Ground target tracking and classification in an unattended wireless sensor network

Benjamin Pannetier[a], Julien Moras[a], Jean Dezert[a], Loic Canevet[b] and Didier Cosson[b]

[a] ONERA, The French Aerospace Lab, Information Processing Department, F-91761 Palaiseau, France.

[b] DGA Techniques Terrestres, Rocade Est - Echangeur de Guerry, 18021 BOURGES Cedex

ABSTRACT

In this paper, we address the problem of multiple ground target tracking and classification with data from an unattended wireless sensor network. A multiple target tracking algorithm, taking into account the road and vegetation information, is studied in a centralized architecture. Despite of efficient algorithms proposed in the literature, we must adapt a basic approach to satisfy embedded processing. The algorithm enables tracking human and vehicles driving both on and off road. Based on our previous works, we integrate road or trail width and vegetation cover, in motion model to improve performance of tracking under constraint. Our algorithm also presents different dynamic models, to palliate the maneuvers of targets including a stop motion model. In order to handle realistic ground target tracking scenarios, the tracking algorithm is integrated into an operational platform (named fusion node) which is an autonomous smart computer abandoned in the surveillance area. After the calibration step of the heterogeneous sensor network, our system is able to handle real data from a wireless ground sensor network. The performance of system is evaluated in a real exercise for Forward Operating Base (FOB) protection and road surveillance.

Keywords: Unattended wireless sensor network, ground target tracking, map information, classification fusion.

1. INTRODUCTION

We present results obtained in an operational exercise with a wireless sensor network (WSN) prototype which consists of a large number of smart heterogeneous sensors with onboard sensing, processing and wireless communication capabilities. The WSN must satisfy severe exigencies in term of survivability (few months), low communications (to be undetectable by communication interception system), and must build in real-time the tactical situation assessment picture for large surveillance areas. The use of WSN network must also be easy and remotely controllable and have a low cost. The system must be easy to deploy, implemented by a limited number of operators with a minimum training through a simple human machine interface (HMI) for its exploitation and for decision-making support. Finally, the system must be modular, flexible and dynamically configurable (depending on the environment, the threat and mission).

Further author information: (Send correspondence to B.P.)
B.P.: E-mail: benjamin.pannetier@onera.fr,
J.D.: E-mail: jean.dezert@onera.fr,
J.M.: E-mail: julien.moras@onera.fr,
L.C.: E-mail: loic.canevet@intradef.gouv.fr,
D.C.: E-mail: didier.cosson@intradef.gouv.fr

Our demonstrator allows to study automatic data processing to fuse detections and create track to follow targets on the battlefield. It will allow us to evaluate several schemes for the data collection and fusion process and to demonstrate the necessity of taking into account high-level information (typically geographic information, as traffic lanes, intersections, areas without terrain obscuration,. . .) for deployment and exploitation of the system.

Several processing levels are considered in this work:

- local processing of raw data at the sensor level: it can provide a detection alert on the presence of a target, and eventually some attributes about the target (as target location and type);
- additional processing on raw data (as basic image processing on sensor nodes);
- data fusion on a sensor node from a set of information collected from other sensors (target kinematics (e.g. tracks), classifications, their number, etc).

In this paper, we study the problem of tracking multiple moving objects observed through a WSN with limited sensing abilities. The aim is to adapt and evaluate a conventional multiple target tracking algorithm in order to maintain high track continuity performance to provide a reliable situation assessment. For this goal, we use heterogeneous sensors to compensate the low amount of data available (due to the weak sensor area coverage) by a better information quality on the data (both in precision of location and in classification information). The proposed data sensor processing presented in this work allows to meet the operational constraints. The originality of the material presented in this paper resides mainly in the application context and not in the algorithm itself.

Several papers have been published on operational sensor processing applied to WSN. For example, Ekman and Pålson described in[1] a modified particle filter (PF)[2] to track a single vehicle through the WSN. A similar approach can be found in[3] . Despite of the well known estimation performances due to the generation of the particles on the road network, we haven't selected a PF algorithm because we need to track several targets in the sensor network with severe processing constraints due to hardware solution used in our demonstrator to preserve the power of a fusion node. In fact, because PF approach uses more CPU than Kalman filter (KF), extended Kalman filter (EKF) or unscented Kalman filter (UKF), we cannot use it in our specific context if one wants to make the surveillance system operational during a long period of time. Parmar and Zaveri in[4] have done similar studies and achieved the same conclusions. They did focus their study of the data association for MTT (Mutliple Target Tracker) in WSN and of the need to limit the power to maintain the WSN in activity during a long time. However, in future work, if the hardware performances improvements allow to satisfy the power constraint, the use of PF will become feasible. In fact, Oh and al.[5] described a complete PF algorithm (called MCMCDA algorithm) applied for tracking multiple targets in a WSN with communication constraints. To improve the MTT algorithm performance, we introduce in this work the geographic information in the tracking process as proposed by Ulmke and Koch[6] . Since we are interested by tracking both ground vehicles (that can move on and off the road), aerial vehicles that are not constrained on the road, and pedestrians as well, we have to consider on-road tracking as well as an off-road tracking algorithms. For doing this, we have adapted the MTT ground target tracking algorithm described in[7] for our WSN tracking demonstrator.

The paper is organized as follows: in section 2 the WSN is briefly presented. Section 3 describes algorithm to multiple target tracking with classification fusion. Section 4 presents the used track segment association technique to correlate tracks when targets evolve through lacunar areas (*i.e.* large

areas without smart sensors coverage). Results obtained on real exercise are given in section 5. Finally, concluding remarks are presented in section 6.

2. WIRELESS SENSOR NETWORK DESCRIPTION

2.1 Architecture

The good quality of communication between the sensor nodes has a strong impact on the ability of WSN to fulfill its task of surveillance. It is also very important that the WSN can communicate with the Command and Control (C2) station. The solution proposed in this paper is based on on-the-shelf existing components. Its multi-cluster architecture is represented in figure 1.

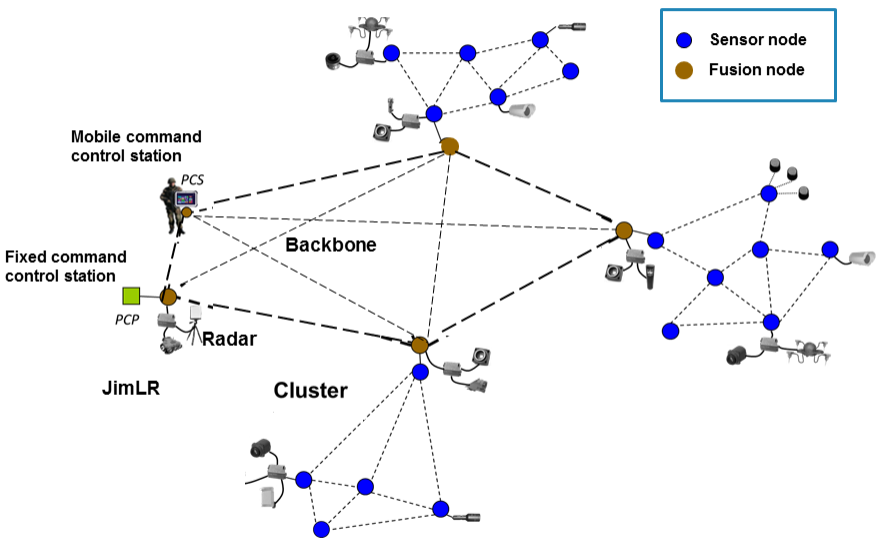

Figure 1: Sensor network architecture.

