Extracting Semantic Video Objects

Fuhui Long, Dagan Feng, Hanchuan Peng, and Wan-Chi Siu
Hong Kong Polytechnic University

An interactive semantic video object extraction system adaptively performs spatial and temporal segmentation and fusion. It also switches between user-interactive and fully automatic modes.

Producing large amounts of digital media data every day requires fast transmission, efficient storage, flexible manipulation, and reuse of visual content. To achieve this goal, the ISO Moving Picture Expert Group's MPEG-4 standard provides a content-based framework. For video, it lets users transmit, retrieve, download, store, and reuse arbitrarily shaped semantic video objects (SVOs) efficiently and also interact with media sources. However, MPEG-4 doesn't provide concrete techniques for SVO extraction. Nonetheless, it's an indispensable process for many digital video applications.

Unfortunately, it's difficult to extract an SVO because:

1. A unique definition of an SVO doesn't exist. Anything that represents a meaningful entity in the world—for instance, a human body, a table, a building, an aircraft, and so on—could be classified as an SVO.
2. SVO extraction is basically a segmentation process, which researchers consider one of the most difficult problems in computer vision and image processing.
3. Traditional low-level visual homogeneity criteria (like color, texture, intensity, and so on) for segmentation don't lead to regions that immediately correspond to meaningful objects in the real world. More sophisticated, semantic, meaningful homogeneity criteria must be employed if possible. However, what's a good homogeneity criterion for a certain semantic meaning? Does it really exist?

Since an SVO usually has different motion features from the background and from other SVOs, most existing automatic SVO extraction schemes use motion information in video sequences as an important cue to produce semantic objects. Based on how the motion information is used, we can divide most current methods into three categories:

1. temporal segmentation,
2. spatial segmentation and temporal tracking, and
3. spatio-temporal segmentation.

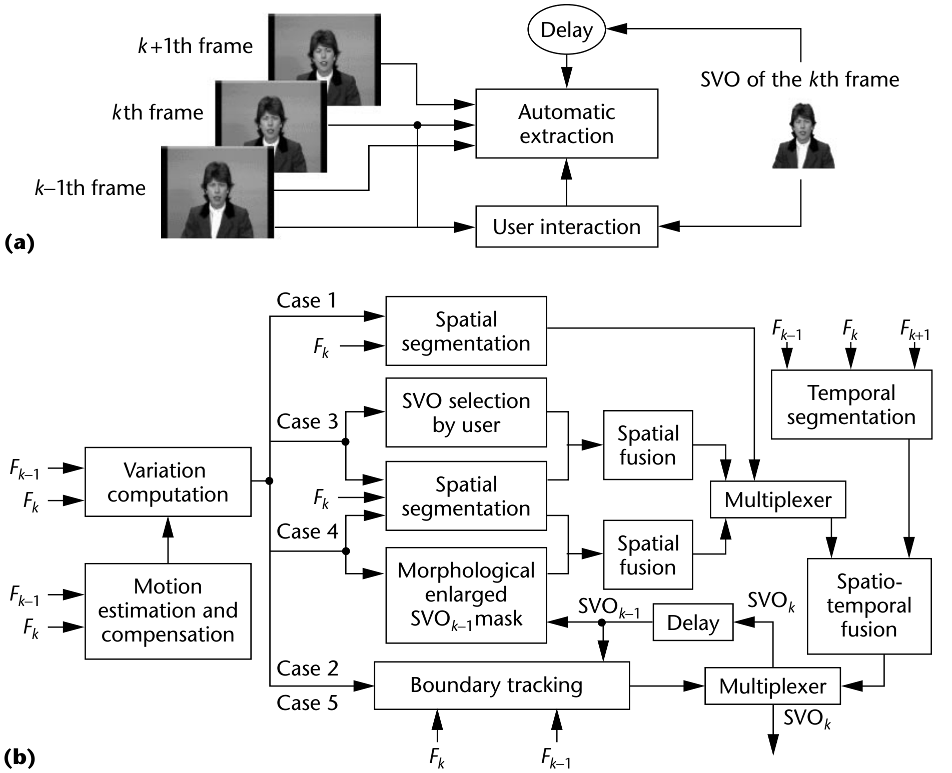
Temporal segmentation only uses motion information deduced from consecutive frames and doesn't consider spatial information. For instance, Wang and Adelson[1] employed the motion estimation, motion segmentation, and temporal integration to obtain video objects. Neri et al.[2] used the higher order statistics-based significance test to separate moving objects from the background. Since spatial information is neglected, extracted video objects aren't accurate at the boundaries.

