Video segmentation of unknown static background using min-cut

Yair Movshovitz under supervision of Prof. Shmuel Peleg
yairmov@cs.huji.ac.il

December 8, 2007

Abstract

In this report we introduce an algorithm for foreground layer extraction based on EM learning and min-cut. The background is unknown but assumed to be static, and the foreground is therefore defined as the dynamic part of the frame. From a single video stream our algorithm uses color cues as well as information from image contrast, that is, the color differences between adjacent pixels, to cut out the foreground layer. Experimental results show that the accuracy is good enough for most practical uses.

1 Introduction

Layer extraction (i.e. the segmentation of a video sequence into *foreground* and *background*) has many uses, such as traffic control [6], video surveillance [7] etc'. In order for layer extraction to work in real life situations it must be robust enough to handle a variety of video features (e.g. a foreground composed of a single component or multiple components, small movements in the background, illumination changes, etc'). Another real life limitation is that the background image may not be known in advance, but instead needs to be constructed using the data available in the video.

This report describes a system that segments video sequences containing dynamic foreground and an unknown, static background. The system has been developed by the author as part of 'guided work' that took place during the spring semester of 2007 under the supervision of Prof. Shmuel Peleg at the School of Engineering and Computer Science of the Hebrew University of Jerusalem. Similar systems for *known* background have been developed [1, 8]. The goal of this work was to achieve good segmentation when the background was unknown.

This report is organized as follows. Background model is explained in Section 2. Section 3 describes the foreground model. In section 4 we talk about the segmentation process. Section 5 shows some experimental results, and section 6 ends the report with possible extensions to the algorithm.

2 Background Model

Let I_r be the 3 dimensional vector of the color values of pixel r in the current frame. A common way of modeling the background layer is using a per-pixel Gaussian model [1] where each pixel r is assumed to be normally distributed over time above the RGB space with mean color μ_r and variance σ_r^2:

$$P_b(I_r) \sim N(\mu_r, \sigma_r)$$

The mean color, μ_r, is the color of r in the background image, if it is known. Otherwise, $\mu_r = \frac{1}{n}\sum_{i=1}^{n} I_{r,i}$. $\sigma_r^2 = \frac{1}{n}\sum_{i=1}^{n} I_{r,i}^2 - \mu_r^2$, where i is the frame number. This technique is useful under the assumption that each pixel in the image represents the background for a significant amount of time (or that the background is known). This assumption can fail under various conditions, e.g., if the foreground contains a large object which moves through a small portion of the frame. To generalize this method, in the learning of μ_r and σ_r^2, we only included frames which showed a significant amount of movement for

the pixel r. We estimated the motion of pixel r at frame i as

$$f_r^i = \frac{I_t}{\sqrt{I_x^2 + I_y^2}}$$

where I_t, I_x, and I_y are the temporal and spatial derivatives at pixel r respectively. We then used this motion estimate to weigh the importance of the data. We defined a damping factor $w_r^i = e^{-K \cdot f_r^i}$ and estimated the parameters of the normal distribution as

$$\mu_r = \frac{1}{\sum w_r^i} \sum_{i=1}^{n} w_r^i \cdot I_{r,i}$$

and

$$\sigma_r^2 = \frac{1}{\sum w_r^i} \sum_{i=1}^{n} w_r^i \cdot (I_{r,i} - \mu_r)^2$$

K is a parameter used to control the strength of the damping. Usually it was used as $k = 5$ but while performing the experiments it showed robustness to changes in the range $5 - 50$.

3 Foreground Model

While the background was modeled as a per pixel probability, i.e., each pixel is distributed as a single Gaussian, we modeled the foreground globally using a Gaussian Mixture Model:

$$P_f(I_r) = \sum_{k=1}^{L} w_k \cdot P_G(I_r | \mu_k, \sigma_k)$$

where P_G is a normal distribution and w_k, μ_k, σ_k^2 are its weight, mean and variance. L is the number of components in the GMM. In our implementation we used a version of EM that uses component annealing, L is initially set to a relatively large value (we used $L = 10$) and trivial components are discarded during the learning.

When modeling the foreground the outcome of the background learning process can be used. We used the per pixel probability parameters learnt in the background model stage to estimate the probability of pixel r at frame i to be part of the background. Pixels which had low probability were marked as definitely foreground and then used as a data set for the learning of the mixture model. The GMM is learnt using EM.

