Adaptive Appearance Model and Condensation Algorithm for Robust Face Tracking

Yui Man Lui, *Student Member, IEEE*, J. Ross Beveridge, *Member, IEEE*, and L. Darrell Whitley

***Abstract*—We present an adaptive framework for condensation algorithms in the context of human-face tracking. We attack the face tracking problem by making factored sampling more efficient and appearance update more effective. An adaptive affine cascade factored sampling strategy is introduced to sample the parameter space such that coarse face locations are located first, followed by a fine factored sampling with a small number of particles. In addition, the local linearity of an appearance manifold is used in conjunction with a new criterion to select a tangent plane for updating an appearance in face tracking. Our proposed method seeks the best linear variety from the selected tangent plane to form a reference image. We demonstrate the effectiveness and efficiency of the proposed method on a number of challenging videos. These test video sequences show that our method is robust to illumination, appearance, and pose changes, as well as temporary occlusions. Quantitatively, our method achieves the average root-mean-square error at 4.98 on the well-known dudek video sequence while maintaining a proficient speed at 8.74 ft/s. Finally, while our algorithm is adaptive during execution, no training is required.**

***Index Terms*—Adaptive appearance model, adaptive condensation algorithm, face tracking, tangent-plane selection.**

I. Introduction

OBJECT tracking is an active computer vision research topic [1]. Among many visual objects, human faces are often the subject of interest. Tracking human faces is particularly important in the tracking community, and it has led to many applications. These applications include video surveillance, human–computer interface, biometrics, etc. However, face tracking continues to be a challenging problem due, in part, to nonrigid motion, appearance variations, illumination changes, and occlusions. Some examples of these challenges are shown in Fig. 1.

For visual tracking, observations take place sequentially, and predictions are determined as soon as image frames arrive. Moreover, real-world data are often complex. To succeed, a tracking algorithm needs to be both efficient and effective. Recently, sequential Monte Carlo (SMC) techniques [2]–[4] have received attention because of their ability to escape from local minima and their applicability to non-Gaussian data. For example, these techniques have been demonstrated to be robust relative to partial occlusion.

Manuscript received November 29, 2008; revised May 27, 2009. This work was supported by the National Science Foundation under Grant 0413284. This paper was recommended by Guest Editor K. W. Bowyer.

The authors are with the Department of Computer Science, Colorado State University, Fort Collins, CO 80523 USA (e-mail: lui@cs.colostate.edu; ross@cs.colostate.edu; whitley@cs.colostate.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCA.2010.2041655

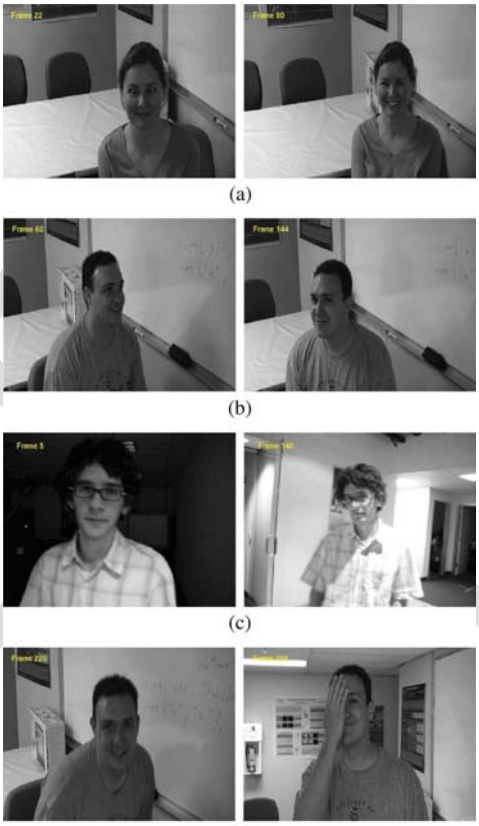

Fig. 1. Challenges of face tracking. (a) Different expressions. (b) Different poses. (c) Different illuminations. (d) Occlusion.

A condensation algorithm [5], also known as bootstrap filters [3], is an example of an SMC method. This technique adopts factored sampling to form a set of weighted samples. As such, the most probable object movements can be located and tracked accordingly. However, most existing condensation algorithms apply sequential importance sampling with a fixed Gaussian envelope [5], [6] and ignore the temporal information and match qualities. This canonical sampling strategy may be widespread rather than concentrating on the most probable regions. Hence, it requires more particles to sample the space. In this paper, we introduce an adaptive condensation algorithm which uses cascade factored sampling to sample the search space.

Facial appearance can change dramatically during a tracking process because of various expressions, poses, and illuminations. An appearance model for face tracking is to capture these variations and models it for a novel image (incoming frame).

1083-4427/$26.00 © 2010 IEEE

For online tracking which does not have any prior training, the appearance model must learn the facial variations using previous frames. Therefore, it is considered as a harder problem but could potentially have more applications. Appearance models can be represented in a variety of ways. The common appearance models are template-based [7], [8], view-based [9], [10], and feature-based models [11], [12]. Our appearance model may be regarded as an online template-based model.

Incremental subspace learning [6], [13], [14] is a popular method for online visual tracking. This technique updates a subspace sequentially, and the tracked object is declared when the minimum distance is found between the observation and the subspace. While these methods update a subspace efficiently, they can only express an image in a linear fashion. It has been argued that images actually reside in a nonlinear space [15], [16]. In this paper, we mark the curved nature of an image space, which we call appearance manifold, and characterize local linearity to form a tangent plane in which the template is built adaptively.

A condensation algorithm utilizes a factored sampling strategy to form multiple hypotheses and computes the likelihood for each prediction in accordance with an observation. For human-face tracking using a condensation algorithm, two factors play a vital role in robustness. The first is to have effective sampling. This factor determines the geometry of the face, including where it appears in the image. The second is to update the appearance effectively. Even a good sampling strategy will fail if the appearance model does not associate a good predicted appearance with an essentially correct location. Therefore, the appearance model should adapt over time, be able to generate high-fidelity matches where a true face appears, and avoid generating false matches for hypothesized facial geometries that are partially or totally in error.

In this paper, we propose a new adaptive framework for condensation algorithms. We introduce an adaptive affine cascade factored sampling and adaptive likelihoods to make factored sampling more efficient. Moreover, we model the appearance changes between video frames by exploiting the local linearity of an appearance manifold. A new tangent-plane-selection criterion is proposed, and a reference image is built using the best linear variety on the selected tangent plane. This adaptive appearance model makes image matching robust to various appearances and poses.

The proposed method can be characterized as an online learning paradigm which does not require prior training data. Unlike many tracking algorithms [5], [17], [18] which employ complex dynamical models, we model the state dynamics using the Brownian motion. This makes our method generic. Other characteristics of the proposed method include the following: 1) adaptive factored sampling; 2) adaptive noise; 3) adaptive likelihoods; and 4) an adaptive appearance model. As our appearance model, we consider appearances on a manifold and utilize its geometrical properties accordingly. This adaptive framework is the core contribution of this paper. Four video sequences, namely, dudek [19], davidin300 [19], Rams, and Smiley, are employed to assess the performance of the proposed method.

The rest of this paper is organized as follows. Sections II and III review related works and the elements of condensation algorithms. The frameworks of adaptive sampling and adaptive appearance are given in Sections IV and V, respectively. Experimental results are described in Section VI. Discussion and conclusions are provided in Sections VII and VIII, respectively.

II. Related Work

Appearance changes can be caused by pose and/or illumination variations. Isard and Blake [5] propose a condensation algorithm for visual tracking using active contours parametrized by low-dimensional vectors. The key idea of condensation algorithms is the iterative use of factored sampling to generate multiple hypotheses on video sequences. MacCormick and Isard [20] propose a partitioned sampling for a condensation algorithm which performs the sampling in a hierarchical fashion using a survival rate. Unlike partitioned sampling, our adaptive sampling is a two-stage cascade factored sampling where a coarse sampling is applied to locate the approximated face locations, followed by a fine sampling to fine tune the face positions.

