Visual Tracking via Geometric Particle Filtering on the Affine Group with Optimal Importance Functions

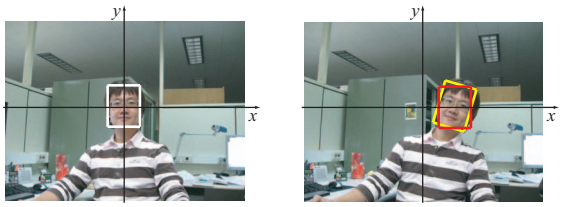
Junghyun Kwon
Department of EECS, ASRI
Seoul National University
Seoul 151-742, Korea
junghyunkwon@gmail.com

Kyoung Mu Lee
Department of EECS, ASRI
Seoul National University
Seoul 151-742, Korea
kyoungmu@snu.ac.kr

Frank C. Park
School of MAE
Seoul National University
Seoul 151-742, Korea
fcp@snu.ac.kr

Abstract

We propose a geometric method for visual tracking, in which the 2-D affine motion of a given object template is estimated in a video sequence by means of coordinate-invariant particle filtering on the 2-D affine group Aff(2). Tracking performance is further enhanced through a geometrically defined optimal importance function, obtained explicitly via Taylor expansion of a principal component analysis based measurement function on Aff(2). The efficiency of our approach to tracking is demonstrated via comparative experiments.



(a) $X_0 = I$ (b) $X_k = \begin{bmatrix} G & t \\ 0 & 1 \end{bmatrix}$

Figure 1. (a) The rectangle represents the object template given in the initial frame. (b) 2-D affine motion tracking (yellow) vs 2-D translation tracking (red). In this paper, the 2-D affine motion tracking is formulated as the filtering on the affine group $Aff(2)$.

1. Introduction

Visual tracking is one of the most fundamental tasks required for advanced vision-based applications such as visual surveillance and vision-based human-robot interaction. In this paper we address the problem of tracking a given object template such as the one shown in Figure 1(a). Tracking such object templates using only 2-D translations is generally difficult, since the object image typically undergoes (under mild assumptions) a 2-D affine transformation. The aim of this paper is to propose a novel geometric framework that can efficiently track such 2-D affine motions of the object image.

Following the seminal work of Isard and Blake [6], various kinds of visual tracking problems have been addressed using particle filtering, which is a Monte Carlo method for general nonlinear filtering problems (see, *e.g.*, [25] and the references cited therein). The common approach to particle filtering-based affine motion tracking is to represent the affine transformation in vector form using a set of local coordinates, and to make use of conventional particle filtering algorithms formulated on vector spaces [12, 16, 26].

It is, however, well-known that the set of affine transformations is not a vector space, but rather a curved space possessing the structure of a Lie group (the affine group). Choosing a set of local coordinates and applying existing vector space methods will more often than not produce results that depend on the choice of local coordinates. More fundamentally, the performance of such local coordinate-based approaches depends to a large extent on whether the geometry of the underlying space is properly taken into account, especially around the extremes of the operating regime.

Given these considerations, we regard the 2-D affine motion tracking as a filtering problem on $Aff(2)$. The approach that we set forth here is based on [10], in which visual tracking is realized via particle filtering on $Aff(2)$ with the geometrically well-defined state equation on $Aff(2)$ and other related geometric necessities. (See [3, 9, 20, 21] for particle filtering on other Lie groups, *e.g.*, the special orthogonal group $SO(3)$ and special Euclidean group $SE(3)$.)

One of the crucial factors in the performance of particle filtering is how to choose the importance function, from which particles are randomly sampled [4]. There have been several attempts to approximate the optimal importance function as closely as possible instead of importance sampling from the state prediction density, because most of the particles sampled from the state prediction density

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

are wasted especially when the state dynamics model is inaccurate and the measurement likelihood is highly peaked [4, 14, 22]. The need for an optimal importance function for particle filtering-based visual tracking is quite evident: the object can make an abrupt movement not predictable via a general smooth motion model, and many robust appearance models are currently available.

Our paper's main contribution is to derive such an optimal importance function for particle filtering on *Aff*(2). We approximate the optimal importance function following the normal distribution approach outlined in [4], in which the normal distribution is determined via first-order Taylor expansion of a nonlinear measurement function with respect to the state. For our purposes we utilize the exponential map to formulate an approximate normal distribution on *Aff*(2) and realize the Taylor expansion of a measurement function whose input argument is *Aff*(2). We then show how the Jacobian of a principal component analysis (PCA) based measurement function can be analytically derived via a simple application of the chain rule.

There is much literature on particle filtering-based visual trackers to approximate the optimal importance function via local linearization of a measurement function [11, 17, 19]. The difference of these from our approach is that the unscented transformation (UT), which is generally recognized as more efficient and accurate than the first-order Taylor expansion [7], is used as a means of local linearization. In this paper, we rely on the Taylor expansion instead of UT for the following reasons. First, UT has tuning parameters whose values must be determined appropriately depending on the application. We believe that it is difficult to find a systematic way to choose such parameter values appropriately for our case, where the measurement function is highly nonlinear with respect to the state and the state itself is also a curved space. Moreover, UT involves repeated trials of the measurement process, *e.g.*, at least thirteen times for our case where the dimension of the affine state is six. Therefore such repeated trials would eliminate the advantage of UT in the case of a complex measurement function.

