Data Fusion for Target Tracking and Classification with Wireless Sensor Network

Benjamin Pannetier[a], Robin Doumerc[a], Julien Moras[a], Jean Dezert[a] and Loic Canevet[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 information obtained from a unattended wireless sensor network. A multiple target tracking (MTT) algorithm, taking into account road and vegetation information, is proposed based on a centralized architecture. One of the key issue is how to adapt classical MTT approach to satisfy embedded processing. Based on track statistics, the classification algorithm uses estimated location, velocity and acceleration to help to classify targets. The algorithms enables tracking human and vehicles driving both on and off road. We integrate road or trail width and vegetation cover, as constraints in target motion models to improve performance of tracking under constraint with classification fusion. Our algorithm also presents different dynamic models, to palliate the maneuvers of targets. The tracking and classification algorithms are integrated into an operational platform (the fusion node). In order to handle realistic ground target tracking scenarios, we use an autonomous smart computer deposited 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 intelligence operation ("hunter hunt" scenario).

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

1. INTRODUCTION

This paper presents MTT and track classification algorithms to track in real-time the targets with a wireless sensor network (WSN). This work follows previous research works[18,19] on WSN and we enhance our tracking system with an approach to classify the tracks by taking into account target state estimation and track behavior towards geographical context. We present results obtained in an operational exercise which consists of a large number of smart heterogeneous sensors with on-board sensing, processing and wireless communication capabilities. The WSN technical requirements are described in.[18]

Our demonstrator allows to study automatic data processing to fuse detection 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 papers have been published as explained in.[18] The most popular comments on our work are on the aglorithm choice for target tracking. In fact, as proposed by Ekman and Pelson in[5] a modified particle filter (PF)[6] is used to track a single vehicle through the WSN. 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

Further author information: (Send correspondence to R.D.)
B.P.: E-mail: benjamin.pannetier@onera.fr,
R.D.: E-mail: robin.doumerc@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

to track several targets in the sensor network with severe processing constraints due to hardware solution used in our demonstrator, and to preserve power, computational preocessing unit (CPU) in a fusion node. In fact, because PF approach uses more computing power 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. To improve the MTT algorithm performance, we introduce in this work the geographic information in the tracking process as proposed by Ulmke and Koch[24] . Since we are interested in 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[17] for our WSN tracking demonstrator.

In order to improve the tracking, we consider a classifier that works alongside the tracker. Some research has been done in this field such as,[10],[12].[11] However, most of the target identification system either use data not available in our case, such as the radar cross section, doppler signals, altitude of targets ; or have a enough training data to use a Hidden-Markov Model classifier, or machine learning algorithm. Therefore, we had to design a classification system using the data available from the tracker, such as estimated kinematics parameters and the geographical position of the target. Different works has been done on target classification using Kinematics,[13],[1],[14] and more specifically research work has been conducted in the past years using using fuzzy logic classifier, such as,[157] or.[10] However, most of the aforementionned papers are interested in the identification of different aerial targets, and therefore does not take into account the geographical data. In his paper Jochumsen et al.,[8] use geographical data with the concept of trafficability: what are the capacity for a vehicle to evolve on a specific terrain, in an littoral area. For our WSN tracking demonstrator, we have developed a fuzzy inference based target classifier that uses kinematics data alongside trafficability to identify ground and aerial target at once.

The aim is to adapt and evaluate a conventional multiple target tracking algorithm in order to maintain high track continuity and classification performance in order to provide a reliable situation assessment. For this aim, 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 algorithms themselves.

The paper is organized as follows: in section 2 the WSN is briefly presented. Section 3 describes algorithm for multiple target tracking. Section 4 presents the algorithm to classify the tracks. 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.

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 are involved in our system: 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

In our application, the sensors have know location and have no mobility.

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. The different types of sensor used during exercises are listed Table 1 in.[18]

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 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

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

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

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.

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.[3] moving on a road network, we have proposed in[?] to adapt the level of the dynamic model's noise based on the length of the road segment s. The idea is to increase the standard deviation σ_n defined in (10) to take into account the error on the road segment location. After the state estimation obtained by a Kalman filter, the estimated state is then projected according to the road constraint e_k^s.

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,[9,25] 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.[25]

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[2] .

3.3 Multiple target tracker

We have previously presented an IMM under road constraint with a variable structure (VS IMMC) to estimate the state 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 Tracker). More details can be found in chapter 16 of[3] . We briefly describe here the main steps of this algorithm.

1. The first functional block of the SB-MHT consists of the track confirmation and the track maintenance. When the new set Z^k of measurements is received, a standard gating procedure[3] 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[3] :

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

with

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

and

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

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 the 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[3] 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.

3.3.1 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[16,23] 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.[16,23] 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[2] to project the smoothed states and their covariances on the most probable road segment.

4. TRACK CLASSIFICATION

The system we designed is used to give a classification information of the confirmed tracks of the tracker. To do so, we used two information from the tracker: the kinematic data of the target, and its estimated position(i.e its

associated geographical layer). With these informations, we defined an expert-knowledge fuzzy inference system, to derive the possibility for a target to belong to the hypotheses defined in 6. From those possibilites, we compute probabilities within the transferable belief model. The following subsections will succintly describe the different mathematical concepts used in the classifier, before going deeper into its architecture.

4.1 Basics of Fuzzy set theory

Fuzzy Logic was initially proposed by L. Zadeh to solve the problems of uncertain data in automatic control applications. Later, this method was extended to differents other fields such as pattern recognition problems.

Fuzzy Logic presents several advantages :

- It is light in computations, based on min and max operators
- It is well adapted to the use of uncertain data and to the description of expert knowledge
- The same method can be us to merge parameters of different nature such as continuous and discrete data.
- Measurement errors are explicitly taken into account

Two basic functions are the core of fuzzy logic for pattern recognition:

- A Membership Function, denoted $m_{\tilde{A}}(x)$, that describe how a given class can be represented on the basis of a given parameter : this represent the available knowledge about this parameter.
- A Density of possibility, that describes how the actual value of a given parameter can be distributed given an estimation of its accuracy.

By combining the two function , we can compute an output membership function,mf($\tilde{\cdot}$), using a set of If-then rules a Possibility that describes the level of fitness between the inputs and the reference data for each class.

The likelihood of a hypothesis given data, is defined as a quantity porportionnal to the probability of observing the data given the hypothesis. As described in[20] , the membership function $\text{mf}_{\tilde{A}}(x)$, as a function of x, reflects the extent to which $x \in \tilde{A}$, i.e. $\text{mf}_{\tilde{A}}(x)$ is an indicator of how likely it is that $x \in A$, hence $\text{mf}_{\tilde{A}}(x)$ may be viewed as the likelihood of x for a specified fuzzy set $\tilde{A}$.

