ON LINE PREDICTIVE APPEARANCE-BASED TRACKING

Namita Gupta[1] *Pooja Mittal*[1] *Kaustubh S. Patwardhan*[2]
Sumantra Dutta Roy[2,*] *Santanu Chaudhury*[3] *Subhashis Banerjee*[4]

[1]Dept of Maths, IIT Delhi, New Delhi 110016
[2]Dept of EE, IIT Bombay, Mumbai 400076
[3]Dept of EE, IIT Delhi, New Delhi 110016
[4]Dept of CSE, IIT Delhi, New Delhi 110016

ABSTRACT

We present a novel predictive statistical framework to improve the performance of an EigenTracker. In addition, we use fast and efficient eigenspace updates to learn new views of the object being tracked on the fly. We also incorporate a new Importance Sampling mechanism which increases the robustness of the EigenTracker, and enables it to track non-convex objects better. Our EigenTracker is flexible – it is possible to use it symbiotically with other trackers. We show its successful application in hand gesture analysis; and face and person tracking.

1. INTRODUCTION

An appearance-based tracker (*EigenTracker* [1]) can track moving objects undergoing appearance changes. Existing extensions of the EigenTracker framework include tracking flexible objects [2], and incorporating the notion of shape in an eigenspace – Active Appearance Models (AAMs) [3]. The Isard and Blake CONDENSATION algorithm [4] can represent simultaneous multiple hypothesis. In [5], they propose the idea of Importance Sampling in a CONDENSATION tracker to improve sample efficacy. We enhance the capabilities of an EigenTracker in three ways. We augment it with a CONDENSATION-based predictive framework to increase its efficiency. We also formulate a novel uniformity predicate as an Importance function to make it more robust. Our predictive EigenTracker learns and tracks unknown views of an object *on the fly* with an on-line eigenspace update mechanism. Our predictive EigenTracker framework is flexible – it can be used to symbiotically augment other trackers with appearance information. The rest of the paper is organized as follows. Section 2 discusses our prediction scheme, eigenspace updates, tracker initialization issues, and the Importance Sampling mechanism. Here, we also describe an interesting extension of our on-line EigenTracker – Symbiotic Tracking. In Section 3, we show the applications of the proposed method, such as in hand gesture analysis; and face and person tracking.

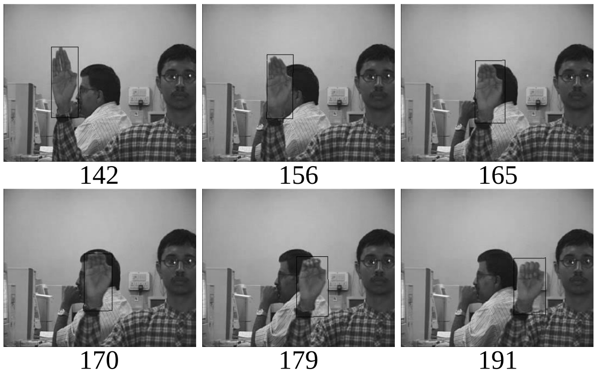


Fig. 1. Our Predictive EigenTracker efficiently tracks a gesticulating hand undergoing appearance changes, in spite of background clutter.

2. ON-LINE PREDICTIVE EIGENTRACKER

2.1. The Prediction Mechanism

One of the main factors for the inefficiency of the EigenTracker is the absence of a predictive framework. The EigenTracker estimates the affine and reconstruction coefficients *after* every frame, requiring a good seed value for the non-linear optimization. The predictive framework helps generating better seed values for diverse object dynamics.

