and therefore, quadrilaterals can

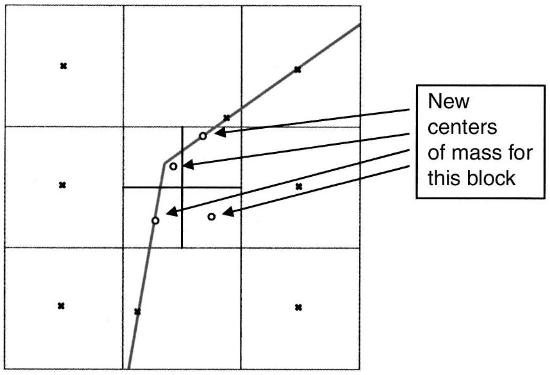
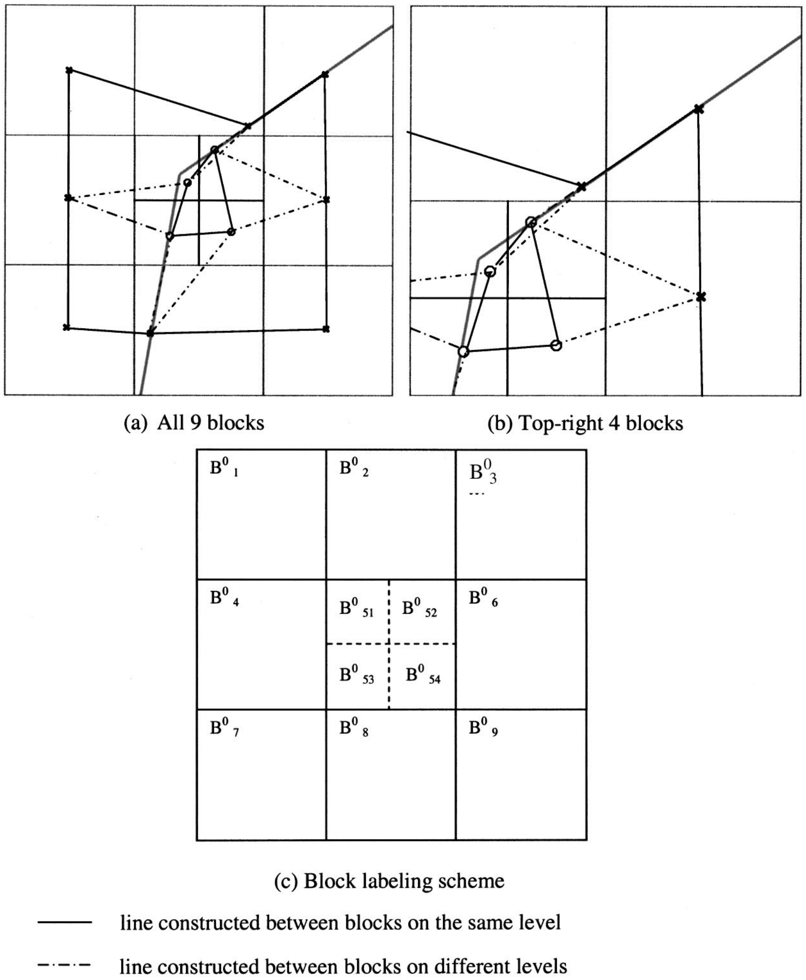


Fig. 7 Dividing the central block into four rectangular subblocks.

be approximated more exactly. It is noted that in some cases, instead of a quadrilateral, a triangle results after the construction. These triangles are treated as degenerated quadrilaterals with two of the vertices merged together.

On the other hand, it is possible that the quadrilaterals approximated may overlap with their neighbors. As depicted in Fig. 8, the quadrilateral formed by B^0_{51}, B^0_{52}, B^0_{54}, and B^0_{53} is overlapped with that formed by B^0_{51}, B^0_{52}, and B^0_2. This occurs when the block labeling is not in the same clockwise direction as the vertices of the quadrilateral. For example, the vertices of the quadrilateral formed by the centers of masses of B^0_{51}, B^0_{52}, and B^0_2 in Fig. 8 are in clockwise direction, whereas the block labels B^0_{51}, B^0_{52}, and B^0_2 are in an anticlockwise direction. To avoid overlapping of quadrilaterals, it is necessary to ensure that a quadrilateral is formed in the same direction as the blocks concerned. Whenever a quadrilateral is not formed in the proper direction, the block associated with that quadrilateral with the highest level of hierarchy is subdivided. For example, in Fig. 8, the block that requires further division is B^0_2. After the subdivision, if the quadrilaterals thus formed are still formed in a wrong direction, the process is repeated. Figure 9 shows the quadrilaterals formed after B^0_2 is subdivided.

By using the preceding scheme, the quadrilaterals approximated for the top-left corner of the object in Fig. 3 are depicted in Fig. 10. The quadrilaterals are drawn in solid lines, whereas the object edge outline is drawn in a thick solid line.