4.2 Basics of belief function Theory and Tranferable Belief Model

The belief function habe been introduced by Shafer in his Mathematical Theory of Evidence, also known as Dempster-Shafer Theory (DST). In DST, we work with a discrete frame of discernement as $\Omega = \{\omega_i, i = 1...c\}$ consisting of c exclusive and exhaustive hypothesis (in our case, classes). A mass function can be defined over the power-set of Ω denoted by 2^Ω, which is the set of all subset of Ω. For example, if the frame of discernment $\Omega = \{\omega_1, \omega_2, \omega_3\}$, then its power-set is $2^\Omega = \{\emptyset, \omega_1, \omega_2, \omega_1 \cup \omega_2, \omega_3, \omega_1 \cup \omega_3, \omega_2 \cup \omega_3, \Omega\}$. A mass function is mathematically defined as a mapping $m(\cdot)$ from $2^\Omega \to [0; 1]$, and is named basic belief assignement when it satisfies $m(\emptyset) = 0$ and

$$\sum_{A \in 2^\Omega} m(A) = 1 \tag{19}$$

With a mass function $m(\cdot)$, one can allow one object to belong to different elements (singletons, as well as their disjunctions) in 2^Ω with differient masses of belief. All the elements $A \in 2^\Omega$ such that $m(A) > 0$ are called the focal elements of the mass function $m(\cdot)$. $m(A)$represents the support degree ot the object associated with class ω_i. In pattern classifciation problem, if A is a set of classes (e.g. $A = \omega_i \cup \omega_j$),m(A) can be used to characterize the imprecision(partial ignorance) degree among the class ω_i and ω_j in classification of the object. $m(\Omega)$ denotes the toal ignorance degree, and it usually plas a particular neutral role in the fusion process because $m(\Omega) = 1$ characterizes the vacuous belief source of evidence.

The lower and upper bounds of the probability associated with a mass function respectively correspond to the belief function $Bel(\cdot)$ and the plausibility function $Pl(\cdot)$ defined $\forall A \subseteq \Omega$

$$Bel(A) = \sum_{B \in 2^{\Omega} |\ B \subseteq A} m(B) \tag{20}$$

$$Pl(A) = \sum_{B \in 2^{\Omega} |\ A \cap B \neq \emptyset} m(B) \tag{21}$$

The Dempster rule of combination (D-S Rule), can be use to combine multiple mass functions, for example in the case of a multi-classifier system, where classifier can be considered as an evidence represented by a mass function.

Considering two distinct source of evidence, whose mass function are defined m_1 and m_2. The D-S combination of m_1 and m_2 is denoted $m_1 \oplus m_2$, is defined by $m(\emptyset) = 0$ and $\forall A, B, C \neq \emptyset \in 2^{\Omega}$ as:

$$m_{1 \oplus 2}(A) = (m_1 \oplus m_2)(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) \cdot m_2(C) \tag{22}$$

Where $K = \sum_{B \cap C = \emptyset} m_1(B) \cdot m_2(C)$, $K \neq 1$ is the total conjunctive conflicting mass.

The decision process, i.e the classification of the track, is done within the Transferable Belief Model (TBM), defined by Smets.[22] The TBM is composed by two level: a credal level, used to represent and entertain beliefs quantified by belief functions, and a pignistic level where those beliefs are used to make decision.

The pignistic transform defined by Smets in[21] use the repartition in an equiprobable way of the mass of an proposition B on the hypothesis belonging to B. To put it in a formal way, $\mathbf{BetP}\{m\}(\omega_k)$ is an application from $\Omega \rightarrow [0, 1]$, where $\mathbf{BetP}\{m\}(\omega_k)$ is defined by:

$$\mathbf{BetP}\{m\}(\omega_k) = \frac{1}{(1 - m(\emptyset))} \sum_{A \subseteq \Omega, \omega_k \in A} \frac{m(A)}{|A|} \tag{23}$$

Once the pignistic transformation has been computed for each element ω, the decision is made by picking the element ω_k with the highest pignistic probability:

$$\omega_0 = \underset{\omega_k \in \Omega}{argmax}\, \mathbf{BetP}\{m\}(\omega_k) \tag{24}$$

4.3 Fuzzy-Inferenced based classifier

The classifier we designed can be divided in three steps:

A fuzzy inference system, that takes as input the kinematics property and position of the track, and give as an output a vector of possibility;

A function that map the possibility vector from the inference system into basic belief assignement;

A fusion fuction, that combine the new mass function for the tracks with previous ones, and derives the associated pignistic probabilities.

4.3.1 Fuzzy Inference system

As described in 4.1, the first step is to define the membership functions regarding the terrain and the velocity for each hypothesis described in 6. Because we do not have enough training data, we build those membership function using expert knowledge.

We have defined a set of terrain:

$$\mathbf{T} \triangleq \{\textit{Light-Forest}, \textit{Dense-Forest}, \textit{Road Network}, \textit{Water Area}, \textit{Unknown}\} \tag{25}$$

Then, for each terrain of T,construct a *"trafficability"* capacity for each class defined in 6. This traficability is build according to the minimum (and maximum) speed capacity for each class on each terrain.

This is then translated on a multi fuzzy logic inference system, where each inference system is defined by a terrain.

When the tracker updates each confirmed track, it computes it estimated velocity and position. Based on the GIS data described in section 2.2, we can estimate the terrain in which the track is located. Based on that terrain estimation, we can use the fuzzy inference system to obtain the possibility for the track to belong for each class defined. Because we are not in an open-world problem, we added an additional class C_u=$\{\textit{Unknown}\}$, as an output of the inference system, that is activated when the inputs does not fit any of the rules.

At this step of the algorithm, the inference system provide us for a given track, a likelihood vector to belong to any of the hypotheses. However, we do know that some targets may have similar kinematics capability, a same track can have many different possibilities, and therefore the fuzzy inference system cannot make the classification itself. In order to make a decision, we need to map this possibility into something which we can based a decision on. This will be done using belief functions defined in the next subsection.

4.3.2 From Likelihood to Belief functions

Based on what we described the in 4.1, the output of the fuzzy inference system is a set $L_{K,T} = \{L\omega_1, L\omega_2, ..., L\omega_c\}$ where $L\omega_c$is the likelihood that the observed track, with kinematic parameter K and Terrain parameter T, belongs to the class ω_c. In order to fuse the output of this inference system, we assign mass function based on those likelihoods. To do so, we used the work describe at[4] to convert the likelihood into plausiblity.

Given $L(.;x): \quad \omega \rightarrow f(x;\omega)$ is the likelihood function, we define a "likelihood based" plausibility function:

$$Pl_\Omega(A;x) = \sup_{\omega \in A} pl(\omega;x) = \frac{sup_{\omega \in A}\, L(\omega;x)}{sup_{\omega \in 2^\Omega}\, L(\omega;x)} \;, \forall A \subseteq 2^\Omega \tag{26}$$

From this plausibility function, we obtain then the plausibility for all subsets of 2^Ω.

And, we can assign the mass function of A using:

$$m_\Omega(A) = \sum_{B \subseteq A} (-1)^{|A|-|B|+1} Pl(\bar{B}) \quad , \forall A \subseteq \Omega \tag{27}$$

Hence, we have at this point for each track, considering its velocity and position, the support degree that it belongs to any of the classes.

4.3.3 Fusion and Decision Making

At each update of a track, i.e at time k, a new estimate of its velocity and position is derived by the tracker. From this new set of input, is assigned a new mass function, denoted $m_{c,k-1}$, representing the evidence for the track to belong to each hypotheses.

