A Region-Based Method for Model-Free Object Tracking

Yu Huang[*], Thomas S. Huang[*], Heinrich Niemann[+]
* IFP, Beckman Institute, UIUC, Urbana, IL61801
+ Chair for Pattern Recognition, U. of Erlangen-Nuremberg, Germany, 91058
e-mail: {yuhuang, huang}@ifp.uiuc.edu, niemann@informatik.uni-erlangen.de.

Abstract

We propose a region-based method for model-free object tracking. In our method the object information of temporal motion and spatial luminance are fully utilized. We first compute dominant motion of the tracked object. Using this result we warp the object template to generate a prediction template. Static segmentation is incorporated to modify this prediction, where the warping error of each watershed segment and its rate of overlapping with warped template are utilized to help classification of some possible watershed segments near the object border. Applications of facial expression tracking and two-handed gesture tracking demonstrate its performance.

1. Introduction

Tracking objects in image sequences is an important task for vision-based control, human computer interaction (HCI), content-based video indexing and structure from motion etc. A great variety of visual tracking algorithms have been proposed, they can be classified roughly into two categories [4]. The first is the *feature-based* method. A typical instance in this category estimates the 3D pose of a target object to fit into the image features such as contours given a 3D geometric model of the object. The second is the *region-based* method. Compared to the feature-based methods the region-based methods are more robust, insensitive to small partial occlusions. The region-based methods can be subdivided into two groups: the *view-based* method and the *parametric* method. Our proposed method belongs to the latter group.

1.1 Related Work

Below we discuss some related work in the literature of region-based visual tracking.

Shi and Tomasi [10] put forward the criterion of “good features” by its texturedness and used it in affine feature tracking. Parry et. al [7] introduced a region-based (formed by segmentation) tracking method, mainly updating the template by projecting it around the detected position of the target and considering its overlap with the segmented image. The tracking results showed its good performance when the camera moves towards the object.

Hager and Belhumeur [4] developed a general framework for region tracking which includes models for image changes due to motion, illumination and partial occlusion. They used a cascaded parametric motion model and a small set of basis images to account for shading changes, which will be solved in a robust estimation framework in order to handle small partial occlusion.

Gleicher [3] introduced *difference decomposition* to solve the registration problem in tracking, where the difference would be linear combination of a set of basis vectors. Sclaroff and Isidoro [9] used this idea for template registration in region-based non-rigid tracking, where the non-rigid deformation was represented in terms of eigenvectors of a finite element method. Photometric variation is considered and a modified Delaunay refinement algorithm is used to construct a consistent triangular mesh for the region of the tracked object.

Nguyen and Worring [6] made their contribution by introducing a contour tracking method incorporating static segmentation by the watershed algorithm. Their method utilized kinds of edge maps from motion (optic flow), intensity (watershed) and prediction (contour warping) to update the object contour. It was claimed this method yielded accurate and robust results.

1.2 Review

In this paper we propose a region-based method of motion estimation which undergoes object tracking. In fact, tracking is performed by means of motion segmentation. Our method fully utilizes information of temporal motion and spatial luminance. We compute dominant motion of the tracked object by a robust IWLS method. With that we warp the object template to generate a prediction. Static segmentation is incorporated to modify this prediction, where the warping error of each watershed segment and its rate of overlapping with warped template are utilized to help classification of some possible watershed segments near the object border. Applications of facial expression tracking and two-handed gesture tracking using our method demonstrate its performance.

The trend of our work is comparable to [6]: we predict and update the object template, instead [6] dealed with the object contour. The idea of “active blob“ [9] also discussed the non-rigid deformation: they used Delaunay triangulation of computer graphics to generate some mesh of the object region, instead our method has employed a powerful segmentation tool — watershed, insufficiently we don’t yet consider lighting changes, like [4, 9].

2. General Framework of Object Tracking

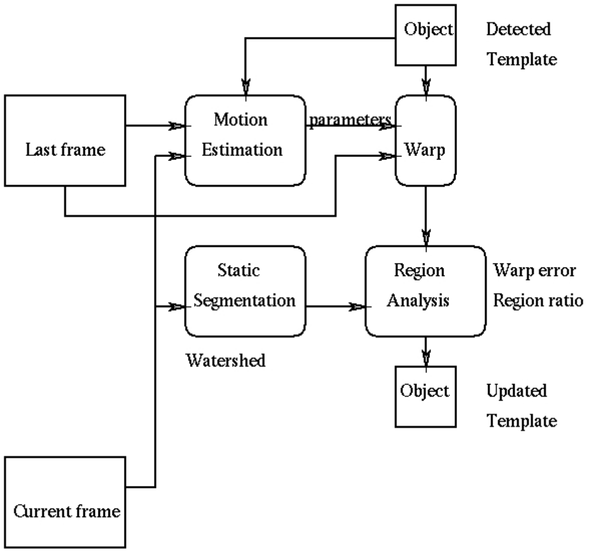


Figure 1 Flow chart of our tracking method

We assume the object region in the image has been detected in the first frame (detection and location of moving objects in an image sequence is another important topic), now the tracking process starts. The flow chart of our method is illustrated in Figure 1. The details of each module are given below.

