Panoramic Video Segmentation Using Color Mixture Models

Abstract

Consider a color camera mounted at a fixed point and a target object to be tracked. We propose a method to replace the common "blue screen" technique in performing segmentation of the object. The target object image in the video sequence is separated from the background through a segmentation process employing the color mixture model techniques. The background information is recovered through a similarity search with the panorama of the original background. The foreground segment can then be encoded by traditional compression while the scene background is represented as a panorama. Finally, the foreground object combines with the corresponding panoramic segment or any desired image on-the-fly to reconstruct the video frame. Our system can be used in low bandwidth applications as well as special demonstration in which the demonstrator can appear/disappear at any time he/she wishes.

1 Introduction

Traditionally, video superimposition techniques always involve the use of color-keying methods. This "blue screen" technique requires special preparation in the studio background, and the foreground people or objects should have no blue colors on the appearance. The actor cannot move out of the studio and thus limited the flexibility in filmmaking. In this work, we propose a new method to replace it in performing segmentation with the help of color modelling technique and a background panorama. We describe a new coding scheme based on a layering concept: a foreground layer with moving objects on top of a background panorama mosaic image of the scene. The background scene mosaic is constructed first. For each frame, the foreground object is segmented and registered using color mixture models. The two layers are handled separately until reconstruction and the background panorama can be replaced by any desired image. Our approach can be considered as a "scenic" instead of "color" or "chroma" keying technique.

Panorama images have been used for video sequence representation [1][2][7][8]. As successive frames usually overlap by a large amount, the use of panorama provides a significant storage size reduction to represent the scene. Color provides many advantages such as tolerance of partial occlusion, resolution and scale distortion etc., over other cues in visual perception. It has been using for segmentation [4], tracking and recognition [3][6]. In our approach, both of the background scene and the foreground objects are modelled using color mixtures.

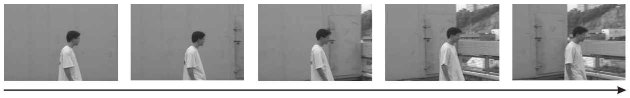

Figure 1. Source video clip 1

This paper is organized as follows. Foreground segmentation and registration are discussed in Section 2. Then we have the video frame reconstruction in Section 3. The result of our experiments, discussions and future directions are in Section 4.

2 Foreground Segmentation & Registration

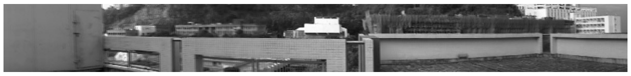

Figure 2. Panorama mosaic sections

Figure 1 shows one of our video clip. The first step is to construct the panorama mosaic for the background layer, as shown in Figure 2, using a set of 16 regular photos. After that, our segmentation algorithm consists of several steps:

1. Color modelling of the foreground object.

2. Color modelling of the i^{th} frame background view.

3. Segment the foreground object from the i^{th} frame by posterior probabilities from the two mixture densities.

4. According to the remaining part, extract the search region out of the $(i+1)^{th}$ frame to search for the next background view from the panorama.

5. Go back to step 2 until all video frames are processed.

2.1 Foreground Object Color Mixture Models

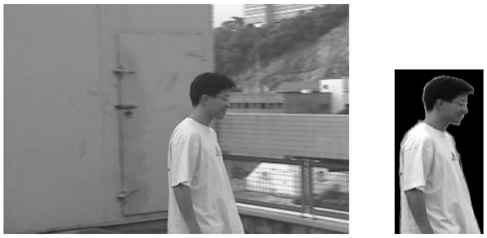

Figure 3. Source frame and foreground object

The foreground object is modelled only once and used for the entire sequence before processing. Figure 3 shows one of the original frame and the foreground object from the video clip 1. This sequence shows the foreground object, a man, walking around in the background scene, the roof of a building. In our work, we use an effective semi-parametric technique, the Gaussian mixture models, for color density estimation. The conditional density for a pixel x, belonging to a class Φ, is modelled as a mixture with n component densities. Then our problem can be viewed as a parametric family of finite mixture densities:

