



A Four-Model Based IMM Algorithm for Real-Time Visual Tracking of High-Speed Maneuvering Targets

Edwards Ernesto Sánchez-Ramírez¹ · Alberto Jorge Rosales-Silva¹ · Jean Marie Vianney-Kinani² · Rogelio Antonio Alfaro-Flores³

Received: 18 September 2017 / Accepted: 26 August 2018

© Springer Nature B.V. 2018

Abstract

In recent years, visual tracking algorithms based on state estimators have been developed in order to improve the performance during tracking tasks. However, this performance changes according to target type, object kinematics and scenario complexity. When working with high-speed maneuvering targets, tracking errors increase considerably due to low response of estimators as well as the kinematic mismatch between the real motion profile and the one assumed by the estimator. Some examples of objects that present this high-speed behavior are rockets, aircrafts and missiles. To overcome this visual tracking problem, this work proposes an interacting multiple model algorithm based on four kinematic models: constant velocity, constant acceleration, constant turn and thrust acceleration. We present three different scenarios with complex maneuvers for comparison study, and experimental results show that visual tracking is improved when using the proposed strategy.

Keywords Interactive multiple model · Visual tracking · Real-time · High-speed maneuvering

1 Introduction

Visual tracking has been an important tool in several applications related with surveillance, surgery, robotics, augmented reality, among others. As stated in [1], visual tracking is defined as detection, extraction, recognition and tracking of the moving objects in video sequences for further analysis.

Among the most recent contributions in the field is the one achieved by Fanxiang et al. [2]. They propose a real-time visual tracking algorithm where a novel kernel based multiple cue adaptive appearance model (KBMCAAM) is proved to be a robust discriminative tracker in challenging scenes. Pang et al. [3] show a Multiway Histogram Intersec-

tion (MHI) strategy for multi-target tracking optimized to provide affinity accuracy and computational efficiency. Can et al. [4] introduce a robust Scene-Adaptive strategy using hierarchical data association. This algorithm is applied to multiple objects tracking, showing very good performance in crowded and partial occlusion scenarios. Leizea et al. [5] propose a method to track deformable 3D objects in surgical environments. The purpose is to register deformations of nonrigid objects when a surgical robot is applying a force. This provides to surgeons an extra visual feedback. Hem et al. [1] propose an adaptive background strategy focused to perform real-time visual tracking with a high adaptivity and this demonstrates its robustness to appearance variations caused by occlusions, illumination changes and exposure variations, which are very useful for noise reduction during position measurements. Bertinetto et al. [6] show an efficient model which uses a simple combination of template and histogram scores that learn independently, suitable for soft real-time operations. Many other works are focused to track specific objects like in the one achieved by McDonagh et al. [7], where real-time face tracking is performed.

State estimators are useful while performing visual tracking tasks. Within this context, the IMM algorithm has proved to be robust in several applications like object tracking and signal reconstruction. It was initially proposed by

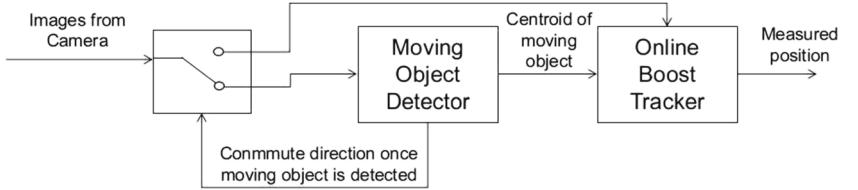
✉ Alberto Jorge Rosales-Silva
arosales@ipn.mx

¹ Instituto Politécnico Nacional de México, Escuela Superior de Ingeniería Mecánica y Eléctrica - Unidad Culhuacan, Av. Santa Ana 951, San Francisco Culhuacan, Culhuacan CTM, C.P. 04260, CDMX, México

² Instituto Tecnológico Superior de Huichapan S/N, Col. El Saucillo, C.P. 42411, Huichapan, Hidalgo, México

³ Instituto de Investigación y Desarrollo Tecnológico de la Armada de México, Veracruz, México

Fig. 1 Visual tracker based on detection and tracking stages



Henk [8], and then defined in Henk et al. [9] and Mazor et al. [10]. Nowadays, the IMM algorithm is still being used due to its interesting features: multiple model application and state interaction. In Farmer et al. [11] they present an IMM strategy for human motion tracking based on three kinematic models, and the obtained results show efficiency of IMM algorithm when tracking high speed motion. The work made by Dhassi et al. [12] is of special interest because they use a combination of the IMM algorithm with particle filters, in order to perform robust visual tracking. However, the performance of the particle filter is related with the number of particles, and consequently, with the computational cost. For real time applications this strategy could not be recommended. The work presented by Genovese [13] is also of special interest, where a scheme is proposed for accurate state estimation of maneuvering targets based on an IMM algorithm which uses three kinematic models: constant velocity (CV), constant acceleration (CA) and constant turn (CT). It is applied for three-dimension target tracking using radars as sensors, and obtained simulations show the effectiveness of the proposed strategy.

In this work, we have focused on visual tracking of high-speed maneuvering targets. We present a two-dimension IMM algorithm based on four kinematic models (IMM2D4M): constant velocity (CV), constant acceleration (CA), constant turn (CT) and thrust acceleration (TA). The latter model is added in order to improve real-time tracking, specially when working with such objects like rockets, aircrafts and missiles. The complex maneuvers of these targets make them hard to track because they are designed to perform evasive, offensive, and disengagement maneuvers. For real-time visual tracking, a multithreading processing scheme was made in order to achieve fast image processing and a high rate of frames per second (fps).

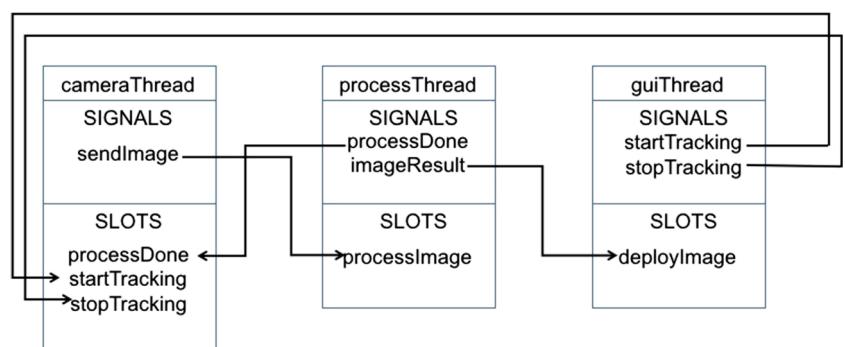
The structure of this work is as follows: Section 2 presents detection and tracking algorithms used from OpenCV libraries, and the real-time treatment based on multithreading processing. Section 3 presents the proposed IMM algorithm based on the four kinematic models (CV-CA-CT-TA). Section 4 presents three cases of study for performance comparison and a discussion of the obtained experimental results. Finally, Section 5 shows the concluding remarks of this work.

2 Detection, Tracking and Real-Time Handling

For object detection and tracking, we have used some algorithms from OpenCV libraries. A block diagram of visual tracking system implementation is shown in Fig. 1. The block of a single moving-objects detector receives images from a single camera and then performs image differentiation with consecutive frames. Image result is conditioned with morphological operations to analyze detected objects for centroid extraction. The obtained spatial coordinates agreeing to the pixels location are then sent to the stage where a tracking algorithm is performed and the moving object detector is deactivated. The tracking algorithm is based in the work proposed by Grabner et al. [14], and it is implemented as a function from OpenCV libraries ("Tracker Boost"). The tracker works after previous stage has detected a moving object. Measured position is then used by the IMM2D4M algorithm in order to improve visual tracking.

Real-time handling is performed using multithreading programming. Figure 2 shows working threads as well as the synchronization strategy. Within this visual tracking

Fig. 2 Thread synchronization



implementation, we have used three threads. One is used to capture images from a single camera, another one is used to perform IMM2D4M algorithm and the last one to manage a Graphical User Interface (GUI).

Threads are represented by boxes having their internal signals and slots. A signal is considered as a message which contains useful data, and a slot as a function that is activated when certain signal is emitted. The process *cameraThread* uses the signal *sendImage* to emit captured images. The slots *processDone*, *startTracking* and *stopTracking* are used as flags to indicate the beginning and the ending of tracking tasks. The process *processThread* uses the signal *processDone* to indicate that image processing is ready, and the signal *imageResult* to emit processed image. The slot *processImage* is a flag used to indicate that received image must be processed. The process *guiThread* uses the signals *startTracking* and *stopTracking* to indicate the beginning and the end of tracking process. They are activated by buttons in a graphical user interface. The slot *deployImage* is used to display the processed image.

