A Fully Automated Object Extraction from Video Stream
: A useful tool for Distributed Object-based Browsing and Content-based Searching Systems

Atsunobu Hiraiwa, Keisuke Fuse, Naohisa Komatsu, Kazumi Komiya, and Hiroaki Ikeda
Atsugi Research Center, Telecommunications Advancement Organization of Japan
7TH floor, Atsugi AXT Maintower, 3050 Okada, Atsugi, Kanagawa-ken 243-0021, Japan

Abstract

This paper proposes a new approach to automatically extract an accurate object from video streams. The new approach provides a useful tool for distributed ***object-based browsing*** *and* ***content-based searching systems****, and consists of a* ***skip-labeling*** *algorithm for feature-based segmentation, an* ***occlusion-killer*** *algorithm for estimating accurately optical flow, and a* ***shrink-merge tracking*** *algorithm for tracking an object. The* ***shrink-merge tracking*** *algorithm is executed, based on the time-continuity of moving-objects, using morphological image processing, such as dilation and erosion. The dilation and erosion are repeatedly executed using the projection processing which the object area in a next frame is derived from the object area in a current frame. The* ***shrink-merge tracking*** *algorithm can also project the area of a rotating-object in a current frame on the rotating-object containing the newly appearing regions in the next frame. The newly automated object extraction method works satisfactory for the objects which are non-linearly moving within the video stream, and also works satisfactory in 450 frames.*

1. Introduction

The *world wide web* (WWW) browsing system establishes a link in hypertext as HTML. The architectural current browsing system has been driven by the static nature of text-based information. In order to incorporate movies into the WWW, the architecture of the WWW extends the object-based linking-information. However, the extended object-based linking-information preparation is much more difficult than the text and images. This problem is evidenced in the lack of automatic extraction of moving-objects. The extraction of moving-objects automatically makes linked-information to identify the objects non-linearly moving within the video streams is much more difficult. There is no such effective support in the present WWW environment.

The case of capture and encoding of digital video has caused a massive amount of visual information to be produced and disseminated rapidly. Hence, efficient tools and system for searching or retrieving visual information are needed. While there are efficient search engines for text document today, there are no satisfactory systems for retrieving visual information. The current context-based visual queries systems such as QBIC [11], have been primarily focused on still image retrieval. This problem also is evidenced in the lack of automatic extraction of moving-objects.

The purpose of this paper is to present the newly automatic extraction of objects from video streams in order to solve the above problems. The newly automatic extraction of objects provides a useful tool for **distributed object-based browsing** and **content-based searching systems**, and consists of a ***skip-labeling*** for feature-based segmentation, an ***occlusion-killer*** for estimating accurately optical flow, and a ***shrink-merge tracking*** for tracking an object. This ***skip-labeling*** algorithm can be used to segment an image into integrated regions of the same feature. The segmented regions belong to such a texture area as waves or forest. In block matching, optical-flow could theoretically be used, but the accuracy of the optical-flow is limited due to limited *block size, aperture*, and *occlusion*. The effects of template matching using results of the ***skip-labeling*** are that no segmented region is equal size and equal shape, and that neighboring same feature area is integrated in one region. The problems *of block size* and *aperture* are resolved to these effects. The ***occlusion-killer*** algorithm is executed using the difference between the results of the forward and backward template matching in segmented frames, which is performed by ***skip-labeling*** *algorithm*. The ***occlusion-killer*** algorithm [8] can be used to accurately estimate optical-flow in the neighboring regions of moving-object on the background, and provides the solution to the *occlusion* problem. The ***shrink-merge tracking*** *algorithm* which was proposed by Hiraiwa etc.[6], uses the time continuity of moving-objects and morphological image processing, such as dilation and erosion. The dilation and erosion repeatedly execute the projection which object area in a next frame is derived from object area in a current frame. The ***shrink-merge tracking*** *algorithm* can also project the current rotating-object area on the next rotating-object containing newly appearing region.

