

Classifier fusion for post-classification of textured images

Hicham Laanaya*[§], Arnaud Martin[§], Driss Aboutajdine* and Ali Khenchaf[§]

*GSCM-LRIT Laboratory

Faculty of sciences of Rabat, Morocco

Email: hicham.laanaya@gmail.com, aboutaj@fsr.ac.ma

[§]ENSIETA-E³I²-EA3876, Brest, France

Email: {laanayhi, Arnaud.Martin, Ali.Khenchaf}@ensieta.fr

Abstract—In this paper, we present various approaches for combining classifiers to improve classification of textured images, which are not generally used in this application framework. This is what we call post-classification step of textured images. Three approaches to combine classifiers are presented: the majority voting approach, belief approach, and classification-based approach. Belief, majority voting and classification-based approaches are compared for classification of real world-data that are sonar images. The obtained results show the interest of this post-classification step, particularly with the belief approach, to improve textured image classification results.

Keywords: Textured images classification, classifier fusion, belief functions.

I. INTRODUCTION

The classification of textured images plays an important role in image processing, pattern recognition and particularly for sonar images classification [7], [9], [10]. This classification is based on the use of supervised classification approaches (requiring an *a priori* knowledge of data classes), and it is based on texture features. The extraction of texture features can not be done at pixel level, because the texture is defined on a set of pixels. Thus the unity used for classification depends on the number of pixels needed to extract these features, for example 8×8 , 16×16 or 32×32 pixels. The precision obtained on the image can be improved using a sliding window with a recovery step, then the class found for this window is assigned to its central part (see Figure 1). This approach provides a more accurate precision and thus gives smoother borders between classes, but it does not take into account any information obtained by the classifications on successive and overlapped areas. Indeed, we have on each part of the image multiple classifiers giving their outputs (classes) that we can combine in order to improve the classification accuracy. We call this step of fusion: post-classification of textured images.

Classification algorithms have their own accuracy which should be investigated. In order to produce thematic image classification it is necessary to perform post-classification algorithms on the result of classification. There are many parameters that tend to make uncertainty in textured image. These parameters arise from sensor system, complexity of the area that is covered by image, geometric and the estimation of the parameters used to characterize the texture. These parameters cause to decrease accuracy of classification

which should be improved in post-classification stage. There are many conventional techniques such as, majority filter, Thomas's filter [11], transition matrix [4], Probability Label Relaxation (PLR) [12] model which are used to improve accuracy of classification results.

The seabed characterization based on sonar image classification is a very difficult task. Indeed, the produced sonar images contains a lot of imperfection coming from the instrumentation and some interference due to the signal traveling on multiple paths (reflection on the bottom or surface), due to speckle, and due to fauna and flora. In order to find the different kind of sediments (characterized by a difference of the texture), we must classify determinate the texture on small tile of size depending on the resolution of the sonar.

We give in section II the principle of the basic approach for classification of textured images, which uses the center of the sliding windows. In section III-B, we recall the principle of different approaches to combine classifiers that we propose to use to combine the classification results of sliding windows. Finally, we present the results of different fusion approaches for sonar images classification.

II. CLASSIFICATION OF TEXTURED IMAGES USING SLIDING WINDOWS

To characterize the texture of an image, it is divided into tiles of size $L \times L$ pixels with a recovery step l ($l < L$) specified by the user in order to obtain an accurate classification. On each tile we calculate texture features using texture analysis methods [9], [10]. These features are used by the classifier. The classification of these tiles gives an image classified on homogeneous areas. To get a better classification (smooth border and better precision) it is necessary to use a l closer to the size of tiles L . Note that the time used for classification increases if l is close to L , since the number of tiles to classify increases with l .

The classification is done on tiles of size $L \times L$ pixels, but in the simplest approach, only the central area, of size $(L - l) \times (L - l)$ pixels is assigned to the class found. Thus, we have a classification resolution of $(L - l) \times (L - l)$ pixels. Figure 1 illustrates this approach.