This architecture is structured in two levels:

- a set of clusters: sensor and fusion nodes connected through a low energy, low Rate 802.15.4 wireless network, managed by a gateway;
- a backbone with higher rate gathering data from clusters which guarantees the expected connectivity and allows two-ways communications.

The main information transmitted on the network are the following: data from sensor to sensor-nodes and to C2, state of the components to sensor node and to C2, command to sensors from C2 or sensor node to components, exchange between sensor nodes to allow data fusion. Two categories of sensors: low consumption sensors that can be kept in operation to provide a continuous surveillance, and sensors

having higher consumption that can be activated in case of presence of a target to acquire more detailed information on it.

The sensor node receives data from other sensors, processes them and transmits the local result to the fusion node. These information will be used in order to detect the presence of a target, to provide a spatial location of the event, and to classify the detection.

2.2 Geographical information

The GIS (Geographic Information System) used in this work contains the following information: the segmented road network, the DTED (Digital Terrain Elevation Data), the vegetation area coverage, the hydro-graphic layer, the building layer. The road network is connected and each road segment is indexed by the road section it belongs to. A road section is defined by a finite set of connected road segments delimited by a road end or a junction. For the topographic information, we use the database called: BD TOPO*. This GIS has a metric precision on the roads segments location. At the beginning of a surveillance battlefield operation, a TCF (Topographic Coordinated Frame) and its origin O are chosen in the manner that the axes X, Y and Z are respectively oriented in the east, north and up local direction. The target tracking process is carried out in the TCF. In addition, starting from the DTED and the sensor location at the current time, it is possible to compute the perceivability (noted P_e) at any point of the DTED. A function named $P_e(x, y, k)$ indicates if the pixel of the DTED at the location (x, y) is observable by the sensor or not at time k.

2.3 Sensors model

The generic sensor j observation model is given by:

$$\mathbf{z}_k^j = h^j(\mathbf{x}_k) + \mathbf{b}_k^j \tag{1}$$

where $h^j(\cdot)$ is the observation function, $\mathbf{x}_k$ is the state of a target (detailed in the next section), and $\mathbf{b}_k^j$ is a zero-mean white Gaussian noise vector with a known covariance matrix $\mathbf{R}_k^j$. The observation function and the associated noise depend on the type of sensor. We distinguish three observations functions: h^{radar}, h^{acou}, h^{optro}, h^{mag} associated respectively to the radar, acoustic, optic and magnetic sensors.

$$h^{radar}(\mathbf{x}_k) = \begin{bmatrix}\rho_k & \theta_k & \dot{\rho}_k\end{bmatrix}' \tag{2}$$

$$h^{acou}(\mathbf{x}_k) = \begin{bmatrix}\theta_k\end{bmatrix} \tag{3}$$

$$h^{opt}(\mathbf{x}_k) = \begin{bmatrix}\theta_k & \phi_k\end{bmatrix}' \tag{4}$$

$$h^{mag}(\mathbf{x}_k) = \begin{bmatrix}x_k & y_k\end{bmatrix}' \tag{5}$$

where θ_k, ϕ_k, ρ_k and $\dot{\rho}_k$ denote respectively bearing elevation, distance and range radial velocity in the sensor reference frame. For the magnetic sensor, we use its own location in the TCF in order to model a measurement because of its short range detection (as shown in Table 1). The different types of sensor used during exercises are listed in Table 1.

The *Volume* indicates the area coverage where the target can be found. This event is emitted as well as measurement to the fusion node in order to correlate this information with another volume, or a sensor detection to get a localized detection in the topographic coordinated frame (TCF). This

*See http://professionnels.ign.fr/bdtop for a description of this GIS.

detection is named a *composite* report, and it is the result of volume intersection between two bearing-only sensors. The result is an ellipsoid included in the volume intersection. The covariance matrix $\mathbf{R}_k^{comp}$ is associated to the ellipsoid and the measurement $\mathbf{z}_k^{comp}$ is the center of the volume intersection expressed in the TCF.

We consider also the probabilities $\mathbf{c}_k^j$ from $\mathbf{C}$ as input parameters of our tracking systems characterizing the global performances of the classifier. The values $\mathbf{c}_k^j$ are the outputs of the classifier. Where the classification reference frame is similar to the set of heterogeneous sensor. It is defined by

$$\mathbf{C} \triangleq \{\textit{light-vehicle}, \textit{heavy-vehicle}, \textit{tracked-vehicle}, \textit{human}, \textit{people}, \textit{aerial targets}\} \tag{6}$$

We denote by $\mathbf{z}_k^{j\star}$ the extended measurement of sensor j including both kinematic part and attribute part defined by

$$\mathbf{z}_k^{j\star} \triangleq \{\mathbf{z}_k^j, \mathbf{c}_k^j\} \tag{7}$$

For notation convenience, the measurements sequence $Z^{k,l} \triangleq \{Z^{k-1,n}, \mathbf{z}_k^{j\star}\}$ represents a possible set of measurements generated by the target up to time k. $Z^{k,l}$ consists in a subsequence $Z^{k-1,n}$ of measurements up to time $k-1$ and a validated measurement $\mathbf{z}_k^{j\star}$ available at time k associated with the track $\mathcal{T}^{k,l}$. At the current time k, the track $\mathcal{T}^{k,l}$ is represented by a sequence of the state estimates.

Sensor type	number	sensor node output	detection characteristic	comments
acoustic antenna	3	θ_k $\mathbf{c}_k$	Spherical $<$ 200 m	Use to update existing track Use to create composite report with another sensor Use to classify track
magnetic	10	Volume $\mathbf{c}_k$	Cartesian $<$ 1 m	Use to update existing track Use to initialize track Use to classify track
radar	1	ρ_k, θ_k,$\dot{\rho}_k$	Sectoral = 90° $<$ 1000 m	Use to update existing track Use to initialize track
PIR	8	θ_k	mono,multi beam $<$ 200 m	Use to update existing track Use to create composite with another sensor
cameras: UIR FIR short Cham	9 4 1 4	θ_k,ϕ_k $\mathbf{c}_k$	Sectoral = 30° , 40° $<$ 100,200 m	Use to update existing track Use to create composite report with another sensor Use to classify track

Table 1: Sensor types used in the demonstrator.

3. DATA FUSION ALGORITHMS

3.1 Context constraint tracking

The target state at the current time t_k is defined in the local horizontal plane of the TCF by the vector:

$$\mathbf{x}_k \triangleq [x_k \ \dot{x}_k \ y_k \ \dot{y}_k]^T \tag{8}$$

where (x_k, y_k) and $(\dot{x}_k, \dot{y}_k)$ define respectively the target location and velocity in the local horizontal plane.

The dynamics of the target evolving on the road are modeled by a first-order plant equation. The target state on the road segment s is defined by $\mathbf{x}_k^s$ where the target position (x_k^s, y_k^s) belongs to the road segment s and the corresponding heading $(\dot{x}_k^s, \dot{y}_k^s)$ in its direction.