The task of visual tracking can be considered as an optimization problem. Hager and Belhumeur [7] propose a tracking algorithm utilizing a parametric model. This algorithm employs a low-dimensional linear subspace representation of appearance based upon a set of images acquired prior to the start of tracking. A gradient-based optimization procedure is used during tracking to adjust appearance. Cootes and Edwards [21] introduce active appearance models (AAMs) for deformable objects. This model applies principal component analysis to encode shape and texture information. Typically, a Gauss–Newton method is used to find model parameters that adjust the model to match closely the current video frame. The major drawback of this model is its generalization to a generic person; in addition, prior training is needed.

Jepson *et al.* [22] propose a $\mathcal{WSL}$ tracker employing a mixture model to account for appearance changes. The image features are extracted using wavelet filters and are modeled using an online expectation-maximization algorithm during the tracking process. However, the appearance update strategy for these mixture models depends upon a Gaussian assumption which may not be valid for real-world data. Instead, we model observations on an appearance manifold.

Visual tracking can be viewed as a classification problem. Avidan [23] formulates the visual tracking problem as a classification task using support vector machines (SVM), where the target is acted as positive examples and background is regarded as negative examples. Song *et al.* [24] combine ensemble classifiers with particle filters for multitarget visual tracking. These classification-based tracking methods usually require either offline training or heavy computational load due to a complicated online learning scheme. Yu *et al.* [25] integrate a generative tracker, which employs multiple subspaces to represent an object, and a discriminative tracker, which is online SVMs. Subspaces are learned and merged online for all appearance variations; in addition, an online SVM classifier is used to focus on recent appearance changes. This method yields very good tracking results. However, it requires heavy computation for subspace merging and updating SVMs and only obtains two frames per second in a C++ implementation.

Adapting to environmental changes is key for visual tracking. Zhou *et al.* [17] employ a particle filter for tracking and recognition. Moment images are used to build a mixture appearance

IEEE Proof

model for various appearances. The adaptive velocity is computed based on the difference between two successive frames. The adaptive noise is determined based on the quality of prediction. The number of particles is also varied based upon the estimated noise variance. Furthermore, the authors treat occluded pixels as outliers in which the properties of robust statistics are assumed. Contrastingly, we do not assume any statistical properties in order to handle occlusion. Li *et al.* [26] propose a cascade particle filter using three discriminative observers. These discriminative observers are trained using different intervals between video frames. A cascade model is applied for importance sampling. There are three stages of sampling corresponding to these three observers, and they employ 3000, 600, and 200 particles, respectively. In contrast, our cascade factored sampling only needs to apply 400 to 600 particles.

Incrementally updating a subspace is a popular technique for online tracking. Ho *et al.* [14] introduce an online subspace learning method for visual tracking. The bases of a subspace are constructed from a set of local means. The local means are computed from a set of consecutive frames, and a new observation is constrained to remain within a preset distance of its local mean. The appearance subspace is incrementally updated. This method maintains the temporal neighborhood (recent frames) as the bases for tracking. The local constraint may eventually cause the tracker to lose valuable information and consequently fail to keep tracking the face. In contrast, our method keeps track of the spatial neighborhood, and consequently, key frames can be used to cope with drifting and various appearance changes.

Lee and Kriegman [27] utilize a pretrained generic appearance representation in conjunction with an online person-specific appearance model. The authors approximate an appearance manifold using a low-dimensional linear subspace and a probabilistic state-transition Bayesian framework. The pose subspaces are incrementally updated during tracking. Recently, Ross *et al.* [6] applied a particle filter and extended the sequential Karhunrn–Loeve algorithm. This sequential algorithm incrementally updates a mean image and associated eigenspace that characterizes the face being tracked. A forgetting factor is suggested to enhance the robustness of tracking. The authors demonstrate that their method outperforms the $\mathcal{WSL}$ [22] and the mean shift [28] trackers on the dudek video sequence. Nevertheless, this method needs to employ lots of particles to achieve good results.

Generally speaking, drifting is the challenge to be overcome by online visual trackers. Matthews *et al.* [8] discuss the template update problem for drifting. The authors proposed a simple mechanism for the drifting problem, which aligns the update template with the first frame. Then, they formulated the tracking problem as a search problem for AAMs. In this paper, we attack the drifting problem by adapting the tangent plane and not allowing small perturbations to contaminate our appearance model.

III. Condensation Algorithm

To review, the objective of condensation algorithms [5] is to estimate the posterior distribution $p(x_t \,|\, \mathcal{Y}_{1:t})$, where x_t is a state at time t and $\mathcal{Y}$ is the observation from time 1 to time t. By Bayes rule, this posterior distribution can be estimated recursively [4]

$$p(x_t \,|\, \mathcal{Y}_{1:t}) \cong \kappa p(y_t \,|\, x_t) p(x_t \,|\, \mathcal{Y}_{1:t-1}) \tag{1}$$

where κ is a normalizing constant, $p(y_t \,|\, x_t)$ is an observation conditional density, and $p(x_t \,|\, \mathcal{Y}_{1:t-1})$ is a prior density.

In practice, $p(y_t \,|\, x_t)$ is generally multimodal, and $p(x_t \,|\, \mathcal{Y}_{1:t})$ cannot be computed in closed form. However, it is assumed that $p(y_t \,|\, x_t)$ can be evaluated at points, and therefore, the posterior distribution can be approximated using factored sampling. The key to factored sampling is to generate n samples $\{s_1, s_2, \ldots, s_n\}$ from $p(x_t)$, where sample $i \in \{1, 2, \ldots, n\}$ is chosen in accordance with probability π_i described as follows:

$$\pi_i = \frac{p(y_t \,|\, x_t = s_i)}{\sum_{j=1}^{n} p(y_t \,|\, x_t = s_j)}. \tag{2}$$

Once the posterior distribution is approximated, the maximum *a posteriori* estimate can be used to draw probabilistic inferences. Note that increasing n improves the estimate and the approximated posterior distribution is weakly convergent to the true posterior distribution using factored sampling [5].

A. Elements of the Condensation Algorithms

Condensation algorithms iteratively exploit propagation and factored sampling. The posterior distribution is sampled from a set of particles $\{s_t^{(p)}, \pi_t^{(p)}\}_{p=1}^{n}$ with $p(x_{t-1} \,|\, \mathcal{Y}_{1:t-1})$. Samples with high probabilities will be sampled multiple times and processed by propagation and observation steps. Four elements of condensation algorithms may be summarized as follows.

1) Initialization: The first step for a condensation algorithm is to form a set of potential states. Each potential state x_0 has an associated likelihood π_0. Let n be the initial number of particles; then, this set of initial particles can be expressed as $\{s_0^{(p)}, \pi_0^{(p)}\}_{p=1}^{n}$, where the superscript denotes the index of a particle and the subscript denotes the time index.

2) Propagation: Propagate samples using a state-transition equation $s_t^{(p)} = s_{t-1}^{(p)} + N(0, \Sigma)$, where Σ is the observation noise.

3) Observation: Compute likelihoods from observations $\pi_t^{(p)} = \mathrm{p}(y_t^{(p)} \,|\, s_t^{(p)})$, and normalize such that $\sum_{p=1}^{n} \pi_t^{(p)} = 1$.

4) Resampling: Resample $s_t^{(p)}$ with the probability $\pi_t^{(p)}$ using factored sampling.

The propagation, observation, and resampling process will be iterated in each time step.

IV. Adaptive Sampling

The canonical condensation algorithm employs a uniform approach to perform sampling. In other words, the same number of new particles are created upon each iteration. In our algorithm, both the number of new particles and the amount of noise vary in response to temporal differences and match qualities. This process is described as follows, beginning with a formal definition of the state space.