To improve accuracy, we must consider spatial segmentation based on color, texture, and so on. One way is to perform spatial segmentation for the initial frame and temporal tracking for the successive frames. Wang[3] developed an algorithm using this two-stage technique to track fast moving objects and detect the appearance or disappearance of objects. However, his approach didn't consider complex motion and moving cameras.

Another way to improve accuracy is to impose spatial segmentation on each frame to modify the temporal segmentation result. Mech and Wollborn's method[4] used the significance test to detect frame change and produce temporal segmentation results. They superimposed luminance edge adaptation on the temporal segmentation results to improve the estimated object's boundary accuracy. Kim et al.[5] performed temporal segmentation (based on hypothesis testing of two successive frames) and spatial segmentation (based on watershed detection and region merging) simultaneously on each frame. They fused these two segmentation results to produce a moving object with an accurate boundary.

In addition to fully automatic methods, researchers have also studied semi-automatic techniques with user interaction. In Toklu et al.'s method,[6] users must manually segment an object of interest in the first frame. The segmented object is then fit to a deformable 2D triangular mesh, which is tracked and updated in the subsequent frames. Gu et al.[7] obtained the precise SVO boundary on the first frame using a combination of

0272-1716/01/$10.00 © 2001 IEEE



1 Diagram of our system. (a) A brief illustration. (b) The SVO extraction scheme in our system divided into five cases.

human assistance and morphological segmentation. They obtained the SVO in each of the remaining frames through motion compensation and boundary refinement based on the previous frame.

These methods are successful to some extent. However, SVO extraction techniques remain in their infancy. To benefit users, a good extraction method should be accurate, user interactive, and simple. Accuracy is an essential requirement. An inaccurate SVO containing parts of the background or losing its own parts can hardly be reused in content-based applications. Nonetheless, SVOs that most methods produce aren't accurate enough at boundaries, especially for video sequences containing complex background and motion. Although performing spatial segmentation on each frame can improve accuracy, it will simultaneously increase the computational complexity and decrease the speed. Fully automatic SVO extraction techniques are premature because it's difficult to formulate semantic concepts in homogeneity criteria. User interaction is extremely important, since it acts as a way to provide semantic constraints to the SVO extraction process. With user interaction, the SVO extractor will learn what an SVO looks like and will focus on the area of interest. However, in most of the existing semi-automatic methods, users interact with the automatic extraction in a two-phase way. That is, users outline the interested SVO (a semantic constraint) in the initial frame and the automatic extraction algorithm uses this semantic constraint to extract the SVO in the following frames. In such a scheme it's difficult to detect a new SVO.

To help solve these problems, we present an accurate and user-interactive SVO extraction system. Although we also obtain an SVO with an accurate boundary by integrating temporal and spatial information, our way is quite different from others' work. Instead of fusing spatial and temporal segmentations on the first[3] or all the frames[5] of a video sequence, our system adaptively performs spatial and temporal segmentation and fusion when necessary. To achieve this, our system detects the variations between successive frames. We only need to fuse the spatial and temporal segmentation when a large variation occurs. Otherwise, the system tracks the previous SVO's boundary. We find this simple method efficient in both speed and accuracy. Since the temporal segmentation, spatial segmentation, spatio-temporal fusion, and boundary tracking all employ simple algorithms, our system has a low computational complexity.

In addition, our system provides a flexible switch between the user-interactive and fully automatic extraction modes. User interactions can be imposed, removed, or changed in the automatic extraction process at any time. Thus the system can extract SVOs of interest with semantic meaning that users provide and detect unexpected SVOs as well. Adaptively performing spatio-temporal segmentation and boundary tracking and flexibly switching between user interactive and fully automatic extraction modes—which to our knowledge hasn't been employed by other methods up to now—make our system accurate, fast, flexible, and thus a powerful tool for SVO extraction in many digital video applications.

System overview

To obtain the SVO of the *k*th frame, our system uses video frames $k-1$, k, $k+1$, and the SVO of frame $k-1$ as inputs (Figure 1a). The system uses the flexible user interaction and automatic extraction modes. Figure 1b shows details of the whole system. The system first estimates and compensates for global motion based on the simplified linear model.[8] To estimate the model's parameters, we use hierarchical block matching and least-square approximation. Then the system computes the variation between successive frames $k - 1$ and k using the sum of the second derivative of the frame luminance on each pixel. After that, the system performs SVO extraction according to two criteria:

1. If the variation between successive frames $k - 1$ and k is larger than a predefined threshold T_v, spatial and temporal segmentation will be performed and fused to obtain an accurate SVO. Otherwise, the system tracks the extracted SVO's boundary in frame $k - 1$ to produce an SVO for frame k. This reduces the computational complexity and amount of processing time.
2. If user interaction is imposed, spatial segmentation must be fused with the SVO's mask that either the user provides or the system obtains from frame $k - 1$ (which is influenced by user interaction) before fusing with temporal segmentation.

