A Unified Bayesian Framework for Adaptive Visual Tracking

Emanuel E. Zelniker
http://www.dcs.qmul.ac.uk/~zelniker
Timothy M. Hospedales
http://www.dcs.qmul.ac.uk/~tmh
Shaogang Gong
http://www.dcs.qmul.ac.uk/~sgg
Tao Xiang
http://www.dcs.qmul.ac.uk/~txiang

School of EECS,
Queen Mary University of London

Abstract

We propose a novel method for tracking objects in a video scene that undergo drastic changes in their appearance. These changes may arise due to out-of-plane rotation, abrupt or gradual changes in illumination in outdoor scenarios, or changing position with respect to near light-sources indoors. The key problem with most existing models is that they are either non-adaptive (rendering them non-robust to object appearance change) or use a single tracker output to heuristically update the appearance model at each iteration (rendering them vulnerable to drift). In this paper, we take a step toward general real-world tracking, in a principled manner, proposing a unified generative model for Bayesian multi-feature, adaptive target tracking. We show the performance of our method on a wide variety of video data, with a focus on surveillance scenarios.

1 Introduction

Tracking is regarded as one of the most fundamental tasks in computer vision. Despite decades of research, the goal of fully automatic tracking of arbitrary types of objects in real world conditions is still an open problem. There are various reasons for the challenging nature of this problem: i) Tracking arbitrary objects requires dealing with various shapes, sizes and movement dynamics. ii) Movement against cluttered backgrounds. iii) Occlusions by fixed obstacles or other mobile objects. iv) Objects may change appearance drastically by moving through changing lighting conditions, moving non-rigidly and with 3D rotation.

The flexibility of the particle filtering approach [1] in dealing with non-linear dynamics and complex observation models has popularized generative models for tracking [12, 13]. Generative models also allow integration of multiple features to improve tracking accuracy in a principled and straightforward way [2, 14, 16, 19]. Nevertheless, such models frequently have the drawback of requiring hand initialization of contours [14], failing the requirement for automatic application; or a fixed target model [13], crucially failing in the requirement of robustness to appearance change required for real-world tracking. Some studies have tried

to update appearances online in various heuristic ways [3, 12, 17, 23]. This, however, brings about the risk of gradual drifting of the target model to focus on background or clutter. These problems are in general due to the fact that we do not know if a bad match is due to target appearance change in which case the model should be updated, or due to bad tracking in which case the model should not be updated to avoid drift [12].

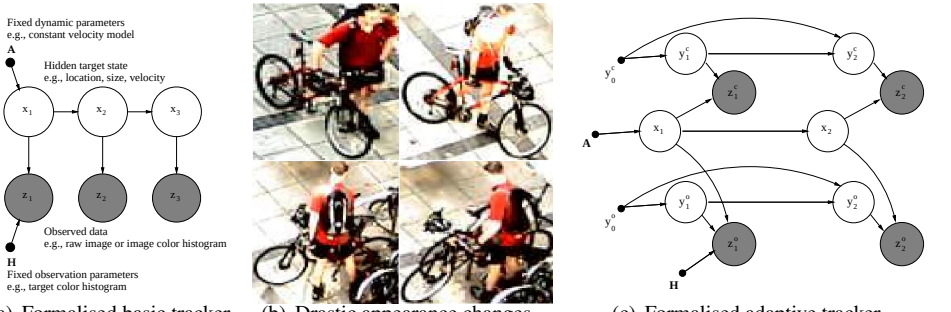
Recently, discriminative methods for tracking have gained popularity [2, 6, 18, 22]. These treat tracking as a classification problem, learning a decision boundary between the target and the background with the aim of avoiding being distracted by clutter. These have also been heuristically updated online to track appearance changes [6, 18, 22]. However there are various general problems with discriminative models. They may preclude useful options available to generative modelers, *e.g.*, to build in machinery to cope with joint target tracking [10], switching dynamics or observation models (*e.g.*, occlusion reasoning [9]), and to make use of standard algorithms for fixed lag smoothing, or whole trajectory (Viterbi) inference [7]. Moreover, they often regress to tracking objects of fixed size and scale [6, 18, 22].

The key problem with most existing models is that they are either non-adaptive (non-robust to object appearance change) or use a single tracker output to update the appearance model at each time [11, 17]. This is known as self-training [24], susceptible to self-reinforcing errors [24]. In the context of tracking, this corresponds to small errors in the target model and inaccuracies in tracking reinforcing themselves until track lost. Some recent studies have tried to get around this with co-training [24] of discriminative models. For example [22] uses two independent trackers and if one tracker is sufficiently confident about its prediction, that prediction is used to train the other and vice-versa. However, this entails crucial tuning of confidence thresholds to decide when to use each feature.

2 A Unified Bayesian Adaptive Multiple Feature Tracker

In this paper, we take a step toward the goal of general real-world tracking, and demonstrate a unified generative model for Bayesian multi-feature, adaptive target tracking, or AMFT for short (Adaptive Multiple Feature Tracker). We derive a unified generative model for multi-sensory adaptive tracking which cleanly integrates tracking and the modeling of appearance change across multiple features in the same framework. The unified multi-feature observation model ensures that if one feature is not confident, *e.g.*, color after an object crosses into a region of shadow, it is automatically down-weighted in its contribution to the appearance model update. In this way, without pre-training of specific object models, we achieve an extensible tracker for general object types, robust to real-world problems of clutter, appearance/lighting change and target model drift.

