A Feature Point Matching Based Approach for Video Objects Segmentation

Yan Zhang, Zhong Zhou, Wei Wu
State Key Laboratory of Virtual Reality Technology and Systems, Beijing, P.R. China
School of Computer Science and Engineering, Beihang University, Beijing, P.R. China
{zy, zz, wuwei}@vrlab.buaa.edu.cn

Abstract

In this paper, we propose an approach to segment the multiple objects in video. For a video sequence with stationary background, our approach combines the feature points with the color and contrast information to extract the multiple objects of different sizes. The idea is that the local features of the feature points are more robust than that of the pixels, and more accurate than the global color feature. So we integrate the local cues of the feature points into the basic color model in graph cut. Our method matches the feature points in the known background and the current image, and classifies them in three categories. Then the influences to their neighbor pixels are computed according to the category, and integrated in the pixels' color model. The max-flow algorithm is applied to obtain the last result of the segmentation. Experimental results demonstrate the effectiveness of our approach.

1. Introduction

Finding the moving objects in the image sequences is one of the basic tasks in computer vision. The common applications in computer vision including video surveillance, visual tracking, crowd density estimation, traffic flow analysis, intelligent user interfaces, need to detect the moving objects first. And the detection and segmentation of the moving objects with a stationary camera is widely useful in most applications, which is focused in this paper.

Background subtraction is usually used to deal with the problem, which detect the foreground objects by comparing the background image with the current image. But there are two issues hard to deal with: 1) the threshold to decide the background/foreground objects is sensitive to noise, for there are luminance changes in the real scenarios. 2) the color of the background and foreground objects may be similar, which could result in the holes in the foreground objects.

1.1 Related work

Some more complex methods have been proposed by modeling the background. The existing moving objects detection techniques can be classified into two categories: region-based methods and pixel-based methods. The first category methods always divide the image into many regions, and process them individually to avoid large amount of computation [1] [2]. But these approaches can not make an accurate segmentation for their results' unit is block, which is coarser than pixel. In the methods of the second category, the single Gaussian model [3] or the Gaussian Mixture Model [4] only use the color as the feature in the model, which can not deal with the problem caused by similar color of the background and foreground objects as mentioned above. Meanwhile, the same problem appears in [5] [6] [7], which just use color feature in their models. Although [8] combines the motion and color features in the background model, the motion feature is not helpful to the segmentation. In [9], the feature points are used to represent the foreground objects, but they can not make up the complete boundaries of the foreground objects.

As an important method in layer extraction, graph cut has been researched and applied in many applications in recent years[10] [11] [12]. Since graph cut combines the color and the contrast cues, it could obtain a more precise result than the models which just use color as the feature. In [11] and [13], the color term of the background is modeled integrating the global color model and pixel-based color model, and then graph cut is used to finish the segmentation. The above model is regarded as the basic model in this paper. Although it usually makes good segmentation results, when there are strong edges in the background, and the foreground objects are near the edges accidentally, the segmentation would always be attracted by the strong edges, and make a wrong boundary. So the contrast of the known background is used to attenuate the strong

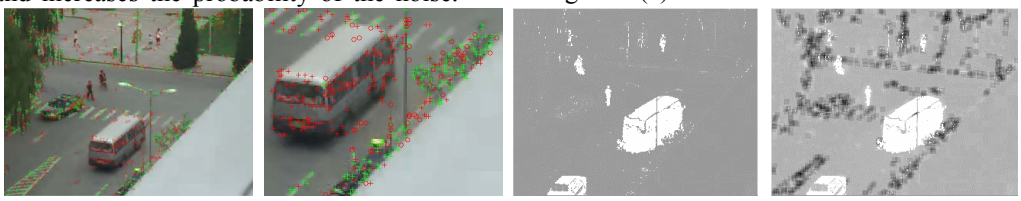
edges in the background to resolve the problem [14].

Although the above graph cut methods have been applied in single foreground object segmentation, and got more or less satisfied results, our aim is to segment the multiple objects. It is not easy when there are multiple objects of different sizes, complex color and the discrepancy of the contrast on their boundaries. When we get the precise boundary on one object using a fixed contrast weight, it may not be proper for the other objects and made a wrong segmentation. So a more robust method is needed to solve the problem.

