

Multiple Sensors and JPDA-IMM-UKF Algorithm for Tracking Multiple Maneuvering Targets

Wisseem Saidani, Yacine Morsly, and Mohand Saïd Djouadi

Abstract—In this paper, we consider the problem of tracking multiple maneuvering targets using switching multiple target motion models. With this paper, we aim to contribute in solving the problem of model-based body motion estimation by using data coming from visual sensors. The Interacting Multiple Model (IMM) algorithm is specially designed to track accurately targets whose state and/or measurement (assumed to be linear) models changes during motion transition. However, when these models are nonlinear, the IMM algorithm must be modified in order to guarantee an accurate track. In this paper we propose to avoid the Extended Kalman filter because of its limitations and substitute it with the Unscented Kalman filter which seems to be more efficient especially according to the simulation results obtained with the nonlinear IMM algorithm (IMM-UKF). To resolve the problem of data association, the JPDA approach is combined with the IMM-UKF algorithm, the derived algorithm is noted JPDA-IMM-UKF.

Keywords—Estimation, Kalman filtering, Multi-Target Tracking, Visual servoing, data association.

I. INTRODUCTION

THIS paper hope to be a contribution within the field of visual-based control of robots, especially in visual-based tracking [3]; tracking maneuvering targets, which may themselves be robots, is a complex problem, to ensure a good track when the target switches abruptly from a motion model to another is not evident. Because of the complexity and difficulty of the problem, a simple case is considered. The study is restricted to 2-D motions of a point, whose position is given at sampling instants in terms of its Cartesian coordinates. This point may be the center of gravity of the projection of an object into a camera plane, or the result of the localisation of a mobile robot moving on a planar ground.

Several of maneuvering targets tracking algorithms are developed. Among them, the interacting multiple model (IMM) method based on the optimal Kalman filter, yields good performance with efficient computation especially when the measurement and state models are linear. However, if the latter are nonlinear, the standard Kalman filter should be substituted, in our study we choose the recent Unscented Kalman Filter (UKF). The other problem treated in this paper, is about the data association. Effectively, at each sample time,

the sensors (cameras) present, several measures and observations, coming from different targets; the problem is how to affect each measure to the correct target, to resolve this problem we choose the JPDA algorithm. The algorithm derived from the combination of JPDA and the non linear IMM algorithms is noted JPDA-IMM-UKF.

The paper is organized as follows. In section II the mathematical formulation of 2-D motion is presented. In section III we briefly present the UKF. We describe in section IV the nonlinear IMM algorithm UKF based. In section V we present the JPDA-IMM-UKF algorithm. In section VI we present and discuss the results of simulations. Finally in section VII we draw the conclusion.

II. MATHEMATICAL FORMULATION OF 2-D MOTION

The mathematical formulation of 2-D motion used is mainly inspired from Danes, Djouadi, and al in [4]. They make the hypothesis that the measurements are only the 2-D Cartesian coordinates of the moving point.

Let $s(\cdot)$ denote the curvilinear abscissa of M over time onto its trajectory, the origin of curvilinear abscissae is set arbitrarily. Functions $x(\cdot)$ and $y(\cdot)$, represent the Cartesian coordinates of M. The measurement equation may be written as:

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \underline{h}(\underline{s}(t), \underline{p}(t)) \quad (1)$$

Where $\underline{p}(\cdot)$ is a parameter vector function of minimal size.

We can see that equation (1) is independent of the type of the motion of M onto its trajectory.

The state equation could be written as:

$$\dot{\underline{X}}(t) = \underline{A}\underline{X}(t) \quad (2)$$

$$\text{with } \underline{X}(t) = \begin{pmatrix} \underline{s}(t) \\ \underline{p}(t) \end{pmatrix}$$

A equals $\begin{pmatrix} \underline{A}_s & 0 \\ 0 & 0 \end{pmatrix}$, with \underline{A}_s the $n \times n$ zero matrix with ones

added on its first upper diagonal, and 0 the matrices of convenient sizes. The continuous time state equation (2) is linear time invariant and independent of M's trajectory, except on the sizes of $\underline{s}(\cdot)$ and $\underline{p}(\cdot)$. Moreover, it may be shown

Authors are with Laboratoire robotique & productique, Ecole militaire polytechnique, Bp :17 Bordj El Bahri 16111 Alger, Algérie (e-mails: [wsaidani; ymorsly; msdjouadi]@yahoo.fr).

that the fundamental matrix \mathbf{F} involved its exact discretization at the period T takes the form

$$\mathbf{F} = \exp(\mathbf{A}T) = \sum_{i=0}^{n-1} \frac{(\mathbf{A}T)^i}{i!}.$$

The dynamic and measurement noises are supposed to be stationary, white and Gaussian, non inter-correlated with known covariances.