An EigenTracker approximates the object motion by an affine model. We use six affine coefficients as elements of the state vector $\mathbf{X}$. A commonly used model for state dynamics is a second order AR process (t represents time): $\mathbf{X}_t = \mathbf{D}_2\mathbf{X}_{t-2} + \mathbf{D}_1\mathbf{X}_{t-1} + \mathbf{w}_t$, where $\mathbf{w}_t$ is a zero-mean, white, Gaussian random vector. The measurement is the set of six affine parameters obtained from the image, $\mathbf{Z}_t = \mathbf{a}$. Similar to [4], the observation model has Gaussian peaks around each observation, and constant density otherwise. We use a pyramidal approach for the CONDENSATION-based predictive EigenTracker. We start at the coarsest level.

[*] Author for Correspondence, sumantra@ee.iitb.ac.in

ALGORITHM PREDICTIVE_EIGENTRACKER

```
1.  Delineate object of interest
REPEAT FOR ALL frames:
2.  Get image MEASUREMENT optimizing
    affine parameters $\mathbf{a}$ and
    reconstruction coefficients $\mathbf{c}$
3.  IF using Importance Sampling THEN
    optimize $\tilde{\mathbf{a}}$ & $\tilde{\mathbf{c}}$ parameters in the
    'importance' eigenspace to compute
    importance MEASUREMENT
4.  ESTIMATE new affine parameters
    using output of steps 2 and 3
5.  FOR EACH eigenspace:
    IF reconstruction error $\in (T_1, T_2]$
    THEN update eigenspace
6.  IF ANY recons.  error very large
    THEN construct eigenspace afresh
7.  PREDICT $\mathbf{a}$ for next frame
```

Fig. 2. Our On-line Predictive EigenTracker: An Overview

Here, we estimate the values of the affine coefficients based on their predicted values and the measurements done at this level. These estimates serve as seeds to the next level of pyramid. For every frame, we thus get sampled version of conditional state density (S_t), and corresponding weights (Π_t) for CONDENSATION. The state estimate at the finest level is used to generate the predictions for the next frame.

2.2. Initialization and On-line Eigenspace Updates

Accurate tracker initialization is a difficult problem because of multiple moving objects and background clutter. Our system performs *fully automatic initialization* under certain conditions. In general, one may use motion cues (dominant motion detection) but depending on the particular application, other cues can be used to advantage – in our hand gesture tracker for example, we augment motion cues with skin colour cues [6] to segment out the moving hand. In most tracking problems, the object of interest undergoes changes in appearance over time. In a hand gesture-based system for example, it is not feasible to learn all possible hand poses and shapes, off-line. Therefore, one needs to learn and update the relevant eigenspaces on the fly. Since a naive $\mathcal{O}(mN^3)$ algorithm (for N images having m pixels each) is time-consuming, we use an efficient, scale-space variant of the $\mathcal{O}(mNk)$ algorithm (for k most significant singular values) of Chandrasekaran *et al.* [7].

2.3. An Importance Sampling Mechanism

An Importance function augments a tracker operating with one type of measurement, with information from an auxiliary measurement source [5]. Each measurement source has its own characteristics and limitations. When combined in an Importance Sampling framework, the two measurement sources complement each other and together enhance the reliability of the tracker. We propose a new uniformity predicate-based Importance Sampling mechanism. Consider a non-convex shape being tracked (Figure 3). We propose the use of an 'Importance eigenspace' – this represents an object view sans its background. We optimize the $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{c}}$ parameters of the Importance eigenspace to obtain the Importance measurement. An on-line EigenTracker may have problems with changing backgrounds in the bounding parallelogram (and might otherwise end up tracking the background). A combination of the two in an Importance Sampling framework results in more reliable tracking.

2.4. The Overall Tracking Scheme

Figure 2 outlines our overall tracking scheme. In the first frame, we initialize the tracker (Section 2.2). For all subsequent frames, the next step is obtaining the measurements – optimizing the predicted values of affine coefficients $\mathbf{a}$ and reconstruction coefficients $\mathbf{c}$. We then obtain the Importance measurement $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{c}}$, independent of the measurements of step 2. The measurements of step 2 and 3 are combined in the Importance Sampling framework to give the final state estimates. We then calculate the reconstruction error (using the robust error norm [1]), and update the eigenspaces if required (steps 5 and 6). Finally, we predict the affine coefficient values for the next frame.

