2009

Vehicle tracking using projective particle filter

Azeddine Beghdadi

Philippe Bouttefroy
philippe@uow.edu.au

Son Lam Phung
University of Wollongong, phung@uow.edu.au

Abdesselam Bouzerdoum
University of Wollongong, bouzer@uow.edu.au

Publication Details

Bouttefroy, P., Bouzerdoum, A., Phung, S. & Beghdadi, A. (2009). Vehicle tracking using projective particle filter. In L. O'Conner (Eds.), Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance (pp. 7-12). Genova, Italy: IEEE.

Vehicle tracking using projective particle filter

Abstract
This article introduces a new particle filtering approach for object tracking in video sequences. The projective particle filter uses a linear fractional transformation, which projects the trajectory of an object from the real world onto the camera plane, thus providing a better estimate of the object position. In the proposed particle filter, samples are drawn from an importance density integrating the linear fractional transformation. This provides a better coverage of the feature space and yields a finer estimate of the posterior density. Experiments conducted on traffic video surveillance sequences show that the variance of the estimated trajectory is reduced, resulting in more robust tracking.

Keywords
vehicle, tracking, filter, projective, particle

Disciplines
Physical Sciences and Mathematics

Publication Details
Bouttefroy, P., Bouzerdoum, A., Phung, S. & Beghdadi, A. (2009). Vehicle tracking using projective particle filter. In L. O'Conner (Eds.), Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance (pp. 7-12). Genova, Italy: IEEE.

Vehicle Tracking using Projective Particle Filter

P.L.M. Bouttefroy, A. Bouzerdoum, S.L. Phung
SECTE, Faculty of informatics,
Wollongong University,
Northfields Av., Wollongong, 2522 NSW, Australia.
{philippe,a.bouzerdoum,phung}@uow.edu.au

A. Beghdadi
L2TI, Institut Galilée, Univ. Paris 13,
99, avenue J. B. Clément,
93430 Villetaneuse, France.
beghdadi@univ-paris13.fr

Abstract

This article introduces a new particle filtering approach for object tracking in video sequences. The projective particle filter *uses a linear fractional transformation, which projects the trajectory of an object from the real world onto the camera plane, thus providing a better estimate of the object position. In the proposed particle filter, samples are drawn from an importance density integrating the linear fractional transformation. This provides a better coverage of the feature space and yields a finer estimate of the posterior density. Experiments conducted on traffic video surveillance sequences show that the variance of the estimated trajectory is reduced, resulting in more robust tracking.*

1. Introduction

Vehicle tracking has been an active field of research within the past decade due to the increase in computational power and the development of video surveillance infrastructure. There is, today, a huge need for automatic traffic control and regulation, automatic video surveillance and abnormal event detection. Robust car tracking is a fundamental low-level task necessary to achieve such intelligence. There have been various techniques developed to track vehicles. The most common ones undoubtedly rely on Bayesian filtering and, in particular, Kalman and particle filters. Kalman filter based tracking usually relies on the kinematic variables and size of the vehicles estimated via background subtraction followed by segmentation [7, 13], although some techniques implement spatial features such as corners and edges [11, 14] or Bayesian energy minimization [5]. Particle filtering is preferred when the hypothesis of multimodality is necessary, *e.g.* in case of severe occlusion [12, 17]. Exhaustive search techniques involving template matching [3] or occlusion reasoning [9] have also been introduced to track vehicles.

However, vehicle tracking techniques seldom exploit the characteristics of video surveillance sequences, namely:

- *Slowly-varying vehicle Speed*—since vehicles appear in the field of view of the camera for a short while only, their speed is quasi uniform;
- *Constrained vehicle trajectory*—the position of vehicle is constrained by the curvature of the road and the different lanes; and
- *Projection of vehicle trajectory on the camera plane*—the trajectory of the vehicle on the camera plane undergoes severe distortion due to the low elevation of the traffic surveillance camera.

