Automatic Aircraft Recognition using DSmT and HMM

Xin-de Li
Jin-dong Pan
Jean Dezert

Originally published as Li X.-D., Pan J.-D., Dezert J., A*utomatic Aircraft Recognition using DSmT and HMM*, in Proc. of Fusion 2014 Int Conf onInformation Fusion, Salamanca, Spain, July 7-10, 2014, and reprinted with permission.

***Abstract*—In this paper we propose a new method for solving the Automatic Aircraft Recognition (AAR) problem from a sequence of images of an unknown observed aircraft. Our method exploits the knowledge extracted from a training image data set (a set of binary images of different aircrafts observed under three different poses) with the fusion of information of multiple features drawn from the image sequence using Dezert-Smarandache Theory (DSmT) coupled with Hidden Markov Models (HMM). The first step of the method consists for each image of the observed aircraft to compute both Hu's moment invariants (the first features vector) and the partial singular values of the outline of the aircraft (the second features vector). In the second step, we use a probabilistic neural network (PNN) based on the training image dataset to construct the conditional basic belief assignments (BBA's) of the unknown aircraft type within the set of a predefined possible target types given the features vectors and pose condition. The BBA's are then combined altogether by the Proportional Conflict Redistribution rule #5 (PCR5) of DSmT to get a global BBA about the target type under a given pose hypothesis. These sequential BBA's give initial recognition results that feed a HMM-based classifier for automatically recognizing the aircraft in a multiple poses context. The last part of this paper shows the effectiveness of this new Sequential Multiple-Features Automatic Target Recognition (SMF-ATR) method with realistic simulation results. This method is compliant with real-time processing requirement for advanced AAR systems.**

Keywords: Information fusion; DSmT; ATR; HMM.

I. Introduction

ATR (Automatic Target Recognition) systems play a major role in modern battlefield for automatic monitoring and detection, identification and for precision guided weapon as well. The Automatic Aircraft Recognition (AAR) problem is a subclass of the ATR problem. Many scholars have made extensive explorations for solving ATR and AAR problems. The ATR method is usually based on target recognition using template matching [1], [2] and single feature (SF) extraction [3]–[7] algorithms. Unfortunately, erroneous recognition often occurs when utilizing target recognition algorithms based on single feature only, specially if there exist important changes in pose and appearance of aircrafts during flight path in the image sequence. In such condition, the informational content drawn from single feature measures cannot help enough to make a reliable classification. To overcome this serious drawback, new ATR algorithms based on multiple features (MF) and fusion techniques have been proposed [8]–[12]. An interesting MF-ATR algorithm based on Back-Propagation Neural Network (BP-NN), and Dempster-Shafer Theory (DST) of evidence [23] has been proposed by Yang et al. in [11] which has been partly the source of inspiration to develop our new improved sequential MF-ATR method presented here and introduced briefly in [12] (in chinese). In this paper we will explain in details how our new SMF-ATR method works and we evaluate its performances on a typical real image sequence.

Although MF-ATR approach reduces the deficiency of SF-ATR approach in general, the recognition results can sometimes still be indeterminate form a single image exploitation because the pose and appearance of different kinds of aircrafts can be very similar for some instantaneous poses and appearances. To eliminate (or reduce) uncertainty and improve the classification, it is necessary to exploit a sequence of images of the observed aircraft during its flight and develop efficient techniques of sequential information fusion for advanced (sequential) MF-ATR systems. Two pioneer works on sequential ATR algorithms using belief functions (BF) have been proposed in last years. In 2006, Huang et al. in [13] have developed a sequential ATR based on BF, Hu's moment invariants (for image features vector), a BP-NN for pattern classification, and a modified Dempster-Shafer (DS) fusion rule[1]. A SF-ATR approach using BF, Hu's moment invariants, BP-NN and DSmT rule has also been proposed in [14] the same year. In these papers, the authors did clearly show the benefit of the integration of temporal SF measures for the target recognition, but the performances obtained were still limited because of large possible changes in poses and appearances of observed aircrafts (specially in high maneuver modes as far as military aircrafts are under concern). The purpose of this paper is to develop a new (sequential) MF-ATR method able to provide a high recognition rate with a good robustness when face to large changes of poses and ppearances of observed aircraft during its flight.

The general principle of our SMF-ATR method is shown on Fig.1. The upper part of Fig. 1 consists in Steps 1 & 2, whereas the lower part of Fig. 1 consists in Steps 3 & 4 respectively described as follows:

- Step 1 (Features extraction) : We consider and extract only two features vectors in this work[2] (Hu's moment

[1] called the abortion method by the authors.
[2] The introduction of extra features is possible and under investigations.

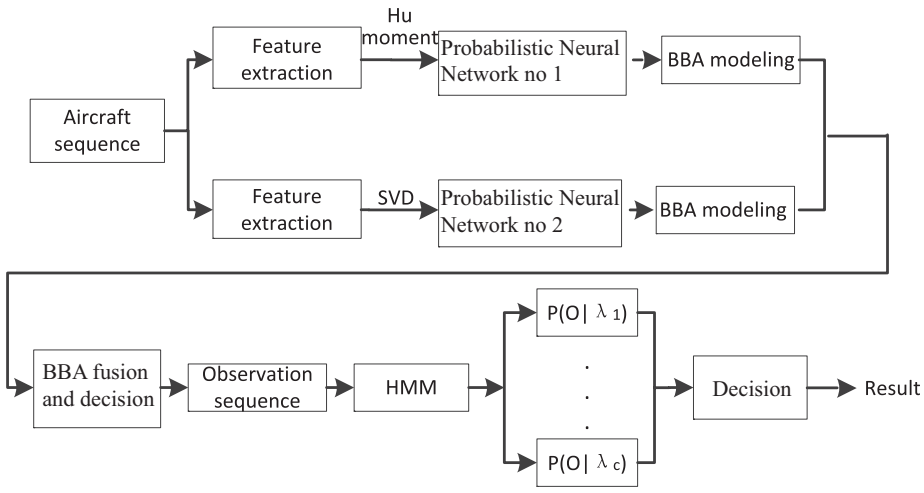

Fig. 1: General principle of our sequential MF-ATR approach.

invariants vector, and Singular Values Decomposition (SVD) features vector) from the binary images[3]

- Step 2 (BBA's construction[4]) : For every image in the sequence and from their two features vectors, two Bayesian BBA's on possible (target type,target pose) are computed from the results of two PNN's trained on the image dataset. The method of BBA construction is different from the one proposed in [12].
- Step 3 (BBA's combination) : For every image, say the k-th image, in the sequence, the two BBA's of Step 2 are combined with the PCR5 fusion rule, from which a decision O_k on the most likely target type and pose is drawn.
- Step 4 (HMM-based classifier) : From the sequence $O^K = \{O_1, \ldots, O_k \ldots, O_K\}$ of K local decisions computed at Step 3, we feed several HMM-based classifiers in parallel (each HMM characterizes each target type) and we find finally the most likely target observed in the image sequence which gives the output of our SMF-ATR approach.