The global mass function at time k,denoted $m_{\mathscr{C},k}$, on which the classification decision will be made, can be derived by fusing the new mass function $m_{c,k-1}$ with the previous global mass function $m_{\mathscr{C},k-1}$:

$$m_{\mathscr{C},k} = m_{\mathscr{C},k-1} \oplus m_{c,k}$$

The mass functions fusion defined in 22 is important in our classification system and more specifically on the decision making. The prior used in our expert system to defined the fuzzy membership function are broad, and for most tracks it is not possible to make a proper estimation of the class of the object. However, we make the assumption that over time, the target will have a discriminant behavior (incompatible kinematic or position) with one or many remaining hypotheses, leading to discard these hypotheses. Yet, because the terrain used in order to assign the mass function is based on an estimate position it can be unprecise and not accurate, and force an incorrect discard of an hypothesis. To counterbalance this effect, we use an additionnal contextual discounting in our mass function from the kinematic classifier.

One of the advantages of our approach is its modularity: the kinematic classifier results can be fused with the other sensors' classification results $\mathbf{c}_k^j$ described in 2.3, in order to improve the global classification of our system. This is represented in figure 1.

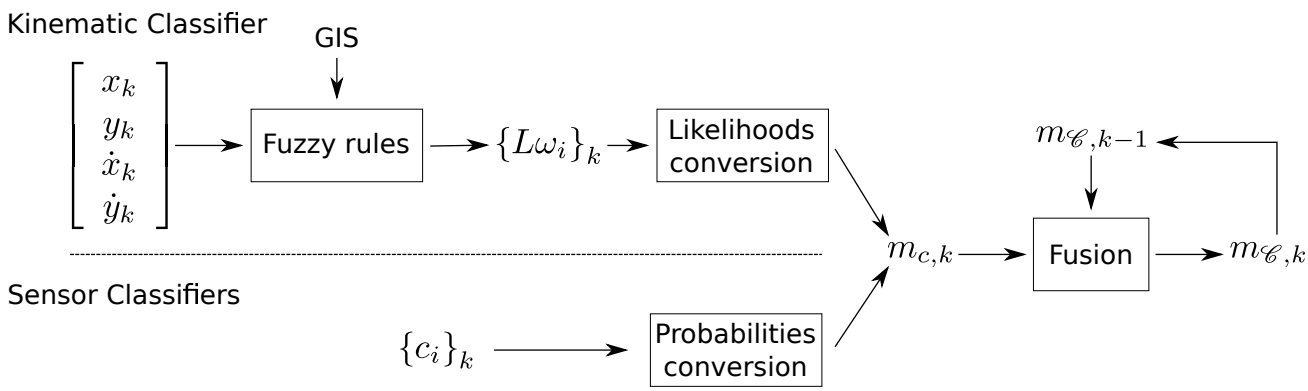

Figure 1. Representation of the sensor and kinematic based classifier

For each tracks, the mass function $m_{\mathscr{C},k}$ alongside the kinematics and terrain are stored, and the results will be presented in section 5.2.

5. RESULTS BASED ON REAL DATA

The feasibility of the developed algorithms deployed in fusion node has been tested on real data. To validate algorithms in our WSN system we have proceeded to operational exercise in October 2015. In this section we present the live recordings that were carried out in the Haguenau's military camp in the east of France. The playing scenario consists to deploy a sensor network for intelligence operation around a hunter hunt.

5.1 Scenario description

The Table 1 describes the types of sensors used during the exercise and their locations are provided on the figure 2. So the ground sensor network is composed of heterogeneous sensor. The tested architecture is a centralized architecture only and data fusion algorithm presented in this paper are tested in our C2 (Command and Control) station. The C2 station, with the tracker and classifier presented in this paper, has been coded in C++, on a x86 tablet, and works in real time.

To describe quickly the scenario, we give in figure 3 the trajectories of the fourth targets. One pedestrian is moving on the road from north to south to reach the hunter hunt. The pedestrian is taking metallic materials. Two armoured light vehicle are moving North-East from South-West to reach the hunter hunt. Both vehicles are moving in close formation. And a small UAV is moving to keep a discreet watch above the interested area. Note that the small UAV data are obtained on another exercise but introduced in our data base. The scenario duration is approximatively 5 minutes.

Sensor type	number	sensor node output	detection characteristic	comments
acoustic antenna	1	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	5	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	2	θ_k	mono,multi beam $<$ 200 m	Use to update existing track Use to create composite with another sensor
cameras: IR visible	6 2 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.

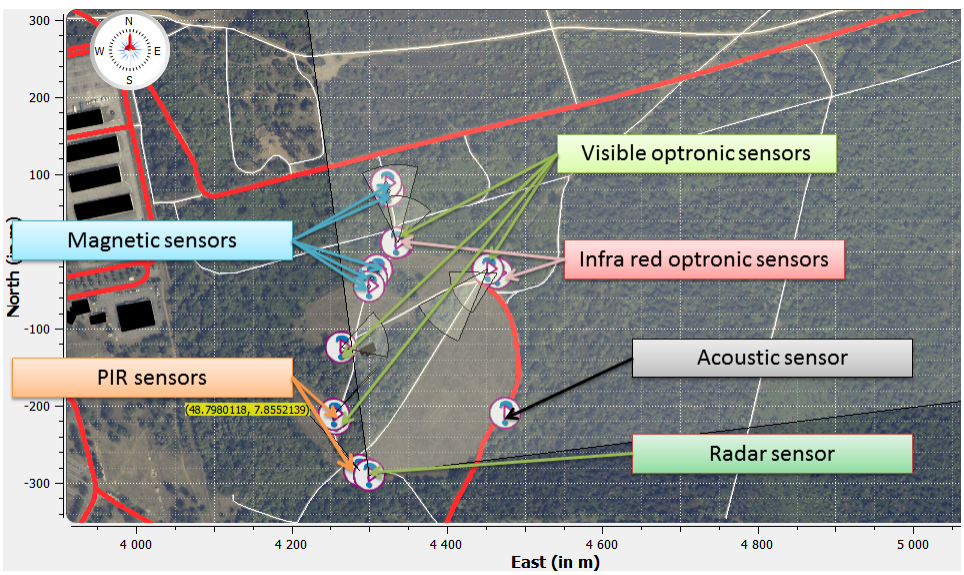

Figure 2. Sensor deployement.

5.2 Results

In this section, we present results (issued from the presented MTT algorithm) displayed in our C2 station named SAFIR.

At beginning of the exercise the pedestrian arrives on trials and is detected by magnetic sensor (see figure 4). A new track in blue (not confirmed) is initialized according magnetic sensor detection. According the road network information the track is constrained on the road network and allows to predict good state estimate.