2.5. Symbiotically Augmenting Other Trackers

We extend our EigenTracking framework for use in conjunction with other trackers, to get its affine parameters. It then optimizes these parameters and returns shape parameters - a tighter fit parallelogram bounding box. We thus take advantage of the other tracker tracking the same object, using a different measurement process, or tracking principle. Such a synergistic combination endows the combined tracker with the benefits of both, the EigenTracker as well as the other one – tracking the view changes of an object in a predictive manner. We have experimented using a CONDENSATION tracker and an EigenTracker for cases of restricted affine motion – rotation, translation and scaling (details in Section 3.1.2).

3. APPLICATIONS

We present two important applications of our approach: gesture analysis; and face and person tracking. Our tracker runs on a 700MHz PIII machine running Linux. In [8], we present some preliminary results of predictive EigenTracking for tracking a moving hand. (Videos: `http://www.ee.iitb.ac.in/~sumantra/icip04a`)

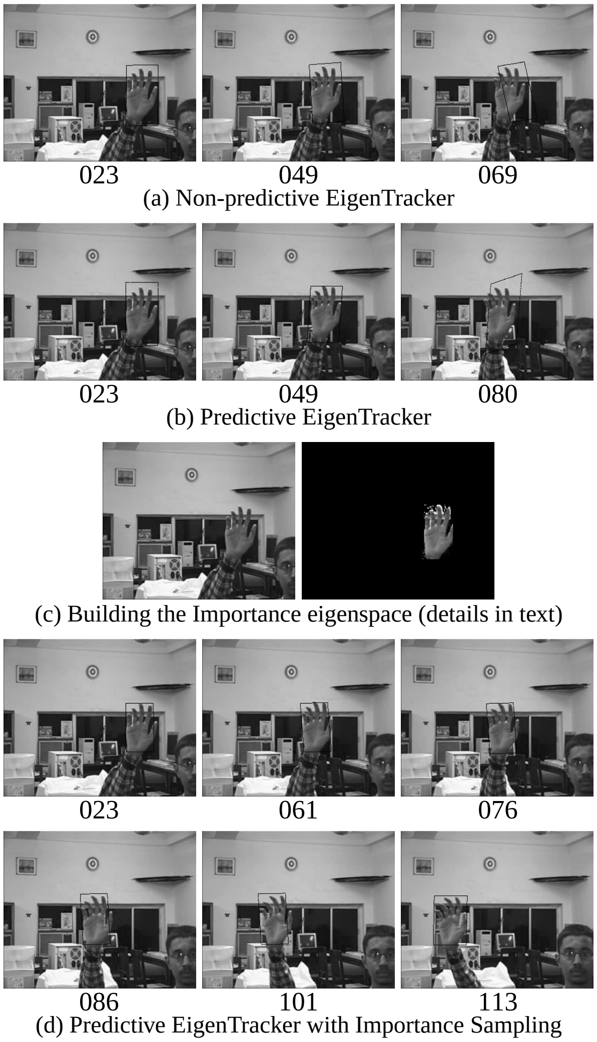
3.1. Hand Gesture Tracking

Figure 1 shows successful application of our tracker to track a hand undergoing extensive shape changes in a typical gesture sequence, filmed against a cluttered background. For the sequence shown in Figure 3(b) and 3(d), average number of iterations decreases from 7.44 to 4.67 due to prediction. For the face tracking example in Figure 5(b), the improvement is from 12.8 to 12.3.

