Sequential Bayesian estimation techniques for the tracking problem in computer vision

Milica Cvjetković

Abstract — **This paper presents review of techniques and algorithms used for filtering and data association in visual tracking. Kalman filter is an optimal Bayesian filter for linear dynamic models with Gaussian noise. Most of the processes and systems in real world are nonlinear, and in these situations there is extension of Kalman filter named the Extended Kalman filter (EKF). In case when the noise is non-Gaussian and/or nonlinearities are more severe, there is approach proved as more reliable for these tracking problems, known as Particle filtering. This paper aims to present the mathematical description of all presented algorithms and performance analysis and simulation results are given.**

Index Terms —**Bayesian estimation, linear models, Kalman filters, nonlinear/non-Gaussian models, particle filters, sequential Monte Carlo, tracking.**

I. Introduction

Tracking in the context of computer vision is the problem of locating a moving object and generating an inference about the actual position and motion of an object. It has been studied and refered in many papers, in recent years, because of the large number of applications which are mentioned in [1], [2], such as: surveillance (vehicle and people tracking) and human-computer interaction. Vehicle tracking systems can be used for collecting traffic data from highway scenes which report about traffic congestion, accidents and. dangerous or illegal behaviour by road users [3], [4], also can be used to analyse complex environments such as airports [5]. Surveillance systems are useful for giving report about people behavior in different situations and specific places, for example on stadium to know who maybe threw a bottle onto the field, in banks to find who is potential robber if something like that happened, at the customs if they want to know who is crossing a border. Human-computer interaction systems use human motions and actions for creating some artificial enviroments [6] or to drive various devices, for example creating an interactive playroom for children. There is also application of tracking in robotics, for example autonomuos robot which should be able to follow objects in their enviroment [7] and in speech recognition where it is used for lip-tracking [8].

Milica Cvjetković, student, the Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia; (e-mail: mile2807@gmail.com).

Mentor: Vladimir Crnojević, the Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia; (e-mail: crnojevic@uns.ns.ac.yu).

The main question in tracking is what can we tell about an object's current position from a sequence of measurements and using that to make a prognosis about its future positions. In tracking problem, we formulate a dynamic model explaining object's state evolution, and a set of measurements from a sequence of images. These measurements could be the position and/or velocity of some image regions. These measurements hopefully come from object of interest but some of them might come from other objects or clutter and therefore some kind of reasoning is necessary in data association [1], but this lies beyond the scope of this paper.

This paper will present the standard algorithms used for tracking purposes, both from theoretical and practical standpoint, and their simulation results which are obtained using MATLAB. In the case when the dynamic and observation models are both linear with additive Gaussian noise, the optimal minimum mean-squared error (MMSE) estimation scheme is the Kalman filter [1], [10] and it will be explained in the section 2. Non-linearities introduce a host of unpleasant problems. There is a special case where both dynamic and observation models are weakly nonlinear but the noises are additive and Gaussian and the Extended Kalman filter (EKF) [10] has been the standard technique which is usually applied here, as optimal filter, but if the nonlinearities are significant or the noise is non-Gaussian, the EKF can be very unreliable and performs very poorly. And in that case, when we have nonlinear/non-Gaussian dynamic models, appropriate method is Particle filter [9], [10], it is Bayesian filtering approach based on sequential Monte Carlo sampling (section 3).

II. Kalman filter

The Kalman filter is optimal Bayesin filtering technique by criterion of MMSE, so the Kalman filter is sequential MMSE estimator [10]. It has been developed in the sixties by R. E. Kalman to try to solve the Wiener problem in a generally easier way. The filter has the advantage to be sequential. Wiener filter is optimal solution for linear estimation of stationary processes and Kalman filter is optimal filter for processes with time-evolving state, but must have a known dynamic model. For linear dynamic models and linear measurement models with additive Gaussian noise, all posterior probability density functions (pdf) remain Gaussian in every iteration of the filter. The filter predicts the process state at some time and then obtains advice in the form of noisy measurements. Kalman filtering consists of an iterative prediction-correction process [11]. In the prediction step, the algorithm uses the current state estimate to obtain the a priori estimate for the next time step; in the correction step, the measurement update uses a new meaasurement and together with a priori estimate it obtains an improved a posteriori estimate for the current state.

