MULTIPLE FEATURE CLUSTERING ALGORITHM FOR AUTOMATIC VIDEO OBJECT SEGMENTATION

Wei Wei[1], *King N. Ngan*[1, 2], *and Nariman Habili*[3]

[1]Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong
[2]School of Computer Engineering, Nanyang Technological University, Singapore
[3]iOmniscient Pty Ltd, Suite 202, 29 Albert Avenue, Chatswood NSW 2067, Australia

ABSTRACT

In this paper, we present an automatic video segmentation algorithm for object-based coding based on *k*-medians clustering algorithm and two-dimensional binary model. Firstly, the *k*-medians algorithm is employed to partition an image into a set of homogeneous regions. Then, a two-dimensional binary model of the moving object is set up, which combined with temporal and spatial information, will guide the extraction process of the VOPs from the video sequence. The performance of the segmentation algorithm is illustrated by simulations carried out on standard video sequences.

1. INTRODUCTION

Segmentation of moving objects in a video sequence has many potential applications in wide range of areas, including video surveillance, object detection and tracking, analysis of medical image sequences, and object-based video compression [1]. Although extensive work has been carried out in the fields of scene analysis and motion segmentation, automatic segmentation of arbitrary video sequence into meaningful objects remains a difficult problem.

This paper describes a technique of unsupervised video segmentation based on the *k*-medians clustering method [2] and a binary model. In our proposed method, clustering is first employed to segment the frame of a video sequence into homogeneous regions based on luminance, chrominance, texture, position and motion information. Then, a binary model is derived for the object of interest and tracked throughout the sequence. Temporal continuity of the segmentation is accomplished by matching the model against subsequent frames and updating them accordingly. This allows the proposed method to track fast moving objects as well as to detect the disappearance of existing objects from the scene. The following sections detail the techniques employed in our proposed video segmentation method. The overall block diagram is depicted in Figure 1.

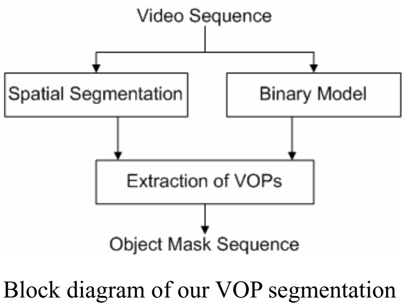


Figure 1. Block diagram of our VOP segmentation algorithm

2. SPATIAL SEGMENTATION

The goal of spatial segmentation is to partition an image into a set of disjoint homogeneous regions whose union is the entire image.

2.1 *k*-Medians Clustering Algorithm

The purpose of clustering is to partition a set of *n* feature vectors $C = \mathbf{u}_1, ..., \mathbf{u}_n$ into k disjoint subsets, $C_1, ..., C_k$. Each subset represents a cluster, with the feature vectors in the same cluster being more similar to each other than to the feature vectors in other clusters. Generally, *C* is partitioned by optimizing some criterion function. The most popular criterion function for clustering is the sum-of-squared-error criterion. Let m_i be the median of those samples,

$$\mathbf{m}_i = \underset{u \in C_i}{median}(\mathbf{u}) \tag{1}$$

Let us consider an N_f - dimensional feature space. The sum-of-squared errors is defined by

$$J_k = \sum_{i=1}^{k} \sum_{\mathbf{u} \in C_i} d^2(\mathbf{u}, \mathbf{m}_i) \tag{2}$$

where

$$d(\mathbf{u}, \mathbf{m}_i) = \|\mathbf{u} - \mathbf{m}_i\| = \left[\sum_{l=1}^{N_f} (u_l - m_{il})^2 \right]^{\frac{1}{2}} \tag{3}$$

0-7803-8484-9/04/$20.00 ©2004 IEEE

J_k measures the total squared error incurred in representing the n samples $\mathbf{u}_1,...,\mathbf{u}_n$ by the k cluster centers, $\mathbf{m}_1,...,\mathbf{m}_k$. The value of J_k depends on how well the samples are grouped into clusters and the number of clusters. The optimal partitioning is defined as one that minimizes J_k.