These two criteria lead to the following five different cases for SVO extraction (Figure 1b):

1. If no user interaction is imposed and the variation between frame k and $k - 1$ is greater than T_v (case 1 in Figure 1b), then the system automatically extracts the SVO based on spatial and temporal segmentation and fusion.
2. If no user interaction is imposed and the variation between frame k and $k - 1$ is subtle and not larger than T_v (case 2 in Figure 1b), then the system automatically extracts the SVO based on tracking the extracted SVO's boundary in frame $k - 1$.
3. If user interaction is imposed on the current frame (case 3 in Figure 1 b), then the system fuses the presumed SVO that the user selects with spatial segmentation to obtain the SVO's spatial mask. It then fuses this mask with temporal segmentation.
4. If user interaction has already been imposed and hasn't been removed, and the variation between successive frames is larger than T_v (case 4 in Figure 1b), then the system enlarges the SVO's mask extracted from frame $k - 1$ morphologically as the semantic constraint and fuses it with spatial segmentation to obtain the SVO's spatial mask in frame k. The system then fuses this mask with temporal segmentation.
5. If user interaction has already been imposed and hasn't been removed, and the variation between successive frames isn't larger than T_v (case 5 in Figure 1b), then the system tracks the boundary of the SVO extracted from frame $k - 1$.

Spatial segmentation

Spatial segmentation divides a frame into homogenous regions in terms of intensity, color, texture, and so on. The morphological filtering algorithm[9] is a popular method for this process. However, the serial steps of morphological simplification, gradient approximation, watershed detection, and region merging make the algorithm relatively complex. We propose an effective and simple method based on hierarchical adaptive thresholding (HAT)[10] and region merging to perform accurate spatial segmentation.

In HAT, multiple thresholds are obtained by hierarchically dichotomizing the intensity histogram into continuous intervals until every interval has a pixel-by-pixel mean square error (MSE) less than a given threshold T_σ. The histogram MSE on the intensity interval $[d, u]$ is defined as

$$\sigma^2_{[d,u]} = \frac{\sum_{k=d}^{u} P(k)\left(k - \eta_{[d,u]}\right)^2}{\sum_{k=d}^{u} P(k)} \tag{1}$$

where d and u are lower and upper limits of the current intensity interval, function $P(k)$ is the normalized intensity histogram ($\Sigma_k P(k) = 1$), and r is the quantatized intensity value of the histogram interval, as defined in Equation 2

$$\eta_{[d,u]} = \frac{\sum_{k=d}^{u} P(k)k}{\sum_{k=d}^{u} P(k)} \tag{2}$$

When a histogram interval's MSE is larger than T_σ, this interval will be split into two subintervals, whose sum of MSE is minimized. That is, the interval division point $c_{[d,u]}$ is chosen as the parameter at which the following sum term is minimized:

$$c_{[d,u]} = \operatorname{argmin}\left\{\sigma^2_{[d,c]} + \sigma^2_{[c+1,u]}\right\} \tag{3}$$

where arg(.) is the operator to extract the parameter.

The threshold T_σ adaptively controls the details of the segmentation result. More regions will be produced if T_σ takes a smaller value. However, visually important regions with sharp intensity variation at their edges can easily be segmented and thus aren't sensitive to T_σ.

The thresholded image that HAT produces usually contains too many small regions. As a result, the region merging is performed according to the following criteria:

- If the difference between the mean values of two neighboring regions is less than a normalized threshold T_m (a constant percent of the maximum value of the region means), then these two regions should be merged.
- If the size of a region is less than a normalized threshold T_a (a constant percent of the frame size), then the region should be combined with its neighboring region whose mean value is closest to that of this region.

Temporal segmentation

Temporal segmentation detects moving regions from the background of each frame. Based on Mech and Wollborn's[4] work, we propose a simplified model with reduced computation complexity for temporal segmentation.

