A Hybrid Object Tracking Method for Scaled and Oriented Objects

Abstract

In this article, we demonstrate a hybrid tracking method for scaled and orientated objects in video. Histogram-based mean shift can track objects quickly. However, it does not utilize shape or dynamics information and typically employs a convex bounding box, such as a rectangle or an ellipse that is decoupled from object scale and orientation. Active shape models and particle filters cope with issues that emerge in the tracking of deformable objects or those whose projected silhouette changes over time. However, they require high computational complexity especially in case of increasing of numbers of landmarks or particles. Combining the strength of mean shift using nonconvex bounding boxes, object dynamics estimation with particle filters, and aligning shape information in consecutive frames using an active shape model, we can track scaled, rotated, and translated objects accurately and efficiently. Experimental results show that the proposed method can realize consistent object tracking in realistic scenarios.

1. Introduction

In computer vision, object tracking, the process of locating and tracking the position orientation and other kinematic state variables of a moving object in consecutive video frames, is a crucial and widely encountered benchmark problem that has utility in applications ranging from smart environments to driver assistance to perceptual user interface and augmented reality. There are numerous proposed approaches to track objects in consecutive frames. Mean shift based object tracking is a recently popularized algorithm that lends itself to relatively simple implementation and its nonparametric nature (using kernel density estimation to model the object's feature distribution), as well as its computational simplicity leading to real time operation and robust tracking performance [1]. Despite its promising performance, two main limitations remain: (i) spatially constant kernel bandwidth, a major source of computational simplicity fails to model object details at different scales and attempts to extend to variable kernel bandwidths [2, 3] diminishes computational efficiency. For non-rigid shapes, the active shape model is trained using the tracked object's shape information a priori and tracking in consecutive frames is attempted by allowing deformations from the mean shape [4]. The ability to encode non-rigid objects using landmarks, this approach has been successfully utilized in a wide variety of applications including medical imaging [5] and body pose estimation [6].

Active shapes introduce shape priors to tracking, however, they still do not incorporate motion dynamics, and kinematic state variables are most suitably estimated (tracked) using recursive Bayesian state estimation; a successful instance of an algorithm that achieves this is the particle filter and has proven very successful especially for tracking non-rigid objects [8]. One of its most important characteristics is to track objects in the presence of occlusions and other similar confusing objects with similar features [7]. However, the original active shape model and particle filter methods are not suitable for real-time tracking in video due to high computational load and large number of iterations required for shape convergence [5].

In this paper, we present an object tracking method for video measurements that combine the strengths of these tools at our disposal. Resulting proposition can exhibit robust and accurate tracking outcomes in real-time with a typical contemporary desktop computer. The proposed method consists of four steps: (i) automatic landmark determination using corner detection [9], which runs only once at initialization, (ii) translating landmarks in consecutive frames using the histogram based mean shift algorithm with non-convex bounding boxes for fast initial convergence in each frame, (iii) searching for the best landmark positions using particle filters, and (iv) aligning the determined position and object to the object's shape observed in the previous frame.

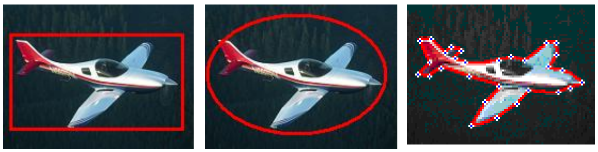
2. Proposed Method

In this section, we describe the proposed method for tracking deformable objects in video. The procedure consists of four steps described above.

2.1. Determining Landmarks at Initialization

Landmark determination has critical importance because it serves as the basis for shape initialization for the object to be tracked in the first frame. The corner detection method of He [9] is used, which works as follows: (i) detect object boundary candidates using the Canny edge detector, (ii) compute curvatures over contours of edges, (iii) remove false corners, and (iv) find the end points of contours. While mean shift would typically utilize a convex bounding box to track the object of interest, the determined landmarks could provide a tightly fit non-convex bounding box that characterizes the object shape and location much more accurately. An example of this comparison is shown in Figure 1. The landmarks are further characterized by the RGB color gradients at these points and in the orthogonal direction to the object boundary, which provides detailed local color/texture baseline information for identifying the landmarks optimally in the following frame.