This predicted track is updated in the sequel by visible and infrared video sensor and confirmed because the track becomes red (see figure 5). At East direction several tracks (corresponding to both armoured light vehicle) are initialized, merged and confirmed according radar detection. Afterwards, the merged track (in yellow because of operator track selection on the HMI) is updated with infrared and visible sensor (see figure 6). We can see

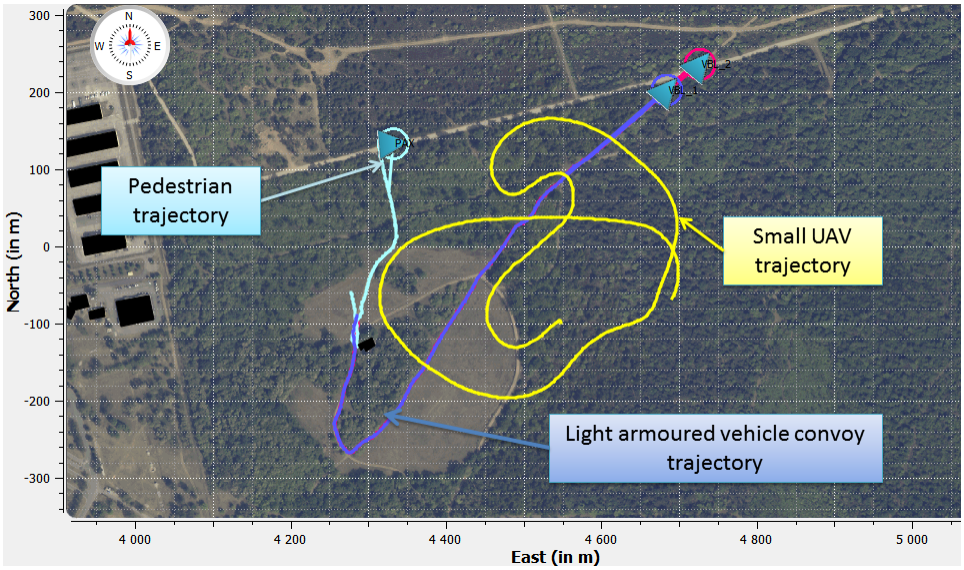

Figure 3. Trajectories of the targets.

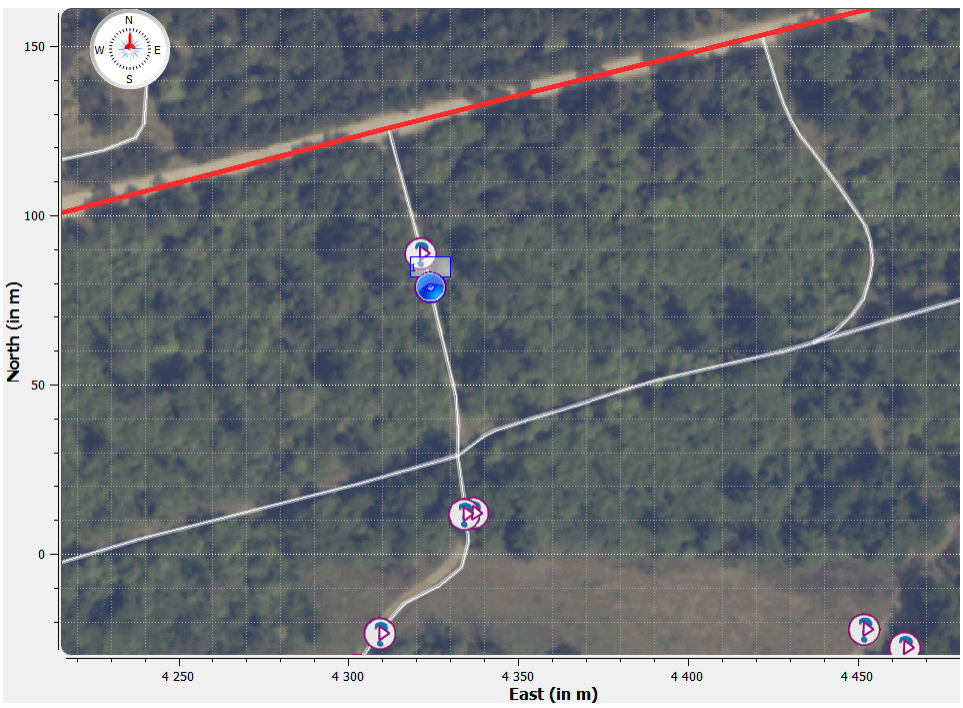

Figure 4. Tactical situation 1.

that the track has passed road intersection and is constrained on the road network.

Despite of false alarm and sensor resolution, the merge track (in yellow) is maintained (see figure 7). The track associated to the pedestrian is also maintained thanks to vegetation information considering the terrain mask. Then, the perception probability of the track is adapted according ray-tracing between estimated track location, sensor location and vegetation (this is the same sing with building information).

The pedestrian's track (in yellow because selected in the HMI) is updated with the next magnetic sensor group

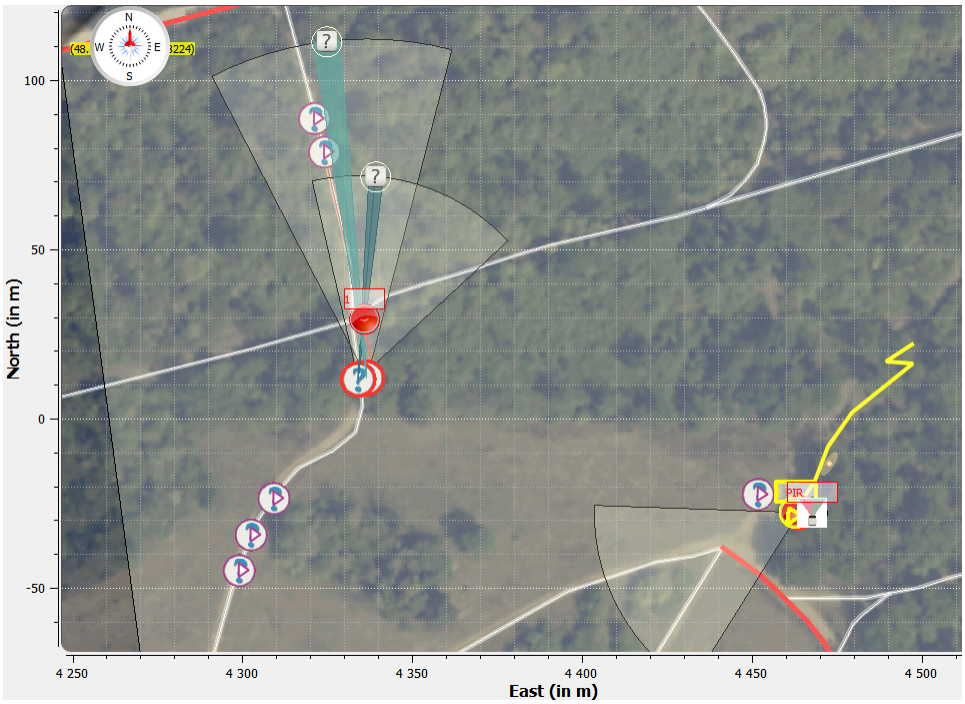

Figure 5. Tactical situation 2.

located at West (see figure 8). The merged track associated to the group of armoured light vehicles stays in stop motion model because it left the field of view of the radar sensor.

The small UAV begins its mission after the pedestrian and group vehicle stop. The MTT algorithm is able to track highly manoeuvrable target despite false alarms and sensor imprecision (see small UAV selected track in yellow in figure 9).

Finally, we are able to track in real time,within an heterogeneous sensor network, several target type. We will address n the following section the track classification according its behaviour using contextual information.