IEEE Proof

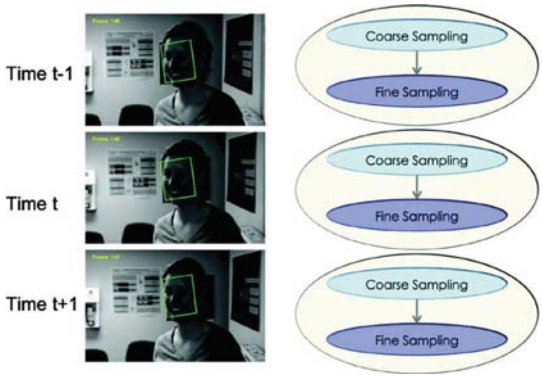

Fig. 2. Illustration of our cascade factored sampling. Every time step consists of a coarse and fine sampling.

A. State Space Model

Face tracking is accomplished by fitting a rectangular bounding box to a region of the image centered upon the face. The position, size, and shape of the box relative to the current frame of video are determined by an affine transformation defined by the following:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & dx \\ \sin(\theta) & \cos(\theta) & dy \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & q & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1+sx & 0 & 0 \\ 0 & 1+sy & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad (3)$$

In this equation, x and y are the 2-D homogeneous input coordinates, and u and v are the affine transformed coordinates. The space of affine transformations is the state space searched by the face tracking algorithm.

As (3) shows, affine transformations can be characterized by six control parameters $[sx, sy, \theta, q, dx, dy]$, where sx is the horizontal scaling, sy is the vertical scaling, θ is a rotated angle, q is a skew parameter, dx is the horizontal translation, and dy is the vertical translation. Hence, our state vector denoted as s_t at time t consists of six parameters of an affine transformation, and the objective of face tracking is to estimate these hidden variables.

B. Adaptive Affine Cascade Factored Sampling

Traditional condensation algorithms [5], [6], [26] employ many particles to sample a state space, making condensation inefficient and impractical for large spaces. To make sampling effective and efficient, we propose a cascade factored sampling with coarse and fine factored sampling strategies shown in Fig. 2. As Fig. 2 shows, our cascade factored sampling consists of two levels of propagation and observation in each time step. The purpose of coarse[1] sampling is to approximately locate the face position; then, the face position is fine tuned during the fine sampling process.

Specifically, the $[sx \quad sy \quad dx \quad dy]$ parameters are coarsely sampled while $[\theta \quad q]$ remains unchanged in order to approximately locate the face. The number of particles used in coarse sampling is between 400 and 600. Particles with high probabilities are selected for fine sampling with all six affine parameters. In our experiments, we select the top 100 particles to perform fine sampling. In the fine sampling step, the noise of $[sx \quad sy \quad dx \quad dy]$ is set to be small constants such that small adjustment is made, and the $[\theta \quad q]$ noise depends upon a match quality estimate, as described in the next section.

[1]For the sake of brevity, here and in future, we will drop the adjective "factored" in front of sampling.

C. Adaptive Noise

Samples are propagated by adding observation noise to states. Unlike traditional condensation algorithms, which add a fixed amount of noise to every image frame, we determine the amount of observation noise based on the temporal difference of each state parameter. Let $u^{(i)}$ be the element of one of the state parameters in $[sx \quad sy \quad dx \quad dy]$, where the superscript indicates the index position. In the coarse sampling step, the amount of observation noise is determined based on the following criterion:

$$\Sigma_u = \max\left(\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}, \begin{bmatrix} b_1 & 0 & 0 & 0 \\ 0 & b_2 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & b_4 \end{bmatrix} \begin{bmatrix} du_t^{(1)} \\ du_t^{(2)} \\ du_t^{(3)} \\ du_t^{(4)} \end{bmatrix} \right), \quad \text{where} \quad \begin{bmatrix} du_t^{(1)} \\ du_t^{(2)} \\ du_t^{(3)} \\ du_t^{(4)} \end{bmatrix} = \begin{bmatrix} \left| u_t^{(1)} - u_{t-1}^{(1)} \right| \\ \left| u_t^{(2)} - u_{t-1}^{(2)} \right| \\ \left| u_t^{(3)} - u_{t-1}^{(3)} \right| \\ \left| u_t^{(4)} - u_{t-1}^{(4)} \right| \end{bmatrix}. \quad (4)$$

In this context, max is an elementwise maximum operation. The term a_i ensures that a minimum amount of noise will always be added to an observation. The term b_i weighs the L_1 norm of the temporal difference. In our experiments, we set a to $[0.03, 0.03, 2, 2]^{\mathrm{T}}$ and $\mathrm{diag}(b)$ to $[0.5, 0.5, 1, 1]^{\mathrm{T}}$.

While coarse sampling is concerned with scale and translation, and uses temporal differences to adjust the amount of noise, fine sampling will add a noise term for rotation and skew based upon the quality of the match. Formally, let v be the element of one of the state parameters in $[\theta \quad q]$. In the fine sampling step, the amount of observation noise is determined as follows:

$$\Sigma_v = \tau \times \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \quad (5)$$

where c_i is determined empirically, and τ is a match quality measure defined in the next section. In our experiments, we set c as $[0.003, 0.003]^{\mathrm{T}}$. In general, the parameters a, b, and c control the range of sampling. The specific settings for a, b, and c just described were chosen based upon pilot experiments conducted using the dudek video sequence [19]. These settings are then held constant in all the tracking experiments presented in Section VI.

Also in a pilot experiment, the importance of adaptive sampling and adaptive noise was tested by running a side-by-side comparison of our algorithm using both and a variant of our algorithm with both disabled. The test was also run on the dudek video sequence. From this comparison, we note that the number

IEEE Proof

of particles had to be increased to 1000 to obtain reasonable tracking performance, whereas the adaptive version of our algorithm uses between 400 and 600 particles. Even with the additional particles, the average root-mean-square error (rmse) increased by 10% relative to the result for our adaptive algorithm. This outcome clearly indicates that the adaptive components of our algorithm not only result in a more efficient tracking algorithm but also improve tracking accuracy.

V. Adaptive Appearance

Every particle has an associated affine transformation, and this is used to warp a video frame to a canonical geometry. The canonical geometry is defined as the face chip from the first frame which is selected manually in our experiments. Formally, let $y_t^{(p)} \in \mathbb{R}^m$ be an observation given by a state $s_t^{(p)}$ at time t, defined as

$$y_t^{(p)} = W\left(I_t, s_t^{(p)}\right) \tag{6}$$

where W is a warp operation,[2] I_t is an image frame at time t, and $s_t^{(p)}$ is the pth state vector at time t.

In general, the likelihood of an observation depends upon the quality of the match between the warped and reference images. In the context of our face tracking, a reference image is a template built from an appearance model, and it is used to match all possible face candidates. Before we discuss how to adaptively build this reference image, we first define our likelihood function and demonstrate our strategy for narrowing the envelope of the likelihood function when a state is found to be a good match. The rationale is to focus the search more finely in the regions of state space where matches are good. Then, we will describe the appearance model and our procedure for adapting the model. The model is based on a local linear approximation to the appearance manifold. The adaptation process is based on a new tangent-plane-selection criterion.

A. Adaptive Likelihood Model

A likelihood function measures the quality of an observation. As is common, the likelihood depends upon the L_2 distance between the observation $y_t^{(p)}$ at time t and the reference image z_{t-1} created from the appearance model at time $t-1$. Specifically, the likelihood is defined as

$$p\left(y_t^{(p)} \mid s_t^{(p)}\right) = \exp\left(-\frac{\left\|z_{t-1} - y_t^{(p)}\right\|^2}{\sigma^2}\right). \tag{7}$$

What makes our likelihood adaptive is our introduction of a dependence between σ^2 and the best match among the current observations through an intermediate variable τ. Specifically, τ represents the smallest L_2 distance between the reference image and the observations at time t

$$\tau = \min_p \left\|z_{t-1} - y_t^{(p)}\right\|^2. \tag{8}$$

[2]In this paper, the warp operation is defined as an affine transformation, (3), followed by a cropping operation.