We propose here to integrate these characteristics to obtain a finer estimate of the vehicle feature vector. More specifically, the mapping of real-world vehicle trajectory through linear fractional transformation enables a better estimate of the posterior density. A particle filter is thus implemented, integrating cues on the projection in the importance density, resulting in a better exploration of the state space and a reduction of the variance in the trajectory estimation. The rest of the paper is organized as follows. Section 2 introduces the general particle filtering framework. Section 3 develops the Projective Particle Filter (PPF). In particular, Section 3.1 derives the linear fractional transformation. An analysis of the PPF performance versus the standard particle filter is presented in Section 4 before concluding in Section 5.

2. Bayesian Filtering and Particle Filtering

Bayesian filtering provides a convenient framework for object tracking due to the weak assumptions on the state space model and the first order Markov chain recursive properties. Without loss of generality, let us consider a system with state $\mathbf{x}$ of dimension n and observation $\mathbf{z}$ of dimension m. Let us denote the set of states $\mathbf{x}_{1:k} \triangleq \{\mathbf{x}_1, ..., \mathbf{x}_k\}$

978-0-7695-3718-4/09 $25.00 © 2009 IEEE
DOI 10.1109/AVSS.2009.60

and the set of observations $\mathbf{z}_{1:k} \triangleq \{\mathbf{z}_1, ..., \mathbf{z}_k\}$, where k is a time index. The state space model can be expressed as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1}\,, \tag{1}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{n}_k\,, \tag{2}$$

when the process and observation noises, $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ respectively, are assumed to be additive. The vector-valued functions $\mathbf{f}$ and $\mathbf{h}$ are the process and observation functions, respectively. Bayesian filtering aims to estimate the distribution of the state $\mathbf{x}$ from the observation $\mathbf{z}$ as $p(\mathbf{x}_k|\mathbf{z}_k)$. The probability density function (pdf) is estimated recursively.

Monte Carlo methods and more specifically particle filters, based on Bayesian inference, have been extensively employed for tracking problems [6, 15]. Multi-modality, in particular, enables the system to evolve in time with several hypotheses on the state in parallel. This property is practical to corroborate or reject an eventual track after several frames. Particle filters rely on Sequential Monte Carlo (SMC) methods. A large number of samples $\{\mathbf{x}_k^i, i = 1..N_S\}$ are drawn from the posterior distribution $p(\mathbf{x}_k|\mathbf{z}_k)$. It follows from the law of large numbers that

$$p(\mathbf{x}_k|\mathbf{z}_k) \approx \sum_{i=1}^{N_S} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i)\,, \tag{3}$$

where w_k^i are the weights, *i.e.* $\sum w_k^i = 1$, and $\delta(.)$ is the Kronecker delta function. However, because it is often difficult to draw samples from the posterior pdf, an importance density $q(.)$ is used to generate the samples $\mathbf{x}_k^i$. It can then be shown that [1]

$$w_k^i \propto w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)\, p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)}\,. \tag{4}$$

The choice of the importance density is crucial to obtaining a good estimate of the posterior pdf. It has been shown that the set of particles and associated weights $\{\mathbf{x}_k^i, w_k^i\}$ will eventually degenerate, *i.e.* most of the weights will be carried by a small number of samples and a large number of samples will have negligible weight [10]. In such a case, and because samples are not drawn from the true posterior, the degeneracy problem cannot be avoided and resampling of the set needs to be performed. Nevertheless, the closer the importance density is from the true posterior density, the slower the set $\{\mathbf{x}_k^i, w_k^i\}$ will degenerate; a good choice of importance density reduces the need for resampling. In this paper, we propose to model the fractional transformation mapping the real world space onto the camera plane and to integrate the projection through the importance density $q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ in the particle filter.