3 The IMM2D4M Algorithm

As stated before, the IMM algorithm is still being used due to its interesting features for state estimation. In this section we present a short description of the algorithm as well as the description of the four used kinematic models.

3.1 IMM Stages

The IMM algorithm is composed of N dynamic models within a Markov chain, with a probability transition matrix T with elements p_{ji} [8–10]. Basically, it consists of four stages: state interaction, state estimation, model probability update and state estimate combination. In a discrete and recursive representation, we can identify the following equations:

1) State Interaction:

$$\mu_i^-(k) = \sum_{j=1}^N p_{ji} \mu_j(k-1), \quad (1)$$

$$\mu_{i|j}(k) = \frac{p_{ji} \mu_j(k-1)}{\mu_i^-(k)}, \quad (2)$$

$$\hat{x}_{0i}(k-1) = \sum_{j=1}^N \mu_{i|j}(k) \hat{x}_j(k-1), \quad (3)$$

$$P_{0i}(k-1) = \sum_{j=1}^N \mu_{i|j}(k) \left\{ P_j(k-1) + \dots \right. \\ \left. [\hat{x}_j(k-1) - \hat{x}_{0i}(k-1)] \dots \right. \\ \left. [\hat{x}_j(k-1) - \hat{x}_{0i}(k-1)]^T \right\}. \quad (4)$$

2) State Estimation (Kalman Filtering):

$$\hat{x}_i^-(k) = \Phi_i \hat{x}_{0i}(k-1), \quad (5)$$

$$P_i^-(k) = \Phi_i P_{0i}(k-1) \Phi_i^T + Q_i, \quad (6)$$

$$K_i = P_i^-(k) H_i^T \left(H_i P_i^-(k) H_i^T + R_i \right)^{-1}, \quad (7)$$

$$\hat{x}_i(k) = \hat{x}_i^-(k) + K_i [z_i(k) - H_i \hat{x}_i^-(k)], \quad (8)$$

$$P_i(k) = (I - K_i H_i) P_i^-(k), \quad (9)$$

$$v_i^-(k) = z_i(k) - H_i \hat{x}_i^-(k), \quad (10)$$

$$S_i^-(k) = H_i P_i^-(k) H_i^T + R_i. \quad (11)$$

3) Model probability update:

$$\Lambda_i(k) = \frac{1}{\sqrt{|2\pi S_i^-(k)|}} \exp \left[-0.5 (v_i^-(k))^T (S_i^-(k))^{-1} (v_i^-(k)) \right], \quad (12)$$

$$c(k) = \sum_{i=1}^N \Lambda_i(k) \mu_i^-(k), \quad (13)$$

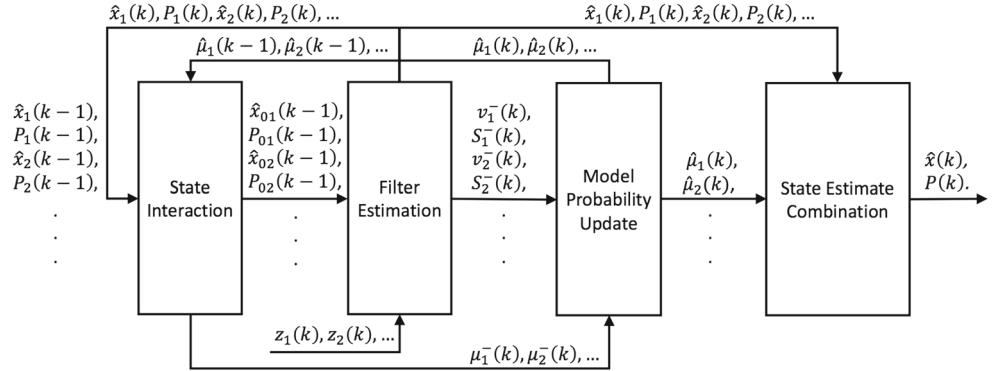
$$\hat{\mu}_i(k) = \frac{1}{c(k)} \Lambda_i(k) \mu_i^-(k). \quad (14)$$

4) Estimated States Combination:

$$\hat{x}(k) = \sum_{i=1}^N \hat{\mu}_i(k) \hat{x}_i(k), \quad (15)$$

$$P(k) = \sum_{i=1}^N \hat{\mu}_i(k) \left\{ P_i(k) + [\hat{x}_i(k) - \hat{x}(k)][\hat{x}_i(k) - \hat{x}(k)]^T \right\}, \quad (16)$$

where, for the i -th dynamic model, x_i is the state vector, x_{0i} the apriori vector state, μ_i the mode probability, $\mu_{i|j}$ the mixing probability, P_i the covariance of the error estimation, P_{0i} is the apriori covariance of the error estimation z_i the measurement vector, Φ_i the state transition matrix, Q_i the covariance of process noise, H_i the observation matrix, R_i the covariance of measurements noise, K_i the Kalman gain, Λ_i the likelihood function, v_i the innovation and S_i the innovation covariance. The notations $\hat{\cdot}$ and $\hat{\cdot}$ represent predicted and estimated values, respectively. The probability transition matrix T is composed by the elements p_{ij} . A block diagram of the IMM algorithm is presented in Fig. 3.

Fig. 3 The IMM algorithm

3.2 Kinematic Models

As mentioned before, we propose the integration of four kinematic models to the IMM algorithm. The description of each one of them is presented as follows. The coordinates (X, Y) represent the position of the tracked object in pixels.

1) Constant Velocity Model (CV) This model assumes that tracked object moves with constant velocity, disturbed by low acceleration noise. It has been widely used for state estimation of non-maneuvering targets. The value q_{CV} is a noise spectral density parameter, and has units of [m^2/s^3]. The value Δt represent time difference between actual and past measurement. The state vector \mathbf{x}_{CV} , the transition matrix Φ_{CV} and the covariance matrix for process noise \mathbf{Q}_{CV} are stated as follows:

$$\mathbf{x}_{CV} = [X \ \dot{X} \ Y \ \dot{Y}]^T, \quad (17)$$

$$\Phi_{CV} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (18)$$

$$\mathbf{Q}_{CV} = q_{CV} \Delta t \begin{bmatrix} \frac{\Delta t^2}{3} & \frac{\Delta t}{2} & 0 & 0 \\ \frac{\Delta t}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{\Delta t^2}{3} & \frac{\Delta t}{2} \\ 0 & 0 & \frac{\Delta t}{2} & 1 \end{bmatrix}. \quad (19)$$

2) Constant Acceleration Model (CA) This model assumes that tracked object moves with constant acceleration, disturbed by high jerk noise. It has been widely used for state estimation of targets with complex maneuvers. The value q_{CA} is a noise spectral density parameter, and has units of [m^2/s^5]. The state vector \mathbf{x}_{CA} , the transition matrix Φ_{CA} and the covariance matrix for process noise \mathbf{Q}_{CA} are stated as follows:

$$\mathbf{x}_{CA} = [X \ \dot{X} \ \ddot{X} \ Y \ \dot{Y} \ \ddot{Y}]^T, \quad (20)$$

$$\Phi_{CA} = \begin{bmatrix} 1 & \Delta t & \Delta t^2/2 & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \Delta t^2/2 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (21)$$

$$\mathbf{Q}_{CA} = q_{CA} \Delta t \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (22)$$

3) Constant Turn Model (CT) This model assumes that tracked object moves according to a constant turn with constant angular velocity ω , disturbed by low jerk noise. It has been successfully applied for three-dimension target tracking in the work made by Genovese [13]. In this case it will be applied to the two-dimension case. The main feature of this model is that it uses a pseudomeasurement to improve estimation of angular velocity. The derivation of this method is presented in the work made by Blair et al. [15]. In the IMM algorithm, the pseudomeasurement has to be performed twice: to improve residuals and to improve estimations. The value q_{CT} is a noise spectral

Table 1 Noise Spectral Density Parameters

Algorithm	Defensive Man.	Disengagement Man.	Offensive Man.
IMM2D2M	$q_{CV} = 1$ $q_{CA} = 450$	$q_{CV} = 1$ $q_{CA} = 450$	$q_{CV} = 1$ $q_{CA} = 450$
IMM2D3M	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 350$	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 350$	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 75$
IMM2D4M	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 350$ $q_{TA} = 25$	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 350$ $q_{TA} = 25$	$q_{CV} = 1$ $q_{CA} = 450$ $q_{CT} = 75$ $q_{TA} = 25$