3.1.1. Incorporating our Importance Sampling Mechanism
The authors in [9] show that human skin occupies a small portion in the entire colour space. For colour $C = [C_b C_r]^T$ in the YC_bC_r colour space, we learn two likelihood functions $P(C|skin)$ and $P(C|not\ skin)$. We then calculate n, a number based on the colour C of a pixel as $\frac{P(skin|C)}{P(not\ skin|C)}$, [6]. We consider those pixels corresponding to the top $p\%$ values of n as skin-coloured pixels. These pixels are used in forming the Importance eigenspace (Section 2.3), as shown in Figure 3(c). The effect of Importance Sampling is evident for cases such as in Figure 3 where the background constitutes a large component of the image of the open hand (a non-convex object) being tracked. The entire hand is better tracked in the latter case (Figure 3(d)).

3.1.2. Synergistic Conjunction with Other Trackers: Restricted Affine Motion (Section 2.5)

We now show experimental results of using the on-line, multi-resolution EigenTracker with a modified version of skin colour-based CONDENSATION tracker described in [6]. The latter uses 4-element state vector, consisting of the rectangular bounding window parameters. We first compute the principal axis of the pixel distribution of the best fitting blob. We then align the principal axis with the vertical Y-axis and compute the new width, height and centroid. These parameters give us the restricted affine matrix (scaling, rotation, translation): $\mathbf{A}_{restricted} = Inv(\mathbf{SRT})$. We use these parameters, as an input to our EigenTracker. The EigenTracker then refines these parameters and computes the reconstruction error. In Figure 4 we show results of successful symbiotic tracking. This scheme allows tracking of large rotations as evident in Figure 4. It also yields a better fitting window and less background pixels, leading to lower eigenspace reconstruction error.



(a) Non-predictive EigenTracker

(b) Predictive EigenTracker

(c) Building the Importance eigenspace (details in text)

(d) Predictive EigenTracker with Importance Sampling

Fig. 3. Our Importance sampling mechanism enables the Predictive EigenTracker to track non-convex objects, better

3.2. Face Tracking, Person Tracking

In this section, we show examples of our Importance Sampling method for tracking faces and persons across frames in video sequences. In Figure 5(a) and 5(b), we use a skin colour-based importance function for face tracking. The object to be tracked (a face) undergoes motion as well as considerable change in appearance. The on-line predictive EigenTracker with Importance Sampling correctly tracks for all 44 frames – a great improvement over simple EigenTracker. Additionally, our on-line predictive EigenTracker has *learnt* different eigenspace views of the object being tracked - this information can be used to recognize the person in other film clips as well.

In Figure 5 (person tracking), the uniformity predicate is based on the colour of the person's shirt. Successful tracking results even though the person moves against a back-

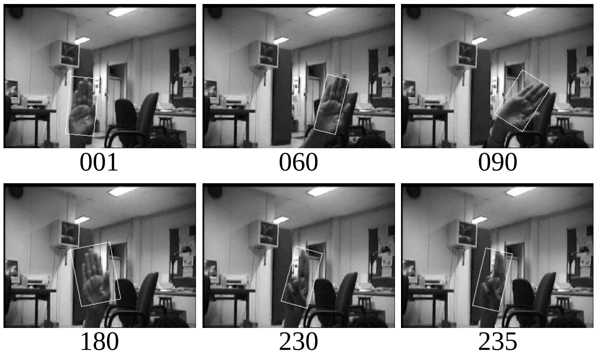


Fig. 4. Synergistic tracking (Section 2.5)

ground of a similar colour. In this case, a simple EigenTracker based on a colour predicate alone, would have failed because the background colour is similar to that of the object being tracked. However, the same in importance framework enables the person to be tracked correctly. While we have used colour, one may use texture or any other uniformity predicate for the region of interest.

4. REFERENCES

[1] M. J. Black and A. D. Jepson, “EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation,” *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63 – 84, 1998.

[2] F. De la Torre, J. Vitria, P. Radeva, and J. Melenchon, “Eigenfiltering for Flexible Eigentracking (EFE),” in *Proc. International Conference on Pattern Recognition (ICPR)*, 2000, pp. III:1118 – 1121.