B^0_1
B^0_2
B^0_3
...
B^0_4
B^0_{51}
B^0_{52}
B^0_{53}
B^0_{54}
B^0_6
B^0_7
B^0_8
B^0_9
line constructed between blocks on the same level
line constructed between blocks on different levels

(a) All 9 blocks

(b) Top-right 4 blocks

(c) Block labeling scheme

Fig. 8 Quadrilateral formation.

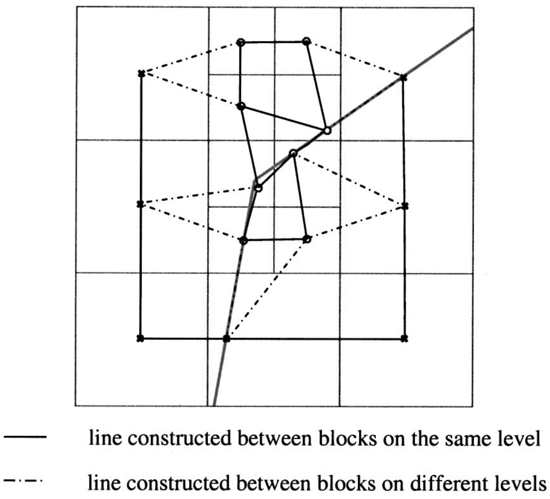

line constructed between blocks on the same level
line constructed between blocks on different levels

Fig. 9 Quadrilaterals approximated with location of center of mass restricted.

3.2 *Merging of Quadrilaterals*

To obtain regions from the quadrilaterals approximated, we must merge neighboring quadrilaterals that are similar to each other. In the proposed method, two neighboring quadrilaterals with similar color attributes are merged together. For each quadrilateral, we first evaluate the mean of each color component (R, G, B) of the pixels within the quadrilateral by considering two quadrilaterals $ABCD$ and $CDEF$, as depicted in Fig. 11.

For any color component c, where $c \in \{\mathrm{R,G,B}\}$ and R, G, B are the red, green and blue color components, respectively, let M^c_{ABCD} be the mean of the color component c of the pixels within $ABCD$, M^c_{DCFE} be the mean of the color component c of the pixels within $DCFE$, and $D(c) = M^c_{ABCD} - M^c_{DCFE}$. The two quadrilaterals are merged if

$$D^2(R)+D^2(G)+D^2(B)<\varepsilon, \tag{3}$$

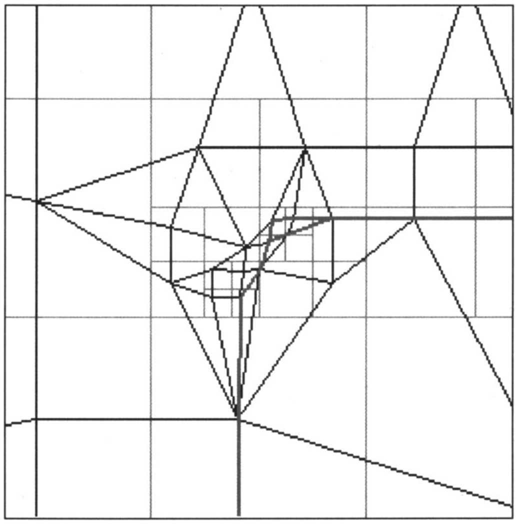


Fig. 10 Quadrilateral approximation of part of the object in Fig. 3.

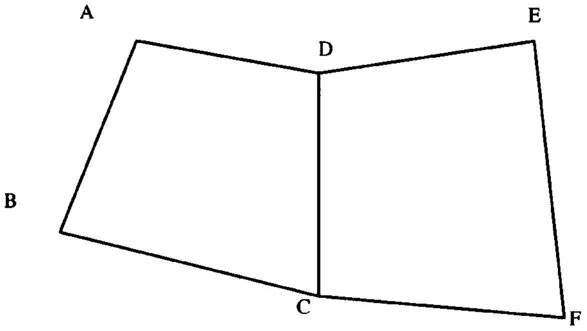


Fig. 11 Two neighboring quadrilaterals.

where ε is the merging threshold. The same merging criterion is used for vertically neighboring quadrilaterals.

The merging process is essentially a region growing process. First, ε is assigned to a value $\varepsilon_{\mathrm{init}}$. A new region label is then assigned to each quadrilateral that satisfies Eq. (3) with all its four-connected neighbors. Then for each labeled quadrilateral, the same test is performed to see if its four-connected neighbors can be merged with it. This process is repeated until no further labeling is possible. If unlabeled quadrilaterals still exist, a new region label is required. To do this, ε is then increase by $\varepsilon_{\mathrm{init}}$. By doing so, either new labeling or further merging is possible. This procedure is repeated until all the quadrilaterals in the image frame are labeled.