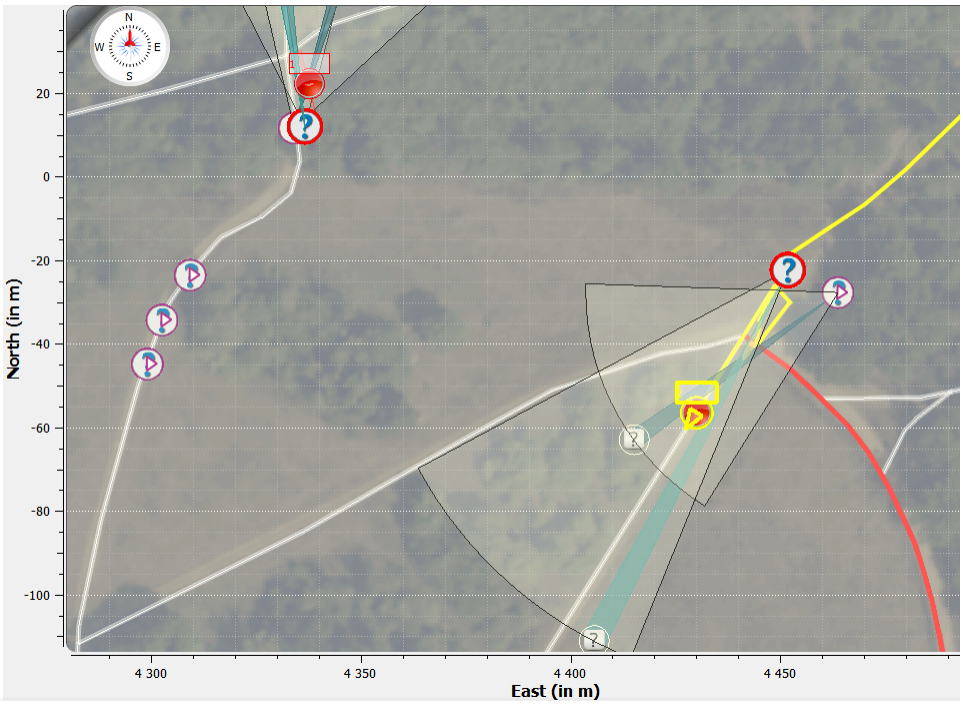

Figure 6. Tactical situation 3.

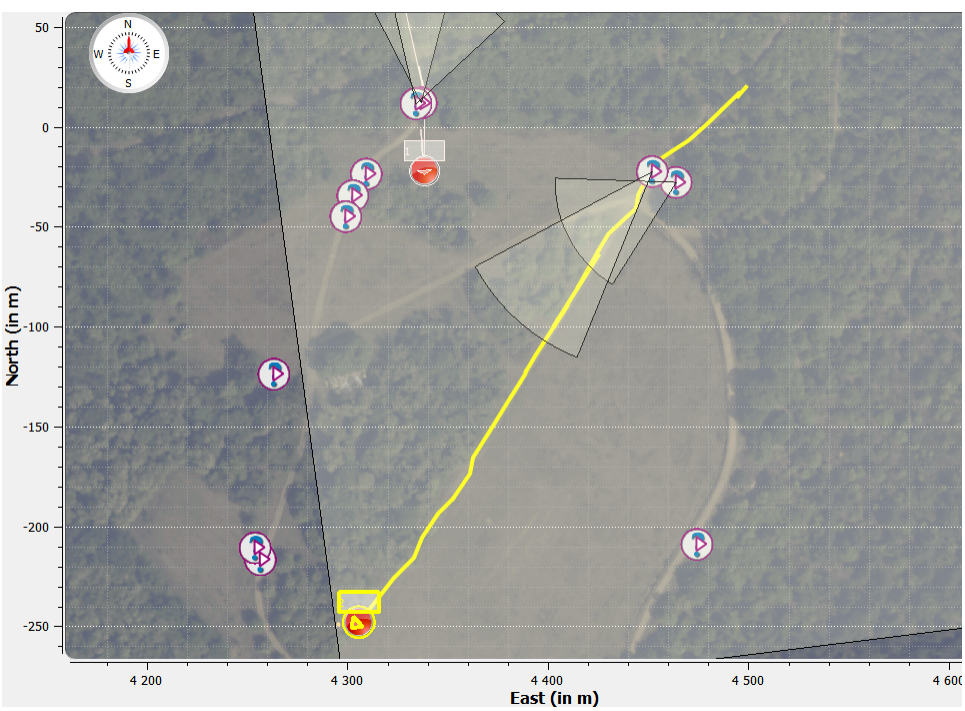

Figure 7. Tactical situation 4.

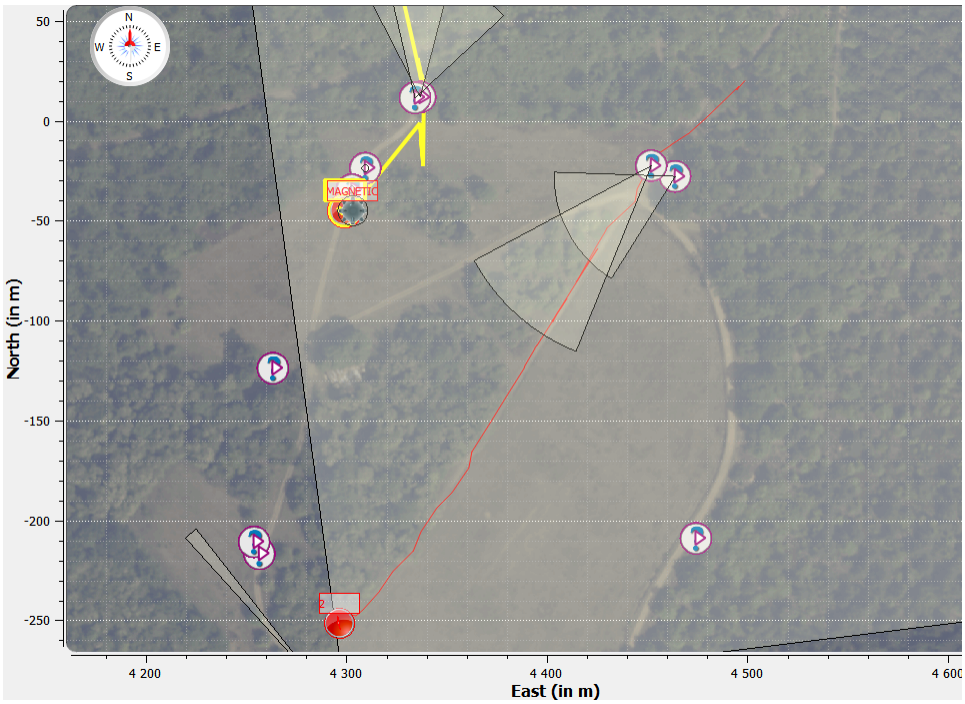

Figure 8. Tactical situation 5.