Table 2 Probability Transition Matrixes

Algorithm	Prob. Transition Matrix
IMM2D2M	$T = \begin{bmatrix} 0.998 & 0.002 \\ 0.100 & 0.900 \\ 0.998 & 0.001 & 0.001 \end{bmatrix}$
IMM2D3M	$T = \begin{bmatrix} 0.050 & 0.900 & 0.050 \\ 0.001 & 0.001 & 0.998 \\ 0.997 & 0.001 & 0.001 & 0.001 \end{bmatrix}$
IMM2D4M	$T = \begin{bmatrix} 0.050 & 0.850 & 0.050 & 0.050 \\ 0.001 & 0.001 & 0.997 & 0.001 \\ 0.001 & 0.001 & 0.001 & 0.997 \end{bmatrix}$

density parameter, and has units of [m^2/s^5]. The state vector \mathbf{x}_{CT} , the transition matrix Φ_{CT} and the covariance matrix for process noise \mathbf{Q}_{CT} are stated as follows:

$$\mathbf{x}_{CT} = [X \dot{X} \ddot{X} Y \dot{Y} \ddot{Y}]^T, \quad (23)$$

$$\Phi_{CT} = \begin{bmatrix} \phi_{CT} & O_3 \\ O_3 & \phi_{CT} \end{bmatrix}, \quad (24)$$

$$\phi_{CT} = \begin{bmatrix} 1 & \omega^{-1} \sin(\omega\Delta t) & \omega^{-2}(1 - \cos(\omega\Delta t)) \\ 0 & \cos(\omega\Delta t) & \omega^{-1} \sin(\omega\Delta t) \\ 0 & -\omega^{-1} \sin(\omega\Delta t) & \cos(\omega\Delta t) \end{bmatrix}, \quad (25)$$

$$O_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (26)$$

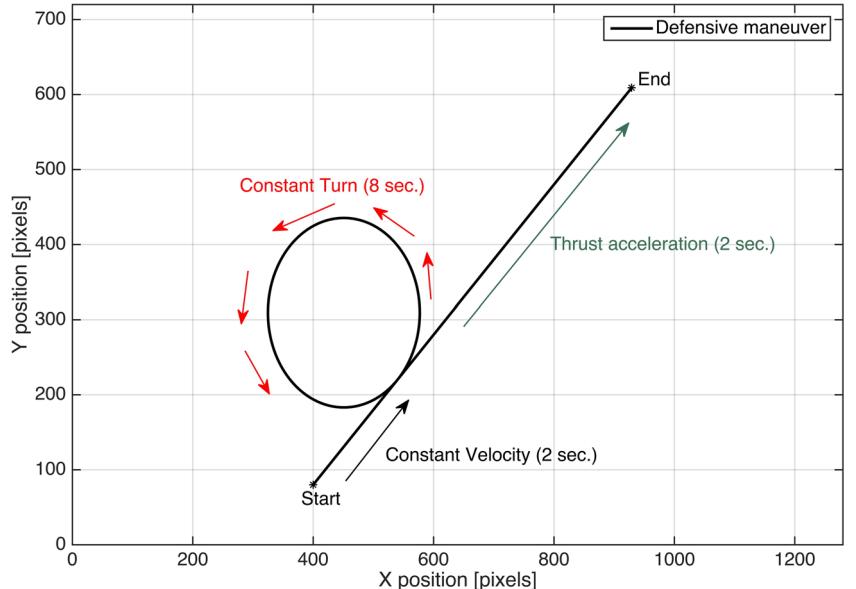
$$\mathbf{Q}_{CT} = q_{CT} \Delta t \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (27)$$

4) Thrust Acceleration Model (TA) This model is proposed to be added to the IMM algorithm in order to improve visual tracking. Here we assume that tracked object moves with constant acceleration in the velocity vector direction, disturbed by low jerk noise. Targets with high-speed maneuvers usually present this behavior because they are equipped with a propulsion mechanism. The value q_{TA} is a noise spectral density parameter, and has units of [m^2/s^5]. The state vector \mathbf{x}_{TA} , the transition matrix Φ_{TA} and the covariance matrix for process noise \mathbf{Q}_{TA} are stated as follows:

$$\mathbf{x}_{TA} = [X \dot{X} \ddot{X} Y \dot{Y} \ddot{Y}]^T, \quad (28)$$

$$\Phi_{TA} = \begin{bmatrix} 1 & e^{\Delta t} - 1 & 0 & 0 & 0 & 0 \\ 0 & e^{\Delta t} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & e^{\Delta t} - 1 & 0 \\ 0 & 0 & 0 & 0 & e^{\Delta t} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (29)$$

$$\mathbf{Q}_{TA} = q_{CA} \Delta t \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (30)$$

Fig. 4 A defensive maneuver

In the next section we present three cases of study in order to evaluate the performance of proposed algorithm IMM2D4M.

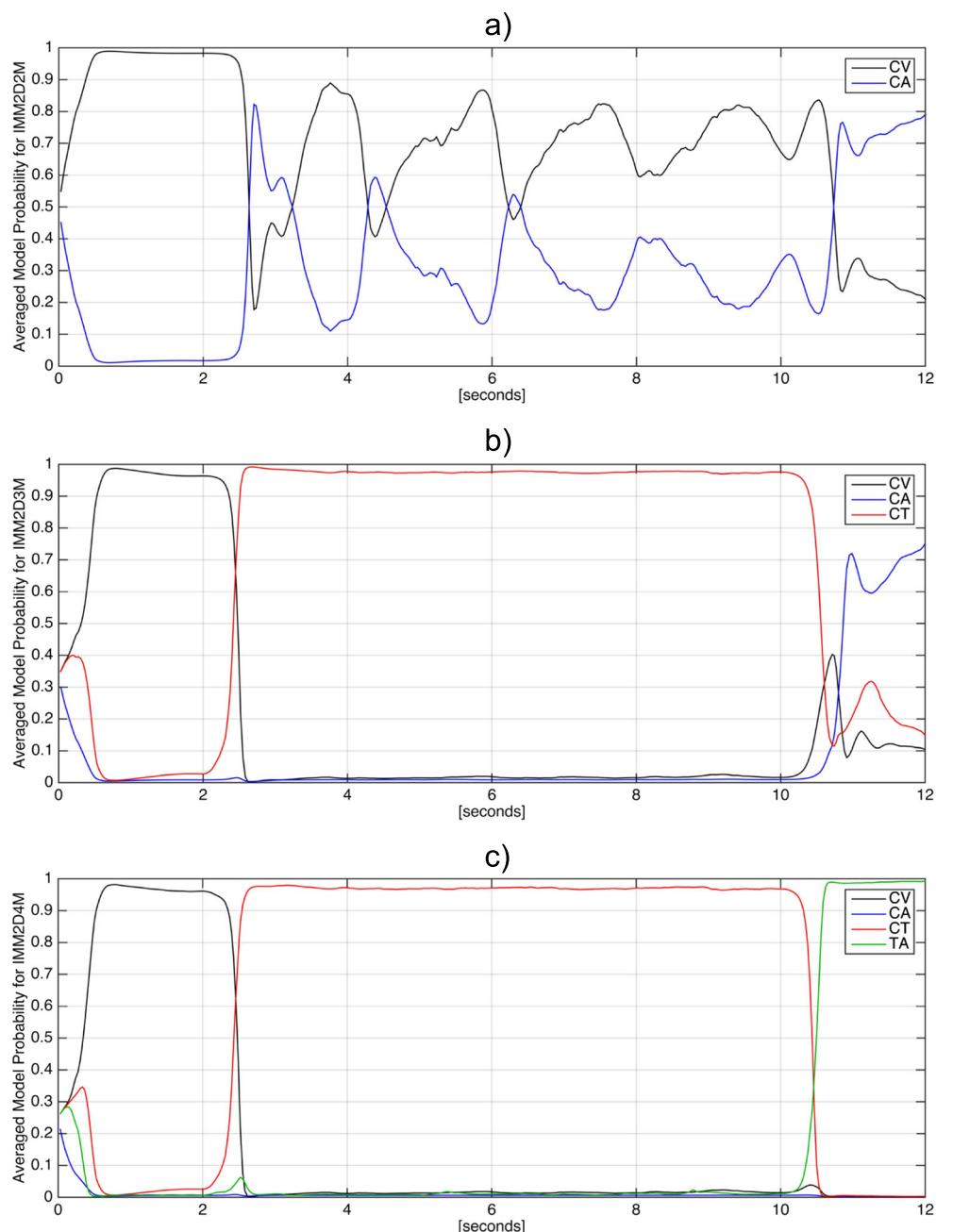
4 Cases of Study and Experimental Results

High-speed maneuvering targets perform complex movements and unexpected turns. Some examples of objects that present these behaviors are rockets, aircrafts and missiles. In this section we present three cases of study for performance evaluation, based in typical maneuvers performed by such

objects: a defensive maneuver, a disengagement maneuver and an offensive maneuver. They are tactical movements performed to gain a positional advantage over the oponent. The following tracking algorithms will be evaluated for comparison study:

- The proposed IMM2D4M algorithm.
- The IMM2D3M algorithm with three kinematic models (CV-CA-CT) proposed by Genovese [13] for accurate state estimation of maneuvering targets.
- A classical IMM2D2M algorithm with two kinematic models (CV-CA).