Denote the squared luminance difference between frame F_{k-1} and F_k as $D^2_{k-1,k}$. Denote the normalized sum of pixels in $D^2_{k-1,k}$ within a $(2w+1)^2$ window centered on (m, n) as $A_{m,n}$:

$$A_{m,n} = \frac{1}{\sigma_t^2} \sum_{i=m-w}^{m+w} \sum_{j=n-w}^{n+w} D^2_{k-1,k}(i,j) \tag{4}$$

$A_{m,n}$ is a X^2 distribution with $(2w+1)^2$ degrees of freedom.[4] Variance σ_t^2 is automatically estimated by

$$\sigma_t^2 = \frac{1}{N_S} \sum_{(p,q)\in S} \left(D_{k-1,k}(p,q) - \mu_S\right)^2 \tag{5}$$

where S is a subset of the static background, N_S is the area of S, μ_S is the mean value of pixels in the area of S in $D_{k-1,k}$. Since the scene's center, left, right, and bottom borders are more likely to be occupied by a moving object than the upper borders, we select those upper border pixels, which are classified as static background in the past L frames, as the subset S.

We define the change detection mask (CDM) that indicates changes between frame $k-1$ and k as

$$\mathrm{CDM}_{k-1,k}(m,n) = \begin{cases} 1 & \text{if } A_{m,n} > T_\alpha \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where T_σ is determined by the significance test.[4]

To get temporal coherent object regions, we labeled a pixel as changed in the kth frame, if it belongs to SVO_{k-1} and labeled as changed in one of the CDMs of the last L frames. To do this, we built a time-variant memory matrix **M**. For the kth frame, M_k on pixel (m, n) can be formulated as

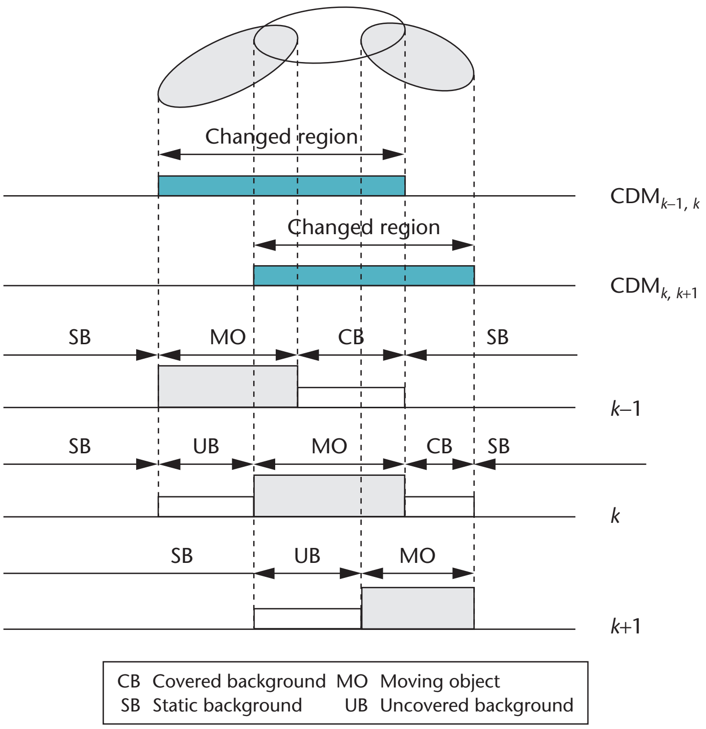
$$M_k(m,n) = \begin{cases} L & \text{if}(m,n) \in \mathrm{CDM}_{k-1,k} \\ \max\left(0, M_{k-1}(m,n) - 1\right) & \text{if}(m,n) \notin \mathrm{CDM}_{k-1,k} \end{cases} \tag{7}$$

Then $\mathrm{CDM}_{k-1,k}$ is updated by

$$\mathrm{CDM}_{k-1,k}(m,n) = \mathrm{CDM}_{k-1,k}(m,n) \vee \begin{cases} \mathrm{SVO}_{k-1}(m,n) & \text{if } M_{k-1}(m,n) > 0 \\ 0 & \text{if } M_{k-1}(m,n) = 0 \end{cases} \tag{8}$$

where SVO_{k-1} is the semantic video object extracted from frame $k-1$.

The changed areas contain moving objects as well as covered and uncovered backgrounds. To extract the moving objects, most existing methods used a hierarchical block-matching algorithm, which is computationally complex. Here we use a simple method based on three successive frames. As Figure 2 shows, the moving objects in frame k can be obtained by detecting the common regions between $\mathrm{CDM}_{k-1,k}$ and $\mathrm{CDM}_{k,k+1}$.