1.2 Our approach

In this paper, we propose a feature points matching based approach to deal with the above problem. The novel component is the feature points matching between the background image and the current image, and then the information could be combined into the color model to suppress the noise and increase the pixels' probability as their right state (background/foreground object). In the basic model and the background cut model, the color term is mixed by the global color model and the pixel-based model, which represent the global color of the background/foreground objects and the color of the single pixel respectively. When it is used in multiple objects segmentation, there are two issues: (1) the global model usually contains ten to twenty Gaussian models. Although it is often enough for single object, it may not sufficient for multiple objects for they have more kinds of color, which makes the decrease of the robustness, and increases the probability of the noise. (2) different objects always have different sizes in image, and also different contrast strength. So it may be impossible to use the unified parameters to control the segmentation for all the objects and produce a satisfied result. The adjustment for different regions is needed to achieve the goal.

The feature points and their descriptors can represent the information of the local patches, which could be combined with the global color model and pixel-based color model. If we match the feature points of the known background and the current image, as shown in Figure 1 (a), there would be three class feature points: 1) the green cross ones, which represent the matched feature points. 2) the red cross ones, which only appear in the current image, but not in the background image. And vise versa 3) the red circle ones. The first class feature points mean the pixels around them have a high probability as the background, while the second and the third class feature points usually mean the high probability as the foreground objects. As demonstrated in Figure 1 (a), although there are some exceptions of above assumption, the global distribution of the feature points agrees with our assumption. There are more first class (green) points in the background patches, and more second and third classes (red) points around the foreground objects. The distribution could more or less encode the real state of the pixels as background or foreground objects. In Figure 1 (c) there is some noise, which has been suppressed in Figure 1 (d). The dark areas mean the high probability as background, which are computed using the result of the feature points matching as shown in Figure 1 (a).



(a) (b) (c) (d)

Figure 1. (a) Result of feature points matching. (b) The magnified patch near the bus in (a). (c) The color probability of the current image in basic color model. (d) The color probability in our color model.

The paper is organized as follows. In Section 2, we give notations and introduce the basic model and background cut model. In Section 3, we present our approach – feature points based matching model. Experimental results are shown in Section 4. Finally, we discuss the limitations of our current approach and give conclusion in Section 5.

2. Basic model and background cut

Let us denote the known background image by I_B. And I is the current image to be processed. The image I can be expressed as an array $I = (I_1, \ldots, I_n, \ldots, I_N)$. Foreground/background segmentation can be posed as a labeling problem, which is finding an array of "opacity" values $X = (X_1, \ldots, X_n, \ldots, X_N)$ at each pixel. In the hard segmentation problem, $X_n \in \{0,1\}$, with 1 for foreground and 0 for background. The labeling X can be obtained by minimizing a Gibbs energy E(X) [15]:

$$E(X) = \lambda \cdot U(X) + V(X) \quad (1)$$

where

$$U(X) = \sum_{n \in N} U(x_n) \quad (2)$$

$$V(X) = \sum_{\{n,m\} \in C} V_{\{n,m\}} \quad (3)$$

U(X) is the color term, representing the cost when the label of the pixel n is x_n. And V(X) is the smoothness (contrast) term, encoding the cost when the labels of adjacent pixels m and n are x_m and x_n respectively. C is the set of pairs of neighboring pixels. The parameter λ balances the influences of the two terms.

2.1 Basic model

The model in [13] is regarded as the basic model in this paper. The likelihoods for color in foreground are modeled using Gaussian mixtures in RGB color space, which is a global model. And the color model of the background is mixed by a global Gaussian mixture model and a per-pixel single Gaussian model.

For the foreground color model, the energy $U(x_n)$ is defined as:

$$U(x_n) = -\log p(I_n \mid x_n = 1) \quad (4)$$

where

$$p(I_n \mid x = 1) = \sum_{k=1}^{K_f} w_k^f N(I_n \mid \mu_k^f, \Sigma_k^f) \quad (5)$$

There are K_f Gaussian components in the foreground model, and μ_k^f, Σ_k^f, and w_k^f are the mean, variance and weight of the kth model respectively. The parameters could be learned from the foreground region which is labeled in advance or detected using a conservative threshold in background subtraction.