$$p(x|\Phi) = \sum_{i=1}^{n} \alpha_i p_i(x|\phi_i) \tag{1}$$

where α_i is the mixing parameter or weighting factor that corresponds to the prior probability that pixel x is generated by component i, and $\sum \alpha = 1$. Φ is the collection of the mixture ϕ_i. In a 2D Hue-Saturation (HS) color space, each mixture component can be viewed as a Gaussian:

$$p(x|i) = \frac{1}{2\pi|\sum_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_i)^T \sum_i^{-1}(x-\mu_i)} \tag{2}$$

where μ_i and $\sum_i$ are the mean and the covariance matrix of the Gaussian respectively. Now we have to estimate the mixture collection Φ. Expectation-Maximization (EM) is an iterative procedure for numerically approximating maximum likelihood estimates for mixture density problem [5]. The estimation process can be adaptive as well, i.e., ϕ_i is non-stationary and new ϕ_i may appear throughout the sampling process. When a model is once learned by EM, then it can be converted into a look-up table for efficient color probability indexing. The EM algorithm can be summarized as two iterative steps: the expectation E-step and the maximization M-step. Given a current approximation of mixture Φ^c, we can obtain the next approximation Φ^+ from:

E-step: Determine $E(\log f(y|\Phi)|x, \Phi')$

M-step: Choose $\Phi^+ \arg\max_\Phi E(\log f(y|\Phi)|x, \Phi')$

2.2 Background Scene Color Mixture Models

Figure 4. Background from panorama

The color distribution of the background view will be modelled in addition to that of the foreground object I_{obj}. Figure 4 shows the background view. For the i^{th} frame of the sequence, $I_{frame}(i)$, we have the corresponding background viewing window of the panorama, $I_{pano}(\phi)$, at panning view angle ϕ. We assume the correct background view $I_{pano}(0)$ of the first frame $I_{frame}(0)$ is given and with a panning view angle $\phi_0 = 0$. Now we can obtain the color mixture model of the first background view $I_{pano}(0)$, using the same methods as stated in the previous section. Given color density estimates for both the foreground object, I_{obj}, and the background view $I_{pano}(\phi)$, the probability that a pixel x belongs to the foreground object can be calculated by the posterior probability $P(I_{obj}|x)$:

$$\frac{p(x|I_{obj})P(I_{obj})}{p(x|I_{obj})P(I_{obj}) + p(x|I_{pano}(\phi))P(I_{pano}(\phi))} \tag{3}$$

Therefore, a pixel x will be classified as the class with the maximum $P(I_{obj}|x)$ and the minimum mis-classifying probability. That is, a pixel x with $P(I_{obj}|x) > 0.5$ will be classified as foreground object and vice versa:

$$x \in \begin{cases} I_{obj} & \text{if } P(I_{obj}|x) > 0.5 \\ I_{pano}(\phi) & \text{otherwise} \end{cases} \tag{4}$$

The foreground regions $I_{fore}(i)$ are thus extracted. Figure 5 shows the extracted foreground regions in the left. Since the background view is changing throughout the sequence, we have to find some way to obtain the correct background view for every remaining frame.

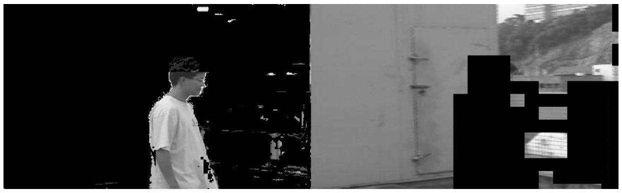

Figure 5. Segmentation results

2.3 Searching the Next Background View

We assumed the camera is fixed with horizontal panning. The frame rate is fast enough such that the background regions of two consecutive frames are more or less the same. Then we can ”cut” the searching region from the next frame, according to the previous one. Figure 5 shows the segmented remaining region on the right, which are bounded by larger blocks. We denote the searching region as a template region $TR(I_{frame}(i))$, which is cut out from $I_{frame}(i)$ according to the remaining part of segmentation of $I_{frame}(i-1)$. Then we have a minimization problem of E_i in the Hue-Saturation-Intensity (HSI) color space:

$$E_i(\phi_i) = [TR(I_{frame}(i)) - TR(I_{pano}(\phi_i))]^2 \quad (5)$$

where ϕ_i is the new panning view angle of the panorama at the i^{th} frame following a small update $\delta\phi_{i-1,i}$. At an optimal $\delta\phi_{i-1,i}$, the difference between the background region of the video frame and the panoramic view would be minimized. The segmented foreground $I_{fore}(i)$ will be registered by the changes in the corresponding panoramic panning view angle $\delta\phi_{i-1,i}$. When the correct background view is found, it will be modelled again and followed by the segmentation process. The result of the segmentation in turn can be used to find the next background view. These procedures are continued until all of the frames are processed. Finally, the foreground sequence will be compressed by MPEG-1 coding.

3 Video Stream Reconstruction

For reconstruction, we first decode the foreground frame sequence and get every $I_{fore}(i)$ back. The background scene is obtained from the corresponding view of the panorama mosaic with cylindrical projection as $I_{pano}(\phi_i)$ according to the panning angle ϕ_i. The viewer simply render the foreground object segments $I_{fore}(i)$ over the background scene from panorama to reconstruct the original frame as shown in Figure 6. This can be done efficiently with hardware acceleration. In addition, our system provides a simple and effective solution for video indexing. The user can access a specific frame by providing the scene information, i.e., indexing through various panning angle ϕ. This approach can be considered as a complement to the content-based (color and texture) indexing method.

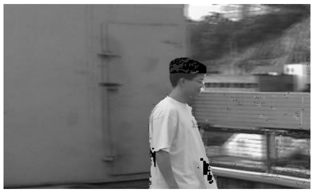

Figure 6. Reconstructed video frame

4 Experimental Result & Discussion

We have captured two video clips with 60 frames in each clip. The first one has been shown in Figure 1. Figure 7 shows the second one with its segmentation results.

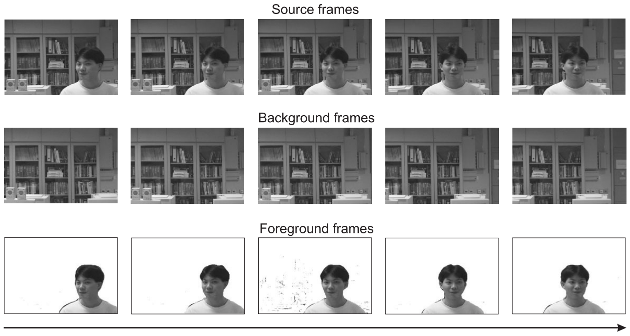

Figure 7. Video clip 2

4.1 Video Compression Performance

Table 1 shows the resulting storage sizes of different components involved in our system of video clip 1 and video clip 2. A partial mosaic image of the background scene is used in each of the experiments. For video clip 1 in Figure 1, the compression ratio is:

$$CR_{1,pano} = 1 - \frac{[V_{1,pano}]}{[V_{1,ori}]} = 1 - \frac{313.3}{21873} = 98.57\% \quad (6)$$

Similarly for video clip 2 in Figure 7, the compression ratio is 98.49%. In general, CR depends on the scene complexity. Moreover, for a longer video clip, the overhead of the size of the mosaic image is relatively small and results in a better compression ratio.

Table 1. Storage sizes of video clip 1 and 2

Items	Size kb (V_1)	Size kb (V_2)
Original video $\|V_{ori}\|$	21873	21873
Panorama image $[I_{pano}]$	21.8	32.2
Foreground $[V_{fore}]$	291.5	329.2
Pano-encoded $[\mathbf{V_{pano}}]$	$\mathbf{313.3}$	$\mathbf{361.4}$

Table 2. Means of HSI differences - clip 1

I_a	I_b	$\frac{1}{60}\sum \overline{E}_{HSI}(I_a, I_b)$	
Original	Pano-encoded	0.0668 (V_1)	0.1231 (V_2)
MPEG-1	Pano-encoded	0.0598 (V_1)	0.1190 (V_2)
Original	MPEG-1	0.0290 (V_1)	0.0806 (V_2)