The event that the target is on road segment s is noted $e_k^s = \{\mathbf{x}_k \in s\}$. Given this event e_k^s and according to a motion model $\mathcal{M}_i$, the estimation of the target state can be improved by considering the road segment s. For a constant velocity motion model, it follows:

$$\mathbf{x}_k^s = \mathbf{F}^{s,i}(\Delta_k) \cdot \mathbf{x}_{k-1}^s + \mathbf{\Gamma}(\Delta_k) \cdot \mathbf{v}_k^{s,i} \tag{9}$$

where Δ_k is the sampling time, $\mathbf{F}^{s,i}$ is the state transition matrix associated to the road segment s and adapted to a motion model $\mathcal{M}_i$; $\mathbf{v}_k^{s,i}$ is a white zero-mean Gaussian random vector with covariance matrix $\mathbf{Q}_k^{s,i}$ chosen in such a way that the standard deviation σ_d along the road segment is higher than the standard deviation σ_n in the orthogonal associated to the road width (given by the GIS). It is defined by:

$$\mathbf{Q}_k^{s,i} = \mathbf{R}_{\theta_s} \cdot \begin{pmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_n^2 \end{pmatrix} \cdot \mathbf{R}_{\theta_s}^T \tag{10}$$

where $\mathbf{R}_{\theta_s}$ is the rotation matrix associated with the direction θ_s defined in the plane (O, X, Y) of the road segment s. The matrix $\mathbf{\Gamma}(\Delta_k)$ is defined in.[8]

3.2 IMM under road segment constraint

The IMM is an algorithm for combining estimated states from multiple models to get a better state estimate when the target is maneuvering. The IMM is near optimal with a reasonable complexity. In section 3.1, a constrained motion model i to segment s, noted $\mathcal{M}_k^{s,i}$, was defined. There is a distinction between the definition of a motion model $\mathcal{M}_k^{s,i}$ (*i.e.* motion model type, noise,...) and the event $M_k^{s,i}$ that the target is moving on the road according the motion model i at time k. Here we extend the segment constraint to the different dynamic models (among a set of $r+1$ motion models) that a target can follow. The model indexed by $r = 0$ is the stop model. The transition between the models is modelled as a Markovian process. In general when the target moves from one segment to the next, the set of dynamic models changes. In a conventional IMM estimator,[9,10] the likelihood function of a model i is given, for a track $\mathcal{T}^{k,l}$, associated with the j-th measurement, $j \in \{0, 1, \ldots, m_k\}$ by:

$$\Lambda_k^i = p\{\mathbf{z}_k^j | M_k^{s,i}, Z^{k-1,n}\}, \quad i = 0, 1, \ldots, r \tag{11}$$

where $Z^{k-1,n}$ is the subsequence of measurements associated with the track $\mathcal{T}^{k,l}$.

Using the IMM estimator with a stop motion model, we get the likelihood function of the moving target mode for indexes $i \in \{0, 1, \ldots, r\}$ and for $j \in \{0, 1, \ldots, m_k\}$ by:

$$\Lambda_k^i = \quad P_D \cdot p\{\mathbf{z}_k^j | M_k^{s,i}, Z^{k-1,n}\} \cdot (1 - \delta_{j,0}) + (1 - P_D) \cdot \delta_{j,0} \tag{12}$$

The likelihood of the stopped target mode (*i.e.* $r = 0$) is:

$$\Lambda_k^0 = p\{\mathbf{z}_k^j | M_k^{s,0}, Z^{k-1,n}\} = \delta_{j,0} \tag{13}$$

where $\delta_{j,0}$ is the Kronecker function defined by $\delta_{j,0} = 1$ if $j = 0$ and $\delta_{j,0} = 0$ otherwise.

The combined (global) likelihood function Λ_k of a track including a stopped model is then given by:

$$\Lambda_k = \sum_{i=0}^{r} \Lambda_k^i \cdot \mu_{k|k-1}^i \tag{14}$$

where $\mu_{k|k-1}^i$ is the predicted model probabilities.[9]

The combined state estimate, called global state estimate, is the sum of each constrained local state estimate weighted by the model probability, *i.e.*

$$\hat{\mathbf{x}}_{k|k} = \sum_{i=0}^{r} \mu_k^i \hat{\mathbf{x}}_{k|k}^{i,s} \tag{15}$$

Here, one has presented briefly the principle of the IMM algorithm constrained to only one road segment s. However, a road section is composed with several road segments. When the target is making a transition from one segment to another, the problem is to choose the segments with the corresponding motion models that can better fit the target dynamics. The choice of a segment implies the construction of the directional process noise. That is why the IMM motions model set varies with the road network configuration and variable-structure IMM (VS IMM) offers a better solution for ground target tracking on road networks. Such algorithm has been denoted VS IMMC (C standing for Constrained) and presented in details in[11] .

3.3 Multiple target tracker

We have previously presented an IMM under road constraint with a variable structure (VS IMMC) to estimate the sate of a target at each time. However, we must estimate the state in a multi-target context. That is why we extend our estimation algorithm with an usual multiple target tracker: the SB-MHT (Structured Branching - Multiple Hypotheses Tracking). More details can be found in chapter 16 of[8] . We briefly describe here the main steps of this algorithm.

1. The first functional block of the SB-MHT is shown in figure 2. It consists of the track confirmation and the track maintenance. When the new set Z^k of measurements is received, a standard gating procedure[8] is applied in order to determine the valid measurement reports for track pairings. The existing tracks are updated with VS IMMC at first, and then extrapolated confirmed tracks are formed. When the track is not updated with reports, the stop-motion model is activated.

2. In order to palliate the association problem, we need a probabilistic expression for the evaluation of the track formation hypotheses that includes all aspects of the data association problem. It is convenient to use the log-likelihood ratio (LLR) as a score of a track $\mathcal{T}^{k,l}$ because it can be expressed at current time k in the following recursive form[8] :

$$L_{k,l} = L_{k-1,n} + \Delta L_{k,l} \tag{16}$$

with

$$\Delta L_{k,l} = \log\left(\frac{\Lambda_k}{\lambda_{fa}}\right) \tag{17}$$

and

$$L(0) = \log\left(\frac{\lambda_{fa}}{\lambda_{fa} + \lambda_{nt}}\right) \tag{18}$$

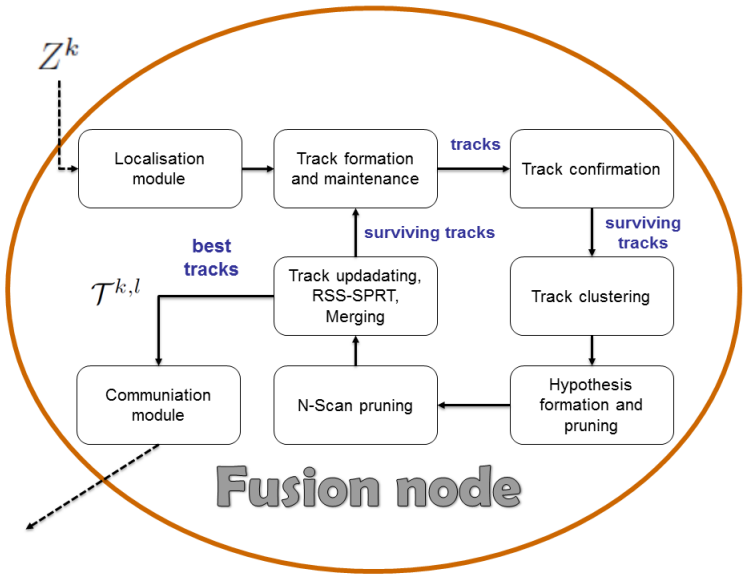

Figure 2: SB-MHT logic flowchart in a fusion node.

where λ_{fa} and λ_{nt} are respectively the false alarm rate and the new target rate per unit of surveillance volume. Λ_k is the global likelihood function described in (14). After the track score calculation of the track $\mathcal{T}^{k,l}$, Wald's Sequential Probability Ratio Test (SPRT) is used to set up the track status either as deleted, tentative or confirmed track. The tracks that fail the SPRT are deleted, and the surviving tracks are kept for the next stage.