For the background color model, $U(x_n)$ is defined as:

$$U(x_n) = -\log p(I_n \mid x_n = 0) \quad (6)$$

where $p(I_n \mid x = 0)$ is mixed by a global color model and a pixel-based color model.

$$p(I_n \mid x = 0) = \alpha \cdot \sum_{k=1}^{K_b} w_k^b N(I_n \mid \mu_k^b, \Sigma_k^b) + (1-\alpha) \cdot N(I_n \mid \mu_n^b, \Sigma_n^b) \quad (7)$$

K_b, w_k^b, μ_k^b and Σ_k^b are the parameters of the global background color model, like the foreground color model. In the latter term, μ_n^b and Σ_n^b are the mean and variance value of the single Gaussian model at pixel I_n. And α is the parameter to adjust the weight between the global model and the pixel-based model.

The contrast energy $V_{\{n,m\}}$ is defined as:

$$V_{\{n,m\}} = \delta(x_m, x_n) \cdot \exp(-\beta \cdot \| I_m - I_n \|^2) \quad (8)$$

where δ is an indicator function.

$$\delta(x_m, x_n) = \begin{cases} 1 & if\ x_m \neq x_n \\ 0 & otherwise. \end{cases} \quad (9)$$

And β is a robust parameter which can be set to $\beta = (2 \cdot E(\|I_m - I_n\|^2))^{-1}$, and E(X) computes the expectation of X.

2.2 Background cut

Since the basic model does not discriminate the contrasts belonged to the background or to the foreground, the result of segmentation is prone to being attracted by the strong background contrasts, instead of the contrasts across the background/foreground boundaries. So [14] proposes an approach to attenuate the contrasts in the background, while preserving the contrasts of the background/foreground boundaries.

The color energy U(X) is as the same as that in the basic model. But contrasts of the background are combined in the energy $V_{\{n,m\}}$. Since the known background image is static, the color of pixels m and n could be obtained, which are denoted by I_m^b and I_n^b. Then $V_{\{n,m\}}$ is defined as:

$$V_{\{n,m\}} = \delta(x_m, x_n) \cdot \exp(-\beta \cdot \| I_m - I_n \|^2 \cdot \theta_{nm}) \quad (10)$$

where

$$\theta_{nm} = (1 + \frac{\| I_n^b - I_m^b \|^2}{K} \exp(-\frac{z_{nm}^2}{\sigma_z}))^{-1} \quad (11)$$

and

$$z_{nm} = \max\{\| I_n - I_n^b \|, \| I_m - I_m^b \|\} \quad (12)$$

The parameter K is used to adjust the attenuation strength. And σ_z could determine the amount of the contrast left in the contrast image.

3. Feature points matching based objects segmentation

Although the background cut method usually produces quite good result in single foreground object segmentation, there are still some problems in the multiple objects segmentation in outdoor video sequences. 1) Multiple objects always have more flexible colors than single object, so the foreground color model's probability may not be accurate. 2) There are more kinds of the objects in the outdoor scenario which will make strong noise than in the indoor environment, such as the trees. 3) Multiple objects may have different contrast on their boundaries, so it is hard to find the appropriate parameters for all the objects to be segmented along the boundaries, while avoid the affection of the noises. It is more or less as shown in Figure 2.