A. Canonical Motion Equations

The point M is supposed to move on straight or circular trajectories at constant or uniformly time-varying speed (constant speed or constant acceleration). Those motions belong to the set of the possible behaviours of a non-holonomic robot whose wheels are driven at constant velocities or accelerations.

1) *Output equations*: One minimal description of a straight line is defined by the vector $\underline{p} = (\alpha, d)^T$ shown in figure 1(a), which is related to Plucker coordinates. Concerning a circular trajectory one minimal description is defined by the vector $\underline{p} = (R0, x_0, y_0)^T$ shown in figure 1(b). The origin of curvilinear abscissa is uniquely defined from those parameterizations.

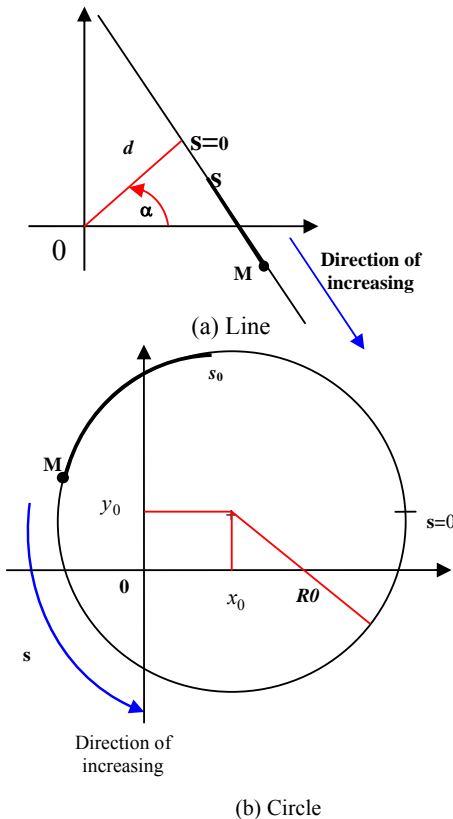


Fig. 1 Trajectory Parameterization

The output equations are as follows (trajectory parameter are considered time-invariant):

$$\text{Straight Line: } \underline{z}(k) = \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} = \begin{pmatrix} d \cos \alpha + s(k) \sin \alpha \\ d \sin \alpha - s(k) \cos \alpha \end{pmatrix} + \underline{v}(k) \quad (3)$$

$$\text{Circle: } \underline{z}(k) = \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} = \begin{pmatrix} x_0 + R0 \cos \frac{s(k)}{R0} \\ y_0 + R0 \sin \frac{s(k)}{R0} \end{pmatrix} + \underline{v}(k) \quad (4)$$

with $s(k)$ distance covered by the target and $\underline{v}(\cdot)$ zero mean Gaussian noise, $E\{\underline{v}(k)\} = 0$, $E\{\underline{v}(k) \cdot \underline{v}^T(k')\} = R \delta_{k,k'}$.

2) State Equations

$$\text{Constant velocity } \begin{cases} \underline{s}(k+1) = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \underline{s}(k) + \underline{w}_s(k) \\ \underline{p}(k+1) = \underline{p}(k) + \underline{w}_p(k) \end{cases} \quad (5)$$