The automated object extraction method fully works well for objects non-linearly moving within video streams, and also works well in 450 frames each with a full frame size of 704 x 480 pixels. In [1], object information obtained by this fully automated object extraction from video-stream can used to create object-based linking information for an **object-based browsing system**. Finally, this paper describes a newly efficient approach for a **content-based**

0-7695-0253-9/99 $10.00 © 1999 IEEE

searching system using the *Hausdorff* distance and object information.

2. Overview of object extraction

The presented algorithms for extracting objects from a video stream consist of by a ***skip-labeling*** algorithm for feature-based segmentation, an **occlusion-killer** algorithm for estimating accurately optical flow, and a ***shrink-merge tracking*** algorithm for tracking an object. The basic procedure extracting an object is shown in Figure 1. First, feature-based segmentation separates each frame into regions. Second, the optical flow is accurately estimated from the difference between the forward and backward optical flows, which the results of region-based template matching are given. Finally, during object tracking control repeatedly executes the projection in which object area in the next frame is derived from object area in the current frame. The projection consists of dilation and erosion.

3. Region segmentation

K-mean and *labeling* (region growing) algorithms are well-known algorithms for segmenting regions within a frame. As a result of segmentation using *K-mean* algorithm, an area of approximately circular shape is obtained depending on the initial value of clusters. A number of segments of small sizes are obtained from the *labeling* algorithm applied to the texture. [3][4][5]

A new algorithm for segmenting a region, the ***skip-labeling*** al*gorithm*, is proposed here. Using the ***skip-labeling*** *algorithm*, an image is separated into a uniform area (i.e., sky), a semi-uniform area (i.e., a mountain or cloud), a texture area (i.e., waves or forest), and an edge area. During the execution of ***skip-labeling*** *algorithm*, threshold T increases stepwise and the searching area S also expands stepwise. The ***skip-labeling*** *algorithm* consists of the following 7 steps.

[step1] Initialize searching area S with a size of 3 x 3. Initialize threshold T to be a small value. Set Minimum size of segment area *MSA*.

[step2] Select a small area P from an image. Let a segment $A_0 = P$. Select a small area Q within searching area S of which P is the center.

[step3] **if** the following condition
$|P - Q| < T$ where P or Q is small area such a pixel. P is an element of A_n, and T is threshold.
then Q and A_n merge into A_{n+1}.

[step4] **repeat** [step3] **until** applied area disappear.

[step5] **if** size of segment $A_n < MSA$ **then** this segment A_n isn't handled as a segment.

[step6] **repeat** [step2-5] within all small area of an image.

[step7] **repeat** Within non-segmented areas, increase threshold T, expand searching area S and execute [step2-6] several times.

Figure 2 shows the basic concept of the skip-labeling algorithm. The longitudinal axis indicates the value of a pixel, and the horizontal axis indicates x of the spatial domain. First, in an uniform area, the growth of a region stops at a dull edge, the result of the labeling is the same as that in the skip-labeling. Second, a threshold to the value of a pixel is increased, and a search-area will be expanded. The search-area indicates the area where a growing region can skip in spatial domain. As parameters, which are a threshold to the value of a pixel and a search-area, correspond to the features of the area, during the skip-labeling algorithm control separates not only an uniform area but also a texture area which is not separated by the labeling algorithm. In the texture, or semi-edge area, the growth of a region stops periodically in a long span or at a sharp edge.

4. Segmentation results

This section describes the results of segmentation obtained using *k-mean*, *labeling* and ***skip-labeling*** *algorithms*. Figure 3 (a) shows an original test image of 704 x 480 pixels. Figure 3 (b) shows the result of segmentation using *k-mean* in which the characteristics of clusters are approximately shaped circular. The sky is partitioned into multiple clusters. Figure 3 (c) shows the result of segmentation using *labeling* in which such an uniform area as blue-sky is picked up. Figure 3 (d) shows the result of segmentation using ***skip-labeling*** *algorithm* whose an image can be separated into such an uniform area as blue-sky, such a semi-uniform area as mountain, such a texture area as waves or forest, and an edge area. No segmented region is contained both the moving object and background. In the segmentation of such a texture area as waves or forest, the ***skip-labeling*** gives better result than *k-mean or labeling*.