The next section presents each step of this new SMF-ATR approach. Section 3 evaluates the performances of this new method on real image datasets. Conclusions and perspectives of this work are given in Section 4.

II. The sequential MF-ATR approach

In this section we present the aforementioned steps necessary for the implementation of our new SMF-ATR method.

[3]In this work, we use only with binary images because our image training dataset contains only binary images with clean backgrounds, and working with binary images is easier to do and requires less computational burden than working with grey-level or color images. Hence it helps to satisfy real-time processing. The binarization of the images of the sequence under analysis is done with the the Flood Fill Method explained in details in [22] using the point of the background as a seed for the method.

[4]The mathematical definition of a BBA is given in Section II-C.

A. Step 1: Features extraction from binary image

Because Aircraft poses in a flight can vary greatly, we need image features that are stable and remain unchanged under translation, rotation and scaling. In terms of aircraft features, two categories are widely used: 1) moment features and 2) contour features. Image moments have been widely used since a long time specially for pattern-recognition applications [16]. Moment features which are the descriptions of image regional characteristics are mainly obtained from the intensity of each pixel of target image. Contour features are extracted primarily by discretizing the outline contour and they describe the characteristic of the outline of the object in the image. In terms of moment features, Hu's moment invariants [6] are used here. As contour features, we use the SVD [15] of outlines extracted from the binary images.

• Hu's moments

Two-dimensional $(p+q)$-th order moments for $p, q = 0, 1, 2, ...$ of an image of size $M \times N$ are defined as follows:

$$m_{pq} \triangleq \sum_{m=1}^{M} \sum_{n=1}^{N} m^p n^q f(m,n) \tag{1}$$

where $f(m,n)$ is the value of the pixel (m,n) of the binary image. Note that m_{pq} may not be invariant when $f(m,n)$ by translation, rotating or scaling. The invariant features can be obtained using the $(p+q)$-th order central moments μ_{pq} for $p, q = 0, 1, 2, ...$ defined by

$$\mu_{pq} \triangleq \sum_{m=1}^{M} \sum_{n=1}^{N} (m-\bar{x})^p (n-\bar{y})^q f(m,n) \tag{2}$$

where $\bar{x}$, and $\bar{y}$ are the barycentric coordinates of image (i.e. the centroid of the image). These values are computed by $\bar{x} = \frac{m_{10}}{m_{00}} = \frac{1}{C}\sum_{m=1}^{M}\sum_{n=1}^{N} m \times f(m,n)$ and $\bar{y} = \frac{m_{01}}{m_{00}} = \frac{1}{C}\sum_{m=1}^{M}\sum_{n=1}^{N} n \times f(m,n)$, where C is a normalization constant given by $C = m_{00} = \sum_{m=1}^{M}\sum_{n=1}^{N} f(m,n)$. The centroid moments μ_{pq} is equivalent to the m_{pq} moment whose

center has been shifted to the centroid of the image. Therefore, μ_{pq} are invariant to image translations. Scale invariance is obtained by normalization [6]. The normalized central moments η_{pq} are defined for $p+q=2,3,\ldots$ by $\eta_{pq} \triangleq \mu_{pq}/\mu_{00}^{\gamma}$, with $\gamma=(p+q+2)/2$. Based on these normalized central moments Hu in [16] derived seven moment invariants that are unchanged under image scaling, translation and rotation as follows

$$\Phi_1 \triangleq \eta_{20}+\eta_{02}$$
$$\Phi_2 \triangleq (\eta_{20}-\eta_{02})^2+4\eta_{11}^2$$
$$\Phi_3 \triangleq (\eta_{30}-3\eta_{12})^2+(3\eta_{21}-\eta_{03})^2$$
$$\Phi_4 \triangleq (\eta_{30}+\eta_{12})^2+(\eta_{21}+\eta_{03})^2$$
$$\Phi_5 \triangleq (\eta_{30}-3\eta_{12})(\eta_{30}+\eta_{12})[(\eta_{30}+\eta_{12})^2-3(\eta_{21}+\eta_{03})^2] + (3\eta_{21}-\eta_{03})(\eta_{21}+\eta_{03})[3(\eta_{30}+\eta_{12})^2-(\eta_{21}+\eta_{03})^2]$$
$$\Phi_6 \triangleq (\eta_{20}-\eta_{02})[(\eta_{30}+\eta_{12})^2-(\eta_{21}+\eta_{03})^2] + 4\eta_{11}(\eta_{30}+\eta_{12})(\eta_{21}+\eta_{03})$$
$$\Phi_7 \triangleq (3\eta_{21}-\eta_{03})(\eta_{30}+\eta_{12})[(\eta_{30}+\eta_{12})^2-3(\eta_{21}+\eta_{03})^2] - (\eta_{30}-3\eta_{12})(\eta_{21}+\eta_{03})[3(\eta_{30}+\eta_{12})^2-(\eta_{03}+\eta_{21})^2]$$

In this work, we use only the four simplest Hu's moments to compute, that is $\mathbf{\Phi}=[\Phi_1\ \Phi_2\ \Phi_3\ \Phi_4]$, to feed the first PNN of our sequential MF-ATR method[5].