3. The process of clustering is used to put altogether the tracks that share common measurements. The clustering limits the number of hypotheses to generate, and therefore it can drastically reduce the complexity of tracking system. The result of the clustering is a list of tracks that are interacting. The next step is to form hypotheses of compatible tracks.

4. For each cluster, multiple compatible hypotheses are formed to represent the different compatible tracks scenarios. Each hypothesis is evaluated according to the track score function associated to the different tracks. Then, a technique is required to find the set of hypotheses that represents the most likely tracks collection. The unlikely hypotheses and associated tracks are deleted by a pruning method, and only the N_{Hypo} best hypotheses are kept in the system.

5. For each track, the *a posteriori* probability is computed, and a classical *N-Scan* pruning approach[8] is used to delete the most unlikely tracks. With this approach the most likely tracks are selected to reduce the number of tracks. However, the *N-Scan* technique combined with the constraint implies that other tracks hypotheses (*i.e.* constrained on other road segments) are arbitrary deleted. To avoid this problem, we modify the *N-Scan* pruning approach in order to select the N_k best tracks on each N_k road sections.

6. The SPRT is used to delete the unlikely hypotheses among the N_k hypotheses. The tracks are then updated and projected on the road network. In order to reduce the number of tracks to keep in the memory of the computer, a merging technique (selection of the most probable tracks which have common measurements) is also implemented.

3.4 Classification fusion

In a fusion node, the target type tracker presented in[12] is used to improve the performance of the data association in the SB-MHT. The principle consists to update the posterior class probability vector at each scan time t_k, with the classifier output. The classifier gives the probability vector $\beta_{k,l}$ of a track $\mathcal{T}^{k,l}$ given by:

$$\beta_{k,l} = \frac{\mathbf{c}_k^j \otimes \beta_{k-1,n}}{\mathbf{c}_k^{j\,\prime} \beta_{k-1,n}} \tag{19}$$

where $\mathbf{c}_k^j$ is the likelihood vector of the jth sensor classifier output, $\beta_{k-1,n}$ is the prior probability provided by the previous updated track $\mathcal{T}^{k-1,n}$ and $\otimes$ is the Schur-Hadamard product. The initial classification vector is given by:

$$\beta_0 = \frac{\mathbf{c}_k^j}{\sum_{n=1}^{n=N} \mathbf{c}_k^j(n)} \tag{20}$$

In assuming the independence of the kinematic and classification observations, the augmented logarithm likelihood ratio $\Delta L_{k,l}^a$ is the sum of the logarithm kinematic-likelihood $\Delta L_{k,l}$ ratio given in (17), and the logarithm of classification ratio $\Delta L_{k,l}^c$. The recursive form of the track score (16) is then given by

$$L_{k,l} = L_{k-1,n} + \Delta L_{k,l}^a \tag{21}$$

with

$$\Delta L_{k,l}^a = \Delta L_{k,l} + \Delta L_{k,l}^c \tag{22}$$

where $\Delta L_{k,l}$ is defined in (17).

The log-likelihood ratio of the classification belonging to the track $\mathcal{T}^{k,l}$ versus belonging to a false or new target is:

$$\Delta L_{k,l}^c = \log\left(\frac{\mathbf{c}_k^{j\,\prime} \beta_{k-1,n}}{\mathbf{c}_k^{j\,\prime} \beta_e}\right) \tag{23}$$

where e defines an extraneous target. If the track is not associated to a measurement at the current time t_k we have $\Delta L_{k,l}^c = 0$.

Finally, the updated target type $\hat{c}_{k,l}$ of the track $\mathcal{T}^{k,l}$ is chosen as the maximum probability of updated classification vector (19).

4. TRACK SEGMENT ASSOCIATION

4.1 Problem formulation

The goal of the TSA (Track Segment Association) is to reduce the number of broken tracks by using a correlation approach between tracks associated to the same target. In this first approach, we do not consider the fusion of track segments based on feature element. We recall that the classification attributes are reduced to few elements, defined in the same discernment frame for all heterogeneous sensor (see part 2.3) and are not discriminant for the correlation function. So, to improve the TSA performances, we study a correlation function based on kinematics information and the track classification information updated with the classification information.

In the multi-target tracking approaches, a data association algorithm is used to associate measurements with predicted tracks in the presence of clutter. The problem is similar here because we try to associate a current track with an old track set in presence of false tracks (the tracker is not perfect and false tracks can appear in dense clutter area). Based on the works of[13,14] on the track segment association, we propose an approach to solve the track segment association taking into account the road network and an IMM-smoother[15,16] .
Two track sets are considered: an old track set $\mathcal{O}_k$ which contains the terminated track ("dead" or "stopped" tracks) at time t_k due to lack of measurement, and a current track set $\mathcal{C}_k$ which contains current updated tracks (the stop model is not activated) at time t_k.

The two sets $\mathcal{O}_k$ and $\mathcal{C}_k$ are updated at each sensor scan time by the following process:

1. Track sets selection. The first step is to build the tracks set $\mathcal{C}_k$ of current tracks and the tracks set $\mathcal{O}_k$.
2. Smoothing. The tracks contained in $\mathcal{O}_k$ and in $\mathcal{C}_k$ are smoothed.
3. Track correlation. On user's request or after a sliding window (in automatic mode), a retrodiction and prediction process are respectively done on tracks contained in $\mathcal{C}_k$ and old tracks contained in $\mathcal{O}_k$. At each scan time track segments are associated based on cost function.
4. Track assignment. The Auction algorithm[8] is used to solve the track segment association problem.

Each step of this TSA algorithm are detailed in the next sections

4.2 TSA Algorithm

4.2.1 Track sets selection

At each time t_k, the terminated and updated tracks are extracted from the tracker. An updated track $\mathcal{T}^{k,l}$ at the current time t_k is defined by the sequence of its estimated states and associated covariance. Its initialization time is noted $t_{k_i^l}$.

$$\forall \mathcal{T}^{k,l} \in \mathcal{C}_k, \quad \mathcal{T}^{k,l} \triangleq \{\hat{\mathbf{x}}^l_{t|t}, \mathbf{P}^l_{t|t}, t = k_i^l, \ldots, k\} \tag{24}$$

Starting from an empty set $\mathcal{C}_k$, the confirmed tracks are added to this one. In the set $\mathcal{O}_k$, there are all terminated tracks since the beginning of the surveillance mission. The deleted tracks $\mathcal{T}^{k_e^m,m}$, are defined with a termination time $t_{k_e^m}$ and a start time $t_{k_i^m}$. An old track is defined by

$$\forall \mathcal{T}^{k_e^m,m} \in \mathcal{O}_k, \quad \mathcal{T}^{k_e^m,m} \triangleq \{\hat{\mathbf{x}}^l_{t|t}, \mathbf{P}^l_{t|t}, t = k_i^m, \ldots, k_e^m\} \tag{25}$$

The times t_{k_e} and t_{k_i} are not necessarily the same for each track contained in $\mathcal{O}_{k-1}$. The current old track set $\mathcal{O}_k$ is based on the previous set $\mathcal{O}_{k-1}$ updated with the current deleted tracks. The tracks deleted by the SB-MHT at the current time t_k are added to set $\mathcal{O}_k$. In addition, to palliate the track discontinuity due to the stop-model activation we add to the set $\mathcal{O}_k$ the stopped tracks. If a stopped track present in $\mathcal{O}_{k-1}$ is moving at time t_k, it is extracted from $\mathcal{O}_k$ and added to $\mathcal{C}_k$. In our simulation, a track is declared as "stopped" if the confirmed track has a stop-model probability greater than 0.9. The cardinality of the sets $\mathcal{C}_k$ and $\mathcal{O}_k$ are not necessarily the same.