Fig. 5 Averaged model probabilities for **a** IMM2D2M, **b** IMM2D3M and **c** IMM2D4M



For real time visual tracking implementation, we have projected three video sequences according to the said maneuvers. A single camera was calibrated and used for acquiring images, and then the multithreading software scheme for object tracking was applied. The average value for image processing was 30 fps. The algorithms were tested 30 times for each case of study.

The performance criteria is based on comparison of visual tracking errors and Root Mean Square Errors (RMSE), given the following assumptions:

- Each algorithm is evaluated under the same noisy conditions. That is to say, when tracking the maneuvering target, all algorithms are implemented online using the same measurement z_k at time t_k .
- All algorithms use the same value for the covariance of measurements noise ($R_i = 25$ [pixels]).
- All algorithms use the same initial conditions for $\hat{x}(k)$ and $P(k)$.

Filter parameters Q_i and T were tunned in order to fit the model probabilities to current motion profile. The performance is improved when selecting the proper values according to motion profile, process noise, measurement noise among other considerations. The selected values for the noise spectral density parameters are shown in Table 1.

Table 3 Averaged RMSE Values

Algorithm	RMSE for X pos.	RMSE for Y pos.
IMM2D2M	6.93	6.32
IMM2D3M	5.38	5.10
IMM2D4M	4.96	4.86

The values for probability transition matrixes are shown in Table 2.

4.1 Defensive Maneuver

This tactical maneuver is used to evade air-to-air weapon attacks. In Fig. 4 is shown the whole 12-seconds maneuver in the image plane. It represents the trajectory of the maneuvering target centroid as a reference for performance evaluation. It begins with a constant velocity motion for two seconds, then a constant turn during eight seconds to evade attack and finally a thrust acceleration during two seconds to continue over the initial direction.

In Fig. 5 are shown the averaged model probabilities $\hat{\mu}_i(k)$ over 30 realizations, for each algorithm. The values were computed using the whole state vectors (Eqs. 17,

Fig. 6 Averaged visual tracking errors for **a** X dimension and **b** Y dimension

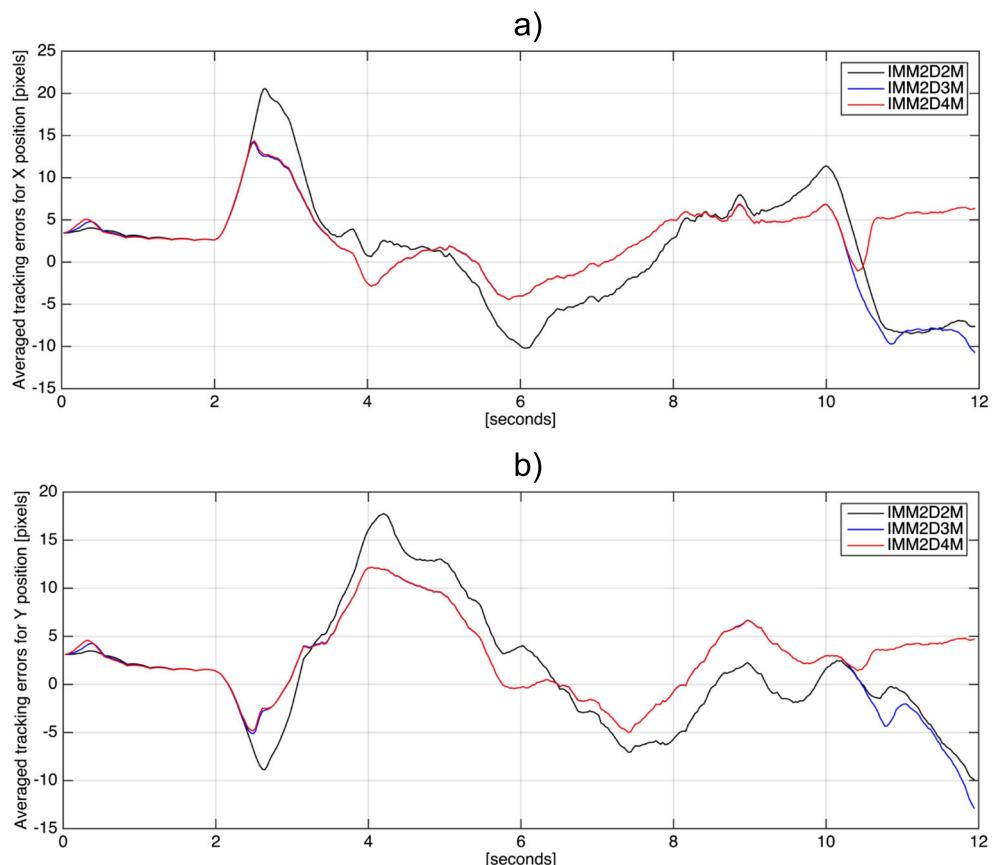


Fig. 7 Visual tracking results for each evaluated algorithm

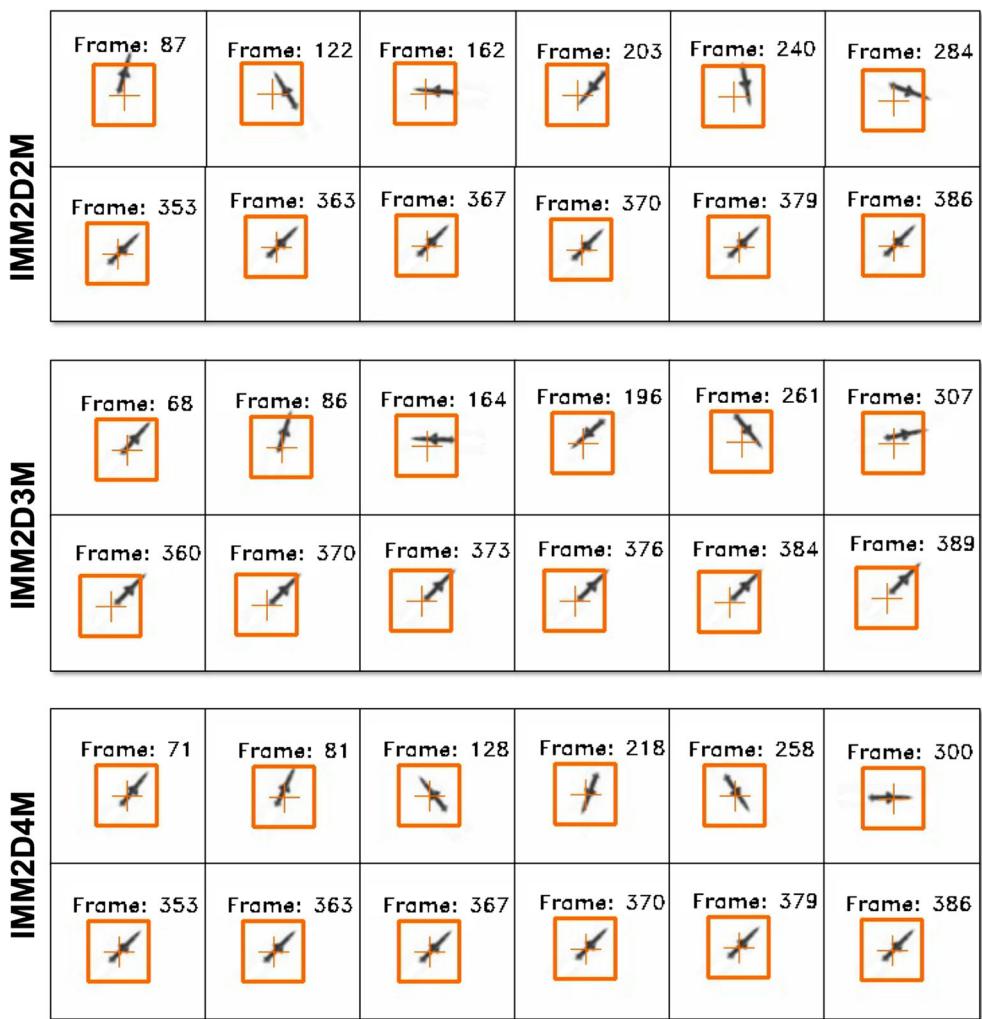
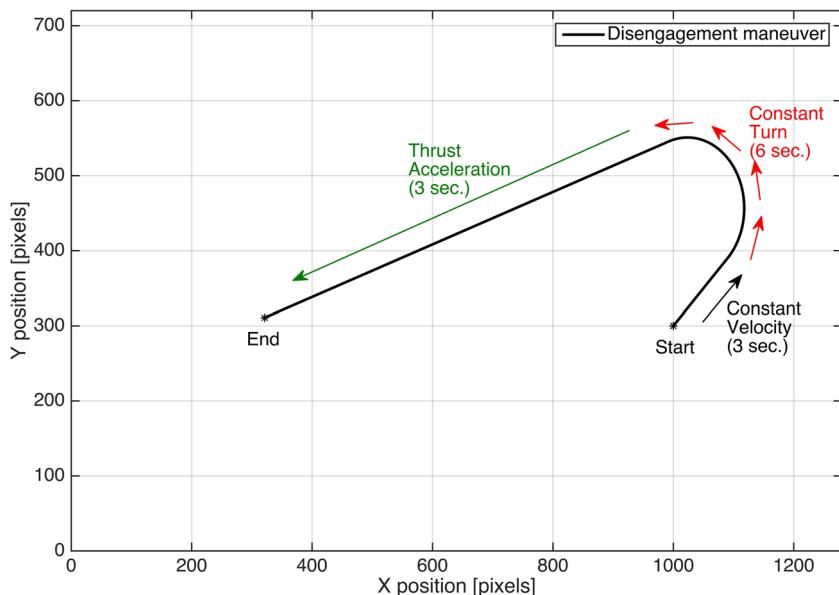


