An Integrated framework for Robust and Fast Automatic Video Segmentation

Xiaodong CAI, F.H.Ali and E.Stipidis
Communication Research Group
School of Engineering and Information Technology
University of Sussex

Abstract: In this paper, we extend previous work on video segmentation [1,2] and propose a novel integrated module-based framework for real-time applications which require automatic, precise, and fast-based features. The framework consists of five main modules: sprite generator, key frame identifier, change detector, video object plane (VOP) extractor and post-processor.

A universal approach for sprite-object-based adaptive background update and a two-level change detector present the core technique of the framework. The sprite object background provides an easy way to perform segmentation automatically and the possibility of extracting video objects whenever they appear without considering the change of the background. The analysis of statistical parameter for normalized and high order statistics feature offers accurate segmentation results. The used key frames selection technique offers efficient computation. In addition, the framework targets flexible applications with different motion features by using optional modules.

1 Introduction.

The "Object-based" video coding has been a good direction to obtain efficient compression whilst maintaining visual quality for advanced real-time multimedia services. MPEG-4 claims to be "object-based" coding which is embodied mainly in its visual part. To take advantage of the new object-based functionalities defined in MPEG-4, a prior decomposition of video sequences into semantically meaningful objects is required. We have seen rigorous efforts focused on video segmentation where the moving objects are extracted using one or more features, such as motion, colour and spatial information to achieve good results. However, most of these proposed algorithms are less used practically due to one or more of the following constraints: the background is variable due to camera motion; the light conditions slightly change; new objects can appear at any time; objects may remain for a long time in the scene; many objects in a scene and possible occlusion.

In [3], an accurate segmentation was presented but designed for off-line applications. The work in [4] shows very good segmentation results, however it needs manual initialization of segmentation. All these designed approaches are not suitable for real-time automatic applications.

In order to serve real-time automatic applications and address the above problems, we present a novel approach which uses sprite-object-based adaptive background technique and statistical feature analysis-based change detector. In this approach, a sprite generator is used to generate a sprite object, which presents the background without moving objects. The reference frames for change detector can be produced from sprite background objects. A change detector detects the video shot and produces the difference-frame which represents the change between a reference frame and the current frame. The differences are analyzed based on a probabilistic method and the moving objects appearing in a scene at a given time are extracted. A key frame identifier can speed up the segmentation procedure. The segmentation is initialised automatically by using a reference frame which is updated adaptively. Finally, a post-processor is used to obtain more accurate segmented shapes.

2. Framework Description

Fig1 illustrates the structure of the framework. There are mainly five modules: sprite generator, key frame identifier, change detector, video object plane (VOP) extractor and post-processor.

2.1 Sprite generator: A *sprite* is an image composed of pixels belonging to a video object visible throughout a video sequence. For instance, in a panning sequence, portions of the background may not be visible in certain frames due to the occlusion of the foreground objects or the camera motion. The sprite generated will contain all the visible pixels of the background object throughout the sequence. This concept of sprite can be illustrated in Fig2. The video sequence consists of a few frames. These frames continually provide changed background information according to the camera panning. From the provided motion information, the sprite object background is generated and the reference frame can be obtained as a part of the sprite object.

The reference frame extracted from a sprite object can be looked as a " pure background" which can be used to easily identify the uncovered background area, which does not belong to a moving object. It is much easier to detect the overlap of two successive instances than using the simple subtraction of two successive frames. With this technique, the initialization of video segmentation procedure is easy. Also, the still video objects which stay in a scene for a long time can be extracted. In addition, theoretically unlimited number of video objects can be extracted whenever they appear. These advantages are very important for real-time applications.