5. *Shrink-merge tracking* algorithm for extraction of movie-object

In this section, newly proposed ***shrink-merge tracking*** *algorithm* is presented. This algorithm is executed, based on time continuity of movie-objects, and time continuity of the background. Morphological image processing such dilation and erosion is repeatedly performed during execution of ***shrink-merge tracking*** *algorithm*. Time continuity of movie-objects is guaranteed by the time-invariant characteristic of a moving-object that consists of an area, a moving vector, a perimeter, a thinness ratio, and a variance. [3] Time continuity of the background is also guaranteed by the spatio-temporal continuity of moving vectors in a fully area without moving-objects. Dilation is useful for matching the current frame of a moving-object to the next frame when the difference between them is small. In [4], erosion processing increase the size of an object by rolling a small circle region around the contour of the object, and erosion processing merges neighboring segments, which satisfy time-continuity between the current and next frames, into the new moving-object. In

Figure 4, the ***shrink-merge tracking*** *algorithm* is repeatedly executed in the projection in which object area in next frame is derived from that in current frame. The projection consists of dilation and erosion. The purpose of dilation of regions within an object area is to prepare an image to establish for optimal template matching between dilated region in current frame and region in next frame. During the execution of erosion, the neighboring regions having the same optical flow as that in the matched region are merged into matched region. The effect of the ***shrink-merge tracking*** *algorithm* can also be to project the area of a rotating-object area in the current frame onto the rotating-object with newly appearing regions in the next frame. This ***shrink-merge tracking*** *algorithm* is described in the following 9 steps.

[step 1] Specify the area of the moving-object on the start frame.

[step 2] Partition the moving-object on the current frame into N parts.

[step 3] In each part, compute the distance of template matching between the current and next frames, and normalize the distance by the area size.

[step 4] **if** normalized distance value (NDV) > threshold T **then** **repeat** dilate or rotate the moving-object and execute [step3] **until** NTV □ T.

[step 5] In each part, the new part is a set of segments obtained by executing the *skip-labeling algorithm*, and is a new matched area on the next frame.

[step 6] Merge each new part into the new area of the moving-object.

[step 7] Get attributes and moving vectors of segments which neighbor new area of the moving-object.
The attribute is the area value, the threshold to value of pixel, and the search-area obtained with the ***skip-labeling***.

[step 8] Merge a neighboring segment into new area of the moving-object. Compute each value of the time invariant of new moving-object. Parameters of time-invariant of moving-object consist of an area, a moving vector, a perimeter, a thinness ratio, and a variance.
if the time-invariant characteristic of the new moving-object is in excess of constrains of the parameters of time-invariant.
then Delete a merged neighboring segment from the new area of the moving-object.

[step 9] **repeat** [step7-8] **until** the merged neighboring segment disappears

6. Tracking results

The ***shrink-merge tracking*** *algorithm* proposed by the author is used for assuring the time continuity of movie-objects and morphological image processing such as dilation and erosion [7]. Figures 5 (a)-(f) show the results of the ***shrink-merge tracking*** *algorithm* in which the fully automated tracking ability of moving-objects is improved using the accurate results of optical-flow estimation within video streams. Evaluated video streams include panned background and two expanding and rotating moving-objects (a boat and a helicopter). Figure 5 (a) shows the start frame which specified moving-objects by human eyes are black-filled parts of this frame. Figures 5 (b)-(h) show results obtained with the ***shrink-merge tracking*** *algorithm* at every 90 frame interval. White-filled parts of these frames show new area of the moving-objects. Black-filled parts of these frames show the neighboring segments. Figures 5 (a)-(h) show that during execution of the ***shrink-merge tracking*** *algorithm* fully automated extraction of moving-objects are obtained in 450 frames with a full frame size of 704 x 480 pixels without operator's intervention.