A. Linear Dynamic Models

In linear model the state vector is updated by multiplying it by some known system matrix and adding a Gaussian random variable of zero mean and known covariance [1]. The measurements are obtained in the similar way, i.e. in this case the state vector is multiplied by some known measurement matrix and both of these matrices may change in time. The used notation is

$$x \sim N(\mu, \Sigma) \tag{1}$$

where x is the Gaussian random variable with mean μ and covariance Σ. Dynamic (state evolution) and observation linear models are given by i

$$\mathbf{x}_i \sim N(\mathbf{D}_i \mathbf{x}_{i-1}; \mathbf{\Sigma}_{d_i}) \tag{2}$$

$$\mathbf{y}_i \sim N(\mathbf{M}_i \mathbf{x}_i; \mathbf{\Sigma}_{m_i}) \tag{3}$$

$\mathbf{x}_i$ - state vector of the object

$\mathbf{D}_i$ - system matrix

$\mathbf{\Sigma}_{d_i}$ - covariance of the system noise

$\mathbf{y}_i$ - measurement vector

$\mathbf{M}_i$ - measurement matrix

$\mathbf{\Sigma}_{m_i}$ - covariance of the measurement noise

In the rest of this section, Kalman filter framework will be explained using some simple linear dynamic and observation models, e.g. drifting point case and constant velocity moving point. Finally, a short review of the extension of Kalman filter for nonlinear case is given.

B. The Kalman filter for a drifting point

For the sake of simplicity of explanation, we will assume that represent the position of a point on the line. In order to estimate the state sequentialy, we need to maintain a representation of $P(x_i | y_0, \ldots, y_{i-1})$ and of $P(x_i | y_0, \ldots, y_i)$ and all we need is to represent the mean and the standard deviation for the prediction, $\bar{x}_i^-$ and σ_i^-, and the mean and standard deviation for correction phase, $\bar{x}_i^+$ and σ_i^+, because Gaussian distribution is completely described by these two parameters. Assumption is that $\bar{x}_{i-1}^+$ and σ_{i-1}^+ are known, and $\bar{x}_0^-$ and σ_0^-, too.

The dynamic and observation models for this scalar case are

$$x_i \sim N(d_i x_{i-1}, \sigma_{d_i}^2) \tag{4}$$

$$y_i \sim N(m_i x_i, \sigma_{m_i}^2) \tag{5}$$

Standard assumption for the model parameters is that they don't change in time, i.e. $d_i = d = const.$, $\sigma_{d_i} = \sigma_d$, $\sigma_{m_i} = \sigma_m$ and specifically $m_i = 1$.

Prediction step of the Kalman filter:

$$\bar{x}_i^- = d\bar{x}_{i-1}^+ \tag{6}$$

$$\sigma_i^- = \sqrt{\sigma_d^2 + (d\sigma_{i-1}^+)^2} \tag{7}$$

Correction step:

$$\bar{x}_i^+ = \frac{\bar{x}_i^- \sigma_m^2 + y_i (\sigma_i^-)^2}{\sigma_m^2 + (\sigma_i^-)^2} \tag{8}$$

$$\sigma_i^+ = \sqrt{\frac{\sigma_m^2 (\sigma_i^-)^2}{\sigma_m^2 + (\sigma_i^-)^2}} \tag{9}$$

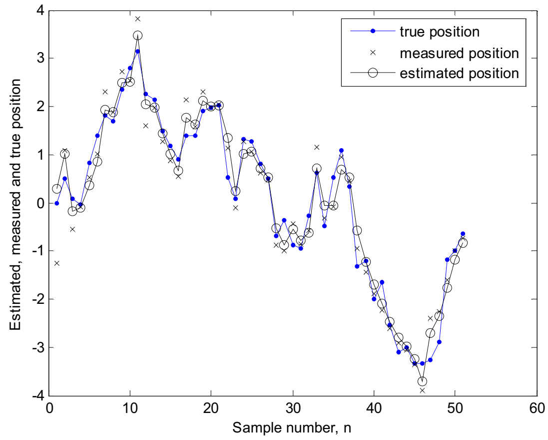

Figure 1. Simulation results for Kalman filter applied to drifting point model.

The figure 1 shows simulation results obtained for the drifting point plotted against time axis, and as we see Kalman filter estimation gives very good results in tracking of the drifting point although measurements are quite noisy.