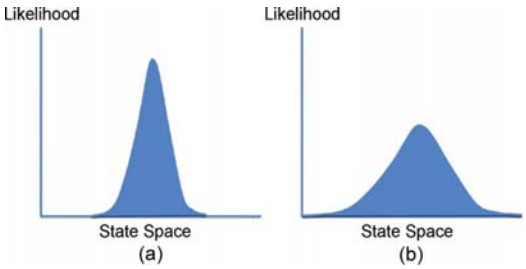

Fig. 3. Essence of adaptive likelihood sampling is adjusting the variance of the noise model. (a) Intensify sampling when confidence in match is high. (b) Diversify sampling when confidence in match is low.

Then, the adaptive variance is defined as

$$\sigma^2 = \max(\tau, T) \times \sigma_0^2. \tag{9}$$

The initial variance σ_0^2 and minimum noise threshold T are user-defined parameters. As (9) shows, the adaptive variance is proportional to the match quality τ, and the match threshold T ensures that at least some minimum amount of variation is always applied to the sampling process.

In condensation algorithms, the purpose of computing likelihood probabilities $p(y_t^{(p)} \mid s_t^{(p)})$ is to govern the factored sampling process where particles are resampled in order to concentrate particles in the regions of the search space with the highest likelihood. The adaptive variance σ^2 intensifies search over a narrow range of affine transformations when a match quality is high and broadens the range of transformations when a match quality is low. This relationship between intensification and diversification is shown in Fig. 3.

We also increase the number of particles by a factor of 1.5 when τ is larger or equal to T, indicating a low match quality. The rationale is that the match quality will degrade if the tracker is starting to fail to keep up with a moving face, and increasing the number of particles increases the chances of finding a better affine transformation and reacquiring the face.

B. Adaptive Appearance Model

The proposed adaptive appearance model is motivated by the topology of a space where images are viewed as residing on an abstract image space [16]. We call this image space as the appearance manifold. One advantage of making the topology of the appearance manifold explicit is that it allows us to take advantage of adjacent appearances in the sense of spatial neighborhood rather than temporal neighborhood. The spatial coherence admits local linearity even though the appearance manifold is globally curved. Since images on an appearance manifold exhibit strong spatial coherence, the appearance model can reconstruct a better reference image. Before showing the benefits of spatial neighborhood, we reveal how the appearance manifold is modeled.

Key operations for our tracking algorithm include forming an appearance manifold, selecting a tangent plane, and building a reference image based upon the tangent plane. These operations are iteratively applied to every frame during tracking such that the appearance is adapted over time. The next three sections describe these three operations.

1) Appearance Manifold Formation: Although frame-to-frame changes in face appearance are usually small, over

IEEE Proof

many frame appearance changes and trackers reliant upon fixed templates will perform poorly. Matthews *et al.* [8] discuss the template update problem by which a tracker may adapt over time. We apply a similar drift correction technique in our appearance manifold update scheme.

Formally, let us define a location-independent state as $\hat{s}_t^{(p)} = [sx, sy, \theta, q]_t^{(p)}$ and the best observation Y_t to be the $y_t^{(p)}$ associated with the particle p that yields the highest likelihood $p(y_t^{(p)} \mid s_t^{(p)})$ at time t. We declare an observation Y_t as an appearance image (*key frame*) and place it on the appearance manifold when the following conditions are satisfied:

$$\tau < T \qquad \epsilon_1 < \min_j \left\| \hat{s}_t^{(p)} - \hat{s}^{(j)} \right\|^2 < \epsilon_2 \tag{10}$$

where $\hat{s}_t^{(p)}$ is associated with Y_t, and the index j ranges over the appearance images previously appended to the appearance manifold. The first condition examines the match quality while the second condition ensures that the pose variability is within a range defined by ϵ_1 and ϵ_2.

Since the first observation y_0 is manually selected, it is automatically appended to the appearance manifold. In addition, we replace the j^*th observation on the appearance manifold when the number of observations exceeds a predefined threshold (40 in our experiments), where j^* is defined as

$$j^* = \operatorname*{argmin}_j \left\| \hat{s}_t^{(p)} - \hat{s}^{(j)} \right\|^2. \tag{11}$$

This update scheme essentially replaces the observation on an appearance manifold that is nearest to the new observation.

2) Tangent-Plane Selection: The distance between an image and a manifold can be characterized using a tangent plane [15]. However, tangent planes are not unique on an appearance manifold. Most existing algorithms employ all available images on an appearance manifold to form a tangent plane. In this paper, we introduce a new criterion for tangent-plane selection.

First, we formally define a tangent plane. Let $\mathcal{F}$ be some unknown transformation acting on appearances. The first-order approximation of an appearance manifold is represented as

$$\mathcal{F}_\alpha = \mathcal{F}_0 + \sum \nabla \mathcal{F} \alpha + H.O.T. \tag{12}$$

where $\mathcal{F}_0$ is an appearance image on an appearance manifold, $\nabla \mathcal{F}$ denotes the tangent vectors, and $\mathcal{F}_\alpha$ is the reconstructed appearance image. The purpose of tangent distance is to seek a reconstructed image using tangent vectors such that the distance between the reconstructed image and an observed image Y_t is minimized. In this context, Y_t is the observation $y_t^{(p*)}$ having the highest likelihood in (7). Mathematically, this may be expressed as

$$\min_\alpha \left\| \mathcal{F}_0 + \sum \nabla \mathcal{F} \alpha - Y_t \right\|^2. \tag{13}$$

In the context of face tracking, there are many alternative ways to select $\mathcal{F}_0$ and $\nabla \mathcal{F}$ given an observation Y_t, and these alternatives play an important role in building a reference image.

To explore three possible alternatives, a simplified illustration is helpful. Fig. 4 shows an observation Y_t in relation to nine appearance images on the appearance manifold. These appearance images are denoted as $g, g_1, \ldots, g_8$. Furthermore, in this example, we assume that three tangent vectors are sufficient to approximate a tangent plane.

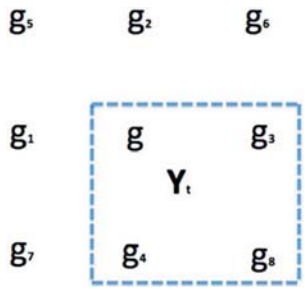

Fig. 4. Example of tangent-plane selection, where Y_t is an observation and g_k is the appearance on a manifold.

1) Set Y_t to be equal to $\mathcal{F}_0$, and select the three nearest neighbors (g, g_3, and g_4) of Y_t to form the tangent vectors. In this case, there is a trivial solution ($\alpha = 0$) that makes Y_t the reference image. This is known as naive update [8] and has been shown to drift easily.
2) Set g to be equal to $\mathcal{F}_0$ and three of its nearest neighbors (g_1, g_3, and g_4) to form the tangent vectors. As Fig. 4 shows, these appearance images may not be the best images to reconstruct Y_t. We call this *direct appearance update*.
3) Set g to be equal to $\mathcal{F}_0$ and three nearest neighbors (g_3, g_4, and g_8) of Y_t to form the tangent vectors around g. As Fig. 4 shows, the nearest images of Y_t are surrounded in a blue box. These appearance images typically provide a much better reconstruction. We call this *adaptive appearance update*.

Based on these observations, we choose the adaptive appearance update as our tangent-plane-selection criterion. This criterion provides better spatial coherence of images on an appearance manifold. Our experimental results conclude that the proposed adaptive appearance framework is resilient to drifting. Further analysis is given in Section VII-B.