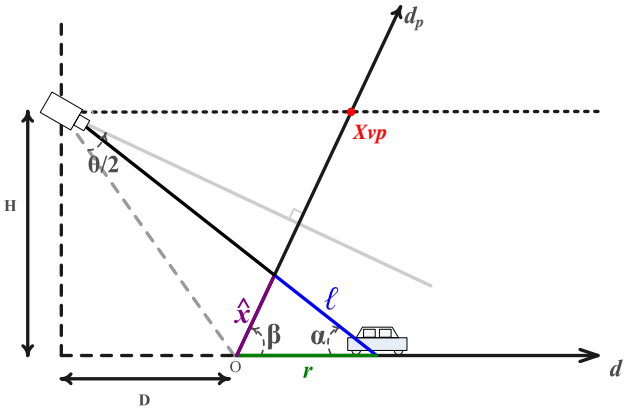


Figure 1. Projection of the vehicle on a plane parallel to the image plane of the camera. The graph shows a cross section of the scene along the direction d (tangential to the road).

3. Projective Particle Filter

The particle filter developed is named Projective Particle Filter (PPF) because the vehicle position is projected on the camera plane and used as an inference to diffuse the particles in the feature space. One of the particularities of the PPF is to differentiate between the importance density and the transition prior pdf whilst the SIR (Sampling Importance Resampling) filter, also called standard particle filter, does not. Therefore, we need to define the importance density from the fractional transformation as well as the transition prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and the likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$ in order to update the weights in Eq. (4).

3.1. Linear Fractional Transformation

The fractional transformation is used to estimate the position of the object on the camera plane (x) from its position on the road (r). The physical trajectory is projected onto the camera plane as shown in Fig. 1. The distortion of the object trajectory happens along the direction d, tangential to the road. The axis d_p is parallel to the camera plane; the projection $\hat{x}$ of the vehicle position on d_p is thus proportional to the position of the vehicle on the camera plane. The value X_{vp}, projection of the vanishing point on d_p, scales $\hat{x}$ to obtain the position of the vehicle in terms of pixels. For practical implementation, it is useful to express the projection along the tangential direction d onto the d_p axis in terms of video footage parameters that are easily accessible, namely:

- Angle of view (θ);
- Height of the camera (H); and
- Ground distance (D) between the camera and the first location captured by the camera.

It can be inferred from Fig. 1, after applying Al-Kashi theorem, that:

$$\hat{x}^2 = r^2 + \ell^2 - 2r\ell\cos(\alpha)\,, \tag{5}$$

and

$$\ell^2 = \hat{x}^2 + r^2 - 2r\hat{x}\cos(\beta)\,, \tag{6}$$

where $\cos(\alpha) = (D+r)/\sqrt{(H^2+(D+r)^2)}$ and $\beta = \arctan(D/H) + \theta/2$. After squaring and substituting ℓ^2 in (5), we obtain:

$$(\hat{x}^2 + r^2 - 2r\hat{x}\cos(\beta))\cos^2(\alpha)r^2 = \left(r^2 - r\hat{x}\cos(\beta)\right)^2 . \tag{7}$$

Grouping the terms in $\hat{x}$ to get a quadratic form leads to

$$\hat{x}^2(\cos^2(\alpha) - \cos^2(\beta)) + 2\hat{x}r\cos(\beta)(1-\cos^2(\alpha)) + r^2(\cos^2(\alpha) - 1) = 0\,. \tag{8}$$

After discarding the non physically acceptable solution, one gets

$$\hat{x}(r) = \frac{rH}{(D+r)\sin\beta + H\cos\beta}\,. \tag{9}$$

However, because $D \gg H$ and θ is small in practice (see Table 1), the angle β is approximately equal to $\pi/2$ and, consequently, Eq. (9) simplifies to $\hat{x} = rH/(D+r)$. Note that this result can be verified using Thales's theorem. Finally, we scale $\hat{x}$ with the position of the vanishing point X_{vp} in the image to find the position of the vehicle in terms of pixel location[1], which yields

$$x = \frac{X_{vp}}{\lim\limits_{r\to\infty}\hat{x}(r)}\,\hat{x}(r) = \frac{X_{vp}}{H}\,\hat{x}(r)\,. \tag{10}$$