The remainder of the paper is organized as follows. In Section 2, the visual tracking framework via particle filtering on *Aff*(2) proposed in [10] is briefly reviewed, with an emphasis on the geometric requirements to perform particle filtering on *Aff*(2). In Section 3, the optimal importance function is derived via analytic first-order Taylor expansion on *Aff*(2) of the PCA-based measurement function. In Section 4, we present experimental results demonstrating the feasibility of our proposed visual tracking framework, while Section 5 concludes with a summary.

2. Visual tracking on the affine group

As aforementioned, we deal with the problem of tracking the 2-D affine motion of an object template as shown in Figure 1. The 2-D affine transformation of the object template coordinates is realized via multiplication in the homogeneous coordinates with a matrix $\left[\begin{smallmatrix} G & t \\ 0 & 1 \end{smallmatrix}\right]$, where G is an invertible 2×2 real matrix and t is a $\Re^2$ translation vector. The matrix $\left[\begin{smallmatrix} G & t \\ 0 & 1 \end{smallmatrix}\right]$ can be identified as a matrix Lie group and is called the 2-D affine group (*Aff*(2)). In this section, we briefly review the visual tracking framework of [10], which is primarily based on particle filtering on *Aff*(2).

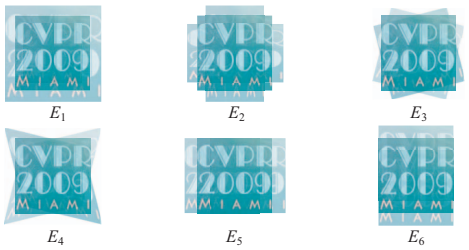


Figure 2. The geometric transformation modes induced by basis elements E_i of *aff*(2). The general affine transformation comes from a combination of these transformation modes.

2.1. Particle filtering on the affine group

A Lie group $\mathbf{G}$ is a group which is a differentiable manifold with smooth product and inverse group operations. The Lie algebra $\mathbf{g}$ associated with $\mathbf{G}$ is identified as a tangent vector space at the identity element of $\mathbf{G}$. A Lie group $\mathbf{G}$ and its Lie algebra $\mathbf{g}$ can be related via the exponential map, $\exp : \mathbf{g} \rightarrow \mathbf{G}$. The 2-D affine group *Aff*(2), that is the semi-direct product of *GL*(2) (2×2 invertible real matrices) and $\Re^2$, is associated with its Lie algebra *aff*(2) represented as $\left[\begin{smallmatrix} U & v \\ 0 & 0 \end{smallmatrix}\right]$ where $U \in gl(2)$ (*gl*(2), which is the space of real 2×2 matrices, denotes the Lie algebra of *GL*(2)) and $v \in \Re^2$. A detailed description of Lie groups can be found in, e.g., [5].

State equation on the affine group. The geometrically well-defined state equation on *Aff*(2) for a left-invariant system can be expressed as

$$dX \quad = \quad X \cdot A(X)\, dt + X \sum_{i=1}^{6} b_i(X) E_i\, dw_i, \qquad (1)$$

where $X \in$ *Aff*(2) is the state, the maps $A : $ *Aff*(2) $\rightarrow$ *aff*(2) and $b_i : $ *Aff*(2) $\rightarrow \Re$ are possibly nonlinear, $dw_i \in \Re$ denote the Wiener process noise, and E_i are the basis elements of *aff*(2) chosen as

$$E_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, E_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, E_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$E_4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, E_5 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, E_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}. \qquad (2)$$

Each geometric transformation mode corresponds to each E_i as shown in Figure 2.

The continuous state equation (1) is usually discretized via the first-order exponential Euler discretization as

$$X_k = X_{k-1} \cdot \exp\left(A(X,t)\Delta t + dW_k\sqrt{\Delta t}\right), \quad (3)$$

where dW_k represents the Wiener process noise on *aff*(2) with a covariance $P \in \Re^{6\times 6}$, *i.e.*, $dW_k = \sum_{i=1}^{6} \epsilon_{k,i} E_i$ with a six-dimensional Gaussian noise $\epsilon_k = (\epsilon_{k,1}, \ldots, \epsilon_{k,6})^\top$ sampled from $N(0, P)$. Then the measurement equation can also be expressed in the discrete setting as

$$y_k = g(X_k) + n_k, \quad (4)$$

where $g : X_k \rightarrow \Re^{N_y}$ is a nonlinear function and n_k is a Gaussian noise with a covariance $R \in \Re^{N_y \times N_y}$, *i.e.*, $n_k \sim N(0, R)$.

Sample mean on the affine group. Since the optimal state estimation $\hat{X}_k$ is given by the weighted sample mean of particles $\{X_k^{(1)}, \ldots, X_k^{(N)}\}$, an appropriate formula to calculate the sample mean on *Aff*(2) is additionally required. Here we concentrate on the sample mean of *GL*(2) because that of $t \in \Re^2$ is trivially obtained. Given a set of *GL*(2) elements $\{G_1, \ldots, G_N\}$, their intrinsic mean $\bar{G}$ is defined as

$$\arg\min_{\bar{G}\in GL(2)} \sum_{i=1}^{N} d(\bar{G}, G_i)^2, \quad (5)$$

where $d(\cdot,\cdot)$ represents the geodesic distance between two *GL*(2) elements. Calculating the sample mean defined as (5) involves a difficult and computationally intensive optimization procedure, however.

