

# Monocular SLAM with Locally Planar Landmarks via Geometric Rao-Blackwellized Particle Filtering on Lie Groups

Junghyun Kwon and Kyoung Mu Lee

Department of EECS, ASRI, Seoul National University, Seoul 151-742, Korea

junghyunkwon@gmail.com, kyougm@snu.ac.kr

## Abstract

We propose a novel geometric Rao-Blackwellized particle filtering framework for monocular SLAM with locally planar landmarks. We represent the states for the camera pose and the landmark plane normal as  $SE(3)$  and  $SO(3)$ , respectively, which are both Lie groups. The measurement error is also represented as another Lie group  $SL(3)$  corresponding to the space of homography matrices. We then formulate the unscented transformation on Lie groups for optimal importance sampling and landmark estimation via unscented Kalman filter. The feasibility of our framework is demonstrated via various experiments.

## 1. Introduction

Visual SLAM is the process of simultaneously estimating the camera pose and 3-D structure of the environment from 2-D image sequences [7, 8, 22]. Usually, landmarks in visual SLAM are regarded as 3-D points. However, in the case of SLAM in man-made environments, the point landmarks are not sufficient to fully represent the 3-D structure of the environment because man-made environments are usually composed of lots of planes, *e.g.*, walls, floors, and various man-made planar objects. Thus, there have been several attempts to discover higher-level structures such as lines and planes from the estimated point landmarks [9, 16].

In this paper, we take a different view from [9, 16] to a richer representation of the environment. We regard the landmark as a locally planar 3-D object rather than a 3-D point, and estimate both landmark 3-D position and its plane normal<sup>1</sup> as shown in Fig. 1. Locally planar landmarks can be considered as a higher-level description of the man-made environment than point landmarks. Moreover, it is evident that locally planar landmarks can give more advanced cues than point landmarks for higher-level structure inference from the estimated landmarks.

<sup>1</sup>Hereafter, we use the term “landmark pose” to mean the landmark 3-D position and its plane normal.

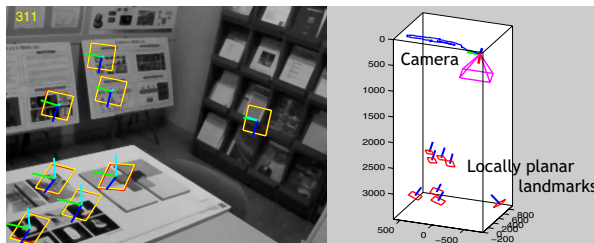


Figure 1. The output example of our monocular SLAM system with locally planar landmarks. Both camera and landmark poses are accurately estimated via our geometric RBPF on Lie groups.

When using Rao-Blackwellized particle filter (RPBF), the landmark pose can be estimated via extended Kalman filter (EKF) or unscented Kalman filter (UKF) based on the sampled camera pose. Since both are Gaussian filters, we should represent the landmark and measurement uncertainties as Gaussian distributions. It is, however, not straightforward to define Gaussian on the space of directional vectors representing the landmark plane normal because it is a curved space. It is also awkward to represent the measurement uncertainty as Gaussian because the measurement error owing to the errors of the camera and landmark poses is represented by the homography transformation of the landmark projection, and the space of homography matrices is also a curved space.

To resolve these difficulties in visual SLAM with locally planar landmarks via RBPF, we take a geometric approach based on Lie group theory. The motivation is both plane normal and homography can be represented by  $SO(3)$  and  $SL(3)$ , respectively, which are both Lie groups, and we can represent their uncertainties as Gaussian distributions on Lie groups defined in [2]. Note that the camera pose can also be considered as another Lie group  $SE(3)$ .

With these considerations, our primary contribution is to propose a *geometric RBPF on Lie groups* to solve *monocular SLAM with locally planar landmarks*. The geometric RBPF on Lie groups means that both camera state particle sampling and landmark state update are explicitly done on Lie groups without any local coordinate parametrization. In

detail, the camera state and the landmark state for the plane normal are defined as  $SE(3)$  and  $SO(3)$ , respectively. The measurement is defined as 2-D coordinates of four corner points of the quadrilateral planar landmark projection, and its error is represented by  $SL(3)$ . The uncertainties of the camera state, the landmark state for the plane normal, and the measurement are then represented as Gaussian on  $SE(3)$ ,  $SO(3)$ , and  $SL(3)$ , respectively. With these geometric definitions, we formulate the unscented transformation (UT) on Lie groups for optimal importance sampling of the camera state particle and landmark pose estimation both via UKF.

In addition to the geometric formulation of RBPF on Lie groups, we also present various convincing experimental results that our geometric framework yields quite satisfactory performance for visual SLAM with locally planar landmarks employing only a monocular camera.

### 1.1. Related work

The first monocular SLAM system dealing with locally planar landmarks is [18]. The limitation of [18] is that the landmark plane normal is separately estimated outside the main EKF, which performs SLAM with point landmarks. This means that the estimated plane normal does not contribute to the camera pose estimation. In our framework, the estimated landmark pose directly contributes to the camera pose estimation since we employ only a single RBPF.

Although the visual SLAM system where the landmark pose is estimated along with the camera pose within a single EKF was proposed in [19], only the limited experimental results using the *stereo* camera were reported. Berger *et al* also relied on the stereo images to estimate the normal of the planar landmarks called planar facets [4].

To the best of our knowledge, there is currently no successful monocular SLAM system with locally planar landmarks within a single filtering framework. In this paper, we realize such visual SLAM system via our geometric RBPF on Lie groups. We remark that the optimization-based approach to monocular SLAM with locally planar landmarks was recently proposed in [21]. We start with discussing the necessary geometric background on Lie groups in Section 2.

## 2. Preliminaries on Lie groups

A Lie group  $G$  is a group which is a differentiable manifold with smooth group operations. The Lie algebra  $g$  for  $G$  is defined as the tangent vector space at the identity of  $G$ .  $G$  and  $g$  are related via the exponential and log maps,  $\exp : g \rightarrow G$  and  $\log : G \rightarrow g$ . For matrix Lie groups,  $\exp$  and  $\log$  correspond to the ordinary matrix exponential and log. The local neighborhood of  $X \in G$  can be well defined by using the *exponential coordinates* as  $X(u) = X \cdot \exp(\sum_i u_i E_i)$  where  $E_i$  are basis elements

of  $g$ . The Lie algebra  $g$  of a matrix Lie group  $G$  can be represented also in a column vector by using the map  $v : g \rightarrow \mathbb{R}^n$  with the basis elements  $E_i$  of  $g$ , i.e.,  $v(\sum_{i=1}^n e_i E_i) = (e_1, \dots, e_n)^\top$ .

