

Extraction de Connaissances à partir de Données pour la Classification des Images Sonar

H. Laanaya^{1 2}, A. Martin¹, A. Khenchaf¹ et D. Aboutajdine²

¹ ENSIETA - E³I² EA3876

2, rue François Verny 29806 Brest cedex 9, France

laanayhi, Arnaud.Martin, Ali.Khenchaf@ensieta.fr

² GSCM-LRIT, Faculté des sciences, Université Mohammed V-Agdal, Rabat, Maroc
aboutaj@fsr.ac.ma

Résumé

La classification des images sonar est d'une grande importance dans différentes applications réalistes. C'est le cas pour la navigation sous-marine ou pour la cartographie des fonds marins. La plupart des approches utilisées pour la caractérisation des sédiments marins sont fondées sur l'utilisation des méthodes d'analyse de la texture. En effet, les images sonar présentent différentes zones homogènes de sédiments qu'on peut considérer comme des entités de texture. En général, les paramètres texturaux extraits sont nombreux et ne sont pas tous pertinents, une extraction et/ou réduction de ces paramètres paraît nécessaire avant l'étape de la classification.

Nous présentons dans cet article une chaîne complète de classification des images sonar en essayant d'optimiser les différentes étapes de cette chaîne. Pour l'élaboration de cette chaîne, nous nous fondons sur le processus d'extraction de connaissance à partir de données

L'environnement sous-marin a un caractère incertain, ce qui se reflète sur les images obtenues à partir des capteurs utilisés pour leur élaboration. Il est donc important de développer des méthodes robustes afin de lutter ces imperfections. Dans ce cadre, nous résolvons ce problème de deux façons différentes en utilisant dans un premier temps des méthodes de classification classiques comme les machines à vecteurs de support ou les k -plus proches voisins et dans un deuxième temps des méthodes de classification floues ou crédibilistes. L'approche de la régression par SVM que nous avons introduite permet une modélisation des imperfections des données.

Nous présentons alors les résultats obtenus en utilisant différentes approches pour l'analyse de la texture et pour la classification. Nous utilisons des approches fondées sur les théories de l'incertain pour pallier au problème des imperfections présentes sur les images sonar.

Mots clés : Classification, extraction de paramètres, sélection de paramètres, algorithmes génétiques, caractérisation de sédiments, sonar, texture, SVM flou et crédibiliste.

1 Introduction

Les ondes électromagnétiques et la lumière pénètrent d'une manière faible dans l'eau, contrairement aux ondes acoustiques [27] qui se propagent aisément dans les fonds marins. Ainsi la seule façon d'imager de grandes zones sous-marines est d'utiliser les ondes acoustiques. Une image sonar est obtenue à partir d'une émission acoustique en utilisant un

sonar attaché à l'arrière d'un bateau. Chaque signal émis est réfléchi sur le fond puis reçu sur l'antenne du sonar avec un décalage et une intensité variable. Plusieurs applications de l'imagerie sonar sont répondues, par exemple pour la navigation des robots sous-marins ou pour la cartographie des fonds marins [27].

Les images sonar présentent des zones homogènes de sédiments qu'on peut considérer comme entités de texture (*cf.* figure 1). Nous adoptons dans cet article le processus d'extraction de connaissances à partir de données (ECD) pour la classification des images sonar (*cf.* figure 2) en optimisant si besoin chaque étape de cette chaîne. Ce processus adopté se décompose en cinq étapes :

1. Acquisition des données : Pour une application donnée, elle est faite en utilisant différents types de capteurs. Les données acquises peuvent être de type et de nature différentes et dépendent de ce que l'on recherche.

Une fois les données acquises, on possède ainsi une collection volumineuse d'objets hétérogènes. On peut dès lors passer à la phase d'extraction de paramètres texturaux pour bien caractériser ces objets.

2. Extraction de paramètres de texture : Elle est primordiale dans le succès de la classification. Les données acquises seront transformées pour donner les attributs (avec signification physique) les plus pertinents. La transformation de ces données est particulièrement complexe.

Lorsque l'étape d'extraction de paramètres de texture est terminée, le nombre de paramètres peut être important. Il convient donc de faire une sélection parmi ces paramètres et/ou d'en extraire des nouveaux par combinaison (linéaire ou non-linéaire). Ceci est le rôle de la phase d'extraction-réduction de paramètres.