7. Content-based searching system

A content-based searching system has emerged as a challenging research area in the past few years. While the systems such as QBIC only support retrieval of still image. Content-based searching system research on video database has not been fully explored yet. Our content-based searching system has the following 3 features:

1) Use of object sequences obtained by automatic object extraction.
2) Use of features of each region within the object
3) Query with multiple objects

In Figure 6, The object sequences represents the following data structure. The data structure of object consists of information of color, rough shape, and moving, and the features of each region within the object. Color information is given the histogram of pixel within the object. And rough shape information is given the maximum height and width along the principal axes of an object. The ratio of this width to this height is the aspect ratio. Moving information is the moving vector or the trajectory of an object. The features of each region within the object are the following 3 items.

1) Average values of pixel (Y,U,V).
2) Two Segmented parameters
(A threshold T and a searching area S).
3) Region ratio = a region area / all region area.

Our content-based searching system consists of rough matching and detail matching processes. As not using template matching, these processes are computationally efficient. First, using the aspect ratio of object and color histogram of pixel within object, the rough matching of objects selects the candidates of objects from objects in all databases. In next matching process the selected candidates are used. Second, detail matching is to compute the distance between the query object *QO* and the rough matched object *RMO* selected by the rough matching in the following equation.

Object distance = W p * H p + W s * H s + W a * H a.

Where as

Hp (*QO*, *RMO*) is the *Hausdorff* distance [10] of average value of pixel. And **Hs** (*QO*, *RMO*) are the *Hausdorff* distance of segmented parameters. And **Ha** (*QO*, *RMO*) is the *Hausdorff* distance of region area ratio.

QO is a vector of the query object, which consist of *N* regions. And *RMO* is a vector of the rough matched object, which consists of *M* regions. **Wp**, **Ws**, and **Wa** are weight value. As the *Hausdorff* distance is computed by min-max operations, the computation cost of our matching using the *Hausdorff* distance is less than template matching.

8. Conclusions

This paper proposed a new approach to automatically extract accurate object from video streams. The new approach provides a useful tool for distributed object-based browsing and content-based searching systems, and consists of a ***skip-labeling*** algorithm for feature-based segmentation, an ***occlusion-killer*** algorithm for estimating accurately optical flow, and a ***shrink-merge tracking*** *algorithm* for tracking an object. The ***skip-labeling*** *algorithm* can be used to segment an image into integrated regions of the same feature. The segmented regions can also belong to such a texture area as waves or forest. The ***shrink-merge tracking*** *algorithm* is executed using morphological image processing, such as dilation and erosion based on the time-continuity of moving-objects. The dilation and erosion repeatedly execute the projection processing in which an object area in next frame is derived from the object area in a current frame. The ***shrink-merge tracking*** *algorithm* can also project the area of a rotating-object in a current frame on the rotating-object containing new regions appearing in the next frame. We demonstrated that proposed object extraction method automatically works satisfactorily for objects non-linearly moving within video streams, and also fully works satisfactorily in 450 frames with a full frame size of 704 x 480 pixels. Finally, this paper described that the object sequences obtained by this object extraction tool is used in a computationally efficient approach for a content-based searching system using the *Hausdorff* distance.

9. References

[1] A. Hiraiwa, K. Fuse, N. Komatsu, K. Komiya. H. Ikeda, "Object Tracking and Creation of Linking Information for Distributed", IS&T/SPIE Conf. on Visual Commun. and Image Processing '99, Vol. 3653, pp. 716-726, Jan. 1999.

[2] A. Hiraiwa, N. Komatsu, K. Komiya. H. Ikeda, "Dynamic Load Balancing for Distributed Movie Based Browser Systems", IEEE International Symposium on Circuits and Systems, June 1, 1998.

[3] Y.Linde, A.Buzo, and R.M. Grey, "An algorithm for vector quantizer design", IEEE Trans. Commun. vol. COM-28, pp.84-95. 1980.

[4] Arther R. Weeks, "Fundamentals of Electric Image Processing". IEEE press, 1996.

[5] Scott E. Umbaugh, "Computer Vision and Image Processing: A Practical Approach Using CVIPtool", Prentice-Hall inc, 1998.

[6] A. Hiraiwa, K. Fuse, N. Komatsu, K. Komiya. H. Ikeda, "Automatic Extraction of Movie-Objects from Video-Stream Data in Distributed Movie-based Browsing System", 1998 IEEE Workshop on Networked Appliances, November 6, 1998.