$SO(3)$ , the space of  $3 \times 3$  real orthogonal matrices with the unit determinant, can represent 3-D rotation matrices or 3-D coordinate frames.  $SE(3)$  representing the rigid body transformation is in the form of  $\begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}$ , where  $R \in SO(3)$  and  $p \in \mathbb{R}^3$ .  $SL(3)$ , the space of  $3 \times 3$  real matrices with the unit determinant, corresponds to the space of homography matrices representing the 8-DOF projective transformation of the planar object image. The Lie algebras of  $SO(3)$ ,  $SE(3)$ , and  $SL(3)$  are denoted by  $so(3)$ ,  $se(3)$ , and  $s(3)$ , respectively, and their basis elements  $E_i$  can be found in, e.g., [3, 24].

The practical definition of Gaussian on semi-simple Lie groups such as  $SL(3)$  and  $SO(3)$  is given in [2]. The underlying principle is to define Gaussian on the Lie algebra using the Riemannian exponential and log maps, which are derived from the geodesics on semi-simple Lie groups. Denoting the Riemannian exponential and log maps at the identity by  $expp$  and  $logg$ , respectively, then Gaussian on  $SL(3)$  or  $SO(3)$  is defined as

$$N(X; \mu, \Sigma) \propto e^{-\frac{1}{2}v(logg(\mu^{-1}X))^\top \Sigma^{-1}v(logg(\mu^{-1}X))}, \quad (1)$$

where  $\Sigma$  is the covariance on the Lie algebra.  $expp$  for  $SL(3)$  and  $SO(3)$  is given by  $expp(x) = \exp(-x^\top) \exp(x + x^\top)$  where  $x$  is the Lie algebra element.  $expp$  and  $logg$  correspond to  $\exp$  and  $\log$  particularly for  $SO(3)$ . The sampling from  $N(X; \mu, \Sigma)$  is realized as

$$\mu \cdot expp(v^{-1}(\epsilon)), \quad (2)$$

where  $\epsilon$  is sampled from Gaussian on the Lie algebra with the zero mean and the covariance  $\Sigma$ .

The efficient gradient-descent algorithm to obtain the sample mean of  $SL(3)$  is presented in [2] while the sample mean formula of  $SO(3)$  is given in [14]. For  $SL(3)$  and  $SO(3)$ , with the sample mean  $\bar{\mu}$  of  $\{X_1, \dots, X_m\}$ , the sample covariance  $\bar{\Sigma}$  is obtained as

$$\bar{\Sigma} = \frac{1}{m+1} \sum_{i=1}^m v(logg(\bar{\mu}^{-1}X_i)) \cdot v(logg(\bar{\mu}^{-1}X_i))^\top. \quad (3)$$

Since  $SE(3)$  is not a semi-simple Lie group, we define Gaussian on  $SE(3)$  by using the exponential coordinates, which actually corresponds to replace  $expp$  and  $logg$  in (1) and (2) by  $\exp$  and  $\log$ . The sample mean of  $SE(3)$  is simply determined with the sample mean of  $SO(3)$  and  $\mathbb{R}^3$ . The sample covariance of  $SE(3)$  is also obtained with (3) using  $\log$  instead of  $logg$ . Our definition of Gaussian on  $SE(3)$  can be considered as valid particularly when the covariance  $\Sigma$  is sufficiently small [24]. Hereafter, we use the subscripts in representing Gaussian on each Lie group as  $N_{SL(3)}$ ,  $N_{SO(3)}$ , and  $N_{SE(3)}$ ; similarly  $v_{sl(3)}$ ,  $v_{so(3)}$ , and  $v_{se(3)}$ .

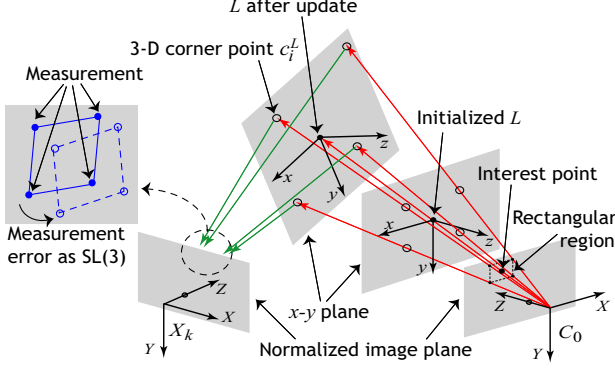


Figure 2. The planar landmark  $L$  is initialized on the ray passing the camera center of  $C_0$  and the detected interest point on the normalized image plane. The  $z$ -axis of  $L$  is initially set to be parallel to the  $Z$ -axis of  $C_0$ .  $L$  is always on the same ray even after the landmark update. The 3-D corner points  $c_i^L$  of  $L$  are determined by the intersections between the  $x-y$  plane of  $L$  and the rays passing the camera center of  $C_0$  and the corner points of the rectangular region on the normalized image plane. The measurement  $y_k$  is the projections of  $c_i^L$  to the image plane of  $X_k$ , and its error is represented as  $SL(3)$ .

### 3. Problem formulation

#### 3.1. State definition

The camera state  $X$  is defined as  $SE(3)$  itself without any local coordinate parametrization. With the assumption of a smooth camera motion, we use the following state equation as [15]:

$$X_k = X_{k-1} \cdot \exp(A_{k-1} + v_{se(3)}^{-1}(w_k)\sqrt{\Delta t}), \quad (4)$$

$$A_{k-1} = a \log(X_{k-2}^{-1}X_{k-1}), \quad (5)$$

where  $a$  is an auto-regressive parameter and  $w_k \in \mathbb{R}^6$  is the Wiener process noise on  $se(3)$  with a covariance  $P \in \mathbb{R}^{6 \times 6}$ . Denoting  $X_{k-1} \cdot \exp(A_{k-1})$  by  $f(X_{k-1})$ ,  $X_k$  can be approximated as  $f(X_{k-1}) \cdot \exp(v_{se(3)}^{-1}(w_k)\sqrt{\Delta t})$ . Then, considering (1) and (2), we can represent  $p(X_k|X_{k-1})$  as  $N_{SE(3)}(f(X_{k-1}), P')$  where  $P' = P\Delta t$ .