C. The Kalman filter for a constant velocity moving point

A model of a point moving along the line with constant velocity is assumed. In this model the state vector consists of the scalar p which gives the position and scalar v which gives the velocity of a point and our state vector will be defined as

$$\mathbf{x} = \begin{bmatrix} p \\ v \end{bmatrix} \tag{10}$$

In this case

$$\mathbf{D}_i = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \tag{11}$$

$$\mathbf{M}_i = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{12}$$

where $\Delta t = 1$, since the time unit is unspecified, and measurements are comprised only of point position measurements. As in the case of the drifting point, an assumption about fixed covariance matrix values for $\mathbf{\Sigma}_{d_i}$ and $\mathbf{\Sigma}_{m_i}$ is made.

Prediction step:

$$\bar{\mathbf{x}}_i^- = \mathbf{D}_i \bar{\mathbf{x}}_{i-1}^+ \quad (13)$$

$$\mathbf{\Sigma}_i^- = \mathbf{\Sigma}_{d_i} + \mathbf{D}_i \mathbf{\Sigma}_{i-1}^+ \mathbf{D}_i^T \quad (14)$$

Correction step:

$$\mathbf{K}_i = \mathbf{\Sigma}_i^- \mathbf{M}_i^T \left[\mathbf{M}_i \mathbf{\Sigma}_i^- \mathbf{M}_i^T + \mathbf{\Sigma}_{m_i} \right]^{-1} \quad (15)$$

$$\bar{\mathbf{x}}_i^+ = \bar{\mathbf{x}}_i^- + \mathbf{K}_i \left[\mathbf{y}_i - \mathbf{M}_i \bar{\mathbf{x}}_{i-1} \right] \quad (16)$$

$$\mathbf{\Sigma}_i^+ = \left[\mathbf{Id} - \mathbf{M}_i \mathbf{K}_i \right] \mathbf{\Sigma}_i^- \quad (17)$$

The matrix K_i is called the Kalman gain.

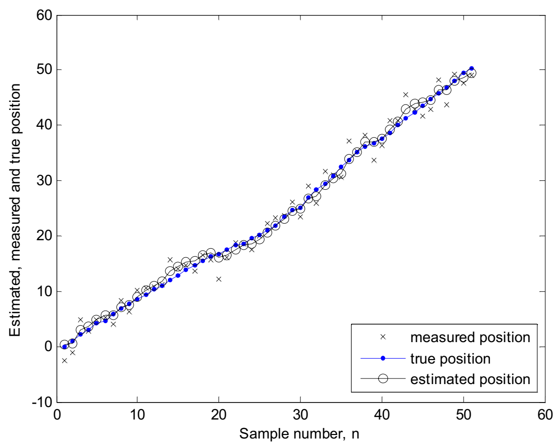

Figure 2. Simulation results for Kalman filter applied to a constant velocity model.

The figure 2 shows simulation results of constant velocity model plotted against time axis and we can say that no algorithm can do better estimation than a Kalman filter in linear Gaussian enviroment.

D. The Extended Kalman filter

The Extended Kalman Filter or EKF is a sub-optimal approach to the nonlinear model case where the model is first linearized in the neighborhood of the current state vector and then the standard Kalman filter is applied. This approach is applicable to non-linear models where the random part of model is still the additive Gaussian noise. It is a linearization method based on the first order Taylor series expansion of non-linear equations governing deterministic parts of the dynamic and observation model. Now the non-linear dynamic model is:

$$\mathbf{x}_i \sim N\left(f_i(\mathbf{x}_{i-1}); \mathbf{\Sigma}_{d_i} \right) \quad (18)$$

$$\mathbf{y}_i \sim N\left(h_i(\mathbf{x}_i); \mathbf{\Sigma}_{m_i} \right) \quad (19)$$

where at least f_i or h_i is a non-linear function and both of them must be differentiable functions of their vector arguments.

Prediction step:

$$\bar{\mathbf{x}}_i^- = f\left(\bar{\mathbf{x}}_{i-1}^+ \right) \quad (20)$$

$$\mathbf{\Sigma}_i^- = \mathbf{\Sigma}_{d_i} + \Im_f\left(\bar{\mathbf{x}}_{i-1}^+ \right) \mathbf{\Sigma}_{i-1}^+ \Im_f^T\left(\bar{\mathbf{x}}_{i-1}^+ \right) \quad (21)$$