2.2 Cluster Number Estimation

A fundamental problem with the *k*-medians algorithm is the lack of knowledge to identify the number of clusters present in the data. In our study, the cluster number is estimated by analyzing the behavior of J_k for $k = 1, \ldots, k = k_{max}$ for the first frame. It is clear that J_k, which measures the total squared error incurred in representing the n samples $\mathbf{u}_1,...,\mathbf{u}_n$ by the k cluster centers $\mathbf{m}_1,...,\mathbf{m}_k$, must decrease monotonically as k increases, because the squared error can be decreased each time k is increased merely by transferring a single sample to new singleton cluster. If the n samples are really grouped into $\hat{k}$ compact, well-separated clusters, J_k should decrease rapidly until $k = \hat{k}$, and then decrease much more slowly thereafter until it reaches zero at $k = n$. Based on this property, the true number of clusters must lie at the "corner" or "elbow" of the J_k versus k curve [3].

2.2.1 Initial Cluster Number Estimation

First, only luminance information is used to obtain an estimate of the cluster numbers in the first frame. The *k*-medians algorithm is run for a range of different cluster number $k = 1, 2, 3\ldots, k = k_{max}$ and J_k evaluated after convergence. The starting points for the (k+1)th cluster were derived from the centers of the kth cluster, plus the sample that is farthest from the nearest cluster center. The first k that satisfies the following conditions is designated as the cluster number:

$$\frac{\Delta J_k}{\Delta J_{k+1}} < \theta \tag{4}$$

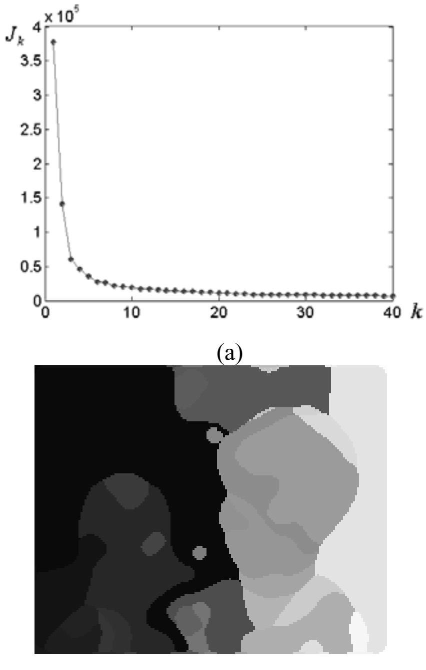
where,

$$\Delta J_k = J_{k-i} - J_{k+i} \tag{5}$$

In our implementation, θ is some predefined threshold and set to 1.2, i is set to 4. Figure 2(a) shows the J_k versus k curve for the first frame of the *Mother & Daughter* sequence. Figure 2(b) depicts the corresponding segmentation field.

2.2.2 Cluster Number Estimation with Multiple Features

There are two problems to be solved before clustering the image. Firstly, the features have quite different ranges of possible values. Thus, normalization of different features should be carried out. Secondly, the features used in the scheme differ not only in their values but also in the level of reliability. It means that different features should be assigned different weights.



(a)

(b)

Figure 2. *Mother & Daughter* sequence: (a) J_k versus k curve. (b) Segmentation fields

The features that we propose to use in our segmentation scheme belong to five groups: luminance, chrominance, texture, position and motion. A common solution is to normalize with respect to the standard deviation over the entire image, which is adopted in [4]. For each feature f_l,

$$m_l = \frac{1}{N}\sum_{q=1}^{N} u_{ql} \tag{6}$$

$$\sigma_l^2 = \frac{1}{N}\sum_{q=1}^{N} (u_{ql} - m_l)^2 \tag{7}$$

$$\hat{f}_l = \frac{f_l}{\sigma_l} \tag{8}$$

for all $l = 1,...,k$, where N is the total pixels in the whole frame.

The method employed to adaptively assign weights to different features is important to the segmentation scheme. A fixed weight is assigned to the position information. The weights assigned to luminance, chrominance and texture features are affected by the variances of the features, which are the medians of the corresponding feature in the region $R_m^{(1)}$, $m = 1,...,M$, obtained by using the *k*-medians clustering algorithm based on luminance information only in the first frame.