4.2.2 IMM fixed-lag smoother

The IMM estimator presented in this paper has been proven to be effective for tracking maneuvering ground targets. This is more significant if the contextual information is taking into account in the tracking process. For intelligent system, the real-time application for tactical situation establishment is primordial. The second main point is to understand the situation assessment in constrained time. So we can use this time period to achieve the best estimates of the target states at a given time based on all measurements up to the current time. In addition, the achievement process contributes to improve the initial estimated state and consequently the track retrodiction precision.

For this, we must take into account all the measurements of a track available in a sliding window. According to the IMM estimator, we use a IMM smoothing algorithms presented in[15,16] which involves forward filtering followed by backward smoothing. The forward recursion is performed using the VS IMMC algorithm. The backward recursion keeps the selected model set of the track and imitates the IMM estimator in the forward direction. In this subsection we describe the IMM smoothing method presented in.[15,16] In addition we use the smoothing step to constraint the past of the current tracks to belong to the road network when possible. After the smoothing process the constrained tracks are always on the road network, but some unconstrained tracks can also belong to the road network. For each unconstrained smoothed state we use the statistical test presented in[11] to project the smoothed states and their covariances on the most probable road segment.

4.2.3 Track correlation

Retrodiction and association The retrodiction is the backward prediction of each track contained in $\mathcal{C}_k$. From each starting time $t_{k_i^l}$ of each track $\mathcal{T}^{k,l}$, we use a back propagation equation of a constant velocity motion model given in (9). For each track $\mathcal{T}^{k,l}$, a sequence of retrodicted states is obtained at previous times for each deleted tracks of $\mathcal{O}_k$. The set of candidate track pairs for TSA is obtained according a two-step validation procedure:

1. The first step of tracks pairing consists of a velocity gating. As illustrated in the figure 3, we associate the current track with old tracks if the maximum ground target speeds are below v_{max}. The set of pairing tracks satisfying this condition is defined by

$$\begin{aligned}\Phi_v = \{(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \text{ such that } \frac{|\hat{x}^l_{k_c|k_i^l} - \hat{x}^m_{k_c|k_o}|}{t_{k_i} - t_{k_o}} \leq v_{max}, \frac{|\hat{y}^l_{k_c|k_i^l} - \hat{y}^m_{k_c|k_c}|}{t_{k_i} - t_{k_o}} \leq v_{max}, \\ \mathcal{T}^{k,l} \in \mathcal{C}_k, \mathcal{T}^{k_o,m} \in \mathcal{O}_k, t_{k_i^m} < t_{k_o} < t_{k_e^m}, t_{k_c} = \frac{t_{k_i^l} - t_{k_o}}{2}\}\end{aligned} \tag{26}$$

   where $(\hat{x}^l_{.|k_i^l}, \hat{y}^l_{.|k_i^l})$ is the retrodicted location of the track $\mathcal{T}^{k,l}$ and $(\hat{x}^m_{k_c|k_o}, \hat{y}^m_{k_c|k_o})$ is the predicted location of the track $\mathcal{T}^{k_o,m}$ as described in 3.1.

   This is the approach used in[14] where the track pairing of the old track $\mathcal{T}^{k_e^m,m}$ is done at each time $\{t_{k_i^m}, \ldots, t_{k_e^m}\}$ and not only at the time end $t_{k_e^m}$.

2. The second step is done to limit the number of track pairings. Under the statistical noise independence between the current tracks of $\mathcal{C}_k$ and the old tracks $\mathcal{O}_k$, we use a classical χ^2_n test (n, the state vector dimension, is the degree of freedom) to validate the pairs in Φ_v. At time t_{k_c} the difference between the retrodicted tracks of $\mathcal{T}^{k_i^l,l}, (\mathcal{T}^{k,l} \in \mathcal{C}_k)$ and predicted tracks of $\mathcal{T}^{k_o,m}, (\mathcal{T}^{k_e^m,m} \in \mathcal{O}_k)$ is defined by:

$$\Delta^{l,m}_{k_c} = \hat{\mathbf{x}}^l_{k_c|k_i^l} - \hat{\mathbf{x}}^m_{k_c|k_o} \tag{27}$$

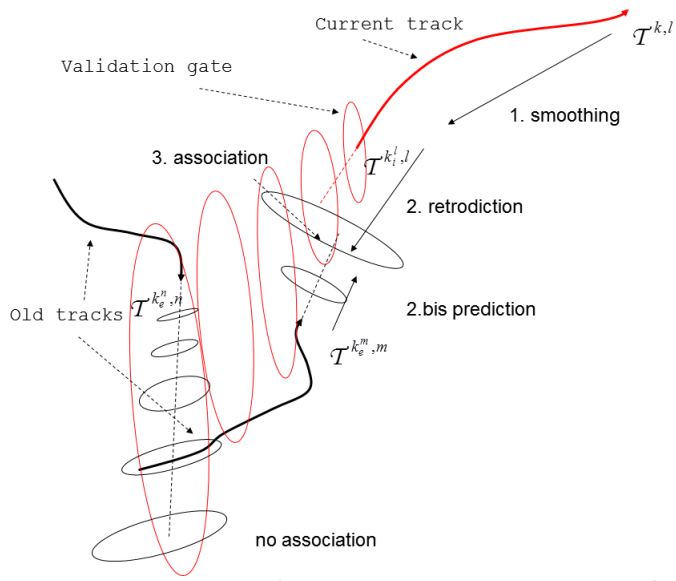

Figure 3: Track segment association principle.

with the covariance:

$$\mathbf{P}_{k_c}^{l,m} = \mathbf{P}_{k_c|k_i^l}^{l} - \mathbf{P}_{k_c|k_o}^{m} \tag{28}$$

The new set of track pairing candidate is defined by the following statistical test:

$$\begin{aligned}\Phi_s = \{(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \mid (\Delta_{k_c}^{l,m})^T [\mathbf{P}_{k_c}^{l,m}]^{-1} (\Delta_{k_c}^{l,m}) \leq \chi_n^2(1-Q),\\ (\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \in \Phi_v, t_{k_i^m} < t_{k_o} < t_{k_e^m}, t_{k_c} = \frac{t_{k_i^l} - t_{k_o}}{2}\}\end{aligned} \tag{29}$$

where Q is a fixed tail probability. The counterpart of this approach is in the set Φ_s, where several pairs can correspond to the same tracks pairing between $\mathcal{T}^{\cdot,l}, (\mathcal{T}^{k,l} \in \mathcal{C}_k)$ and $\mathcal{T}^{\cdot,m}, (\mathcal{T}^{k_e^m,m} \in \mathcal{O}_k)$. In order to limit the complexity in the assignment algorithm we keep the pairs which have the minimal statistical distance among all pairs of the same tracks.