3) Reference Image Update: To update the reference image, we first select the base image g on the appearance manifold, which is the nearest observation of Y_t shown as follows:

$$g = \operatorname*{argmin}_j \|Y_t - g_j\|^2 \tag{14}$$

where g_j is an appearance image on the appearance manifold. Because local linearity is a reasonable presumption for the appearance manifold, we can form a tangent plane using a set of k nearest neighbors expressed as

$$T_g\mathcal{M} : g + \text{span}\{g - g_1, g - g_2, \ldots, g - g_k\} \tag{15}$$

where $\{g_1, g_2, \ldots, g_k\}$ are the nearest k neighbors of Y_t, and $\{g - g_1, g - g_2, \ldots, g - g_k\}$ are the tangent vectors around g. In our experiment, we set k to be equal to 15. In the case where the number of images on the appearance manifold is less than k, we use all the images that are currently available on the appearance manifold.

As discussed in the previous section, tangent-plane selection plays an important role in building a reference image. We should note that the reference image resides on a tangent plane $T_g\mathcal{M}$. More specifically, the reference image is formed by

IEEE Proof

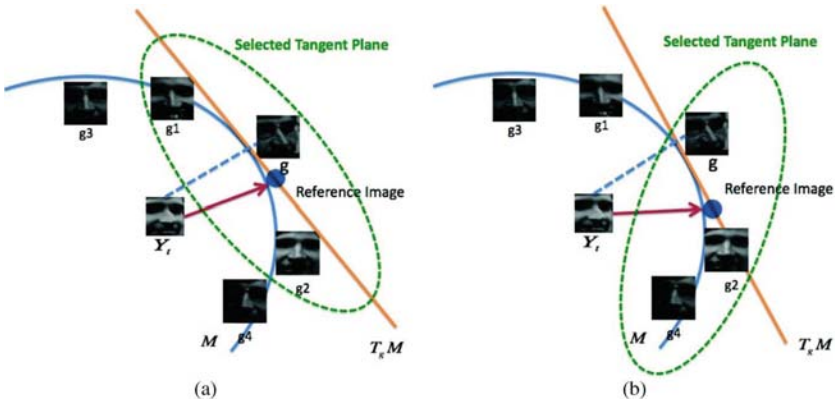

Fig. 5. Direct appearance update, the adaptive appearance update, and an appearance manifold. The blue dot represents a reference image residing on a tangent plane. Different appearance updates select different sets of tangent bases; different tangent bases form a different tangent plane and, therefore, a different reference image. (a) Direct update. (b) Adaptive update.

seeking the best linear combination of the tangent vectors at g relative to the observation Y_t. Thus, the k nearest neighbors (a dashed ellipse in Fig. 5) govern the quality of the reference image. Fig. 5 shows how the reference image changes when we apply alternative tangent-plane-selection criteria.

Fig. 5(a) shows the direct appearance update and further shows how the set of basis images may not be spatially closest to the observation Y_t. In this case, the reference image may not be coherent relative to the topology of the appearance manifold. The adaptive appearance update shown in Fig. 5(b) considers the spatial arrangement on the appearance manifold and builds the reference image using a set of nearest bases. This approach, which respects the topology of the appearance manifold, is much more likely to create a reference image from a coherent choice of basis images.

Recall that a linear variety allows a hyperplane to shift from its origin. In this paper, we further extend the linear combination to the best linear variety such that the tangent plane would be shifted closer to the observation. In other words, changes in the overall image brightness are trivially accommodated. This extension can be written in a matrix form as

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_m \end{bmatrix} + \begin{bmatrix} 1 & V_1^{(1)} \dots V_1^{(k)} \\ 1 & V_2^{(1)} \dots V_2^{(k)} \\ \vdots & \vdots \\ 1 & V_m^{(1)} \dots V_m^{(k)} \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} \tag{16}$$

where $V^{(i)}$ denotes $\{g - g_i \,|\, i = 1, \dots, k\}$, and the additional column "1" is augmented for brightness compensation; thus, any linear variety can be reconstructed with a suitable α. Now, the goal is to seek the best α^* such that the difference between the linear variety and an observation Y_t is minimized as follows:

$$\min_{\alpha} \| g + V\alpha - Y_t \|^2. \tag{17}$$

Taking a derivative with respect to α, the solution becomes

$$\alpha^* = V^{-1}(Y_t - g). \tag{18}$$

Substituting α^* back to (16), the reconstructed appearance image z_t at time t is the best linear variety against Y_t and will be used as a reference image for computing likelihoods at time $t + 1$.

To summarize our adaptive condensation algorithm, we sketch our method in Algorithm 1. As Algorithm 1 reveals, our method exhibits four major elements. They are propagation and observation, tangent-plane selection, reference image update, and appearance manifold update. As such, the propagation and observation step can be considered as an adaptive sampling process, while the tangent-plane selection, reference image update, and appearance manifold update can be regarded as an adaptive appearance framework. Both elements are interdependent and key ingredients for our face tracker.

Algorithm 1 Our Adaptive Condensation Algorithm
1: Initialize particles
2: **while** frame $\neq$ empty **do**
3: Apply propagation and observation
4: Coarse factored sampling
5: Compute likelihoods (7)
6: Normalize
7: Select top particles
8: Fine factored sampling
9: Compute likelihoods (7)
10: Normalize
11: Select a tangent plane
12: Update a reference image
13: Solve α (18)
14: Build a reference image z (16)
15: Update an appearance manifold
16: **end while**

VI. Experimental Results

We demonstrate the effectiveness and robustness of our adaptive condensation algorithm using four challenging video sequences. These video sequences are acquired indoors using hand-held video cameras so that both camera motion and human motion are concurrently revealed. The dudek video sequence [19] is a well-known benchmarked video for face tracking and comes with hand-labeled ground-truth positions. The second video sequence, davidin300 [19], has mixed shadowing