The projected speed and the observed size of the object onto the camera plane are also important variables for the problem of tracking and are thus necessary to derive. They can be directly extrapolated from the position of the object in the camera plane. The observed speed of the vehicle $\dot{x}$ is

$$\dot{x} = f_{\dot{x}}(x) = \frac{(H-x)^2\,v}{H(D-v)+xv}\,, \tag{11}$$

where v is the parameter representing the real speed (assumed constant) of the object. The observed size of the vehicle b can also be derived from the position x if the real size of the vehicle s is known:

$$b = f_b(x) = \frac{sD}{(\frac{HD}{H-x})^2 - (\frac{s}{2})^2}\,. \tag{12}$$

[1]The position of the vanishing point can be approximated either manually or automatically [16]. For the experiment purpose, we manually estimated the vanishing point.

3.2. Importance Density and Transition Prior

The projective particle filter integrates the linear fractional transformation into the importance density $q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$. The state vector is modeled with the position, the speed and the size of the vehicle in the image such as $\mathbf{x} = \{x, y, \dot{x}, \dot{y}, b\}^T$, where x and y are the Cartesian coordinates of the vehicle, $\dot{x}$ and $\dot{y}$ are the respective speeds and b is the apparent size of the vehicle. Object tracking is traditionally performed using a standard kinematic model (Newton's Laws), taking into account the position, the speed and the size of the object[2]. In this paper, the kinematic model is refined with the estimation of the speed and the object size via linear fractional transformation on the distorted direction d. Let us define the vector-valued function $\mathbf{f}$ as

$$\begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \\ b_k \end{bmatrix} = \mathbf{f}(\mathbf{x}_{k-1}) = \begin{bmatrix} x_{k-1} + f_{\dot{x}}(x_{k-1}) \\ y_{k-1} + \dot{y}_{k-1} \\ f_{\dot{x}}(x_{k-1}) \\ \dot{y}_{k-1} \\ f_b(x_{k-1}) \end{bmatrix}\,. \tag{13}$$

It is important to note that since the linear fractional transformation is along the x axis, the distortion is severe and the function $f_{\dot{x}}$ provides a better estimate than a simple kinematic model taking into account the speed of the vehicle. On the other hand, the distortion along the y axis is much weaker and such an estimation is not necessary. The novelty in this paper resides in the estimation of the vehicle position along the x axis and its size through $f_{\dot{x}}$ and $f_b(x)$, respectively. It is worthwhile noting that the standard kinematic model of the vehicle is recovered when $f_{\dot{x}}(x_{k-1}) = \dot{x}_{k-1}$ and $f_b(x) = b_{k-1}$. The vector-valued function $\mathbf{g}(\mathbf{x}_{k-1}) = \{\mathbf{f}(\mathbf{x}_{k-1}) | f_{\dot{x}}(x_{k-1}) = \dot{x}_{k-1}, f_b(x) = b_{k-1}\}$ denotes the standard kinematic model in the sequel. The samples are drawn from the importance density $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k) = \mathcal{N}(\mathbf{x}_k, \mathbf{f}(\mathbf{x}_{k-1}), \Sigma_q)$ and the standard kinematic model is used in the prior distribution $p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_k, \mathbf{g}(\mathbf{x}_{k-1}), \Sigma_p)$, where $\mathcal{N}(., \boldsymbol{\mu}, \Sigma)$ denotes the normal distribution of covariance matrix Σ centered on $\boldsymbol{\mu}$. The distributions are considered Gaussian and isotropic to evenly spread the samples around the estimated state vector at time step k.

