In Situ Evaluation of Tracking Algorithms Using Time Reversed Chains

Hao Wu, Aswin C Sankaranarayanan and Rama Chellappa
Center for Automation Research and Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
{wh2003, aswch, rama}@umiacs.umd.edu

Abstract

Automatic evaluation of visual tracking algorithms in the absence of ground truth is a very challenging and important problem. In the context of online appearance modeling, there is an additional ambiguity involving the correctness of the appearance model. In this paper, we propose a novel performance evaluation strategy for tracking systems based on particle filter using a time reversed Markov chain. Starting from the latest observation, the time reversed chain is propagated back till the starting time $t = 0$ of the tracking algorithm. The posterior density of the time reversed chain is also computed. The distance between the posterior density of the time reversed chain (at $t = 0$) and the prior density used to initialize the tracking algorithm forms the decision statistic for evaluation. It is postulated that when the data is generated true to the underlying models, the decision statistic takes a low value. We empirically demonstrate the performance of the algorithm against various common failure modes in the generic visual tracking problem. Finally, we derive a small frame approximation that allows for very efficient computation of the decision statistic.

1. Introduction

Tracking an object continuously over a period of time to generate a persistent trajectory is crucial in video surveillance. Most subsequent applications, such as object recognition, activity analysis are dependent on the accuracy and robustness of the tracking algorithms. Although many sophisticated algorithms exist for tracking, each of them will have failure modes or scenarios when the performance will be poor. Typically, this will happen when the data (the frames of the video) does not fit the assumptions made in the modeling. Most algorithms find difficulty in tracking a target (or targets) through crowded environments, clutter or poor/dramatically changing illumination and self-occlusion. This often leads to a loss of track. In this context, performance evaluation plays an important role in practical visual tracking systems. However, existing performance evaluation algorithms concentrate on off-line statistical comparisons with manually created ground truth data. While comparison with ground truth can inform which tracking algorithm has better overall performance on a specific sequence, it does not extend gracefully for testing on new sequences without additional ground truth. In the absence of ground truth, off-line performance evaluation can not help to detect the loss of track and/or improve the robustness of the tracking systems.

In many surveillance scenarios, *in situ* evaluation of performance is desired. Here, *in situ* means that the evaluation is automatic, without use of any ground truth, and that it is also an online and causal evaluation method.

Prior Work: Online evaluation of tracking algorithms has received some attention in existing literature. In [5], Erdem *et al*. address on-line performance evaluation for contour tracking. Metrics based on the color and motion differences along the boundary of the estimated object are used to localize regions where segmentation results are of similar quality, and combined to provide quantitative evaluation of the boundary segmentation and tracking. As an extension, [6] proposes the use a feedback loop to adjust the weights assigned to the features used for tracking and segmentation. This method of evaluation is specific to contour based tracking systems. Wu and Zheng also present a method for the performance self-evaluation in [14]. This empirical method evaluates the trajectory complexity, motion smoothness, scale constancy, shape and appearance similarity, combining each evaluation result to form a total score of the tracking quality. However, this method can only be applied to a static camera system.

For stochastic nonlinear systems, measurements based on the innovation error forms a common choice as an evaluation metric, for example, the tracking error (TE) or observation likelihood (OL) statistics, and their corresponding cumulative summation in time series (CUSUM) [12]. TE and OL detect only sharp changes which results in loss of track, and do not register slow changes. A statistic for detection of slow changes called ELL and its generalization gELL are given in [12]. ELL is defined as a measure of

1-4244-1180-7/07/$25.00 ©2007 IEEE

inaccuracy between the posterior at time t and the t-step ahead prediction of the prior state distribution. Interesting, as we point out later, the proposed evaluation methodology mirrors ELL in spirit.

In [1][7][13], under the hypothesis that the model is correct, the authors identify a random process in the scalar observation space to be a realization of independent identically distributed variables uniformly distributed on interval $[0, 1]$. This result holds for any time series and may be used in statistical tests to determine the adequacy of the model. An extensions to vector valued measurements is presented in [2], where the authors utilize a χ^2-test for multi-dimensional uniform distribution to find if the system behaves consistently. However, when it comes to visual tracking, as the observation could be in a very high dimensional image space, the computation of the test statistics is infeasible.

[11] uses an entropy based criterion to evaluate the statistical characteristics of the tracked density function. The definition of good performance for tracking a single object is that the posterior distribution is unimodal and of low variance. In contrast, a multi-modal and a high variance distribution implies poor or lost tracking. In practice, tracking in the presence of multiple targets and clutter does lead to the presence of multi-modality in the target's posterior density. This, however does not necessarily imply poor tracking.