In this stage, the calculation of the normalized gradient intensity profile of each landmark is needed. Figure 2 illustrates how to get any intensity profile vector of a determined landmark on a video frame and its normalized gradient vector.



(a) (b) (c)

Figure 1. (a, b) Symmetric convex bounding boxes commonly used for object characterization in mean shift tracking **(c)** tight-fit landmark based object shape model at initialization.

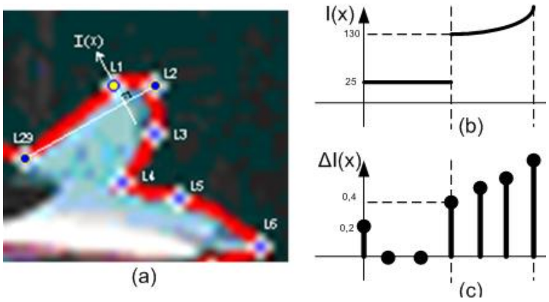


Figure 2. (a) Intensity profile vector of L1 landmark, I(x) **(b)** Graphic draw of I(x) **(c)** Normalized gradient intensity profile of I(x).

2.2. Translating Landmarks with Mean Shift

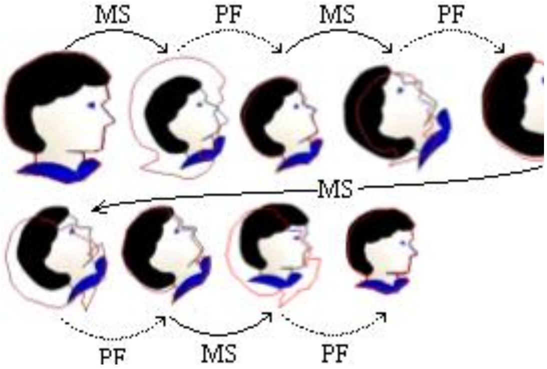


Figure 3. Results obtained from the proposed tracking method for an artificial video sequence (MS: mean shift and PF: particle filter).

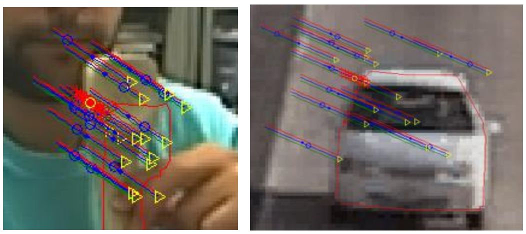


Figure 4. Particle filter landmark search stage illustration for phone and car tracking scenarios. The search lines for each color channel are shown in its color at each particle. The selected particle, which has minimum error, is shown as with a red line and a yellow circle.

Traditionally, mean shift is understood as a kernel density estimation-based fixed-point hill-climbing algorithm, and this nonparametric feature density modeling provides flexibility in tracking objects in video; histogram based simplifications are necessary for real-time operability of this tracking approach [1, 2]. The mean shift tracking method typically employs a convex symmetric bounding box that contains the object of interest and attempts to move this box in consecutive frames using the hill-climbing procedure mentioned. As shown in Figure 1, these bounding boxes typically contain background or other unwanted and irrelevant pixels, thus contaminating the object feature distribution model – this might be problematic in backgrounds with varying texture or color as the video progresses and the object moves. A bounding non-convex box that tightly fits to the object of interest encompassing only pixels that belong to the object region as shown in Figure 1 will circumvent issues that might arise from such problems. Once landmarks are determined at initialization as described earlier, the center of the object (e.g. estimated from the landmarks) is tracked approximately using the histogram based mean shift algorithms assuming a rigid object shape model. This accurately resolves most of the translation component of object motion and might resolve some rotation if not excessive (which is usually the case assuming high frame rates relative to speeds).