The standard modeling assumptions made by a non-adaptive generative model are illustrated by the probabilistic graphical model in Figure 1(a). The unknown target state (*e.g.*, location, size, velocity) $\mathbf{x}_t$ is assumed to change with time t according to some process parameterized by $\mathbf{A}$. At every time t, we make some noisy observations $\mathbf{z}_t$ of the target $\mathbf{x}_t$ (*e.g.*, raw image or color histograms). The target is then tracked online by computing the posterior, $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ over the true target location recursively. In the case of the Kalman filter (KF), all the distributions involved are Gaussian. In the case of the particle filter (PF), all the distributions involved are represented non-parametrically by a set of samples [1]. The true target model, *e.g.*, the appearance or color histogram to search for, is assumed to be part of the parameters H, *i.e.*, it is known and fixed by an operator or initialized by some external process. In many cases however, the true appearance of the target H may change



(a) Formalised basic tracker. (b) Drastic appearance changes. (c) Formalised adaptive tracker.

Figure 1: Graphical models for the tracking problem. Figure 1(b) shows how the appearance of a target can drastically change over a short period of time.

significantly in time, *e.g.*, the appearance changes when a subject moves between shade and sunlight. This is the case for outdoor surveillance applications and is the motivation for this research. Figure 1(b) motivates how adaptation is crucial for tracking by illustrating from ground truth data how the appearance of a target can drastically change over a period of $9\frac{1}{3}$ seconds in realistic outdoor conditions.

As discussed in Section 1, adaptive trackers [6, 12, 17, 23] have been proposed to update the target appearance online in various heuristic ways. We can formalise this more general modeling assumption generatively, by the generalized dynamic Bayesian network illustrated in Figure 1(c). In contrast to Figure 1(a), the true target model which was previously included in the fixed parameters H, is now included as the the initial condition $\mathbf{y}_0$ of a dynamic latent variable $\mathbf{y}_t$, formalizing the modeling assumption that the target appearance can change over time. In addition to the target state $\mathbf{x}_t$, the target appearance $\mathbf{y}_t$ will therefore be incrementally and recursively updated as part of the process of inferring the latent variables in this model $p(\mathbf{x}_t, \mathbf{y}_t | \mathbf{z}_{1:t})$. The latent space is of course now greatly expanded, and poses a more challenging inference problem than that of Figure 1(a). In Section 2.1, we will detail the specific parametric form of the model and an efficient inference algorithm.

2.1 Mathematical Framework

Our model is very generic and trivially extensible to any number of features, which assuming they are not correlated or degenerate, will increase performance. For concreteness, we will describe the model in terms of target color $\mathbf{y}_t^c$ and orientation $\mathbf{y}_t^o$ histograms and their observations $\mathbf{z}_t^c$ and $\mathbf{z}_t^o$ as well as target state $\mathbf{x}_t$. The joint probability is:

$$p(\mathbf{z}_{1:t}^o, \mathbf{z}_{1:t}^c, \mathbf{x}_{1:t}, \mathbf{y}_{1:t}^o, \mathbf{y}_{1:t}^c | \theta) = \prod_t p(\mathbf{z}_t^o | \mathbf{y}_t^o, \mathbf{x}_t, \theta) p(\mathbf{z}_t^c | \mathbf{y}_t^c, \mathbf{x}_t, \theta) p(\mathbf{y}_t^o | \mathbf{y}_{t-1}^o, \theta) p(\mathbf{y}_t^c | \mathbf{y}_{t-1}^c, \theta) p(\mathbf{x}_t | \mathbf{x}_{t-1}, \theta), \quad (1)$$

$$p(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t | \mathbf{A}^{\mathbf{x}} \mathbf{x}_{t-1}, \mathbf{Q}^{\mathbf{x}}), \quad (2)$$

$$p(\mathbf{y}_t^c | \mathbf{y}_{t-1}^c) = \mathcal{N}(\mathbf{y}_t^c | \mathbf{A}^c \mathbf{y}_{t-1}^c + \mathbf{B}^c \mathbf{y}_0^c, \mathbf{Q}^c), \quad (3)$$

$$p(\mathbf{y}_t^o | \mathbf{y}_{t-1}^o) = \mathcal{N}(\mathbf{y}_t^o | \mathbf{A}^o \mathbf{y}_{t-1}^o + \mathbf{B}^o \mathbf{y}_0^o, \mathbf{Q}^o), \quad (4)$$

$$p(\mathbf{z}_t^c | \mathbf{y}_t^c, \mathbf{x}_t) = \mathcal{N}(\mathbf{z}_{\mathbf{x}_t,t}^c | \mathbf{y}_t^c, \mathbf{R}^c), \quad (5)$$

$$p(\mathbf{z}_t^o | \mathbf{y}_t^o, \mathbf{x}_t) = \mathcal{N}(\mathbf{z}_{\mathbf{x}_t,t}^o | \mathbf{y}_t^o, \mathbf{R}^o). \quad (6)$$