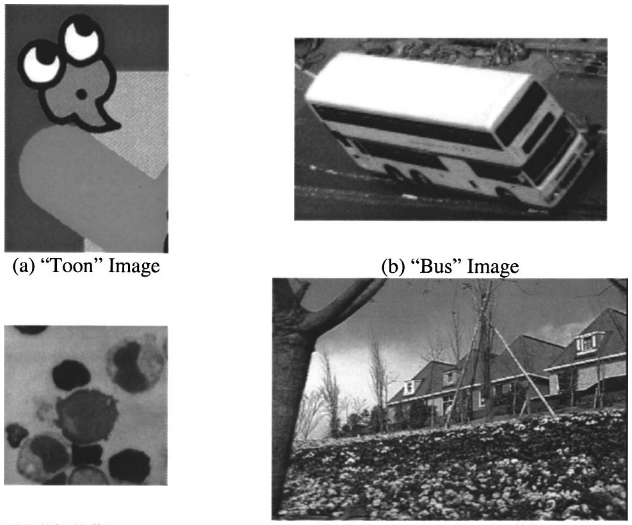
4 Evaluation

To see how well the proposed method performs, we compare its performance with three other algorithms, KMC, SRG, and CGC. Both objective measurement and subject inspective were considered. For the objective measurement there are many different evaluation methods. The one proposed by Liu and Yang[24] is adopted in our work because it can be used to evaluate real images without requiring a reference image. However, note also that this method gives merely a broad and general indication, where subjective inspection should not be undervalued. The evaluation function L is defined as

$$L[h(x,y)]=\frac{\sqrt{R}}{1000A_T}\sum_{i=1}^{R}\frac{e_i^2}{\sqrt{A_i}}, \tag{4}$$

where $h(x,y)$ is the segmented image; R is the total number of regions resulted in $h(x,y)$; A_i is the number of pixels in the i'th region, A_T is the total number of pixels in the image; and e_i^2 is the color error of region i, which is defined as the sum of Euclidean distance of the color vector between the input image and the segmented image for each pixel in the region.

The subjective inspection is based on three criteria: smoothness of the boundaries, boundary correctness, and ease of object tracking. Four test images were used in the evaluation, namely, ''Toon,'' ''Bus,'' ''Cells,'' and ''Flower and Garden,'' as shown in Fig. 12. ''Toon'' is an image of a high contrast simple cartoon figure with small amount of noise. This image presents a lower degree of segmentation difficulty. ''Cells'' is an image of some purple-dyed white



(a) “Toon” Image

(b) “Bus” Image

(c) “Cells” Image

(d) “Flower and Garden” Image

Fig. 12 Test images.

cells taken from a slide under a microscope. The image is slightly out of focus and its contrast is normal. It presents a higher degree of segmentation difficulty than the ‘‘toon’’ image. The ‘‘Bus’’ image is part of a photo taken from an overhead camera looking down at a busy road. The double-decker bus is mainly white with a faint pattern on the side. The boundaries look fuzzy, but the front right side is clearly visible. The background includes part of a road where construction work can be seen at the top of the image and broken road markings near the bottom. This image is considered to be more difficult to segment when compared with ‘‘Cells.’’ Finally, ‘‘Flower and Garden’’ is an image that consists of complex background with a blurred tree, houses, and flower patch. This image is considered to be the most difficult among the four, but it is a more generic type of image.

Table 1 Smallest L of each algorithm.

	KMC	SRG	CGC	Proposed
“Toon”				
L	0.780	0.131	0.339	0.105
R	157	153	918	25
Rank	4	2	3	1
“Cells”				
L	0.430	0.241	0.248	0.260
R	196	316	723	156
Rank	4	1	2	3
“Bus”				
L	3.380	1.019	1.298	0.949
R	664	166	640	111
Rank	4	2	3	1
“Flower and Garden”				
L	19.042	17.522	3.585	7.834
R	2121	3815	7373	956
Rank	4	3	1	2

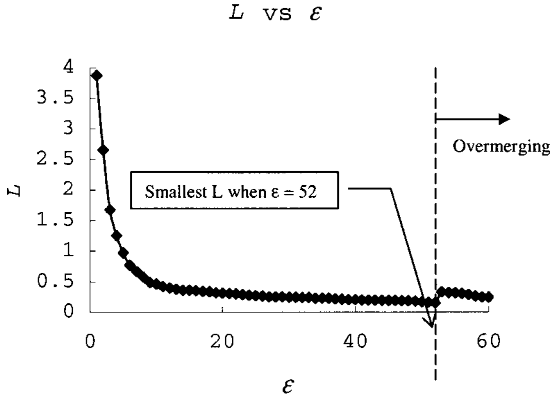


Fig. 13 L versus ε for the “Toon” image.

For each algorithm, its parameters were varied over a range where their segmented results and L for the two images were determined. Table 1 shows the best objective performance (L) obtained in each case and the corresponding number of regions (R) of each algorithm. Ranking of these algorithms are also given.

In particular, the parameter to be varied in the proposed algorithm is ε. Its value is varied over a range until a smallest L is obtained. Figure 13 shows the L value plotted against ε, while Fig. 14 shows the R value plotted against ε for the ‘‘Toon’’ image.