2 **Removing uncovered and covered backgrounds using three successive frames.**

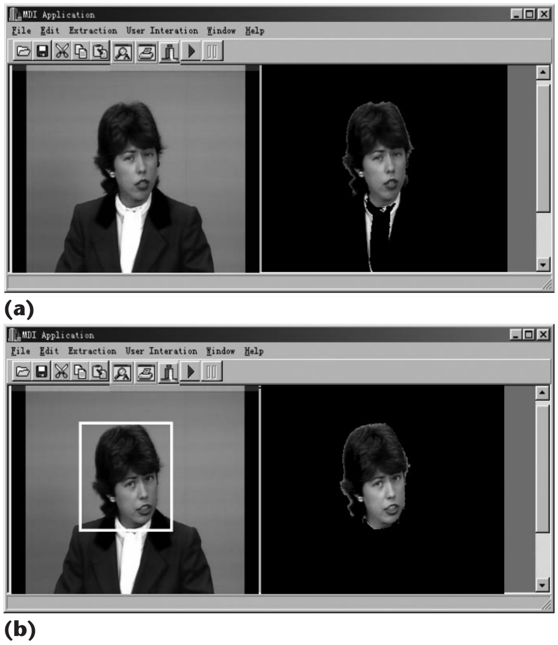
Because of the missing probability (type II error) problem in the significance test, unless the regions corresponding to the SVO are highly textured, the change-detection algorithm will produce some holes inside the moving object and some noisy regions in the temporal segmentation result. Therefore, we use morphological filtering to refine the result. First, we remove the connected components with an area less than threshold T_A (a constant percentage of the frame size) by using a morphological opening operation. Then we remove holes inside the moving objects areas by performing opening and closing operations sequentially. Here we use circular structuring elements with radius R.

Spatio-temporal fusion

To make the video object's boundary correspond accurately to its spatial feature variation, we fuse spatial and temporal segmentations. Denote the spatial and temporal segmentation results on the kth frame as S_k^{spt} and S_k^{tmp}, respectively. We superimpose S_k^{spt} on S_k^{tmp} and define the following ratio η:

$$\eta = \frac{\mathrm{Area}\left(S_k^{\mathrm{spt}} \cap S_k^{\mathrm{tmp}}\right)}{\mathrm{Area}\left(S_k^{\mathrm{spt}}\right)} \tag{9}$$

where Area(.) is the region area operator. If η exceeds a threshold T_η, we declare the whole region in S_k^{spt} as part of the moving object. Otherwise, we declare the whole region in the spatial segmented image as the background.


MDI Application
File Edit Extraction User Interation Window Help
(a)
MDI Application
File Edit Extraction User Interation Window Help
(b)

3 **Main interface of our system. (a) Fully automatic SVO extraction without user interaction. (b) Semi-automatic SVO extraction with Claire's head selected by the user.**

Boundary tracking

We perform boundary tracking when the variation between successive frames is subtle (less than T_v). Denote p_{mn}^{k-1} as a pixel at (m, n) on the boundary of SVO$_{k-1}$ extracted from frame k–1, and p_{mn}^{k} as the pixel at the same location in frame k. Denote Q as a $(2g+1)^2$ size searching window centered on p_{mn}^{k}, where g is the radius of the window. (Note that the value of g should be determined by the moving object's speed. As the object moves faster, g becomes larger.) Then we obtain $\forall p_{ij}^{k} \in Q$, the best matching point p_{st}^{k} in frame k for the boundary point p_{mn}^{k-1} by

$$p_{st}^{k} = \min_{ij}\left[\text{Dis}\left(p_{mn}^{k-1}, p_{ij}^{k}\right)\right] \qquad (10)$$

where Dis(.) is a distance operator defined as

$$\text{Dis}\left(p_{mn}^{k-1}, p_{ij}^{k}\right) = \sum_{a=-1}^{1}\sum_{b=-1}^{1}\left|I_{m+a,n+b}^{k-1} - I_{i+a,j+b}^{k}\right| \qquad (11)$$

and I is the frame luminance.

Finally, for all the boundary points in frame $k-1$, we link their best matching points in frame k to obtain the closed boundaries of SVO$_k$.

User interaction

Users should be able to interact with the automatic extraction process in an easy and flexible way. It's impractical for users to enter complex and time-consuming inputs like outlining the SVO boundary in detail. Moreover, a flexible switch should exist between user-interactive and fully automatic extraction modes so that the SVO extractor can get the SVO's semantic meaning provided by the users and automatically detect unexpected video objects.

In our system, users interact with the system by drawing a simple rectangular bounding box to select the object of interest in any frame. As Figure 1b illustrated previously, the system will perform a spatial fusion to integrate the user's selection with the spatial segmentation result to obtain a spatial mask for the SVO. The fusion process resembles the mechanism of the spatio-temporal fusion introduced previously. If the rectangular bounding box covers the major part (exceeds the threshold $T\eta$) of a region in the spatial segmentation, then we declare the whole region as part of the spatial mask of the SVO. Otherwise, the system removes the whole region from the mask. The obtained spatial mask will later be fused with the temporal segmentation results.