3. Extraction-réduction de paramètres : Elle permet de réduire la dimensionnalité des données et/ou d'extraire de nouveaux paramètres (sans avoir forcément une signification physique) plus pertinents que les données de départ.

Les méthodes de réduction de dimension sont nombreuses et ont pour objectif de conserver le maximum d'information dans un espace de dimension inférieure. On fera ici la distinction entre les méthodes d'extraction qui créent de nouvelles variables à partir des anciennes et les méthodes de sélection qui cherchent seulement un sous-ensemble d'attributs optimal suivant un critère donné, ici les paramètres ne perdent pas leur signification physique contrairement à la phase d'extraction.

Lorsque l'acquisition, l'extraction de paramètres de texture et la réduction/extraction de paramètres sont achevées, il reste à effectuer une classification sur l'espace des paramètres.

4. Classification : Il y a deux types de méthodes de classification : les méthodes supervisées et les méthodes non supervisées. Si l'on dispose d'un ensemble de données étiquetées, on parle de classification supervisée, dans le cas contraire on parle d'une classification non supervisée.

On distingue deux approches pour la classification supervisée :

- (a) La recherche des frontières et de fonctions discriminantes, par exemple la méthode des machines à vecteurs de support (SVM),
- (b) la recherche d'une partition de données en sous-ensembles, par exemple la méthode des k -plus proches voisins.

Dans le cas supervisée, les classes d'appartenance des données sont connues. La recherche des frontières entre les classes peut être faite par la recherche d'une fonction discriminante, c'est le cas pour la méthode des SVM.

5. Evaluation du système : Une fois la classification effectuée, le système doit être validé et ce en utilisant des méthodes rigoureuses d'évaluation. En général l'évaluation est fondée sur un critère visuel (on utilise une comparaison visuelle des résultats de la classification automatique), cependant une mesure numérique est plus fiable qu'une simple évaluation visuelle.

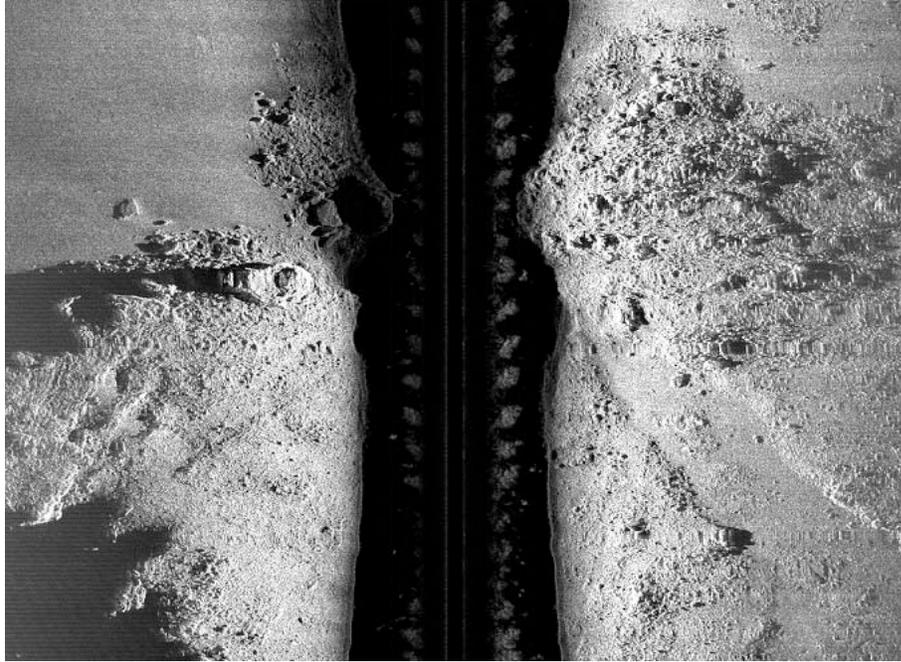


FIG. 1 – Exemple d'une image sonar fournie par le GESMA (Groupes d'Etudes Sous-Marines de l'Atlantique)

Plusieurs études ont été réalisées pour la caractérisation des sédiments marins. Citons les travaux réalisés dans [26, 25] pour le recalage à long terme des images sonar en se fondant sur la classification, et aussi les travaux de [32], [42] et [43] pour la segmentation des images sonar. Dans [46], les travaux sont aussi basés sur le processus ECD pour la classification des images sonar mais avec des classes différentes que celles utilisées dans notre étude. Ils utilisent les cartes auto-organisatrices (*Self Organizing Map* ou SOM, un réseau de neurones non supervisé) pour la caractérisation des images sonar et les matrices de cooccurrence pour l'analyse de la texture.