4.2 Quality of Reconstructed Video Sequence

The measurement of image quality is based on the absolute difference of each pair of corresponding pixels, in the Hue-Saturation-Intensity (HSI) color space. For two pictures I_a and I_b, we define the absolute difference between a pair of corresponding pixels $I_a(x, y)$ and $I_b(x, y)$ be $E_{HSI}(I_a(x, y), I_b(x, y))$, which is normalised to fall within the range $[0, 1]$. For the entire images with $m \times n$ pixels, we define the normalized average difference $\overline{E}_{HSI}(I_a, I_b)$ between them by:

$$\overline{E}_{HSI}(I_a, I_b) = \frac{1}{mn}\sum_{x,y=1}^{m,n} E_{HSI}(I_a(x, y), I_b(x, y)) \quad (7)$$

Table 2 shows the overall means of the normalized average differences $\overline{E}_{HSI}$ of video clip 1 and video clip 2, among the uncompressed original video sequence, the MPEG-1 compressed video sequence and the reconstructed video sequence from our system.

4.3 Superimposition from New Background

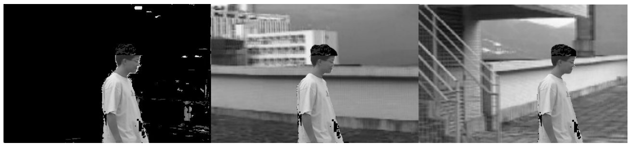

Figure 8. Applying new background

By replacing the original panorama, we can synthesize various virtual environments as shown in Figure 8. Our system can also provide certain interesting features, like interactive controls on panning view angle and zoom factor, to explore the whole scene or examine details of any particular frame. We may also allow zooming and vertical panning of camera motion. However, these modifications will lead to problems in the simulation of out-focusing (depth of view) effects of the background, and the estimation of zooming factor and the vertical panning angle of the camera. Another promising direction is to investigate how to handle complex scenes with multiple dynamic foreground objects.

5 Conclusion

We have presented a method to replace the common blue color-keying technique in performing superimposition with the help of a background panorama. A video stream is decomposed by color-based segmentation, and represented as a combination of background panorama and foreground objects. During reconstruction, the foreground segments are combined with their corresponding views in the background panorama, which can also be replaced by any desired image to perform superimposition. Our experiments demonstrate the effective results.

References

[1] M. Irani, P. Anandan and S. Hsu, “Mosaic Based Representations of Video Sequences and Their Applications”, Proc. of ICCV ’95, pp. 605-611, June 1995.

[2] K.S. Lee, Y.F. Fung, K.H. Wong, S.H. Or, T.K. Lao, “Panoramic Video Representation using Mosaic Image”, Proc. of CISST’99, pp. 390-396, Las Vegas, USA, June 1999.

[3] J. Matas, R. Marik and J. Kittler, “On Representation and Matching of Multi-coloured Objects”, Proc. of ICCV ’95, pp. 726-732, June 1995.

[4] Y. Raja, S. Mckenna and S. Gong, “Segmentation and Tracking Using Colour Mixture Models”, Asian Conference on Computer Vision, ACCV’98, Hong Kong, 1998.

[5] R. A. Redner and H. F. Walker, “Mixture Densities, Maximum Likelihood and the EM Algorithm”, SIAM Reviews, 26(2): pp. 195-239, 1984.

[6] M. J. Swain and D. H. Ballard, “Colour Indexing”, IJCV, pp. 11-32, 1991.

[7] R. Szeliski, “Image Mosaicing for Tele-reality Applications”, Technical Report CRL 94/2, Digital Equipment Corp., 1994.

[8] J. Y. A. Wang, E. H. Adelson and U. Desai, “Applying mid-level vision Techniques for Video Data Compression and Manipulation”, Proc. of the SPIE, vol. 2187, San Jose, Feb 1994.