A User-Assisted Segmentation Method Using Semantic Information

Daehee Kim

Kwangju Institute of Science and Technology
1 Oryong-dong Puk-gu, Kwangju, 500-712, Korea

Yo-Sung Ho

Kwangju Institute of Science and Technology
1 Oryong-dong Puk-gu, Kwangju, 500-712, Korea

***Abstract* – Most automatic segmentation techniques cannot represent individual video object in a single frame accurately. They are somewhat premature to obtain desirable segmentation results from various kinds of image sequences. However, if the user can define video objects or provide semantic information in the first frame in a user-assisted manner, we can obtain improved segmentation results over the following successive frames. The user-assisted semi-automatic approach for video segmentation is more practical in generating VOPs of moving objects. In this paper, we propose a new user-assisted video segmentation algorithm using semantic active contour.**

1. INTRODUCTION

The MPEG-4 standard [1] enables content-based functionalities by introducing the concept of video object planes (VOP's). In order to process the image based on its content, we should segment the image into a set of meaningful objects. Knowledge about the shape of video objects in the scene helps us for better image reconstruction, especially along object boundaries.

Among the conventional approaches for video segmentation, the spatio-temporal segmentation technique is particularly interesting to us because MPEG-4 includes a typical spatio-temporal algorithm in its informative annex [1]. This algorithm extracts edge information by the morphological operation and obtains information of moving objects by the change detection mask.

However, the spatio-temporal segmentation algorithm cannot extract individual video object in a single frame accurately. Besides, this algorithm is somewhat premature to obtain satisfactory segmentation results from various kinds of image sequences, because mathematical model for video objects is not defined adequately and the definition of video objects are quite subjective. If the user can utilize video objects at the first frame in a partially or completely user-assisted manner, as in the active contour algorithm, we can obtain better segmentation results in the succeeding picture frames. User-assisted segmentation is more practical in generation VOP's of moving objects.

An active contour method is one of the user-assisted segmentation approaches which allow user interactivity to obtain the semantic information about arbitrary video objects. In this paper, we propose a new video segmentation algorithm based on active contour algorithm to find the shape of visual objects and to track them.

II. USER-ASSISTED SEGMENTATION

In user-assisted segmentation, the user can provide high-level semantic information for video object segmentation. The user-assisted method is also called the semi-automatic segmentation. A user-assisted segmentation algorithm consists of two parts: intra-frame segmentation and inter-frame segmentation. Fig. 1 shows the overall block diagram of user-assisted segmentation.

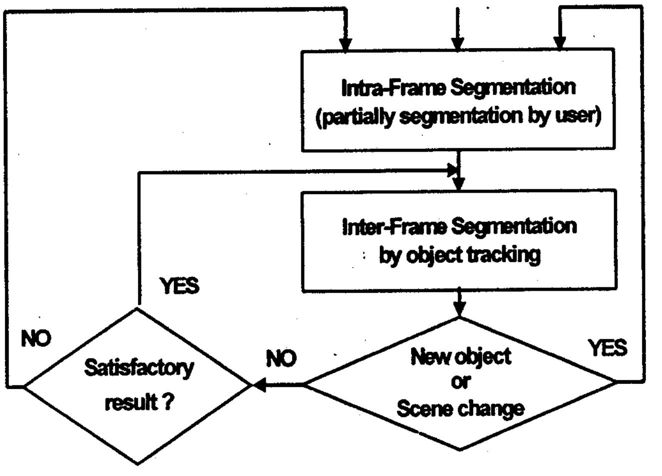


Fig. 1 Block diagram of user-assisted segmentation

In the intra-frame segmentation step, the user can supply semantic information of video objects in the first frame or any subsequent frames which have newly appeared objects of interest. The semantic information of video objects can be supplied by a user's pointing device, such as a PC mouse, in the way that the user initially points video objects of interest around their boundaries. Then, the user can extract predefined video objects with the semantic information by using some adequate methods, such as active contour estimation.

Once predefined video objects are recognized in the intra-frame segmentation stage, the boundary of the predefined video object is transferred to the inter-frame segmentation. In the inter-frame segmentation, the predefined video object is continuously separated from unselected areas through time evolution in the image sequence until unsatisfactory results or new video objects of interest occur, in such case, we go to the intra-frame segmentation in order to find more accurate boundaries or to define new video objects. The inter-frame segmentation step consists of contour tracking and shape fitting.