2.1 Motion Estimation using the M-estimator

Here we describe the problem as follows: the inter-frame motion is defined as

$$f(\mathbf{x}, t+1) = f(\mathbf{x} - \mathbf{u}(\mathbf{x};\mathbf{a}), t), \qquad (1)$$

with *f*(**x**, *t*) as the brightness function in time instant *t*, $\mathbf{x} = (x, y)$ as the coordinate of the image pixel, and $\mathbf{u}(\mathbf{x};\mathbf{a})$ as the motion vector. Without loss of generality, we simply select affine transform as the motion model,

$$\mathbf{u}(\mathbf{x};\mathbf{a}) = \begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} = \begin{bmatrix} a_0 + a_1 x + a_2 y \\ a_3 + a_4 x + a_5 y \end{bmatrix}, \qquad (2)$$

where $\mathbf{a} = (a_0, a_1, a_2, a_3, a_4, a_5)^{\mathrm{T}}$ are the parameters of the affine model. So, the dominant motion estimation of the given region *R* is formulated as the following robust M-estimator,

$$\min_{(u,v)} E_D = \sum_{(x,y)\in R} r(uf_x + vf_y + f_t, s\,), \qquad (3)$$

here f_x, f_y, f_t is partial derivatives of brightness function with respect to *x*, *y* and *t*, the r - function is chosen as the Geman-McClure function [1] and s is the scale parameter. To solve the problem, there are two different ways to find robustly the motion parameters: one is gradient-based, like the SOR method in [1], another is least squares-based, such as the Iterative Weighted Least Squares (IWLS) method. We test both iteration methods and find the latter one is more stable.

The algorithm begins by constructing the Gaussian pyramid (we set up three levels). When the estimated parameters are interpolated into the next level, they are used to warp (realized by bilinear interpolation) the last frame to the current frame. In the current level only the change are estimated in the iterative update scheme.

2.2 Static Segmentation by Watershed

In static segmentation, the watershed algorithm of mathematical morphology is a powerful method[11]. Early watershed algorithms are developed to process digital elevation models and are based on local neighborhood operations on square grids. Some approaches use “immersion simulations“ to identify watershed segments by flooding the image with water starting at intensity minima [11]. Improved gradient following methods are devised to overcome plateaus and square pixel grids [2]. Here we use the former method.

A severe drawback to the computation of watershed algorithm is over-segmentation. Normally watershed merging is performed along with the watershed generation. But here over-segmentation is welcome, so during tracking we omit the merging process, which saves some computation costs.

2.3 Template Warping and Region Anlaysis

Once the motion parameters have been computed, we warp the object template from the last frame to the current frame. Then the warped template is used to determine which watershed segments enter the template according to the following measure: Given that the number of pixels belonging to the warped template in the sub-region (watershed segment) R_i is Cp_i and the number of all pixels in R_i is C_i, a ratio r_i is computed,

$$r_i = Cp_i \,/\, C_i. \qquad (4)$$

Based on this measure, we discuss further the classification problem of each subregion in these following cases:

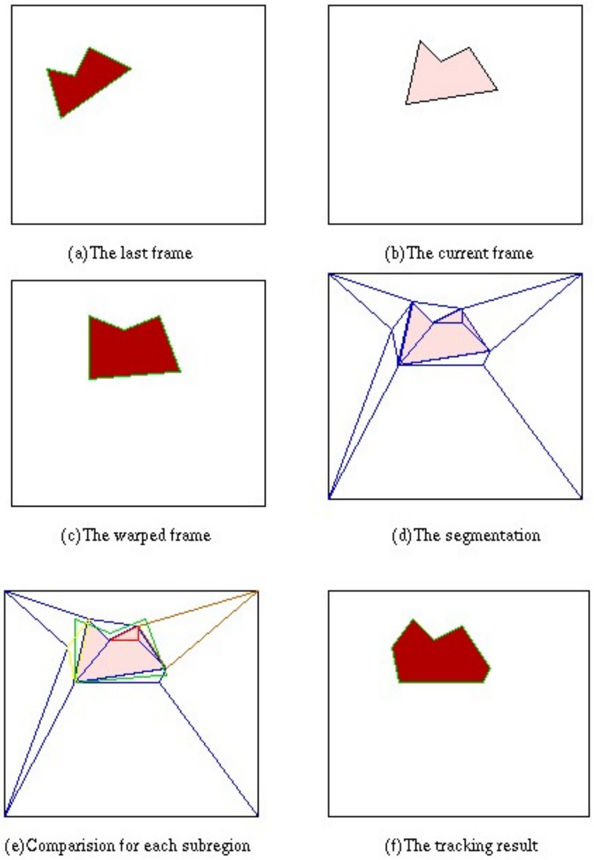
1) When $r_i \geq$ r0 (in this paper r0 = 0.9), classify R_i as part of the final object template;