4 Segmentation process

The foreground extraction problem can be viewed as a labeling procedure, where each pixel r should be labeled as either *foreground (1)* or *background (0)* . In [3] it was shown that the labeling problem can be solved using min-cut if it is stated as an energy minimization problem. To solve the problem we created a graph as follows: we defined a source and a sink (foreground and background) and for each pixel r in the image we defined a node N_r. Each node is connected to the source and to the sink, There is also a connection between each two nodes that represent neighboring pixels in the image (4 neighbors). For each pixel r we defined a *color term:*

$$ColorTerm(x_r) = \begin{cases} -logP_f(I_r) & x_r = 0 \\ -logP_b(I_r) & x_r = 1 \end{cases}$$

The color term is the cost of assigning r with the label x_r. In min-cut notation this is the cost of removing the edge between N_r and the source/sink. For each two adjacent pixels r, s we also defined a *contrast term*:

$$\text{ContrastTerm}(x_r, x_s) = |x_r - x_s| \cdot e^{-\beta \cdot d_{rs}}$$

where $d_{rs} = ||I_r - I_s||^2$ is the L_2 norm of the color difference and $\beta = (2 \cdot \text{E}[||I_r - I_s||^2])^{-1}$. The contrast term is the cost of assigning r with the label x_r and s with x_s. This is the cost of preforming the cut in the graph between N_r and N_s. The desired labeling is then achieved using the implementation of the min-cut algorithm in [2].

5 Results

All our videos were taken from standard, off the shelf, cameras. Most were taken from webcams found on

Figure 1: Segmentation results of typical frames. Each section is taken from a different sequence. The video of the girl shows a perfect segmentation result, the other two are lesser in accuracy but still good enough for most practicle uses.

the internet.[1] Figure 1 shows the segmentation results for typical frames from some of the video sequences used. The reader is encouraged to look at the full length segmented videos[2] .The segmentation process produced good results on various types of sequences, mainly on videos with sufficient length in order for the learning process to have a good data set, and on videos which satisfy the main assumptions, i.e., static background and a dynamic foreground. When experimenting we found that, while the system is quite robust for small illumination changes, errors in labeling can occur when shadows are involved. Figure 2 shows a frame in which shadows create a problem. The reason for the mistakes in labeling is that shadows change the pixel color values and therefore are considered as a moving object, in section 6 we talk about a possible solution to this problem.

[1] Billiard club - http://216.254.68.223/view/index4.shtml?newstyle=One&cam=2
Stuttgart Airport - http://195.243.185.195/view/index.shtml

[2] for supplementary material visit http://www.cs.huji.ac.il/~yairmov/videoSeg.html

Figure 2: Shadows can cause mistakes in pixel labeling.

6 Conclusions and Extensions

In this report we have described an algorithm for extracting dynamic foreground from unknown, static, background. Our method uses cues obtained from pixel colors as well as cues from image contrast. This method has proven to produce good segmentation results.

The algorithm has difficulties when confronted with some types of video sequences. Shadows can cause labeling mistakes as shadowed pixels exhibit properties that resemble movement. There are a number of techniques for shadow detection ([4, 5] just to name a couple) and these can be used to identify and correct the labeling of such pixels. Another difficult type of videos is one where a large foreground object dominates the scene and moves little compared to its size. This might be corrected by a better use of temporal cues. Currently, temporal information is only used in the learning process - the background model is learnt using the entire frame set of the sequence. By enforcing temporal coherence (could be done by adding a *temporal term* to the segmentation process , that is, creating nodes to represent past and future color values of each pixel) a better labeling might be achieved.

References

[1] Jian Sun, Weiwei Zhang, Xiaoou Tang and Heung-Yeung Shum. Background Cut. ECCV 2006, Vol. 2, pp. 628-641

[2] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In Energy Minimization Methods in CVPR, 2001.

[3] Y. Boykov and M. Pi. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In Proceedings of ICCV, pages 105–112, 2001.

[4] Andrea Prati, Rita Cucchiara, Ivana Mikic, Mohan M. Trivedi, "Analysis and Detection of Shadows in Video Streams: A Comparative Evaluation," cvpr, p. 571, 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01) - Volume 2, 2001.

[5] P.L Rosin and T. Ellis. Image difference threshold strategies and shadow detection. In Proceedings of the 6th British Machine Vision Conference, pages 347–356. BMVA Press, 1995.

[6] Dieter Koler, Joseph Weber, and Jitendra Malik, "Robust Multiple Car Tracking With Occlusion Reasoning" (January 1, 1994). California Partners for Advanced Transit and Highways (PATH). Working Papers: Paper UCB-ITS-PWP-94-1.

[7] W.E.L. Grimson, C. Stauffer, R. Romano, L. Lee, "Using Adaptive Tracking to Classify and Monitor Activities in a Site," cvpr, p. 22, 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'98), 1998

[8] Howe, Nicholas R.; Deschamps, Alexandra, "Better Foreground Segmentation Through Graph Cuts," ArXiv Computer Science e-prints, 2004, Provided by the Smithsonian/NASA Astrophysics Data System