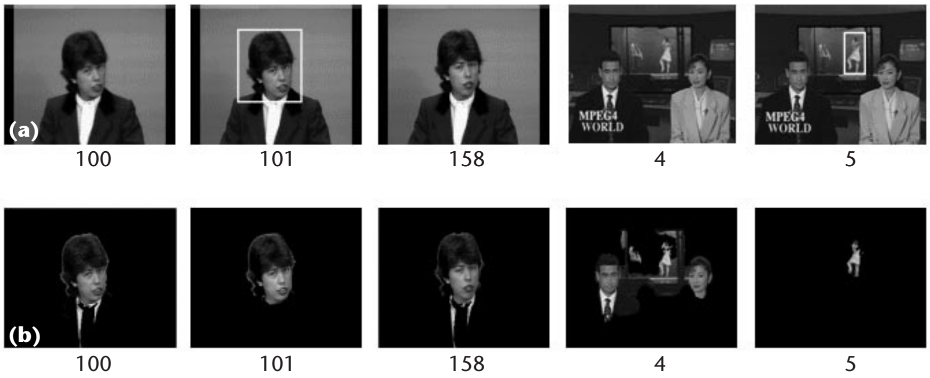
Figure 3 shows our system's interface. Users can impose, remove, or change an interaction. The impose operation lets users select an object of interest by drawing a rectangular bounding box. The remove operation will make user interaction invalid and extracts the SVO automatically. To change an object of interest already selected, users need to perform the remove operation first, then redraw a bounding box to select a new object.

Experimental results

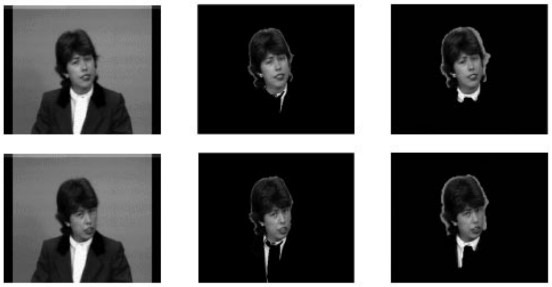
For brevity we'll only show the partial experimental results with the standard video sequences *Claire*, *News*, and *Children*. Parameters in our experiments are as follows:

1. For frame variation test, $T_v = 20$
2. For spatial segmentation, $T_\sigma = 138.4$, T_m is 0.01 of the maximum mean value of the regions to be merged, T_a is 0.001 of the frame size
3. For temporal segmentation, $L = 4$, $T_\alpha = 23$, $R = 3$, $w = 1$, T_A is 0.002 of the frame size
4. For spatial fusion and spatio-temporal fusion, $T\eta = 0.8$
5. For boundary tracking, $g = 1$

Figure 4 illustrates the flexible switch between the user interactive and automatic extraction process. We use *Claire* and *News* to compose a new test video sequence. (In this case, all frames in *News* and *Claire* are 352 × 288.) Figure 4a shows frames 100, 101, and 158 of *Claire* and frames 4 and 5 of *News*. The white rectangular boxes indicate the objects users selected. Figure 4b shows the SVOs extracted using our method. For frame 100 of *Claire*, no user interaction is performed. The system automatically extracts the SVO. At frame 101, the user selects the woman's head as the object of interest. The SVO extracted from that frame excludes the woman's collar. At frame 158, the user's interaction is removed. The system performs automatic extraction again. The system detects the woman's collar again as part of the SVO. When a new video scene appears, the system can detect a totally dif-

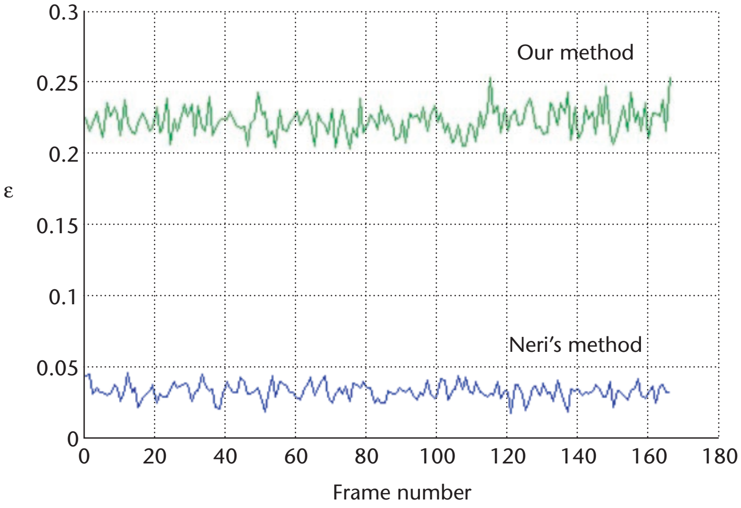


4 Experiment of flexible user interaction. (a) Original frames with or without user interactions. (b) SVOs extracted from corresponding frames using our system.