Fig. 8 A disengagement maneuver



20 and 23). Therefore, each sub-Figure applies for both dimensions (X, Y) using only a tracking algorithm. For the IMM2D2M algorithm, model probabilities fit properly to real motion profile only during constant velocity motion. During the rest of the maneuver, CA model is dominant for the constant turn, and CV model for thrust acceleration. For the IMM2D3M algorithm, model probabilities fit properly to real motion profile only during constant velocity and constant turn motion. During the rest of the maneuver, CA model is dominant. For the IMM2D4M algorithm, model

probabilities fit properly during the whole maneuver. The thrust acceleration model (TA) is highly dominant during the thrust maneuver, and it only applies for the IMM2D4M algorithm.

In Fig. 6 are shown the averaged visual tracking errors over 30 realizations for each algorithm. Error signals were obtained as a difference between the reference trajectory (Fig. 4) and the estimated centroid position (\hat{X}, \hat{Y}). Figure 6a shows tracking errors for X dimension and Fig. 6b for Y dimension in the image plane. Best results are

Fig. 9 Averaged model probabilities for **a** IMM2D2M, **b** IMM2D3M and **c** IMM2D4M

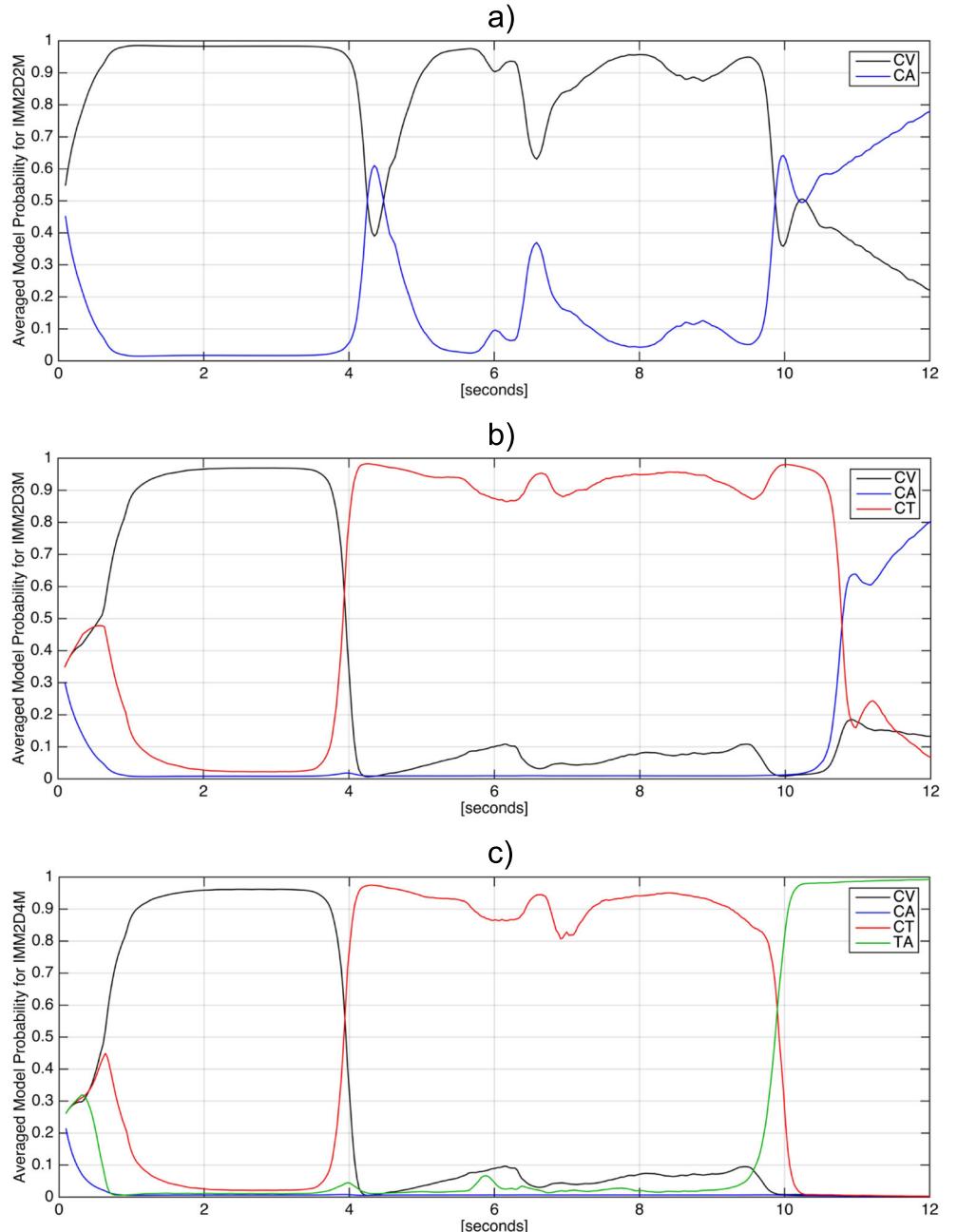
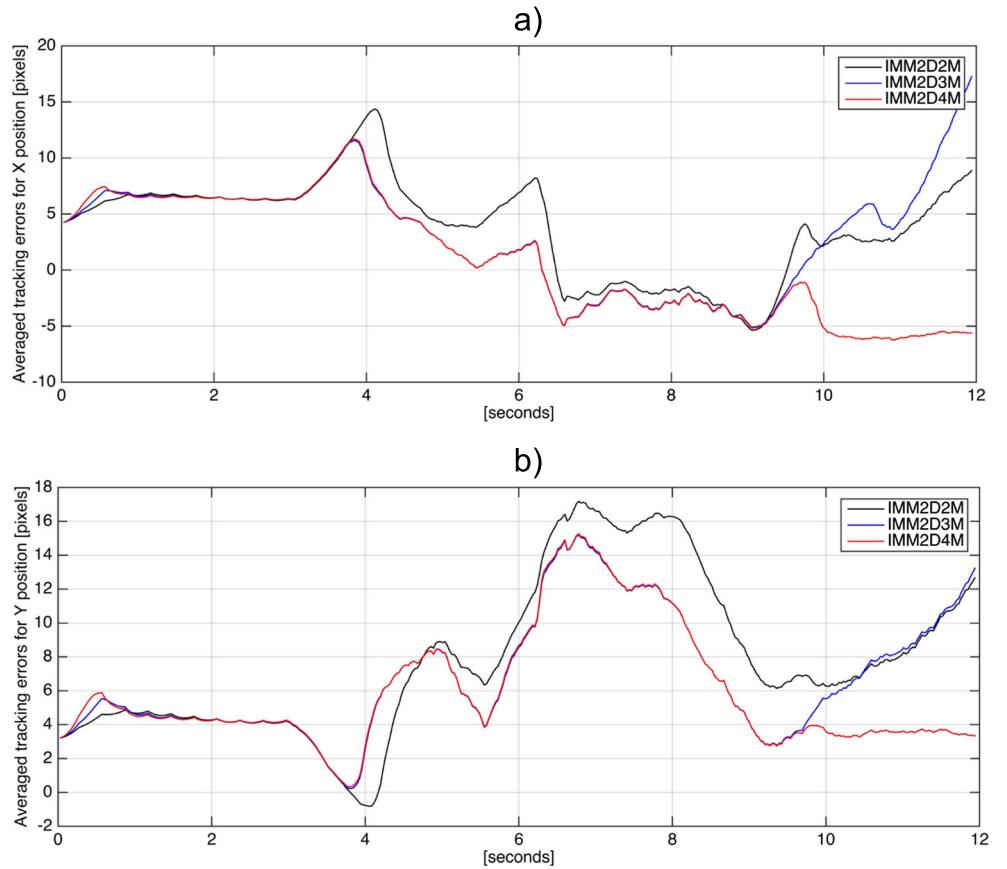


Fig. 10 Averaged visual tracking errors for **a** X dimension and **b** Y dimension



achieved with the IMM2D4M. Table 3 shows the averaged Root Mean Square Error (RMSE) values over 30 realizations for each algorithm.