[7] T. Meier, K. N. Ngan, "Automatic Segmentation of Moving Objects for video Plane Generation", IEEE Trans. on circuits and systems for video technology, vol.8, no.5, pp.525-538 Sep. 1998.

[8] A. Hiraiwa, K. Fuse, N. Komatsu, K. Komiya. H. Ikeda, "Accurate Estimation of Optical Flow for Fully Automated Tracking of Moving-objects within Video Streams", IEEE International Symposium on Circuits and Systems, June 1, 1999.

[9] Shih-Fu Chang, William Chen, Horace J. Meng, Hari Sundaram, and Di Zhong "A Fully Automated Content-Based Video Search Engine Supporting Spatiotemporal Queries", IEEE Trans. on circuits and systems for video technology, vol. 8, no. 5, pp.602-615, Sep. 1998.

[10] D.P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "comparing images using the Hausdorff distance", IEEE trans. Pattern Anal. Machine Intell., vol. 15, pp. 850-863, Sept. 1993.

[11] M. Faloutsos, etc., "Query by image and video content: The QBIC system" IEEE comput. Mag., vol. 28, pp. 23-32, Sep. 1995.

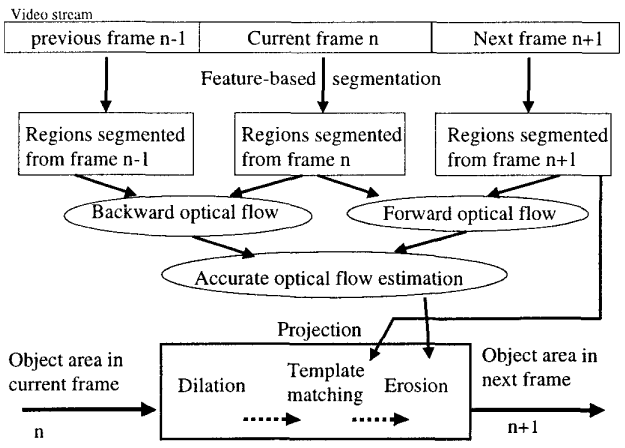


Figure.1 Flow of extracting object from video stream.

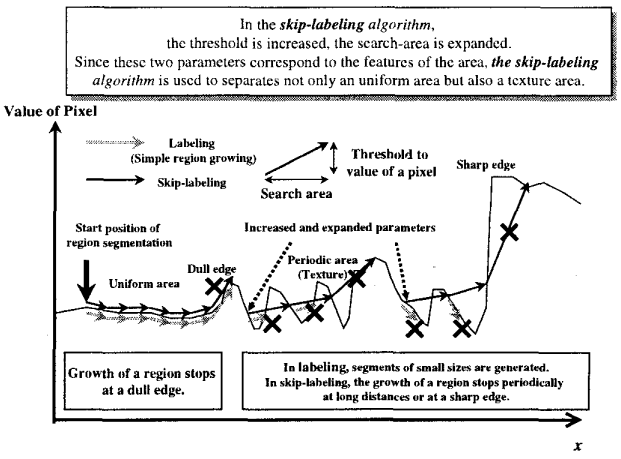


Figure. 2 Skip-labeling algorithm.

(a) Original 704 ×480 image (b) K-mean□ □□□ (c) Labeling (d) Skip-labeling

Figure. 3. Results of region segmentation.

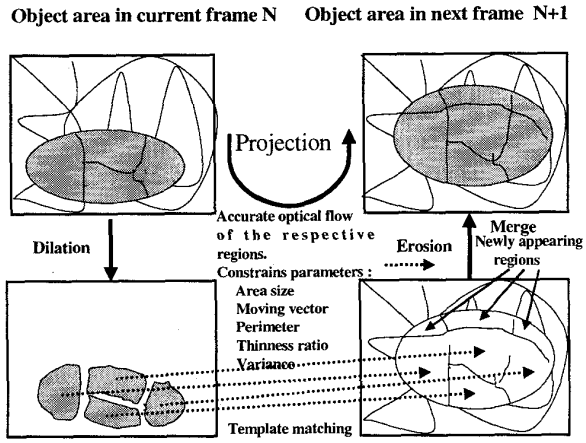


Figure 4 Projection processing in accordance with the *Shrink-merge Tracking* Algorithm.