The landmark state  $L$  is defined as  $L = \{s, N\}$ , where  $s$  is the landmark 3-D position and  $N \in SO(3)$  is the coordinate frame attached to the planar landmark so that the  $z$ -axis represents the plane normal as shown in Fig. 2. Since we employ only a monocular camera, we represent  $s$  as a  $\mathbb{R}^6$  vector by using the inverse depth parametrization of [6].

#### 3.2. Landmark initialization

When the interest point is detected on the video frame, we initialize the landmark  $L = \{s, N\}$  as follows. First,  $s$  is initialized to be on the ray passing through the interest point on the normalized image coordinates and the camera center as shown in Fig. 2. Then we can represent the depth

uncertainty including the infinity by a single Gaussian over the inverse depth  $\rho$  along the ray. Please refer to [6] for the detailed description of inverse depth parametrization.  $N$  for the landmark plane normal is initialized as  $SO(3)$  with the  $z$ -axis opposite to the camera viewing direction as shown in Fig. 2. We set  $x$ - and  $y$ -axes of  $N$  to be parallel to the image coordinate axes. This initialization should be done for each camera state particle in an RBPF framework.

Assuming there is no error in the position of the detected interest point, the uncertainty of  $s$  is only about the inverse depth  $\rho$ . We set the standard deviation for  $\rho$ ,  $\sigma_\rho$ , to include the infinity in the 95% confidence region. To represent the uncertainty of  $N$ , we use the Gaussian on  $SO(3)$  as  $N_{SO(3)}(N, \Sigma_N)$  where  $\Sigma_N = \text{diag}(\sigma_x^2, \sigma_y^2, 0)$ . The standard deviations for  $x$ - and  $y$ -axes of  $N$ ,  $\sigma_x$  and  $\sigma_y$ , are set to be  $\frac{\pi}{3}$  to cover large plane normal error in the initialization. Note that we do not have to assign the uncertainty for the  $z$ -axis of  $N$ . Thus, the covariance for  $L$  is represented as  $\Sigma_L = \text{diag}(\sigma_\rho^2, \sigma_x^2, \sigma_y^2)$ .

#### 3.3. Measurement equation

Along with the initialization of  $L$ , we also designate the rectangular region centered at the interest point for the measurement  $y_k$  from  $L$ . Using the rectangular corners in the normalized image coordinates and the camera pose at the landmark initialization, we can determine  $c_i^L$ , the 3-D corner points of  $L$ , as depicted in Fig. 2. Then,  $y_k$  is given by the projections of  $c_i^L$  in the homogeneous coordinates as

$$y_k = g(X_k, L, n_k) = \begin{bmatrix} n_k \cdot h(X_k^{-1} \cdot c_1^L) \\ \vdots \\ n_k \cdot h(X_k^{-1} \cdot c_4^L) \end{bmatrix}, \quad (6)$$

where  $h$  is the camera projection function with the internal parameters and  $n_k$  is a random  $3 \times 3$  homography matrix representing the measurement error. When  $L$  is updated,  $c_i^L$  should be re-determined.

To represent the measurement uncertainty as Gaussian, we regard  $n_k$  as a random  $SL(3)$  element sampled from  $N_{SL(3)}(I, R)$  via (2) with a covariance  $R \in \mathbb{R}^{8 \times 8}$  on  $sl(3)$ . The homography between the two images is determined from the simple linear equation with four point correspondences [11]. Denoting the function calculating the homography transformation from the point set  $A$  to another point set  $B$  by  $r(A, B)$ , the measurement likelihood  $p(y_k|X_k)$  is represented as  $N_{SL(3)}(r(\bar{y}_k, y_k); I, R)$  where  $\bar{y}_k$  is the predicted measurement, *i.e.*,  $\bar{y}_k = g(X_k, L, I)$ .

### 4. Geometric RBPF on Lie groups

In our RBPF, the state particles  $S_k^{(i)}$  with  $K$  landmarks are in the form of  $\{X_k^{(i)}, L_l^{(i)}, \Sigma_{L_l}^{(i)}, l = 1, \dots, K\}$ . The camera state particles  $X_k^{(i)}$  are first sampled via importance

sampling, and  $L_l^{(i)}$  and  $\Sigma_{L_l}^{(i)}$  are updated based on the sampled  $X_k^{(i)}$  and  $y_k$ . Now we formulate UT on Lie groups in consideration of the geometric properties discussed in Section 2 for optimal importance sampling and landmark update both via UKF. One of the reasons to use UKF instead of EKF is the calculus on Lie groups is not straightforward in general. Moreover, the superiority of UKF to EKF in the RBPF-based SLAM has been recently reported in [13].

#### 4.1. Optimal importance sampling

The optimal importance function for particle filtering is given by  $p(X_k|X_{k-1}^{(i)}, y_k)$ , which means that  $y_k$  should be considered when the particles are sampled. Adopting the approach of [15], we can approximate the optimal importance function as  $N_{SE(3)}$  using the exponential coordinates.

If  $p(X_k, y_k|X_{k-1}^{(i)})$  is jointly Gaussian with the following mean  $\mu$  and covariance  $\Sigma$ :

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}^\top & \Sigma_{22} \end{pmatrix}, \quad (7)$$

then  $p(X_k|X_{k-1}^{(i)}, y_k)$  is given by  $N_{SE(3)}(m_k, \Sigma_k)$  with

$$\bar{u} = \Sigma_{12}(\Sigma_{22})^{-1} \cdot v_{sl(3)}(\text{logg}(r(\mu_2, y_k))), \quad (8)$$

$$m_k = \mu_1 \cdot \exp(v_{se(3)}^{-1}(\bar{u})), \quad (9)$$

$$\Sigma_k = \Sigma_{11} - \Sigma_{12}(\Sigma_{22})^{-1}\Sigma_{12}^\top. \quad (10)$$

This is the generalization of Gaussian filtering on the vector space to  $SE(3)$  using the exponential coordinates. Note that the measurement error in (8) is represented as the one on  $sl(3)$  using  $r$  and  $\text{logg}$ . In (7),  $\mu_1$  and  $\Sigma_{11}$  correspond to the mean and covariance of  $p(X_k|X_{k-1}^{(i)})$ . Since  $p(X_k|X_{k-1}^{(i)})$  is approximated as  $N_{SE(3)}(f(X_{k-1}^{(i)}), P')$ ,  $\mu_1$  and  $\Sigma_{11}$  are given by  $f(X_{k-1}^{(i)})$  and  $P'$ , respectively. We now utilize UT to estimate the remaining  $\mu_2$ ,  $\Sigma_{12}$ , and  $\Sigma_{22}$  in (7).