In order to improve the classification results, we propose to use fusion approaches to combine classification results of

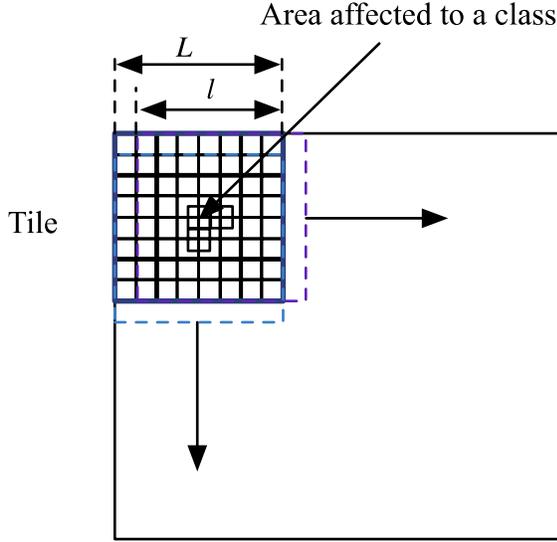


Figure 1. Image classification using a sliding window.

covered areas.

III. FUSION FOR POST-CLASSIFICATION

The approach outlined above is based on the assignment of the class found to the central area of the tile. But this area belong to other neighbors tiles that can be classified into other classes. Our approach is based on the fusion of results for classification of tiles that contain “this central area”.

Let N_c be the number of classes, L the size of used sliding tile, l the recovery step and r the effective size of the studied central area ($r = L - l$). This area z of size $r \times r$ pixels, belongs to a set of tiles $I_i^z, i = 1, \dots, N_z$ where N_z is the number of tiles containing the zone z and depends on the position of z in the image, that is at maximum $(2r + 1)^2$ tiles contain partially the area z and $(2r)^2$ tiles contain completely the area z . Each tile I_i^z belongs to a class $C_q, q = 1, \dots, N_c$.

Figure 2 illustrates this approach by drawing some tiles containing the studied area. In the case of this figure, the studied area is far away from the image borders, in this case the z belongs to a maximum number of tiles unlike the case where the area is close to the edges (less than L pixels between area z and the borders of the image I).

Figure 3 gives the steps of fusion of informations calculated on tiles $I_i^z, i = 1, \dots, N_z$ containing the area z ($I_{i_0}^z$ centered in z): on each tile I_i^z , we calculate information F_i^z , this information can be: (i) the class of the tile I_i^z (fusion by vote (cf. section III-A), (ii) the features extracted from the tile I_i^z and the class of this tile (fusion by classification (cf. section III-B) or (iii) mass functions (belief fusion (cf. section III-C)).

Therefore, it is interesting to use all this information to calculate the class of the area z . So it is an information fusion problem for an optimal decision. We can find in [8] a general description of various approaches to combine classification results.

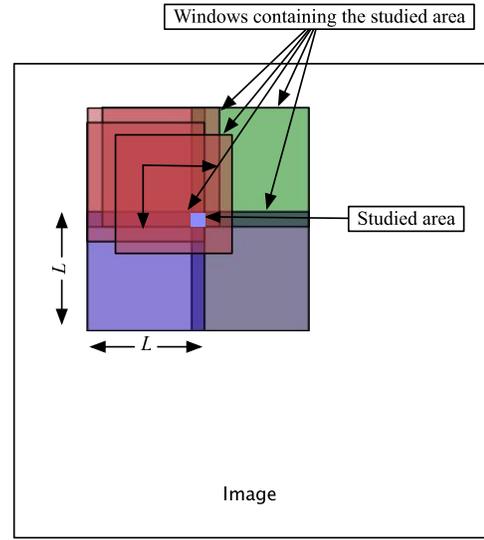


Figure 2. Classification of an image by fusion.

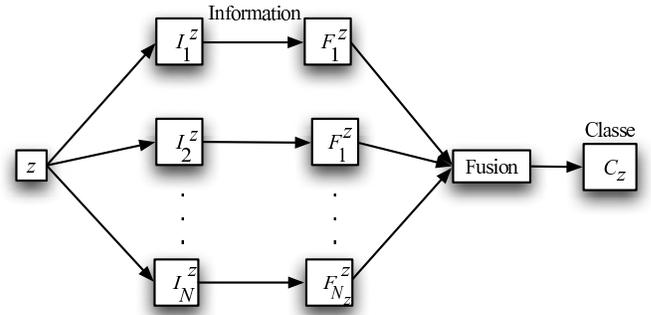


Figure 3. Information fusion for classification of an area z .