Constant acceleration

$$\begin{cases} \underline{s}(k+1) = \begin{pmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix} \underline{s}(k) + \underline{w}_s(k) \\ \underline{p}(k+1) = \underline{p}(k) + \underline{w}_p(k) \end{cases} \quad (6)$$

where the random vectors

$$\underline{x}(0) = (\underline{s}(0)^T, \underline{p}(0)^T)^T, \quad \underline{w}(\cdot) = (\underline{w}_s(\cdot)^T, \underline{w}_p(\cdot)^T)^T \quad \text{and} \quad \underline{v}(\cdot)$$

are jointly Gaussian and:

$$E\{\underline{x}(0)\} = \hat{\underline{x}}(0), \quad E\{\underline{w}(k)\} = 0, \quad E\{\underline{v}(k)\} = 0,$$

$$E\left\{ \begin{pmatrix} \underline{x}(0) - \hat{\underline{x}}(0) \\ \underline{w}(k) \\ \underline{v}(k) \end{pmatrix} \begin{pmatrix} \underline{x}(0) - \hat{\underline{x}}(0) \\ \underline{w}(k') \\ \underline{v}(k') \end{pmatrix}^T \right\} = \text{diag}(\mathbf{P}(0), \mathbf{Q} \delta_{k,k'}, \mathbf{R} \delta_{k,k'})$$

$$\text{and } \underline{s}(k) = \begin{pmatrix} s(k) \\ \dot{s}(k) \\ \ddot{s}(k) \end{pmatrix} \quad \text{vector of the point M dynamics}$$

$$\underline{p}(k) \quad \text{trajectory parameters vector}$$

III. THE UNSCENTED KALMAN FILTER

The basis of the UKF is the unscented transform, where a distribution is approximated using a number of vectors which are passed through the nonlinear function to determine the probability distribution of the output from the function.

A. The Unscented Transform

According to Julier and Uhlmann in [5, 6], the unscented transform is based on the intuition that it is simpler to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function. A set of vectors, selected to be representative of the probability distribution, are chosen so that their mean and covariance are respectively \bar{x} and P_{xx} .

The nonlinear function is applied to each point; the result is a set of transformed points with the statistics \bar{y} and P_{yy} . The selection of these vectors is not arbitrarily done but according to a deterministic algorithm.

The n -dimensional random variable x with mean and covariance respectively \bar{x} and P_{xx} is approximated by $2n+1$ weighted points given by:

$$\begin{aligned} \chi_0 &= \bar{x} & W_0 &= \frac{\kappa}{(n+\kappa)} \\ \chi_i &= \bar{x} + \left(\sqrt{(n+\kappa)P_{xx}} \right)_i & W_i &= \frac{1}{2(n+\kappa)} \quad i=1, \dots, n \\ \chi_{i+n} &= \bar{x} - \left(\sqrt{(n+\kappa)P_{xx}} \right)_i & W_{i+n} &= \frac{1}{2(n+\kappa)} \end{aligned} \quad (7)$$

Where $\kappa \in \mathbb{R}$, $\left(\sqrt{(n+\kappa)P_{xx}} \right)_i$ is the i th row or column of the matrix square root of $(n+\kappa)P_{xx}$ and W_i is the weight which is associated with the i th point, note that $\sum_{i=0}^{2n} W_i = 1$.

The transformation procedure occurs in three steps:

1. The transformed set of vectors are: $y_i = f(\chi_i)$ (8)

2. The mean is given by: $\bar{y} = \sum_{i=0}^{2n} W_i y_i$ (9)

3. The covariance is given by:

$$P_{yy} = \sum_{i=0}^{2n} W_i \{y_i - \bar{y}\} \{y_i - \bar{y}\}^T \quad (10)$$

According to Julier and Uhlmann [5,6], many interesting properties could be noticed about this algorithm, for instance it leads to greater accuracy and permit rapid implementation, thanks to the mean and covariance which are calculated using standard vector and matrix operations.

B. The Unscented Kalman Filter

The unscented Kalman filter is obtained by occurring little modifications on standard one; this, by restructuring the state vector and process and measurement models [5, 6].

Consider the following discrete-time nonlinear dynamical system:

$$\begin{aligned} x(k+1) &= f(x(k), u(k), k) + w(k) \\ z(k) &= h(x(k), u(k), k) + v(k) \end{aligned}$$

Where $x(k+1) \in \mathfrak{R}^n$ is the state vector, $z(k) \in \mathfrak{R}^m$ the measurement vector, $w(k) \in \mathfrak{R}^q$ and $v(k) \in \mathfrak{R}^m$ are Gaussian random vectors with zero mean and covariance matrices $Q \geq 0$ and $R > 0$, respectively.