3.3. Likelihood Estimation

The estimation of the likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is based on the distance between color histograms as in Comaniciu *et al.* [4]. Let us define an M-bin histogram $H = \{H[u]\}_{u=1..M}$, representing the distribution of J color pixel values $\mathbf{c}$, as follows:

[2]The size of the object is essentially maintained for the purpose of likelihood estimation.

$$H[u] = \frac{1}{J}\sum_{i=1}^{J}\delta[\kappa(\mathbf{c}^i) - u]\,, \tag{14}$$

where u is the set of bins regularly spaced on the interval $[1, M]$, κ is a linear binning function providing the bin index of pixel value $\mathbf{c}^i$, and $\delta(.)$ is the Kronecker delta function. The pixels $\mathbf{c}^i$ are selected from a circle of radius b centered on (x, y). Indeed, after projection on the camera plane, the circle is the standard shape that delineates the vehicle best. Let us denote the target and the candidate histograms by H_t and $H_{\mathbf{x}}$, respectively. The Bhattacharyya distance between two histograms is defined as:

$$\Delta(\mathbf{x}) = \left(1 - \sum_{u=1}^{m}\sqrt{H_t[u]H_{\mathbf{x}}[u]}\right). \tag{15}$$

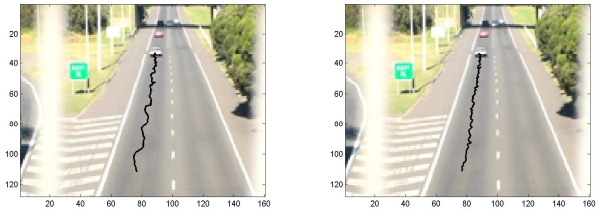
Finally, the likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is calculated as $p(\mathbf{z}_k|\mathbf{x}_k^i) \propto \exp\left(-\Delta(\mathbf{x}_k^i)\right)$.

3.4. Projective Particle Filter Implementation

Algorithm 1 Projective Particle Filter Algorithm

Require: $\mathbf{x}_0^i \sim q(\mathbf{x}_0|\mathbf{z}_0)$ and $w_0^i = 1/N_S$
  for $i = 1$ to N_S **do**
    Compute $\mathbf{f}(\mathbf{x}_{k-1}^i)$ from Eq. (13)
    Draw $\mathbf{x}_k^i \sim q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k) = \mathcal{N}(\mathbf{x}_k^i, \mathbf{f}(\mathbf{x}_{k-1}^i), \Sigma_q)$
    Compute ratio $\gamma_k = \mathcal{N}(\mathbf{x}_k|\mathbf{x}_{k-1}, \boldsymbol{\mu}_\gamma, \Sigma_\gamma)$
    Update weights $w_k^i = w_{k-1}^i \times \gamma_k p(\mathbf{z}_k|\mathbf{x}_k)$
  end for
  Normalize w_k^i
  if $N_{eff} < N$ **then**
    $l = 0$
    for $i = 1$ to N_S **do**
      $\sigma_i = \text{cumsum}(w_k^i)$
      while $\frac{l}{N_S} < \sigma_i$ **do**
        $x_k^l = x_k^i$
        $w_k^l = 1/N_S$
        $l = l + 1$
      end while
    end for
  end if

The implementation of the projective particle filter algorithm is summarized in Algorithm 1. Because most approaches to tracking take the prior distribution as importance density, the samples $\mathbf{x}_k^i$ are directly drawn from the standard kinematic model. In this paper we differentiate between the prior and the importance density to obtain a better distribution of the samples. The initial state $\mathbf{x}_0$ is chosen as $\mathbf{x}_0 = [x_0, y_0, 10, 0, 20]^T$ where x_0 and y_0 are the initial coordinates of the object. The value $\mathbf{x}_0$ is thus used to draw the set of samples $\mathbf{x}_0^i \sim q(\mathbf{x}_0|\mathbf{z}_0) = \mathcal{N}(\mathbf{x}_0^i, \mathbf{f}(\mathbf{x}_0), \Sigma_q)$.