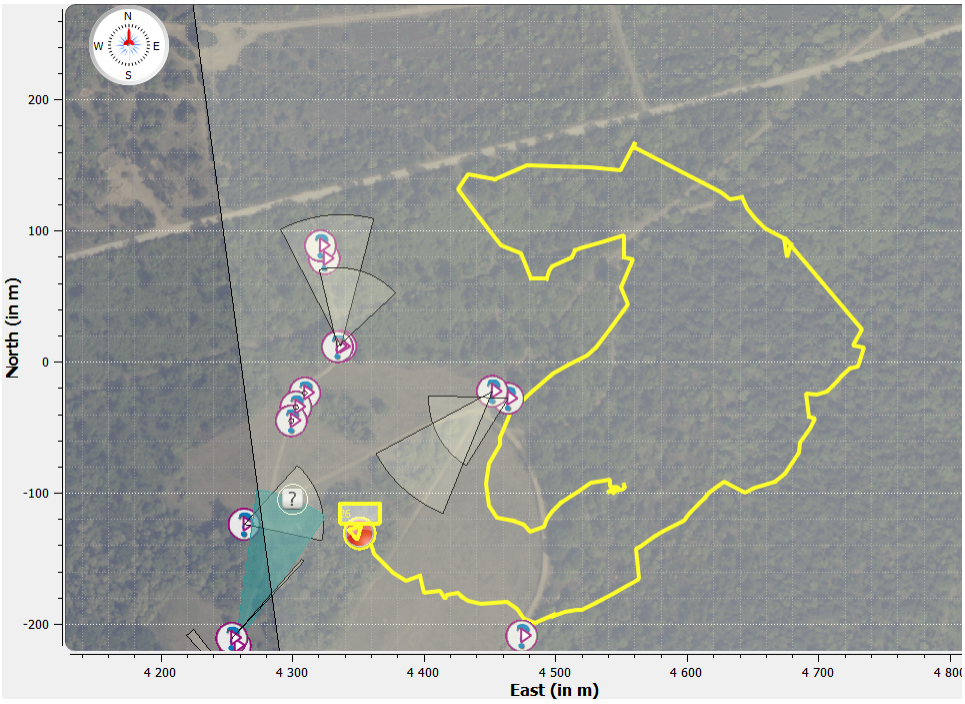

Figure 9. Tactical situation 6.

5.3 Classification result

In this section, we show the classification results of the tracks from the scenario described in 5.1. The classification results are computed directly from the C2 station, and stored in csv format in order to plot them.

Each result will be presented with the class' probabilities for the target, alongside its estimated velocity and its geographical layer.

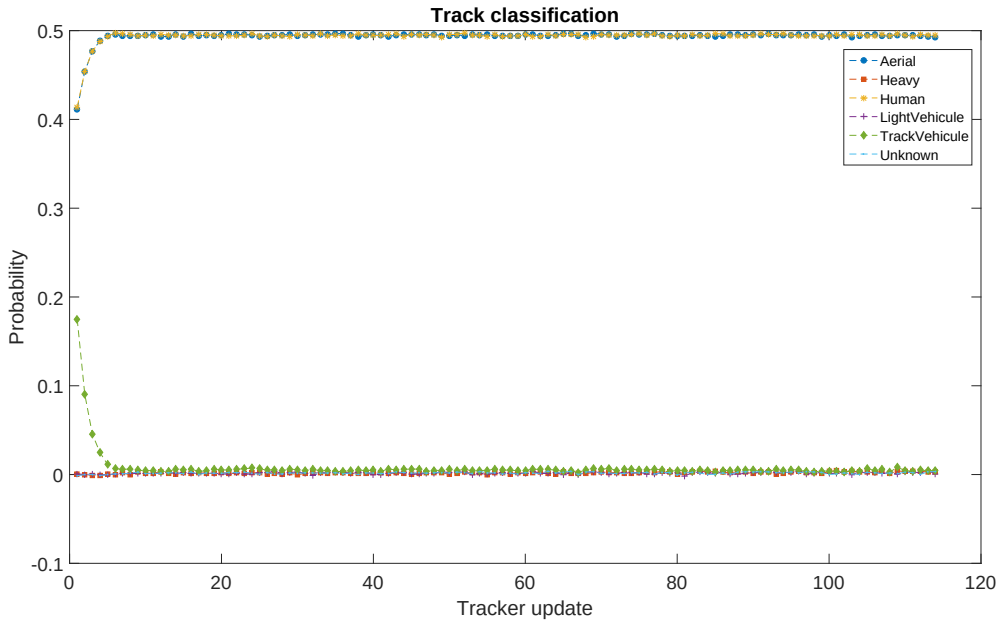

Figure 10. Classification of the pedestrian track

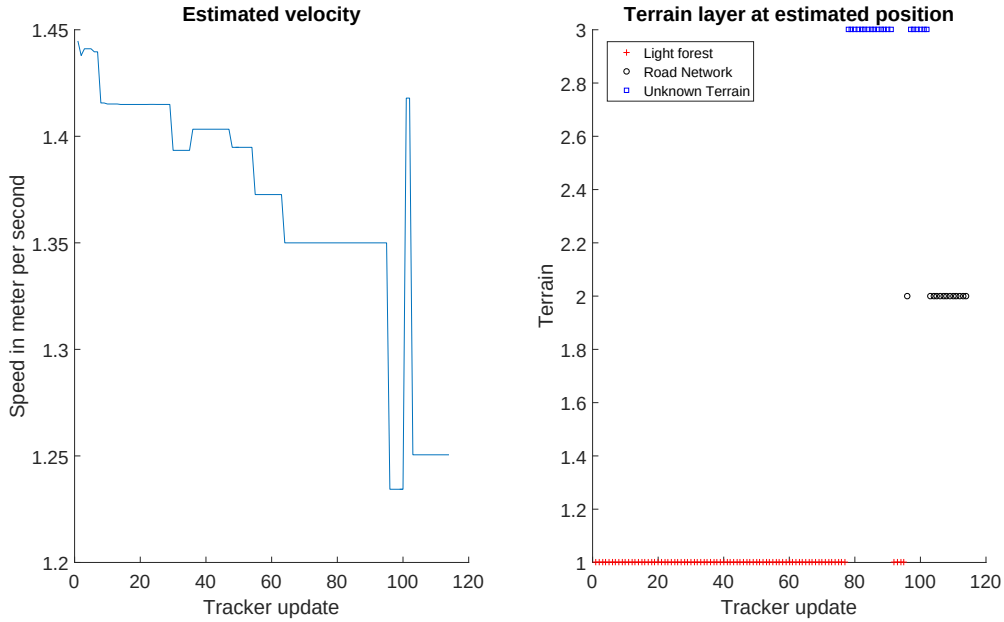

Figure 11. Velocity and terrain type for the pedestrian

We can see from the probabilities evolutions (see figure 10) that the classifier consider the Aerial class and the Human class as the most probable. This result seem coherent refering to the behavior of the track (see figure 11): a low speed target, evolving mostly in a light forest area and on the road.