where $\mathbf{R}$ and $\mathbf{Q}$ are covariance parameters [20], $\mathbf{A}$ and $\mathbf{B}$ control how strongly the initial appearance is weighted (discussed further below) and $\theta = \{\mathbf{Q}^{\mathbf{x},c,o}, \mathbf{A}^{\mathbf{x},c,o}, \mathbf{R}^{c,o}, \mathbf{B}^{c,o}, \mathbf{y}_0^{c,o}\}$ includes all the model parameters (for brevity we shall subsequently assume conditioning on

all relevant parameters θ). Multi-feature appearance adaptive tracking is carried out by recursively inferring the posterior $p(\mathbf{x}_{1:t}, \mathbf{y}^c_{1:t}, \mathbf{y}^o_{1:t}, |\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t})$. The exact Bayesian filter for this distribution is given by

$$p(\mathbf{y}^c_t, \mathbf{y}^o_t, \mathbf{x}_t|\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t}) = p(\mathbf{z}^c_t|\mathbf{x}_t, \mathbf{y}^c_t)p(\mathbf{z}^o_t|\mathbf{x}_t, \mathbf{y}^o_t) \int_{\mathbf{x}_{t-1}, \mathbf{y}^c_{t-1}, \mathbf{y}^o_{t-1}} \frac{p(\mathbf{x}_t, \mathbf{y}^c_t, \mathbf{y}^o_t|\mathbf{x}_{t-1}, \mathbf{y}^c_{t-1}, \mathbf{y}^o_{t-1})p(\mathbf{x}_{t-1}, \mathbf{y}^c_{t-1}, \mathbf{y}^o_{t-1}|\mathbf{z}^c_{1:t-1}\mathbf{z}^o_{1:t-1})}{p(\mathbf{z}^c_t, \mathbf{z}^o_t|\mathbf{z}^c_{1:t-1}, \mathbf{z}^o_{1:t-1})}. \quad (7)$$

This is not, however, analytically tractable and in practice, the target model, *e.g.*, appearance, color or orientation histograms, is in a high dimensional space. It is therefore prohibitive to infer the entire model with any standard particle filtering approach. However, by assuming the appearance distribution is Gaussian, we can derive a collapsed, or Rao-Blackwellised [5], inference algorithm which exploits the conditional Gaussianity of $\mathbf{y}$ given $\mathbf{x}$ for an efficient hybrid of approximate and exact inference. Specifically, given the following general factorization $p(\mathbf{y}^c_{1:t}, \mathbf{y}^o_{1:t}, \mathbf{x}_{1:t}|\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t}) = p(\mathbf{y}^c_{1:t}, \mathbf{y}^o_{1:t}|\mathbf{x}_{1:t}, \mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t})p(\mathbf{x}_{1:t}|\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t})$ of the complete posterior and exploiting the conditional independences encoded in the model, we have

$$p(\mathbf{y}^c_{1:t}, \mathbf{y}^o_{1:t}, \mathbf{x}_{1:t}|\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t}) = p(\mathbf{y}^o_{1:t}|\mathbf{x}_{1:t}, \mathbf{z}^o_{1:t})p(\mathbf{y}^c_{1:t}|\mathbf{x}_{1:t}, \mathbf{z}^c_{1:t})p(\mathbf{x}_{1:t}|\mathbf{z}^c_{1:t}, \mathbf{z}^o_{1:t}). \quad (8)$$

Conditioned on target state $\mathbf{x}$, the distributions involving target appearance $\mathbf{y}$ are all Gaussian. We can therefore solve this inference problem by recursively sampling $\mathbf{x}$, integrating $\mathbf{y}$ exactly, and propagating the exact distributions over $\mathbf{y}$ given these samples.

Stability & Adaptability: One explicit or implicit assumption made by adaptive trackers is whether to exploit any memory of the initial appearance of the target beyond that of the initial condition for the learned model. Purely using the initial appearance to define an initial condition for online learning (*e.g.*, [11, 12, 23]) usually means that the tracker can potentially adapt to any possible appearance, but may make it more prone to instability and drift as it can potentially lock on to a wider range of distracting clutter. Other schemes which retain the initial appearance, such as the semi-supervised learning [6, 18] may improve stability by preferring adaptation within a region around the initial appearance of the target, at the cost of potentially being unable to track targets which change appearance dramatically. Which of these approaches is preferable depends on the kind of data, *i.e.*, how dramatically appearance is expected to change. Our model explicitly recognizes the continuum between non-adaptive fixed appearance tracking and fully adaptive tracking in which the initial appearance is only an initial condition–both of which are special cases. The parameters $\mathbf{A}$ and $\mathbf{B}$ are manually set and control how strongly the initial appearance is weighted (Equations 3 and 4), with $\mathbf{A} = \mathbf{1}$ and $\mathbf{B} = \mathbf{0}$ giving fixed and fully adaptive template tracking respectively. If $\mathbf{A}$ and $\mathbf{B}$ are non-zero and sum to $\mathbf{1}$, this reflects the generative modeling assumption (which can be seen from sampling from this model) that we expect the appearance to vary, but around a mean value of the initial $\mathbf{y_0}$.