For a state of dimension  $n$ , it is necessary to generate  $2n+1$  sigma points [12]. Thus 13 sigma points are required for our case because the intrinsic dimension of  $SE(3)$  is 6. We generate the sigma points  $\tilde{X}_j$ , which are symmetrically distributed around  $f(X_{k-1}^{(i)})$  using the exponential coordinates as

$$\tilde{X}_0 = f(X_{k-1}^{(i)}),$$

$$\tilde{X}_j = f(X_{k-1}^{(i)}) \cdot \exp(v_{se(3)}^{-1}(\chi_j)), \quad j = 1, \dots, 6,$$

$$\tilde{X}_j = f(X_{k-1}^{(i)}) \cdot \exp(-v_{se(3)}^{-1}(\chi_{j-6})), \quad j = 7, \dots, (11)$$

where  $\chi_j$  is a  $\mathfrak{R}^6$  vector corresponding to the  $j$ -th column of the square root of  $(6+\lambda)P'$  with  $\lambda = \alpha^2(6+\kappa) - 6$ . Then the associated weights are chosen to preserve the mean  $f(X_{k-1}^{(i)})$  and the covariance  $P'$ . The weights  $W_j^m$

for the mean calculation are determined as  $W_0^m = \frac{\lambda}{(6+\lambda)}$  and  $W_j^m = \frac{1}{2(6+\lambda)}$  for  $j = 1, \dots, 12$ . The weights  $W_j^c$  for the covariance calculation are determined as  $W_0^c = W_0^m + (1 - \alpha^2 + \beta)$  and  $W_j^c = W_j^m$  for  $j = 1, \dots, 12$ . The chosen parameter values following the recommendation of [23] are  $\alpha = 0.001$ ,  $\beta = 2$ , and  $\kappa = 0$ . The sample mean of  $\tilde{X}_j$  calculated with  $W_j^m$  is  $f(X_{k-1}^{(i)})$  since  $\tilde{X}_j$  are symmetrically distributed around  $f(X_{k-1}^{(i)})$ . We can also easily verify that the sample covariance  $\tilde{X}_j$  is  $P'$  by applying the covariance formula (3) as  $\sum_{j=0}^{12} W_j^c \cdot v_{se(3)}(\log(f(X_{k-1}^{(i)})^{-1} \cdot \tilde{X}_j)) \cdot v_{se(3)}(\log(f(X_{k-1}^{(i)})^{-1} \cdot \tilde{X}_j))^\top$ .

We now obtain  $\tilde{y}_j$  by transforming  $\tilde{X}_j$  via  $g$  in (6), i.e.,  $\tilde{y}_j = g(\tilde{X}_j, L_l^{(i)}, I)$ . Then  $\mu_2$  in (7) is given by the weighted mean of  $\tilde{y}_j$  as  $\mu_2 = \sum_{j=0}^{12} W_j^m \cdot \tilde{y}_j$ .  $\Sigma_{12}$  is obtained as

$$\Sigma_{12} = \sum_{j=0}^{12} W_j^c \cdot v_{se(3)}(\log(f(X_{k-1}^{(i)})^{-1} \cdot \tilde{X}_j)) \cdot v_{sl(3)}(\text{logg}(r(\mu_2, \tilde{y}_j)))^\top, \quad (12)$$

and  $\Sigma_{22}$  is similarly obtained as

$$\Sigma_{22} = R + \sum_{j=0}^{12} W_j^c \cdot v_{sl(3)}(\text{logg}(r(\mu_2, \tilde{y}_j))) \cdot v_{sl(3)}(\text{logg}(r(\mu_2, \tilde{y}_j)))^\top, \quad (13)$$

where  $R$  is the covariance for  $n_k$  in (6). Note that the measurement error is represented as the one on  $sl(3)$  again in (12) and (13).

Since all the required  $\mu$  and  $\Sigma$  are obtained via UT, we can approximate the optimal importance function as  $N_{SE(3)}(m_k, \Sigma_k)$  via (8), (9), and (10). The particles are then sampled from  $N_{SE(3)}(m_k, \Sigma_k)$ , i.e.,  $X_k^{(i)} \sim N_{SE(3)}(m_k, \Sigma_k)$ . When there are multiple measurements from multiple landmarks, we treat them as a single measurement by concatenating them in a column vector; correspondingly,  $R$  in (13) becomes a block diagonal matrix. Unlike [23], we regard  $X_k^{(i)}$  as not having their own uncertainties in order to use the smaller number of sigma points for efficiency. We also consider  $n_k$  as the additive term in (13) for the same purpose.

#### 4.2. Landmark estimation

The procedure to estimate the landmark pose using UT is similar to the optimal importance sampling of  $X_k^{(i)}$ ; the sigma points for the plane normal are generated using  $exp$  and the measurement error is represented as the one on  $sl(3)$  again. For  $L_l^{(i)}$ , 7 sigma points are required since the number of uncertain components of  $L_l^{(i)}$  is 3. Denoting a  $\mathfrak{R}^3$  vector corresponding to the  $j$ -th column of the square

root of  $(3 + \lambda)\Sigma_{L_l}^{(i)}$  by  $(\chi_j^x, \chi_j^y, \chi_j^z)^\top$ , the sigma points  $\tilde{L}_j = \{\tilde{s}_j, \tilde{N}_j\}$  are generated as

$$\begin{aligned}\tilde{s}_0 &= s_l^{(i)}, \\ \tilde{s}_j &= s_l^{(i)} + (0, 0, 0, 0, 0, \chi_j^x)^\top, j = 1, 2, 3, \\ \tilde{s}_j &= s_l^{(i)} - (0, 0, 0, 0, 0, \chi_{j-3}^x)^\top, j = 4, 5, 6, \\ \tilde{N}_0 &= N_l^{(i)}, \\ \tilde{N}_j &= N_l^{(i)} \cdot \text{exp}(v_{so(3)}^{-1}((\chi_j^x, \chi_j^y, 0)^\top)), j = 1, 2, 3, \\ \tilde{N}_j &= N_l^{(i)} \cdot \text{exp}(-v_{so(3)}^{-1}((\chi_{j-3}^x, \chi_{j-3}^y, 0)^\top)), j = 4, 5, 6.\end{aligned}$$