• SVD features of the target outline

The SVD is widely applied signal and image processing because it is an efficient tool to solve problems with least squares method [21]. The SVD theorem states that if $\mathbf{A}_{m\times n}$ with $m>n$ (representing in our context the original binary data) is a real matrix[6], then it can be written using a so-called singular value decomposition of the form

$$\mathbf{A}_{m\times n}=\mathbf{U}_{m\times m}\mathbf{S}_{m\times n}\mathbf{V}_{n\times n}^T$$

where $\mathbf{U}_{m\times m}$ and $\mathbf{V}_{n\times n}$ are orthogonal[7] matrices. The columns of $\mathbf{U}$ are the left singular vectors. $\mathbf{V}^T$ has rows that are the right singular vectors. The real matrix $\mathbf{S}$ has the same dimensions as $\mathbf{A}$ and has the form[8]

$$\mathbf{S}_{m\times n}=\begin{bmatrix}\mathbf{S}_{r\times r} & \mathbf{0}_{r\times(n-r)}\\ \mathbf{0}_{r\times(m-r)} & \mathbf{0}_{(m-r)\times(n-r)}\end{bmatrix}$$

where $\mathbf{S}_{r\times r}=Diag\{\sigma_1,\sigma_2,\ldots,\sigma_r\}$ with $\sigma_1\geq\sigma_2,\geq\ldots\geq\sigma_r>0$ and $1\leq r\leq\min(m,n)$.

Calculating the SVD consists of finding the eigenvalues and eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$. The eigenvectors of $\mathbf{A}^T\mathbf{A}$ make up the columns of $\mathbf{V}$, the eigenvectors of $\mathbf{A}\mathbf{A}^T$ make up the columns of $\mathbf{U}$. The singular values σ_1,..., σ_r are the diagonal entries of $\mathbf{S}_{r\times r}$ arranged in descending order, and they are square roots of eigenvalues from $\mathbf{A}\mathbf{A}^T$ or $\mathbf{A}^T\mathbf{A}$.

A method to calculate the set of discrete points $\{a_1,a_2,\ldots,a_n\}$ of a target outline from a binary image is proposed in [17]. The SVD features are then computed from the eigenvalues of the circulant matrix built from the discretized shape of the outline characterized by the vector $\mathbf{d}=[d_1,d_2,\ldots,d_n]$ where d_i is the distance of the centroid of the outline to the discrete points a_i, $i=1,2,\ldots,n$ of the outline.

In our analysis, it has been verified from our image dataset that only the first components of SVD features vector $\boldsymbol{\sigma}=[\sigma_1,\sigma_2,\ldots,\sigma_r]$ take important values with respect to the other ones. The other components of $\boldsymbol{\sigma}$ tend quickly towards zero. Therefore only few first components of $\boldsymbol{\sigma}$ play an important role to characterize the main features of target outline. However, if one considers only these few main first components of $\boldsymbol{\sigma}$, one fails to characterize efficiently some specific features (details) of the target profile. By doing so, one would limit the performances of ATR. That is why we propose to use the partial SVDs of outline as explained in the next paragraph.

To capture more details of aircraft outline with SVD, one has to taken into account also additional small singular values of SVD. This is done with the following procedure issued from the face recognition research community [24]. The normalized distance vector $\tilde{\mathbf{d}}=[\tilde{d}_1,\tilde{d}_2,\ldots,\tilde{d}_n]$ is built from $\mathbf{d}$ by taking $\tilde{\mathbf{d}}=[1,d_2/d_1,\ldots,d_n/d_1]$, where d_1 is the distance between the centroid of outline and the first chosen points of the contour of the outline obtained by a classical[9] edge detector algorithm. To capture the details of target outline and to reduce the computational burden, one works with partial SVDs of the original outline by considering only l sliding sub-vectors $\tilde{\mathbf{d}}_w$ of $\tilde{\mathbf{d}}$, where w is the number of components of $\tilde{\mathbf{d}}_w$. For example if one takes $w=3$ points only in the sub-vectors and if $\tilde{\mathbf{d}}=[\tilde{d}_1,\tilde{d}_2,\ldots,\tilde{d}_9]$, then one will take the sub-vectors $\tilde{\mathbf{d}}_w^1=[\tilde{d}_1,\tilde{d}_2,\tilde{d}_3]$, $\tilde{\mathbf{d}}_w^2=[\tilde{d}_4,\tilde{d}_5,\tilde{d}_6]$ and $\tilde{\mathbf{d}}_w^3=[\tilde{d}_7,\tilde{d}_8,\tilde{d}_9]$ if we don't use overlapping components between sub-vectors. From the sub-vectors, one constructs their corresponding circulant matrix and apply their SVD to get partial SVD features vectors $\boldsymbol{\sigma}_w^{l=1}$, $\boldsymbol{\sigma}_w^{l=2}$, etc. The number l of partial SVD of the original outline of the target is given by $l=(n-w)/(w-m)+1$, where m is the number of components overlapped by each two adjacent sub-vectors, and n is the total number of discrete contour points of the outline given by the edge detector.

B. Step 2: BBA's construction with PNN's

In order to exploit efficiently fusion rules dealing with conflicting information modeled by belief mass assignments (BBA's) [18], [23], we need to build BBA's from all features computed from images of the sequence under analysis. The construction of the BBA's needs expert knowledge or knowledge drawn from training using image dataset. In this paper, we propose to utilize probabilistic neural networks (PNN) initially developed in nineties by Specht [19] to construct the BBA's because it is a common technique used in the target recognition and pattern classification community that is able to

[5]It is theoretically possible to work with all seven Hu's moments in our MF-ATR method, but we did not test this yet in our simulations.

[6]For a complex matrix $\mathbf{A}$, the singular value decomposition is $\mathbf{A}=\mathbf{U}\mathbf{S}\mathbf{V}^H$, where $\mathbf{V}^H$ is the conjugate transpose of $\mathbf{V}$.

[7]They verify $\mathbf{U}_{m\times m}^T\mathbf{U}_{m\times m}=\mathbf{I}_{m\times}$ and $\mathbf{V}_{n\times n}^T\mathbf{V}_{n\times n}=\mathbf{I}_{n\times n}$, where $\mathbf{I}_{m\times m}$ and $\mathbf{I}_{n\times n}$ are respectively the identity matrices of dimensions $m\times m$ and $n\times n$.

[8]$\mathbf{0}_{p\times q}$ is a $p\times q$ matrix whose all its elements are zero.

[9]In this work, we use the *cvcontour* function of opencv software [22] to extract the target outline from a binary image.

achieve with large training dataset performances close to those obtained by a human expert in the field. The details of PNN's settings for BBA's construction are given in [12]. However, because the neural network after training to some extent has a good discriminant ability (close to an expert in the field), the BBA is constructed by the neural network directly based on the PNN's output, which is different from the construction of the BBA based on the confusion matrix described in [12].