2.2 Inference Procedure

Prior Probability: Assume the approximate filtered distributions from the previous time steps are available as properly weighted sets of P target state samples, and sample conditional Gaussian $\{w_{1:t-1}, \mathbf{x}_{1:t-1}, p(\mathbf{y}^c_{1:t-1}|\mathbf{z}^c_{1:t-1}, \mathbf{x}_{1:t-1}), p(\mathbf{y}^o_{1:t-1}|\mathbf{z}^o_{1:t-1}, \mathbf{x}_{1:t-1})\}^P_{i=1}$ appearance distributions. That is,

$$p(\mathbf{y}^c_{1:t-1}, \mathbf{y}^o_{1:t-1}, \mathbf{x}_{1:t-1}|\mathbf{z}^c_{1:t-1}, \mathbf{z}^o_{1:t-1}) \approx \sum_i^P w^i_{1:t-1}\delta(\mathbf{x}^i_{1:t-1})\mathcal{N}(\mathbf{y}^c_{1:t-1}|\mu^{c,i}_{1:t-1}, P^{i,c}_{1:t-1})\mathcal{N}(\mathbf{y}^o_{1:t-1}|\mu^{i,o}_{1:t-1}, P^{i,o}_{1:t-1}), \quad (9)$$

where the conditional Gaussian appearance distributions are parameterized by mean and variance sufficient statistics $\mu^{i,c}_t$ and $P^{i,c}_t$.

Prediction: The predictive distribution $p(\mathbf{x}_t, \mathbf{y}_t^c, \mathbf{y}_t^o | \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}^c, \mathbf{y}_{1:t-1}^o, \mathbf{z}_{1:t-1}^c, \mathbf{z}_{1:t-1}^o)$ is given by sampling $\mathbf{x}_t^i$ from its forward model $\mathbf{x}_t \sim p(\mathbf{x}_t | \mathbf{x}_{t-1})$. Then, for each sample i, the Gaussian sufficient statistics for each feature, is effectively Kalman prediction, *e.g.*, for color:

$$p(\mathbf{y}_t^c | \mathbf{z}_{1:t-1}^c, \mathbf{x}_{1:t-1}^i) = \int p(\mathbf{y}_t^c | \mathbf{y}_{t-1}^c) p(\mathbf{y}_{t-1}^c | \mathbf{z}_{1:t-1}^c, \mathbf{x}_{1:t-1}^i) d\mathbf{y}_{t-1}^c = \mathcal{N}(\mu_{t-1}^c, \mathbf{P}_c^{f,-}), \tag{10}$$

where superscript $-$ indicates pre-observation statistics, $\mu_t^{c,-} = \mathbf{A}^c \mu_{t-1}^c + \mathbf{B}^c \mathbf{y}_0^c$ and $\mathbf{P}_t^{c,-} = \mathbf{Q}^c + \mathbf{A}^c \mathbf{P}_{t-1}^c \mathbf{A}^{cT}$.

Posterior Update: The posterior sufficient statistics for the $\mathbf{x}_t^i$ conditional appearance model of each particle i, and the sample weighting w_t^i are updated to reflect the new observations $\mathbf{z}_t^c$ and $\mathbf{z}_t^o$. The appearance update for each feature is given by, *e.g.*, for c:

$$p(\mathbf{y}_t^c | \mathbf{z}_{1:t}^c, \mathbf{x}_{1:t}^i) \propto p(\mathbf{z}_t^c | \mathbf{y}_t^c, \mathbf{x}_t^i) p(\mathbf{y}_t^c | \mathbf{z}_{1:t-1}^c, \mathbf{x}_{1:t-1}^i) = \mathcal{N}(\mathbf{y}_t^c | \mu_t^c, \mathbf{P}_t^c), \tag{11}$$

where sufficient statistics $\mu_t^c = \mu_t^{c,-} + \mathbf{K}_t^c (\mathbf{z}_t^c - \mu_t^{c,-})$, $\mathbf{P}_t^c = (\mathbf{I} - \mathbf{K}_t^c) \mathbf{P}_t^{c,-}$ and $\mathbf{K}_t^c = \mathbf{P}_t^{c,-} (\mathbf{P}_t^{c,-} + \mathbf{R}^c)^{-1}$ are as in the standard Kalman update equations [20].

Given that we used the forward model $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ as the proposal for $\mathbf{x}_t$, the posterior importance sampling weights are given by the marginal likelihood of the new data. Given the model structure (Figure 1(c)), we can exploit the factorization $p(\mathbf{z}_t^c, \mathbf{z}_t^o | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^c, \mathbf{z}_{1:t-1}^o) = p(\mathbf{z}_t^c | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^c) p(\mathbf{z}_t^o | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^o)$ as follows:

$$w_t^i \propto w_{t-1}^i p(\mathbf{z}_t^c, \mathbf{z}_t^o | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^c, \mathbf{z}_{1:t-1}^o), \tag{12}$$

$$p(\mathbf{z}_t^c, \mathbf{z}_t^o | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^c, \mathbf{z}_{1:t-1}^o) = \int p(\mathbf{z}_t^c, \mathbf{y}_t^c | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t}^c) d\mathbf{y}_t^c \int p(\mathbf{z}_t^o, \mathbf{y}_t^o | \mathbf{x}_{1:t}^i, \mathbf{z}_{1:t-1}^o) d\mathbf{y}_t^o, \tag{13}$$

$$p(\mathbf{z}_t^c, \mathbf{z}_t^o | \mathbf{x}_t, \mathbf{z}_{1:t-1}) = \mathcal{N}(\mathbf{z}_t^c | \mu_{t-1}^c, \mathbf{P}_{t-1}^c + \mathbf{R}^c) \mathcal{N}(\mathbf{z}_t^o | \mu_{t-1}^o, \mathbf{P}_{t-1}^o + \mathbf{R}^o). \tag{14}$$