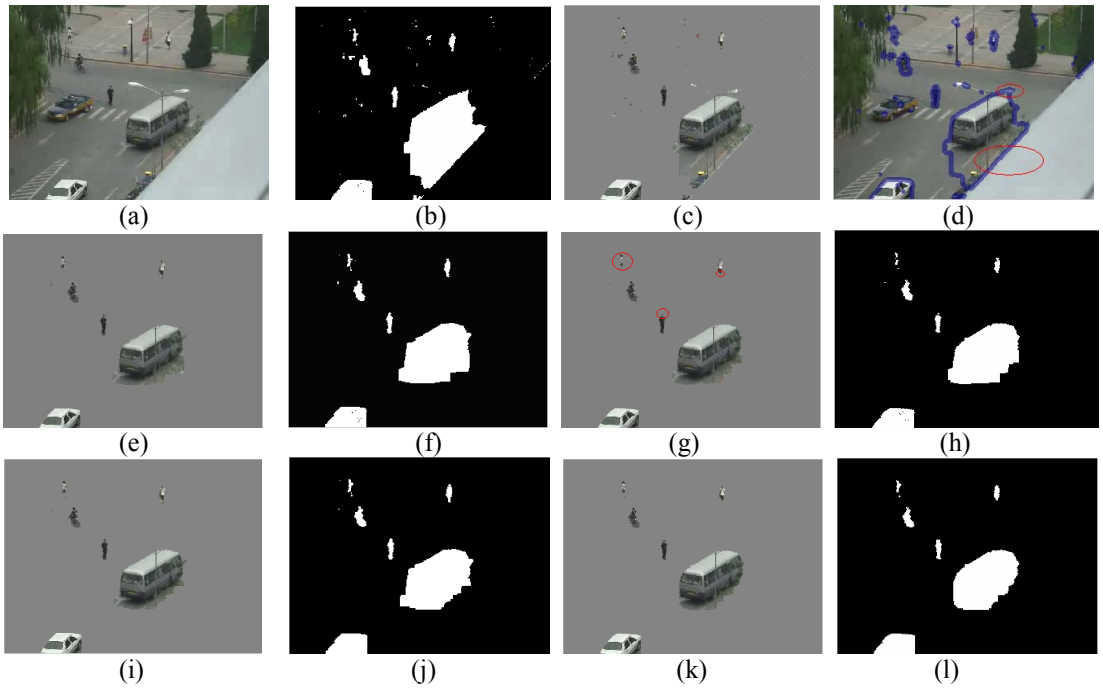


Figure 2. Objects segmentation by different approaches

Figure 2. (a) is the current image to be processed. Since the bus is near the trees, there is much noise. (b),(c),(d) are the results of the basic model, which are shown in different forms to give a clear representation. The noticeable regions in red circle demonstrate that the segmentation is prone to being attracted by strong contrasts. (e) and (f) are the results of background cut model. We can see that the segmentation avoids the strong background contrasts and makes a more accurate result. (g) and (h) are the results of the background cut model, which uses a stronger weight of contrast term to make the segmentation of bus more accurate. Although it achieves the goal, it makes a worse segmentation for the small size objects, whose boundaries are not as strong as the bus. The left person's body, the middle one's neck and the right one's legs disappear in the result. (i) and (j) are the results of our approach in the same contrast weight as (e) and (f), in which the bus and the persons are all more accurate. If we increase the strength of the contrast in the model, we can get the result as shown in (k), (l), and the bus has a more accurate contour.

3.1 Feature points matching

As mentioned in section 1.2, the feature point represents the local color information, and it can be used to suppress the noise in the image. In Equation (7), there are two terms in the basic color model. One encodes global color information, and the other one represents the pixel's color information. But just as what we found in the segmentation results shown in Figure 2, the two terms are not enough to keep a good result at every frame. To maximize the robustness, an ideal system should adaptively adjust the color term: it should decrease the color energy of the noises and increase the energy of the foreground objects. To achieve this goal, we integrate the local information of the feature points into the color model.

Since the background is known, we could obtain the feature points in the background. Here, Harris feature points [16] are used. And then the descriptor D(p) is computed for each feature point p. The descriptor is computed based on a histogram representation of image gradient orientations in its local neighborhood. More specifically, a 5×5 grid of histograms, each with eight orientation bins encodes the image patch around the feature point.

The descriptor is used to match the feature points between background image and current image. And the correlation between two descriptors is defined as:

$$correlation=\frac{\sum_{i}^{N}\min(U_i,V_i)}{\sum_{i}^{N}\max(U_i,V_i)} \tag{13}$$

where U and V are the descriptors, and N is the dimension of U, V. At each feature point P_{back} of the background image, a window of size s × s is used to find the candidates of the current image. Then the correlation is computed between the descriptor of P_{back} and each candidate feature point. The candidate P_{max}

who has the max correlation $Corre_{max}$ is selected and a threshold T is used to measure the correlation. If $Corre_{max}$ is larger than T, then C_{max} is the matched feature point of P. Otherwise, there is no matched feature point for P in the current image. Two feature points P_1 and P_2 are called matched pair if both of them are matched to each other. It means P_1 is matched to P_2 and at the same time P_2 is matched to P_1.

After the above procedure, there are three classes feature points: 1) the matched ones, denoted as P_b. 2) the ones in the current image without matched feature point in the background image, denoted as P_c. 3) the ones in background image without matched feature point in the current image, denote as P_d. The samples are shown in Figure 3. (a) is the background image, and (b) is the current image. (c) is the detected three kinds of feature points. The green ones are P_b, the red ones are P_c, and the blue ones are P_d. We can see that there are lots of green points P_b indicating that the regions around them are background. Meanwhile, the red ones P_c and blue ones P_d always mean they are foreground objects, although some of them are made by the noise. Most of P_c are on the boundaries of the background/foreground objects or the boundaries across the surfaces of the foreground objects, and most of P_d are on the surfaces of the foreground objects, indicating that the regions are occluded by the foreground objects.