Here we present how the two PNN's (shown in Figure 1) work. In our application, we have $N_c = 7$ types of aircrafts in our training image dataset. For each type, the aircraft is observed with $N_p = 3$ poses. Therefore we have $N_{cp} = N_c \times N_p = 21$ types of distinct cases in our dataset. For each case, one has $N_i = 30$ images available for the training. Therefore the whole training dataset contains $N_{cpi} = N_c N_p N_i = 7 \times 3 \times 30 = 630$ binary images. For the first PNN (fed by Hu's features vector), the number of input layer neurons is 4 because we use only $\mathbf{\Phi} = [\Phi_1, \Phi_2, \Phi_3, \Phi_4]$ Hu's moment invariants in this work. For the second PNN (fed by partial SVD features vector), the number of input layer neurons is constant and equal to $l \times w$ because we take l windows with the width w (so one has w singular values of partial SVD for every window). The number of hidden layer neurons of each PNN is the number of the training samples, $N_{cpi} = 630$. The number of output layer neurons is equal to $N_{cp} = 21$ (the number of different possible cases).

Our PNN's fed by features input vectors (Hu's moments and SVD outline) do not provide a hard decision on the type and pose of the observed target under analysis because in our belief-based approach we need to build BBA's. Therefore the competition function of the output layer for decision-making implemented classically in the PNN scheme is not used in the exploitation[10] phase of our approach. Instead, the PNN computes the $N_{cp} \times N_i$ (Euclidean) distances between the features vectors of the image under test and the $N_{cpi} = 630$ features vectors of the training dataset. A Gaussian radial basis function (G-RBF) is used in the hidden layer of the PNN's [19] to transform its input (Euclidean) distance vector of size $1 \times N_{cpi}$ into another $1 \times N_{cpi}$ distance (similarity) that feeds the output layer through a weighting matrix of size $N_{cpi} \times N_{cp} = 630 \times 21$ estimated from the training samples. As a final output of each PNN, we get an unnormalized similarity vector $\mathbf{m}$ of size $(1 \times N_{cpi}) \times (N_{cpi} \times N_{cp}) = 1 \times N_{cp} = 1 \times 21$ which is then normalized to get a Bayesian BBA on the frame of discernment $\Theta = \{(target_i, pose_j), i = 1, \ldots, c, j = 1, \ldots, p\}$. Because we use only two[11] PNN's in this approach, we are able to build two Bayesian BBA's $m_1(.)$ and $m_2(.)$ defined on the same frame Θ for every image of the sequence to analyze.

C. Step 3: Fusion of BBA's and local decision

A basic belief assignment (BBA), also called a (belief) mass function, is a mapping $m(.) : 2^\Theta \mapsto [0; 1]$ such that $m(\emptyset) = 0$ and $\sum_{X \in 2^\Theta} m(X) = 1$, where Θ is the so-called frame of discernment of the problem under concern which consists of a finite discrete set of exhaustive and exclusive hypotheses[12] θ_i, $i = 1, \ldots, n$, and where 2^Θ is the power-set of Θ (the set of all subsets of Θ). This definition of BBA has been introduced in Dempster-Shafer Theory (DST) [23]. The focal elements of a BBA are all elements X of 2^Θ such that $m(X) > 0$. Bayesian BBA's are special BBA's having only singletons (i.e. the elements of Θ) as focal elements.

In DST, the combination of BBA's is done by Dempster's rule of combination [23] which corresponds to the normalized conjunctive consensus operator. Because this fusion rule is known to be not so efficient (both in highly and also in low conflicting) in some practical situations [25], many alternative rules have been proposed during last decades [18], Vol. 2.

To overcome the practical limitations of Shafers' model and in order to deal with fuzzy hypotheses of the frame, Dezert and Smarandache have proposed the possibility to work with BBA's defined on Dedekind's lattice[13] D^Θ [18] (Vol.1) so that intersections (conjunctions) of elements of the frame can be allowed in the fusion process, with eventually some given restrictions (integrity constraints). Dezert and Smarandache have also proposed several rules of combination based on different Proportional Conflict Redistribution (PCR) principles. Among these new rules, the PCR5 and PCR6 rules play a major role because they do not degrade the specificity of the fusion result (contrariwise to most other alternative rule), and they preserve the neutrality of the vacuous BBA[14]. PCR5 and PCR6 provide same combined BBA when combining only two BBA's $m_1(.)$ and $m_2(.)$, but they differ when combining three (or more) BBA's altogether. It has been recently proved in [26] that PCR6 is consistent with empirical (frequentist) estimation of probability measure, unlike other fusion rules[15].These two major differences with DST, make the basis of Dezert-Smarandache Theory (DSmT) [18].

In the context of this work, we propose to use PCR5 to combine the two (Bayesian) BBA's $m_1(.)$ and $m_2(.)$ built from the two PNN's fed by Hu's features vector and SVD outline features vector. Because for each image of the observed target in the sequence, one has only two BBA's to combine, the PCR5 fusion result is same as the PCR6 fusion result. Of course, if one wants to include other kinds of features vectors with additional PNN's, the PCR6 fusion rule is recommended. The PCR principle consists in redistributing the partial conflicting masses[16]only to the sets involved in the conflict and proportionally to their mass. The PCR5 (or PCR6) combination of

[10] when analyzing a new sequence of an unknown observed aircraft.

[11] A first PPN fed by Hu's features, and a second PNN fed by SVD outline features – see Fig. 1.

[12] This is what is called Shafer's model of the frame in the literature.

[13] Dedekind's lattice is the set of all composite subsets built from elements of Θ with $\cup$ and $\cap$ operators.

[14] A vacuous BBA is the BBA such that $m(\Theta) = 1$.

[15] except the averaging rule.

[16] For two BBA's, a partial conflicting mass is a product $m_1(X)m_2(Y) > 0$ of the element $X \cap Y$ which is conflicting, that is such that $X \cap Y = \emptyset$.

two BBA's is done according to the following formula[17] [18]

$$m_{PCR5/6}(X) = \sum_{\substack{X_1,X_2\in 2^\Theta \\ X_1\cap X_2 = X}} m_1(X_1)m_2(X_2) + \sum_{\substack{Y\in 2^\Theta\setminus\{X\} \\ X\cap Y=\emptyset}} \left[\frac{m_1(X)^2 m_2(Y)}{m_1(X)+m_2(Y)} + \frac{m_2(X)^2 m_1(Y)}{m_2(X)+m_1(Y)}\right] \quad (3)$$

where all denominators in (3) are different from zero, and $m_{PCR5/6}(\emptyset) = 0$. If a denominator is zero, that fraction is discarded. All propositions/sets are in a canonical form. Because we work here only with Bayesian BBA's, the previous fusion formula is in fact rather easy to implement, see [18] (Vol. 2, Chap. 4).