At first the state vector is augmented with the process and noise terms this leads to an $n^a = n + q$ dimensional vector.

$$x^a(k) = \begin{bmatrix} x(k) \\ w(k) \end{bmatrix} \quad (11)$$

The process model is now a function of $x^a(k)$,

$$x(k+1) = f[x^a(k), u(k), k] \quad (12)$$

The unscented transform needs to $2n^a+1$ vectors which are selected from:

$$\hat{x}^a(k|k) = \begin{pmatrix} \hat{x}(k|k) \\ 0_{q \times 1} \end{pmatrix} \quad (13)$$

and

$$P^a(k|k) = \begin{bmatrix} P(k|k) & P_{xw}(k|k) \\ P_{xw}(k|k) & Q(k) \end{bmatrix}$$

The Kalman filter algorithm computes on two principles phases:

- Prediction of the state vector and its covariance matrix
- Estimation of the state vector and its covariance matrix by updating the prediction with the current measurement.

The prediction phase using the unscented transform is as follow:

1. The set of vectors are created by applying equation (7) to the augmented system given by equation (13)

2. The transformed set of vectors are given by

$$\chi_i(k+1|k) = f[\chi_i^a(k|k), u(k), k] \quad (14)$$

3. The predicted mean is calculated by

$$\hat{x}(k+1|k) = \sum_{i=0}^{2n^a} W_i \chi_i^a(k+1|k) \quad (15)$$

4. The predicted covariance is calculated by

$$P(k+1|k) = \sum_{i=0}^{2n^a} W_i \{ \chi_i(k+1|k) - \hat{x}(k+1|k) \} \{ \chi_i(k+1|k) - \hat{x}(k+1|k) \}^T \quad (16)$$

5. Each prediction vector is instantiate through the measurement model

$$z_i(k+1|k) = h(\chi_i(k+1|k), u(k), k) \quad (17)$$

6. The predicted observation is calculated by

$$\hat{z}(k+1|k) = \sum_{i=0}^{2n^a} W_i z_i(k+1|k) \quad (18)$$

7. Assuming the fact the measurement noise is additive and independent, the innovation covariance is

$$\begin{aligned} P_{zz}(k+1|k) &= R(k+1|k) + \\ &\sum_{i=0}^{2n^a} W_i \{ z_i(k+1|k) - \hat{z}(k+1|k) \} \{ z_i(k+1|k) - \hat{z}(k+1|k) \}^T \end{aligned} \quad (19)$$

8. The cross correlation matrix is calculated by

$$P_{xz}(k+1|k) = \sum_{i=0}^{2n^2} W_i \{ \chi_i(k+1|k) - \hat{x}(k+1|k) \} \{ z_i(k+1|k) - \hat{z}(k+1|k) \}^T \quad (20)$$

Given these predicted values the state and covariance estimates are computed according to the equations:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1)\gamma(k+1) \quad (21)$$

$$P(k+1|k+1) = P(k+1|k) - K(k+1)P_{\gamma\gamma}(k+1)K^T(k+1)$$

with $\gamma(k+1)$ the innovation and $K(k+1)$ the weight chosen to minimise the mean squared error of the estimate.

$$\begin{aligned} \gamma(k+1) &= z(k+1) - \hat{z}(k+1|k) \\ K(k+1) &= P_{xz}(k+1|k)P_{\gamma\gamma}^{-1}(k+1|k) \end{aligned} \quad (22)$$

IV. THE NONLINEAR INTERACTING MULTIPLE MODEL ALGORITHM

A. The IMM Algorithm

Let a system be described by the equations (notations are chosen according to [2])

$$\begin{aligned} x(k) &= F[M(k)]x(k-1) + w[k-1, M(k)] \\ z(k) &= H[M(k)]x(k) + v[k, M(k)] \end{aligned} \quad (23)$$

with $w(\cdot)$ Gaussian process noise and $v(\cdot)$ measurement Gaussian noise.

$$E\{w(k)\} = 0, E\{w(k) \cdot w^T(k)\} = Q\delta_{k,k'}$$

$$E\{v(k)\} = 0, E\{v(k) \cdot v^T(k)\} = R\delta_{k,k'}$$

Where $M(k)$ denotes the model at time k . It's a finite state Markov process taking values in $\{M_j\}_{j=1}^r$, according to a

Markov transition probability matrix p assumed to be known, v and w represent white Gaussian processes and are assumed to be mutually independent.