[3] T. Cootes, G. J. Edwards, and C. Taylor, “Active Appearance Models,” in *Proc. European Conference on Computer Vision (ECCV)*, 1998.

[4] M. Isard and A. Blake, “CONDENSATION - Conditional Density Propagation For Visual Tracking,” *International Journal of Computer Vision*, vol. 28, no. 1, pp. 5 – 28, 1998.

[5] M. Isard and A. Blake, “ICONDENSATION: Unifying Low-level and High-level Tracking in a Stochastic Framework,” in *Proc. European Conference on Computer Vision (ECCV)*, 1998, pp. 893 – 908.

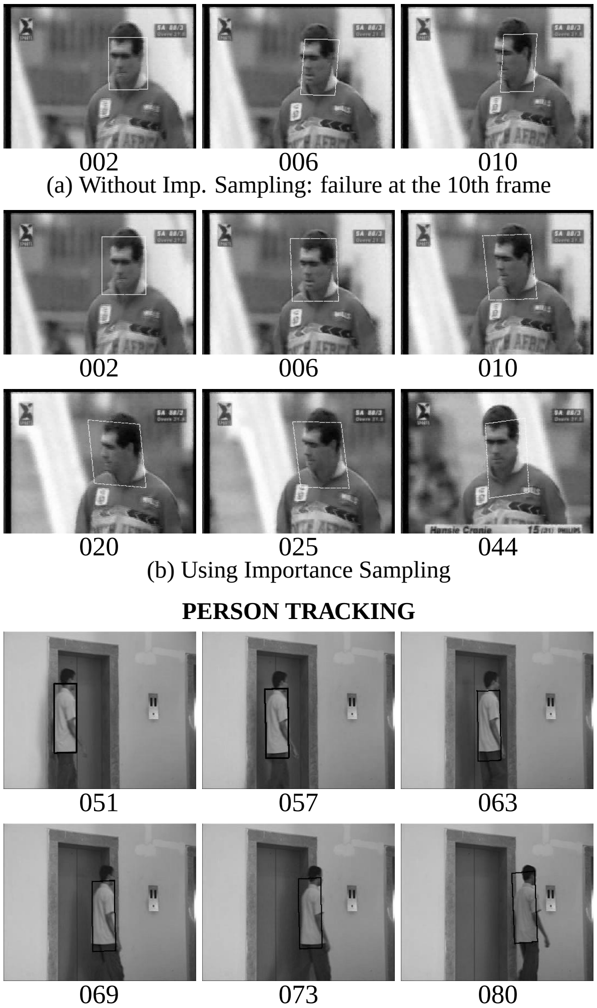
[6] J. Mammen, S. Chaudhuri, and T. Agrawal, “Tracking of both hands by estimation of erroneous observations,” in *Proc. British Machine Vision Conference (BMVC)*, 2001.

[7] S. Chandrasekaran, B. S. Manjunath, Y. F. Wang, J. Winkeler, and H. Zhang, “An Eigenspace Update Algorithm for Image Analysis,” *Graphical Models and Image Processing*, vol. 59, no. 5, pp. 321 – 332, September 1997.

[8] N. Gupta, P. Mittal, S. Dutta Roy, S. Chaudhury, and S. Banerjee, “Developing a gesture-based interface,” *IETE Journal of Research: Special Issue on Visual Media Processing*, 2002.

[9] R. Kjeldsen and J. Kender, “Finding Skin in Color Images,” in *Proc. Intl. Conf. on Automatic Face and Gesture Recognition*, 1996, pp. 312 – 317.

FACE TRACKING IN A SPORTS VIDEO



(a) Without Imp. Sampling: failure at the 10th frame

(b) Using Importance Sampling

PERSON TRACKING

Fig. 5. Using our Importance Sampling mechanism (Section 2.3) for two applications: Face tracking in a sports video, and person tracking in a movie sequence