A. Fusion by majority voting

The simplest approach for fusion is the majority voting approach based on the combination of classes of classified tiles [6]. The fusion is done using the principle of majority voting by taking the maximum on the number of times that the area z is assigned to a given class. We calculate the normalized histogram p_z of number of times that tiles $I_i^z, i = 1, \dots, N_z$, are classified to a class $C_q, q = 1, \dots, N_c$:

$$p_z(q) = \frac{\text{card}\{i = 1, \dots, N_z ; I_i^z \in C_q\}}{N_z}. \quad (1)$$

Using this approach, the class C_z of area z is the maximum of p_z :

$$C_z = \underset{q=1, \dots, N_c}{\text{argmax}} p_z(q). \quad (2)$$

If the maximum is reached for several values of q , we can choose, for example, the class of the area above z .

Let $S_z = p_z(C_z)$ be the value of the maximum of p_z . Using a sliding window over the image I , we form a matrix of class of areas z noted I_s and a matrix I_c containing the values S_z . This matrix (with a maximum of 1) indicates a kind of “certainty” on the classification of each zone: a value close to

1 indicates that the classifier is “sure” about the class assigned to this zone.

We did not make a difference, in terms of distance between the images containing the area z for the decision of its class. We can use weights of these tiles using a “high” weight for the neighborhood tiles of $I_{i_0}^z$ centered in z and a “low” weight for the other tiles. This weight is a function ψ_ρ decreasing with distance d between the area z and centers of tiles containing this area. For example:

$$\psi_\rho(d) = e^{-\rho d}, \rho \geq 0, \quad (3)$$

Or step function:

$$\psi_\rho(d) = \mathbb{I}_{[0,\rho]}(d). \quad (4)$$

The use of a step function limits the number of used tiles in the decision of the class of area z .

In this case, to find C_z the class of the area z , we use a weighted vote by weighting the value of the histogram p_z by the summation of weights of tiles of the same class (cf. Figure 4) for distances and the associated weight):

$$p_w(q) = p_z(q) \sum_{i \in C_q} \psi_\rho(d_E(G_z, G_i^z)), \quad (5)$$

where G_z and G_i^z are, respectively, centers of tiles $I_{i_0}^z, I_i^z$ and d_E is the Euclidean distance in \mathbb{R}^2 . Of course, other distances could be chosen.

Hence, the class C_z of the area z is given by:

$$C_z = \operatorname{argmax}_{q=1,\dots,N_c} p_w(q). \quad (6)$$

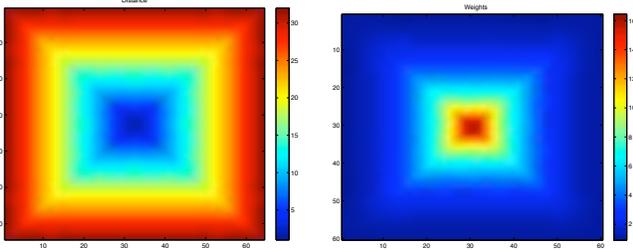


Figure 4. Distances and associated weights ($L = 32, l = 28$ and $\rho = 0.1$ (cf. Equation (3))).

In this case, the “certainty” matrix I_c is defined by $I_c(z) = p_w(C_z)$.

B. Fusion by classification

Suppose that we search the class of an area z of size $r \times r$. We use, for this, the N_z tiles $I_i^z, i = 1, \dots, N_z$ of sizes $L \times L$ pixels containing the area z . Each tile $I_i^z, i = 1, \dots, N_z$ of features x_i^z is classified to a class y_i^z among N_c classes ($x_i^z \in \mathbb{R}^p$ and $y_i^z \in \{1, \dots, N_c\}$). Let $X_z = \{(x_i^z, y_i^z); i = 1, \dots, N_z\}$. Therefore, we can predict the class C_z of the area z using X_z for classifier learning.

We used here a supervised classification for decision because we have an *a priori* knowledge of the initial class (the classes of tiles I_i^z are found from the first classification of the

image). Therefore, the fusion by classification depends on the quality of the first classification. Note that the classifier used for fusion and the one used for initial classification can be different.

There are many classification approaches, for example, the k -nearest neighbors [3], the neural networks, or the support vector machines (SVM) [21]. If we use the k -nearest neighbors, $k_z := k$ must be less than N_z for each area z . Note that the fusion by k -nearest neighbors with $k_z = N_z$ is equivalent to the majority voting approach.

The choice of the classification approach depend on the number of tiles we can use in the learning step. Indeed, if $L - l$ is near L , we can not use easily the approaches such as neural networks and SVM.

In our case, with the sonar images, we consider a classifier based on support vector machines in the experiments part.

C. Belief fusion

The belief function theory have been used before for image segmentation [14], [15] and also in order to combine image classification results [9], [18].

We propose here the use of the belief functions theory for fusion of classification results of sliding windows.