A cycle of the IMM algorithm could be summarized in four steps:

1. The mixed initial condition for filters

Starting with $\hat{x}^i(k-1|k-1)$, we compute the mixed initial condition for the filter matched to $M_j(k)$

$$\hat{x}^{0j}(k-1/k-1) = \sum_{i=1}^r \hat{x}^i(k-1/k-1) \mu_{ij}(k-1/k-1) \quad (24)$$

$$i, j = 1, \dots, r$$

the covariance corresponding to the above is

$$\begin{aligned} P^{0j}(k-1/k-1) &= \sum_{i=1}^r \mu_{ij}(k-1/k-1) \cdot \\ &\left\{ P^i(k-1/k-1) + [\hat{x}^i(k-1/k-1) - \hat{x}^{0j}(k-1/k-1)] \cdot \right. \\ &\left. [\hat{x}^i(k-1/k-1) - \hat{x}^{0j}(k-1/k-1)]^T \right\} \quad i, j = 1, \dots, r \end{aligned} \quad (25)$$

Where:

$$\mu_{ij}(k-1|k-1) = \frac{1}{\bar{c}_j} p_{ij} \mu_i(k-1) \quad i, j = 1, \dots, r \quad (26)$$

is the probability that model M_i was in effect at $k-1$ given that M_j is in effect at time k conditioned on Z^{k-1} .

And where:

- $\mu_i(k-1)$ the probability that the mode M_i is in effect at time $k-1$,
- $\bar{c}_j = \sum_{i=1}^r p_{ij} \mu_i(k-1)$ the normalizing constants.

2. Model-matched filtering

The above estimate and covariance are used as input to the filter matched to $M_j(k)$, which uses $z(k)$ to obtain $\hat{x}^j(k|k)$ and $P^j(k|k)$.

3. Model probability update

The r model probabilities are updated from the innovation of the r Kalman filters.

The likelihood functions corresponding to the r filters are given by:

$$\begin{aligned} \Lambda_j(k) &= N\{x(k); \hat{z}^j(k/k-1); \hat{x}^{0j}(k-1/k-1)\}, \\ &S_j\{k; P^{0j}(k-1/k-1)\} \quad j = 1 \dots r \end{aligned} \quad (27)$$

$$\mu_j(k) = \frac{1}{c} \Lambda_j(k) \bar{c}_j \quad j = 1, \dots, r \quad (28)$$

where $c = \sum_{j=1}^r \Lambda_j(k) \bar{c}_j$ is the normalization constant.

4. Estimation and covariance combination

The output estimates and covariances are computed according to the mixture equations

$$\hat{x}(k/k) = \sum_{j=1}^r \hat{x}^j(k/k) \mu_j(k) \quad (29)$$

$$\begin{aligned} P(k/k) &= \sum_{j=1}^r \mu_j(k) \\ &\left\{ P^j(k/k) + [\hat{x}^j(k/k) - \hat{x}(k/k)] [\hat{x}^j(k/k) - \hat{x}(k/k)]^T \right\} \end{aligned}$$

The problem here is that this algorithm is designed with the assumption that the target motion models are linear, so, for its optimality, the use of the Kalman filter is recommended. However, in the case on which the motion models are nonlinear we propose to use the UKF detailed in section III. The modification to operate on the IMM algorithm is to use as state and covariance estimator the UKF.

Thanks to this modification we hope to track accurately targets whose models are nonlinear, the resulting algorithm is called the Nonlinear Interacting Multiple Model and noted (IMM-UKF).

V. JPDA-IMM ALGORITHM

The principle of the JPDA algorithm is the computation of probabilities association for each track and new measurement.

These probabilities are then used as weighting coefficients in the formation of the averaged state estimate, which is used for updating each track. For a better description of the JPDA algorithm, see [2,7].

The combination of the JPDA and the IMM-UKF algorithms done as follows. A single set of validated measurements for JPDA-IMM-UKF is obtained by considering the intersection Z_k of r sets of measurements corresponding to individual models:

$$Z_k = \bigcap_{j=1}^r Z_k^j$$

Where Z_k^j represents the set of validated measurements under the assumption that model j is effective. The combined likelihood functions for the r modes of the IMM-UKF algorithm are computed as in [8].

The prior mixed state estimates for model j and the validation regions for individual models are also computed as in [2,8]. The new mode probabilities, output state estimates, and corresponding error covariances are obtained as in [2,8].