Instead we can efficiently approximate the sample mean of *GL*(2) elements using the fact that minimal geodesics near the identity are given by the left and right translations of the one-parameter subgroups of the form e^{tU}, $U \in gl(2)$, $t \in \Re$. Moreover, if we resample the particles according to their weights at every time-step, all the resampled particles can be expected to be quite similar to each other. Thus the sample mean of $\{G_k^{(1)}, \ldots, G_k^{(N)}\}$, *GL*(2) components of the resampled particles $X_k^{(i)}$, can be approximated as

$$\bar{G}_k = G_{k,\max} \cdot \exp\left(\bar{U}_k\right), \quad (6)$$

$$\bar{U}_k = \frac{1}{N}\sum_{i=1}^{N} \log\left(G_{k,\max}^{-1} G_k^{(i)}\right), \quad (7)$$

where $G_{k,\max}$ denotes *GL*(2) part of the particle possessing the greatest weight before resampling. Then, the sample mean of $\{X_k^{(i)}, \ldots, X_k^{(N)}\}$ can be readily obtained as $\left[\begin{smallmatrix} \bar{G}_k & \bar{t}_k \\ 0 & 1 \end{smallmatrix}\right]$, where $\bar{t}_k \in \Re^2$ is the arithmetic mean of t.

2.2. Visual tracking via particle filtering on the affine group

We assume that the initial object template is automatically given as shown in Figure 1(a). Then the aim of tracking here is to estimate X_k representing the 2-D affine transformation of the object template in the k-th frame with respect to the 2-D image coordinates placed at the center of the object template in the initial frame (see Figure 1). Such a visual tracking task can be managed by the particle filtering procedure on *Aff*(2) described so far.

The term $A(X,t) \in$ *aff*(2) in (3) can be understood as the state dynamics determining the particle propagation. The simplest choice for the state dynamics is a random walk model, *i.e.*, $A(X,t) = 0$. The random walk model can be effective provided a sufficiently large number of particles are used, and the covariance P in (3) is sufficiently large. However, a more effective way to enhance tracking performance is to use a more appropriate state dynamics.

Here the state dynamics is modeled via the first-order autoregressive (AR) process on *Aff*(2). The state equation with the state dynamics based on the AR process on *Aff*(2) can be expressed as

$$X_k = X_{k-1} \cdot \exp\left(A_{k-1} + dW_k\sqrt{\Delta t}\right), \quad (8)$$

$$A_{k-1} = a \log\left(X_{k-2}^{-1} X_{k-1}\right), \quad (9)$$

where a is the AR process parameter. Since it is not possible to compute $A_k^{(i)}$ from $X_k^{(i)}$ and $X_{k-1}^{(i)}$ at the k-th time-step owing to the resampling process, the state is augmented as $\{X_k, A_k\}$ in practice. This AR-based state dynamics model can be understood as an infinitesimal constant velocity model.

Visual tracking on *Aff*(2) can now be efficiently performed via particle filtering with the AR-based state equation ((8) and (9)) and the appropriate measurement equation (4) depending on the appearance model employed.

3. Tracking using optimal importance function

The particle filtering algorithm mainly relies on the importance sampling [4]. The particles $X_k^{(i)}$ are sampled from the importance function $\pi(X_k|X_{0:k-1}^{(i)}, y_{0:k})$; and the weights $w_k^{(i)}$ for $X_{0:k}^{(i)}$ are evaluated as

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{p(y_k|X_k^{(i)})p(X_k^{(i)}|X_{k-1}^{(i)})}{\pi(X_k^{(i)}|X_{0:k-1}^{(i)}, y_{0:k})}. \quad (10)$$

The most popular choice for $\pi(X_k|X_{0:k-1}, y_k)$ is the state prediction density $p(X_k|X_{k-1})$ because the weights are simply determined proportionally to the measurement likelihood $p(y_k|X_k)$. Since information about recent measurements y_k are not incorporated in $p(X_k|X_{k-1})$, reliable performance cannot be expected when applied to visual tracking as mentioned in Section 1.

In [4], the optimal importance function minimizing the variance of particle weights (and eventually maintaining the

number of effective particles as large as possible) is given by $p(X_k|X_{k-1}, y_k)$. In this case, the particle weight calculation (10) generally cannot be derived analytically, one exception being the well-known case of Gaussian state space models with linear measurement equations.

For our tracking problem, since the pixel intensities of the image region determined by X_k is generally nonlinear in X_k, y_k is also nonlinear in X_k regardless of the form of the measurement function. Therefore, the use of the optimal importance function for our tracking problem generally results in an infeasible particle weight calculation. As a remedy for the nonlinear measurement case, an optimal importance function approximated via local linearization of the measurement function is proposed in [4].

3.1. Optimal importance function approximation

General vector space case. We first consider the following general nonlinear vector state space model:

$$x_k = f(x_{k-1}) + w_k, \quad (11)$$
$$y_k = g(x_k) + n_k, \quad (12)$$

where $x_k \in \Re^{N_x}$, $y_k \in \Re^{N_y}$, and w_k and n_k are Gaussian noise with covariances $P \in \Re^{N_x \times N_x}$ and $R \in \Re^{N_y \times N_y}$, respectively.