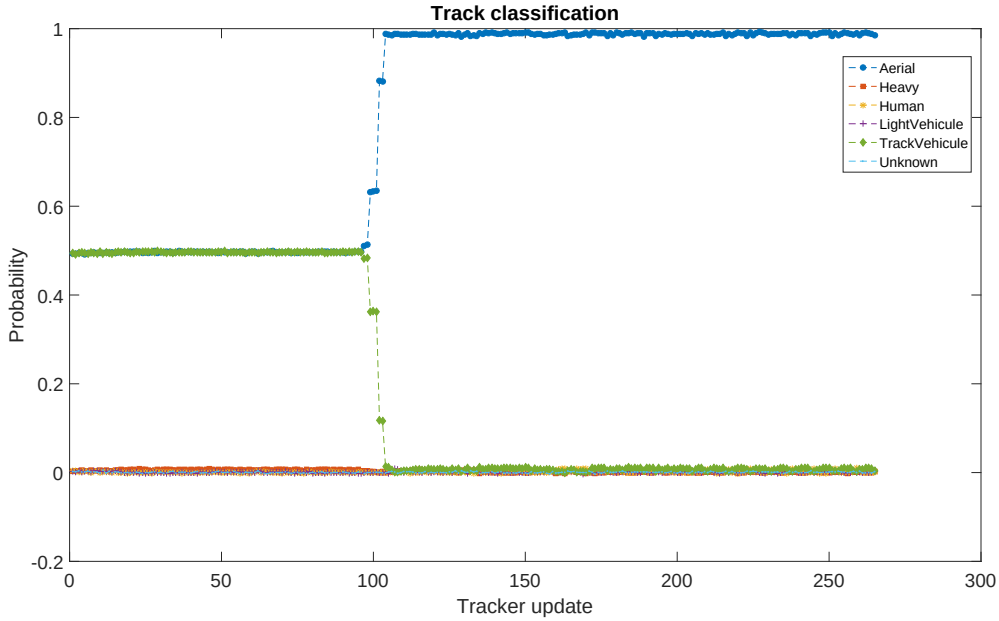

Figure 12. Classification of the Light Vehicule track

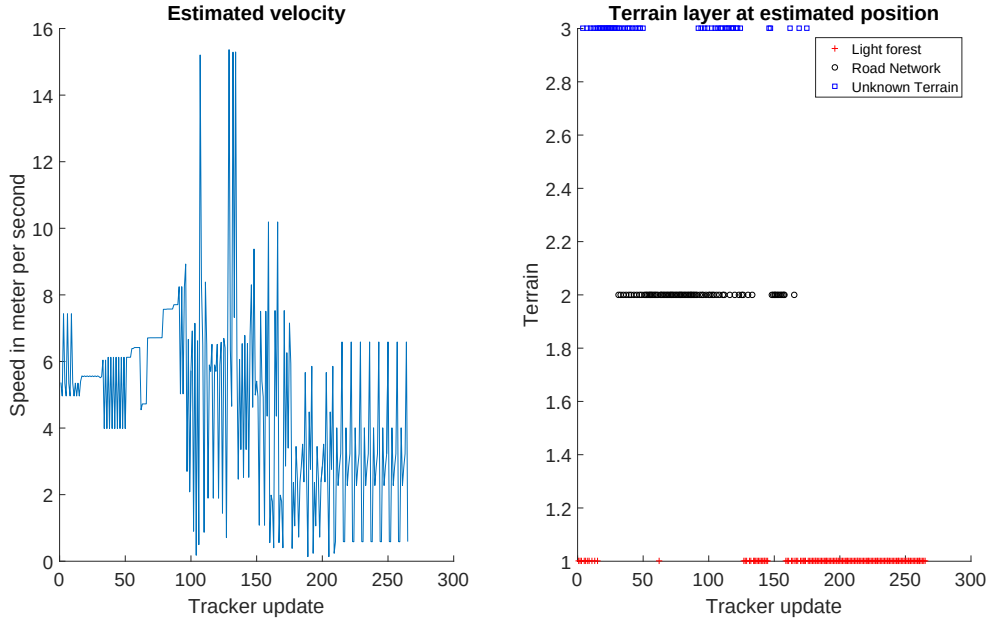

Figure 13. Velocity and terrain type for the light Vehicle

For the case of the light vehicule classification (see figure 12) we can see that the classifier fails to correctly identify the light vehicle, but instead, first it hesitates between the track vehicle and the aerial one, then discards the track vehicle one. By observing the terrain the target is evolving (see figure 13) this can be easily explained:

it can be seen that the vehicule evolves partialy in a light forest area. But, in our fuzzy inference system, based on expert knowledge, it has been considered that light vehicle does not have the capacity to evolve in a forest area, but only on roads. Hence, this leads to the discard of the light vehicule hypothesis. Furthermore, at some point the target has a velocity above 15 meters per seconds, which is the upper limit considered for the track vehicule, leading in this case to discard the track vehicle hypothesis, resulting in the UAV decision. This case show the limitation of our current system: by definition the track estimate position is not completely accurate. And by considering crisp definition of the geographical layer, this leads to misclassification of the terrain the target is evolving on, resulting then to misclassification of the target.

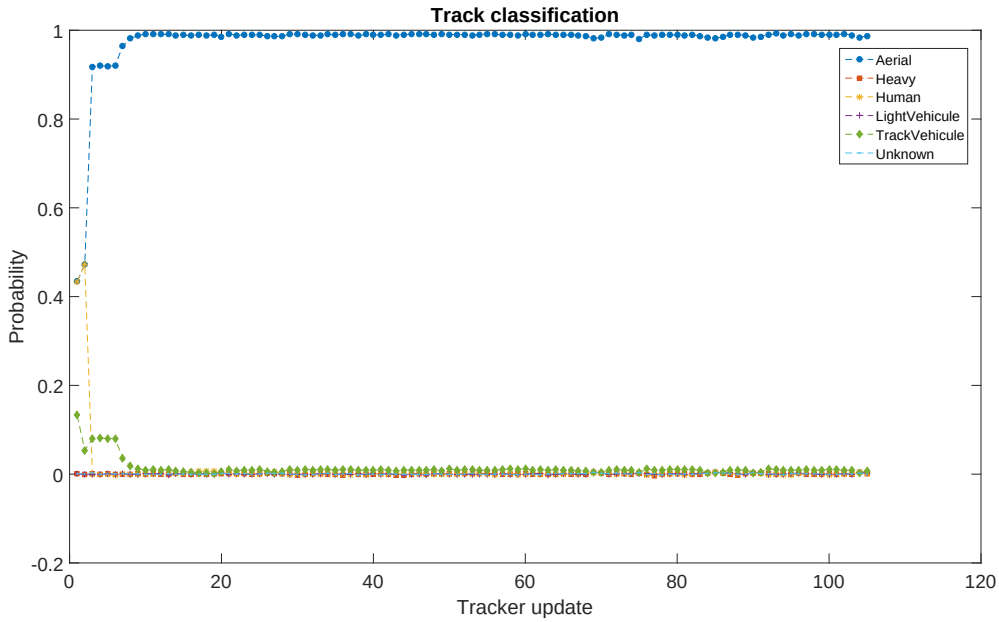

Figure 14. Classification of the UAV track

We can see from the figure 14 that the classifier is able in this case to quickly determine the UAV from it's behavior. Looking the the behavior of the target (see figure 15) ,at the beginning of the track, the target has a low velocity in a light forest area, and so three hypotheses are considered: the pedestrian, the track vehicle and the UAV. However, the target's velocity increase over time, which is a discriminant factor for the light forest layer, hence the classification of UAV.