To consider multiple sensors, modifications on the later algorithm are made according to [10].

VI. SIMULATIONS AND RESULTS

In this section, we perform some simulations to evaluate our algorithm (JPDA-IMM-UKF). In our simulations we considered the case of two sensors.

The motion models considered are: - constant velocity on straight line (M_1), -constant acceleration on straight line (M_2), - constant velocity on circle (M_3), - constant acceleration on circle (M_4).

To explore the capability of our JPDA-IMM-UKF algorithm to track maneuvering targets, various scenarios are considered; among of them we select the typical case of three highly maneuvering targets with crossing trajectories.

We assume that the target is in a 2-D space and its position is sampled every $T=1s$. For all cases we assume:

- The measurement noise is zero mean, white, independent of the process noise, and with variance $\sigma_v^2=0.01$.
- The process noise is zero mean, white, independent of the measurement noise, and with variance $\sigma_w^2=0.001$.
- The probability transition matrix of four models is

$$P = \begin{bmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{bmatrix}$$

- The initial probability of selecting a model is 0.25, that's to say, at the start all models have the same chance to be selected.
- The curvilinear abscissa $s(\cdot)$ remains continuous even if a trajectory jump occurs.

A. Considered Scenario

We consider that we have to track simultaneously three maneuvering targets. In order to complicate the scenario, we suppose that the targets follow during their movements, crossing trajectories.

a) Target 1(black)

The target starts moving according to model M_1 until the 50th sample when an abrupt trajectory change occur and still moving according to this during 50 samples (switching from model M_1 to M_3).

b) Target 2 (blue)

The target starts moving according to model M_3 until the 50th sample when an abrupt acceleration about 0.2 m/s^2 occur and still moving according to this during 50 samples (switching from model M_3 to M_4).

c) Target 3 (green)

The target starts moving according to model M_1 until the 50th sample when an abrupt acceleration about 0.2 m/s^2 occur and still moving according to this during 50 samples (switching from model M_1 to M_2).