IEEE PROOF

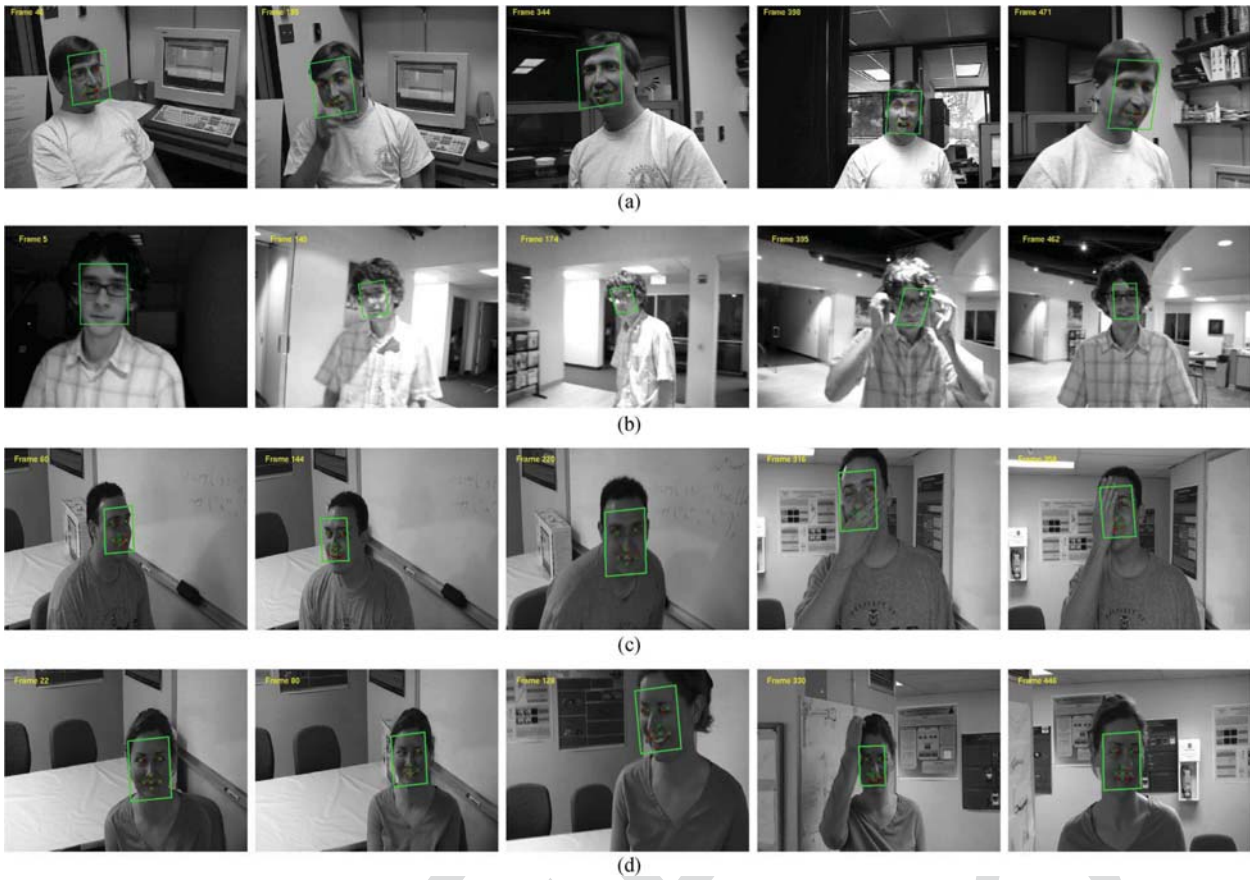

Fig. 6. Demonstration of robustness to illumination, appearance and pose variations, and occlusions. (a) Dudek video sequence. (b) Davidin300 video sequence. (c) Rams video sequence. (d) Smiley face video sequence.

Fig. 7. Examples of occlusion recovery.

and pose variations. The third and fourth video sequences were collected at Colorado State University,[3] namely, the Rams and Smiley video sequences. These video sequences also exhibit various appearances, poses, and occlusions. Example frames from these videos are shown in Fig. 6. In our experiments, the initial tracking position is manually selected. The green bounding boxes shown in Fig. 6 indicate where our tracker has found the face in these frames.

[3]Available at http://www.cs.colostate.edu/ vision/SMC-FaceTracking/.

In the next two sections, we will discuss specific aspects of performance as it relates to illumination, appearance, pose, and occlusion. This highlights how aspects of our algorithm relate to these specific challenges. Then, in Section VI-C, we present our quantitative evaluation.

A. *Illumination, Appearances, and Poses*

Lighting, appearance, and pose variations are the primary challenges in face tracking. It is imperative for a robust

algorithm to handle such variations. The sample tracking results shown in Fig. 6(a)–(d) show that our method is successfully tracking the human face under variations in all three of these factors. In addition, the video sequences include additional confounding factors such as pronounced facial shadowing, occlusion of the face, and changes in facial expression and pose.

Our appearance model is robust to lighting, appearance, and pose changes because it is adaptive. This is accomplished by rebuilding the reference image in every frame. To be specific, our tangent-plane selection is driven by the observation (see Section V-B2) that the lighting, appearance, and pose from the observation are considered when we select the tangent vectors. To put this another way, our reference image is always created from a set of prior images that are most similar to the current image, even when that means it is best to step over the recently acquired images in favor of images seen less recently but exhibiting characteristics that are more useful for explaining the appearance of the current frame. Finally, from a practical standpoint, it is also important that our method compensates for the overall changes in brightness by using a linear variety that enables the tangent plane to shift away from the origin.

B. Occlusion

Focusing in more precisely on occlusion, Fig. 7 shows successive frames of a video sequence in which our algorithm is successfully handing occlusion. In Fig. 7, the white bounding boxes are sampling positions, and the green bounding box is the face position with the highest posterior probability. Observing in the first few frames, where the face is fully visible, the number of sampling regions is small. As the hand moves in front of the chin in the middle frame of the top row, the expansion of the sampling pattern is immediately evident. Then, in successive frames where the face is partially occluded, the sampling pattern continues to expand. This adaptive sampling allows the algorithm to react and avoid losing a face even when the face is moving and/or temporarily occluded.

There are two primary reasons why our method is robust to temporary occlusion. First, our adaptive sampling strategy is able to follow the movement of the face since we are using the temporal differences to adjust the amount of noise. Second, because only observations with a quality score above a threshold described in (10) are added to the model, occluded objects will not typically participate in the process of building a reference image, and consequently, they do not corrupt the appearance model.

Of course, these strategies have their limits. The longer a face is occluded, the higher the risk that the tracker will expand its search region to such an extent that it will lose the object. It is also conceivable, although we have not observed such a case, that very slow introduction of ever more occluded images of a face might allow partially covered instances of the face to be added to the reference model.

C. Quantitative Evaluation

Ground-truth face location information is available for three video sequences: dudek, Rams, and Smiley. For these, we are able to carry out a quantitative evaluation. Each video frame in these sequences is annotated with seven fiducial positions. We employ the rmse between the ground-truth and estimated positions to assess tracking performance. An example of ground-truth and estimated positions is shown in Fig. 8, where the red crosses are ground-truth positions and the green crosses are the estimated positions.

Fig. 8. Ground-truth and estimated positions, where ground-truth positions are red and estimated positions are green.

We directly compare our tracking results with the state-of-the-art tracking algorithms that exploit the advantages of a learned subspace method [14] and incremental visual tracking (IVT) [6]. Both these methods are online subspace learning techniques where the learned subspace method constructs a set of local means as the bases of a subspace and the IVT learns the subspace incrementally with a forgetting factor. The quantitative results for the dudek, Rams, and Smiley video sequences are shown in Fig. 9 and summarized in Table I.

As can be seen in Fig. 9(a), all three algorithms successfully track the face over the full duration of the dudek video. This is not true for the Rams video [Fig. 9(b)] or the Smiley video [Fig. 9(c)], where distinct points in the video can be observed where an algorithm experiences a substantial jump in rmse and then never recovers. For example, in the Rams video [Fig. 9(b)], a distinct jump in the rmse for the IVT algorithm can be seen at frame 99. A similar jump can be seen for the learned subspace algorithm at frame 532. Notably, our method maintains a relatively stable, and comparatively low, rmse over the entire duration of the video for both the Rams and Smiley video sequences.

Average rmse values over the entire dudek video for all three algorithms are presented in the first three rows of Table I. The average rmse values for the learned subspace method and IVT are 6.3 and 5.32,[4] respectively, whereas our method achieves 4.98 average rmse. More importantly, our method only employs 400 to 600 particles and runs at 8.74 frames per second, whereas the IVT uses 4000 particles and runs at 1.1 frames per second. Note that all experiments are run on the same machine, and so, this difference in time required per frame is a reliable indication of just how much more efficient is the adaptive framework.

Since both the IVT and learned subspace algorithms lose track of the face at some point in both the Rams and Smiley video sequences, Table I presents the average rmse values in two ways. First, the average rmse values are presented over the full video sequences, and as a result of the algorithms losing track of the face, the average rmse is very high for both algorithms on both videos. To address the question of how

[4]The MATLAB source code was downloaded from http://www.cs.toronto.edu/~dross/ivt/, and we used the best parameter settings suggested by the authors.