2) When r0 >$r_i \geq$ r1 (here r1 = 0.4), another measure as MAE (Mean Absolute Error) of difference between the warped frame and the current frame is taken into account,

$$M_i = \sum_{\mathbf{x}\in R_i} \left| f(\mathbf{x}, t+1) - f^w(\mathbf{x}, t) \right| / C_i. \qquad (5)$$

where $f^w(\mathbf{x}, t)$ is the warped image of $f(\mathbf{x}, t)$ using the estimated dominant motion parameters; If the warped error M_i of R_i is smaller enough (less than a given threshold, for instance, 10), R_i is still regarded as part of

the updated template; Otherwise, we exclude R_i out of the object region.
3) When r_i < r1, R_i will NOT be included in the updated template.



(a)The last frame (b)The current frame

(c)The warped frame (d)The segmentation

(e)Comparision for each subregion (f)The tracking result

Figure 2 Illustration of template update in tracking

Figure 2 give an illustration to this process: 2(a) and 2(b) are a pair of consecutive frames. For sake of simplicity, we assume the detected object is equivalent to the real object. 2(c) is the warped object, and 2(d) is static segmentation (in blue). In 2(e) the warped template(enclosed in green), watershed segments and the real object are superimposed to clearly illustrate the comparison. The sub-region enclosed in red agrees with the first case, the sub-region enclosed in yellow agrees with the second case, and the subregion enclosed in brown agrees with the third case. The final object template is shown in 2(f).

In our experiments, it is found the warping error analysis is efficient to avoid some misclassification of small regions near the tracked object in the cluttered background. Indeed, a Kalman filter could be considered to smooth the estimation of motion parameters based on an object kinematic model [5].

2.4 Multiple Objects Tracking

This flow chart is easily extended to multiple-object-tracking, but some specific problems need to be handled accordingly.

In this paper, we focus on the applications of facial expression and two-handed gesture tracking. When people make facial expression movements, especially behaving emotionally (we mainly discuss six universal facial expressions, i.e. Disgust, Sadness, Happiness, Fear, Anger and Surprise), in most of cases head motion is accompanied. We divide our procedure into two steps: 1) Head tracking is realized first, then the estimated motion is used to stabilize the face region; 2) The local motion of each facial features is estimated relative to the stabilized face. Human face motion is complex with rigid and non-rigid movements, so we adopt the idea in [1] using a modified affine model to describe the local motion of facial features (mouth, eyes and eyebrows) and a planar projective transform to model the head motion. We expect also the IWLS method to estimate these motion parameters.

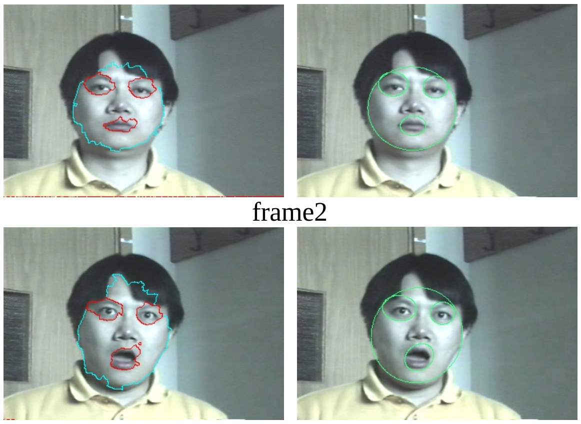
We define a small set of two-handed command gestures, including gestures like “forward“, “backward“, “left“, “right“, “begin” and “stop“ etc. Basically the region-based method has the capability to handle *slight* partial hand-hand occlusions and self-occlusions, which usually appear in two-handed gestures. Here we assume this slight occlusion only happens during a few frames. Besides, heavy deformation of the hands, like the opening and closing actions of the palm, will make our method invalid; So we ask the state of the palm (either opening or closing) unchanged during tracking. Actually the head and both hands are tracked independently.

3. Experiment Results

We realize this approach in Visual C++ in Pentium II 400M. Now the processing speed is about 3-4 seconds per frame. In order to show the tracking results clearly, we use an ellipse to approximate the detected object region. At the initialization, we manually put an ellipse on each tracked object.