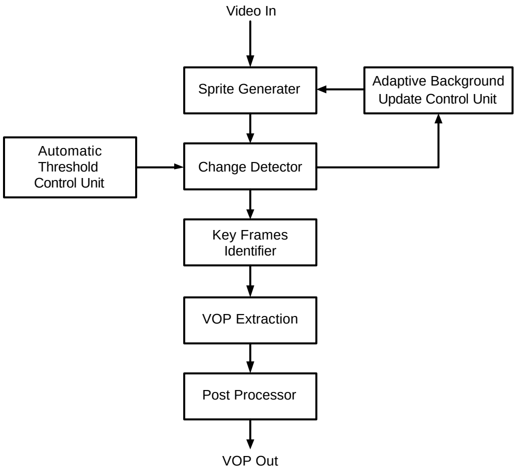


Fig1: Structure of the framework

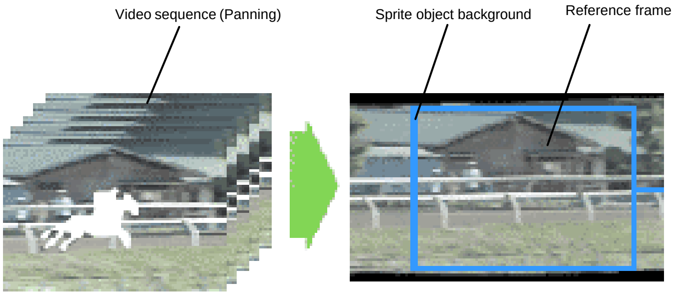


Fig2: Sprite object and background reference frame generation

The method of sprite object compression has been defined in MPEG-4 Verification Model (VM). However, the generation of sprite is still an open issue. Global Motion Estimation (GME) is the main method of sprite generation. In our work, a hierarchical GME algorithm is used to speed up the system and an improved sprite generation algorithm based on MPEG-4 VM is used to obtain better visual quality. The basic steps of the implementation can be found in [5].

2.2 Adaptive background update control unit: The main change of the background of the video sequence can be camera break, backgrounds dissolve and camera motion such as zooming and panning. Adaptive concept is an important issue during the design of this framework. The adaptive reference frame update addresses many problems in other video segmentation approaches. For instance, the framework is suitable in the situations when the background change due to camera motion. Also, it works very well in a light variable environment.

The adaptive background update control unit is activated by video sequence change events such as regular time interval, the detection of camera break, backgrounds dissolve, camera zooming and panning. These events are sent from the change detector.

2.3 Change detector: The change detector detects the differences between the current frame and the reference frame. Two levels of change can be detected. One is the distinct change of the background, which is called change event, and another is the change of the objects moving in a comparatively still background.

In the first level, a change event detector module is designed with a twin-histogram comparison technique to detect camera break and transition. The camera motion, such as panning or zooming is detected based on an optical flow technique. When the background is comparatively still, an automatic thresholding technique is used to discount the effect of noise. In traditional thresholding approaches, the threshold for video has to be tuned manually or empirically according to the video features. To solve this problem, a method in [2] models the noise statistics is adopted here.

Firstly, the background reference frame *B* is taken from sprite generator. The difference-frame *D* can be obtained by the subtraction of the background frame B from the current frame *C*,

$$D_i = | C_i - B_i | \qquad (1)$$

Where $B_i = (b_{yi}, b_{ui}, b_{vi})$ and $C_i = (c_{yi}, c_{ui}, c_{vi})$ are the three components of the *ith* pixel in the reference frame and current frame. In each pixel of the difference frame, statistical parameter$d$ is introduced,

$$d_i = \frac{D_{yi} + N_1 \times D_{ui} + N_2 \times D_{vi}}{b_{yi} + b_{ui} + b_{vi}} \qquad (2)$$

Where constant *N1* and N_2 are both empirically set as 1.2.

The conditional probability distribution of the normalized statistical feature δ_i defined in (2) with a Gaussian distribution can be written as,

$$P(d_i \mid H_0) = \frac{1}{\sqrt{2p}\, s_0} e^{\frac{-(d_i - m_0)^2}{2s_0^2}} \text{ and } P(d_i \mid H_1) = \frac{1}{\sqrt{2p}\, s_1} e^{\frac{-(d_i - m_1)^2}{2s_1^2}} \qquad (3)$$

In the case of H_0, the hypothesis indicates that there is no change in the *ith* pixel. In the case of H_1, it indicates that there is a scene change in the *ith* pixel.