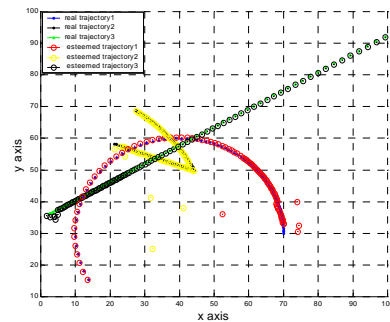


Fig. 1 Real and Estimated Trajectories

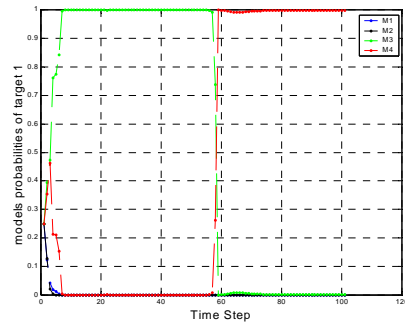


Fig. 2 Models Probabilities for target 1

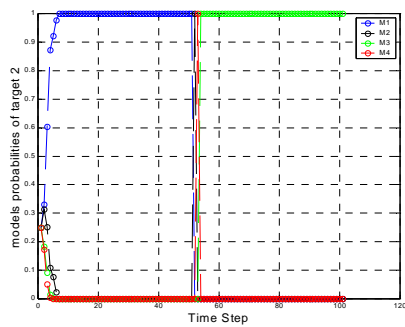


Fig. 3 Models Probabilities 2

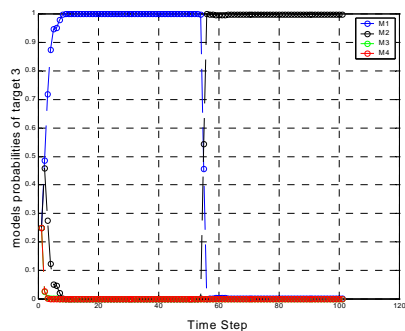


Fig. 4 Models Probabilities 3

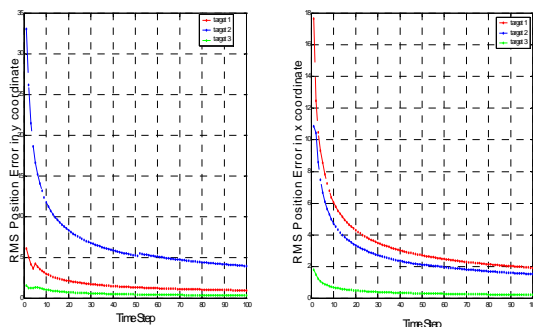


Fig. 5 RMS x and y position error

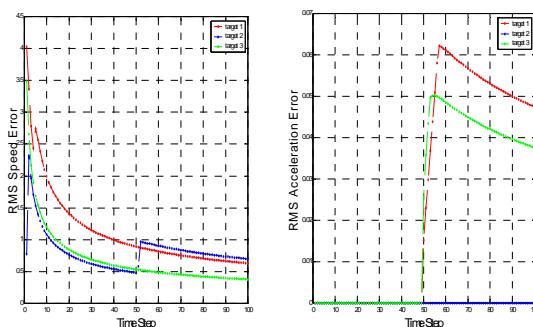


Fig. 6 RMS Acceleration and Speed Error

B. Results Interpretation

Fig. 1 shows that the esteemed and the real trajectory for the three targets are superposable and almost identical even if an abrupt change occurs on the tracked target dynamic. This result is confirmed by the Figs. (2,3,4,5,6), from this we can say that the tracker based IMM-UKF algorithm is a pertinent solution to the problem of visual-based tracking highly maneuvering targets. In the Other hand Fig. 1 shows also that the data association is correctly done even if the trajectories cross each other. This should permit us to say that the JPDA algorithm computes perfectly and its combination with the IMM-UKF algorithm (JPDA-IMM-UKF) would be an efficient solution to the problem of highly maneuvering multi-target visual-based tracking.

VII. CONCLUSION

The model-based body motion estimation by using data coming from visual sensors still an open problem on which we try to provide a contribution. In this paper we presented a nonlinear algorithm which attempts to track efficiently a highly maneuvering target whose trajectory and/or dynamic could change abruptly, the algorithm proposed is noted IMM-UKF. To extend this algorithm to multi-target case, we combined the later with the JPDA algorithm to ensure good data association.

Simulations show that the JPDA-IMM-UKF is a good investment while we are asked to track a highly maneuverable targets whose measurement and/or state models present a strong nonlinearities, and when there different trajectories cross each other.

REFERENCES

- [1] Y.Bar-Shalom and T.E.Fortman, "Tracking and Data Association," *Mathematics in Science and Engineering*, volume 179, Academic Press, 1988.
- [2] Y.Bar-Shalom and X. R. Li, "Estimation and Tracking, *Principles, Techniques and Software*," Artech House, Boston, MA (USA), 1993.
- [3] B.Espiau, F.Chaumette, and P.Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, 8(3):313-326, June 1992.
- [4] P.Danes, M.S.Djouadi, D.Bellot, "A 2-D Point-Wise Motion Estimation Scheme for Visual-Based Robotic Tasks," *7th International Symposium on Intelligent Robotic Systems (SIRS'99)*, Coimbra (Portugal), 20-23 July 1999, pp.119-128.
- [5] S.J. Julier, and J.K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of IEEE*, volume: 92, Issue: 3, March 2004, pages: 401-422.
- [6] S.J. Julier, and J.K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," *In International Symposium on Aerospace/Defense Sensing, Simulation and Control*, Orlando, FL, 1997.
- [7] Y.Bar-Shalom and X. R. Li, "Multi-target Multi-sensor Tracking: Principles and Techniques," *Storrs, CT*, YBS Publishing, 1995.
- [8] M.S.Djouadi, A.Sebbagh, D.Berkani, "IMM-UKF and IMM-EKF Algorithms for Tracking Highly Maneuvring Target," *Archive of Control Sciences*, Vol: 1, issue: 1, 2005.
- [9] M. Hadzagic, H. Mchalska, A. Jouan, "IMM-JVC and IMM-JPDA for closely maneuvering targets," *IEEE*, 2001.
- [10] Jitendra K. Tugnait, "Tracking of Multiple Maneuvering Targets in Clutter Using Multiple Sensors, IMM, and JPDA Coupled Filtering," *IEEE Transactions on Aerospace and Electronic Systems* Vol. 40, no. 1 January 2004.