The weights  $\bar{W}_j^m$  and  $\bar{W}_j^c$  are determined similarly as before, *i.e.*,  $\bar{W}_0^m = \frac{\lambda}{(3+\lambda)}$ ,  $\bar{W}_0^c = \bar{W}_0^m + (1 - \alpha^2 + \beta)$ , and  $\bar{W}_j^m = \bar{W}_j^c = \frac{1}{2(3+\lambda)}$  for  $j = 1, \dots, 6$ . The parameter values are the same as in the optimal importance sampling.

The next step is to predict  $\bar{y}_k$  from  $L_l^{(i)}$  and the sampled  $X_k^{(i)}$  using the sigma points  $\tilde{L}_j$ . Here, the important step is to calculate anew the four 3-D corner points for each  $\tilde{L}_j$ .  $\bar{y}_k$  is determined as  $\bar{y}_k = \sum_{j=0}^6 \bar{W}_j^m \cdot \tilde{z}_j$  where  $\tilde{z}_j = g(X_k^{(i)}, \tilde{L}_j, I)$  with the newly calculated 3-D corner points of  $\tilde{L}_j$ . The cross-covariance is obtained as

$$\Sigma_{Lz} = \sum_{j=0}^6 \bar{W}_j^c \cdot (d_j^x, d_j^y, d_j^z)^\top \cdot v_{sl(3)}(\text{logg}(r(\bar{y}_k, \tilde{z}_j)))^\top, \quad (15)$$

where  $d_j^x$  is the 6-th element of  $\tilde{s}_j - s_l^{(i)}$ , and  $d_j^y$  and  $d_j^z$  are the first and second elements of  $v_{so(3)}(\text{logg}((N_l^{(i)})^{-1} \cdot \tilde{N}_j))$ .  $\Sigma_{zz}$  is also obtained with  $v_{sl(3)}(\text{logg}(r(\bar{y}_k, \tilde{z}_j)))$  and  $\bar{W}_j^c$  similarly to (13). Then  $L_l^{(i)}$  and  $\Sigma_{L_l}^{(i)}$  are updated with  $y_k$  as

$$\bar{u}_L = \Sigma_{Lz}(\Sigma_{zz})^{-1} \cdot v_{sl(3)}(\text{logg}(r(\bar{y}_k, y_k))), \quad (16)$$

$$s_l^{(i)} \leftarrow s_l^{(i)} + (0, 0, 0, 0, 0, \bar{u}_L(1))^\top, \quad (17)$$

$$N_l^{(i)} \leftarrow N_l^{(i)} \cdot \text{exp}(v_{so(3)}^{-1}((\bar{u}_L(2), \bar{u}_L(3), 0)^\top)), \quad (18)$$

$$\Sigma_{L_l}^{(i)} \leftarrow \Sigma_{L_l}^{(i)} - \Sigma_{Lz}(\Sigma_{zz})^{-1} \Sigma_{Lz}^\top, \quad (19)$$

where  $\bar{u}_L(k)$  represents the  $k$ -th component of  $\bar{u}_L$ .

The importance weights for  $S_k^{(i)} = \{X_k^{(i)}, L_l^{(i)}, \Sigma_{L_l}^{(i)}\}$  are calculated from  $N_{SL(3)}(r(\bar{y}_k, y_k); I, \Sigma_t)$  where  $\Sigma_t = \Sigma_{12}^\top \Sigma_k^{-1} \Sigma_{12} + \Sigma_{zz}$ . As the case of the optimal importance sampling, the multiple measurements are treated as a single measurement in the importance weight calculation. That is,  $v_{sl(3)}(\text{logg}(r(\bar{y}_k, y_k)))$  for each landmark is concatenated in a column vector, and each  $\Sigma_{zz}$  is placed in the block diagonal matrix accordingly. Finally, the optimal state estimation  $\bar{S}_k$  is obtained by the sample mean of particles after resampling. For  $L_l^{(i)}$ ,  $\bar{s}_l$  is given by the ordinary arithmetic mean while  $\bar{N}_l$  is obtained by the sample mean of  $SO(3)$ .  $\bar{X}_k$  is also obtained by the sample mean of  $SE(3)$ .

Now monocular SLAM with locally planar landmarks can be solved via the geometric RBPF on Lie groups derived so far. In practice,  $y_k$  is provided by the homography tracking using the rectangular region, which is designated when the landmark is initialized, as the template. It can be understood that the data association step is included in the homography tracking. The strategies for the interest point detection and the homography tracking are detailed in the following subsections.

### 4.3. Interest point detection

If the detected interest point is actually on the non-planar regions, *e.g.*, edges and vertexes of a box, the homography tracking providing  $y_k$  may fail owing to the large camera viewpoint change. In addition, the homography tracking may also fail if there are insufficient features in the template. Therefore, it is required that the interest points on the actual planar regions with enough distinct features be detected. To fulfill this requirement, we use the MSER blobs [17] in conjunction with the Harris corner points [10]. We consider the MSER blobs as possessing high probability to be placed on the actual planar regions, and the Harris corner points are regarded as distinct features for tracking.

We first generate the probability map  $M_1$  for the video frame by Gaussian smoothing of the binary image representing the center points of the MSER blobs. Similarly, we also generate the probability map  $M_2$  for the Harris corner points. The final probability map  $M_f$  is obtained by multiplying  $M_1$  and  $M_2$  pixel-wise. We then extract the candidate interest point set  $C_p$  from  $M_f$  by non-maxima suppression, using the mean of  $M_f$  as the threshold. The interest point selection from  $C_p$  is started with the empty interest point set  $I_p$ . Among  $C_p$ , the point with the highest score of  $M_f$  is selected as the element of  $I_p$ . After eliminating the points of  $C_p$  in some neighborhood of  $I_p$ , the interest point selection from the remaining  $C_p$  is repeated.