Le système d'identification développé dans cet article est constitué d'un outil d'analyse de texture des images sonar, d'extraction-réduction de paramètres et d'un classifieur. Le système identifie les sédiments marins comme les roches, les cailloutis, les rides et les vases, en utilisant les paramètres calculés sur des segments d'images sonar (*cf.* figure 1).

Ainsi, le présent article s'organise autour de quatre parties suivies d'une conclusion. Nous présentons dans la première partie l'étape de l'analyse de texture en donnant trois exemples à savoir les matrices de cooccurrence, les ondelettes de Haar et les filtres de Gabor. Nous donnons ensuite dans la seconde partie une description des méthodes d'ex-

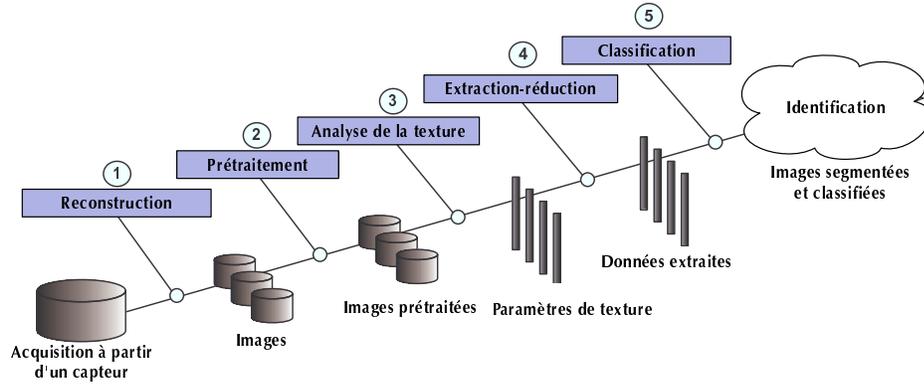


FIG. 2 – ECD pour la classification des images sonar

traction et réduction de paramètres. La troisième partie traite de la classification à partir de données avec des classes dures et floues en donnant le principe des deux méthodes que nous allons utiliser, les SVM et la régression par SVM sur les fonctions d'appartenances et crédibilistes. Quant à la quatrième section, elle présente le principe de la méthode d'évaluation automatique des résultats de la classification utilisée dans cet article. Ces différentes approches (de texture, d'extraction et/ou réduction et de classification) sont comparées en utilisant des méthodes d'évaluation automatique sur des données sonar, dans la dernière section.

2 Analyse de la texture

Il existe plusieurs définitions de la texture, il est ainsi difficile de donner une définition précise. Mais nous pouvons dire que la texture est une zone de l'image qui présente certaines caractéristiques d'homogénéité qui la fait apparaître comme une zone unique.

Le but consiste à extraire d'une image présentant une texture unique, un nombre de paramètres, censés être représentatifs de la texture. Cette texture devra être aussi peu sensible que possible à des transformations de l'image qui laissent la texture inchangée pour un observateur humain (en particulier la translation).

L'étude de la texture a fait l'objet de nombreux travaux de recherche qui ont engendré une multitude de méthodes d'analyse [5], [14]. Ces méthodes ont pour objet de caractériser, décrire, discriminer, segmenter les textures. Nous abordons ici différentes méthodes permettant l'extraction d'attributs texturaux dans les images à niveau de gris. L'accent est mis sur les matrices de cooccurrence, la transformée en ondelettes de Haar et les filtre de Gabor. Nous développons dans les sections suivantes les trois techniques.

2.1 Matrices de cooccurrence

Cette approche est l'une des plus connues et des plus utilisées pour extraire des caractéristiques de texture. Les matrices de cooccurrence estiment des propriétés des images relatives à des statistiques sur des couples de pixels [16].

Une matrice de cooccurrence C_δ est une matrice de taille $N_G \times N_G$, où N_G est le nombre de niveaux de gris de l'image. Pour une direction θ et un déplacement δ donnés,

l'élément (i, j) de la matrice est défini par le nombre de pixels de l'image de niveau de gris j situé à δ pixels d'un pixel de niveau de gris i dans la direction θ .

Les matrices de cooccurrence sont de grande dimension (elles sont de taille 256x256 pour une image codée sur 256 niveaux de gris) et sont donc difficilement exploitables directement.