In summary, the target features extraction in a sequence of K images allows us to generate, after Step 3, a set of BBA's $\{m^{Image_k}(.), k = 1, 2, \ldots, K\}$. Every BBA $m^{Image_k}(.)$ is obtained by the PCR5/6 fusion of BBA's $m_1^{Image_k}(.)$ and $m_2^{Image_k}(.)$ built from the outputs of two PNN's. From this combined BBA, a local[18] decision O_k can be drawn about the target type and target pose in $Image_k$ by taking the focal element of $m^{Image_k}(.)$ having the maximum mass of belief.

D. Step 4: Hidden Markov Model (HMM) for recognition

Usually (and specially in military context), the posture of an aircraft can continuously change a lot during its flightpath making target recognition based only on single observation (image) very difficult, because some ambiguities can occur between extracted features with those stored in the training image data set. To improve the target recognition performance and robustness, one proposes to use the sequence of target recognition decision O_k drawn from BBA's $\{m^{Image_k}(.), k = 1, 2, \ldots, K\}$ to feed HMM classifiers in parallel. We suggest this approach because the use of HMM has already been proved to be very efficient in speech recognition, natural language and face recognition. We briefly present HMM, and then we will explain how HMMs are used for automatic aircraft recognition.

Let us consider a dynamical system with a finite set of possible states $S = \{s_1, s_2, \ldots, s_N\}$. The state transitions of the system is modeled by a first order Markov chain governed by the transition probabilities given by $P(s(t_k) = s_j | s(t_{k-1}) = s_i, s(t_{k-2}) = s_k, \ldots) = P(s(t_k) = s_j | s(t_{k-1}) = s_i) = a_{ij}$, where $s(t_k)$ is the random state of the system at time t_k. A HMM is a doubly stochastic processes including an underlying stochastic process (i.e. a Markov chain for modeling the state transitions of the system), and a second stochastic process for modeling the observation of the system (which is a function of the random states of the system). A HMM, denoted $\lambda = (\mathbf{A}, \mathbf{B}, \Pi)$, is fully characterized by the knowledge of the following parameters

1) The number N of possible states $S = \{s_1, s_2, \ldots, s_N\}$ of the Markov chain.
2) The state transition probability matrix[19] $\mathbf{A} = [a_{ij}]$ of size $N \times N$, where $a_{ij} \triangleq P(s(t_k) = s_i | s(t_{k-1}) = s_j)$.
3) The prior mass function (pmf) Π of the initial state of the chain, that is $\Pi = \{\pi_1, \ldots, \pi_N\}$ with $\sum_{i=1}^N \pi_i = 1$, where $\pi_i = P(s(t_1) = s_i)$.
4) The number M of possible values $V = \{v_1, \ldots, v_M\}$ taken by the observation of the system.
5) The conditional pmfs of observed values given the states of the system characterized by the matrix $\mathbf{B} = [b_{mi}]$ of size $M \times N$, with $b_{mi} \triangleq P(O_k = v_m | s(t_k) = s_i)$, where O_k is the observation of the system (i.e. the local decision on target type with its pose) at time t_k.

In this work we consider a set of N_c HMMs in parallel, where each HMM is associated with a given type of target to recognize. We consider the following state and observation models in our HMMs:

- **State model:** For a given type of aircraft, we consider a finite set of distinct aircraft postures available in our training image dataset. In our application, we consider only three states corresponding to s_1 = *top view*, s_2 = *side view* and s_3 = *front view* as shown (for a particular aircraft) in Figure 2.

Fig. 2: Example of HMM states.

- **Observation model:** In our HMMs, we assume that each state (posture) of aircraft is observable. Since we have only $N_p = 3$ states $S = \{s_1, s_2, s_3\}$ for each aircraft, and we have $N_c = 7$ types of aircrafts in the training dataset, we have to deal with $N_{cp} = 3 \times 7 = 21$ possible[20] observations (local decisions) at each time t_k. As explained previously, at the end of Step 3 we have a set of BBA's $\{m^{Image_k}(.), k = 1, 2, ..., K\}$ that helps to draw the sequence of local decisions $O^K \triangleq \{O_1, \ldots, O_k, \ldots, O_K\}$. This sequence of decisions (called also *recognition observations*) is used to evaluate the likelihood $P(O^K|\lambda_i)$ of the different HMMs described by the parameter $\lambda_i = (\mathbf{A}_i, \mathbf{B}_i, \Pi_i)$, $i = 1, 2, \ldots, N_c$. The computation of these likelihoods will be detailed at the end of this section. The final decision for ATR consists to infer the true target type based on the maximum likelihood criterion. More precisely, one will decide that the target type is $i^\star$ if $i^\star = \arg\max_i P(O^K|\lambda_i)$.

• Estimation of HMM parameters

To make recognition with HMMs, we need at first to define a HMM for each type of target one wants to recognize. More precisely, we need to estimate the parameters $\lambda_i =$

[17]Here we assume that Shafers' model holds. The notation $m_{PCR5/6}$ means PCR5 and PCR6 are equivalent when combining two BBA's.

[18]because it is based only on a single image of the unknown observed target in the sequence under analysis.

[19]We assume that the transition matrix is known and time-invariant, i.e. all elements a_{ij} do not depend on t_{k-1} and t_k.

[20]We assume that the unknown observed target type belongs to the set of types of the dataset, as well as its pose.

$(\mathbf{A}_i, \mathbf{B}_i, \Pi_i)$, where $i = 1, \ldots, N_c$ is the target type in the training dataset. The estimation of HMM parameters is done from observation sequences drawn from the training dataset with Baum-Welch algorithm [20] that must be initialized with a chosen value $\lambda_i^0 = (\mathbf{A}_i^0, \mathbf{B}_i^0, \Pi_i^0)$. This initial value is chosen as follows:

1) – State prior probabilities Π_i^0 for a target of type i: For each HMM, we consider only three distinct postures (states) s_1, s_2 and s_3 for the aircraft. We use a uniform prior probability mass distribution for all types of targets. Therefore, we take $\Pi_i^0 = [1/3, 1/3, 1/3]$ for any target type $i = 1, \ldots, N_c$ to recognize.