(a) (b) (c)

Figure 3. Feature points matching.

(a) Background image. (b) Current image. (c) Feature points matching result of (a) and (b).

3.2 Color model

Once we have classified the feature points in the current image to P_b, P_c, and P_d, we can use them to adjust the color term of the basic model to make the color probability more reliable. The idea in our approach is that P_b can give a high confidence of the background, while P_c and P_d give a high confidence of the foreground objects. Although it is not true for every feature point, it is always true for most of them. We can use P_b to increase the regions' probability as background, and then some noise caused by the pixel-based color model will be suppressed. Meanwhile, P_c and P_d could be used to increase the regions' probability as foreground. It is useful for the small size foreground objects to keep their boundaries, while a strong weight of contrast is used.

We propose the following new color term of $p'(I_n|x=0)$ and $p'(I_n|x=1)$ to represent the probability of background and foreground respectively.

$$p'(I_n \mid x=1) = p(I_n \mid x=1)\cdot([I_n \in N(P_b)]\cdot B_b(I_n,P) + [I_n \in N(P_c)]\cdot C_b(I_n,P) + [I_n \in N(P_d)]\cdot D_b(I_n,P)) \quad (14)$$

$$p'(I_n \mid x=0) = p(I_n \mid x=0)\cdot([I_n \in N(P_b)]\cdot B_f(I_n,P) + [I_n \in N(P_c)]\cdot C_f(I_n,P) + [I_n \in N(P_d)]\cdot D_f(I_n,P)) \quad (15)$$

where $p(I_n|x=1)$ and $p(I_n|x=0)$ is computed as Equation (5) and (7). N(P) means the neighbor of the feature point P. [x] is the indicator function, which is 1 when x is true, otherwise, it is 0. B_b is the function to strengthen the probability as background, and C_b and D_b are used to weaken the probability. B_f could weaken the probability as foreground; meanwhile C_f and D_f take the reverse effect.

In our implementation, the above functions are defined as following:

$$N(P) = \{I_n \mid dist(I_n,P) < d_t\} \quad (16)$$

where $dist(I_n, P)$ is the L2 norm distance of the locations of I_n and P.

$$B_b(I_n,P) = \exp(w_B^b \cdot \frac{1}{dist(I_n,P)+S}) \quad (17)$$

where w_B^b is positive, which controls the strength of the B_b. And S is the constant to adjust the influence of the distance.

$$C_b(I_n,P) = \exp(-w_C^b \cdot \frac{1}{dist(I_n,P)+S}) \quad (18)$$

$$D_b(I_n,P) = \exp(-w_D^b \cdot \frac{1}{dist(I_n,P)+S}) \quad (19)$$

where both w_C^b and w_D^b are positive. C_b and D_b are used to control the strength of attenuation. In Equation (15), B_f has the same form as that of C_b, because it is used to attenuate the foreground probability. C_f and D_f

are similar with B_b, which could strengthen the foreground probability.

4. Experimental results

The videos in our experiments are captured in outdoor environment, and there are some trees which may make the noise. The sizes of the frames are 320×240 and 360×288. Figure 4 shows the results by background cut and our approach. The top rows are the frames in the video. The second and third rows are the results by background cut. And the last two rows are the results by our approach, which used the same contrast weight as that in background cut. In the third row, the results are represented by the binary mask. And the red circles indicate the errors. There are some holes in the objects, some noise which are segmented as foreground objects, and some errors on the background/foreground objects boundaries. And the results by our approach are more accurate than that by background cut, though they use the same contrast weight to control the segmentation.