In practice, we set the minimal number of landmarks in a frame to be  $M_L$ , and first initialize  $M_L$  landmarks at the initial frame. When the number of predicted projections in a frame is less than  $M_L$ , the required number of landmarks are initialized anew to maintain the number of landmarks in a frame as  $M_L$ . In this case, the projections from the landmark positions are used as the initial  $I_p$ , instead of the empty set, when selecting the interest points from  $C_p$ .

### 4.4. Homography tracking

We use the inverse-compositional (IC) tracker of [1] for the homography tracking to provide  $y_k$  of each landmark. The objective function to be maximized is the normalized cross-correlation (NCC) with the landmark template. Though the IC tracker is very fast, its limitation is that it can be trapped in local optima especially when the camera motion between adjacent frames is relatively large.

Thus, we use the *particle swarm optimization* (PSO) algorithm [20] before the IC tracker to determine the starting position near the global optimum for the IC tracker. In [25], PSO has been utilized for visual tracking via multi-layer importance sampling. Unlike [25], we run the PSO algorithm with few iterations since the goal is simply to determine the starting position near the global optimum for the IC tracker. Since the optimization space is  $SL(3)$ , we formulate the PSO algorithm on  $SL(3)$  using *expp* and *logg*.

Given  $\bar{X}_{k-1}$  and  $\bar{L}_l$ , which are the estimated states at time  $k - 1$ , the homography tracking at time  $k$  is initiated with  $\bar{y} = g(f(\bar{X}_{k-1}), \bar{L}_l, I)$ . The goal is to find  $H_k$  to give the best NCC score for the image region represented by transforming  $\bar{y}$  by  $H_k$ . The particles  $H_i \in SL(3)$  are sampled from  $N_{SL(3)}(I, \Sigma_H)$ , and their velocities  $V_i \in sl(3)$  are sampled from Gaussian on  $sl(3)$  with the zero mean and  $\Sigma_H$ . We next identify the individual best  $H_{i,ind}$  of each  $H_i$  and the global best  $H_{glo}$  of all particles based on the NCC scores of the image regions represented by transforming  $\bar{y}$  by  $H_i$ . Then  $H_i$  and  $V_i$  are updated as

$$V_i \leftarrow \omega[V_i + U(0, \phi_1) \otimes v_{sl(3)}(\text{logg}(H_i^{-1} \cdot H_{i,ind}) + U(0, \phi_2) \otimes v_{sl(3)}(\text{logg}(H_i^{-1} \cdot H_{glo})))] \quad (20)$$

$$H_i \leftarrow H_i \cdot \text{expp}(v_{sl(3)}^{-1}(V_i)), \quad (21)$$

where  $U(0, \phi_i) \in \mathbb{R}^8$  is a uniform random vector between 0 and  $\phi_i$ , and  $\otimes$  represents the component-wise multiplication. Since the smaller value of  $\omega$  represents the faster convergence to  $H_{glo}$ , we set  $\omega$  to 0.3, which is rather small, with  $\phi_1 = \phi_2 = 1$ . The update procedure is iterated with the changes of  $H_{i,ind}$  and  $H_{glo}$  according to the NCC scores.

$v_{sl(3)}(\text{logg}(H_i^{-1} \cdot H_{i,ind}))$  and  $v_{sl(3)}(\text{logg}(H_i^{-1} \cdot H_{glo}))$  in (20) can be understood as the directional velocities from  $H_i$  to  $H_{i,ind}$  and  $H_{glo}$ . The ability of PSO to find the global optimum avoiding being trapped in local optima stems from the stochasticity of the update process represented by the uniform random vectors in (20). To increase the possibility of finding the global optimum within the limited iterations, we employ the quantum particles  $Q_i$ . At every iteration,  $Q_i$  are generated around  $H_{glo}$  as  $Q_i = H_{glo} \cdot \text{expp}(v_{sl(3)}^{-1}(U(0, \phi_3)))$ , and used to find the new global best, which is better than the current  $H_{glo}$ .

After running the PSO on  $SL(3)$ , the IC tracker with Levenberg-Marquardt is started at  $H_{glo}$ . The IC tracker on  $SL(3)$  with NCC can be implemented by consulting [1, 3, 5]. Then  $y_k$  is determined by transforming  $\bar{y}$  by  $H_k$ , which is the final output of the IC tracker. If the point on the non-planar region is incorrectly selected in spite of the procedure in Section 4.3, the tracking may fail even with the proposed method. When the tracking failure occurs, we use  $H_{glo}$  as the tracking result. The criterion for tracking failure is the too-large homography update in the IC tracker iteration, which usually occurs when the IC tracker diverges.

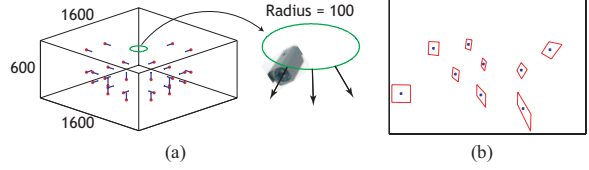


Figure 3. (a) The simulation environment and the camera trajectory. Blue lines represent the landmark plane normals. (b) Red quadrilaterals represent  $y_k$  of each landmark while blue dots represent the landmark position projections.

## 5. Experiments

### 5.1. Experiment 1: simulated data

We first show the superiority of our SLAM framework to conventional SLAM with point landmarks via simulation experiments<sup>2</sup>. Figure 3 shows the simulation environment, the camera trajectory, and the measurements. The camera moves along the circle and returns to its initial position closing a loop. For  $n_k$ , we use  $N_{SL(3)}(I, R)$  with  $R = \text{diag}(1^2, 1^2, 0.01^2, 0.01^2, 0.01^2, 0.001^2, 0.0005^2, 0.0005^2)$  in the order of  $E_i$  in [3]. For the measurement of conventional SLAM with point landmarks, we use the landmark position projections shown in Fig. 3(b). The additive Gaussian noise with a covariance of  $\text{diag}(1^2, 1^2)$  is also added to the landmark position projections. For fair comparison, we also represent the camera state as  $SE(3)$  for conventional SLAM with point landmarks, and use our geometric UT for optimal importance sampling. The landmark estimation for conventional SLAM is done via EKF since the Jacobian of the camera projection function with respect to the point landmark is trivially obtained. We run our SLAM with 100 particles while 500 particles are used for conventional SLAM with point landmarks to balance the computational complexity. The state covariance  $P$  and the initial value of  $\rho$  are set to be the same for both cases.

