12—4

Motion Segmentation and Tracking by Embedding Global Model within a Contextual Relaxation Scheme

Yunqiang Chen, Thomas S. Huang
chenyq@ifp.uiuc.edu , huang@ifp.uiuc.edu
Beckman Inititute, ECE , University of Illinois
Urbana, IL, USA 61801

Abstract

In this paper, we integrate the model-based tracking and local contexture (temporal and spatial) relaxation scheme into a MAP framework to track non-rigid objects such as human hands or faces. By combining them together, our algorithm is much more accurate and robust. The global shape model enables us to represent the dynamic and kinematic constraints. It also helps us to get a better initial segmentation and hence greatly reduce the computation of contextual relaxation. The local contextual constraints help us to get a more accurate support map by considering the local information. Hence it will reduce the significance of the error in the global model. Our algorithm can work autonomously and no need to initialize it. It can detect new moving objects and track them with new blobs. It can also keep track of a stalled object because we also utilize the spatial-temporal constraints. Some promising experiment results are shown.

1 Introduction

Human motion analysis has become more and more important. Real-time applications such as video surveillance, human-computer intelligent interface and video conferencing all require the ability to track moving objects. An efficient multiple objects tracking algorithm in complex environments is a challenging task.

There have been mainly two kinds of methods to track multiple objects. One is model-based tracking methods that utilize the prior knowledge about the objects to be tracked. It is widely used in human body tracking. Human bodies are often represented as stick figures, 2D contours or volumetric blobs connected by joints [1][2][3]. Because the articulated structure of the human body is known, it is possible to impose both the dynamic and kinematic constraints into the tracker and hence such algorithms can give more reasonable and accurate configurations of the body parts. But the models are always not so accurate and we cannot model all the things that may happen. We need not only the global model, but also some low level local constraints such as smoothness constraints in both temporal and spatial to make the tracker more robust to the global model error. In [1], they use some ad-hoc methods to increase the accuracy such as smoothing the class likelihood and using morphology to refine the support maps.

Another kind of methods is multi-layer motion analysis. Compared to the model-based approach, the layer representation is data-driven and imposes weaker prior constraints on segmentation and motion of objects. The key idea behind layer estimation is to simultaneously estimate the object motion and segmentation based on motion consistency. Various constraints on layer motion and layer segmentation have been proposed. In [4] [5], The motion of each layer is modeled either as a single 2D affine or projective motion. But such kind of method always focuses on understanding the motion from 2 or 3 successive frames. They have to estimate the number of the layer, the motion parameters of each layer and the support map of each layer. The heavy computation cost prevents this kind of methods to be used in real time.

In this paper, we integrate the model-based method and multi-layer motion estimation into a MAP framework to track non-rigid motion of human hands and faces in the typical office environment. This framework gives us the good properties of both methods. Some very promising results are shown in the experiments.

2 Mathematical Formulation

Our goal is to track the moving hands and face of the user as a method to control and manipulate in an immersive desk environment. Many real time algorithms use background subtraction to detect foreground [1]. But background maintenance is difficult and is easy to fail. We use frame difference and color information to track hands and faces. In this section, we give the priori models for foreground objects and background and explain how to integrate the local temporal-spatial constraints in a MAP framework.

2.1 The Priori Model

First we assume that each object (hands, faces) can be approximated by a coherent color blob [1]. For every pixel in the image, we need to estimate which blob it belongs to. The set of blob hypotheses is represented as a mixture of multivariate Gaussians $\theta(t)$. Each single Gaussian $\theta_k(t)$ encodes the coherent color value, the centroid and second moments of each blob. An additional layer $\theta_0(t)$ which has uniform distribution is defined for background. The observation includes the color $I(t,x,y)$ and the position (x,y) of each pixel. Assuming the color distribution is independent of its position. The a priori probabilities for the objects can be defined as:

$$P(I,x,y|\theta_k(t)) = P(x,y|\theta_k(t))\,P(I|\theta_k(t))$$

Color is expressed in YUV color space in our algorithm. For simplicity, we assume the distribution of color is Gaussian. The spatial proximity prior for blob k, $P(x,y|\theta_k(t))$, is also a Gaussian. Then the likelihood of a pixel belonging to blob k is defined as: (we define $s=(x,y)^T$)

$$\begin{aligned} P(I,s|\theta_k(t)) \quad = \quad & \frac{e^{-\left(\frac{1}{2}(I-\mu_c)'K_c^{-1}(I-\mu_c)\right)}}{\sqrt{(2\pi)^3 K_c}} \\ & \cdot\frac{e^{-\left(\frac{1}{2}(s-\mu_s)'K_s^{-1}(s-\mu_s)\right)}}{\sqrt{(2\pi)^2 K_s}} \end{aligned}$$