As shown in Table 1, our method gives the smallest L value for the ‘‘Toon’’ and ‘‘Bus’’ images, second best L value for ‘‘Flower and Garden,’’ and ranks third for ‘‘Cells.’’ KMC, on the other hand, always gives the largest L value for the four test images. SRG gives the smallest L value for ‘‘Cells,’’ second best L value for ‘‘Toon’’ and ‘‘Bus,’’ and ranks third for ‘‘Flower and Garden.’’ CGC gives the smallest L value for ‘‘Flower and Garden,’’ second best L value for ‘‘Cells,’’ and ranks third for both ‘‘Toon’’ and ‘‘Bus.’’ We can see that none of the methods always gives the smallest L value. Objectively, it seems that the proposed method is the best with SRG follows closely behind since the proposed method has two first rank compared with one first rank for SRG. But when the number of segmented regions is concerned, our method gives the smallest number of regions for all the test images. This shows that our method may be more suitable in semantic

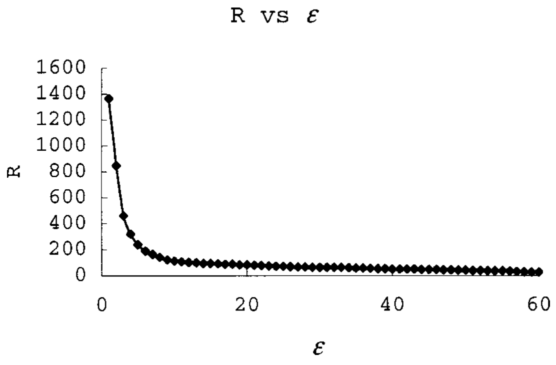
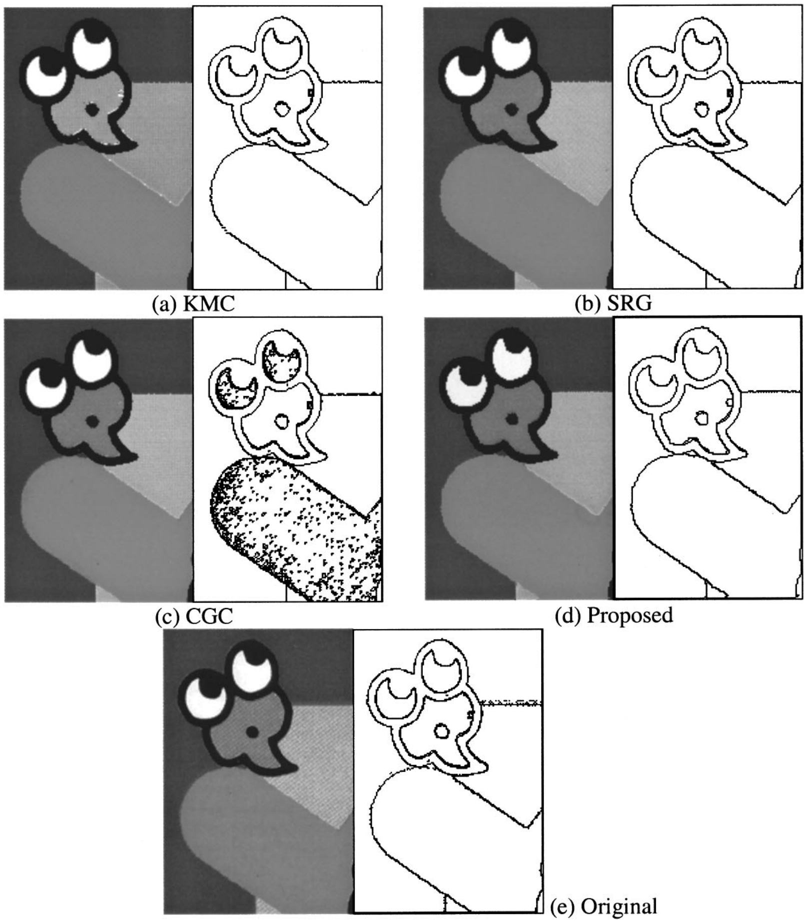


Fig. 14 R versus ε for the “Toon” image.