(a) Standard (b) Projective

Figure 2. Vehicle track for (a) the standard and (b) the projective particle filter. The projective particle filter exhibits a lower variance in the position estimation.

The transition prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and the importance density $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$ are both modeled with Gaussian noises. The prior covariance matrix and mean are initialized as $\Sigma_p = \text{diag}([6\ 1\ 1\ 1\ 4])$ and $\boldsymbol{\mu}_p = \mathbf{g}(\mathbf{x}_0)$, respectively, and $\Sigma_q = \text{diag}([1\ 1\ 0.5\ 1\ 4])$ and $\boldsymbol{\mu}_q = \mathbf{f}(\mathbf{x}_0)$, for the importance density. As a result, the variable γ_k is itself drawn from a Gaussian process $\mathcal{N}(\mathbf{x}_k|\mathbf{x}_{k-1}, \boldsymbol{\mu}_\gamma, \Sigma_\gamma)$ with covariance matrix $\Sigma_\gamma = (\Sigma_p^{-1} - \Sigma_q^{-1})^{-1}$ and $\boldsymbol{\mu}_\gamma = \Sigma\left(\Sigma_p^{-1}\boldsymbol{\mu}_p - \Sigma_q^{-1}\boldsymbol{\mu}_q\right)$ and $\Sigma_p \neq \Sigma_q$.

A resampling scheme is necessary to avoid the degeneracy of the particle set. Systematic sampling [8] is performed when the variance of the weight set is too large, *i.e.* when the number of the effective sample size N_{eff} falls below a given threshold N, arbitrarily set to $0.6N_S$ in the implementation. The number of effective samples is evaluated as

$$N_{eff} = \frac{1}{\sum_{i=1}^{N_S}(w_k^i)^2}\,. \tag{16}$$

4. Experiments and Results

The standard and the projective particle filters are evaluated in this section on traffic surveillance data. An important measure in vehicle tracking is the variance of the trajectory. Indeed, high-level tasks, such as abnormal behavior or DUI detection, require an accurate tracking of the vehicle and, in particular, a low Mean Squared Error (MSE) for the position. Figure 2 displays a track estimated with the projective particle filter and the standard particle filter. We run two experiments aiming to evaluate the variance for the standard and the projective particle filters: one with automatic variance estimation and the other one with ground truth labeling. A third experiment is conducted to evaluate the suitability of the importance pdf. The video sequences are footage of vehicles traveling on a highway. Although the roads are straight in the dataset, the algorithm can be applied to curved roads with approximation of the parameters on the short distance because the projection tends to

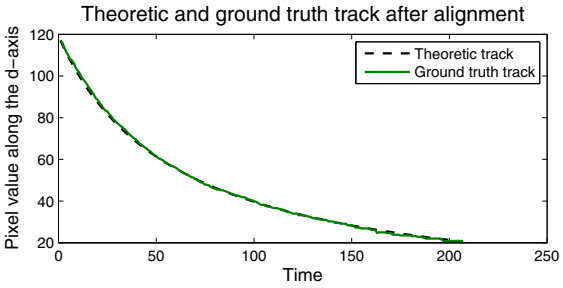


Figure 3. Alignment of theoretic and extracted trajectories along the d-axis. The difference between the two tracks represents error in the estimation of the trajectory.

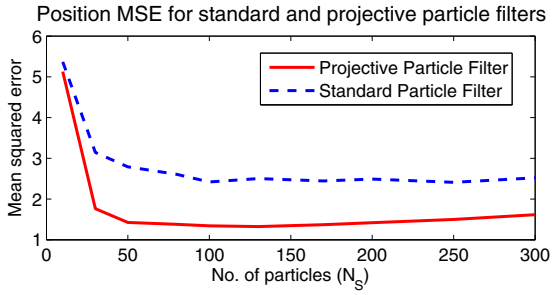


Figure 4. Position mean squared error versus number of particles for the standard and the projective particle filter.

linearize the curves on the camera plane. The parameters θ, H and D defining the linear fractional transformation are recorded in Table 1. The dataset is composed of 205 vehicles assumed to have a constant speed of $v = 25\text{m.s}^{-1}$. Note that the constraint on the speed can be relaxed as long as the variations are slow.