Visual tracking results are shown in Fig. 7. The presented frames correspond to critical maneuvers of the target, that is during constant turn and thrust acceleration. The orange cross represents the estimated centroid position of the target. Algorithms IMM2D2M and IMM2D3M shows clear discrepancies between the real centroid of the object, and the estimated one. The algorithm IMM2D4M shows a better tracking performance because the estimated centroid is very close to the real one during the whole maneuver.

4.2 Disengagement Maneuver

This typical maneuver is used to help facilitate an escape. In Fig. 8 is shown the whole 12-seconds maneuver in the image

plane. It represents the trajectory of the maneuvering target centroid as a reference for performance evaluation. It begins with a constant velocity motion for three seconds, then a constant turn during six seconds to change direction, and finally a thrust acceleration during three seconds to escape from the field of view of the attacker.

In Fig. 9 are shown the averaged model probabilities $\hat{\mu}_i(k)$ over 30 realizations, for each algorithm. The values were computed using the whole state vectors (Eqs. 17, 20 and 23). Therefore, each sub-Figure applies for both dimensions (X, Y) using only a tracking algorithm. For the IMM2D2M algorithm, model probabilities fit properly to real motion profile only during constant velocity motion. For the IMM2D3M algorithm, model probabilities fit properly to real motion profile only during constant velocity and constant turn motion. For the IMM2D4M algorithm, model probabilities fit properly during the whole maneuver. Results show that the behavior of all model probabilities are similar to the previous case of study.

In Fig. 10 are shown the averaged visual tracking errors over 30 realizations for each algorithm. Error signals were obtained as a difference between the reference trajectory (Fig. 8) and the estimated centroid position (\hat{X}, \hat{Y}). Figure 10a shows tracking errors for X dimension and Fig. 10b for Y dimension in the image plane. Best results are

Table 4 Averaged RMSE Values

Algorithm	RMSE for X pos.	RMSE for Y pos.
IMM2D2M	5.82	9.12
IMM2D3M	5.86	7.71
IMM2D4M	5.38	6.97