(a) KMC

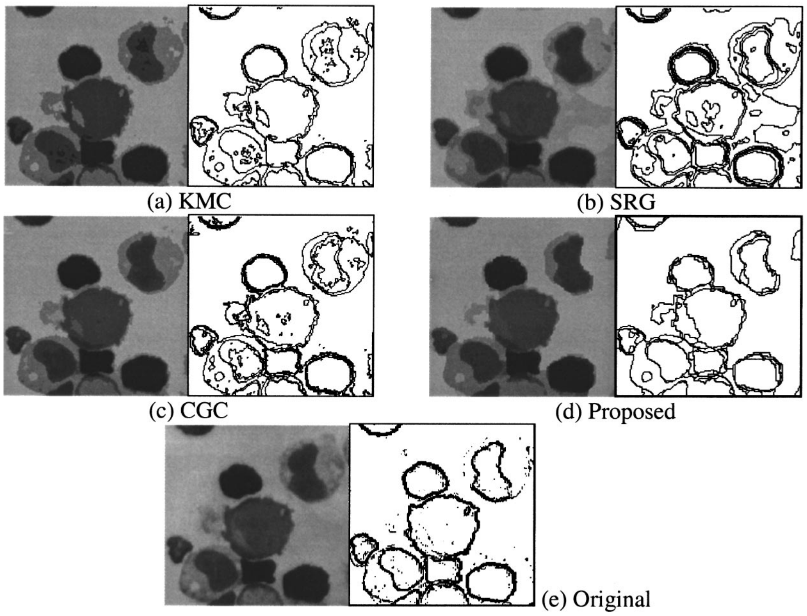
(b) SRG

(c) CGC

(d) Proposed

(e) Original

Fig. 15 Segmented images and region maps of ''Toon'' according to smallest *L*.



(a) KMC

(b) SRG

(c) CGC

(d) Proposed

(e) Original

Fig. 16 Segmented images and region maps of ''Cells'' according to smallest *L*.

video object tracking, as smaller number of regions means less computation power is required in object tracking. However, a smaller number of regions might not necessarily imply better segmentation results as it might indicate the possibility of overmerging. Therefore, it is necessary to compare the segmented images subjectively.

Figures 15–18 depict the segmented images for each algorithm resulted from the best L. In Fig. 15, we can see that the segmented images of ''Toon'' of the four methods are roughly the same with smooth and correct boundaries. However, when we consider the region map, CGC have numerous tiny regions. This would make further processing necessary before object tracking can be performed. In their original algorithm,[11] a region merging method was adopted to eliminate these tiny regions. In contrast, the other three seems to contain reasonable regions segmented from the region maps. We see that for this simple image, KMS, SRG, and the proposed method have almost the same visual performance. But when the number of regions is of concern, KMS and SRG have the problem of undermerging as they resulted in more regions than the proposed method.

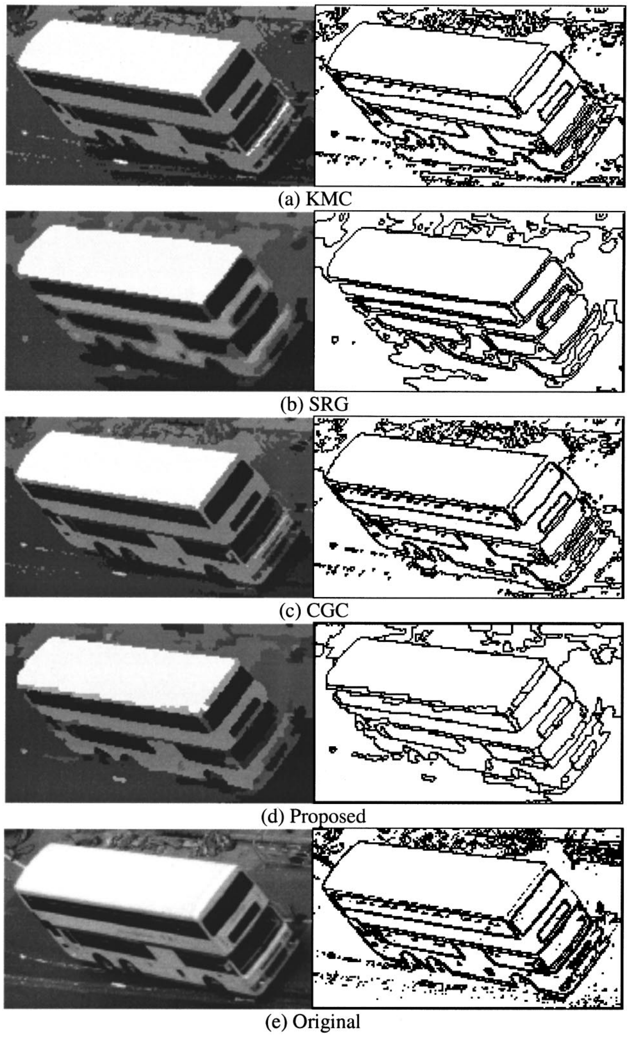
In Fig. 16, KMC, CGC, and the proposed method have the background correctly segmented. SRG suffers from the undermerging problem as the background is not correctly segmented and multiple regions appear at the object boundaries. KMC and CGC perform more or less the same except CGC has more tiny regions. The proposed method, although it resulted in the smallest number of regions, appears to have overmerging problem around the top-right and middle-bottom cells. When the boundary smoothness and correctness are of concern, the four methods are roughly the same with a little smoother boundary for KMC and little rougher boundary for the proposed method.

In Fig. 17, KMC and CGC perform roughly the same with smooth and correct boundaries. The proposed method exhibits rougher boundaries but it contains fewer regions and the key features of the bus are correctly identified. SRG suffers from overmerging this time, which is noticeable at the lower front side of the bus.