To discriminate the moving foreground and static background, we also need to estimate the motion of each pixel. For real time application, we use frame difference to determine if one pixel is moving. But the difference between two frames is not enough because uncovered background will have large difference too. So we use 3 consecutive frames together to discriminate the moving foreground and background. Supposing the background is static (If background is moving, we can do background motion compensation first), we employ 3 consecutive frames, I_{t-1}, I_t, I_{t+1}, to calculate both the forward difference map ($e^f = I_{t+1} - I_t$), and backward difference map ($e^b = I_t - I_{t-1}$). Then, foreground pixels tent to have large e^b and e^f, while uncovered background will have large e^b and small e^f. If we assume the independence among the error vector, color distribution and position, we get the a priori probabilities for the observations $a=(e,I_{xy},x,y)$: (λ_0 is the label for unchanged pixels and λ_1 for changed pixels.)

Background pixels:

$$p(a|\lambda_B) = p(e^f|\lambda_0)p(e^b|\lambda_0)P(I,x,y|\theta_0(t))$$

Uncovered Background:

$$p(a|\lambda_{UB}) = p(e^f|\lambda_0)p(e^b|\lambda_1)P(I,x,y|\theta_0(t))$$

Will-be-covered Background:

$$p(a|\lambda_{CB}) = p(e^f|\lambda_1)p(e^b|\lambda_0)P(I,x,y|\theta_0(t))$$

Foreground kth blob:

$$p(a|\lambda_{Fk}) = p(e^f|\lambda_1)p(e^b|\lambda_1)P(I,x,y|\theta_k(t))$$

In our experiment, we just use a simple binary model for the priori of the frame difference of changed and unchanged pixels:

$$p(e|\lambda_0) = \begin{cases} 1/6 & if\ e > th \\ 5/6 & if\ e < th \end{cases}$$

$$p(e|\lambda_1) = \begin{cases} 5/6 & if\ e > th \\ 1/6 & if\ e < th \end{cases}$$

We set the parameters by hand and it gives us very good results. Further detailed model or some adaptive methods can be used to make the frame difference models more accurate.

Given all the priori probabilities, the best class for each pixel may be computed using the standard MAP probability decision rule:

$$\widehat{\lambda} = \arg\max_{\lambda\in\Lambda}\ p(a|\lambda)p(\lambda)$$

Most global model based tracking algorithms will assume $p(\lambda)$ to be a constant and the label of each pixel is independent from its neighbors. They try to classify each pixel to different blobs considering only the $p(a|\lambda)$. Unfortunately, this assumption is not right. The label of each pixel correlates with the labels of its spatial- temporal neighbors, i.e. that the support map of each object tends to be smooth and the label of one pixel tends to remain the same if there is no difference from previous frame. Such spatial-temporal constraints can be naturally incorporated in a MAP framework.

2.2 MAP Framework

To incorporate the spatial temporal constraints, $p(\lambda)$ should be estimated according to its spatial-temporal neighbors. This has been extensively studied in the literature of motion segmentation. The local constrains can be modeled by a Gibbs/Markov random field [6]:

$$p(\lambda) = \frac{1}{Z}\exp(-E(\lambda))$$

The constant Z is for normalization, while E denotes an energy term for smoothness. C is the set of all the cliques in the spatial temporal neighborhood. s_1 and s_2 are two different pixels in images. We adopt the commonly used energy function for motion segmentation:

$$\begin{aligned} E(\lambda) &= \sum_{s\in C} \alpha_d \left\| I(s_1) - I(s_2) \right\|^2 \delta(\lambda(s_1) - \lambda(s_2) \\ &\quad + \alpha_s(1 - \delta(\lambda(s_1) - \lambda(s_2)) \end{aligned}$$

The delta function in the first term suspends the smoothness constraint across region boundaries. The second term favors compact regions with short boundaries.