Fig3 illustrates that the change caused by a foreground object is large while the change caused by noise is small and varies only around the mean value of the corresponding pixel in the background reference frames. It also indicates that if a threshold *Th* is used to classify the change, some foreground pixels will be miss-classified as background or vice versa. The probability of this total error can be written as ,

$$p_{er} = p_1 \int_{H0} p(d \mid 1)dy + p_0 \int_{H1} p(d \mid 0)dy \qquad (4)$$

Where p_1 and p_0 are the probability of 1 and 0, which present the object and noise pixels individually. To minimize P_{er}, a likelihood ratio test is used as follows:

$$\mathrm{L(\ y\ )=}\ \frac{p(y|1)}{p(y|0)} \underset{H0}{\overset{H1}{\gtrless}} \frac{p(0)}{p(1)} \qquad (5)$$

For a given error rate by selecting different values of H_0 and H_1, the decision threshold can be obtained automatically from (5).

It has been found that using the forth order variance of the spatially windowed statistical feature for classification can speed up the segmentation procedure and improve the result of the change detection [6]. Here, a 3x3 window centred at each pixel is set in the frame *D* and the 4th-order moments are calculated as following,

$$M_i = \frac{1}{9}\sum_{k\in s_i} d_{yk} \qquad (6)$$

$$s_i^{\,4} = \frac{1}{9}\sum_{k\in s_i}\left(d_{yk} - m_i\right)^4 \qquad (7)$$

Where d_{yk} is the luminance of *k-th* pixel in *D* frames, and S_i represents the windows centred at pixel *i*. When the si^4 is bigger than a given threshold, it indicates the current pixel belongs to the foreground object. A joint decision will be made to identify the current pixel belongs to a background or foreground object according to the result of normalized and high order statistical analysis.

2.4 Key frames identifier: The key frame is defined as a set of images, which consist of the main information of the video sequence. If the segmentation procedure only deals with the key frames, the procedure can be accelerated while the main information is remained.

In the proposed framework, the key frames are identified in two levels. In the first level, the current frame is considered as a key frame in the situation of a change event is active. It indicates that camera breaks or dissolves or camera motions are detected and the current frame carries very important background information. In the second level, one or more key frames will be extracted between last change-event frame and the current frame. In the last situation, a clustering based approach [7] for determining key frames is adopted. In this algorithm, a 16x8 2D HS color histogram in the HSV color space is used to define the similarity between frames *i* and *j*.

$$Sim(i, j) = \sum_{h=1}^{16}\sum_{s=1}^{8} \min\left(H_i(h,s), H_j(h,s)\right) \qquad (8)$$

The detailed steps of this algorithm are illustrated in Fig4 with a block diagram.

2.5 The extraction of VOP's: The area of interest of the moving object are extracted in horizontal and vertical direction and finally VOP is obtained by using the logical AND operation.

2.6 Post-processor: The segmented VOP shape is refined with a morphological closing operation, which produces more compact regions, without adding noise and without altering the original shapes.

3 Conclusion and further work

In this paper, we presented a robust integrated framework for video segmentation. It provides a module-based approach which is used in real-time applications. A very important issue presented in this paper is that we developed the idea that getting the reference frames from sprite object and the background frame can be adaptively updated. This approach can address all the previously identified problems in the introduction and hence, it can be used in many practical situations.

Another presented issue is the flexibility of the designed framework. The module based system design provides flexible and efficient techniques for different applications. Further more, we do not only focus on designing the robust functionalities but also considering the computation cost such that the system can be more practical in particular for the key frame identifying and automatic thresholding.

Research is currently underway to implement this technique for different applications. Further study will be focused on speed issues and complexity analysis.