In this paper, we design an *in situ* evaluation methodology that can be applied to most tracking algorithms to detect most tracking failures and evaluate performance of tracking. We illustrate the efficiency of the proposed method for a general visual tracking system driven by a particle filter. To conduct performance evaluation in real time, a time reversed Markov chain is constructed and the posterior probability distribution of the reversed Markov chain at the initial time is computed and compared to the prior distribution used to initialize the tracker. The posterior of the reversed time system at the initial time (under a certain condition) is the conditional density of the prior state. Estimates from each density such as the mean can be shown to be unbiased and equal. In this sense, for a well behaved system, the two probability distributions should show proximity in some statistical sense, with significant discrepancies between them in the presence of tracking error. While tracking back to the initial time is costly with increasing time, we propose an approximation by tracking back and comparing the performance against a point in time where by prior verification we know for sure that the performance is good. The efficiency of the algorithm against various common failure modes for visual tracking problems is empirically demonstrated and compared to the ground truth and other methods.

The remainder of the paper is organized as follows: in section 2, we summarize the necessary background of the dynamic tracking system including its evaluation methods and failure modes. A detailed description of the proposed method is presented in section 3. Experimental validation and discussions are given in section 4.

2. Evaluation in Dynamical Systems

In this section, we summarize the necessary background of Bayesian filtering methods used in dynamical systems, in particular, the particle filtering method which is used widely in visual tracking systems [9].

2.1. Particle Filtering

In particle filtering [3], we address the problem of Bayesian inference for dynamical systems. Let $x_t \in \mathbb{R}^d$ denote the state at time t, and $y_t \in \mathbb{R}^p$ the noisy observation at time t. We model the state sequence $\{x_t\}$ as a Markovian random process. Further we assume that the observations $\{y_t\}$ to be conditionally independent given the state sequence. Under these assumptions, the models defining the system are given as follows: 1) $p(x_t|x_{t-1})$: The state transition probability density function, describing the evolution of the system from time $t-1$ to t; 2) $p(y_t|x_t)$: Observation likelihood density, describing the conditional likelihood of observation given state; and 3) $p(x_0)$: The prior state probability at $t = 0$.

Given statistical descriptions of the models and noisy observations till time $t, \mathcal{Y}_t = \{y_1, \ldots, y_t\}$, we would like to estimate the posterior density function $\pi_t = p(x_t|\mathcal{Y}_t)$.

The problem is that this computation need not have an analytical representation. However, foregoing the requirement for analytic solution, the particle filter approximates the posterior π_t with a discrete set of particles or samples $\{x_t^{(i)}\}_{i=1}^N$ with associated weights $\{w_t^{(i)}\}_{i=1}^N$ suitably normalized. The set $S_t = \{x_t^{(i)}, w_t^{(i)}\}_{i=1}^N$ is the weighted particle set that represents the posterior density at time t, and is estimated recursively from S_{t-1}. The initial particle set S_0 is obtained from sampling the prior density $\pi_0 = p(x_0)$.

2.2. Common Failure Modes in Visual Tracking

The premise of most work on model validation and change detection assume that without an exact model, the system will behave abnormally. However, in visual tracking systems, due to the unavoidable and unexpected degradations, it is nearly impossible to design models that are robust against all possible failure modes.

Generally, there are four main classes of failure modes in visual systems:

Pose Change: In general, for a moving 3D object, the appearance projected in any particular frame depends on its pose and position with respect to the camera. However, most visual tracking systems lack knowledge of the 3D appearance of different objects. This is especially true when appearance modeling is done online, when all the available information about appearance is the initialization of the system, which contains information only from one pose of the object.

Occlusion: A common failure mode for most tracking algorithms is when the target is occluded. Typically, given the wide nature of possible occluding conditions, occlusion is handled as an outlier to the appearance model.

Imaging System: Image measurements are corrupted by noise and blur. In a static optical visual system, this may not have significant consequences. However, when the imaging modality is infra-red or when we are considering aerial images, this would be a very important problem.

Clutter: In practical systems, as the objects are moving in some background, the background clutter can contribute to loss of tracking. This distractions can not be modeled before hand either.

It is very tricky to incorporate knowledge of all these failure modes into visual tracking systems. For a tracking algorithm, if it uses a fixed appearance template, then it can not handle the changes in the video. On the other hand, if the appearance model changes rapidly (say from the latest estimated appearance), it is susceptible to drift [16]. Therefore, many sophisticated tracking systems [16] use adaptive online appearance model (OAM) or multiple features to improve the performance. For such complicated systems, it is often not immediately obvious what the failure modes will be, or how the algorithm performs on a particular dataset. This makes *in situ* performance evaluation all the more important but challenging.