We update landmark positions using the following procedure: (1) the center position of each landmark is calculated; (2) 3D color histogram distribution function q belonging to the model object region (surrounded by landmarks) is determined; (3) at each mean shift iteration, we calculate the 3D color histogram distribution function p belonging to the candidate object region and eliminate the differences between q and p (process could be referred to as background elimination); (4) given q and p, the weight at any element $\mathbf{x}$ of 3D histogram is derived from the Bhattacharyya measure and is given by:

$$\mathrm{w}(\mathbf{x}) = \sqrt{\mathrm{q}(\mathrm{I}(\mathbf{x}))/\mathrm{p}(\mathrm{I}(\mathbf{x}))}$$

where $\mathrm{w}(\mathbf{x})$ defines how likely the pixel color $\mathrm{I}(\mathbf{x})$ belong to an object model q. After estimating the center translation amount via mean shift, it is applied to all landmarks.

In Figure 3, an artificial video of a head moving to the right (x_+ direction) is illustrated with the algorithm output at various stages, starting from a good initialization. The first step shows how mean shift accounts for translation with a simple and computationally efficient search for color features. In subsequent steps, mean shift and particle filter landmark tracking achieve precise tracking object location, orientation, and shape (i.e. size).

2.3. Tracking Landmarks with the Particle Filter

Following the mean-shift based translation step, the landmarks are fit to their corresponding optimal locations using a particle filter. To find the best location for each landmark, its baseline color gradient profile from the previous frame (encoded over k pixels extending towards the interior of the object during initialization) is used. The following steps are applied independently to each landmark:

1. Create m particles randomly centered on the current estimate of the landmark's position (random walk model – this could be improved further if needed).
2. Obtain RGB gradient profiles with $2k$ length (extending k pixels to either side) for each particle parallel to the orthogonal to the estimated boundary at the estimated landmark.
3. Find the best position for the landmark on the profile of each particle and calculate minimum squared error with respect to the reference profile from the previous frame.
4. Select the best landmark prototype, which has minimum error, and assign it as the original landmark.
5. Iterate 1-4 until convergence or as needed.

In the Figure 4, particle filter results for phone and car tracking scenarios are illustrated. As is understood from above mentioned steps, we try to find where the best candidate landmark gradient profile is on the search line, also taking into account the initial color gradient profile of each landmark.

2.4. Fine-alignment of Landmarks to New Shape

As the final fine-tuning step of the shape tracking algorithm, the alignment procedure in [4] is applied to the output of the particle filter. According to this procedure, given two similar shapes $\mathbf{S}_1$ and $\mathbf{S}_2$, a rotation angleϕ, a scale factor s, and a translation $\mathbf{t}$ in the x-y image plane, the rigid transformation $\mathbf{M}(\mathbf{S_2}) = \mathbf{Rotate}(\phi) \cdot \mathbf{S_2} + \mathbf{t}$ is optimized to minimize the error function:

$$E = (\mathbf{S}_1 - \mathbf{M}(\mathbf{S}_2))^T \mathbf{W} (\mathbf{S}_1 - \mathbf{M}(\mathbf{S}_2))$$

where $\mathbf{W}$ is a (diagonal) weighting matrix for Mahalanobis distance metric of shape mismatch. The optimization of this error metric for each landmark is carried iteratively by rotating and translating, and convergence is typically achieved quickly since the preceding steps provide good initial estimates.

3. Experimental Results

The proposed method described in the preceding section has been successfully tested on various video sequences. Illustrative results are presented here on two video sequences: car tracking on highway and phone tracking in office. The objects are tracked using features extracted from color in RGB coordinates as described above, the video frames are 320x240 pixel^2 in size, and the algorithm is implemented in Matlab 7.1 (using elementwise matrix operations effectively) and runs on a desktop PC with 1.6 GHz Pentium III CPU and 1GB or RAM with Windows XP operating system. Table 1 shows the elapsed times during the processing of a portion of the video using the proposed hybrid tracking method for Highway and Phone sequences which have 15 and 23 landmarks for the tracked object, respectively. This particular setup achieves about 3-5fps processing speed, which could be further improved.