Table 1. Linear Fractional Transformation Parameters

Video Sequence	H	θ	D
Video_1	5.5 m	12.5 ± 0.15 deg	80 m
Video_2	5.5 m	19.2 ± 0.2 deg	57 m
Video_3	5.5 m	19.2 ± 0.2 deg	57 m

In the first experiment, the performance of each tracker is evaluated in terms of MSE using the entire dataset. In order to avoid the tedious task of manually extracting the ground-truth of every track, a synthetic track is generated automatically based on the parameters of the real world projection of the vehicle trajectory on the camera plane. Figure 3 shows that the theoretic and the manually extracted tracks match. The initialization of the tracks is performed as in [2]. However, because the initial position of the vehicle when the tracking starts may differ from one track to another, it is necessary to align the theoretic and the extracted tracks in order to cancel the bias in the estimation of the MSE. The average MSE for each video sequence is summarized in Table 2. It can be inferred that the projective particle filter performs better on the entire dataset than the standard particle filter.

Table 2. MSE for the standard and the projective particle filters

Video Sequence	Video_1	Video_2	Video_3
Avg. MSE Std PF	2.26	0.99	1.07
Avg. MSE Proj. PF	1.89	0.83	1.02

In the second experiment, we evaluate the performance of the two tracking algorithms w.r.t. the number of particles. Here, the ground truth is manually labeled in the video sequence. We arbitrarily decide to ground truth the first 5 trajectories of the first video to ensure the impartiality of

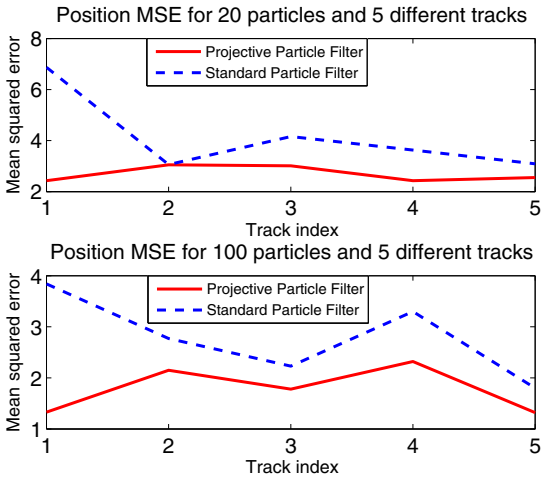


Figure 5. Position mean squared error for 5 ground truth labeled vehicles using the standard and the projective particle filter. *Top*: with 20 particles; *bottom*: with 100 particles.

the evaluation. Figure 4 displays the average MSE over 10 epochs for the first trajectory and for different values of N_S. Figure 5 presents the average MSE for 10 epochs on the 5 ground truth tracks for $N_S = 20$ and $N_S = 100$. It is clear that the projective particle filter outperforms the standard particle filter in terms of MSE. The better accuracy of the PPF is due to the finer estimation of the sample distribution by the importance density and the consequent adjustment of the weights, all parameters being identical in the comparison.

In the third experiment, we propose to compare the standard and the projective particle filters without the resampling step. This evaluation determines the suitability of the importance density to the problem tackled. Indeed, the closer the importance density is from the posterior density, the less resampling is needed. However, because the two pdfs are different, a larger number of particles is required to draw the results. We choose $N_S = 300$ for the evaluation. Figure 6 shows the position MSE for the standard and the projective particle filters for 80 trajectories in Video_3; the average MSE are 1.10 and 0.58, respectively. For the