2) – State transition matrix $\mathbf{A}_i^0$ of a target of type i: The components a_{pq} of the state transition matrix $\mathbf{A}_i^0$ are estimated from the analysis of many sequences[21] of target i as follows

$$a_{pq} = \frac{\sum_{k=1}^{K-1} \delta(s(t_k), s_p) \times \delta(s(t_{k+1}), s_q)}{\sum_{k=1}^{K-1} \delta(s(t_k), s_p)} \quad (4)$$

where N_p is the number of states of the Markov chain, $\delta(x, y)$ is the Kronecker delta function defined by $\delta(x, y) = 1$ if $y = x$, and $\delta(x, y) = 0$ otherwise, and where K is the number of images in the sequence of target i available in the training phase. For example, if in the training phase and for a target of type $i = 1$, we have the following sequence of (target type, pose) cases given by $[(1,1),(1,1),(1,2),(1,1),(1,3),(1,1),(1,1)]$, then from Eq. (4) with $K = 7$, we get[22]

$$\mathbf{A}_{i=1}^0 = \begin{bmatrix} 2/4 & 1/4 & 1/4 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

3) – Observation matrix $\mathbf{B}_i^0$ for a target of type i: The initial observation matrix $\mathbf{B}_i^0$ is given by the confusion matrix learnt from all images of the training dataset. More precisely, from every image of the training dataset, we extract Hu's features and partial SVD outline features and we feed each PNN to get two BBA's according to Steps 1-3. From the combined BBA, we make the local decision $(target_i, pose_j)$ if $m((target_i, pose_j))$ is bigger than all other masses of belief of the BBA. This procedure is applied to all images in the training dataset. By doing so, we can estimate empirically the probabilities to decide $(target_i, pose_j)$ when real case $(target_{i'}, pose_{j'})$ occurs. So we have an estimation of all components of the global confusion matrix $\mathbf{B}^0 = [P(decision = (target_i, pose_j) \mid reality = (target_{i'}, pose_{j'}))]$. From $\mathbf{B}^0$ we extract the c sub-matrices (conditional confusion matrices) $\mathbf{B}_i^0$, $i = 1, \ldots, N_c$ by taking all the rows of $\mathbf{B}^0$ corresponding to the target of type i. In our application, one has $N_c = 7$ types and $N_p = 3$ postures (states) for each target type, hence one has $N_{cp} = 7 \times 3 = 21$ possibles observations. Therefore the global confusion matrix $\mathbf{B}^0$ has size 21×21 is the stack of $N_c = 7$ sub-matrices $\mathbf{B}_i^0$, $i = 1, ..., N_c$, each of size $N_p \times N_{cp} = 3 \times 21$.

[21]The video stream of different (known) aircraft flights generate the sequences of images to estimate approximately a_{pq}

[22]One verifies that the probabilities of each raw of this matrix sum to 1.

• Exploitation of HMM for ATR

Given a sequence O^K of K local decisions drawn from the sequence of K images, and given N_c HMMs characterized by their parameter λ_i $(i = 1, \ldots, N_c)$, one has to compute all the likelihoods $P(O^K|\lambda_i)$, and then infer from them the true target type based on the maximum likelihood criterion which is done by deciding the target type $i^\star$ if $i^\star = \arg\max_i P(O^K|\lambda_i)$. The computation of $P(O^K|\lambda_i)$ is done as follows [20]:

- generation of all possible state sequences of length K, $S_l^K = [s_l(t_1)s_l(t_2)\ldots s_l(t_K)]$, where $s_l(t_k) \in S$ (k=1,..., K) and $l = 1, 2, \ldots, |S|^K$
- computation of $P(O^K|\lambda_i)$ by applying the total probability theorem as follows [23]

$$P(S_l^K|\lambda_i) = \pi_{s_l(t_1)} \cdot a_{s_l(t_1)s_l(t_2)} \cdot \ldots \cdot a_{s_l(t_{K-1})s_l(t_K)} \quad (5)$$

$$P(O^K|\lambda_i, S_l^K) = b_{s_l(t_1)O_1} \cdot b_{s_l(t_2)O_2} \cdot \ldots \cdot b_{s_l(t_K)O_K} \quad (6)$$

$$P(O^K|\lambda_i) = \sum_{l=1}^{|S|^K} P(O^K|\lambda_i, S_l^K) P(S_l^K|\lambda_i) \quad (7)$$

III. Simulations results

For the simulations of SMF-ATR method, we have used $N_c = 7$ types of aircrafts in the training image dataset. Each image of the sequence has 1200×702 pixels. The sequences of aircraft observations in the training dataset take 150 frames. The $N_p = 3$ poses of every aircraft is shown in Fig. 3. For evaluating our approach, we have used sequences (test samples) of images of 7 different aircraft, more precisely the Lockheed-F22, Junkers-G.38ce, Tupolev ANT 20 Maxime Gorky, Caspian Sea Monster (Kaspian Monster), Mirage-F1, Piaggio P180, and Lockheed-Vega, flying under conditions that generate a lot of state (posture) changes in the images. The number of the images in each sequence to test varies from 400 to 500. The shaping parameter of the G-RBF of PNN's has been set to 0.1. The simulation is done in two phases: 1) the training phase (for training PNN's and estimating HMM's parameters), and 2) the exploitation phase for testing the real performances of the SMF-ATR with test sequences.

A - Performances evaluation

In our simulations, we have tested SMF-ATR with two different fusion rules: 1) the PCR5 rule (see Section II-C), and 2) Dempster-Shafer (DS) rule[24] [23]. The percentages of successful recognition (i.e. the recognition rate R_i) obtained with these two SMF-ATR methods are shown in Table I for each type $i = 1, 2, \ldots, N_c$ of aircraft. The performances of these SMF-ATR versions are globally very good since one is able to recognize with a minimum of 85.2% of success the types of aircraft included in the image sequences under test when using DS-based SMF-ATR, and with a minimum of

[23]The index i of components of $\mathbf{A}_i$ and $\mathbf{B}_i$ matrices has been omitted for notation convenience in the last two formulas.

[24]Because Dempster's rule is one of the basis of Dempster-Shafer Theory, we call prefer to call it Dempster-Shafer rule, or just DS rule. This rule coincides here with Bayesian fusion rule because we combine two Bayesian BBA's and we don't use informative priors.