A. Intra-Frame Segmentation

Basically, the active contour algorithm tries to find an energy-minimizing spline curve from an initial curve drawn by the user. The performance of an active contour algorithm depends on how to define energy functional *E(s)*. The shape of the active contour is controlled by internal force, external force and constraint force. In addition, since the formula itself of the active contour allows the constraint forces, the active contour is proper for user-assisted segmentation. In this paper, we propose a robust active contour algorithm for intra-frame segmentation to define the shape of video object.

Internal force is as a smoothness constraint. External force guides the active contour towards image features. Constrain force allows interactivity in manipulating the active contour. The energy functional *E(s)* is represented as a parametric curve $v(s)=(x(s), y(s))$, where the arc length s is a parameter [2]. A functional of the snake is defined by

$$\begin{aligned} E^{*}_{snake} &= \int_0^1 E_{snake}(v(s))ds \\ &= \int_0^1 \left[E_{\text{int}}(v(s)) + E_{\text{ext}}(v(s)) + E_{\text{con}}(v(s))\right] \quad (1) \end{aligned}$$

where E_{int}, E_{ext} and E_{con} represent at the internal energy of the contour, image force and external constraint force, respectively. The final location of the active contour corresponds to the local minimum of the energy functional.

In order to discretize and simplify Eq. (1), we need to represent the functional of the contour by a finite number of control point s.

$$E^{*}_{snake} = \sum_{i=1}^{n} \lambda_i E_{\text{int}}(v_i) + (1-\lambda_i) E_{\text{ext}}(v_i) \quad (2)$$

where n is the total number of control points on the contour. The internal energy of (1) can be written as

$$E_{\text{int}} = (\alpha(s)|v_s(s)|^2 + \beta(s)|v_{ss}(s)|^2)/2 \quad (3)$$

Eq. (2) can also be discretized as

$$E_{\text{int}}(v_i) = \alpha_i |v_i - v_{i-1}|^2 + \beta_i |v_{i+1} - 2v_i + v_{i-1}|^2 \quad (4)$$

If there is no external energy in this approximation, the contour could converge to only one point. In order to prevent the contour from being shrunk to one point, we define an internal energy as

$$E_{\text{int}} = \frac{1}{l(V)} |v_i - \alpha_i(v_{i-1} + v_{i+1})|^2 \quad (5)$$

where α_i is the shape control parameter. Eq. (5) resolves the contour-shrinking problem from the internal energy [3]. The internal energy can be normalized by the averaged distance $l(V)$ between two successive control points.

In this paper, we define two kinds of external energy functions. For the first one, we use the gradient information

$$E_{grd} = 1 - |n_i^T g(v_i)| \quad (6)$$

where $g_i(v_i)$ is 2×1 gradient vector and $n_i(v_i)$ is the unit normal vector at the control point v_i. Therefore, when the gradient vector has the same direction as the normal vector of the contour, the external energy has the minimum value.

If we use a simple edge detector as in conventional active contour algorithms, it is difficult to obtain satisfactory segmentation results from images of complex background. Therefore, we employ morphological tools to simplify the image and we define the second external energy function by the morphological gradient value of the simplified image.

In order to derive a new morphological energy function, we need to simplify the input images by morphological operations. Besides, this simplification operation suppresses the number of watershed lines. Morphological open-close and close-open by reconstruction filters are employed for image simplification. These filters remove regions that are smaller than the size of the structuring element, while preserving object contours [4].

The spatial gradient of the simplified images is approximated by a morphological gradient operator [5]. The spatial gradient can be used as an input to the watershed algorithm to partition the image into homogeneous intensity regions.

We then apply the watershed algorithm on the gradient values of the image to obtain the object contour. A watershed algorithm can be performed by immersion simulation in discretized grid points [6]. We obtain a new binary image W_{img} by the watershed algorithm. Pixel values on the watershed lines are set to one in W_{img}. We can define a morphological external energy on the binary image W_{img} by

$$E_{\text{int}} = 1 - \sum_{m=-1}^{1} \sum_{n=-1}^{1} W_{\text{img}}(v_{ix} + m)(v_{iy} + n) \quad (7)$$

where v_{ix} and v_{iy} are the components of v_i. Since the thickness of the watershed line is one pixel, we use the local window in Eq. (7).

B. Minimization Process

In order to find a set of control points minimizing Eq. (2) around the initial contour, we decompose the minimization process into n independent stages. In each stage, we include only three neighboring points. This idea initially was proposed under the framework of dynamic programming [7].