Track assignment After applying the gating (29), we obtain a set of track pairs candidates between current track in $\mathcal{C}_k$ and deleted or stopped tracks in $\mathcal{O}_k$. The association is formulated as a 2-D assignment problem. For this, we define a binary assignment variable as

$$a(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \triangleq \begin{cases} 1 & \text{track } \mathcal{T}^{k_i^l,l} \text{ is originated from the track } \mathcal{T}^{k_o,m} \text{ at time } t_{k_o} \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

The cost of the track association between $\mathcal{T}^{k_i^l,l}$ and $\mathcal{T}^{k_o,m}$ is denoted by $c(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m})$, and is defined by

$$c(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \triangleq \begin{cases} -\log \frac{\mathcal{N}(\Delta_{k_c}^{l,m}; 0, \mathbf{P}_{k_c}^{l,m})}{\mu} & \text{if } (\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \in \Phi_s \\ -\log(1 - P_{D_s}) & \text{otherwise.} \end{cases} \tag{31}$$

where μ is given by the spatial density of the extraneous tracks in the state space and P_{D_s} is the probability that a target is tracked.[14]

The optimal set of track pairs (optimal assignment) is obtained by minimizing the following global cost C:

$$C = \sum_{l=1}^{M_c} \sum_{m=1}^{N_o} a(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) c(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \tag{32}$$

under the constraints:

$$\sum_{l=1}^{M_c} a(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) = 1, m = 1, \ldots, N_o \tag{33}$$

$$\sum_{m=1}^{N_o} a(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) = 1, l = 1, \ldots, M_c \tag{34}$$

where M_c and N_o are respectively the number of current associated tracks and the number of dead associated tracks. This 2-D assignment problem is solved using the Auction algorithm.[8]

Track segment association algorithm execution In our application, the track segment association algorithm is applied every minute. The track retrodiction is done for each current track after the smoothing process. We use the validation tests to determine the cost between current track segment and dead track segments. A new current track segment is obtained after the TSA algorithm. A dead track at the current time t_k is extracted from the set $\mathcal{O}_k$ if it is not used for the track segment association during two minutes.

4.3 Introduction of the target type information in the TSA algorithm

We propose also to introduce the track classification scoring information in the TSA process in order to increase the discrimination between the old and current tracks. The method consists to modify the cost association presented in (31) by introducing a track classification cost. After the pairing test (29) between the current track $\mathcal{T}^{k,l}$ and the old track $\mathcal{T}^{k_e^m,m}$, the track class vectors $\beta_{k_i^l,l}$ and $\beta_{k_e^m,m}$ are compared based on the Bhattacharyya distance:

$$c(\beta_{k,l}, \beta_{k_e^m,m}) = -\log \sum \sqrt{\beta_{k,l} \cdot \beta_{k_e^m,m}} \tag{35}$$

The global cost association becomes:

$$c(\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) = \begin{cases} -\log \frac{\mathcal{N}(\Delta_{k_c}^{l,m};0,\mathbf{P}_{k_c}^{l,m})}{\mu} + c(\beta_{k,l}, \beta_{k_e^m,m}) & \text{if } (\mathcal{T}^{k_i^l,l}, \mathcal{T}^{k_o,m}) \in \Phi_s \\ -\log(1 - P_{D_s}) & \text{otherwise.} \end{cases} \tag{36}$$

5. RESULTS BASED ON REAL DATA

The feasibility of the developed algorithms deployed in fusion node has been tested on simulated results. To validate algorithms in our WSN system we have proceed to operational exercise in October 2014. In this section we present the live recordings that were carried out in the Valdahon's military camp in the east of France. The playing scenario consists to deploy sensor network for intelligence surveillance application around a FOB (Forward Operating Base).

5.1 Scenario description

The Table 1 describes the types of sensors used during the exercise and their location is given on the figure 4. The ground sensor network is composed of heterogeneous principal sensors only that are always activated to test the fusion node (located in the FOB). Radar sensor, sensor group 1 and group 2 are associated with a transmission node connected together to the fusion node. So, the tested architecture is a centralized architecture only. The radar sensor detection coverage represented in the figure 4 in red is limited to the field area and is stopped at the limit of the forest.

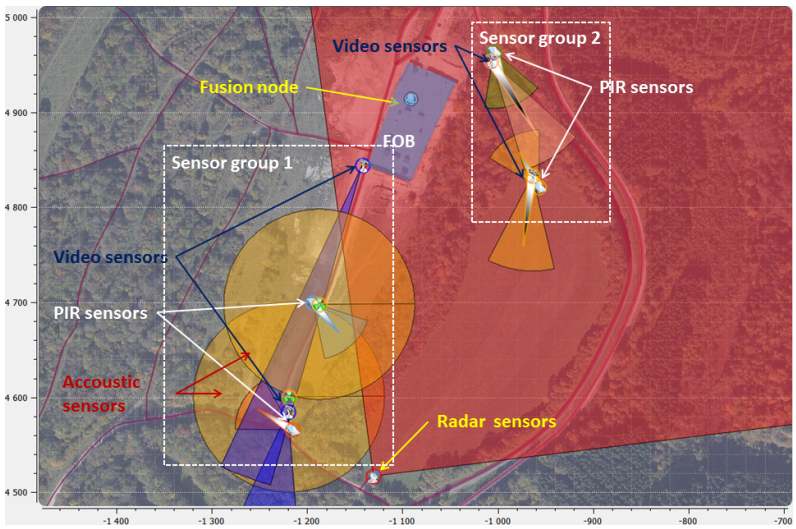

Figure 4: Sensor and fusion node deployement.

Two pedestrian targets are leaving the FOB and maneuvering around it on and out the road network and according the trajectories given in figure 5. The yellow target has been asked to make strong maneuver in order to test our the ability of our IMM algorithm.

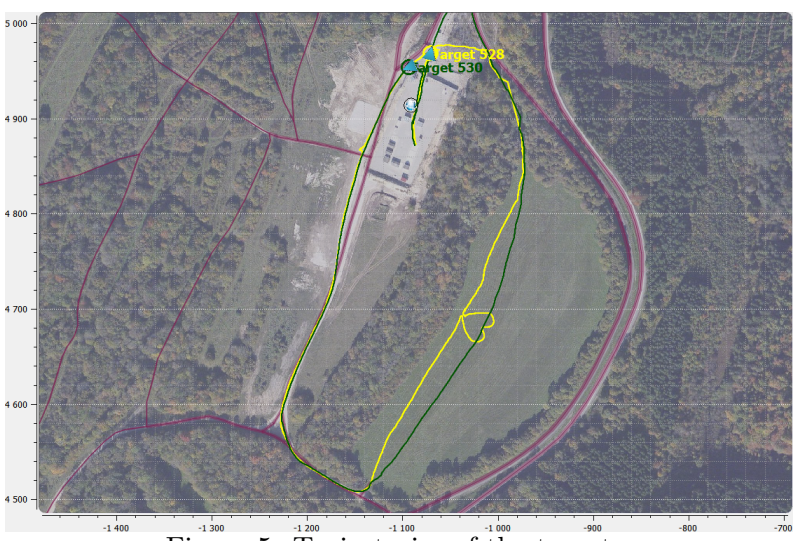

Figure 5: Trajectories of the targets.

5.2 Results

In this section we present the tracks displayed in the C2 issued from the presented MTT algorithm.