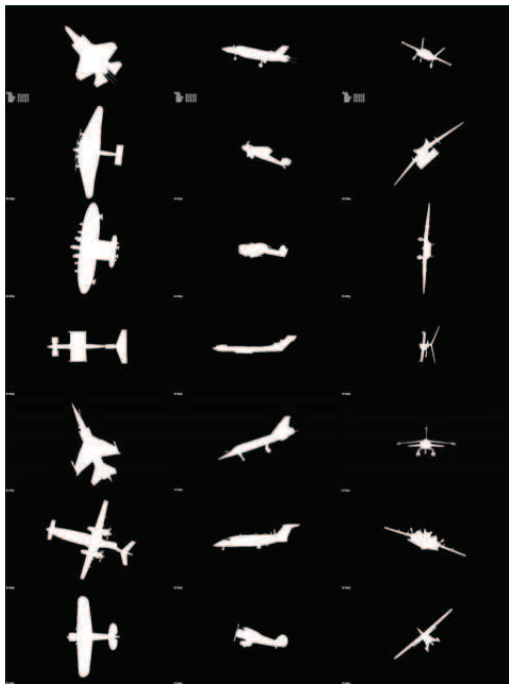

Fig. 3: Poses of different types of aircrafts.

93.5% of success with the PCR5-based SMF-ATR. In term of computational time, it takes between 5ms and 6ms to process each image in the sequence with no particular optimization of our simulation code, which indicates that this SMF-ATR approach is close to meet the requirement for real-time aircraft recognition. It can be observed that PCR5-based SMF-ATR outperforms DS-based SMF-ATR for 3 types of aircraft and gives similar recognition rate as with DS-based SMF-ATR for other types. So PCR5-based SMF-ATR is globally better than DS-based SMF-ATR for our application.

Target type	1	2	3	4	5	6	7
R_i (PCR5 rule)	95.7	93.5	96.3	98.2	96.3	98.5	97.3
R_i (DS rule)	95.7	93.5	85.2	97.8	96.3	98.5	97.2

TABLE I: Aircraft recognition rates R_i (in %).

B - Robustness of SMF-ATR to image scaling

To evaluate the robustness of (PCR5-based) SMF-ATR approach to image scaling effects, we did apply scaling changes (zoom out) of $ZO = 1/2$, $ZO = 1/4$ and $ZO = 1/8$ in the images of the sequences under test. The performances of the SMF-ATR are shown in Table II. One sees that the degradation of recognition performance of SMF-ATR due to scaling effects is very limited since even with a 1/8 zoom out one gets 90% of successful target recognition. The performance will decline sharply if the targets zoom out goes beyond 1/16.

C - Robustness to compound type

Table III gives the performances of SMF-ATR on sequences with two types of targets (475 images with type 1, and 382 images with type 2).

The two left columns of Table III show the performances

Target type	1	2	3	4	5	6	7
R_i (no ZO)	95.7	93.5	96.3	98.2	96.3	98.5	97.3
R_i (ZO=1/2)	95.0	92.0	95.2	94.7	96.1	96.6	95.4
R_i (ZO=1/4)	95.0	92.0	94.7	91.7	93.6	91.6	95.7
R_i (ZO=1/8)	95.0	92.2	93.1	89.3	93.6	94.5	90.7

TABLE II: Aircraft recognition rates R_i (in %) of (PCR5/6-based) SMF-ATR with different zoom out values.

Aircraft	Single Type 1	Single Type 2	Compound Type
R_i (SMF-ATR)	96.3 %	98.5%	97.3%

TABLE III: Robustness to target compound.

obtained when recognizing each type separately in each subsequence. The last column shows the performance when recognizing the compound type Type 1 $\cup$ Type 2. One sees that the performance obtained with compound type (97.3%) is close to the weighted average[25] 97.5% recognition rate. This indicates that no wide range of recognition errors occurs when the targets type change during the recognition process, making SMF-ATR robust to target type switch.

D - Performances with and without HMMs

We have also compared the performances of SMF-ATR, with two methods using more features but which do not exploit sequences of images with HMM. More precisely, the recognition is done locally from the combined BBA for every image without temporal integration processing based on HMM. We call these two Multiple Features Fusion methods MFF1 and MFF2 respectively. In MMF1, one uses Hu's moments, NMI (Normalized Moment of Inertia), affine invariant moments, and SVD of outline, PNN and PCR5 fusion, whereas MMF2 uses same features as MMF1 but with BP network as classifier and DS rule of combination. The recognition performances are shown in Table IV. One sees clearly the advantage to use the image sequence processing with HMMs because of significant improvement of ATR performances. The recognition rate of MFF2 declines seriously because the convergence of the BP network is not good enough.

Target type	1	2	3	4	5	6	7
R_i (SMF-ATR)	95.7	93.5	96.3	98.2	96.3	98.5	97.3
R_i (MFF1)	89.2	92.0	91.2	86.9	92.2	93.5	95.0
R_i (MFF2)	64.9	51.6	82.8	82.2	70.8	48.3	58.9

TABLE IV: Performances (in %) with and without HMMs.

E - SMF-ATR versus SSF-ATR

We have also compared in Table V the performances SMF-ATR with those of two simple SSF-ATR[26] methods, called SSF1-ATR and SSF2-ATR. The SSF1-ATR uses only Hu's moments features whereas SSF2-ATR uses only SVD of outline as features. SSF1-ATR exploits image sequence information using BP networks as classifier and DS rule for combination, while SSF2-ATR uses PNN and PCR5/6 rule.

[25] According to the proportion of the two types in the whole sequence.

[26] SSF-ATR stands for Single-feature Sequence Automatic Target Recognition.

Target type	1	2	3	4	5	6	7
R_i (SMF-ATR)	95.7	93.5	96.3	98.2	96.3	98.5	97.3
R_i (SFF1-ATR)	39.3	42.3	74.3	56.7	60.1	33.9	44.3
R_i (SFF2-ATR)	88.8	66.4	86.7	66.9	73.6	52.9	63.8

TABLE V: Performances (in %) of SMF-ATR and SFF-ATR.

One clearly sees the serious advantage of SMF-ATR with respect to SFF-ATR due to the combination of information drawn from both kinds of features (Hu's and SVD of outline) extracted from the images.