Afin d'extraire des attributs texturaux de ces matrices, 14 paramètres ont été définis par Haralick [14]. Les six paramètres les plus utilisés [16] et que nous employons dans cette étude sont : l'homogénéité, le contraste, l'entropie, la corrélation, l'uniformité et la directivité.

L'homogénéité qui a une valeur élevée pour des images uniformes ou possédant une texture périodique dans la direction δ est donnée par :

$$HM = \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} C_{\delta}^2(i, j) \quad (1)$$

où N_G est le niveau de gris des imagerie.

L'estimation du contraste est donnée par :

$$CT = \frac{1}{N_G - 1} \sum_{k=0}^{N_G-1} k^2 \sum_{i,j=1,|i-j|=k}^{N_G} C_{\delta}(i, j) \quad (2)$$

L'entropie qui a de faibles valeurs s'il y a peu de probabilités de transition élevées dans C_{δ} , est définie par :

$$EN = 1 - \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} C_{\delta}(i, j) \log(C_{\delta}(i, j)) \quad (3)$$

La corrélation entre les lignes et les colonnes de la matrice est donnée par :

$$CR = \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \frac{(i - \mu_x)(j - \mu_y)C_{\delta}(i, j)}{\sigma_x \sigma_y} \quad (4)$$

où $\mu_x, \sigma_x, \mu_y, \sigma_y$ représentent respectivement les moyennes et écart-types des distributions marginales des éléments de la matrice de cooccurrence.

La directivité qui définit l'existence d'une direction privilégiée de la texture est calculée à partir de la trace de C_{δ} par :

$$DR = \sum_{i=1}^{N_G} C_{\delta}(i, i) \quad (5)$$

L'uniformité qui caractérise la proportion d'un même niveau de gris est donnée par :

$$UN = \sum_{i=1}^{N_G} C_{\delta}(i, i)^2 \quad (6)$$

2.2 Transformée en ondelettes

La transformée en ondelettes est une technique en traitement du signal qui permet de décomposer un signal (telle une image) en différentes sous-bandes de fréquences et à des résolutions multiples [28]. Pour chaque niveau de décomposition, les sous-bandes de hautes

fréquences capturent les discontinuités du signal. Les sous-bandes de basses fréquences sont des sous-images de l'image originale, avec des propriétés statistiques et spatiales similaires à celles du signal original. Comme résultat, les sous-bandes de basses fréquences peuvent être à leur tour décomposées à des niveaux de résolution hauts. Typiquement, la transformée en ondelettes est représentée par deux filtres passe-bas et passe-haut avec plusieurs fonctions de base d'ondelettes disponibles dans la littérature [6].

La théorie des paquets d'ondelettes, qui est une variante de la transformée en ondelettes, a été motivée par la spécificité des signaux « naturels ». La décomposition par paquets d'ondelettes se réalise comme dans la formulation de base par une décomposition en ondelettes mais s'effectue, dans le cas d'une image par exemple, à la fois sur le signal d'approximation et sur le signal de détail. Ce qui conduit à un arbre structuré (cf. figure 3) donnant le signal image sur les différents canaux de fréquences A(LL), B(HL), C(LH), D(HH).

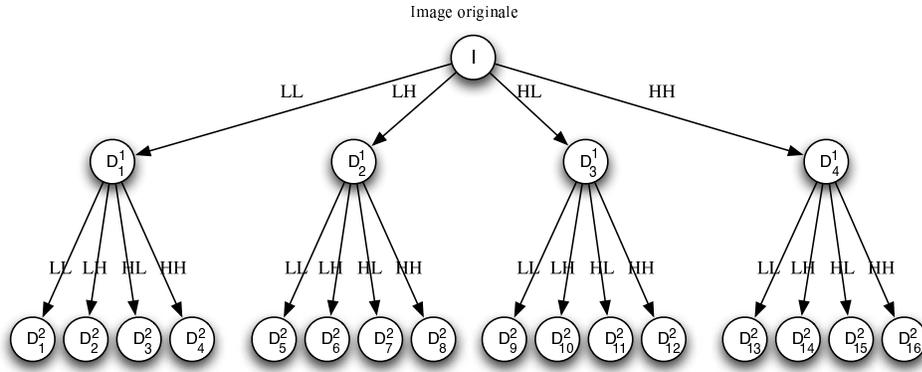


FIG. 3 – Décomposition en paquets d'ondelettes

La transformation par ondelettes, pour des images texturées, peut être récapitulée comme suit :