3. Evaluation Using the Time Reversed Markov Chain

3.1. Intuition

Our goal is to provide a general, online evaluation method for most visual tracking systems. The key idea is to formulate a time reversed Markov chain, compute the posterior distributions along the time reversed Markov chain all the way to the initializing frame of the forward Markov chain, and design a statistic to evaluate the distance between the initialization (the prior at $t = 0$) and the time-reversed posterior distribution. Alternatively, for algorithms employing OAMs, the *identity* of the target is defined in the initializing frame and the prior used to initialize the system. This prior information encodes all the knowledge given to the tracking algorithm, and arguably is most critical in determining the performance of the algorithm. In this sense, the tracking performance can be determined by verifying the output of the tracker at any particular time instant (say $t = t_0$) against the prior with suitable time normalization.

From the point of view of information captured in the tracking algorithms, the underlying intuition is that if, at time t, the tracker contains enough information about the target, then the ability to track well till time t along the forward Markov chain implies that it should be able to track back to the end along time reversed Markov chain equally well with a high probability.

3.2. Proposed Algorithm

The forward Markov chain state and observation models of the tracking system considered in our paper are as follows:

$$\begin{array}{rcl} \text{Prior at time } t=0 & : & p(x_0) \\ \text{State Transition Model} & : & p(x_t|x_{t-1}) \\ \text{Observation Model} & : & p(y_t|x_t) \end{array} \quad (1)$$

At time T, given an observation sequence $\mathcal{Y}_T = \{y_1, \ldots, y_T\}$, the posterior is $\pi_T = p(x_T|\mathcal{Y}_T)$. To evaluate the performance of the system, we propose a reverse time tracker that uses π_T as its prior and the observation sequence $\mathcal{Y}_T$ in the time reversed order. Using the notation $q(\cdot)$ for probability density functions associated with the time reversed system, the reverse tracker is formulated as follows. For evaluation at time T, the system is initialized at time $T+1$ and filtered through the observations $\mathcal{Y}_T$.

- **Prior at time $T+1$:**

$$\begin{array}{rcl} q(x_{T+1}) & = & p(x_{T+1}|\mathcal{Y}_T) \\ & = & \int p(x_{T+1}|x_T)p(x_T|\mathcal{Y}_T)dx_T \end{array} \quad (2)$$

- **State Transition Model:** For $t \in (0, T)$,

$$q(x_t|x_{t+1}) = \frac{p(x_{t+1}|x_t)p(x_t)}{p(x_{t+1})} \quad (3)$$

  This can be directly computed from the models for most systems used to define the tracking problem.

- **Observation Model:** We retain the same observation model used in the forward model.

$$\forall t, q(y_t|x_t) = p(y_t|x_t) \quad (4)$$

With this characterization of the system, we can now filter the observation sequence $\mathcal{Y}_T^b = \{y_T, \ldots, y_1\}$ in reverse time. The posterior density function of this filter is of great interest to us. At time t, the posterior density $\pi_t^b = q(x_t|\mathcal{Y}_t^b) = q(x_t|y_T, y_{T-1}, \ldots, y_t)$.

We can now estimate the posterior density at time $t = 0$, π_0^b by recursion. From intuition, we expect this density to be close in some statistical sense to the prior density $p(x_0)$. To this extent, we postulate the following property.

Proposition: Suppose the reverse tracker is initialized with the prior $q(x_{T+1}) = p(x_{T+1})$, then the posterior density of the time reversed system at time $t = 0$ and the prior density $p(x_0)$ are close to each other on distance metrics comparing the means of the corresponding random variables, provided the underlying model completely fits the data.

Suppose we initialize the reverse time Markov chain using the density $p(x_{T+1})$ as opposed to $p(x_{T+1}|\mathcal{Y}_T)$. It is easy to verify that the final posterior distribution in the time reversed process is equal to the smoothing result [10] at the beginning of the forward process using all the observations till time T, i.e, $\pi_0^b = p(x_0|y_{1,\ldots,T})$.

Now, π_0^b and the $p(x_0)$ are close in the sense that

$$\int x_0 p(x_0) dx_0 = \int_{\mathcal{Y}_t} \int_{x_0} x_0 \pi_0^b d\mathcal{Y}_t dx_0 \tag{5}$$

Suppose we compare $E(x_0)$ and $E_{\mathcal{Y}_t}(x_0)$, then on an average (over the ensemble set of possible observations) the two means will be the same.

It should be noted that the above result is true only when the reverse time system is initialized with the prior $p(x_{T+1})$. In practice, the prior $p(x_{T+1}|\mathcal{Y}_T)$ is expected to have a better localization of the target at time $T+1$ when the data fits the modeling correctly. Hence, the system defined with the prior $p(x_{T+1}|\mathcal{Y}_T)$ is *over-trained* and provides a model that fits the data better.

To evaluate the tracking performance of a system, we verify if the tracker output at current time has sufficient information to allow a reversed time system to track back to the initial frame.