Combine the $p(\lambda)$ and the $p(a|\lambda)$ together, we get the energy function as following:

$$\begin{aligned} \widehat{\lambda} &= \arg\min_{\lambda\in\Lambda} \left(-\log\left(p(a|\lambda)p(\lambda)\right)\right) \\ &= \arg\min_{\lambda\in\Lambda} \left(-\log p(a|\lambda) + E(\lambda)\right) \end{aligned}$$

So the classification problem turn into an energy minimization problem and a lot of methods have been proposed to find the sub-optimal solution. In our algorithm, we adopt the iterated conditional modes (ICM) algorithm [7] to find the sub-optimal classification iteratively.

We first give an initial classification based on ML estimation only. Then refine the classification by minimizing the previous energy function recursively. ICM is easy to be trapped in local maxima. But because we use both the motion information (frame difference), color distribution, shape prior and dynamic constraints to predict the new position of the objects when makeing ML estimation, the initial classification is good enough for ICM to converge and give us much more smooth and accurate support map of each object.

3 Tracking Algorithm

Besides the local constraints, global temporal coherence (Dynamic constrain) is also used in global blob model update. Because the motion of body parts will not change dramatically, we use Extended Kalman Filter [7] to predict the new position and orientation of each blob:

$$\widehat{X}_{[n|n]} = \widehat{X}_{[n|n-1]} + \widehat{G}_{[n]}\left(\widehat{Y}_{[n]} - \widehat{X}_{[n|n-1]}\right)$$

where the estimated state vector includes the blob's position and velocity, the observations are the centroid of the blobs in current image, and is the Kalman gain matrix assuming simple Newtonian dynamics.

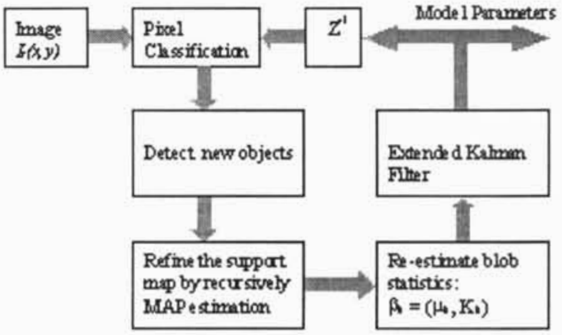

Figure 1: Diagram of tracking algorithm

For every image I_t, we compute the forward and backward error. Then use MAP criteria to classify each pixel according its observations vector and its spatial temporal neighbors recursively, just like the traditional multi-layer motion estimation methods. But because of the introduction of blob model, we can fully utilize the dynamic constrain of the object motion. The global model also gives us a much better initial classification result and hence greatly reduces the number of iterations of relaxation. In our experiments, only one or two iteration will give us fairly good support map of each object.

Initialization for the new intruding objects is also addressed in this paper. After classifying all the pixels in the image with all detected blobs, we analysis all the moving pixels that can not be classified into any existing blobs and try to interpret them with a new blob if their color is close to skin color. Some unsupervised learning algorithms such SOM [10] can be used to decide which color to be tracked. The simple block diagram in Figure 1 summarizes the tracking algorithm described in this section.

4 Experiments

Experiments on real sequences have been shown in Figure 2. It shows very promising results even when multiple persons are in the scene. In the experiment, the different objects are indicated by ellipses in different color.

In the first frame, there are only two objects (one face and one hand) tracked. In the next frame, the intruder's arm is detected and tracked with a new blob. The intruder's face is not tracked because it is smaller than the threshold of the size we set for the object to be tracked. When his arm is occluded in the fourth image, the algorithm stops