The basic assumption in approximating $p(x_k|x_{k-1}, y_k)$ is $p(x_k, y_k|x_{k-1})$ is jointly Gaussian [18]. If the mean μ and covariance Σ of $p(x_k, y_k|x_{k-1})$ are given by

$$\mu = (\mu_1, \mu_2)^\top = [E(x_k|x_{k-1}), E(y_k|x_{k-1})]^\top, \quad (13)$$
$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ (\Sigma_{12})^\top & \Sigma_{22} \end{pmatrix} = \begin{pmatrix} E(x_k x_k^\top|x_{k-1}) & E(x_k y_k^\top|x_{k-1}) \\ E(x_k y_k^\top|x_{k-1})^\top & E(y_k y_k^\top|x_{k-1}) \end{pmatrix}, \quad (14)$$

then $p(x_k|x_{k-1}, y_k)$ can be approximated as $N(m_k, \Sigma_k)$ incorporating the recent measurement y_k, where

$$m_k = \mu_1 + \Sigma_{12}(\Sigma_{22})^{-1}(y_k - \mu_2), \quad (15)$$
$$\Sigma_k = \Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}(\Sigma_{12})^\top. \quad (16)$$

In [4], the approximated mean $\tilde{\mu}$ and covariance $\tilde{\Sigma}$ of $p(x_k, y_k|x_{k-1})$ are given by

$$\tilde{\mu} = [f(x_{k-1}), g(f(x_{k-1}))]^\top, \quad (17)$$
$$\tilde{\Sigma} = \begin{pmatrix} P & PJ^\top \\ JP & JPJ^\top + R \end{pmatrix}, \quad (18)$$

where $J \in \Re^{N_y \times N_x}$ represents the Jacobian of $g(x_k)$ with respect to x_k evaluated at $f(x_{k-1})$, *i.e.*, $y_k \approx g(f(x_{k-1})) + J \cdot (x_k - f(x_{k-1})) + n_k$. Then $p(x_k|x_{k-1}, y_k)$ is approximated as $N(m_k, \Sigma_k)$ via (15) and (16) with (17) and (18). Particles are sampled from the derived normal distribution $N(m_k, \Sigma_k)$ and weighted via (10).

The affine group case. Note that the above normal distribution approximation approach should be performed geometrically in our visual tracking problem because the state is not a vector but an affine matrix. Therefore, the notions of normal distribution and Taylor expansion on *Aff*(2) need to be clarified.

For our purposes, we make use of the exponential map on *Aff*(2) which, as is well-known, locally defines a diffeomorphism between a neighborhood of *Aff*(2) containing the identity, and an open set of the Lie algebra *aff*(2) centered at the origin, *i.e.*, given $X \in Aff(2)$ sufficiently near the identity, the exponential map $X = \exp(\sum_{i=1}^{6} u_i E_i)$, where E_i are the basis elements of *aff*(2) shown in (2), is a local diffeomorphism. This local exponential coordinates can be extended to cover the entire group by left or right multiplication, *e.g.*, the neighborhood of any $Y \in Aff(2)$ can be well defined as $Y(u) = Y \cdot \exp\left(\sum_{i=1}^{6} u_i E_i\right)$ in our case. Recall also that in a neighborhood of the identity, the minimal geodesics are given by precisely these exponential trajectories.

The normal distribution on *Aff*(2) is then obtained as the exponential of a normal distribution on *aff*(2) (where well-defined), provided that the covariance values are sufficiently small to guarantee the local diffeomorphism of the exponential map. Let $N_{Aff(2)}(X, S)$ denote the approximated normal distribution on *Aff*(2) centered at $X \in Aff(2)$ with a covariance $S \in \Re^{6\times 6}$ for the normal distribution on *aff*(2). Then the random sampling from $N_{Aff(2)}(X, S)$ can be realized via the exponential mapping of the Gaussian noise on *aff*(2) as

$$X \cdot \exp(\sum_{i=1}^{6} \epsilon_i E_i), \epsilon = (\epsilon_1, \ldots, \epsilon_6)^\top \sim N(0, S). \quad (19)$$

It can be assumed that the covariance value for the Wiener noise dW_k in (8) is rather small because the frame-rate is generally sufficiently high enough (between 30 and 60 fps, and at least 15 fps) for the object motion between adjacent frames to be well described by our AR process-based dynamics and small covariance value for dW_k, as long as the object movement is not abrupt. With a small P for dW_k in (8), the following approximation to (8) can be considered as valid:

$$X_k \approx f(X_{k-1}) \cdot \exp(dW_k \sqrt{\Delta t}), \quad (20)$$

where $f(X_{k-1}) = X_{k-1} \cdot \exp(A_{k-1})$. Then $p(X_k|X_{k-1})$ can be approximated as $N_{Aff(2)}(f(X_{k-1}), Q)$, where $Q = P\Delta t$. Accordingly, μ_1, μ_2, and Σ_{11} in (13) and (14) are given by $f(X_{k-1})$, $g(f(X_{k-1}))$, and Q, respectively.

For a unimodular Lie group such as *SO*(3) and *SE*(3), it is shown in [24] that tightly concentrated distributions around the group identity element are essentially the distributions on its Lie algebra. Thus the normal distribution on, *e.g.*,

SE(3) can be identified via the exponential map with those on *se*(3), the Lie algebra of *SE*(3), in a similar way to our approximation. Unfortunately, such a notion for concentrated distributions does not hold for *Aff*(2), because *Aff*(2) is not connected, and accordingly not unimodular. In such cases one should attempt to at least verify, either analytically or experimentally, whether or not $N_{Aff(2)}(X, S)$ possesses the general properties of normal distributions; here we defer such verification to future work.

Now we focus on the Taylor expansion of a measurement function g on *Aff*(2) required to obtain the remaining Σ_{12} and Σ_{22} in (14) as (18). We can again make use of the exponential mapping on *Aff*(2). Representing the neighborhood of $f(X_{k-1})$ as $X(u)$ using the exponential map, *i.e.*, $X(u) = f(X_{k-1}) \cdot \exp\left(\sum_{i=1}^{6} u_i E_i\right)$, the first-order Taylor expansion of g around $f(X_{k-1})$ can be expressed as

$$y_k \approx g(f(X_{k-1})) + J \cdot u + n_k, \tag{21}$$

where $J \in \Re^{N_y \times 6}$ is the Jacobian of $g(X(u))$ with respect to u evaluated at $u = 0$, *i.e.*, $f(X_{k-1})$. Then Σ_{12} and Σ_{22} are respectively given by $QJ^\top$ and $JQJ^\top + R$ as (18). The exponential coordinate-based approach to obtain the Taylor expansion of a function on general Lie groups can also be found in the literature on gradient and Hessian-based optimization methods on Lie groups [13],[23].