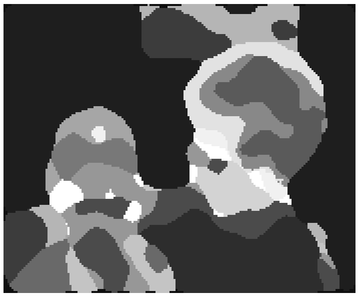


Figure 3. *Mother* & *Daughter* sequence: segmentation field based on multiple features.

$$\bar{f}_{ml} = \frac{1}{M}\sum_{m=1}^{M}(\underset{f_l \in R_m^{(1)}}{median}(f_l)) \qquad (9)$$

$$\sigma_{ml}^2 = \frac{1}{M}\sum_{m=1}^{M}(f_{ml} - \bar{f}_{ml})^2 \qquad (10)$$

$$w_l = \frac{\sigma_{ml}}{\max_N(f_l) - \min_N(f_l)} \qquad (11)$$

where M is the total number of regions in the first frame.

The inclusion of the standardization and weighting terms modifies (3) to:

$$d'(\mathbf{u}, \mathbf{m}_i) = \left[\sum_{l=1}^{N_f} w_l (\frac{u_l - m_{il}}{\sigma_l})^2\right]^{\frac{1}{2}} \qquad (12)$$

The k value will be obtained based on the multiple features, which is the same way as in section 2.2.1. The weight of motion information is computed in the similar fashion as the luminance, chrominance and texture information. The difference between them is that the weight of motion is computed adaptively by using the region information obtained in the previous frame, but the weights of other features are decided in the first frame, unless a scene change occurs. Figure 3 depicts the corresponding segmentation fields of *Mother* & *Daughter* sequence based on multiple features.

3. BINARY MODEL OF MOVING OBJECTS

3.1 Model Initialization

The main assumption underlying our algorithm is the existence of a dominant global motion that can be assigned to the background. The six-parameter affine transformation [5] is normally sufficient to describe the global motion. The least square (LS) method is used to estimate the six parameters by minimizing the sum of the squared residuals. After compensating the background motion, the areas that do not follow this background motion indicate the presence of independently moving physical objects, which are named Independently Moving Components (IMC).

Suppose the model for the object of interest is initialized in frame *i*. Initial model $O_i = \{o_1, ..., o_m\}$ is defined as a set of *m* model points. Similarly, $E_i = \{e_1, ..., e_n\}$ is denoted as the set of all edge pixels detected by the Canny operator in frame *i*. The initial model is given by selecting all edge pixels within a small distance T_{init} of IMC, i.e.,

$$O_i = \{e \in E_i \mid \min_{x \in IMC} \|e - x\| \le T_{init}\} \qquad (13)$$

where x is an element of IMC and $\|\cdot\|$ is the Euclidean distance [6].

3.2 Model Update

The object of interest might rotate or change its shape as it is moving through the video sequence, and as a consequence the corresponding model must be updated every frame. The updated model is given by combining the two components, which are the *Updated Existing Component* O_{q+1}^E and the *Newly Appearing Component* O_{q+1}^N [6]. The purpose of *Updated Existing Component* is to track the binary model of the previous frame using motion compensation and *Newly Appearing Component* is employed to incorporate newly appearing object. The updated model O_{q+1} is given by combining the two components

$$O_{q+1} = O_{q+1}^E \cup O_{q+1}^N \qquad (14)$$

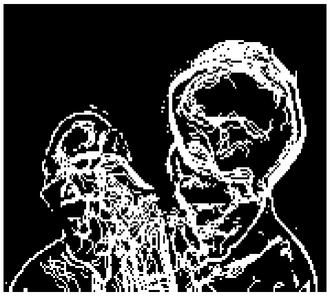


Figure 4. *Mother* & *Daughter* sequence: Updated binary model

4. VOP EXTRACTION

The output of the model update stage is a sequence of binary edge images that model the tracked objects. A two-step process determines the shape of the objects. Firstly, the initial VOPs are obtained using a simple filling-in technique. Secondly, the regions with a homogenous chrominance are matched with the initial VOPs to extract the moving objects from the sequence.

4.1 Filling-in Technique

A simple filling-in technique is employed to determine the initial VOPs. The horizontal candidates are declared to be the region inside the first and last edge points in each row and the vertical candidates for each column. After finding both horizontal and vertical candidates, the intersection regions through the logical AND operation are further processed by the alternative use of morphological operators.