5 SVOs extracted from frames 90 and 100 of *Claire* using our method and Neri's method. Orginal frames 90 and 100 (left column). Corresponding SVOs extracted using our method without user interaction (middle column). Corresponding SVOs extracted using Neri's method (right column).

ferent SVO, as illustrated by the SVO extracted from frame 4 of *News*, which shows a man, woman, and dancer (in the background). Of course, users can impose their interest again, for instance, on the female dancer of frame 5 of *News*, the SVO extracted excludes other objects as illustrated. This experiment demonstrates our system's flexibility. By imposing and removing user interaction, the system can perform automatic extraction, focus on the possible SVO of interest, and detect new SVOs.



6 Boundary accuracy rate on all 168 frames of *Claire*. The upper curve shows the results of our method. The lower curve shows the results of Neri's method.[2]

Table 1. Average boundary accuracy rate ε of *Claire*, *News*, and *Children*.

Video Sequence	Our Method	Neri's Method
Claire	0.2225	0.0312
News	0.2933	0.0530
Children	0.2590	0.0334

Figure 5 compares the SVOs obtained by our method to those obtained by Neri's method.[2] The left column images are frames 90 and 100 of *Claire*. The middle column images are the SVOs extracted using our method without user interaction. The right column images are the SVOs extracted using Neri's method. Obviously, our results are perceptually more precise at the object boundaries than Neri's results.

To measure the boundary accuracy quantitatively, we used the Sobel edge detector to obtain a frame's standard boundary. We matched the extracted SVO's boundary with the standard boundary. We then calculated the ratio ε between the number of pixels on the matched boundary and the number of pixels on the boundary of the extracted SVO. The higher the ε, the more accurate the result. We tested all 168 frames in *Claire* using Neri's method and ours. Figure 6 shows the results, which indicate that our method produces more precise boundaries on all the frames than Neri's method does.

We also tested *News* and *Children*. Table 1 lists the average ε on all the frames of each sequence. Again, the results demonstrate that our method is better than Neri's in producing an SVO with more precise boundaries. This is natural because we use not only temporal information but also spatial information, which makes the SVO boundaries as close as possible to the locations where large spatial feature variation occur.

Future work

Extracting an SVO is an indispensable and difficult task in many MPEG-4-based digital video applications like fast transmission, efficient storage, flexible manipulation, and reuse of visual contents. In our proposed approach, adaptively performing spatio-temporal segmentation and boundary tracking results in fast extraction of accurate SVOs. Users may interact with our system by imposing, removing, and changing the interested semantic object at any frame.

In our system, the threshold T_v has a significant influence on both SVO accuracy and SVO extraction speed. A smaller T_v leads to more accurate SVO boundaries but slower speed, while a larger T_v speeds up the SVO extraction but decreases the accuracy. Currently, we're investigating some strategies on how to control the parameter T_v for different video sequence contents. We also plan to introduce more robust statistical methods for the purpose of extracting, or mining, SVOs. Better user interfaces and multimedia human-machine interaction modules are also under development to enhance the whole system. ■

Acknowledgment

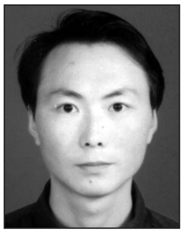
This work is supported by a Research Grants Council (RGC) grant and funded by the Center for Multimedia Signal Processing at Hong Kong Polytechnic University.