Finally, the optimal importance function can be approximated as $N_{Aff(2)}(m_k, \Sigma_k)$ by rewriting (15) and (16) as

$$\bar{u} = \Sigma_{12}(\Sigma_{22})^{-1}(y_k - \mu_2), \tag{22}$$

$$m_k = \mu_1 \cdot \exp\left(\sum_{i=1}^{6} \bar{u}_i E_i\right), \tag{23}$$

$$\Sigma_k = \Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}(\Sigma_{12})^\top, \tag{24}$$

where $\mu_1 = f(X_{k-1})$, $\mu_2 = g(f(X_{k-1}))$, $\Sigma_{11} = Q$, $\Sigma_{12} = QJ^\top$, and $\Sigma_{22} = JQJ^\top + R$ as previously derived.

3.2. Analytic Jacobian derivation via a chain rule

For the sake of an analytic Jacobian derivation, we borrow the expressions about the image region determined by X_k from [1]. The warping function $w(p; X_k)$ represents the affine transformation of pixel coordinates $p = (p_x, p_y)$ of the initial object template induced by X_k, *i.e.*, $w(p; X_k) = X_k \cdot p$ in the homogeneous coordinates. Then let $I(w(p; X_k))$ represent the image region in the current video frame, which is determined by X_k transforming the pixel coordinates of the object template and warped back to the initial pixel coordinates. Then the measurement equation (4) can be more explicitly expressed as

$$y_k = g(X_k) + n_k = h(I(w(p; X_k))) + n_k, \tag{25}$$

where h is a real-valued nonlinear function taking an image as an input argument. The only constraint for h is that it should be differentiable with respect to its input argument in order that one may obtain an analytic Jacobian of $g(X(u))$. Now the analytic Jacobian J of $g(X(u))$ with respect to u evaluated at $u = 0$ can be obtained via a chain rule as

$$J_i = \left.\frac{\partial g(X(u))}{\partial u_i}\right|_{u=0} = \frac{\partial h(I(w(p; X_k)))}{\partial I(w(p; X_k))} \cdot \frac{\partial I(w(p; X_k))}{\partial w(p; X_k)} \cdot \frac{\partial w(p; X_k)}{\partial X_k} \cdot \left.\frac{\partial X(u)}{\partial u_i}\right|_{u=0} \tag{26}$$

where J_i is the i-th column of J corresponding to u_i, and X_k represents $f(X_{k-1})$ for concise expression.

The first term $\frac{\partial h(I(w(p;X_k)))}{\partial I(w(p;X_k))}$ surely depends on the choice of h; its analytic form for the measurement function based on the principal component analysis (PCA) is derived in the next subsection. The second term $\frac{\partial I(w(p;X_k))}{\partial w(p;X_k)}$ corresponds to the image gradient ∇I calculated at the pixel coordinates on the current video frame and warped back to the object template coordinates. The third term $\frac{\partial w(p;X_k)}{\partial X_k}$ can be obtained via differentiation of the transformed pixel coordinates with respect to a vector $a_k = \{a_{k,1}, \ldots, a_{k,6}\}$ constituting X_k as $\begin{bmatrix} a_{k,1} & a_{k,3} & a_{k,5} \\ a_{k,2} & a_{k,4} & a_{k,6} \\ 0 & 0 & 1 \end{bmatrix}$; the result is simply $\begin{bmatrix} p_x & 0 & p_y & 0 & 1 & 0 \\ 0 & p_x & 0 & p_y & 0 & 1 \end{bmatrix}$. Finally, the last term can be easily computed as

$$\left.\frac{\partial X(u)}{\partial u_i}\right|_{u=0} = \left.\frac{\partial X_k \cdot \exp(\sum_i u_i E_i)}{\partial u_i}\right|_{u=0} = X_k E_i. \tag{27}$$

Note that $X_k E_i$ should also be represented in a $\Re^6$ vector as a_k for consistency with the representation of $\frac{\partial w(p;X_k)}{\partial X_k}$.

3.3. PCA-based measurement function

The measurement function h is directly related with the object appearance model. In our framework, we adopt the PCA-based appearance model whose applicability to object tracking beyond classical object recognition has been previously shown in [2, 8]. Since we assume that the training data are not available *a priori*, the principal eigenvectors and object mean are updated during tracking via the incremental PCA learning algorithm in [16]. The specific algorithm for the incremental PCA learning can be found in [16]. In the remaining part of this paper, $I(p)$ is often used instead of $I(w(p; X_k))$ without notification for concise expression when there is no confusion.

Let $\bar{T}(p)$ and $b_i(p), i = 1, \ldots, M$, respectively denote the mean and first M principal eigenvectors which are incrementally updated from the set of the tracked object images determined by the optimal state estimation $\hat{X}_{0:k}$. Then the reconstruction error e for $I(p)$ can be expressed as

$$e^2 = \sum_p \left(I(p) - \bar{T}(p)\right)^2 - \sum_{i=1}^{M} c_i^2, \tag{28}$$

where c_i are the projection coefficients of the mean-normalized image to each principal eigenvector $b_i(p)$, *i.e.*, $c_i = \sum_p b_i(p)(I(p) - \bar{T}(p))$. In the probabilistic PCA framework of [15], this error is understood as the "distance-from-feature-space" (DFFS) representing how much the warped image $I(p)$ differs from the object image represented by $\bar{T}(p)$ and $b_i(p)$ while the "distance-in-feature-space" (DIFS) is defined as the Mahalanobis distance, *i.e.*, $\sum_{i=1}^{M} \frac{c_i^2}{\lambda_i}$ where λ_i are the eigenvalues for $b_i(p)$.