The posterior distribution $p(\mathbf{y}_{1:t}^c, \mathbf{y}_{1:t}^o, \mathbf{x}_{1:t} | \mathbf{z}_{1:t}^c, \mathbf{z}_{1:t}^o, \theta)$ is represented by the updated set of weighted samples, $\{w_{1:t}, \mathbf{x}_{1:t}, p(\mathbf{y}_{1:t}^c | \mathbf{z}_{1:t}^c, \mathbf{x}_{1:t}), p(\mathbf{y}_{1:t}^o | \mathbf{z}_{1:t}^o, \mathbf{x}_{1:t})\}_{i=1}^P$ through the appending of the updates from Equations (11, 12).

To summarize intuitively, this inference algorithm for Figure 1(c) jointly models target state and appearance by attaching a hypothesized distribution over the target appearance and position to each particle (Equation 9). The position likelihoods are determined by the match between the hypothesized appearance and the pixels at the associated image location (Equation 12), and the target appearance models are then updated according to the pixels at the hypothesized image location (Equation 11). If only a filtered estimate of the target state and appearance is required, the historical samples may be discarded.

2.3 Implementation

We use standard features suitable for representation of generic targets, specifically, normalised color histograms and normalized histograms of gradients (HoG). The target spatial state is represented by $\mathbf{x} = [x, y, w, h, v_x, v_y]$, *i.e.*, spatial (x, y), size (w, h) and velocity (v_x, v_y) parameters represented as particles, and the target appearance models ($\mathbf{y}^c$ and $\mathbf{y}^o$) and observations thereof ($\mathbf{z}^c$ and $\mathbf{z}^o$) are represented by the bin values of color and orientation histograms, modeled as Gaussians. The color histograms are defined on R, G and B channels with 256 bins, and the orientation histogram uses 181 bins. Covariance parameters $\mathbf{R}^{c,o}$ and $\mathbf{Q}^{c,o,x}$ are assumed to be diagonal. These details can be generalized in a straightforward manner in various ways, *e.g.*, more features or using a more complicated mixture proposal to spawn particles at locations detected by a foreground/background model. For convenience

we provide a summary of the algorithm below

Algorithm Summary:

Assume the full posterior over target appearance and state at time t represented approximately as $\{w_{t-1}, \mathbf{x}_{t-1}, p(\mathbf{y}^c_{t-1}|\mathbf{z}^c_{1:t-1}, \mathbf{x}_{1:t-1}), p(\mathbf{y}^o_{t-1}|\mathbf{z}^o_{1:t-1}, \mathbf{x}_{1:t-1})\}_{i=1}^{P}$, then

- For particles $i = 1 \ldots P$, do:
  1. Sample state particles $\mathbf{x}^i_t \sim \mathcal{N}(\mathbf{x}_t|\mathbf{x}^i_{t-1}, \mathbf{Q}^{\mathbf{x}})$.
  2. Compute exact appearance predictions $p(\mathbf{y}^c_t|\mathbf{z}^c_{1:t}, \mathbf{x}^i_t)$ (10).
  3. Compute new weights w^i_t as the marginal likelihood of the observations given $\mathbf{x}^i_t$ (12,14).
  4. Update appearance posterior $p(\mathbf{y}^c_t|\mathbf{z}^c_{1:t}, \mathbf{x}^i_{1:t}, \theta)$ (11).
- Resample particles in proportion to their weights w^i_t.
- Point estimate state $\hat{\mathbf{x}}$ and appearances $\hat{\mathbf{y}}^c, \hat{\mathbf{y}}^o$ as mean of particle distribution.

3 Experiments

We evaluate our method (AMFT) against three contemporary approaches: A standard single feature particle filter (PF), mean-shift (MS) [2] and incremental visual tracking (IVT) [17]. The PF and MS trackers are non-adaptive color-based trackers, while IVT aims for pose and illumination change robustness by performing online adaptation in a subspace appearance model. Note that the AMFT, PF and IVT trackers track object scale, but MS does not.

We evaluated these methods on a series of challenging video clips exhibiting a wide variety of data and object types for tracking, including far-field indoor and outdoor pedestrians with and without carried objects, vehicle tracking, and near-field indoor face tracking. Extensive appearance variations were due to out-of-plane rotation, shadows and lighting. These occurred over short time intervals, the clips ranged from approximately two seconds to 14.6 seconds.

PETS2006 Pedestrian [15]: Figure 2(a-d) illustrate a relatively easy indoor pedestrian tracking sequence from the PETS 2006 database. In this sequence, all the trackers perform fairly well except MS (blue). This is because MS depends heavily on the color content of the target and spatial overlap between frames, which are both fairly weak in this case. There is relatively little appearance change, so the standard PF performs as well as the adaptive trackers.