Figure 4 shows the SLAM results by one typical run of both cases. For meaningful error comparison, we scaled the estimated camera trajectory and the landmark positions using the average distance between the landmarks and the center of the estimated camera trajectory because monocular SLAM results usually settle on a meaningless scale.

From Fig. 4(a) and Fig. 4(b), we can clearly see that both camera and landmark poses are quite accurately estimated by our SLAM framework with only 100 particles; please compare the estimated plane normals in Fig. 4(b) with the true plane normals in Fig. 3(a). Although many more particles are used for conventional SLAM with point landmarks, its SLAM results shown in Fig. 4(a) and Fig. 4(c) are unsatisfactory compared with the results of our SLAM frame-

<sup>2</sup>All the figures in Section 5 are best viewed in color. The video containing all the experimental results shown in Section 5 is available at <http://cv.snu.ac.kr/jhkwon/slam/>.

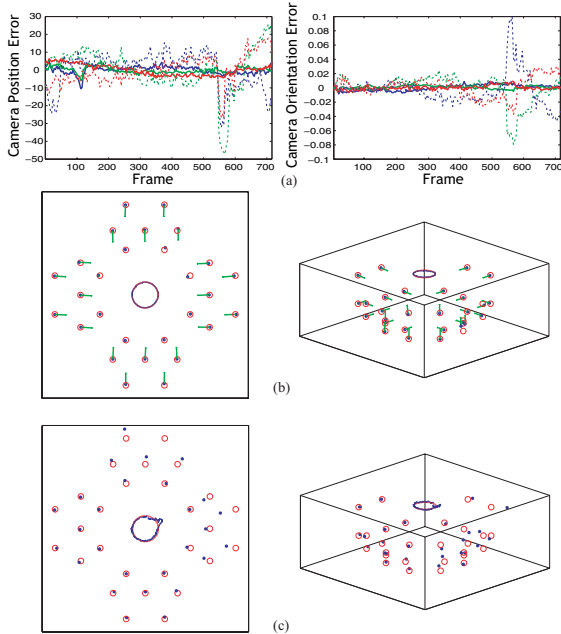


Figure 4. SLAM results with the simulated data. The camera pose estimation errors are shown in (a) (solid: our SLAM framework, dotted: conventional SLAM with point landmarks). The estimated camera trajectory and landmark positions by our SLAM framework and conventional SLAM with point landmarks are shown in (b) and (c), respectively (red: ground-truth, blue: estimation results). The estimated landmark plane normals by our SLAM framework are also shown in (b) in green lines.

Estimation error	Our framework	Conventional
Camera position	4.45	16.43
Camera orientation	0.0063	0.027
Landmark position	11.48	41.78

Table 1. The estimation errors for SLAM results with the simulated data. The errors for the camera and the landmarks are averaged over the number of frames and landmarks, respectively.

work. Table. 1 shows the estimation errors for the camera pose and landmark positions averaged over 10 independent runs. From these results, it can be said that the accurately estimated landmark pose helps the accurate camera pose estimation and vice versa in our SLAM framework.

## 5.2. Experiment 2: real sequences

We now demonstrate the feasibility of our proposed framework via the experiments with real sequences. For all the test sequences, we use 100 particles for the geometric RBPF on Lie groups, and 30 particles including 10 quantum particles for the PSO on  $SL(3)$  with 10 iterations. The state covariance  $P$  and the initial value of  $\rho$  are set to be the same for all the sequences.

The first experiment with the “Desk” sequence is in-

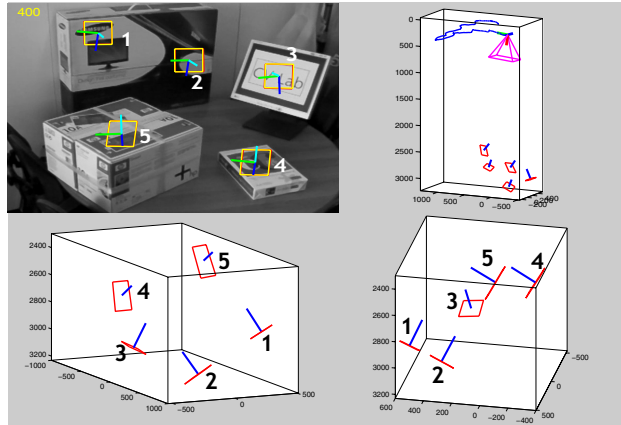


Figure 5. The SLAM result with the “Desk” sequence. The landmark normals are overlaid on the frame. The red and yellow quadrilaterals on the frame are the measurements and the projections from the estimated landmarks, respectively. The estimated landmark maps viewed from different directions are shown in the bottom row.

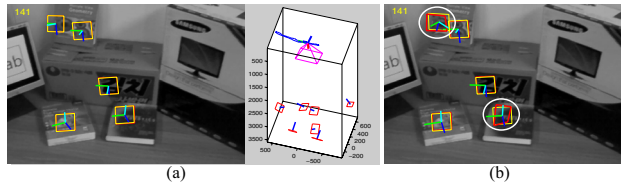


Figure 6. The SLAM result with the “Lab” sequence is shown in (a). When we employ only the IC tracker without the PSO on  $SL(3)$ , the tracking failure frequently occurs as shown in (b) in white circles.

tended to show the accuracy of our SLAM framework since it contains various plane normals whose ground truth can be easily inferred from the scene. The SLAM result with the “Desk” sequence is shown in Fig. 5. The estimated landmark plane normals overlaid on the frame seem to be almost the same as the ground truth inferred from the scene. The accuracy of the landmark pose estimation can be more easily verified from the bottom row of Fig. 5. From the landmark maps viewed from different directions, we can identify the spatial relationships of the numbered landmarks. Landmark 1 and 2 are on the same plane, and Landmark 4 and 5 have the same plane normals with different positions. These spatial relationships exactly correspond to the ground truth inferred from the video sequence. From the supplementary video, we can see that the planar landmark poses are gradually corrected over the course of time. The camera trajectory in the video also seems to be accurate.