Fig. 11 Visual tracking results for each evaluated algorithm

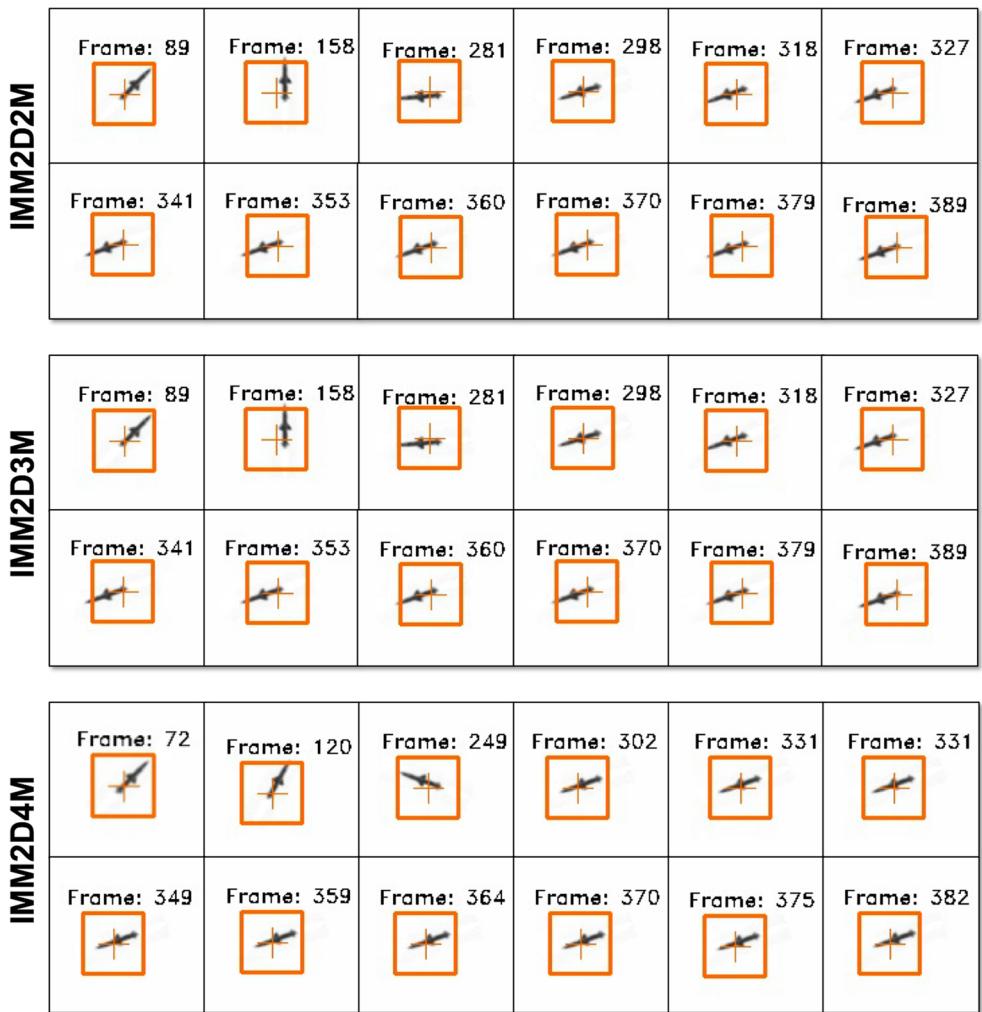


Fig. 12 An offensive maneuver

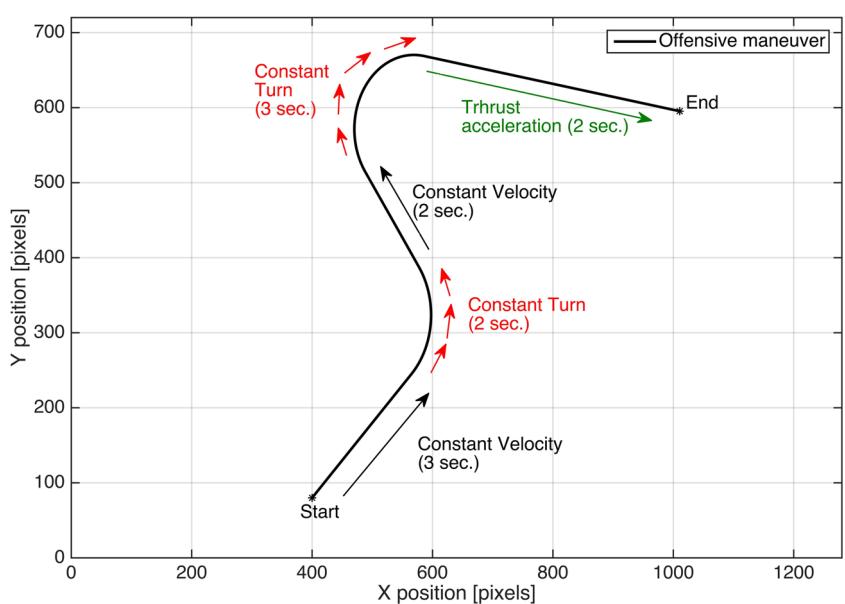
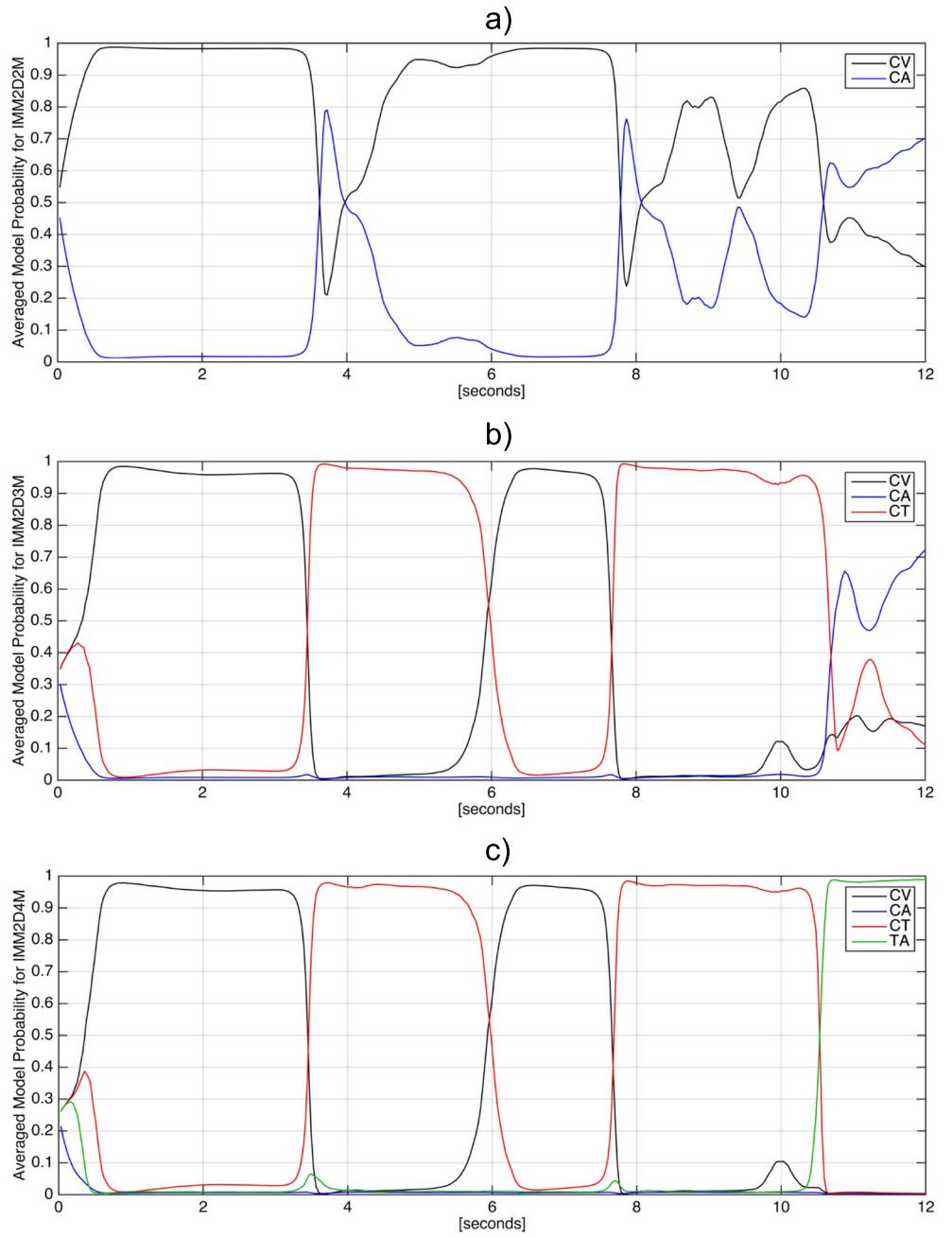


Fig. 13 Averaged model probabilities for **a** IMM2D2M, **b** IMM2D3M and **c** IMM2D4M



achieved with the IMM2D4M. Table 4 shows the averaged Root Mean Square Error (RMSE) values over 30 realizations for each algorithm.

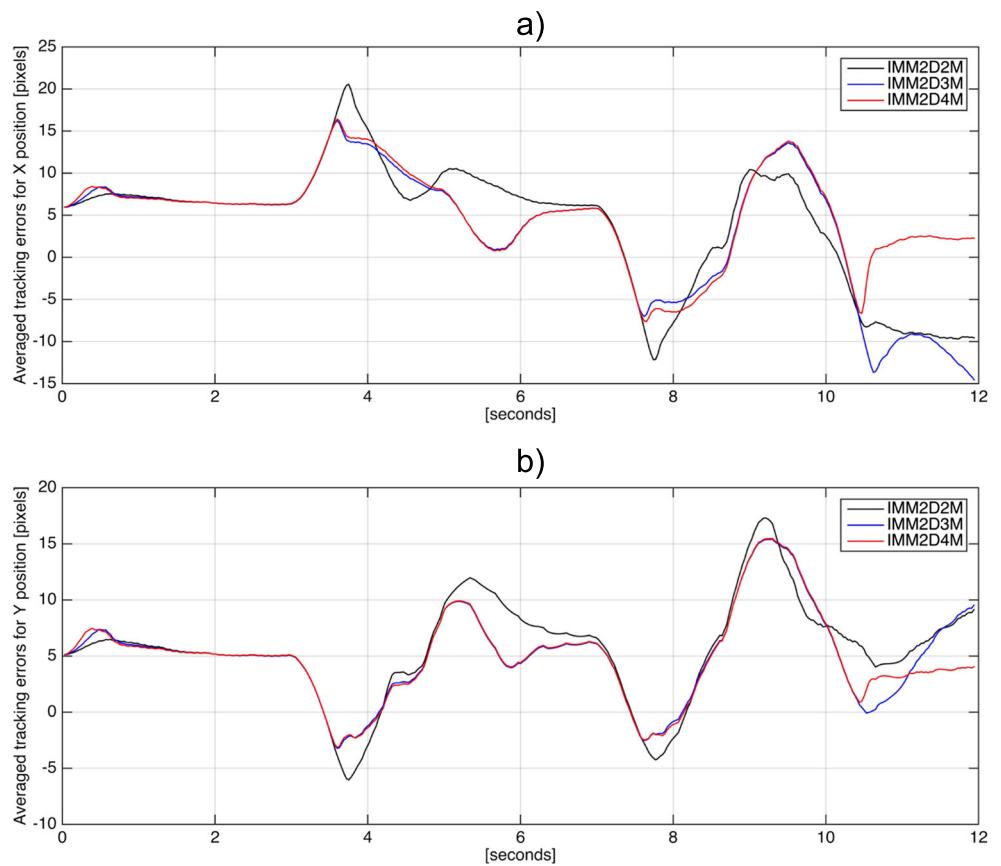
Visual tracking results are shown in Fig. 11. The presented frames correspond to critical maneuvers of the target, that is during constant turn and thrust acceleration. The orange cross represents the estimated centroid position of the target. In this case, algorithms IMM2D2M and IMM2D3M also shows clear discrepancies between the real

centroid of the object and the estimated one, especially during the thrust maneuvers. The algorithm IMM2D4M shows a better tracking performance because the estimated centroid is very close to the real one during the whole maneuver.

4.3 Offensive Maneuver

This is a typical maneuver used to help an attacker get behind an enemy. In Fig. 12 is shown the whole 12-seconds

Fig. 14 Averaged visual tracking errors for **a** X position and **b** Y position



maneuver in the image plane. It represents the trajectory of the maneuvering target centroid as a reference for performance evaluation. It begins with a constant velocity motion for three seconds, then a constant turn during two seconds to change direction, then a constant velocity motion for two seconds, once again a constant turn during two seconds to change direction and finally a thrust acceleration during two seconds to hit or attack a target.

In Fig. 13 are shown the averaged model probabilities $\hat{\mu}_i(k)$ over 30 realizations, for each algorithm. The values were computed using the whole state vectors (Eqs. 17, 20 and 23) and each sub-Figure applies for both dimensions (X, Y). For the IMM2D2M algorithm, model probabilities fit properly to real motion profile only during the first constant velocity motion. For the IMM2D3M algorithm,

model probabilities fit properly to real motion profile only during constant velocity and constant turn motion. For the IMM2D4M algorithm, model probabilities fit properly during the whole maneuver. Results show that the thrust acceleration model (TA) is highly dominant during the thrust maneuver, and it only applies for the IMM2D4M algorithm as in the previous case.