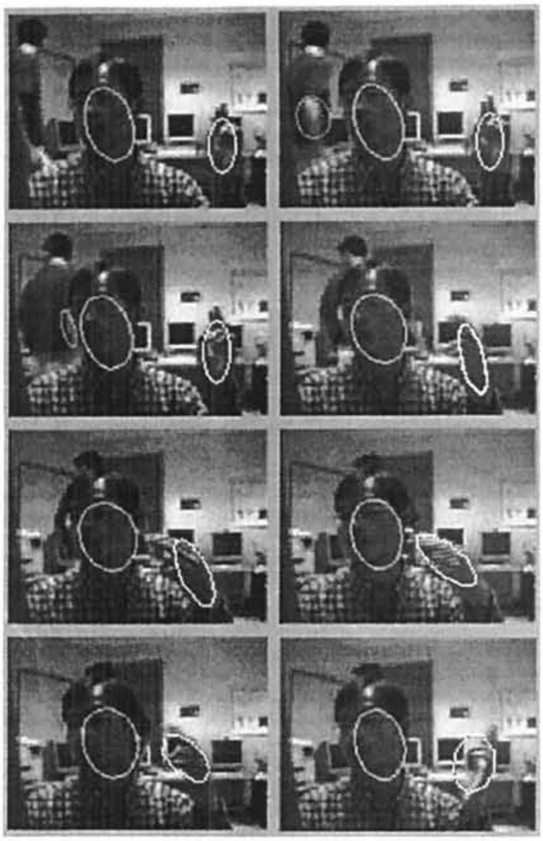

Figure 2: Diagram of tracking algorithm

tracking it. During this whole sequence, the waving hand and the rather static face of the user are well tracked, even when the intruder's arm is very close to the user's face. Without the spatial-temporal constraints, it would be impossible to detect the occlusion of the intruder's arm.

5 Conclusion

In this paper, we integrate both the model-based method and the contextual relaxation method into a MAP framework to track multiple objects. It enables us to utilize both the object-level prior knowledge (such as the shape prior of the objects or dynamic properties of the motion) and pixel-level constraints (such as local spatial-temporal constraints or color distribution and noise models) during the tracking.

Not like the multi-layer motion estimation, we have the shape prior of the objects and know the dynamic properties of the motion. We can predict where the objects will be in the next frame. Then we can get much more accurate initial support map and hence it only takes one or two iteration for contextual relaxation. The relaxation will give us more accurate support map and hence more accurate estimation of the objects' position and properties. The algorithm is quite fast and has great potential for real time tracking.

6 Acknowledgements

This work was supported in part by National Science Foundation Grant CDA 96-24396

References

[1] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland, "Pfinder: realtime tracking of the human body." IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 19, no. 7, pp.780-5, July 1997

[2] C. Bregler, "Learning and recognizing human dynamics in video sequences", Proceedings, 1997 IEEE Computer Society Conference on Computer Vision and pattern Recognition, pp.568-74, 1997

[3] Nebojsa Jojic, Matthew Turk, Thomas S. Huang, "Tracking Articulated Objects in Dense Disparity Maps," International Conf. on Computer Vision (ICCV), Korfu, Greece, September 1999, pp. 123-130.

[4] N. Vasconcelos, "Empirical Bayesian EM-based motion segmentation", in Proc. Of IEEE conference on Computer Vision and Pattern Recognition, pp. 527-532, 1997

[5] P. H. S. Torr, R. Szeliski, and P. Anandan, "An integrated Bayesian approach to layer extraction from image sequences", in Proc. Of IEEE conference on Computer Vision and Pattern Recognition, pp. 983-990, 1999

[6] Stiller, Christoph. Konrad, Janusz. "Estimating motion in image sequences, a tutorial on modeling and computation of 2D motion", IEEE Signal Processing Magazine. v 16 n 4 1999. p 70-91

[7] J. Besag, "On the statistical analysis of dirty pictures," J. Royal Statistical Society, Ser.B, vol. 48, pp. 259-302, Aug. 1986

[8] H. V. Poor, "An Introduction to Signal Detection and Estimation", Springer-Verlag, 1994

[9] Giaccone P, Amanatidis D, Jones GA. Segmenting image sequences by embedding motion and colour cues within a contextual relaxation scheme. [Conference Paper] IEE Colloquium on Motion Analysis and Tracking (Ref. No.1999/103). IEE. 1999, pp.18/1-6. London, UK.

[10] Ying Wu, Thomas S. Huang, "Color Tracking by Transductive Learning", IEEE Int'l Conf. on Computer Vision and Pattern Recognition, Hilton Head Island, SC, June, 2000