Since it is quite difficult to select parameter values λ_i for various situations, we try to find a curve that minimizes the total energy of the active contour.

$$E^{final}_{snake} = \min_{V} \sum_{i=1}^{n} \max\left[E_{\text{int}}(v_i), E_{grd}(v_i), E_{mor}(v_i)\right] \quad (8)$$

This function satisfies the minimax criterion when λ_i=1 or λ_i=0 in Eq. (2). Since the energy at each control point is related to the internal energy, the energy at each control point can be defined by three adjacent control points.

$$E_i(v_i, v_{i+1}, v_{i+2}) = \max[E_{\text{int}}(v_i, v_{i+1}, v_{i+2}), \\ E_{grd}(v_i), E_{mor}(v_i)] \quad (9)$$

We can solve our problem by dynamic programming.

$$\begin{aligned} s_1(v_2, v_3) &= \min_{v_1} E_2(v_1, v_2, v_3) \\ s_2(v_3, v_4) &= \min_{v_2}(s_1(v_2, v_3) + E_3(v_2, v_3, v_4)) \\ &\vdots \\ s_{n-2}(v_{n-1}, v_n) &= \min_{v_{n-2}}(s_{n-3}(v_{n-2}, v_{n-1}) \\ &\quad + E_{n-1}(v_{n-2}, v_{n-1}, v_n)) \end{aligned} \quad (10)$$

C. Inter-Frame Segmentation

In this paper, we also propose an object tracking method for predefined video objects over the successive frames. Its framework consists of contour tracking and shape fitting.

In general, successive frames of image sequences are highly correlated. Thus, an object tracking method can be applied in the consecutive frames after inter-frame segmentation. In order to track the contour of the previous video object, we first select control points of the contour. We project these control points into the current frame by motion estimation and motion compensation. Motion estimation should be performed in point by point along the control points of the previous video object.

We perform motion estimation using a block-matching algorithm. The estimated motion vector is considered as the motion vector of the center point of the block. In order to obtain reliable results, we set four neighboring blocks and find a motion vector corresponding to each block [8]. The center of each neighboring block is two pixels distant from the center of the original block. We take the median value of five motion vectors as motion vector corresponding to the control point. This operation for overlapped motion estimation can provide better prediction.

Since block-matching algorithm considers the foreground and the background equally, it may cause problems when the uncovered background region appears due to the object movement. In order to avoid this problem, we exclude the background region during the motion estimation operation. Thus, a simple block-matching algorithm needs to be modified to accommodate this situation.

In this paper, we locate the center of each square block at the control points in the previous frame and ignore the background region. The size of the square block is 7×7 and the search range is from –8 to 8. This arbitrary shaped block (ASB) is employed to estimate the motion vector. In the motion estimation step, we find the motion vector to match the arbitrary shape in the previous frame to that in the current frame.

Our contour estimation scheme employs five arbitrary shaped blocks. We take the median value of these motion vectors. The matching criterion is the mean absolute difference (MAD).

$$MAD = \frac{1}{M} \sum_{(x,y)\in A} \left| F_{k-1}(x,y) - F_k(x + MV_x, y + MV_y) \right| \quad (11)$$

where A is the set of an arbitrarily shaped block, M is the cardinality of the set A, F is the value of luminance or color components, and MV is the motion vector.

We then refine the video objects. If only motion estimation is applied to obtain the shape of the video objects in the current frame, this result can produce incorrect video object boundaries in the current frame.

After all the control points are projected to the current frame, we apply the active contour algorithm to each object in the current frame by taking the projected control points as the initial contour.

III. SIMULATION RESULTS

In order to evaluate the performance of the proposed intra-frame segmentation algorithm, we have chosen "PEPPER" image of 512×512 pixels, because this image has more complex background than any other images used in active contour algorithms. In order to avoid heavy computational requirement for the minimization process, we reduce the search range gradually as in the three-step motion estimation algorithm [10]. Fig. 2(a) shows the initial contour provided by the user-pointing device. Fig 2(b) exhibits binary image edges obtained by morphological filters and the watershed algorithm.

Fig. 2(c) is the segmentation result obtained from the given initial contour by Lai's generalized snake algorithm [12]. If there is a local minimum point of gradient-based energy inside the area of the same object, the active contour algorithm fails to extract the object boundary accurately.

Fig. 2(d) is the final result obtained by our proposed method. If there is no difference of the morphological energy in a certain region, the contour converges to the local minimum value of the gradient-based energy. If the gradient value is very small in a certain region, the contour converges to the local minimum point of the morphological external energy. Fig. 2(d) demonstrates that the morphological approach produces improved results for images of complex background.