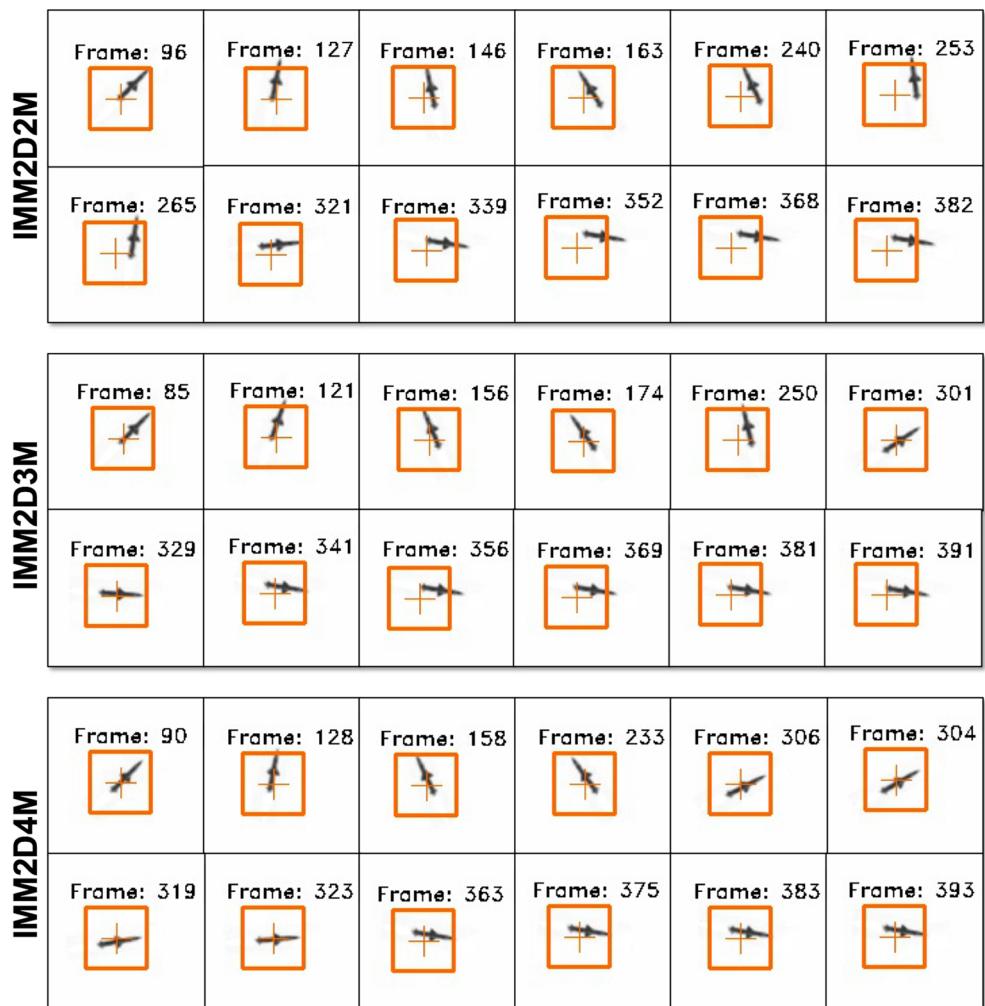
In Fig. 14 are shown the averaged visual tracking errors over 30 realizations for each algorithm. Error signals were obtained as a difference between the reference trajectory (Fig. 12) and the estimated centroid position (\hat{X}, \hat{Y}). Figure 14a shows tracking errors for X dimension and Fig. 14b for Y dimension in the image plane. Best results are achieved with the IMM2D4M. Table 5 shows the averaged Root Mean Square Error (RMSE) values over 30 realizations for each algorithm.

Visual tracking results are shown in Fig. 15. The presented frames correspond to critical maneuvers of the target, that is during constant turn and thrust acceleration. The orange cross represents the estimated centroid position of the target. In this case, algorithms IMM2D2M and IMM2D3M also shows clear discrepancies between the real centroid of the object and the estimated one. The

Table 5 Averaged RMSE Values

Algorithm	RMSE for X pos.	RMSE for Y pos.
IMM2D2M	8.34	7.18
IMM2D3M	8.22	6.46
IMM2D4M	7.42	6.30

Fig. 15 Visual tracking results for each evaluated algorithm



performance of said algorithms is considerably reduced during the thrust maneuver. The algorithm IMM2D4M shows a better tracking performance because the estimated centroid is very close to the real one during the whole maneuver.

5 Conclusion

In this work, we have presented a four-model IMM algorithm designed for real-time visual tracking of high-speed maneuvering targets. For comparison study, we presented three different scenarios with typical high-speed maneuvers. Performance of proposed strategy was compared against two IMM algorithms: a three-model based (IMM2D3M) and a two-model based (IMM2D2M).

When applying the performance criteria, the IMM2D4M algorithm had the lower RMSE values in the three presented study cases (Tables 3, 4 and 5). The interpretation of this result is that this algorithm had the best performance.

The accurate visual tracking of proposed algorithm is associated to its thrust acceleration model which permits a robust state estimation during the thrust maneuver of the target. The algorithms IMM2D3M and IMM2D2M had similar results, but their performance was considerably reduced during thrust maneuvers.

Acknowledgments The authors would like to thank the Instituto Politécnico Nacional de México, CONACyT and Instituto de Investigación y Desarrollo Tecnológico de la Armada de México for funding this project.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

- Hem, L., Daiqin, Y., Zhenzhong, C.: Adaptive background for real-time visual tracking. In: International Conference on Multimedia and Expo, Seattle, USA (2016)

2. Fanxiang, Z., Xuan, L., Zhitong, H., Yuefeng, J.: Kernel based multiple cue adaptive appearance model for robust real-time visual tracking. *IEEE Signal Processing Letters* **20**(11), 1094–1097 (2013)
3. Pang, Y., Shi, X., Jia, B., Blasch, E., Sheaff, C., Pham, K., Chen, G., Ling, H.: Multiway Histogram Intersection for Multi-target Tracking. In: 18th International Conference on Information Fusion, Washington DC, pp. 1938–1945 (2015)
4. Can, W., Hong, L., Yuan, G.: Scene-Adaptive Hierarchical data association for multiple objects tracking. *IEEE Signal Processing Letters* **21**(6), 697–701 (2014)
5. Leizea, I., Mendizabal, A., Alvarez, H., Aguinaga, I., Borro, D., Sanchez, E.: Real-time visual tracking of deformable objects in robot assisted surgery. *IEEE Comput. Graph. Appl.* **37**(1), 56–68 (2017)
6. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.: Staple: Complementary learners for Real-Time tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Nevada, USA, pp. 1401–1409 (2016)
7. McDonagh, S., Klaudiny, M., Bradley, D., Beeler, T., Matthews, I., Mitchell, K.: Synthetic prior design for real-time face tracking. In: 4th International Conference on 3D Vision, California, USA, pp. 639–648 (2016)
8. Henk, A.: An Efficient Filter for Abruptly Changing Systems. In: IEEE 23rd Conference on Decision and Control, Las Vegas, Nevada (1984)
9. Henk, A., Yaakov, B.: The interacting multiple model algorithm for systems with markovian switching coefficients. *IEEE Trans. Automatic Control* **33**(8), 780–783 (1988)
10. Mazor, E., Aberbuch, A., Bar-Shalom, Y., Dayan, J.: Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerospace Electronic Syst* **34**(1), 103–123 (1998)
11. Farmer, M., Hsu, R., Jain, A.: Interacting multiple model (IMM) Kalman filters for robust high speed human motion tracking. In: I16th International Conference on Pattern Recognition, Quebec, Canada (2002)
12. Dhassi, Y., Aarab, A., Alfidi, M.: IMM-PF visual tracking based on multi-feature fusion. In: 11th International Conference on Intelligent Systems: Theories and Applications (SITA), Mohammedia, Morocco (2016)
13. Genovese, A.: The Interacting multiple model algorithm for accurate state estimation of maneuvering targets. *Johns Hopkins Apl Technical Digest* **22**(4), 614–623 (2001)
14. Grabner, H., Grabner, M., Bischof, H.: Real-time tracking via on-line boosting. In Proc BMVC **1**, 47–56 (2006)
15. Blair, W., Watson, G., Alouani, A.: Use of kinematic constraint for tracking constant speed maneuvering targets, INAVSWCTR 91-561, Naval Surface Warfare Center, Dahlgren, VA (1991)

Edwards Ernesto Sánchez-Ramírez is a PhD student at Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacán, Instituto Politécnico Nacional, México. He received a Master of Science degree in Electronics Engineering from Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco in 2015. His work has focused in the development of control systems, Kalman filtering based estimators, visual servoing and visual tracking. His e-mail address is esanchezr1604@alumno.ipn.mx.

Alberto Jorge Rosales-Silva PhD in Communications and Electronics (2008), Master of Science in Telecommunications Engineering (2004) and Engineer Communications and Electronics (1999) by ESIME-Instituto Politécnico Nacional de México. Topics currently being developed are: image processing, multispectral and multichannel video in real time, visual computing, as well as medical image processing.

Jean Marie Vianney-Kinani received a BSc with honors from the University of Nairobi in 2006 and a PhD in Communication and Electronics Engineering from the Instituto Politécnico Nacional, México in 2016, where his research was supported by a Mexican council of Science and Technology. He is currently an Assistant Professor in the Department of Computer Science at the Instituto Tecnológico Superior de Huichapan, México. His research interests are in computer vision, natural language processing, and computational modeling of human vision and analysis of biological images.

Rogelio Antonio Alfaro-Flores received his Master of Science degree in Computer Sciences from Instituto Nacional de Astrofísica, Óptica y Electrónica, México, in 2006. His research interests include image processing and tracking by vision-based control.