The key point of our algorithm is that for many visual tracking systems using OAMs, the identity of the target that is tracked comes from prior information. *With no additional knowledge of the target, the prior is the equivalent of a ground truth, defining a point of reference against which the performance of the system can be verified against.*

3.3. Evaluation Statistic

There exists distance metrics and measures for comparing density functions such as the Kullback-Leibler (KL) divergence and the Bhattacharya distance [4]. However, in our case, the distributions are represented by particles or samples from the density function. In general, given the differences in the individual proposal densities and the random number generation, the exact locations at which the densities are sampled will be different. Computing the KL divergence or the Bhattacharya distance for such non-overlapping sample sets would require interpolation (such as Parzen windowing [4]) or the use of approximations such as the Unscented Transformation [8]. We circumvent this problem with the use of the Mahalanobis distance that uses only the moments of the distributions.

The distance $d(p,\pi)$ between the distributions p and π computed at time t is

$$\begin{aligned} d(p,\pi) &= (\mu_p - \mu_\pi)^T \Sigma_p^{-1} (\mu_p - \mu_\pi) + \\ &\quad (\mu_p - \mu_\pi)^T \Sigma_\pi^{-1} (\mu_p - \mu_\pi) \end{aligned} \tag{6}$$

where μ_p and Σ_p are the mean and the var-covariance matrix of the distribution p and μ_π and Σ_π are those of the distribution π, all of which can be easily computed or estimated from the particles or in some cases, analytically. *We also note that just using the part of the state space corresponding to location (or translation on the image plane) gave better results, or results more in tune with the perceptual notion of tracking performance.*

An outline of the proposed evaluation framework is in Table 1.

The proposed algorithm also encompasses another interesting idea. Suppose we have a video sequence in which the first frame and the last frame are identical, then we would expect the tracker to localize the target in the last frame at the same location as the prior given in the first frame. Such an idea is exploited for detecting drift in feature point tracking in [15]. The proposed algorithm is an extension of that

To evaluate the performance of the tracking at time T, the density π_T represented by the samples $\{x_t^{(i)}\}_{i=1}^N$,

1. Propagate the particles using $p(x_{T+1}|x_T)$ to get samples from $p(x_{T+1}|\mathcal{Y}_T)$,
$$\tilde{x}_{T+1}^{(i)} \sim p(x_{T+1}|x_T^{(i)}), i = 1, \ldots, N \tag{7}$$
2. Using the prior represented by the particle set $\{\tilde{x}_{T+1}^{(i)}\}_{i=1}^N$, iterate the steps 3, 4 and 5 for $t \in \{T, T-1, \ldots, 1\}$,
3. Proposition: At time t, propose a new particle set $\{\tilde{x}_t^{(i)}\}_{i=1}^N$ using the state transition model,
$$\tilde{x}_t^{(i)} \sim p(x_t|\tilde{x}_{t+1}^{(i)}), i = 1, \ldots, N \tag{8}$$
4. Weight Computation: Compute the weight $w_t^{(i)}$ associated with the particle $\tilde{x}_t^{(i)}$,
$$w_t^{(i)} = p(y_t|x_t^{(i)}) \tag{9}$$
5. Resample to obtain an unweighted particle set.
6. Using the particle set $\tilde{x}_0^{(i)} \sim q(\tilde{x}_0|\mathcal{Y}_T)$, compute mean $\hat{\mu}_\pi$ and var-covariance matrix $\hat{\Sigma}_\pi$ using sample statistics.
7. The evaluation statistic is computed using (6).

Table 1. Outline of the proposed algorithm.

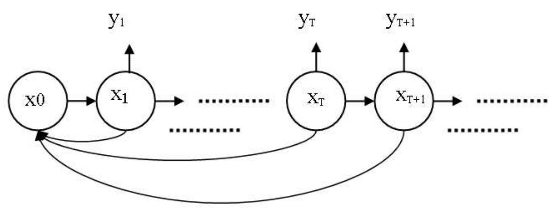

Figure 1. Schematic of the reference point used in the proposed algorithm.

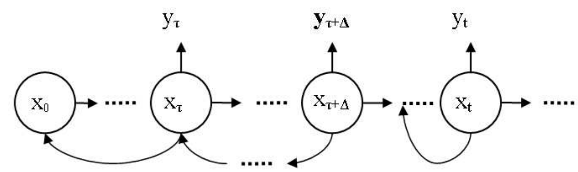

Figure 2. Schematic of the reference point used in the faster approximation to the proposed algorithm.

idea for performance characterization.

Finally, the proposed framework extends gracefully even to systems where the inference is not driven by particle filters. For example, if the system is linear Gaussian, then the posterior can be computed using a Kalman filter. The time reversed system is also linear Gaussian, and its posterior can also be computed using a Kalman filter. In this case, the time reversed posterior and the prior can be compared using (6). Given the Gaussian distribution of both distributions,as an alternative similarity score, one could analytically compute their KL divergences too. Finally, it might be possible to provide theoretical guarantees for the algorithm in this simple case.