With the PCA-based appearance model, the measurement function h can be defined using the DFFS and DIFS as $h(I(p)) = (e^2, \sum_{i=1}^{M} \frac{c_i^2}{\lambda_i})^\top \in \Re^2$. Then the measurement equation (25) can be explicitly expressed as

$$y_k = h(I(w(p; X_k))) + n_k = \begin{bmatrix} e^2 \\ \sum_{i=1}^{M} \frac{c_i^2}{\lambda_i} \end{bmatrix} + n_k, \quad (29)$$

where $n_k \sim N(0, R), R = \left[\begin{smallmatrix} \sigma^2 & 0 \\ 0 & 1 \end{smallmatrix}\right] \in \Re^{2\times 2}$. Correspondingly, the measurement likelihood can be calculated as

$$p(y_k|X_k) \propto \exp\left(-\frac{1}{2} y_k^\top R^{-1} y_k\right). \quad (30)$$

The derivative of the DFFS term of $h(I(p))$ with respect to $I(p)$ can be straightforwardly derived as

$$\begin{aligned} \frac{\partial e^2}{\partial I(p)} &= \sum_p 2(I(p) - \bar{T}(p)) - \sum_{i=1}^{M} 2c_i \sum_p b_i(p) \\ &= \sum_p \left(2(I(p) - \bar{T}(p)) - \sum_{i=1}^{M} 2c_i b_i(p)\right) \quad (31) \end{aligned}$$

The derivative of the DIFS term of $h(I(p))$ with respect to $I(p)$ can also be derived similarly as

$$\frac{\partial \sum_{i=1}^{M} \frac{c_i^2}{\lambda_i}}{\partial I(p)} = \sum_p \left(\sum_{i=1}^{M} \frac{2c_i}{\lambda_i} b_i(p)\right). \quad (32)$$

Then the overall Jacobian J is obtained via plugging (31) and (32) into (26) appropriately.

In the initial phase before the minimum number of tracked object images required to perform PCA is collected, the measurement function h becomes the SSD between $I(p)$ and the initial object template $T(p)$, *i.e.*, $h = \sum_p (I(p) - T(p))^2$; and its derivative is simply given by $\sum_p 2(I(p) - T(p))$. The overall algorithm of our visual tracking framework via particle filtering on *Aff*(2) using an optimal importance function is shown in Algorithm 1.

Algorithm 1 Overall algorithm

1. Intialization
- a. Set $k = 0, l = 0$.
- b. Set l_{update} for the incremental PCA.
- c. Set number of particles as N.
- d. For $i = 1, \ldots, N$, set $X_0^{(i)} = I, A_0^{(i)} = 0$.

2. Importance sampling step
- a. Set $k = k + 1, l = l + 1$.
- b. For $i = 1, \ldots, N$, draw $X_k^{(*i)} \sim p(X_k|X_{k-1}^{(i)}, y_k)$, *i.e.*,
  - Calculate $J^{(i)}$ of $g(X(u))$ at $f(X_{k-1}^{(i)})$.
  - Determine $m_k^{(i)}$ and $\Sigma_k^{(i)}$ via (22), (23), and (24).
  - Draw $X_k^{(*i)} \sim N_{Aff(2)}(m_k^{(i)}, \Sigma_k^{(i)})$ via (19).
  - Compute $A_k^{(*i)}$ with (9).
- c. For $i = 1, \ldots, N$, calculate the weights $w_k^{(i)}$ via (10).
- d. For $i = 1, \ldots, N$, normalize the weights: $\tilde{w}_k^{(i)} = \frac{w_k^{(i)}}{\sum_j w_k^{(j)}}$.

3. Selection step (resampling)
- a. Resample from $X_k^{(*i)}$ and $A_k^{(*i)}$ according to $\tilde{w}_k^{(i)}$ to produce i.i.d. $X_k^{(i)}$ and $A_k^{(i)}$.
- b. For $i = 1, \ldots, N$, set $w_k^{(i)} = \tilde{w}_k^{(i)} = \frac{1}{N}$.
- c. If $l = l_{\text{update}}$, update $\bar{T}(p)$ and $b_i(p)$, and set $l = 0$

4. Go to the importance sampling step

4. Experiments

4.1. Experiment 1

We first demonstrate the validity of our geometric approach to the 2-D affine motion tracking problem via comparison with the tracker of [16], which is considered to be one of the state-of-the-art particle filtering-based affine motion trackers. In [16], the state is represented by a 6-D vector using a set of local coordinates. Since we use the same appearance model as that of [16], it can be verified how much the state representation affects the overall performance. For fair comparison, instead of using the optimal importance function derived in Section 3, we run our tracker using the same importance function as the one used in [16], *i.e.*, the state prediction density. The number of particles used is 600.

The video sequences used for comparison include the "David", "Trellis", and "Sylvester" sequences used in [16]; and the tracking results are shown in Figure 3[1]. We can see that both trackers yield almost the same performance for the "David" and "Trellis" sequences. The tracker of [16], however, fails to track the object at the latter part of the "Sylvester" sequence owing to the abrupt object pose and illumination changes, while our tracker tracks the object well. Therefore it is fair to say that the tracking results in Figure 3 are one of the proofs of the validity of our geometric approach to the 2-D affine motion tracking.