Cyclist: The outdoor cyclist tracking sequence illustrated in the Introduction is evaluated in Figure 2(e-h). Here the brightness saturation in the middle of the sequence, out-of-plane rotation of the person with a backpack and bicycle, and movement of the target into a cluttered region of other bicycles make this sequence challenging. Our AMFT model (green) successfully tracks the cyclist to the end of the sequence. In contrast, by Figure 2(g-h) the standard PF (magenta) and MS (blue) trackers get lost in the clutter and the IVT tracker (red) fails to follow the cyclist into the bike rack at all.

Face: A completely different type of in-door near-field face tracking problem is evaluated in Figure 2(i-l). Out-of-plane rotation of the face and walking under celling lighting result in challenging appearance and color-variations. Moreover, the moving camera results in a non-stationary background. These challenges cause MS (blue) and the standard PF (magenta) to

fail. The facial structure is relatively consistent for most of the sequence so IVT does fairly well until Figure 2(l), where in-plane rotation causes it to drift slightly. The AMFT model successfully adapts to the variations performing fairly well throughout.

iLIDS cars [8]: Figure 2(m-p) illustrates a car-tracking problem from the iLIDS vehicle database. The strong sunlight casts a shadow across the road causing a challenge for color-based trackers. By Figure 2(o), the color change and fast movement has caused MS to loose the target. As the car moves closer to the corner, its pixel-wise velocity increases as it gets closer to the camera, and its appearance change increases as it turns, rotating out-of-plane. As a result the standard PF and IVT trackers are loosing the target by Figure 2(p), while AMFT continues to track well.

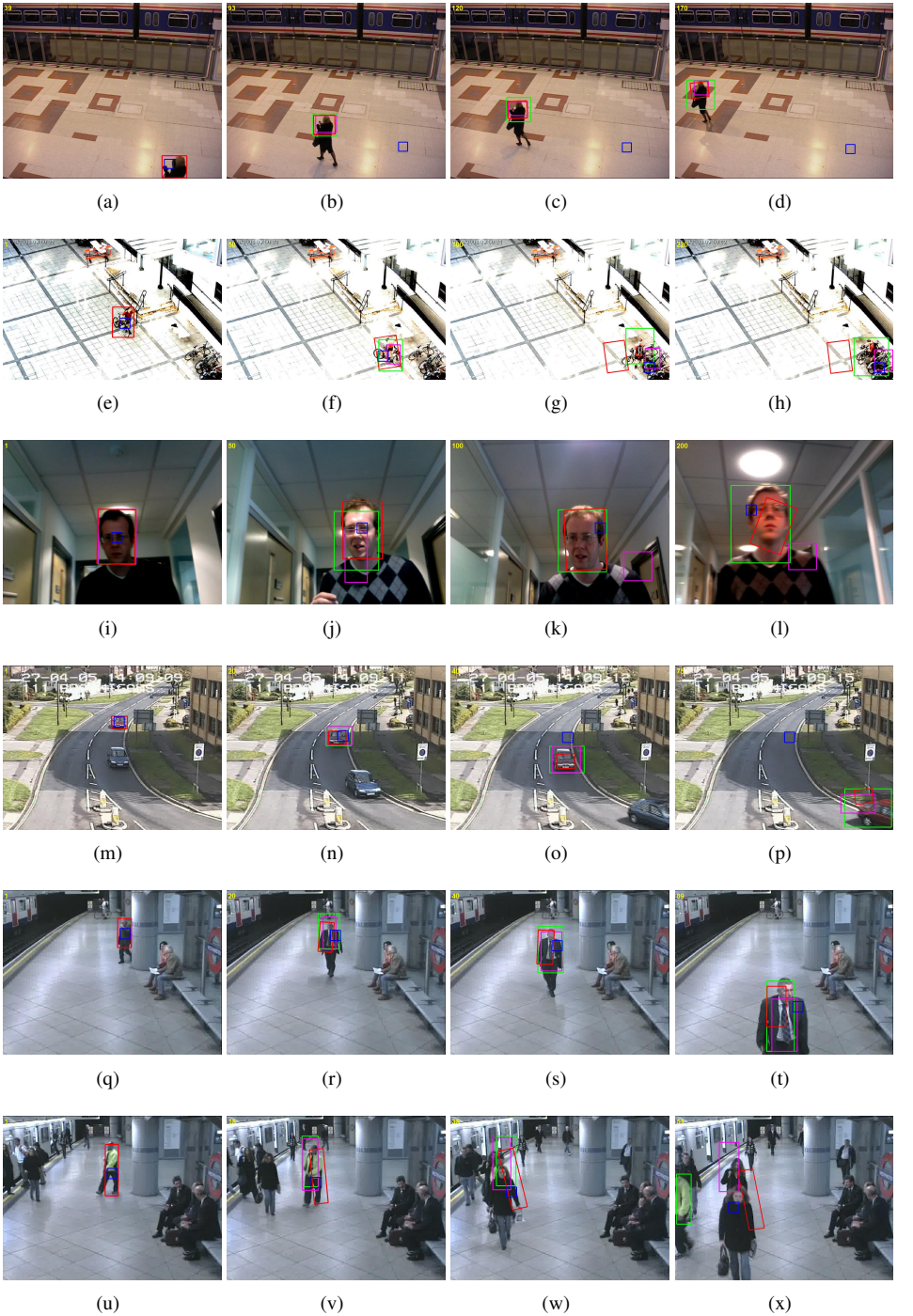
iLIDS underground 1: Figure 2(q-t) illustrates a pedestrian-tracking problem from the iLIDS underground database. In this case there is very strong scale and appearance change as the target turns and walks towards the camera. These changes turn out to be too strong for MS, the standard PF and IVT to cope with, while the AMFT copes better. This sequence demonstrates the importance of the key contributions of this paper, *i.e.*, adaptation and multi-feature fusion, as can be seen in Figure 3. Specifically, Using standard non-adaptive trackers (magenta) is insufficient. Adaptively, using either color alone (red dashed window) or orientation alone (red dotted window) the model does not track the target accurately, and it eventually looses focus and learns to track a distracting nearby person. Using both features combined (solid green window, as in Figure 2(q-t)), the target tracking is more accurate, the model maintains focus on the target and adapts to the scale and appearance changes.