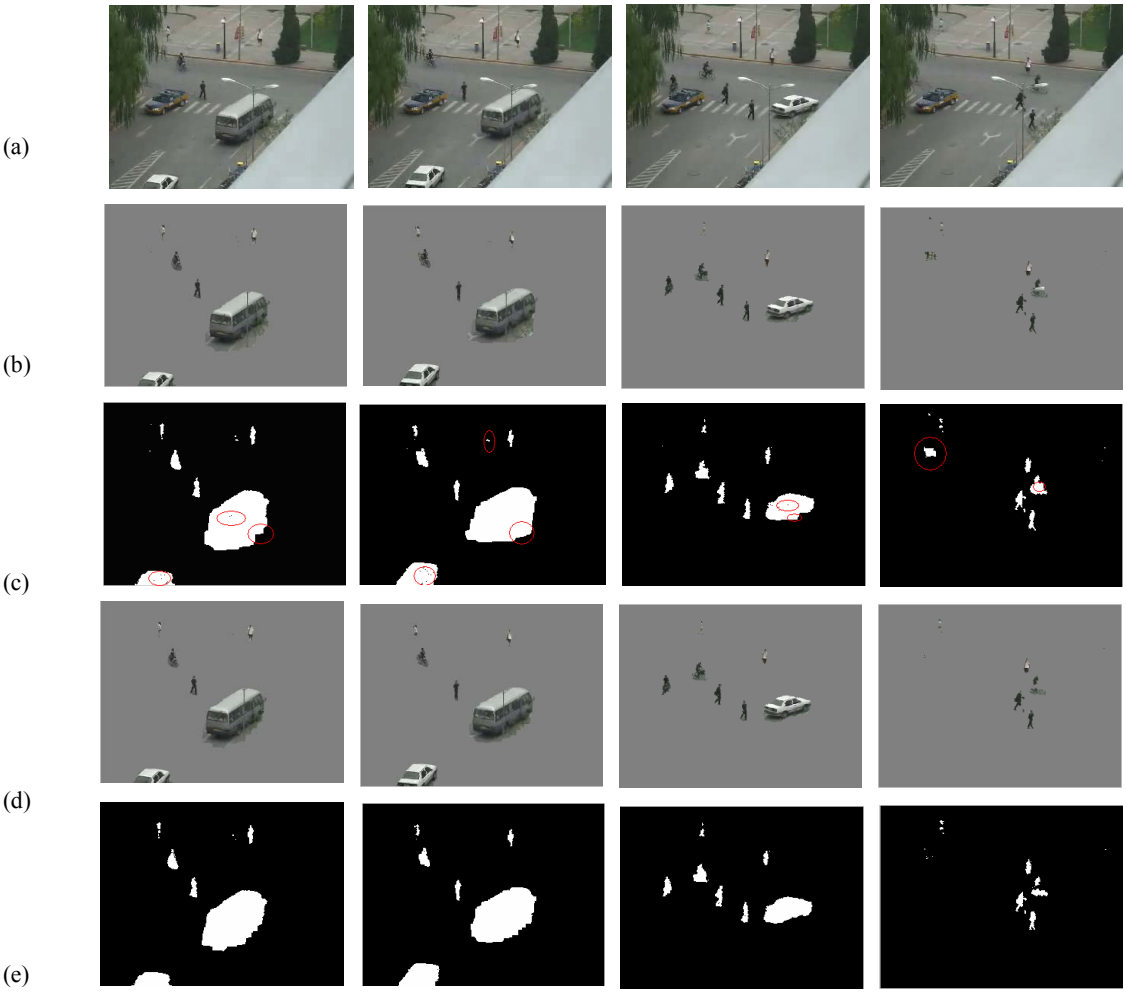


Figure 4. Comparison with background cut.

(a): Several frame in a video sequence. (b) and (c): Results by the background cut. (d) and (e): Results by our approach, using the same contrast weight as background cut.

The results in Figure 5 show the feature points matching results and their effect on the last results by our approach. The first columns are the frames in a video sequence. The second columns are the results by background cut. The third columns are the feature points matching results. The green points are P_b, the red ones are P_c, and the blue ones are P_d. The right columns are the results by our approach, with the same contrast weight as that in background cut. We can see, since the feature points indicating the right states of the regions around them as background or foreground objects, the results by our approach can get more accurate and

compact segmentation.