4.2. Experiment 2

We next examine the effectiveness of the use of the optimal importance function derived in Section 3. We compare the tracking results, which are obtained by our trackers us-

[1]The video containing all the tracking results shown in Section 4 is available at http://cv.snu.ac.kr/jhkwon/tracking/.

Figure 3. The tracking results for "David" (top row), "Trellis" (middle row), and "Sylvester" (bottom row), by our tracker (yellow rectangle) and the tracker of [16] (red rectangle).

ing two importance functions, *i.e.*, the optimal importance function and the state prediction density. Both trackers run with the same number of particles (400) and the same covariance P for the Wiener noise on $aff(2)$.

The test video sequences are the "Cube", "Vase", and "Toy" sequences containing the 2-D affine motion of the object template, which is occasionally difficult to predict correctly via the state dynamics model with smooth motion assumption. Since our AR-based dynamics model also assumes smooth motion between infinitesimal time intervals, the need for an optimal importance function instead of state prediction density for such sequences is clear.

The tracking results are shown in Figure 4. For the "Cube" and "Vase" sequences, the accuracy of the state prediction density-based tracker is degraded especially when the object makes an abrupt movement. The tracker using the optimal importance function, however, tracks the object quite accurately over all the frames regardless of the abrupt motion change. For the "Toy" sequences, which can be regarded as the most difficult one, the tracker using the optimal importance function also tracks the object well while the state prediction density-based tracker loses the object entirely during tracking.

The efficiency of both trackers can also be compared in terms of the number of effective particles N_{eff}, which is defined as $[\sum_{i=1}^{N}(\tilde{w}_k^{(i)})^2]^{-1}$ [4]. It can be easily checked that N_{eff} varies between one (in the worst case) and N (in the ideal case). Figure 5 shows the plots of N_{eff} for the "Cube" sequence over all the frames. We can see that N_{eff} of the tracker using the optimal importance function is greater than that of the state prediction density-based tracker over almost all the frames; the consequence is the superior performance of the tracker using the optimal importance function as shown in Figure 4.

Table 1 shows $\bar{N}_{\text{eff}}$, the averages of N_{eff} over all the frames, for each sequence. For all the sequences, we can see that $\bar{N}_{\text{eff}}$ of the tracker using the optimal importance function is consistently greater than that of the state prediction density-based tracker. Note that, when a tracker fails to localize the object correctly, $\bar{N}_{\text{eff}}$ may not represent the actual tracking performance well because $\bar{N}_{\text{eff}}$ may increase in such a situation owing to a possibility that many particles have similar weights with very low value. Therefore, for the "Toy" sequence, $\bar{N}_{\text{eff}}$ is calculated only over the first 130 frames before the tracking accuracy of the state prediction density-based tracker becomes much worse.

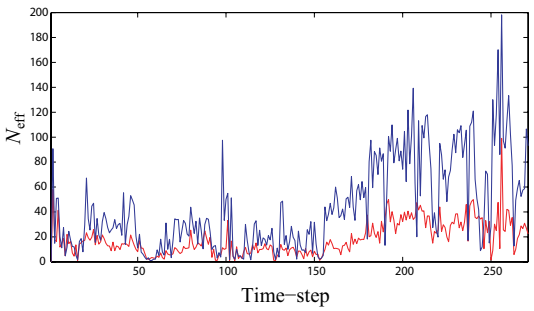


Figure 5. Plots of N_{eff} for the "Cube" sequence. The blue and red lines respectively represent the cases using the optimal importance function and state prediction density as the importance function.

Importance function	Cube	Vase	Toy
Optimal importance function	43.43	13.69	15.69
State prediction density	18.12	8.37	10.32

Table 1. $\bar{N}_{\text{eff}}$, the averages of N_{eff} over all the frames.

From these results, we can conclude that the optimal importance function derived for our geometric framework actually enhances the visual tracking performance in situations where the tracking may fail if the state prediction density were used as the importance function.

5. Conclusions

In this paper, we have proposed a novel geometric framework to efficiently solve the 2-D affine motion tracking problem. In our framework, the 2-D affine motion is basically recast as the sequential filtering problem on the affine group $Aff(2)$, which is a matrix Lie group representing a set of 2-D affine transformations. The optimal importance function required to enhance the filtering performance has been geometrically derived for the affine group via the first-order Taylor expansion of a measurement function on $Aff(2)$ with careful clarification of notions of the neighborhood and normal distribution on $Aff(2)$. We have also derived the Jacobian of a PCA-based measurement function whose input argument is $Aff(2)$ analytically via a chain rule. The feasibility of our proposed framework has been effectively