The belief functions theory is based on the use of mass functions. The mass functions are defined over 2^Θ that represent the set of the disjunctions of the frame of discernment $\Theta = \{C_1, \dots, C_{N_c}\}$ and of values in $[0.1]$, where C_q represents the assumption “the observation belongs to the class q ”. Generally, it is added a condition of normalization, given by:

$$\sum_{A \in 2^\Theta} m(A) = 1, \quad (7)$$

where $m(\cdot)$ represents the mass function. The first difficulty is to define these mass functions according to the problem. Other belief functions can be defined from these mass functions, such as the functions of credibility, representing the intensity that all sources believe in an element, and as the functions of plausibility representing the degree with which we believe on an element.

In order to estimate the mass functions to be combined, [1] suggests two models addressing three axioms that involve the use of $N_c \times N_z$ mass functions to the unique focal elements $\{C_q\}, \{C_q^c\}$ and Θ . Furthermore, an axiom guarantees the equivalence with the Bayesian approach where the reality is perfectly known. Both models are substantially equivalent for our data, we use in this article the model given by:

$$\begin{cases} m_{iq}(C_q)(z) = \frac{\alpha_{iq} R_i p(F_i^z/C_q)}{1 + R_i p(F_i^z/C_q)} \\ m_{iq}(C_q^c)(z) = \frac{\alpha_{iq}}{1 + R_i p(F_i^z/C_q)} \\ m_{iq}(\Theta)(z) = 1 - \alpha_{iq} \end{cases} \quad (8)$$

Where p is the probability, $R_i = (\max_{i,q} p(F_i^z/C_q))^{-1}$ is a normalization factor, and $\alpha_{iq} \in [0, 1]$ is a weakening factor

to take into account the reliability of the information provided by the tile $I_i^z : F_i^z$ for a class C_q that we choose equal to 0.95. The difficulty for this model is the estimation of probabilities $p(F_i^z/C_q)$. Where the given F_i^z of the tile I_i^z is the response of a classifier given in the form of class (symbolic data), the estimation of these probabilities can be done by confusion matrices on the learning step.

The combination of $N_c \times N_z$ mass functions can be a real problem if this number is high. Indeed, the first combination rule in the belief function framework proposed by Dempster and Shafer is the normalized conjunctive combination rule given for two basic belief assignments m_1 and m_2 and for all $X \in 2^\Theta$, $X \neq \emptyset$ by:

$$m_{\text{DS}}(X) = \frac{1}{1-k} \sum_{A \cap B = X} m_1(A)m_2(B), \quad (9)$$

where $k = \sum_{A \cap B = \emptyset} m_1(A)m_2(B)$ is the inconsistency of the combination.

Smets [19] proposes to consider an open world, therefore the conjunctive rule is non-normalized and we have for two basic belief assignments m_1 and m_2 and for all $X \in 2^\Theta$ by:

$$m_{\text{Conj}}(X) = \sum_{A \cap B = X} m_1(A)m_2(B). \quad (10)$$

$m_{\text{Conj}}(\emptyset)$ can be interpreted as a non-expected solution. In the Transferable Belief Model of Smets, the repartition of the inconsistency is done in the decision step by the pignistic probability (12).

These two rules (9) and (10) are not idempotent. So the combination of n times the same mass function m , $m_{\text{Conj}}(\emptyset)$ and $m_{\text{DS}}(\emptyset)$ tend to 1 when n tends toward ∞ , that is what we call the auto-conflict (or auto-inconsistency) in [13]. Hence the normalization by $1-k$ in the combination rule (9) or in the pignistic probability can be problematic.

In our application the number $n = N_c \times N_z$ can be high, it is why, we prefer use the rule proposed by [22] defined for two mass functions m_1 and m_2 and for all $X \in 2^\Theta$ by:

$$\begin{cases} m_Y(X) &= m_{\text{Conj}}(X), \\ m_Y(\Theta) &= m_{\text{Conj}}(\Theta) + m_{\text{Conj}}(\emptyset). \end{cases} \quad (11)$$

Many other rules have been proposed, a brief state of the art as well as new rules for managing the conflict in combination is given by [17].