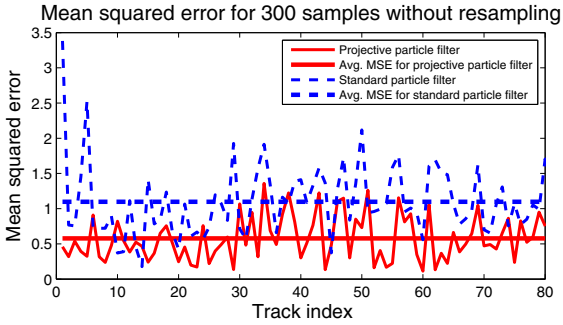


Figure 6. Position mean squared error for the standard and the projective particle filter without resampling step.

problem of vehicle tracking, the importance density q used in the projective particle filter is therefore more suitable to draw samples from compared to the prior density, used in the standard particle filter. Less resampling is required as a consequence of the adequate choice of importance density. It is also worth noting that the lower MSE in this experiment compared to the one exhibited in Table 2 for Video_3 is due to the higher number of particles.

5. Conclusion

The paper proposed a new particle filter integrating the linear fractional transformation in the importance density. This projection maps the real world position of a vehicle onto the camera plane providing a better distribution of the samples in the feature space. However, because the prior is not used to sample, the weights of the designed Projective Particle Filter have to be readjusted. The standard and the projective particle filters have been evaluated on traffic surveillance videos. It has been shown that the MSE on the trajectory of the vehicles is reduced with the projective particle filter. Furthermore, the proposed technique outperforms the standard particle filter in terms of MSE regardless of the number of particles. It has also been shown that the degeneracy of the samples set is reduced when the importance density integrates the linear fractional transformation.

Acknowledgments

This work was supported in part by the Australian Research Council.

References

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

[2] P. L. M. Bouttefroy, A. Bouzerdoum, S. L. Phung, and A. Beghdadi. Vehicle tracking by non-drifting mean-shift using projective kalman filter. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 61–66, 2008. 5

[3] J.-Y. Choi, K.-S. Sung, and Y.-K. Yang. Multiple vehicles detection and tracking based on scale-invariant feature transform. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 528–533, 2007. 1

[4] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577, 2003. 3

[5] F. Dellaert, D. Pomerlau, and C. Thorpe. Model-based car tracking integrated with a road-follower. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 1889–1894, 1998. 1

[6] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437, 2002. 2

[7] Y.-K. Jung and Y.-S. Ho. Traffic parameter extraction using video-based vehicle tracking. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 764–769, 1999. 1

[8] G. Kitagawa. Monte carlo filter and smoother for non-gaussian non-linear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996. 4

[9] D. Koller, J. Weber, and J. Malik. Towards realtime visual based tracking in cluttered traffic scenes. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 201–206, 1994. 1

[10] A. Kong, J. S. Lui, and W. H. Wong. Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Assocation*, 89(425):278–288, 1994. 2

[11] C.-P. Lin, J.-C. Tai, and K.-T. Song. Traffic monitoring based on real-time image tracking. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 2091–2096, 2003. 1

[12] E. B. Meier and F. Ade. Tracking cars in range images using the condensation algorithm. In *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pages 129–134, 1999. 1

[13] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor. Detection and classification of highway lanes using vehicle motion trajectories. *IEEE Transactions on Intelligent Transportation Systems*, 7(2):188–200, 2006. 1

[14] Z. Qiu, D. An, D. Yao, D. Zhou, and B. Ran. An adaptive kalman predictor applied to tracking vehicles in the traffic monitoring system. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 230–235, 2005. 1

[15] Y. Rathi, N. Vaswani, and A. Tannenbaum. A generic framework for tracking using particle filter with dynamic shape prior. *IEEE Transactions on Image Processing*, 16(5):1370–1382, 2007. 2

[16] D. Schreiber, B. Alefs, and M. Clabian. Single camera lane detection and tracking. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 302–307, 2005. 3

[17] T. Xiong and C. Debrunner. Stochastic car tracking with line- and color-based features. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, volume 2, pages 999–1003, 2003. 1