iLIDS underground 2: Figure 2(u-x) illustrates another pedestrian-tracking problem with more clutter and partial occlusion. In this case the brightly colored man is fairly easy to follow, except that an adaptive tracker has the risk of learning and locking onto other nearby people while he is partially occluded (see Figure 4 and Section 2.1 Stability & Adaptability). The standard non-adaptive PF tracker drifts, as does MS, the adaptive IVT tracker learns the occluder and fails and the AMFT tracker tracks fairly smoothly the whole time as it adapts somewhat to represent the partially occluded target, but with enough memory to regain it easily at the end.

Finally, Table 1 summarises Root Mean Square Error (RMSE) results between the tracks in each sequence and the corresponding manual ground truth trajectories for all sequences, quantitatively demonstrating the performance of our method compared to the others. We note that for the face sequence, IVT had a lower RMSE, however this tracker was designed specifically for face type data and still failed to estimate rotation correctly.

4 Conclusions

We have presented a novel, principled method for adaptively tracking targets based on multiple features. By fusing multiple features, the overall posterior over target location $p(\mathbf{x}|\mathbf{z}_{1:t})$ is sharper and more accurate even when one individual feature is uninformative due to low contrast or clutter (Figure 3). In addition to improving tracking accuracy, this counteracts some risks of vanilla *self-training* [24] often used by adaptive trackers [11, 17] in which the output of a single tracker is used to train its model of the target, potentially allowing some frames with inaccurate or low-confidence estimates to induce errors in the target model which are progressively accumulated until target loss (Figure 2(q-t)). The second way in which we address the stability-adaptability issue compared to [11, 17] is to represent the modeling assumption that targets may change appearance arbitrarily, but do so around a mean value of



(a) (b) (c) (d)

(e) (f) (g) (h)

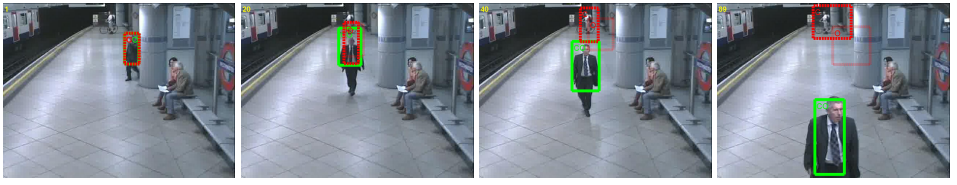
(i) (j) (k) (l)

(m) (n) (o) (p)

(q) (r) (s) (t)

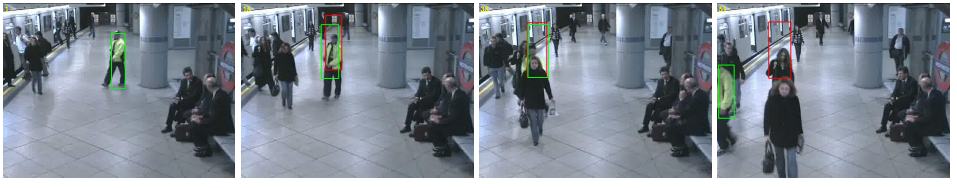
(u) (v) (w) (x)

Figure 2: Adaptive multi-feature tracking results with comparisons to other methods. AMFT (green), standard PF (magenta), IVT [] (red) MS [] (blue).



(a) (b) (c) (d)

Figure 3: Importance of fusion for adaptive tracking. Adaptive color only (dashed red), orientation only (dotted red), and fused tracking (solid green; same as Figure 2(q-t))



(a) (b) (c) (d)

Figure 4: Comparison of AMFT with full adaptation (solid red) versus AMFT with adaptation and memory (solid green; same as Figure 2(u-x)).

their initial condition. This improves the robustness of our adaptive tracker, especially when dealing with robustness to partial occlusion (Figure 2(u-x)).