Correction step:

$$\mathbf{K}_i = \mathbf{\Sigma}_i^- \Im_h^T\left(\bar{\mathbf{x}}_i^- \right) \left[\Im_h\left(\bar{\mathbf{x}}_i^- \right) \mathbf{\Sigma}_i^- \Im_h^T\left(\bar{\mathbf{x}}_i^- \right) + \mathbf{\Sigma}_{m_i} \right]^{-1} \quad (22)$$

$$\bar{\mathbf{x}}_i^+ = \bar{\mathbf{x}}_i^- + \mathbf{K}_i \left[\mathbf{y}_i - h_i\left(\bar{\mathbf{x}}_i^- \right) \right] \quad (23)$$

$$\mathbf{\Sigma}_i^+ = \left[\mathbf{Id} - \mathbf{K}_i \Im_h\left(\bar{\mathbf{x}}_i^- \right) \right] \mathbf{\Sigma}_i^- \quad (24)$$

The Jacobian of some multidimensional vector argument function $g(\mathbf{x})$ which is evaluated at some point $\mathbf{x}_j$ is written as $\Im_g(\mathbf{x}_j)$. The example and results for some simple non-linear models will be presented in the next section on Particle filters, together with comparison of these two methods and show that Particle filtering gives better results than EKF.

III. PARTICLE FILTERS

Particle filter is a variant of the Sequential Monte Carlo (SMC) methods. SMC is a set of flexible simulation-based methods used for making a representation of some probability distribution function by sampling from a arbitrary probability distribution [9], [10]. These methods were originally introduced in the early 50's by physicists. Particle filter is a kind of recursive Bayesian filter operating on the pdf representations generated by Monte Carlo simulation. These representations assist in finding pdf moments, for the tracking problem most of all the mean (first order moment) and variance (the second order centralized moment).

The main idea of particle filter is to represent the sequence of probability distributions and of using a large set of random samples, named particles with associated weights and to compute estimates based on these samples and weights. This representation is called sampling distribution. The number of these particles is propagated over time using Importance Sampling and resampling mechanisms. For practical implementations, a finite and sometimes quite restricted number of particles has to be considered.

Here will be presented particle filter alghorithm with resampling the posterior known as Sequential Importance Sampling (SIS) algorithm. One sample is given by

$$\left\{ \left(s_i^{k,-}, w_i^{k,-} \right) \right\} \quad (25)$$

where the superscript k indexes the samples for a given step i, and the subscript gives the step i and the superscripts '-' and '+' indicate that it is representation of the i'th state before and after measurement has been obtained, respectively.

Initialisation: Representing $P(X_0)$ by a set of N samples

$$\left\{ \left(s_0^{k,-}, w_0^{k,-} \right) \right\} \quad (26)$$

that are drawn from the known distribution

$$s_0^{k,-} \sim P(X_0) \quad (27)$$

so that

$$w_0^{k,-} = 1 \quad (28)$$

Prediction: Represent $P\left(X_i \middle| y_0, \ldots, y_{i-1} \right)$ by

$$\left\{ \left(s_i^{k,-}, w_i^{k,-} \right) \right\} \quad (29)$$

where

$$s_i^{k,-} = f\left(s_{i-1}^{k,+} \right) + \xi_i^k \quad (30)$$

$$w_i^{k,-} = w_{i-1}^{k,+} \quad (31)$$

$$\xi_i^k \sim N\left(0, \Sigma_{d_i} \right) \quad (32)$$

Correction: Represent $P(X_i|y_0,\ldots,y_i)$ by

$$\{(s_i^{k,+}, w_i^{k,+})\} \quad (33)$$

where

$$s_i^{k,+} = s_i^{k,-} \quad (34)$$

$$w_i^{k,+} = P(Y_i = y_i | X_i = s_i^{k,-}) w_{i-1}^{k,-} \quad (35)$$

Resampling: First is done normalisation of the weights so that $\sum_{k=1}^{N} w_i^{k,+} = 1$ and then computing the variance of the normalised weights. If this variance exceeds some threshold, then construct a new set of samples by drawing, with replacement, N samples from the old set, using the weights as the probability for a sample to be drawn. Finally, the weights of each sample in the new set are 1/N.

Now will be presented and compared results obtained by application of Extended Kalman filter and Particle filter for the same nonlinear 1D dynamic model $x_i = N(x_{i-1} + 0.5\sin(x_{i-1}), \sigma_{d_i}^2)$. The observation model is the same as in part *B*.

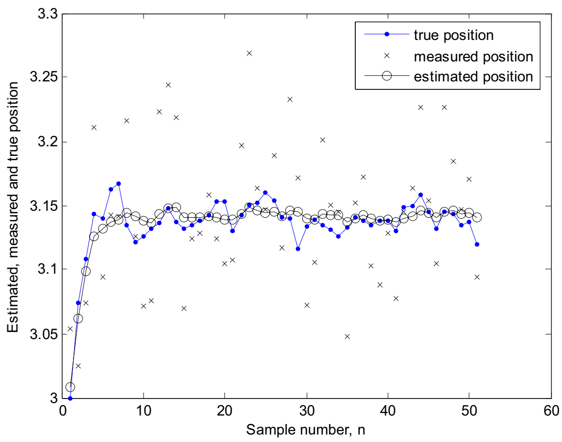

Figure 3. Simulation results for the Extended Kalman filter applied to given nonlinear dynamic model.