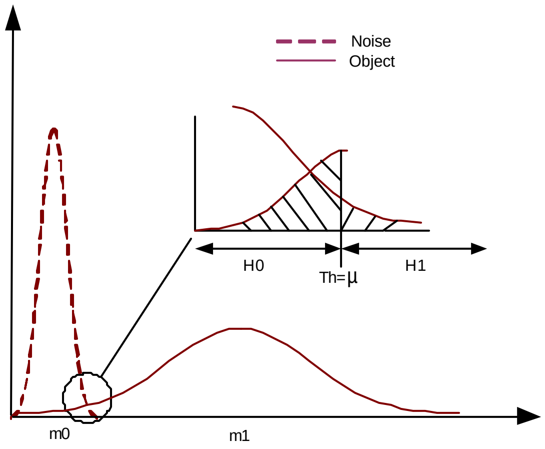
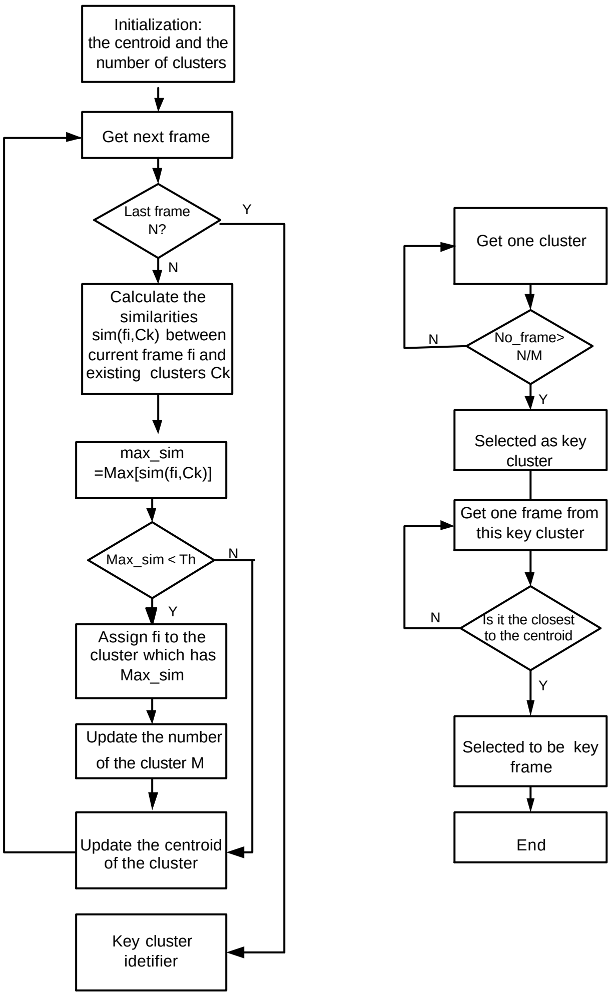


Fig3: Normalized statistics distribution: Change by noise and change by object



Initialization: the centroid and the number of clusters
Get next frame
Last frame N?
Y
N
Calculate the similarities sim(fi,Ck) between current frame fi and existing clusters Ck
max_sim =Max[sim(fi,Ck)]
Max_sim < Th
N
Y
Assign fi to the cluster which has Max_sim
Update the number of the cluster M
Update the centroid of the cluster
Key cluster idetifier
Get one cluster
N
No_frame> N/M
Y
Selected as key cluster
Get one frame from this key cluster
N
Is it the closest to the centroid
Y
Selected to be key frame
End

Fig4: Key frames identifier

4 References

1 C. Kim and J.N. Hwang. “An integrated scheme for object-based video abstraction”. In *ACM*, pages 303-311, October 2000.

2 Jinhui Pan, Chia-Wen Lin, Chung Gu, and Ming-Ting Sun, “A robust spatio-temporal video object segmentation scheme with prestored background information,” *Proc. IEEE Int. Symp. Circuits and System*, May 2002.

3 Correia, Paulo; Pereira, Fernando; “Proposal for an Integrated Video Analysis Framework”, *ICIP'98*, Chicago, October 1998.

4 Di Zhong and Shih-Fu Chang, "AMOS: An Active System For MPEG-4 Video Object Segmentation", IEEE International Conference on Image Processing, October 4-7, 1998, Chicago, Illinois, USA.

5 Yan Lu, Wen Gao, Feng Wu: “ Fast and Robust Sprite Generation for MPEG-4 Video Coding”. *IEEE Pacific Rim Conference on Multimedia 2001*: 118-125.

6 T. Meier and K.N. Ngan, "Segmentation and tracking of moving objects for content-based video coding," *IEE Proceedings - Vision, Image and Signal Processing*, U.K., Vol. 146, No. 3, June 1999, pp. 144-150.

7 Yueting Zhuang, Yong Rui, Thomas S. Huang, and Sharad Mehrotra,” Adaptive Key Frame Extraction Using Unsupervised Clustering “, Proc. of *IEEE Int Conf on Image Processing*, pp866-870, Oct, 1998, Chicago.