3.4. Similarity to the ELL

The proposed evaluation methodology is similar to the ELL statistic in spirit, both involving posterior of the tracking algorithm and the prior at time $t = 0$. ELL propagates the prior density to time t and computes the inaccuracy between the t-step predicted prior and the the posterior π_t. In contrast, the proposed methodology time reverses the posterior π_t back to initial time using a time reversed system and compares it against the prior at time $t = 0$. The main difference in our formulation is the t-step reverse prediction is *conditioned* on the observed data, while the t-step prediction in ELL is unconditional.

3.5. Fast Approximation

The proposed evaluation framework poses a requirement to process (or track) across all the frames seen by the tracking algorithm. For such an algorithm, the computational requirements increase linearly with the number of frames (see Figure 1). This is in practice a steep requirement.

However, a set of sufficient (though not necessary) conditions can be designed to alleviate this problem. We argue that if the performance at time T is good, then not only does the final posterior match well with the prior density, but that the posterior densities of the forward and reverse tracker should match at all intermediate time instants. A fast approximation is now proposed using this observation. Suppose at time t_0, the performance of the system is evaluated to be good, then for an evaluation at a future time instant $t' > t_0$, the time t_0 can be used as a reference point in the place of the $t = 0$ (see Figure 2). However, the suitability of the approximation depends on the length $\Delta t = t' - t_0$. The trade-off here is between the computation time, that is proportional to Δt and the ability to detect slow changes that are of the order Δt. A clever choice of Δt can go a long way in reducing the computation requirements of the proposed algorithm.

4. Experimental Results and Discussions

4.1. Experimental Results

The proposed evaluation algorithm can detect various common failure modes in visual tracking systems. The particle filter based visual tracking system proposed in [16] is used in our experiments. The system uses an OAM in its observation model and the six-dimensional affine deformation matrices as its state space. We used the proposed evaluation methodology to study the performance of the OAM based tracker intensively over various failure modes.

Figure 3 shows results over a complete occlusion scenario, with evaluation performed once every 15 frames. The target undergoes occlusion around 110th frame. The proposed statistic and its fast approximations register peaks or sharp rises in value around this frame. Further, a fast approximation with $\Delta t = 5$ does not seem large enough to capture the tracking failure. However, a higher value of Δt registers the loss of track. It is also noticed that inference using fast approximations is not useful after a track failure is registered. This is because that reference point against which the algorithm is being compared is corrupted.

Figure 4 shows evaluation on a sequence in which a target exhibits a small change in pose, which the tracker can easily keep track of. As expected, the proposed evaluation methodology generates test statistics which takes low values indicating a good tracking performance. Figure 5 shows evaluation results on an aerial sequence in which the tracker loses track of the target due to jerky motion of the camera. The test statistics register sharp peaks around the point where the loss of track happens.

4.2. Testing with Ground Truth

The proposed algorithm was tested over sequences of the PETS-2001 dataset and the evaluation is compared with the ground truth. The comparison with the ground truth is done by computing the distance between the center of the target as hypothesized by the tracker to the ground truth. Figures 6 and 7 show the results on two sequences from the dataset. In Figure 6, the tracker tracks the object fairly well. Both the proposed statistic and the comparison against the ground truth take a low value. Figure 7 shows evaluation results over a scenario involving tracking failure. While all statistics register the failure of track, the proposed statistic registers the track failure before the ground truth. This is because of the specific evaluation criterion used with the ground truth, which involves comparing only the centers of the target, while the bounding box is inaccurate before the loss of track (frame 60).

4.3. Discussion

To summarize the results, the following properties of the proposed evaluation scheme are highlighted. The proposed evaluation algorithm is shown to detect common failure modes in visual tracking and also compares favorably with ground truth based evaluation. The value of Δt is shown to be critical in the efficiency of the fast approximations. A value of $\Delta t = 30$ seems reasonably large enough to register failures. It is also noteworthy that fast approximations are meaningless after detection of failure, as the reference point against which they are compared does not correspond to good tracking. Finally, the choice of threshold to declare poor performance can be easily decided for a specific tracking system by inspection. The choice is also influenced by the value of Δt. It can be seen that for all the experiments in this paper, the inference from the proposed evaluation agrees well with human perception.