In order to preserve maximum of information, it is preferable to stay on a credal level (*i.e.* to handle belief functions) during the information combination stage to make the decision on the belief functions resulting from this combination. If the decision taken by the maximum of credibility is too pessimistic, the decision obtained by the maximum of plausibility is too optimistic. The maximum of the pignistic probability, introduced by [20], is the most used compromise. The pignistic probability is given for all $X \in 2^\Theta$, with $X \neq \emptyset$ by:

$$\text{betP}(X) = \sum_{Y \in 2^\Theta, Y \neq \emptyset} \frac{|X \cap Y|}{|Y|} s \frac{m(Y)}{1-m(\emptyset)}. \quad (12)$$

The database is composed of 42 sonar images provided by the GESMA (Groupe d'Etudes Sous-Marines de l'Atlantique), which were obtained from a Klein 5400 sonar. These images were labeled using a software developed specifically by specifying the kind of sediment (sand, ripple, sand, rocks, cobbles or shade) (see Figure 5) and the degree of certainty given by the expert (sure, moderately sure or not sure) [7]. We considered three separate classes, which are extremely important to submarine navigation and for seabed mapping. The first class combines rock and cobbles, the second class contains ripples and the third class contains sand and silt.

The measurement unit used for texture extraction and for classification is windows of size 32×32 pixels (about $640 \text{ cm} \times 640 \text{ cm}$).

A. Texture feature extraction

We calculated on these windows six parameters extracted from the cooccurrence matrices approach [9]. The co-occurrence matrices are calculated, by numbering the occurrence of identical gray level of two pixels in a given direction and a given distance.

Let I be an image of dimension $R \times C$, and n_g the number of gray levels. By definition, the co-occurrence matrix p of I in the direction θ and a distance d is $p(i, j, d, \theta) = \{\#(i, j); \text{dist}(i, j) = d, \text{angle}(i, j) = \theta, i, j \in [1, \dots, n_g]\}$.

Co-occurrence matrix is of high dimension (a $n_g \times n_g$ matrix), thus, it is suitable to extract the relevant features. Haralick suggest the calculus of fourteen features [5], not all relevant for a given application. We consider here four directions: 0° , 45° , 90° and 135° . In these four directions, six parameters are calculated and then averaged over the four directions:

- 1) Homogeneity:

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i, j, d, \theta)^2 \quad (13)$$

it has a big value for uniform images and for periodic textured image in the direction θ ,

- 2) Contrast:

$$\frac{1}{n_g - 1} \sum_{k=0}^{n_g-1} k^2 \sum_{i, j=1: |i-j|=k}^{n_g} p(i, j, d, \theta) \quad (14)$$

it characterizes the big transition probabilities between pixels with high differences in gray level,

- 3) Entropy:

$$1 - \sum_{i=1}^R \sum_{j=1}^{n_g} p(i, j, d, \theta) \log(p(i, j, d, \theta)) \quad (15)$$

it represents information present on the image,

- 4) Correlation between image rows and columns:

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \frac{(i - \mu_x)(j - \mu_y)p(i, j, d, \theta)}{\sigma_x \sigma_y} \quad (16)$$

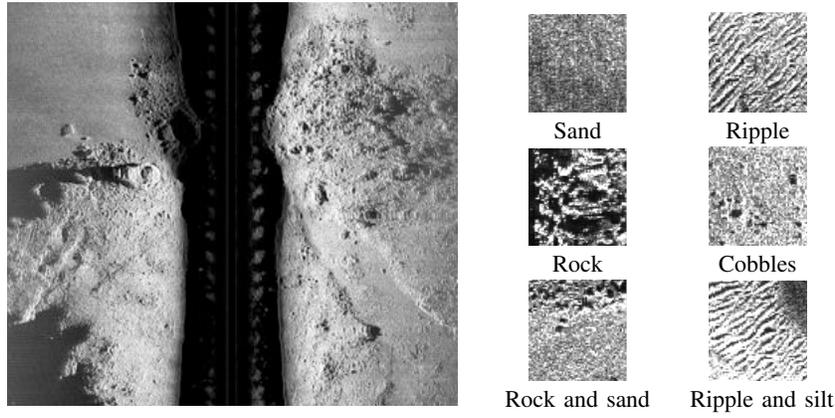


Figure 5. Example of sonar image (provided by GESMA) and labeled small-images

where μ_x and μ_y describe the mean on rows and columns of p respectively, and σ_x and σ_y are the standard deviations.

5) The directivity:

$$\sum_{i=1}^{n_g} p(i, i, d, \theta) \quad (17)$$

texture with high direction privilege in the direction d has a big directivity,

6) Uniformity:

$$\sum_{i=1}^{n_g} p(i, i, d, \theta)^2 \quad (18)$$

it gives the proportion of each gray level in the image.

where $p(i, j, d, \theta)$ is the estimation of the probability of transition of the pixel i to the pixel j in the direction θ with a distance d .