The figure 3 shows the EKF estimation and here position is plotted against the time. As we see effects of linearisation are visible, but results obtained by the EKF are quite poor which is partly due to the severe nonlinearities present in the dynamic model.

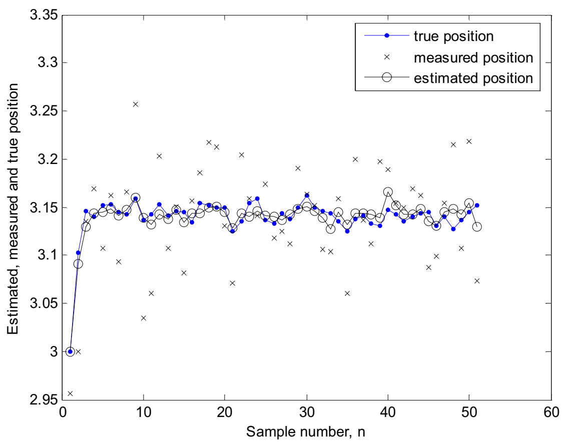

Figure 4. Simulation results for the Particle filter applied to given nonlinear dynamic model.

The figure 4 shows position plotted against the time and there is shown the Particle filter estimation. It is obviously that the Particle filter gives very good results in tracking point which moving is determinated by nonlinear dynamic model and we can say that Particle filter certainly gives better results than EKF .

Averaged squared error ASE is an indicator for the accuracy of state estimation based on a single process realization.

ASE obtained by the Extended Kalman filter is:

ASE= 2.9351e-004

ASE obtained by the Particle filter is:

ASE= 1.2662e-004

These results prove once again that particle filter is, in general, better estimator than EKF for an arbitrary non-linear model.

IV. Conclusion

At the end, beside description and explanation of the Kalman filter and Particle filter as estimation techniques in visual tracking problems, this paper presented their simulation results for some dynamic models and upon which is confirmed that Kalman filter is optimal Bayesian filter in linear Gaussian environment, the EKF is good solution for models with weak nonlinearities and Gaussian noise, but Particle filter is more reliable method for models with significant nonlinearities and non-Gaussian noise.

References

[1] D. Forsyth, "Computer Vision: A Modern Approach", Prentice Hall, 2002

[2] M. A. Isard, "Visual Motion Analysis by Probabilistic Propagation of Conditional Density", Robotics Research Group, Department of Engineering Science, University of Oxford, September 1998

[3] Ferrier, N., Rowe, S., and Blake, A., "Real-time traffic monitoring". In Proc. 2nd IEEE Workshop on Applications of Computer Vision, 81-88 IEEE, 1994

[4] Koller. D., Weber, J., and Malik, J., "Robust multiple car tracking with occlusion reasoning". In Proc. 3rd European Conf. Computer Vision, 189-196, Springer-Verlag, 1994

[5] Sullivan, G., "Visual interpretation of known objects in constrained scenes". Phil. Trans. R. Soc. Lond. B., 337, 361-370, 1992

[6] Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A., Pffinder: "Real-time tracking of the human body". IEEE Transactions on Pattern Analysis and Machine Intelligence, 19, 7, 780-785, 1997

[7] Davison, A. J. and Murray, D. W., "Mobile robot localisation using active vision". In "Submitted to the 5th European Conference on Computer Vision", Freiburg, 1998

[8] Bregler, C. and Omohundro, S., "Nonlinear manifold learning for visual speech recognition". In Proc. 5th Int. Conf. on Computer Vision, 494-499, Boston, 1995

[9] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", IEEE Trans. on Signal Processing, Vol. 50, no. 2, Feb. 2002

[10] Z. Chen, "Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond", Technical report, the Communications Research Laboratory, McMaster University, Hamilton, Ontario, Canada 2003

[11] G. Welch M, G. Bishop, "An introducing to the Kalman filter", University of North Carolina at Chapel Hill, Department of Computer Science, SIGGRAPH 2001

[12] A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian Processes, A.J. Haug, The MITRE Corporation, January 2005