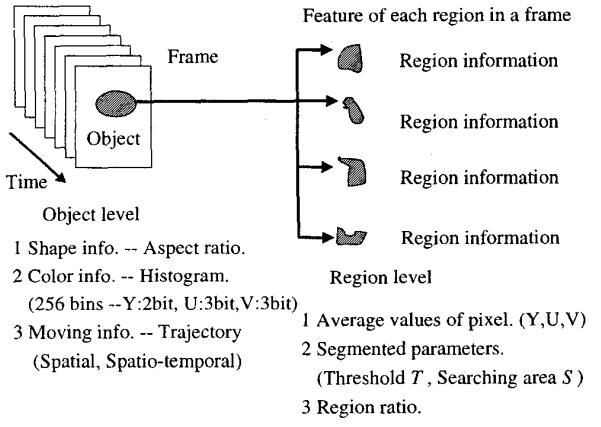
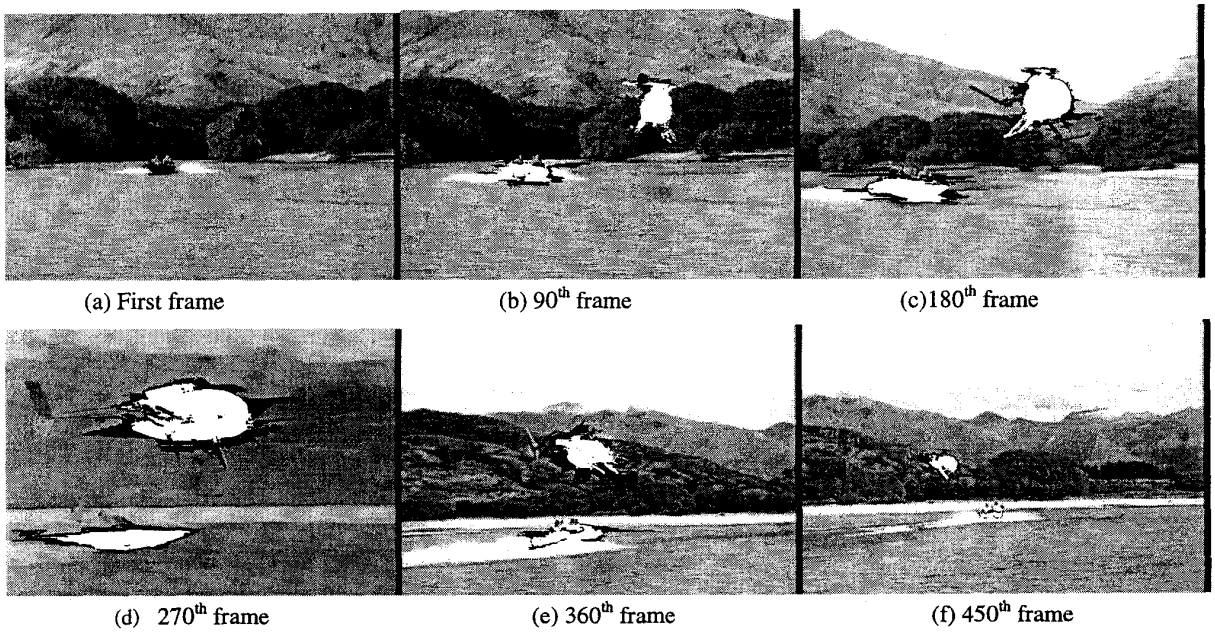


Figure 6 Data structure of an object in content-based searching system.



(a) First frame (b) 90th frame (c)180th frame

(d) 270th frame (e) 360th frame (f) 450th frame

Figure 5. Tracking results of an expanding and rotating boat and an expanding and rotating helicopter under panned background within video streams at full frame size of 704 x 480 pixels using *shrink-merge tracking algorithm.*