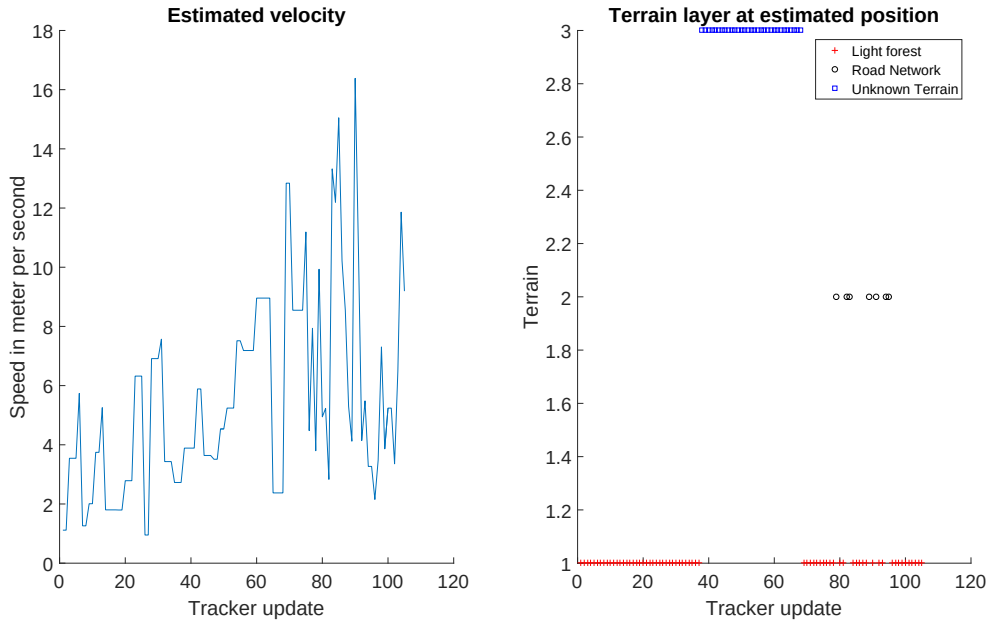

Figure 15. Velocity and terrain type for the UAV

6. CONCLUSION

In order to validate the MTT algorithm, we have proceeded to operational exercise with the French MOD. The MTT proposed algorithm should track and classify in real time unknown targets with several heterogeneous sensors. The emboarded algorithm takes into account several motion model, sensor performances and contextual information. We have proved with real data the capability to treat the assessment in real time with sensor network constraint.

Regarding the tracker, we are able to build tactical situation in real time. Some weakness have been detected as the group tracking (not studied yet), the need to use more motion model to reach better precision . . . Regarding the classifier, we have proceeded to build a behavioural context based track classification system. Despite the uneven results, the target classification using fuzzy logic seems promising, as a virtual sensor to give classification indication for the tracker. Different path can be explored as future work, such as the improvement of the prior used to build the membership functions,a better use of the GIS data by adding more uncertainty in the estimate layer, or how the fusion between our classifier and other sensors' classifier within the deployed sensor network can improve the tracking system. On the other hand, starting from those results, recommendations to use or develop new generation of abandoned sensors in operational exercises have been identified and are subjected to new research at ONERA.

7. ACKNOWLEDGMENTS

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

REFERENCES

1. Donka Angelova and Lyudmila Mihaylova. Joint target tracking and classification with particle filtering and mixture kalman filtering using kinematic radar information. *Digital Signal Processing*, 16(2):180–204, 2006.
2. V. Nimier B. Pannetier and M. Rombaut. Multiple ground target tracking with a gmti sensor. oct 2006.
3. S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. 1999.

4. I. Couso, L. Sanchez, and T. Denœux. Special issue: Harnessing the information contained in low-quality data sources likelihood-based belief function: Justification and some extensions to low-quality data. *International Journal of Approximate Reasoning*, 55:1535–1547, 2014.
5. M. Ekman and H. Palsson. Ground target tracking of vehicles in a wireless sensor network. 2012.
6. M. Ekman and E. Sviestins. Multiple model algorithm based on particle filters for ground target tracking. 2007.
7. W. Hong and W. Jie. *The Application of Intuitionistic Fuzzy Theory in Radar Target Identification*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
8. L. W. Jochumsen, E. Nielsen, J. Ostergaard, S. H. Jensen, and M. O. Pedersen. Using position uncertainty in recursive automatic target classification of radar tracks. In *Proc. of IEEE Radar Conference*, pages 0168–0173, May 2015.
9. T. Kirubarajan and Y. Bar-Shalom. Tracking evasive move-stop-move targets with an mti radar using a vs-imm estimator. *Proc. of IEEE Transactions aerospace and electronics systems*, 39:1098–1103, July 2003.
10. G. Kouemou and F. Opitz. Radar target classification in littoral environment with hmms combined with a track based classifier. pages 604–609, Sept 2008.
11. Y. Lei, Y. Lei, and W. Kong. Technique for target recognition based on intuitionistic fuzzy reasoning. *IET Signal Processing*, 6(3):255–263, May 2012.
12. H. Leung and J. Wu. Bayesian and dempster-shafer target identification for radar surveillance. *IEEE Transactions on Aerospace and Electronic Systems*, 36:432–447, 2000.
13. G. Manish and Upasna S. *Comparative Analysis of Diverse Approaches for Air Target Classification Based on Radar Track Data*, volume 1. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
14. Wei Mei, Gan lin Shan, and Yue feng Wang. A second-order uncertainty model for target classification using kinematic data. *Information Fusion*, 12(2):105 – 110, 2011.
15. M. Moruzzis and N. Colin. Radar target recognition by fuzzy logic. *IEEE Aerospace and Electronic Systems Magazine*, 13:13–20, 1998.
16. M. McDonald N. Nadarajah, R. Tharmarasa and T. Kirubarajan. Interacting multiple model forward filtering and backward smoothing for maneuvering target. *Proc. of IEEE Transactions aerospace and electronics systems*, October 2012.
17. B. Pannetier and J. Dezert. Track segment association with classification information. *Sensor Data Fusion: Trends Solutions Applications (SDF),Workshop on*, 2012.
18. B. Pannetier, J. Dezert, and J. Moras. Ground target tracking and classification in an unattended wireless sensor network. May 2015.
19. B. Pannetier, J. Dezert, and G. Sella. Multiple target tracking with wireless sensor network for ground battlefield surveillance. Jul 2014.
20. Nozer D Singpurwalla and Jane M Booker. Membership functions and probability measures of fuzzy sets. *Journal of the American Statistical Association*, 99(467):867–877, 2004.
21. Philippe Smets. Decision making in the tbm: the necessity of the pignistic transformation. *International Journal of Approximate Reasoning*, 38(2):133 – 147, 2005.
22. Philippe Smets and Robert Kennes. The transferable belief model. *Artificial Inte lligence*, 66(2):191 – 234, 1994.
23. M. Sundkvist and N. Bergman. Fixed lag smoothing for target tracking applications based on imm. *Proc. of IEE worshop target tracking: algorithms and applications*, 2, October 2001.
24. M. Ulmke and W. Koch. Road-map assisted ground moving target tracking. *IEEE Transactions on Aerospace and Electronic Systems*, (42):1264–1274, 2006.
25. X.-R. Li Y. Bar-Shalom and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. 2001.