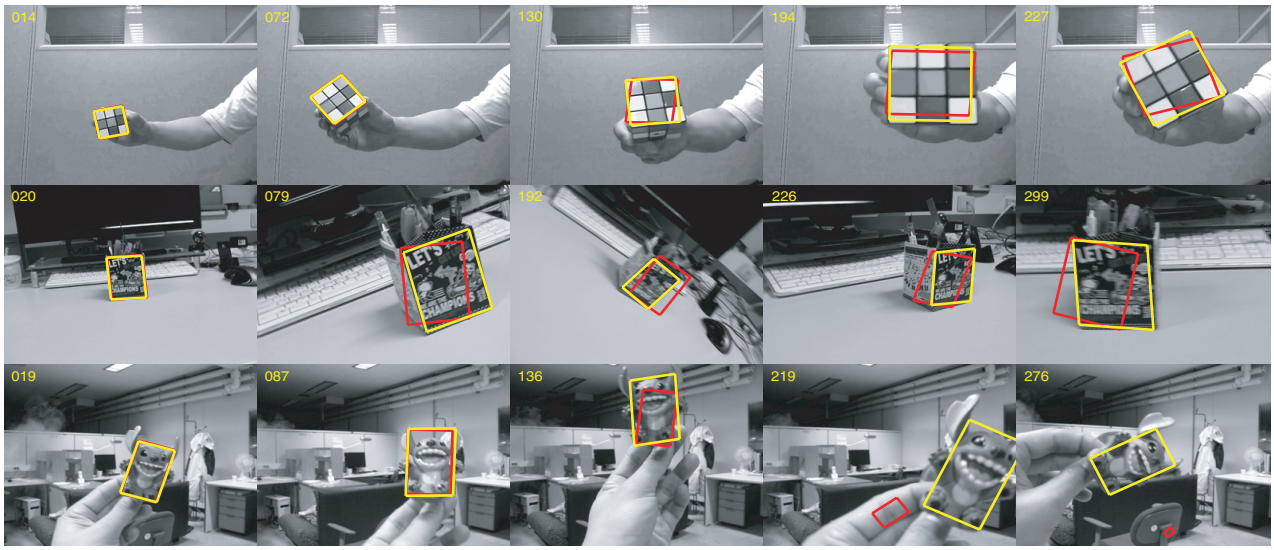


Figure 4. The tracking results for "Cube" (top row), "Vase" (middle row), and "Toy" (bottom row), by our trackers, respectively, using the optimal importance function (yellow rectangle) and the state prediction density (red rectangle) as the importance function.

demonstrated via comparative experimental studies.

Acknowledgements

This research was supported in part by the Defense Acquisition Program Administration and Agency for Defense Development through IIRC [UD070007AD], and in part by the IT R&D program of MKE/IITA [2008-F-030-01], Korea.

References

[1] S. Baker and I. Matthews. Lucas-kanade 20 years on: a unifying framework. *Int. J. Comupt. Vision*, 56(3):221–255, 2004.

[2] M. Black and A. Jepson. Eigentracking: robust matching and tracking of articulated objects using a view-based representation. In *Proc. ECCV*, 1998.

[3] A. Chiuso and S. Soatto. Monte carlo filtering on lie groups. In *Proc. IEEE CDC*, pages 304–309, 2000.

[4] A. Doucet, S. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Stat. Comput.*, 10:197–208, 2000.

[5] B. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Springer, 2004.

[6] M. Isard and A. Blake. Condensation–conditional density propagation for visual tracking. *Int. J. Comput. Vision*, 29:5–28, 1998.

[7] S. Julier and J. Uhlmann. Unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(3):401–422, 2004.

[8] Z. Khan, T. Balch, and F. Dellaert. A rao-blackwellized particle filter for eigentracking. In *Proc. CVPR*, 2004.

[9] J. Kwon, M. Choi, F. Park, and C. Chun. Particle filtering on the euclidean group: framework and applications. *Robotica*, 25(6):725–737, 2007.

[10] J. Kwon and F. Park. Visual tracking via particle filtering on the affine group. In *Proc. IEEE ICIA*, pages 997–1002, 2008.

[11] P. Li, T. Zhang, and A. Pece. Visual contour tracking based on particle filters. *Image Vision Comput.*, 21:111–123, 2003.

[12] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo. Robust visual tracking based on incremental tensor subspace learning. In *Proc. ICCV*, 2007.

[13] R. Mahony and J. Manton. The geometry of the newton method on non-compact lie groups. *J. Global Optim.*, 23:309–327, 2002.

[14] M.K.Pitt and N. Shephard. Filtering via simulation: auxiliary particle filter. *J. Amer. Stat. Assoc.*, 94(446):590–599, 1999.

[15] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):696–710, 1997.

[16] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *Int. J. Comput. Vision*, 77(1-3):125–141, 2008.

[17] Y. Rui and Y. Chen. Better proposal distributions: object tracking using unscented particle filter. In *Proc. CVPR*, 2001.

[18] S. Saha, P. Mandal, Y. Boers, and H. Driessen. Gaussian proposal denstiy using moment matching in smc. *Stat. Comput.*, 19:203–208, 2009.

[19] C. Shen, A. van den Hengel, A. Dick, and M. Brooks. Enhanced importance sampling: unscented auxiliary particle filtering for visual tracking. *Lect. Notes Artif. Intell.*, 3339:180–191, 2005.

[20] A. Srivastava. Bayesian filtering for tracking pose and location of rigid targets. In *Proc. SPIE AeroSense*, 2000.

[21] A. Srivastava, U. Grenander, G. Jensen, and M. Miller. Jump-diffusion markov processes on orthogonal groups for object pose estimation. *J. Stat. Plan. Infer.*, 103:15–37, 2002.

[22] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. In *Proc. NIPS*, 2000.

[23] T. Vercauteren, X. Pennec, E. Malis, A. Perchant, and N. Ayache. Insight into efficient image registration techniques and the demons algorithm. *Lect. Notes Comput. Sci.*, 4584:495–506, 2008.

[24] Y. Wang and G. Chirikjian. Error propagation on the euclidean group with applications to manipulator kinematics. *IEEE Trans. Robotics*, 22(4):591, 602 2006.

[25] A. Yilmaz, O. Javed, and M. Shah. Object tracking: a survey. *ACM Comput. Surv.*, 38(4), 2006.

[26] S. Zhou, R. Chellappa, and B. Moghaddam. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.*, 13(11):1491–1506, 2004.