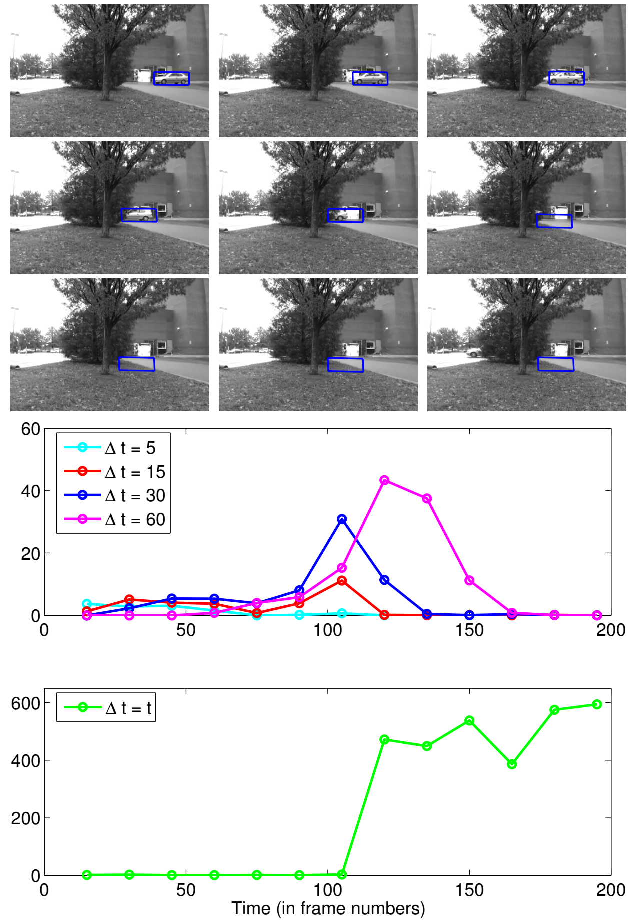

Figure 3. Performance evaluation over occlusion. Target is completely occluded by frame number 100. (Top left to bottom right) Tracking results at frame numbers 1, 20, 40, 60, 80, 100, 120, 135 and 150 (Bottom row) Evaluation results using the proposed algorithm ($\Delta t = t$) and its fast approximations ($\Delta t = 5, 15, 30, 60$).

5. Conclusions

In this paper, we present a novel method to provide automatic and online evaluation of the tracking performance in visual systems without the knowledge of ground truth. The proposed evaluation algorithm works by verifying the prior at time $t = 0$ against the posterior of a time reversed chain. The time reversed chain is initialized using the posterior of the tracking algorithm. It is postulated that when the data obey the modeling, the posterior of the time reversed chain at time $t = 0$ is close to the prior $p(x_0)$. We propose fast approximation schemes that reduce the computational cost for filtering across the whole observation sequence. The proposed algorithm has been tested extensively on datasets and it is empirically shown that the evaluation scheme works well in detecting common failure modes. While the focus in the paper has been on systems using particle filtering, the underlying algorithm is fairly independent of the actual filtering tools used. We expect that such an algorithm would also be useful for validation in linear Gaussian models. Future work involves deriving strong guarantees for the detection of failure modes, in terms of receiver operating characteristics.

Acknowledgement

This work was partially supported through the Advanced Sensors Consortium sponsored by the U.S Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-02-0008. The authors would like to thank Dr. Volkan Cevher for his valuable remarks and insights.

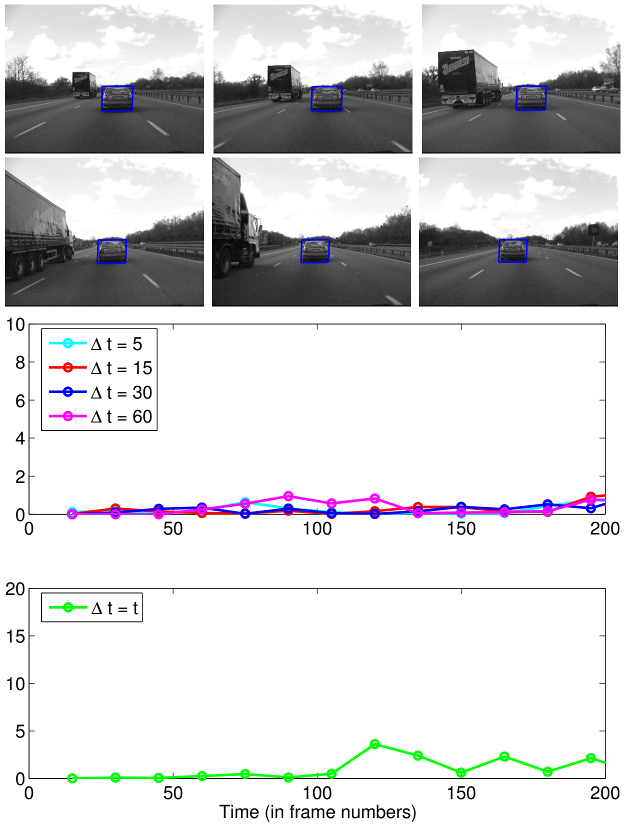

Figure 4. Performance evaluation over slow pose change. (Top three rows) Tracking results at frame numbers 1, 40, 80, 120, 160 and 200 (Bottom rows) Evaluation results using the proposed algorithm ($\Delta t = t$) and its fast approximations ($\Delta t = 5, 15, 30, 60$).