Table 1. Elapsed times for the proposed method

Frame Index	Elapsed Time in seconds (Highway)	Elapsed Time in seconds (Phone)
70	0.1944	0.2734
71	0.1812	0.2483
72	0.1948	0.2624
73	0.2076	0.2534
74	0.2319	0.2249
75	0.2769	0.2755
76	0.1790	0.2545

In the implementation, the mean shift algorithm employs a 16-bit image histogram quantization and the particle filter search employs 20 particles per landmark. Tracking results obtained using the proposed method are presented in Figure 5 on selected intermediate frames from the two video sequences. As seen from this figure, car and phone objects move smoothly through various scales and orientation angles (in 3D) while translating in the image plane and the proposed algorithm successfully keeps track of the objects. In each frame, red and blue contours represent the mean shift and particle filter step outputs respectively.

4. Conclusion

A shape and feature based robust tracking algorithm that merges the strengths of mean shift, active shape, and particle filter based target tracking techniques is presented for object tracking in color video sequences. The algorithm has relatively low computational requirements and can be implemented in contemporary computing platforms for real-time object tracking in typical webcam size video inputs. With parallelization of certain calculations (such as procedures over particles), even further speed-up is possible. The cascade approach to the algorithm design exploits the strengths of each particular component in reducing optimization time for translation, rotation, scale, and deformation search tasks in a consecutive manner. Background modeling or subtraction is not necessary, but object modeling is required. Results presented indicate robust and accurate tracking performance in real video sequences with noise.

References

[1] D. Comaniciu, P. Meer, “Mean Shift: a Robust Approach Toward Feature Space Analysis,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, pp. 603-619, 2002.

[2] D. Comaniciu, V. Ramesh, P. Meer, “Kernel-based Object Tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, pp. 564–575, 2003.

[3] R.T. Collins, “Mean-shift Blob Tracking Through Scale Space,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1063-6919, 2003.

[4] T.F. Cootes, C.J. Taylor, “Active Shape Models – Smart Snakes,” Proceedings of the British Machine Vision Conference, pp. 266-275, 1992.

[5] D. Seghers, P. Slagmolen, Y. Lambelin, J. Hermans, D. Loeckx, F. Maes, P. Suetens, “Landmark-based Liver Segmentation Using Local Shape and Local Intensity Models”, 3d Segmentation in the Clinic: A Grand Challenge, pp. 135-142, Medical Image Computing, (ESAT/PSI), Belgium, 2007.

[6] J.F. Vasconcelos, R. Cunha, C. Silvestre, P. Oliveira, “Landmark-based Nonlinear Observer for Rigid Body Attitude and Position Estimation,”, Proceedings of the 46th IEEE Conference on Decision and Control, USA, 2007.

[7] E. Cuevas, D. Zaldivar, R. Rojas, “Particle Filter in Vision Tracking,” Technical Report B 05-13, Freie University Berlin, 2005.

[8] M. Isard, A. Blake, “CONDENSATION: Conditional Density Propagation for Visual Tracking,” International Journal on Computer Vision, vol. 1, no. 29, pp. 5-28, 1998.

[9] X.C. He, N.H.C. Yung, “Curvature Scale Space Corner Detector with Adaptive Threshold and Dynamic Region of Support,” Proceedings of International Conference on Pattern Recognition, vol. 2, pp. 791-794, 2004.

[10] Z. Zivkovic, B. Krose, “An EM-like Algorithm for Color Histogram-based Object Tracking,” Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol 1, pp. 798-803, 2004.