In Fig. 18, both KMC and CGC suffer the problem of the color shift, which is common in clustering types of segmentation methods. On the other hand, SRG and the proposed method resulted in a color that more closely resembles the original image. KMC, CGC, and SRG all suffer from the problem of undermerging. This is noticeable at the roof of the houses for KMC and CGC and around the tree for SRG. The proposed method does not seem to have this problem in those two regions. The boundaries extracted by the four methods appear to be smooth enough.

The subjective evaluation is summarized in Table 2, where complexity refers to the number of meaningful regions formed. Note that a segmented image with low complexity is more favorable for tracking.

From the objective and subjective evaluations, we observe that no single method outperforms the others in all the four test images. When the average performance is considered, the proposed method appears to be superior to the other three except for the ''cells'' image. This is because the proposed method generates reasonably smooth and correct boundaries with no noticeable overmerging and undermerging problem and results in the smallest number of regions for the other three test images. Besides, the boundary of a region is represented by the vertices of quadrilaterals instead of every boundary pixels, resulting in better data reduction, which is more favorable for object or region tracking. In this case, further description techniques are required for the other three methods to achieve a similar data reduction effect.


(a) KMC

(b) SRG

(c) CGC

(d) Proposed

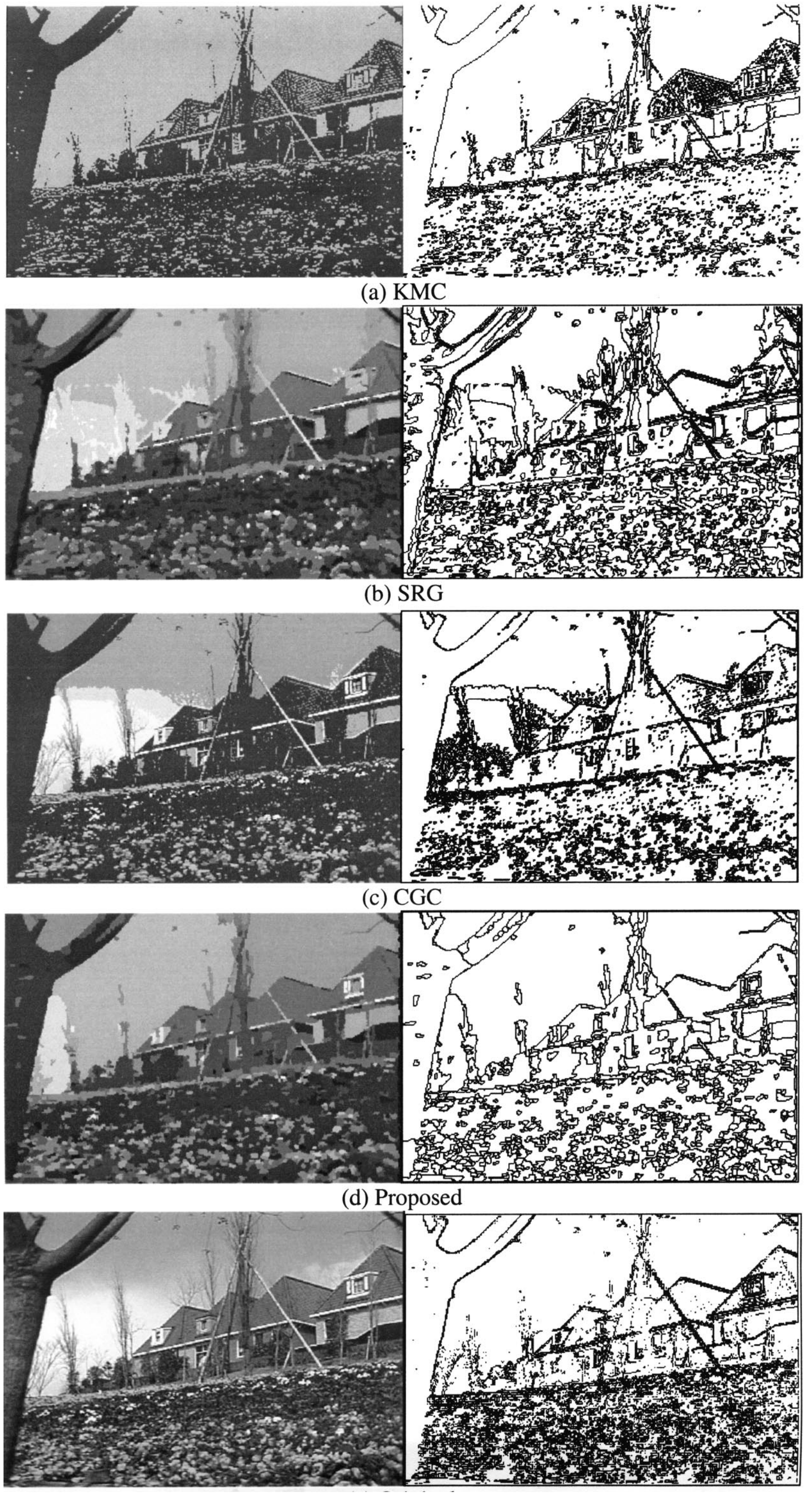
(e) Original

Fig. 17 Segmented images and region maps of ''Bus'' according to smallest L.