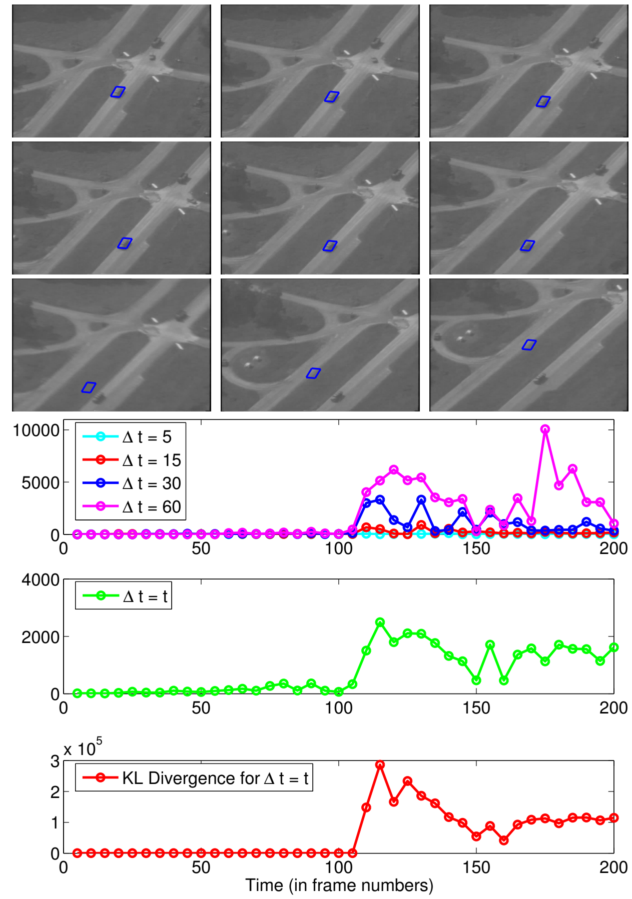

Figure 5. Performance evaluation on an aerial sequence. The tracker loses track of the object around frame 110 due to jerky camera motion. (Top three rows) Tracking results at frame numbers 1, 20, 40, 60, 80, 100, 120, 140 and 160 (Bottom row) Evaluation results using the proposed algorithm ($\Delta t = t$), its fast approximations and the KL divergence between prior density and posterior of time reversed chain.

References

[1] C. Andrieu, A. Doucet, S. Singh, and V. Tadic. Particle methods for change detection, system identification, and control. *Proceedings of the IEEE*, pages 423–438, March 2004.

[2] F. der Heijen. Consistency checks for particle filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, (1):140–145, January 2006.

[3] A. Doucet, N. D. Freitas, and N. Gordon. *Sequential Monto Carlo Methods in Practice*. New York, NY: Springer-Verlag, 2001.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[5] C. Erdem, A. Murat Tekalp, and B. Sankur. Metrics for performance evaluation of video object segmentation and tracking without ground-truth. *Proc. IEEE Int'l Conf. on Image Processing*, pages 69–72, October 2001.

[6] C. Erdem, B. Sankur, and A. Tekalp. Non-rigid object tracking using performance evaluation measures as feedback. *Proc. IEEE Int'l Conf. on Comput. Vis. and Patt. Recog.*, pages II–323– II–330, 2001.

[7] R. Gerlach, C. Carter, and R. Kohn. Diagnostics for time series analysis. *J. Time Series Anal.*, (3):309 – 330, 1999.

[8] J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures. *Proc. IEEE Int'l Conf. on Computer Vision*, pages 487–493, October 2003.

[9] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[10] M. Klass, M. Briers, N. D. Freitas, A. Doucet, S. Maskell, and D. Lang. Fast particle smoothing: If i had a million particles. *Proc. IEEE Int. Conf. on Machine Learning*, 2006.

[11] L. Lu, X. Dai, and G. Hager. A particle filter without dynamics for robust 3d face tracking. *IEEE Proc. Computer Vision and Pattern Recognition Workshops*, 2004.

[12] N. Vaswani. Additive change detection in nonlinear systems with unknown change parameters. *Accepted for IEEE Trans. Signal Processing*, 2006.