4.2 Region Matching

Let R_i^t be a homogenous region. VOP V_j^t denotes the filled VOP in the current frame *t*. A decision rule of region matching is defined as

$$T_m = \frac{N_{R_i^t \cap V_j^t}}{N_{R_i^t}} \tag{15}$$

where $N_{R_i^t}$ is the number of pixels in R_i^t. If the value of T_m is greater than or equal to a given threshold, the region R_i^t is considered to belong to a moving object; otherwise it is background. In our implementation, T_m is set to 0.6.

After combining all the regions that belong to the moving object, the VOP will then be extracted.

5. SIMULATION RESULTS

Several experiments are carried out on different video sequences in quarter-common intermediate format (QCIF) to test the performance of this VOP segmentation method. Due to the limitation of space, only the segmentation result of *Mother* & *Daughter* sequence will be shown.

In *Mother* & *Daughter* sequences, there are multiple objects required to be extracted. In this sequence, the head of the mother has a relatively large motion, while her body exhibits little motion. The motion of the daughter is little throughout the sequence. Our algorithm is still capable of determining the locations of the moving objects reasonably well, as demonstrated in Figures 6.

Since different foreground object regions are detected and tracked in the sequence, several frames may be needed to extract the complete foreground object. The foreground objects are fully extracted in frame 12 of the *Mother* & *Daughter* sequence.

6. CONCLUSION

In this paper, an automatic VOPs generation method in the context of object-based coding has been presented that continuously separates moving objects in image frames through time evolution. The spatial information is obtained using the weighted *k*-medians clustering algorithm. Binary model is derived for the object of interest and tracked throughout the sequence. The features that have been considered were luminance, chrominance, texture, position and motion information. The weights of the features are determined by the variances in the first frame, except for the position and motion information. The weight of position is constant and the weight of motion is given adaptively by using the region information obtained in the previous frame.

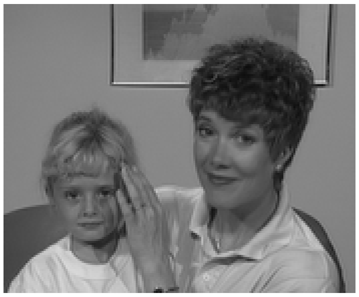


Figure 5. *Mother* & *Daughter* sequence: original frame

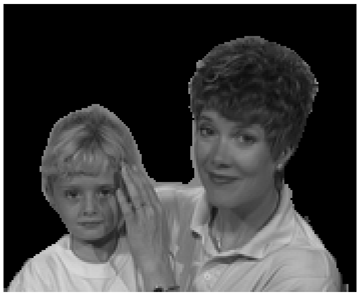


Figure 6. *Mother* & *Daughter* sequence: segmentation result

Experimental results demonstrate that the proposed method is able to successfully extract moving objects from the sequence.

7. REFERENCES

[1] ISO/IEC JTC1/SC29/WG11 N2202 “Information Technology – Coding of Audio-Visual Objects: Visual”, Tokyo, March 1998.

[2] S. Arora, P. Raghavan, and S. Rao, “Approximation schemes for Euclidean k-medians and related problems,” Proc. 30th Annual ACM Symposium on Theory of Computing, 106-113, 1998.

[3] N. Nariman and K. N. Ngan, “Automatic Multi-cue VOP Extraction for MPEG-4,” Picture Coding Symposium 2003, Saint Malo - France, April 2003.

[4] R. Castagno, T. Ebrahimi, and M. Kunt, “Video segmentation based on multiple features for interactive multimedia applications,” IEEE Trans. Circuits Systems for Video Technology, vol. 8, no. 5, pp. 562-571, Sept. 1998.

[5] H. Gharavi and M. Mills, “Blockmatching motion estimation algorithms-new results, ” IEEE Trans. Circuits and Systems, vol. 37, no. 5, pp. 649-651, May 1990.

[6] W. Wei and K. N. Ngan, “Automatic Video Object Segmentation for MPEG-4, ” SPIE Visual Communications and Image Processing (VCIP), Switzerland, Jul. 2003.