The figure 6 shows the situation assessment at 1:20:18 pm. We observe 3 tracks :

- track -80871421, associated to the yellow target. This track has been initialized according composite report built with camera and PIR sensors from group 2. After its initialization, it is updated with an IMM Unscented Kalman Filter (UKF) with bearing only data from both sensors. When the track has left the forest, it has been updated by the radar in the field.
- track -80871418, associated to the green target. This track has been initialized according also according composite report built with camera and PIR sensors from group 1. It has been updated with a IMM UKF with bearing only data from video, PIR and acoustic sensors. Thanks to the road network location, the MTT algorithm was able to track the target on the trail. When the track has left the group 1, it has been deleted because it was not updated during 30 seconds.
- track -80871413, was a track associated to a real target but not recorded in this exercise.

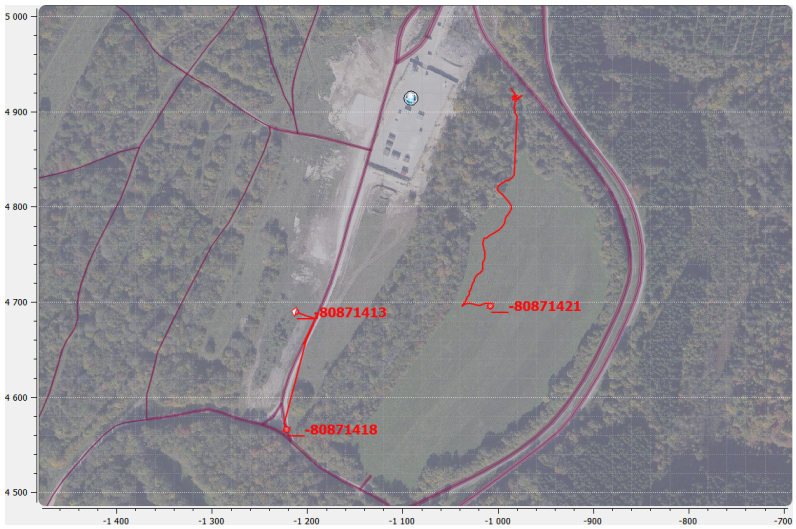

Figure 6: Tracks at time 1:20:18 pm

The figure 7 shows the situation assessment at 1:22:13 pm. Because tracks -80871418, -80871413 were not updated during 30 seconds, the MTT algorithm has deleted these tracks. Then when the green target had moved through the radar sensor field of view, a new track -80871406 has been initialized and updated. The IMM algorithm has detected the maneuver of the yellow target and maintained the track continuity. Without the IMM, the track would have been lost during the maneuver.

The figure 8 shows the situation assessment at 1:22:21 pm. The TSA algorithm had selected old tracks and associated the old tracks with current tracks. The TSA had declared that the track -80871406 was the old track -80871418 and so improved the track continuity. The association is represented by dot line (this is not the extrapolated trajectory but really the relational link between new and old track as required by the end-user). The MTT algorithm has generated several hypotheses association and kept the tracks.

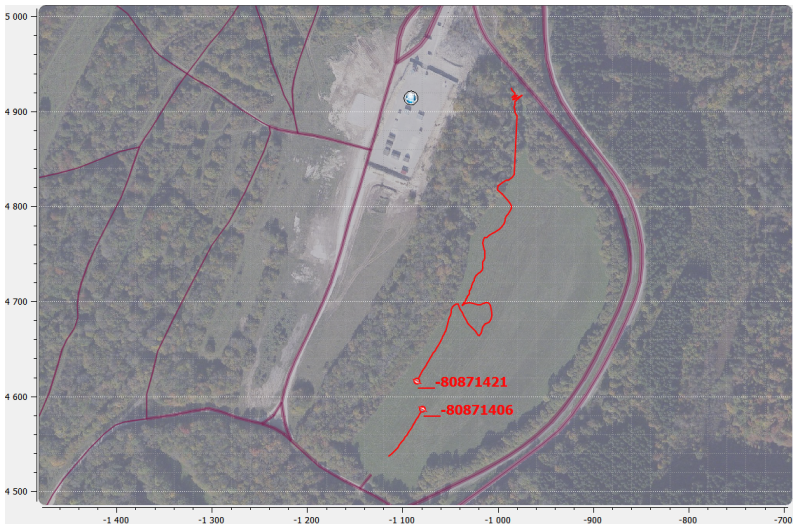

Figure 7: Tracks at time 1:22:13 pm.

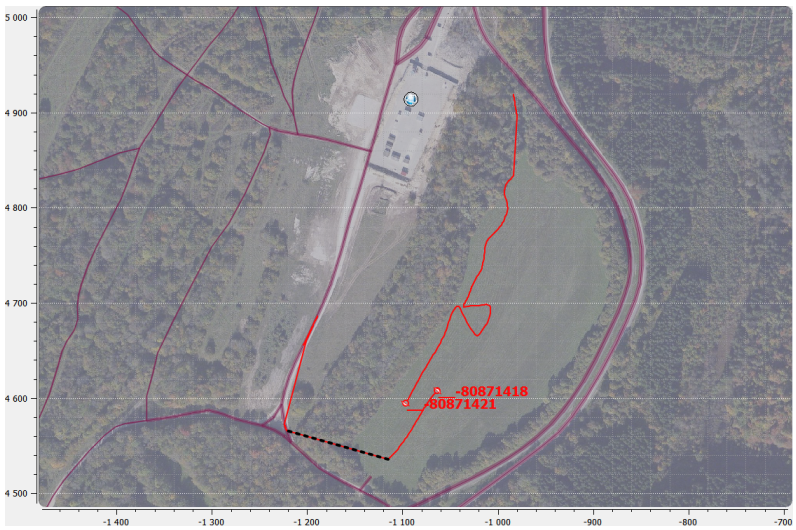

Figure 8: Tracks at time 1:22:21 pm.

The figure 9 shows the situation assessment at 1:26:34 pm. The track -80871421 has been deleted because it had left the radar sensor's field of view and evolved in lacunar area and consequently it has not been updated during 30 seconds. So when the yellow target arrived in the group 2 sensor area, a new track -80871405 was created. At time 1:26:41 pm (figure 10), the TSA has been activated and the new track -80871405 has been correlated correctly to the old track -80871421. The figure 10 shows the situation assessment at 1:26:45 pm. The yellow target had left the group 1 sensor area and entered in the FOB. So, the track -80871421 has been deleted because no sensor was engaged to provide data near the FOB. The green target associated to the track -80871416 evolved through the forest and has been updated by the sensors of the group 2.

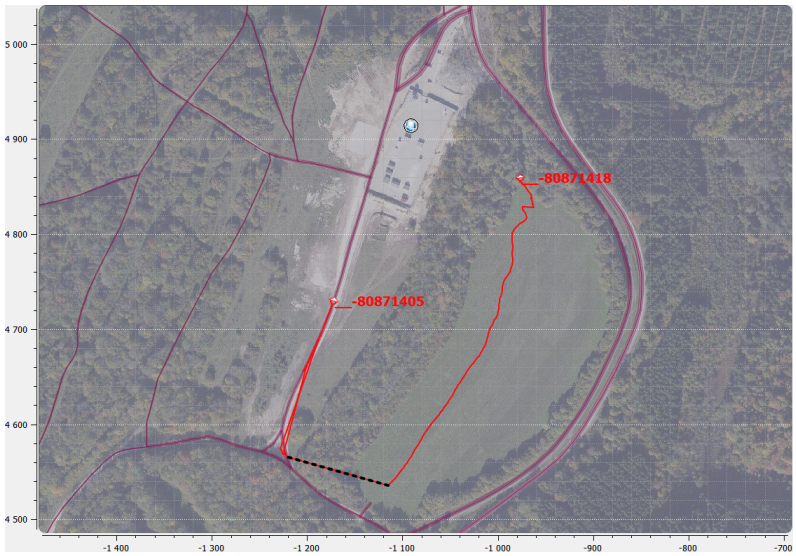

Figure 9: Tracks at time 1:26:34 pm.