IV. Conclusions and perspectives

A new SMF-ATR approach based on features extraction has been proposed. The extracted features from binary images feed PNNs for building basic belief assignments that are combined with DSmT PCR rule to make a local (based on one image only) decision on target type. The set of local decisions acquired over time for the image sequence feeds HMMs to make the final recognition of the target. The evaluation of this new SMF-ATR approach has been done with realistic sequences of aircraft observations. SMF-ATR is able to achieve higher recognition rates than classical approaches that do not exploit HMMs, or SSF-ATR. Another complementary analysis of the robustness of SMF-ATR to target occultation is currently under progress and will be published in a forthcoming paper. Our very preliminary results based only on few sequences indicate that SMF-ATR seems very robust to target occultations occurring randomly in single (non consecutive) images, but a finer analysis based on Monte-Carlo simulation will be done to evaluate quantitatively its robustness in different conditions (number of consecutive occultations in the sequences, the level of occultation, etc). As interesting perspectives, we want to extend SMF-ATR approach for detecting new target types that are not included in image data set. Also, we would want to deal with the recognition of multiple crossing targets observed in a same image sequence.

Acknowledgment

This work was supported by NNSF of China (No. 60804063, 61175091), and Excellent Young Teacher Foundation sponsored by Qing Lan Project.

References

[1] G. Marsiglia, L. Fortunato, A. Ondini, G. Balzarotti, Template matching techniques for automatic IR target recognition in real and simulated scenarios: tests and evaluations, In Proc. of SPIE: Target Recognition XIII, Vol. 5094, pp. 159–169, 2003.

[2] O.K. Kwon, D.G. Sim, R.H. Park, Robust Hausdorff distance matching algorithms using pyramidal structures, Pattern Reco., Vol. 34 (10), pp. 2005–2013, 2001.

[3] S. Marouani, A. Huertas, G. Medioni, Model-based aircraft recognition in perspective aerial imagery, In Proc. of Int. Symp. on Computer Vision, pp. 371–376, Coral Gables, FL, USA, 1995.

[4] S. Das, B. Bhanu, A system for model-based object recognition in perspective aerial images, Pattern Reco., Vol. 31 (4), pp. 465–491, 1998.

[5] J.F. Gilmore, Knowledge-based target recognition system evolution, Optical Eng., Vol. 30 (5), pp. 557–570, 1991.

[6] S.A. Dudani, K.J. Breeding, R.B. McGhee, Aircraft identification by moment invariants, IEEE Trans. on Comp., Vol. 100, no. 1, 1977.

[7] Y.-J. Du, H.-C. Zhang, Q. Pan, Three-dimensional aircraft recognition using moments, Journal of Data Acquisition and Processing, Vol. 15 (3), pp. 390–394, 2000.

[8] L. Gu, Z.-Q. Zhuang, G.-Y. Zheng, Z.-J. Wang, Algorithm for hand shape matching based on feature fusion, Computer Applications, Vol. 25 (10), pp. 2286–2288, 2005.

[9] C.-Q. Deng, G. Feng, Content-based image re-trieval using combination features, Computer Applications, Vol. 23 (7), pp. 100–102, 2003.

[10] L. Chen, J. Chen, Multi-feature fusion method based on support vector machine and K-nearest neighbor classifier, Computer Applications, Vol. 29 (3), pp. 833–835, 2009.

[11] F.-P. Yang, Z.-X. Bai, Target Recognition Method based on Combination Of BP Neural Network with-Evidence Theory, Fire Control and Command Control, Vol. 31 (10), pp. 88–90, 2006.

[12] X.-D. Li, W.-D.Yang, J. Dezert, An Airplane Image Target's Multi-feature Fusion Recognition Method, Acta Automatica Sinica, Vol. 38 (8), pp. 1298–1307, 2012.

[13] J. Huang, Y. Liang, Y.-M. Cheng, Q. Pan, J.-W.Hu, Automatic target recognition method based on sequential images, J. of Computer Applications, Vol. 27 (1), pp. 87–93, 2006.

[14] J. Hou, Z. Miao, Q. Pan, Intelligent target recognition method of sequential images based on DSmT, J. of Computer Applications, Vol. 26 (1), pp. 120–122, 2006.

[15] S. Chen, J. Feng, Visual pattern recognition by moment invariants, in Proc. of 2010 IEEE Int. Conf. on Inform. and Autom., Harbin, China, pp. 857–860, 2010.

[16] M.K. Hu, Visual pattern recognition by moment invariants, IRE Trans. on Information Theory, Vol. IT-8 (2), pp. 179–187, 1962.

[17] J.-D. Pan, X.-D. Li, An airplane target recognition method based on singular value and PNN, Aero Weaponry, Vol. 1, pp. 45–50, 2013.

[18] F. Smarandache, J. Dezert, Advances and applications of DSmT for information fusion (Collected works), American Research Press, USA Vol.1, 2 & 3, 2004–2009. http://www.onera.fr/staff/jean-dezert?page=2

[19] D.F. Specht, Probabilistic neural networks, Neural networks, Vol. 3, No. 1, pp. 109–118, 1990.

[20] L.R. Rabiner, A tutorial on Hidden Markov Models and selected applications in speech recognition, Proc. of the IEEE, Vol. 77, No. 2, pp. 257–286, 1989. http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/

[21] J.C. Nash, The Singular-Value Decomposition and Its Use to Solve Least-Squares Problems, Ch. 3 in Compact Numerical Methods for Computers: Linear Algebra and Function Minimisation, 2nd ed. Bristol, England: Adam Hilger, pp. 30–48, 1990.

[22] G. Bradski, A. Kaehler, Learning OpenCV: Computer vision with the OpenCV library, O'Reilly, pp. 124–129, 2008.

[23] G. Shafer, A Mathematical theory of evidence, Princeton University Press, Princeton, NJ, U.S.A., 1976.

[24] L. Cao Lin, D.-F. Wang, X.-J. Liu, M.-Y. Zou, Face recognition based on two-dimensional Gabor wavelets, Journal of Electronics & Information Technology, Vol. 28, No. 3, pp. 490–494, 2006.

[25] J. Dezert, A. Tchamova, On the validity of Dempster's fusion rule and its interpretation as a generalization of Bayesian fusion rule, Int. J. of Intel. Syst., Vol. 29, No. 3, Dec. , 2013.

[26] F. Smarandache, J. Dezert, On the consistency of PCR6 with the averaging rule and its application to probability estimation, in Proc. of Fusion 2013, Istanbul, Turkey, July 9-12, 2013.