IEEE Proof

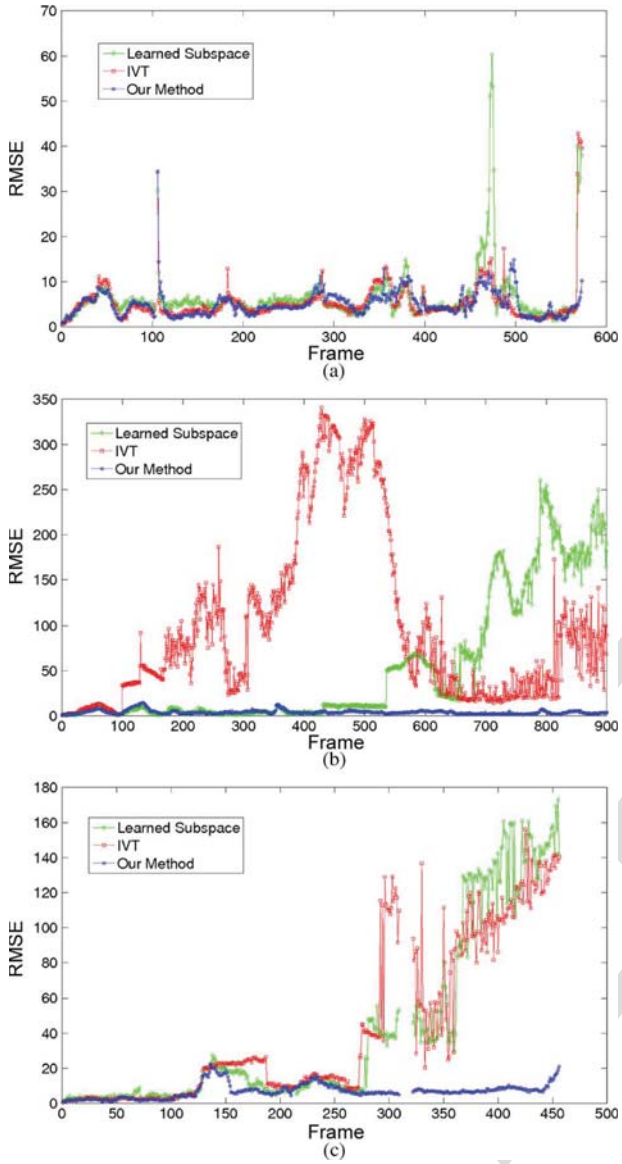

Fig. 9. Quantitative comparisons to learned subspace [14] and IVT [6] for face tracking. (a) Dudek video sequence. (b) Rams video sequence. (c) Smiley video sequence.

well the algorithms are doing relative to each other over the portion of the video where all three algorithms remain intact in tracking, Table I also shows the average rmse over just the first 99 and 273 frames of the Rams and Smiley videos, respectively. Even on these truncated video sequences, our method still outperforms the learned subspace method and IVT in both Rams and Smiley video sequences, where our method achieves 3.51 and 6.43 average rmse values for the Rams and Smiley video sequences, respectively.

The key differences between our method and the subspace methods tested in this paper are the data representation of the appearance model and the update strategy. While both subspace methods build a subspace incrementally, we use key frames and represent them on an appearance manifold. As we consider key frames on an appearance manifold, the geometric interpretation is taken into account. We apply the property of local linearity and make a local update for a tangent plane instead of an incremental update for a subspace like [6] and [14]. As such, our bases are a set of local[5] (similar) key frames rather than global bases. This spatial coherency allows us to select a tangent plane from adjacent appearances and builds a better reference image. In contrast, the subspace methods such as [6] and [14] adopt all the bases to form a subspace and compute the subspace distance for each observation. Therefore, these methods ignore the spatial coherency which plays an important role in building a good reference image.

Finally, we note that all the tracking results are obtained using MATLAB version 7.6.0.324 on an HP-xw4600-Core2Duo-SATA 2x2.83G 64-b machine. The tracking results, tracking videos, and the landmark points can be downloaded from http://www.cs.colostate.edu/~vision/SMC-FaceTracking/.

VII. Discussion

A. Drift Analysis

Template update is an essential step to account for various expression, pose, and illumination changes. However, it can easily result in a tracker drifting off the object of interest, in this case, a particular person's face. To combat drifting, we need to first understand that drift is caused by accumulated error in a template position. Whenever the template is updated, small errors accumulate to the position of the template. Once drifting starts, it can rapidly cause a tracker to lose the face.

The best way to minimize drift is to avoid the small errors in the first place, and this, in turn, requires that the adaptive model makes the correct balance between generality and specificity. Controlling the observations, images, which are added to the appearance manifold, and adaptation to the current frame when constructing the reference image together help our approach strike this balance.

Our approach is selective about what observations are added to the appearance manifold. As defined in (10), there are three aspects to the screening of images that are important. First, there must be a good match between the observation and the reference image. Second, if the change in position, which is the magnitude of the transformation between frames, is very large, the observation is not added to the appearance manifold. These steps ensure the integrity of the appearance manifold. Third, only observations with a sufficient change in position are added to the appearance manifold. This protects against the appearance manifold becoming overwhelmed with a set of nearly identical images.

The steps just described do a lot to balance generality and specificity in the construction of the appearance manifold. However, that alone is not enough. Only observations related to the current frame should contribute to the construction of the reference image, and our algorithm accomplishes this through the tangent-plane construction process described in Section V-B2. Through this mechanism, our algorithm can maintain history about different aspects of appearance, for example, partial profile versus head on, without making the

[5] *Local* implies a spatial relationship rather than a temporal relationship.

PROOF

TABLE I
QUANTITATIVE COMPARISONS IN TERMS OF RMSE AND SPEED

Method	RMSE	Speed (fps)	Number of Particles	Number of Frames	Sequence
Learned Subspace [15]	6.30	8.27	600	573	dudek
IVT [7]	5.32	1.10	4000	573	dudek
Our Method	4.98	8.74	400 $\sim$ 600	573	dudek
Learned Subspace [15]	52.17	1.08	4000	900	Rams
IVT [7]	98.71	0.89	4000	900	Rams
Our Method	3.82	11.27	400 $\sim$ 600	900	Rams
Learned Subspace [15]	40.71	0.89	4000	456	Smiley
IVT [7]	41.77	0.87	4000	456	Smiley
Our Method	6.83	6.63	400 $\sim$ 600	456	Smiley
Learned Subspace [15]	4.52	-	4000	99	Rams
IVT [7]	6.24	-	4000	99	Rams
Our Method	3.51	-	400 $\sim$ 600	99	Rams
Learned Subspace [15]	8.31	-	4000	273	Smiley
IVT [7]	10.13	-	4000	273	Smiley
Our Method	6.43	-	400 $\sim$ 600	273	Smiley

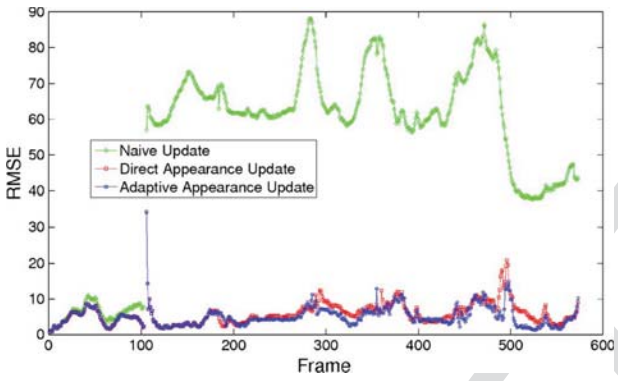

Fig. 10. Quantitative comparison for various appearance update methods using the dudek video sequence.

mistake of combining these two cases when matching to a specific frame.

B. Update Strategy Analysis

The selection of tangent plane plays a vital role in building a reference image. To further analyze the effectiveness of our adaptive appearance model, we compare the adaptive appearance update with the direct and naive appearance updates discussed in Section V-B2. Fig. 10 shows the quantitative results for various update strategies using the dudek video. The naive appearance update loses track after the first occlusion.

The direct and adaptive appearance updates perform identically in the beginning since, initially, the same images are added to the appearance manifold for both algorithms. It is only after the first 174 frames that the performance diverges and our adaptive appearance model can be seen to perform somewhat better. Over the entire video sequence, the average rmse is 5.8 for the direct appearance update and 4.98 for the adaptive appearance update. The argument of these two updates is from various spatial considerations between the observation and the appearances on a manifold. The experimental results described in Fig. 10 shows the importance of spatial coherency. Our adaptive appearance update considers the spatial arrangement properly, and therefore, it is more robust to handle variability and drifting.