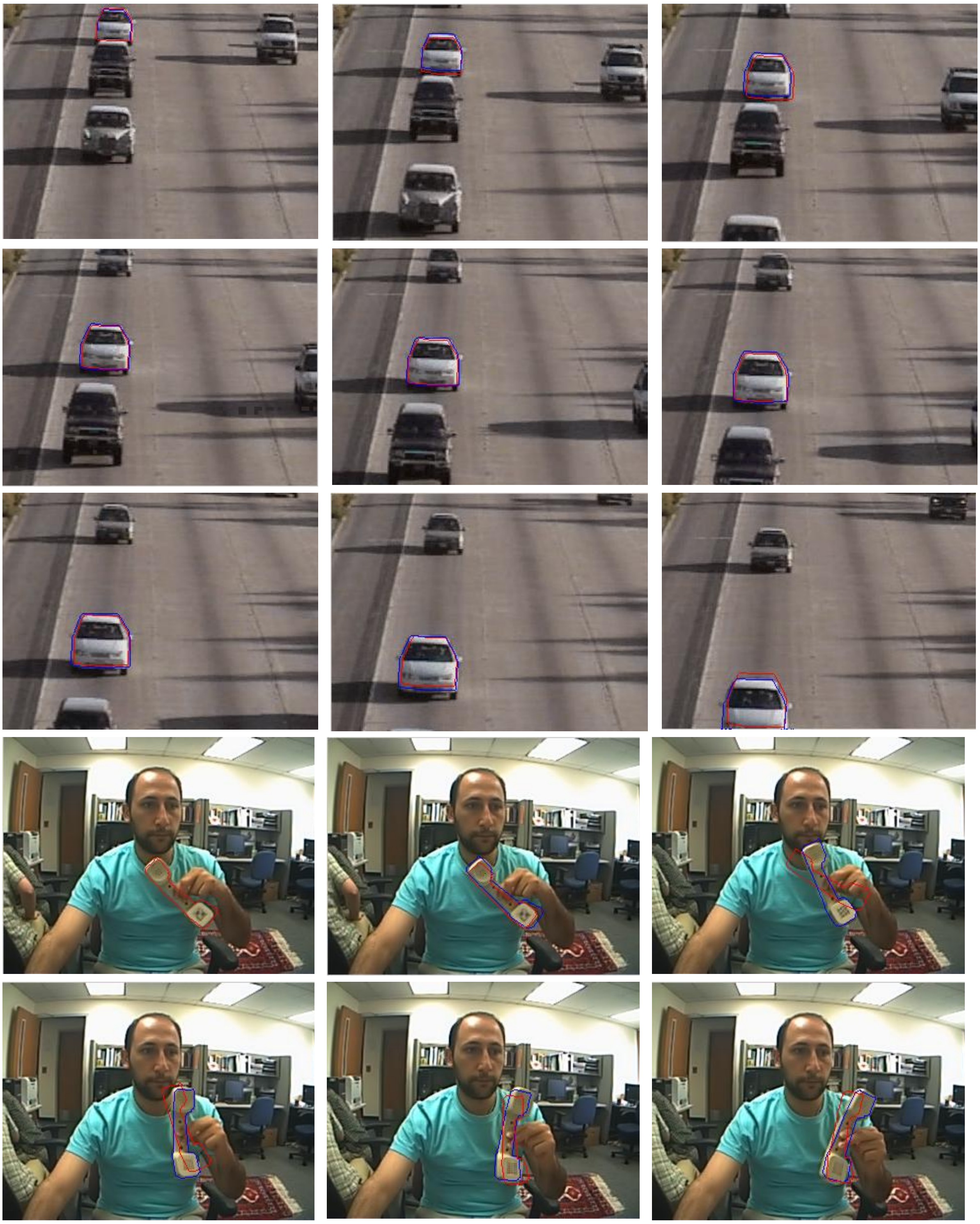


Figure 5. Tracking results obtained using the proposed method for (top) car on highway sequence, (bottom) phone in office sequence. Intermediate mean-shift object translation estimate is shown in red and the final output is shown in blue for each frame.