B. Evaluation of the classification

The evaluation of the classification of sonar images is important [10], [16]. The result of image classification can be visually evaluated by comparing it with the ground truth (reference image). But, to evaluate the classification algorithm, we must consider several possible configurations. Typically, classification algorithms are evaluated using confusion matrix. This matrix (CM) is composed by the numbers $CM_{q_1 q_2}$ of elements of the class q_1 which are classified in class q_2 . We can normalize this matrix to get rates that are easy to interpret:

$$CM_{N_{q_1 q_2}} = \frac{CM_{q_1 q_2}}{N_c} = \frac{CM_{q_1 q_2}}{N_{q_1}}, \quad (19)$$

N_c is the number of classes and N_{q_1} is the number of elements of class q_1 . We calculate from this confusion matrix a normalized vector of good classification rates (CR):

$$CR_q = CM_{N_{qq}}, \quad (20)$$

The mean classification rate CR_m , where N represents the total number of images used by the classification algorithm, is given by:

$$CR_m = \frac{\sum_{q=1}^{N_c} CM_{qq}}{N}, \quad (21)$$

And the vector of probability error (PE) is defined by:

$$PE_{q_1} = \frac{1}{2} \left(\sum_{q_2=1, q_2 \neq q_1}^{N_c} CM_{N_{q_1 q_2}} + \sum_{q_2=1, q_2 \neq q_1}^{N_c} \frac{CM_{N_{q_2 q_1}}}{N_c - 1} \right), \quad (22)$$

Hence, we define the weighted mean probability of PE_{q_1} by:

$$PE_m = \frac{\sum_{q=1}^{N_c} N_q PE_q}{N}. \quad (23)$$

We proposed in [10], [16] an approach that takes into account the certainty and the imprecision of the expert: if a tile of class q_1 with certainty, associated with a weight $w \in [0, 1]$, is classified into class q_2 then $CM_{N_{q_1 q_2}}$ will be $CM_{N_{q_1 q_2}} + w$, and if a tile contains more than one class is classified as a class q_1 , then, for $q_2 = 1, \dots, N_c$ $CM_{N_{q_1 q_2}}$ will be $CM_{N_{q_1 q_2}} + N_{q_2}/L^2$, where N_{q_2} is the number of pixels q_2 on the tile.

C. Results

We present below the results obtained using the method of Support Vector Machines for the classification [21]. We used *libSVM* [2] for our tests using a gaussian kernel with $C = 1$, $\gamma = 1/6$ the value of the gaussian kernel parameter (6 is the number of textural parameters). The same classifier is used for classification fusion.

We used 24 sonar images for learning and 18 sonar images for test. We used homogeneous tiles of size 32×32 pixels for learning step. We have 20424 tiles with 2353 tiles of rocks and cobbles, 2583 of ripples and 15488 tiles of sand and silt. The classifier used in the fusion considers a database for learning of size N_z for each area z .

Figure 6 gives a manual segmentation by an expert of a used sonar image. Figure 7 shows the result of classification of a sonar image without fusion and with the three proposed approaches. We note that the post-classification with the three fusion approaches provide less false detection of ripple. Moreover the borders between the different classes are more smooth.

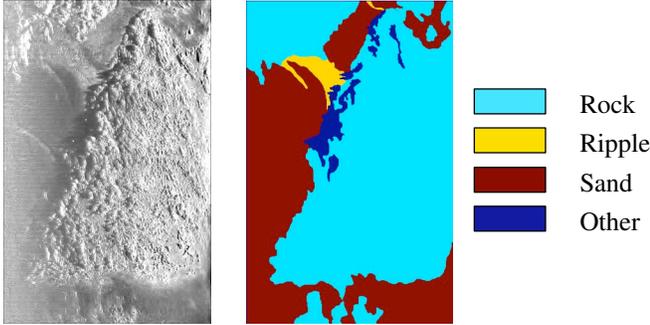


Figure 6. Sonar image and a manual segmentation of this image.