When tracking is of concern, geometric invariants[23,24] can be used to match quadrilaterals across different frames. By doing so, the linkage of quadrilaterals across frames can be built up and thus correlation of regions across frames is possible. To match quadrilaterals across frames, many methods[25–27] proposed in literature can be applied.

5 Conclusion

We have presented a new segmentation method based on the concept of quadrilateral approximation that has favorable properties for object tracking in generic video sequences. The new method was evaluated and compared both objectively and subjectively against three other algorithms, namely, KMC, SRG, and CGC. We observed that the proposed method appears to perform best in terms of the average performance of both objective and subjective



(a) KMC

(b) SRG

(c) CGC

(d) Proposed

Fig. 18 Segmented images and region maps of ''Flower and Garden'' according to smallest *L*.

Table 2 Subjective Evaluation.

	KMC	SRG	CGC	Proposed
"Toon"				
Smoothness	Good	Good	Good	Good
Correctness	Good	Good	Fair	Good
Complexity	Low	Low	High	Low
"Cells"				
Smoothness	Good	Good	Good	Fair
Correctness	Good	Poor	Good	Fair
Complexity	Moderate	High	Moderate	Moderate
"Bus"				
Smoothness	Good	Fair	Fair	Fair
Correctness	Fair	Poor	Fair	Good
Complexity	Moderate	High	Moderate	Low
"Flower and Garden"				
Smoothness	Good	Good	Good	Good
Correctness	Fair	Poor	Fair	Good
Complexity	High	High	Moderate	Low

measures. The regions obtained by the proposed method are approximated by quadrilaterals, which results in data reduction, and the region representation is ready to be used in tracking. Besides, the proposed method also tends to generate a reasonable number of regions with low complexity. By these observations, it seems that the proposed method is more suitable for tracking when compared with the other three algorithms. Future directions would be focused on two areas: investigate the merging criterion so that parameterless segmentation is possible and develop a object-tracking methodology based on the proposed segmentation method.

References

1. G. Iannizzotto and L. Vita, ''Fast and accurate edge-based segmentation with no contour smoothing in 2-D real images,'' *IEEE Trans. Image Process.* **9**(7), 1232–1237 (2000).
2. W. Y. Ma and B. S. Manjunath, ''Edge flow: a technique for boundary detection and image segmentation,'' *IEEE Trans. Image Process.* **9**(8), 1375–1388 (2000).
3. P. Salembier and F. Marques, ''Region-based representations of image and video: segmentation tools for multimedia services,'' *IEEE Trans. CASVT* **9**(8), 1147–1169 (1999).
4. S. Yamamoto, Y. Mae, Y. Shirai, and J. Miura, ''Realtime multiple object tracking based on optical flows,'' in *Proc. IEEE Conf. Robotics and Automation*, pp. 2328–2333 (1995).
5. R. Castagno, T. Ebrahimi, and M. Kunt, ''Video segmentation based on multiple features for interactive multimedia applications,'' *IEEE Trans. CASVT* **8**(5), 562–571 (1998).
6. D. Zhong and S. F. Chang, ''An integrated approach for content-based video object segmentation and retrieval,'' *IEEE Trans. CASVT* **9**(8), 1259–1268 (1999).
7. I. Weiss, ''Geometric invariants and object recognition,'' *Int. J. Comput. Vis.* **10**(3), 207–231 (1993).
8. J. Munday and A. Zisserman, ''Introduction—towards a new framework for vision,'' in *Geometric Invariance in Machine Vision*, MIT Press, Cambridge, MA (1992).
9. R. Adams and L. Bischof, ''Seeded region growing,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(6), 641–647 (1994).
10. J. A. Hartigan, *Clustering Algorithms*, John Wiley & Sons, Canada (1975).
11. H. C. Yung and H. S. Lai, ''Segmentation of color images based on the gravitational clustering concept,'' *Opt. Eng.* **37**(3), 989–1000 (Mar. 1998).
12. D. K. Park, H. S. Yoon, and C. S. Won, ''Fast object tracking in digital video,'' in *Proc. ICCE2000*, pp. 356–357, IEEE (2000).
13. D. K. Park, H. S. Yoon, and C. S. Won, ''Fast object tracking in digital video,'' *IEEE Trans. Consum. Electron.* **46**(3), 785–790 (2000).
14. C. S. Won, ''A block-based MAP segmentation for image compression,'' *IEEE Trans. CSVT* **8**(5), 592–601 (1998).
15. A. Cretual, F. Chaumette, and P. Bouthemy, ''Complex object tracking by visual servoing based on 2D image motion,'' *IEEE Proc. Pattern Recog.* **2**, 1251–1254 (1998).
16. G. L. Foresti and C. S. Regazzoni, ''Localisation and tracking of multiple unknown objects in real environments,'' *Electron. Lett.* **31**(5), 355–356 (1995).
17. Y. Altunbasak, R. Oten, R. J. P. de Figueiredo, ''Simultaneous object segmentation, multiple object tracking and alpha map generation,'' in *Proc. ICIP97*, Vol. 1, pp. 69–72, IEEE (1997).
18. N. K. Paragios and R. Deriche, ''A PDE-based level-set approach for detection and tracking of moving objects,'' in *Proc. ICCV98*, pp. 1139–1145, IEEE (1998).
19. J. W. McCandless, ''Detection of aircraft in video sequences using a predictive optical flow algorithm,'' *Opt. Eng.* **38**(3), 523–530 (Mar. 1999).
20. E. Oron, ''Motion estimation and image difference for multi-object tracking,'' in *Proc. IEEE Aerospace Conf. 1999*, Vol. 4, pp. 401–409 (1999).
21. Y. Deng and B. S. Manjunath, ''Spatio-temporal relationships and video object extraction,'' in *Conf. Record of the 32nd Asilomar Conf. on Signals, Systems & Computers*, Vol. 1, pp. 895–899 (1998).
22. I. Kompatsiaris and M. G. Strintz, ''Spatiotemporal segmentation and tracking of objects for visualization of videoconference image sequences,'' *IEEE Trans. CSVT* **10**(8), 1388–1402 (2000).
23. J. F. Canny, ''A computational approach to edge detection,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **8**, 679–698 (1986).
24. J. Liu and Y. H. Yang, ''Multiresolution color image segmentation,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(7), 689–700 (1994).
25. U. Uenohara and T. Kanade, ''Geometric invariants for verification in 3-D object tracking,'' in *Proc. IEEE IROS96*, pp. 785–790 (1996).
26. S. W. Lee, B. J. You, and G. D. Hager, ''Model-based 3-D object tracking using projective invariance,'' *IEEE Proc. Robot. Automat.* **2**, 1589–1594 (May 1999).
27. G. Lei, ''Recognition of planar objects in 3-D space from single perspective views using cross ratio,'' *IEEE Trans. Rob. Autom.* **6**(4), 432–437 (1990).