References

1. J.Y.A. Wang and E.H. Adelson, "Representing Moving Images with Layers," *IEEE Trans. Image Processing*, vol. 3, no. 5, Sept. 1994, pp. 625-638.
2. A. Neri et al., "Automatic Moving Object and Background Separation," *Signal Processing*, vol. 66, no. 2, Apr. 1998, pp. 219-232.
3. D. Wang, "Unsupervised Video Segmentation Based on Watersheds and Temporal Tracking," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, Sept. 1998, pp. 539-545.
4. R. Mech and M. Wollborn, "A Noise Robust Method for 2D Shape Estimation of Moving Objects in Video Sequences Considering a Moving Camera," *Signal Processing*, vol. 66, no. 2, April 1998, pp. 203-217.
5. M. Kim et al., "A VOP Generation Tool: Automatic Segmentation of Moving Objects in Image Sequences Based on Spatial-Temporal Information," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 8, Dec. 1999, pp. 1216-1226.
6. C. Toklu et al., "Simultaneous Alpha Map Generation and 2D Mesh Tracking for Multimedia Applications," *Proc. IEEE Int'l Conf. Image Processing* (ICIP 97), vol. 1, IEEE Computer Society Press, Los Alamitos, Calif., 1997, pp. 113-116.
7. C. Gu, and M.C. Lee, "Semiautomatic Segmentation and Tracking of Semantic Video Objects," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, Sept. 1998, pp. 572-584.
8. H. Nicolas and C. Labit, "Motion and Illumination Variation Estimation Using a Hierarchy of Models: Application to Image Sequence Coding," *J. Visual Communication Image Representation*, vol. 6, no. 4, 1995, pp. 303-316.
9. P. Salembier and M. Pardas, "Hierarchical Morphological Segmentation for Image Sequence Coding," *IEEE Trans. Image Processing*, vol. 3, no. 5, 1994, pp. 639-651.
10. H.C. Peng et al., "Hierarchical Genetic Image Segmentation Algorithm Based on Histogram Dichotomy," *Electronic Letters*, vol. 36, no. 10, 2000, pp. 872-874.

Fuhui Long *is a research associate in the Center for Multimedia Signal Processing, Department of Electronic and Information Engineering, at Hong Kong Polytechnic University. Her research interests include multimedia signal processing, pattern recognition, image and video processing, computer vision, multimedia database management systems, evolutionary computing, and content-based information retrieval. She received her PhD in electronic and information engineering from Xi'an Jiaotong University, China, in 1998 and an MS and BS in computer science and engineering from Northwestern Polytechnic University, China, in 1995, and 1992, respectively. She received the Excellent PhD Graduate Award from the Educational Committee of Shaanxi Province, China, and the First-Class Scholarship Award from the Pan Wenyuan Educational Foundation.*

Dagan (David) Feng *is a professor and head of the Department of Computer Science at the University of Sydney. He is also a professor and deputy director of the Center for Multimedia Signal Processing, Department of Electronic and Information Engineering at Hong Kong Polytechnic University. His research interests include biomedical image modeling, simulation, compression, and multimedia signal processing. He received his PhD in computer science and MS in biocybernetics from the University of California, Los Angeles, in 1988 and 1985, respectively. He earned an MS in electrical engineering and computing science from Shanghai JiaoTong University, China, in 1982. Feng is the vice chair of the International Federation of Automatic Control (IFAC) Technical Committee on biomedical engineering.*



Hanchuan Peng *is a postdoctoral research fellow in the NeuroImaging Lab, Department of Radiology, School of Medicine, at Johns Hopkins University. His research interests include multimedia and biomedical signal processing, pattern recognition, neural networks, data mining and heterogeneous database applications, and biological computing theories and models. He received his PhD in signal processing and biomedical engineering, an MS in biological electronics, and a BS in biomedical engineering and instruments from Southeast University, China, in 1999, 1996, and 1994, respectively. He was a research associate at Hong Kong Polytechnic University from 1999 to 2000. In 1997 he won the Champion of National Computer Software Competition (co-awarded by the Ministry of Electronic Industry, China and the Ministry of Education, China).*

Wan-Chi Siu *is Chair Professor and Director of the Center for Multimedia Signal Processing, Department of Electronic and Information Engineering, and dean of the engineering faculty at the Hong Kong Polytechnic University. His research interests include digital signal processing, fast computational algorithms, transforms, video coding, computational aspects of image processing and pattern recognition, and neural networks. He received a PhD from Imperial College of Science, Technology, and Medicine, London; an MPhil from Chinese University of Hong Kong; and an Associateship from Hong Kong Polytechnic University in 1984, 1977, and 1975, respectively.*

He is an editorial board member of Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology *and* IEE Review. *He is the general chair of the 2001 International Symposium on Intelligent Multimedia, Video, and Speech Processing and of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing. He is a chartered engineer, a fellow of the IEE and the Hong Kong Institute of Engineers, and a senior member of the IEEE.*

Readers may contact Peng at the NeuroImaging Lab, Dept. of Radiology, School of Medicine, Johns Hopkins University, 601 N. Caroline St., JHOC 3230, Baltimore, MD 21287, email phc@cbmv.jhu.edu.