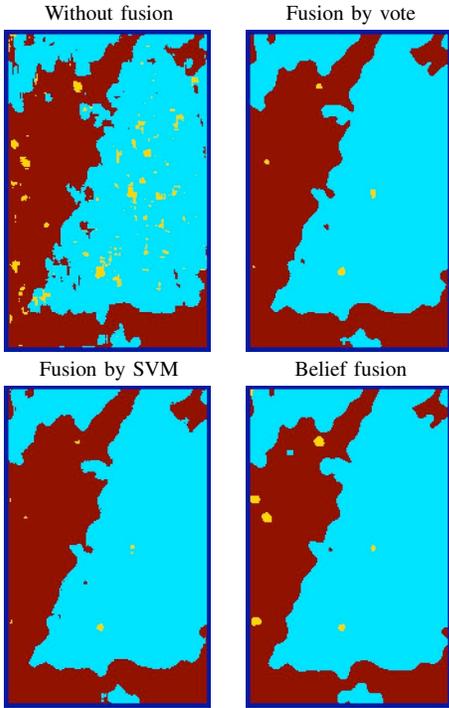


Figure 7. Classification of a sonar image.

As noted in section III-A, we can calculate a certainty matrix for the decision of the class using the majority voting approach. Figure 8 gives certainty matrix for the sonar image presented in Figure 6. We present the same figure on the “ignorance” image for the belief fusion approach: weight on the set Θ .

The confusion matrix, calculated on the basis of learning to

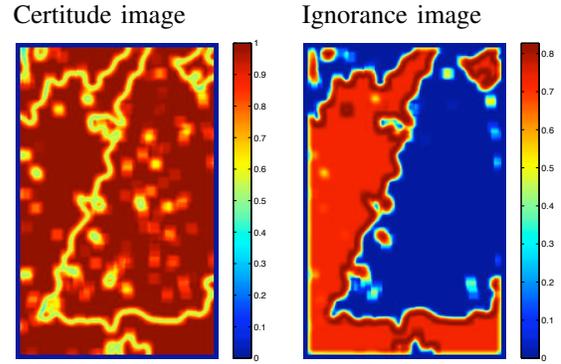


Figure 8. Certitude image for majority voting and ignorance image for belief fusion.

estimate the masses of small-images, is given by:

$$\begin{pmatrix} 1753 & 141 & 459 \\ 221 & 1353 & 1009 \\ 138 & 91 & 15259 \end{pmatrix} \quad (24)$$

We present in Table I the results with and without fusion using the three approaches described above. These results are presented in the form of normalized confusion matrices using the approach presented in [16]. We will adopt the confidence interval at 95% to represent the classification rate and error means. This table, (*cf.* Table I), also gives the CPU time for the three approaches to classification fusion of the image shown in Figure 6. We note that the belief fusion approach is much slower than the majority voting and classification approaches.

Note also the classification rates obtained by fusion approaches are significantly better than the rate found by the first approach without fusion. Indeed, in terms of classification rate, belief approach provides a significantly better rate with $89.22 \pm 0.09\%$. Both approaches, by majority voting and by classification give similar classification rates ($89.03 \pm 0.09\%$). In terms of probabilities of error, the majority voting approach gives the low values with $17.14 \pm 0.09\%$ for belief approach, $18.10 \pm 0.09\%$ for the majority voting approach and $18.11 \pm 0.09\%$ for the classification fusion approach. The classification rate found considering the sliding windows center is $88.11 \pm 0.08\%$ with a probability error of $19.26 \pm 0.09\%$. In terms of classification rate per class, we note that all three approaches were able to improve rates for the three classes, except in the case of a belief combiner for classification of the third class (sand and silt).

V. CONCLUSION

We studied in this paper novel approaches to combine classifiers to improve the classification performance of textured image, a step that we call: post-classification. Thus, the fusion of classification results of sliding windows, uses all the windows containing the area to classify. We studied an approach based on majority voting, an approach using a classifier, and a belief approach providing a significant improvement to results for the classification of sonar images.

	CM _N (%)	CR _m (%)	PE _m (%)	CPU time (s)
Before fusion	$\begin{pmatrix} 68.68 & 4.48 & 26.84 \\ 13.51 & 55.42 & 31.07 \\ 0.71 & 0.43 & 98.86 \end{pmatrix}$	88.11±0.08	19.26±0.09	0
Majority voting	$\begin{pmatrix} 70.99 & 1.42 & 27.60 \\ 8.93 & 57.38 & 33.69 \\ 0.49 & 0.28 & 99.23 \end{pmatrix}$	89.03±0.08	18.10±0.09	11.67
SVM fusion	$\begin{pmatrix} 70.92 & 1.42 & 27.66 \\ 8.82 & 57.39 & 33.79 \\ 0.49 & 0.27 & 99.24 \end{pmatrix}$	89.03±0.09	18.11±0.09	15.37
Belief fusion	$\begin{pmatrix} 74.14 & 1.55 & 24.31 \\ 11.92 & 58.91 & 29.17 \\ 1.04 & 0.57 & 98.39 \end{pmatrix}$	89.22±0.09	17.14±0.09	2599.40

Table I
CLASSIFICATION RESULTS WITH AND WITHOUT FUSION.