[13] J. Vermaak, C. Andrieu, A. Doucet, and S. Godsill. Particle methods for bayesian modelling and enhancement of

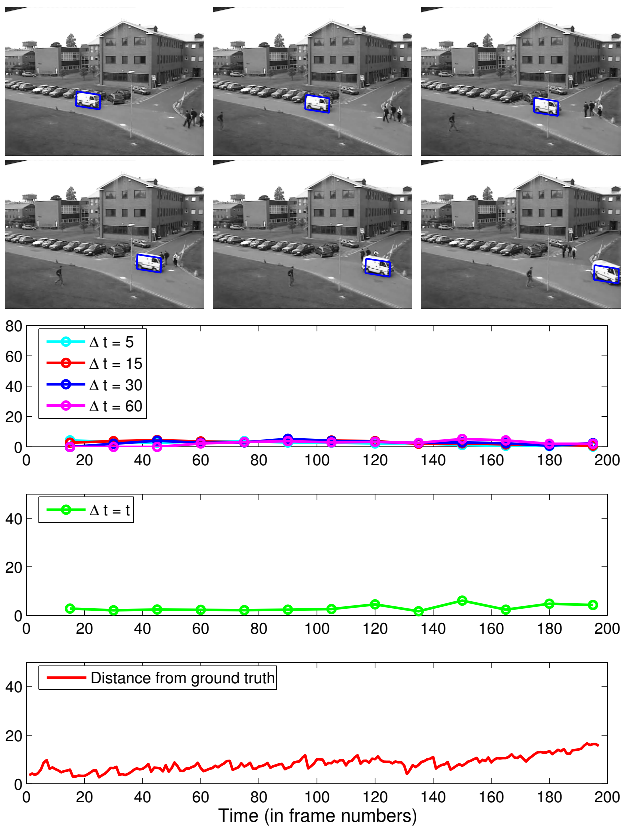

Figure 6. Performance evaluation on a PETS sequence including ground truth. (Top three rows) Tracking results at frame numbers 1, 30, 60, 90, 120 and 160. (bottom three rows) Evaluation results using proposed statistics and its fast approximations and the ground truth. Tracking performance remains fairly constant as shown by both the ground truth and the proposed evaluation.

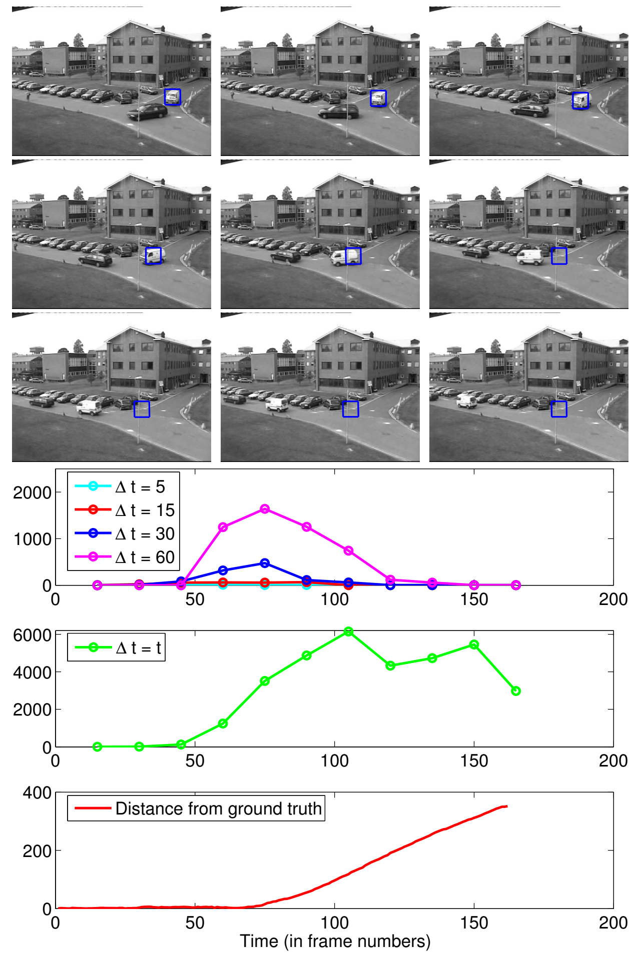

Figure 7. Performance evaluation on a PETS sequence including ground truth. (Top three rows) Tracking results at frame numbers 1, 20, 40, 60, 80, 100, 120, 140 and 160. (bottom three rows) Evaluation results using proposed statistics and its fast approximations and the ground truth.

speech signals. *IEEE Trans. Speech and Audio Processing*, (3), March 2002.

[14] H. Wu and Q. Zheng. Performance self-evaluation of visual tracking systems. *Army Science Conference*, 2004.

[15] Q. Zheng and R. Chellappa. Automatic feature point extraction and tracking in image sequences for arbitrary camera motion. *International Journal of Computer Vision*, pages 31–76, June 1995.

[16] S. K. Zhou, R. Chellappa, and B. Moghaddam. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Processing*, (11):1491–1506, November 2004.