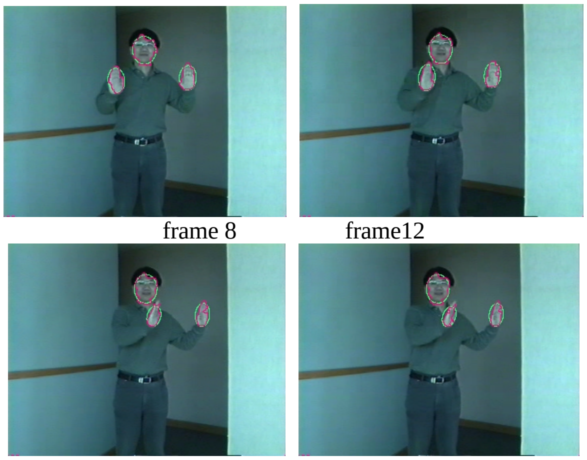
Figure 3 shows the tracking result from the facial expression “Fear” sequence (25 frames). Because the eyebrows of normal people are sparse (thin) in appearance and difficult to be segmented into meaningful regions, instead we use one ellipse to approximate the region of one eye and its eyebrow. These figures on the left column give the region contours of tracked features, and those on the right column correspondingly show the fitted ellipses of tracked features.

Figure 4 displays the results from the “Left” gesture sequence (25 frames). We depict simultaneously the tracked region contours (in red) and corresponding ellipses (in green) on each frame. There are some time instants while the face is closer to some hand. If only using the skin color information, it is very hard to distinguish the skin regions of the head and those of the hand, and the region correspondence between consecutive frames like [12] will fail in this case.


frame2

frame27

Figure 3 Facial Expression Tracking Results


frame 8 frame12

frame15 frame18

Figure 4 Two-Handed Gesture Tracking Results

4. Conclusion

In this paper, we have proposed a region-based approach of motion estimation which undergoes object tracking, its character is full utilization of the object spatio-temporal features in tracking. The template warping only gives a prediction to the tracked feature position, and the comparison of each sub-region with the warped template are able to modify the prediction result. Applications of our method in facial expression tracking and two-handed gesture tracking are encouraging.

The disadvantages of our method are also clear in the experiments. First, we rely on the motion estimation of the tracked object; Even though the IWLS method is more stable, we still confronted the divergence in iterations. Second, while we introduce the static segmentation result our method has strong dependence on the performance of the employed watershed algorithm; We expect the images in the sequence are with enough resolution, and the motion blur are also not welcome; Indeed the segmentation on the face region is more difficult compared with other objects like hands.

In future, we will consider the variations of illumination during tracking [4, 9], which is also an important factor in tracking. Meanwhile, the object region information such as histogram or probability distribution would be useful to detect whether some doubtful sub-regions are classified as part of the tracked objects or not.

Acknowledgement

This research is supported partially by the NSF Grants CDA96-24396, EIA-99-75019 and IIS-00-85980. The authors would really appreciate the anonymous reviewers for their valuable comments.

References

[1] Black M, Yacoob Y, “Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion”, ICCV’95, 1995.

[2] Gauch J, „Image segmentation and analysis via multiscale gradient watershed hierarchies“, *IEEE T-IP*, 8(1): 69-79, 1999.

[3] Gleicher M, “Projective registration with difference decomposition”, IEEE CVPR’97, pp331-337, 1997.

[4] Hager G. and Belhumeur P., “Efficient region tracking with parametric models of geometry and illumination”. *IEEE T- PAMI*, 20(10):1025-1039, 1998.

[5] Jebara T, Pentland A, “Parametrized structure from motion for 3D adaptive feedback tracking of faces”, CVPR’97, 1997.

[6] Nguyen H., Worring M., “Multifeature object tracking using a model-free approach”, IEEE CVPR, pp 145 –150, 2000.

[7] Parry et. al, “Region Template Correlation for FLIR Target Tracking”, British Machine Vision Conference’96.

[8] Saber E, Tekalp A, “Face detection and facial feature extraction using color, shape and symmetry-based cost functions”, ICPR’96, pp654-658, 1996.

[9] Sclaroff S. and Isidoro J., “Active blobs”, ICCV’98.

[10] Shi J. and Tomasi C, “Good features to track”. In *Proc. Computer Vision and Pattern Recognition*, 1994.

[11] Vincent L, Soille, “Watersheds in digital spaces: an efficient algorithm based on immersion simulations”, *IEEE T-PAMI*, 13(6): 583-589, 1991.

[12] Yang M, Ahuja N, “Recognizing hand gesture using motion trajectories“, CVPR’99, pp892-897, 1999.