The belief approach is time consuming but allows the best performance in the context of this application.

ACKNOWLEDGMENT

We would like homage this work to Patrick Vannoorenberghe, died in January 2007. The principal ideas of this paper were first discussed with Patrick, but he does not let us the time to formalize them.

REFERENCES

- [1] A. Appriou, *Décision et Reconnaissance des formes en signal*. Hermes Science Publication, 2002.
- [2] C.C. Chang, and C.J. Lin, "Libsvm: a library for support vector machines," *Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>*, 2001.
- [3] R.O. Duda, and P. E. Hart, *Pattern Classification and Scene Analysis*. John Wesley and Sons, 1973.
- [4] Van Der Wel, F. J. M, Assessment and visualisation of uncertainty in remote sensing land cover classifications, *Phd. Thesis, Utrecht University*, 2000.
- [5] R. Haralick, "Statistical and textural approaches to textures," in *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786-804, 1979.
- [6] L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.
- [7] H. Laanaya, "Classification en environnement incertain : application à la caractérisation de sédiment marin," in *Ph.D. Thesis of the universities of Britany Occidental, Brest, France, and Mohamed-V Agdal, Rabat, Maroc*, December 2007.
- [8] A. Martin, and E. Radoi, "Effective ATR Algorithms Using Information Fusion Models," in *The Seventh International Conference on Information Fusion, Stockholm, Sweden*, 28 June- 1 July 2004.
- [9] A. Martin, "Comparative study of information fusion methods for sonar images classification," in *The Eighth International Conference on Information Fusion, Philadelphia, USA*, 25-29 July 2005.
- [10] A. Martin, "Fusion for Evaluation of Image Classification in Uncertain Environments," in *The Ninth International Conference on Information Fusion, Florence, Italy*, 10-13 July 2006.
- [11] Mather, P. M, *Computer processing of remotely-sensed images*, John Wiley & Sons, pp. 202-203, 1999.
- [12] Richards, J. A, John, X. Jia, *Remote Sensing Digital Image Analysis*, Springer Berlin Heidelberg, 2006
- [13] C. Osswald, and A. Martin, "Understanding the large family of Dempster-Shafer theory's fusion operators – a decision-based measure," in *The Ninth International Conference on Information Fusion, Florence, Italy*, 10-13 July 2006.
- [14] A.S. Cappelle, O. Colot and M.C. Fernandez-Maloigne, "Evidential segmentation scheme of multi-echo MR images for the detection of brain tumors using neighborhood information," in *Information Fusion*, vol. 5, pp. 203-216, 2004.
- [15] P. Zhang, P. Vannoorenberghe, O. Gallocher, and I. Gardin, "Segmentation d'images par étiquetage crédibiliste : Application à l'imagerie médicale par tomodensitométrie en cancérologie," in *Revue Traitement du Signal*, vol. 23, no. 3-4, pp. 289-305, 2006.
- [16] A. Martin, H. Laanaya, and A. Arnold-Bos, "Evaluation for uncertain image classification and segmentation," in *Pattern Recognition*, vol. 39, 2006.
- [17] A. Martin, and C. Osswald, "Toward a combination rule to deal with partial conflict and specificity in belief functions theory," in *International Conference on Information Fusion, Québec, Canada*, 2007.
- [18] N. Milisavljevic, and I. Bloch, "Sensor fusion in anti-personnel mine detection using a two-level belief function model," in *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 33, no. 2, pp. 269-283, 2003.
- [19] Ph. Smets, "The Combination of Evidence in the Transferable Belief Model," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 5, pp. 447-458, 1990.
- [20] Ph. Smets, "Constructing the pignistic probability function in a context of uncertainty," in *Uncertainty in Artificial Intelligence*, vol. 5, pp. 29-39, 1990.
- [21] V.N. Vapnik, *Statistical Learning Theory*. John Wesley and Sons, 1998.
- [22] R. Yager, "On the Dempster-Shafer Framework and New Combination Rules," in *Informations Sciences*, vol. 41, pp. 93-137, 1987.