Nelson H. C. Yung received his BSc and PhD degrees in 1982 and 1985, respectively, from the University of Newcastle-Upon-Tyne, where he was a lecturer from 1985 until 1990. From 1990 to 1993, he was a senior research scientist with the Department of Defence, Australia. He became an associate professor with the University of Hong Kong (HKU) in late 1993. He leads a research team in digital image processing and intelligent transportation systems, and leads a number of research collaborative projects with various companies and organizations in the HKSAR, China, Japan, and Australia. He is the founding director of the Laboratory for Intelligent Transportation Systems Research at HKU. Dr. Yung has coauthored a computer vision book, and has published over 100 journal and conference papers in the areas of digital image processing, parallel algorithms, visual traffic surveillance, and autonomous vehicle navigation. He serves as reviewer for the *IEEE Transactions of SMC, Signal Processing, IEE Pt. G, SPIE Optical Engineering, International Journal of Vehicle Mechanics and Mobility, HKIE proceedings, Microprocessors and Microsystems*, and *Robotics and Autonomous Systems Journal.* He is a member of the advisory panel of the *ITS Strategy Review*, Transport Department, HKSAR, a council member of ITSHK, and chair of the Computer Division, International Institute for Critical Infrastructures. He is a chartered electrical engineer, a member of the HKIE and IEE and a Senior member of the IEEE. He was regional secretary of the IEEE Asia-Pacific Region for 1995 to 1997. His biography is published in *Who's Who in the World* (Marquis, USA) since 1998. He was a Croucher Scholar and his team won the Silver Award from the Hong Kong Electronic Industry Association for Outstanding Innovation and Technology Product, 2000, for the MOVER (Mobile and Online Vending EnableR) solution.

Hing Y. Chung received his BEng degree in computer engineering and his MPhil degree in 1997 and 1999, respectively, from the Department of Electrical and Electronic Engineering, University of Hong Kong, where he is currently working toward his PhD degree. His research interests include digital image processing and video technology.

Paul Y. S. Cheung received his BSc degree in engineering and his PhD degree from the Imperial College of Science and Technology, University of London, in 1973 and 1978 respectively. He joined the University of Hong Kong in 1980, where he served as associated dean of engineering from 1991 to 1994 and Dean of the faculty of engineering from 1994 to 2000. In October 2000, he took leave from the university and joined PCCW, the main telecom company in Hong Kong, as senior vice-president in technology. He was elected as the Asia Pacific region director of IEEE in 1995 to 1996 and was the first from Hong Kong to serve as the institute's secretary of the IEEE in 1997. He has graduated over 20 research students and published over 70 research papers. His research interests include parallel computer architecture, Internet computing, very large scale integration design, signal processing, and pattern recognition. He is a chartered engineer and a senior member of the IEEE. He was an independent nonexecutive director of PCCW from 1999 to 2000. Since March 2002, he has been seconded to the Hong Kong Government as policy advisor in the Commission on Innovation and Technology.