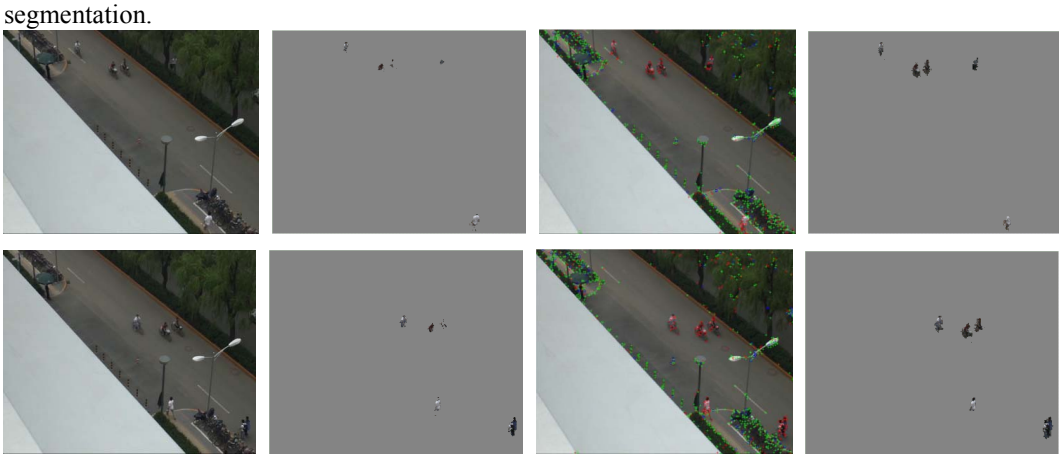


Figure 5. Comparison of results by background cut and our approach.
Left column: Two frames in a video sequence. Second column: Result by background cut. Third column: Feature points matching result. Right column: Result by our approach.

5. Discussion and conclusion

In this paper, we propose a feature-point matching based method to segment the multiple objects in the video sequences, which utilizes the local information of the pixels around the feature points to adjust the color probability. This method combines feature points, background subtraction, color and contrast cues. The background subtraction procedure does not only extract the information of the color, contrast of the background, but also the feature points. The feature points matching is used to suppress the noise and increase the probability of the pixels as what they should be. Our experimental results have shown the improvement of the segmentation.

There are also some limitations in our system. First, if two of the foreground objects are close enough, it is hard to segment them away. Maybe the history of their shapes could help to segment. Second, there is no consideration about the background updating. For a practical system, the background updating problem is the one must to be resolved. These would be studied in our feature work.

6. References

[1] Monnet, A., et al. Background modeling and subtraction of dynamic scenes. in International Conference on Computer Vision, 2003. p. 1305-1312 vol. 2.

[2] Jing, Z. and S. Sclaroff. Segmenting foreground objects from a dynamic textured background via a robust Kalman filter. in International Conference on Computer Vision, 2003. p. 44-50.

[3] Wren, C.R., et al., Pfinder: real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997. 19(7): p. 780-785.

[4] Stauffer, C. and W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in Computer Vision and Pattern Recognition, 1999. p. 246-252 vol. 2.

[5] Rittscher , J., et al. A Probabilistic Background Model for Tracking. in European Conference on Computer Vision, 2000. p. 336-250 vol. 2.

[6] Stenger, B., et al. Topology Free Hidden Markov Models: Application to Background Modeling. in International Conference on Computer Vision, 2001. p. 294-301.

[7] Elgammal, A., D. Harwood, and L. Davis. Non-parametric model for background subtraction. in European Conference on Computer Vision, 2000. p. 751-767.

[8] Mittal, A. and N. Paragios. Motion-based background subtraction using adaptive kernel density estimation. in Computer Vision and Pattern Recognition, 2004. p. 302-309 vol. 2.

[9] Qiang, Z., S. Avidan, and C. Kwang-Ting. Learning a sparse, corner-based representation for time-varying background modelling. in International Conference on Computer Vision, 2005. p. 678-685 vol. 1.

[10] Rother, C., A. Blake, and V. Kolmogorov, Grabcut - interactive foreground extraction using iterated graph cuts. Proceedings of ACM SIGGRAPH, 2004: p. 309-314.

[11] Wang, J., et al., Interactive video cutout. Proceedings of ACM SIGGRAPH, 2005: p. 585-594.

[12] Li, Y., J. Sun, and H.Y. Shum, Video object cut and paste. Proceedings of ACM SIGGRAPH, 2005: p. 595-600.

[13] Kolmogorov, V., et al., Bi-layer segmentation of binocular stereo video. in Computer Vision and Pattern Recognition, 2005: p. 1186-1193.

[14] Sun, J., et al., Background Cut. in European Conference on Computer Vision, 2006: p. 628-641.

[15] Boykov, Y. and M.P. Jolly, Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. in International Conference on Compute Vision, 2001: p. 105-112.

[16] Harris, C. and M. Stephens. A combined corner and edge detector. in Fourth Alvey Vision Conference, 1988. p. 147-151.