The second experiment is with the “Lab” sequence, where the camera motion is somewhat fast at the latter part. When only the IC tracker is employed for the homography tracking, the tracking failure is unavoidable when the camera motion is fast as shown in Fig. 6(b). In contrast, by

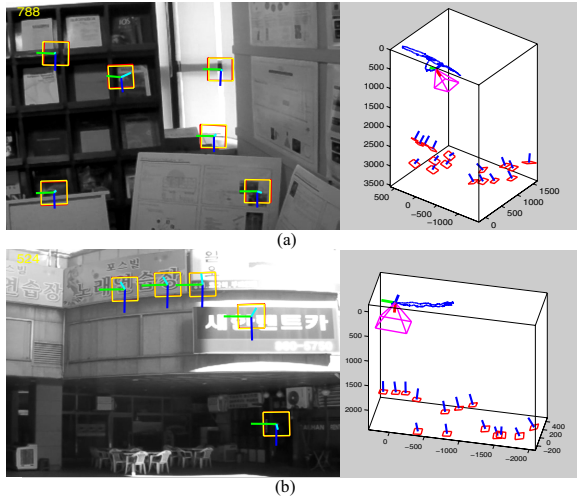


Figure 7. The SLAM results with the (a) “Office” and (b) “Outdoor” sequences.

the use of PSO on  $SL(3)$  in conjunction with the IC tracker as proposed in Section 4.4, our SLAM framework successfully works in spite of the relatively fast camera motion as shown in Fig. 6(a) and the supplementary video. We can see that new landmarks are added to the map as time goes by following the procedure described in Section 4.3.

The third and fourth experiments are with the “Office” and “Outdoor” sequences, and the results are shown in Fig. 7. From the supplementary video, we can see that the estimated landmark poses and camera pose are quite accurate for both sequences. There are 24 landmarks in a map for the “Office” sequence. Though the distances from the camera to the objects in the “Outdoor” sequence are much longer than those in the other sequences, the SLAM result for the “Outdoor” sequence is quite satisfactory even with the same initial value of  $\rho$  and the same state covariance  $P$  as the other sequences. From the supplementary video, we can clearly see that the interest points are detected on the regions with sufficient features for the “Outdoor” sequence.

Our current implementation in C++ takes about 0.45 seconds per frame with 100 particles and 8 landmarks in a scene on an Intel Core-2 Quad 2.4 GHz processor. The homography tracking occupies about 30% of computation time, and the rest is for geometric RBPF on Lie groups. We expect that further speedup up to 15fps is attainable via code optimization with parallel computing.

## 6. Conclusions

We have proposed a geometric RBPF framework for monocular SLAM with locally planar landmarks. We have defined the camera state and the landmark state for plane normal as  $SE(3)$  and  $SO(3)$ , respectively. The state uncertainties have been defined as Gaussian distributions on

$SE(3)$  and  $SO(3)$  while the measurement uncertainty has been defined as Gaussian on  $SL(3)$ . For optimal importance sampling and landmark pose estimation in an RBPF framework, we have formulated UT on Lie groups in accordance with their geometric properties. The feasibility of our framework has been effectively shown via various experiments.

## References

- [1] S. Baker and I. Matthews. Lucaskanade 20 years on: a unifying framework. *Int. J. Comput. Vision*, 56(3):221–255, 2004.
- [2] E. Beltracchi and M. Werman. How to put probabilities on homographies. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10), 2005.
- [3] S. Benhimane and E. Malis. Homography-based 2d visual tracking and servoing. *Int. J. Rob. Res.*, 26(7):661–676, 2007.
- [4] C. Berger and S. Lacroix. Using planar facets for stereovision slam. In *IROS 2008*.
- [5] R. Brooks and T. Arbel. Generalizing inverse composition and esm image alignment. *Int. J. Comput. Vision*, Online first published.
- [6] J. Civera, A. Davison, and J. Montiel. Inverse depth parametrization for monocular slam. *IEEE Trans. Robotics*, 24(5):932–945, 2008.
- [7] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In *ICCV 2003*.
- [8] E. Eade and T. Drummond. Scalable monocular slam. In *CVPR 2006*.
- [9] A. Gee, D. Chekhlov, A. Calway, and W. Mayol-Cuevas. Discovering higher level structure in visual slam. *IEEE Trans. Robotics*, 24(5):980–990, 2008.
- [10] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. of the 4th Alvey Vision Conference*, 1988.
- [11] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge, 2000.
- [12] S. Julier and J. Uhlmann. Unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(3):401–422, 2004.
- [13] C. Kim, R. Sakthivel, and W. Chung. Unscented fastslam: a robust and efficient solution to the slam problem. *IEEE Trans. Robotics*, 24(4):808–820, 2008.
- [14] J. Kwon, M. Choi, F. Park, and C. Chun. Particle filtering on the euclidean group: framework and applications. *Robotica*, 25(6):725–737, 2007.
- [15] J. Kwon, K. Lee, and F. Park. Visual tracking via geometric particle filtering on the affine group with optimal importance functions. In *CVPR 2009*.
- [16] J. Martinez-Carranza and A. Calway. Appearance based extraction of planar planar structure in monocular slam. In *SCIA 2009*.
- [17] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC 2002*.
- [18] N. Molton, A. Davison, and I. Reid. Locally planar patch features for real-time structure from motion. In *BMVC 2004*.
- [19] T. Pietzsch. Planar features for visual slam. In *KI 2008*.
- [20] R. Poli, J. Kennedy, and T. Blackwell. Particle swarm optimization: an overview. *Swarm Intelligence*, 1(1):33–57, 2007.
- [21] G. Silveira, E. Malis, and P. Rives. An efficient direct approach to visual slam. *IEEE Trans. Robotics*, 24(5):969–979, 2008.
- [22] R. Sim, P. Elinas, and J. Little. A study of the rao-blackwellised particle filter for efficient and accurate vision-based slam. *Int. J. Comput. Vision*, 74(3):303–318, 2007.
- [23] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. In *NIPS 2000*.
- [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] X. Zhang, W. H. amd S.J. Maybank, X. Li, and M. Zhu. Sequential particle swarm optimization for visual tracking. In *CVPR 2008*.