We demonstrated the effectiveness of our method by evaluating it on real-world data on targets undergoing strong appearance change (lighting and out-of-plane rotation) where other adaptive and non-adaptive trackers failed. Moreover, the flexibility of our approach is illustrated by its successful application to a wide variety of object types (pedestrians, cyclists, cars, faces) and scenarios (indoor, and outdoor, near and far field). This is in contrast to typical application customized trackers, *e.g.*, indoor near-field faces [17] indoor medium-field pedestrians [2].

There are some limitations of our approach which are our topics of future research. Our tracker is only single-target aware: we can track multiple targets independently, but without joint target tracking [2, 10] there is potential for nearby targets to be confused. Another consideration with the model as presented is that we learn a single probabilistic model of the target appearance given variations seen up to the current time. We can track targets that are ultimately *completely* different in appearance from the initial frame (unlike *e.g.*, [6] which require some similarity to the initial frame), but in the event of track loss, reacquisition is unlikely. A straightforward solution we have used is to include a hierarchical switching dynamic model which can optionally recreate particles with the initial condition mean. Another key topic we are investigating is automatic learning of model parameters from data. Like most other contemporary models in the literature [11, 17] there are a few parameters in our model (observation and process covariances in this case) which may need to be set per-application scenario, but should ideally be learned automatically. One advantage of our generative framework is that, as a first step, standard dynamic linear Gaussian learning algorithms can be used for calibrating these from a sample ground truth trajectory in each application scenario. Finally, the scene background should also be modeled [10] to help

	Cyclist	PETS 2006	Face	iLIDS cars	iLIDS underground 1	iLIDS underground 2
AMFT	**78.04**	**29.94**	51.78	**48.89**	**105.68**	**88.92**
PF	106.29	53.24	201.60	53.58	123.17	97.53
IVT	100.49	69.86	**24.98**	70.91	140.43	92.63
MS	425.12	86.95	178.78	186.88	209.02	189.82

Table 1: RMSE results between the tracks in each sequence and the corresponding manual ground truth trajectories for all sequences.

distinguish the target from the surroundings to ensure even more robust adaptation.

References

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.

[2] S. Avidan. Support vector tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):1064–1072, 2004.

[3] C. Bibby and I. Reid. Robust real-time visual tracking using pixel-wise posteriors. In *Proceedings of ECCV*, 2008.

[4] Y. Chen and Y. Rui. Real-time speaker tracking using particle filter sensor fusion. *Proceedings of the IEEE*, 92(3):485–494, March 2004.

[5] A. Doucet, N. de Freitas, K. Murphy, and S. Russel. Rao-blackwellised particle filtering for dynamic bayesian networks. In *UAI*, 2000.

[6] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *Proceedings of ECCV*, 2008.

[7] M. Han, W. Xu, H. Tao, and Y. Gong. Multi-object trajectory tracking. *Machine Vision Applications*, 18:221–232, 2007.

[8] HOSDB. Imagery library for intelligent detection systems (i-lids). IEEE Conf. on Crime and Security, 2006.

[9] T. Hospedales and S. Vijayakumar. Structure inference for bayesian multisensory scene understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (12):2140–2157.

[10] Z. Khan, T. Balch, and F. Dellaert. MCMC-based particle filtering for tracking a variable number of interacting targets.

[11] Z. Khan, T. Balch, and F. Dellaert. A Rao-Blackwellized particle filter for eigentracking. In *Proceedings of CVPR*, 2004.

[12] A. Lehuger, P. Lechat, and P. Perez. An adaptive mixture color model for robust visual tracking. In *Proceedings of ICIP*, pages 573–576, 2006.

[13] E. Maggio, F. Smeraldi, and A. Cavallaro. Adaptive multi-feature tracking in a particle filtering framework. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(10):1348–1359, October 2007.

[14] F. M. Noguer, A. Sanfeliu, and D. Samaras. Dependent multiple cue integration for robust tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (4):670–685, April 2008.

[15] IEEE Performance Evaluation of Tracking and Surveillance. PETS2006 benchmark data, 2006. `http://www.cvg.rdg.ac.uk/PETS2006/data.html`.

[16] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, 2004.

[17] D. Ross, J. Lim, R. S. Lin, and M. H. Yang. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77:125–141, 2008.

[18] F. Tang, S. Brennan, Q. Zhao, and H. Tao. Co-tracking using semi-supervised support vector machines. In *Proceedigns of ICCV*, 2007.

[19] P. Torr, R. Szelinski, and P. Anandan. An integrated bayesian approach to layer extraction from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):297–303, March 2001.

[20] G. Welch and G. Bishop. An introduction to the kalman filter. Technical Report TR 95-041, University of North Carolina at Chapel Hill, 2006.

[21] C. Yang, R. Duraiswami, A. Elgammal, and L. Davis. Real-time kernel-based tracking in joint feature-spatial spaces. Technical Report CS-TR-4567, University of Maryland, 2004.

[22] Q. Yu, T. B. Dinh, and G. Medioni. Online tracking and reacquisition using co-trained generative and discriminative trackers. In *Proceedings of ECCV*, 2008.

[23] E. E. Zelniker, S. Gong, and T. Xiang. Global abnormal behaviour detection using a network of CCTV cameras. In *Proceedings of ECCV Visual Surveillance Workshop*, 2008.

[24] X. Zhu. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Science.