We have tested the overall user-assisted segmentation algorithm on a video conferencing sequence "MOTHER AND DAUGHTER" of CIF (352×288) format. Fig. 3(a) shows the initial contour provided by a user's pointing device. Fig. 3(b) is segmentation results obtained by the active contour algorithm of the intra-frame segmentation.

Fig. 3(c) demonstrates advantages obtained by the morphological energy as an external energy. Fig. 3(c) and Fig. 3(d) are results of inter-frame segmentation. In those figures, we estimate control points at Frame 1 using the control points of Fig. 3(b), set these control points to the initial contour at Frame 1, and obtain the video object at Frame 1 by using the proposed active contour algorithm. This procedure is performed repeatedly until the end of the image sequence. Fig. 3 shows the results of Frame 14 and Frame 28.

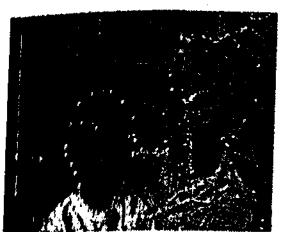
(a) Initial Control Points

(b) Morphological Edge

(c) G-Snake

(d) Proposed Scheme

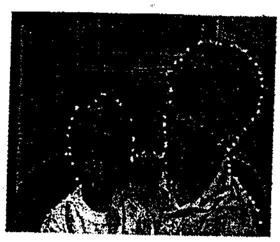
Fig. 2 Intra-Frame Segmentation



(a) Initial Contour (0)

(b) Intra-Segmentation (0)

(c) Inter-Segmentation (14)



(d) Inter-Segmentation (28)

Fig. 3 Inter-Frame Segmentation

IV. CONCLUSIONS

In this paper, we proposed a new user-assisted active contour algorithm to extract video objects from a given image sequence. The proposed algorithm consists of intra-frame segmentation and inter-frame segmentation. Our proposed algorithm performs quite well for objects of complex background image. In addition, we proposed an object-tracking scheme using an arbitrary shaped block-matching algorithm and overlapped block motion. In order to extract accurate object boundaries efficiently, we use the active contour algorithm of the intra-frame segmentation.

V. ACKNOWLEDGMENTS

This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber-Optic Networks (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

VI. REFERENCES

[1] M. Kim, J. Choi, D. Kim, H. Lee, C. Ahn and Y. S. Ho, "VOP generation tool: Automatic segmentation of moving objects in image sequences based on spatio-temporal information", *IEEE Trans. Circuit and System for Video Technology*, vol. 9, no. 8, pp. 1216-1226, 1999.

[2] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: active contour models", *First International Conference on Compuetr Vision*, pp. 259-269, 1987.

[3] K. F. Lai and R. T. Chin, "Deformable contours: modeling, extraction", *IEEE Trans. Patt. Anal. Mach. Intel.*, vol. 17, no 11, pp. 1084-1090. Nov. 1994.

[4] P. Salembier and M. Pardas, "Hierarchical morphological segmentation for image sequence coding", *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 639-651, Sep. 1994.

[5] P. Salembier, "Morphological multiscale segmentation for image coding", *Signal Processing*, vol. 38, no. 3, pp. 359-386, Jan. 1994.

[6] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations", *IEEE Trans. Patt. Anal. Mach. Intel.*, vol. 13, no.5, pp. 583-598, June 1991.

[7] A. Amimi, T. Weymouth and R. C. Jain, "Using dynamic programming for solving variational problems in vision", *IEEE. Trans. Patt. Anal. Mach. Intel.*, vol. 12, no. 9, pp. 855-867, Sep. 1990.

[8] M. Orchard and G. Sullivan, "Overlapped block motion compensation: an estimation-theoretic approach", *IEEE Trans. Image Processing*, vol. 3, no.5, pp. 693-699, 1994.

[9] C. Auyeung and J. Kosmach, "Overlapped block motion compensation", *SPIE Visual Communications and Image Processing*, vol. 1818, pp. 561-572, 1992.

[10] A. Tekalp, *Digital Video Processing*, Prentice-Hall, 1995.

[11] D. Kim and Y. Ho, "Image segmentation using active contour for images of complex background", *VLBV'99*, pp. 97-100, Oct. 1999.

[12] K. Lai and R. Chin, "Deformable contours: modeling, extraction", *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no 11, pp. 1084-1090, Nov 1994.