C. Limitation

While our algorithm performs well in face tracking, the state model is still an affine transformation realized as a global alignment. Adaptively building a reference image to best model the current frame does go some ways toward handling nonlinear aspects of facial movement; however, it still cannot account for all nonridge facial expressions. Affine transformations do not have the expressive power to model the details of various facial expressions, for instance, smiling or neutral expression.

One approach to handling nonlinear factors, such as expression variations, would be to adopt more sophisticated appearance models. For example, AAMs [21] might be used. However, the dimensions of the state space would be tremendously increased since it encodes both shape and texture information. The increased expressive power of AAMs clearly comes at considerable extra computational expense.

Another factor to consider in reviewing the quantitative results presented is that the ground-truth positions are manually annotated; thus, there may be some noise in these positions. In fact, Jepson *et al.* [22] report that the lower bound of the average rmse for the dudek video is 3.1 pixel error. This baseline error has both ground-truth and fiducial-position errors that cannot be recovered by similarity transformations. Here too, AAMs could break out of the limits imposed by the assumption of a global affine transformation but, again, at a considerable additional computational expense.

Finally, it should be understood that, using our method or, indeed, any of the algorithms against which we compared our method, face tracking is limited to a single face and reacquisition is needed when the object is lost. Our experimental results show that our method works reasonably well on short-term occlusion where the lengths of occlusion are 6, 20, and 22 frames for the dudek, Rams, and Smiley video sequences, respectively. However, tracking failure would definitely occur when long time occlusion is present or the facial image appears significantly different after occlusion. This is because the assumption of smooth transitions between frames is invalid.

VIII. CONCLUSION AND FUTURE WORK

This paper has presented an adaptive framework for human-face tracking. Our adaptive affine cascade strategy performs two-stage factored sampling that effectively reduces the search

space. In our test videos, we employ 400 to 600 particles for coarse factored sampling and 100 particles for fine factored sampling. This factored sampling strategy uses fewer particles and makes sampling more efficient. Equally important, we have adaptively built a reference image in every frame. This is achieved by properly selecting a tangent plane and using a linear variety from an appearance manifold. With these two key ingredients, our method is robust to variation of illumination, appearance, pose, and temporary occlusions. Furthermore, three video sequences with ground-truth positions have been adopted to assess the performance of our method, and encouraging results have been obtained. Finally, since our method is an online learning paradigm, no prior training is required. Human-face tracking is still a difficult task due, in part, to large freedom of face movement and appearance changes. Our future work will focus on temporary disappearance and continue to explore appearance manifolds.

ACKNOWLEDGMENT

The authors thank David A. Ross from University of Toronto for the permission to use his videos and source code.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM J. Comput. Surv.*, vol. 38, no. 4, pp. 1–45, 2006.
[2] N. Gordon, "Bayesian methods for tracking," Ph.D. dissertation, Univ. London, London, U.K., 1993.
[3] N. Gordon, D. Salmond, and A. Smith, "A novel approach to non-linear and non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng.—F*, vol. 140, no. 2, pp. 107–113, Apr. 1993.
[4] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.
[5] M. Isard and A. Blake, "Condensation—Condition density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, Aug. 1998.
[6] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, no. 1–3, pp. 125–141, May 2008.
[7] G. Hager and P. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 10, pp. 1025–1039, Oct. 1998.
[8] I. Matthews, R. Ishikawa, and S. Baker, "The template update problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 810–815, Jun. 2004.
[9] M. J. Black and A. D. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," in *Proc. ECCV*, 1996, pp. 329–342.
[10] S. Von Duhn, L. Yin, M. J. Ko, and T. Hung, "Multiple-view face tracking for modeling and analysis based on non-cooperative video imagery," in *Proc. CVPR*, 2007, pp. 1–8.
[11] K.-C. Lee and D. Kriegman, "Face tracking in video with hybrid of Lucas–Kanade and condensation algorithm," in *Proc. Int. Conf. Multimedia Expo*, 2003, pp. 293–296.
[12] W. Zhang, Q. Wang, and X. Tang, "Real time feature based 3-D deformable face tracking," in *Proc. ECCV*, 2008, pp. 720–732.
[13] A. Levy and M. Lindenbaum, "Sequential Karhunen–Loeve basis extraction and its application to images," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1371–1374, Aug. 2000.
[14] J. Ho, K.-C. Lee, M.-H. Yang, and D. Kriegman, "Visual tracking using learned linear subspace," in *Proc. CVPR*, 2004, pp. 782–789.
[15] P. Simard, Y. L. Cun, and J. Denker, "Efficient pattern recognition using a new transformation distance," in *Proc. NIPS*, 1992, pp. 50–58.
[16] H. Seung and D. Lee, "The manifold ways of perception," *Science*, vol. 290, no. 5500, pp. 2268–2269, Dec. 2000.
[17] S. Zhou, R. Chellapa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1491–1506, Nov. 2004.
[18] B. North, A. B. M. Isard, and J. Rittscher, "Learning and classification of complex dynamics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 9, pp. 1016–1034, Sep. 2000.
[19] D. Ross. [Online]. Available: http://www.cs.toronto.edu/~dross/ivt/
[20] J. MacCormick and M. Isard, "Partitioned sampling, articulated objects, and interface-quality hand tracking," in *Proc. ECCV*, 2000, pp. 3–19.
[21] T. Cootes and G. Edwards, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.
[22] A. D. Jepson, D. J. Fleet, and T. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1296–1311, Oct. 2003.
[23] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.
[24] X. Song, J. Cui, H. Zha, and H. Zhao, "Vision-based multiple interacting targets tracking via on-line supervised learning," in *Proc. ECCV*, 2008, pp. 642–655.
[25] Q. Yu, T. Dinh, and G. Medioni, "Online tracking and reacquisition using co-trained generative and discriminative trackers," in *Proc. ECCV*, 2008, pp. 678–691.
[26] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different lifespans," in *Proc. CVPR*, 2007, pp. 1–8.
[27] K.-C. Lee and D. Kriegman, "Online learning of probabilistic appearance manifolds for video-based recognition and tracking," in *Proc. CVPR*, 2005, pp. 852–859.
[28] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

Yui Man Lui (S'07) is currently working toward the Ph.D. degree in the Department of Computer Science, Colorado State University, Fort Collins.

His current research interests include special manifolds, face recognition, face tracking, and action classification.

Mr. Lui is the recipient of the Honeywell Best Student Paper Award from the 2009 IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS); the coauthor of the Best Paper Award from the 2008 IEEE International Conference on Automatic Face and Gesture Recognition; and the recipient of an honorable mention for the Honeywell Best Student Paper Award from BTAS in 2007.

J. Ross Beveridge (M'83) received the Ph.D. degree from the University of Massachusetts, Amherst, in 1993.

He is currently an Associate Professor with the Department of Computer Science, Colorado State University, Fort Collins. He has served as an Associate Editor for *Pattern Recognition* and is currently an Associate Editor for *Image and Vision Computing*.

Dr. Beveridge has served as an Associate Editor for the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE. He was the Program Cochair for the 1999 IEEE Conference on Computer Vision and Pattern Recognition. He was the recipient of the Outstanding Reviewer Awards in 2007 and 2008 and the Best Paper Award from the IEEE International Conference on Automatic Face and Gesture Recognition in 2008.

L. Darrell Whitley received the Ph.D. degree from Southern Illinois University in 1985.

He is currently the Chair with the Department of Computer Science, Colorado State University, Fort Collins. He serves on the editorial board of the journals *Artificial Intelligence* and *Evolutionary Computation*. He also serves as the Chair of the Association for Computing Machinery Special Interest Group on Genetic and Evolutionary Computation, Sigevo.

IEEE Proof