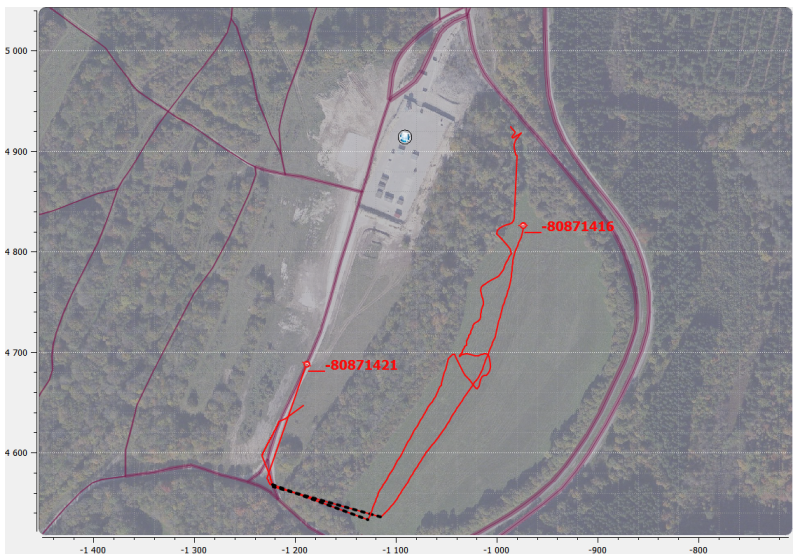

Figure 10: Tracks at time 1:26:41 pm.

6. CONCLUSION

In order to validate the MTT architecture in centralized environment, we have proceeded to operational exercise with french MOD. The MTT proposed algorithm should track and classify targets around a FOB with several heterogeneous sensors in a centralized architecture. Our tracking system takes into account the road network configuration, and thanks to IMM it is able to improve the track estimation and continuity when the targets are detected. To improve ISR capabilities, the TSA approach offers some intelligence for situation assessment by correlating deleted tracks with current tracks.
The main next steps of the project are:

- to confirm these results with other exercises;

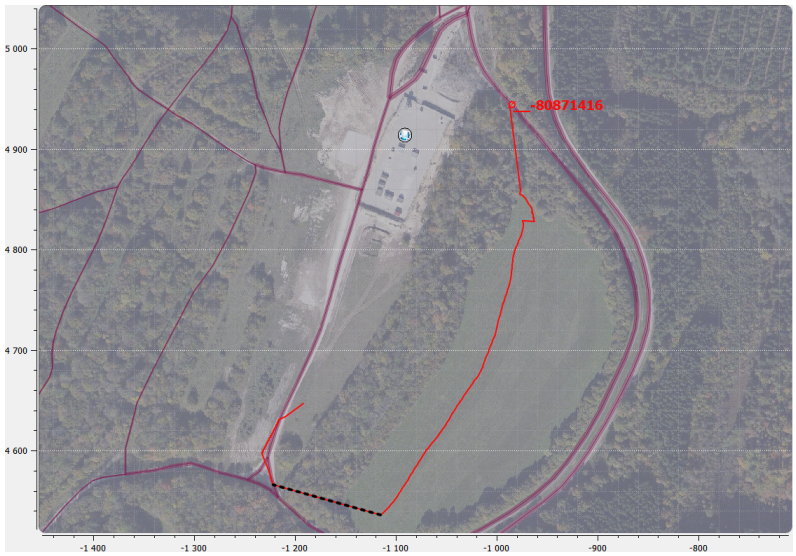

Figure 11: Tracks at time 1:26:45 pm.

- to evaluate the performance of the TSA with classification information;
- to introduce smart sensor information;
- to test hierarchical data fusion algorithms and compare the results with centralized architecture.
- to study new approach for abnormal behavior detection.

7. ACKNOWLEDGMENTS

The authors thank the SAGEM company for the material (sensors and fusion node), operational support and project lead.

REFERENCES

1. M. Ekman and H. Palsson, "Ground target tracking of vehicles in a wireless sensor network," *Proc. of FUSION 2012, Conf., Singapore*, 2012.
2. M. Ekman and E. Sviestins, "Mutiple model algorithm based on particle filters for ground target tracking," *in Proc. of FUSION 2007, Conf., Quebec, July 2007.*, 2007.
3. D. Salmond, M. Clark, R. Vinter, and S. Godsill, "Ground target modelling, tracking and prediction with road networks," *in Proc. of FUSION 2007, Conf., Quebec, July 2007.*, 2007.
4. P. Parmar and M. Zaveri, "Mutiple target tracking and data association in a wireless sensor network," *in Proc. of fourth international conference on computational intelligence and communications network 2012*, 2012.
5. S. Oh, S. Sastry, and L. Shenato, "A hierarchical multiple-target tracking algorithm for sensor networks," *in Proc. of the 2005 IEEE international conference on robotics and automation, Conf., Barcelona, Spain*, 2005.
6. M. Ulmke and W. Koch, "Road-map assisted ground moving target tracking," *IEEE Transactions on Aerospace and Electronic Systems* **42**, pp. 1264–1274, 2006.
7. B. Pannetier and J. Dezert, "Track segment association with classification information," *Sensor Data Fusion: Trends Solutions Applications (SDF),Workshop on*, 2012.

8. S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*, Artech House, 1999.
9. Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley and Sons, New York, 2001.
10. T. Kirubarajan and Y. Bar-Shalom, "Tracking evasive move-stop-move targets with an mti radar using a vs-imm estimator," *in Proc. of IEEE Transactions aerospace and electronics systems, July 2003* **39**, pp. 1098–1103, 2003.
11. B. Pannetier, V. Nimier, and M. Rombaut, "Multiple ground target tracking with a gmti sensor," *Proceedings of MFI*, 2006.
12. Y. Bar-Shalom, T. Kirubarajan, and C. Gokberk, "Tracking with classification-aided multiframe data association," *IEEE Transactions on Aerospace and Electronic systems* **41**, pp. 868– 878, 2005.
13. S.-W. Yeom, T.-Kirubarajan, and Y. Bar-Shalom, "Improving track continuity using track segment association," *in Proc. of IEEE Aerospace Conference, March 2003*, 2003.
14. S. Zhang and Y. Bar-Shalom, "Track segment association for gmti tracks of evasive move-stop-mve maneuvering targets," *IEEE transactions on Aerospace and Electronic Systems* **47**, pp. 1899 – 1914, 2011.
15. M. Sundkvist and N. Bergman, "Fixed lag smoothing for target tracking applications based on imm," *in Proc. of IEE worshop target tracking: algorithms and applications, October 2001* **2**, 2001.
16. N. Nadarajah, R. Tharmarasa, M. McDonald, and T. Kirubarajan, "Interacting multiple model forward filtering and backward smoothing for maneuvering target," *in Proc. of IEEE Transactions aerospace and electronics systems, October 2012* **2**, 2012.