- Une image texturée donnée doit être décomposée, en utilisant la transformée d'ondelette, en 4 sous-images, qui peuvent être à leur tour décomposées en 4 sous-images. Plus précisément, si I est une image texturée de taille $m \times n$, les 4 sous-images obtenues après la transformation par ondelettes sont données par :

$$D_1^1 = \sum_{i=1}^m h[i] \sum_{j=1}^n h(j) I(2m - i, 2n - j), \quad (7)$$

$$D_2^1 = \sum_{i=1}^m h[i] \sum_{j=1}^n g(j) I(2m - i, 2n - j), \quad (8)$$

$$D_3^1 = \sum_{i=1}^m g[i] \sum_{j=1}^n h(j) I(2m - i, 2n - j), \quad (9)$$

$$D_4^1 = \sum_{i=1}^m g[i] \sum_{j=1}^n g(j) I(2m - i, 2n - j). \quad (10)$$

où h et g sont, respectivement, deux filtres passe-haut et passe-bas avec plusieurs fonctions de base d'ondelettes disponible dans la littérature [6].

- sur chaque sous-image sont calculés les paramètres suivants :
 - Energie

$$E_s^q = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (D_s^d(i, j))^2, \quad (11)$$

- Entropie

$$H_s^q = -\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n D_s^d(i, j) \log[D_s^d(m, n)], \quad (12)$$

- Moyenne

$$M_s^q = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n |D_s^d(i, j)|, \quad (13)$$

où D_s^q est le résultat de la décomposition q de l'image I et $s = 1, 2, 3$ ou 4 , l'indice de l'image résultat.

2.3 Filtres de Gabor

Les filtres de Gabor introduit par Gabor [9] ont montré des propriétés de localisation optimale dans les domaines spatial et fréquentiel. Ils sont donc adéquats pour des problèmes de classification de texture.

Dans le domaine spatial, un filtre de Gabor est une sinusoïde autour d'une fréquence (A, B) modulée par une gaussienne $g(x, y)$. C'est donc un filtre particulièrement adapté pour repérer une fréquence particulière en un lieu donné de l'image étudiée. On peut l'écrire [47] :

$$h(x, y) = g(x, y)e^{-j2\pi(Ax+By)} \quad (14)$$

où $g(x, y)$ est une gaussienne bidimensionnelle et définie par :

$$g(x, y) = \frac{1}{2\pi\sigma_g^2} e^{-\frac{x^2+y^2}{2\sigma_g^2}} \quad (15)$$

Dans plusieurs études, $g(x, y)$ est une fonction supposée être symétrique par simplicité. La réponse fréquentielle spatiale $H(a, b)$ du filtre de Gabor est :

$$H(a, b) = G(a - A, b - B) \quad (16)$$

où :

$$G(a, b) = e^{-2\pi^2\sigma_g(a^2+b^2)} \quad (17)$$

Le filtre de Gabor est un filtre passe-bande centré autour de la fréquence (A, B) avec une largeur de bande déterminée par σ_g . L'étendue (*scale*) du filtre est déterminée par la valeur de σ_g .

Plusieurs études ont été faites pour l'extraction de paramètres à partir d'images filtrées par des filtres de Gabor. Une étude a été faite dans [11] pour comparer plusieurs types de paramètres extraits incluant par exemple le paramètre d'énergie, des moments complexes et l'image filtrée elle même.

On peut utiliser un banc de filtres de Gabor pour l'analyse de texture des images. Un banc de filtres de Gabor est un ensemble de filtres, chacun sélectionnant une fréquence et un angle particulier dans l'image. Les filtres de Gabor sont très utilisées dans l'analyse de textures [26, 1].

Comme nous l'avons déjà vu, les filtres de Gabor sont, dans le domaine spatial, des sinusoïdes modulées par une gaussienne. Ils vont donc réagir aux différentes fréquences présentes localement dans les textures. On va donc pouvoir, à partir de ces décompositions, extraire des attributs en vue de la classification. Ces attributs sont calculés à partir de l'écart-type local (noté $stdGab$) des sorties des filtres de Gabor sur 6 angles allant de 0° à 150° avec un pas de 30° et sur 5 échelles [26, 25] :

- la valeur moyenne sur tous les angles et échelles de $stdGab$ (notée $Moy(stdGab)$),
- le maximum de $stdGab$ normalisé par le premier paramètre (noté $MoyAng90(stdGab)$),
- la moyenne de $stdGab$ sur toutes les échelles mais seulement dans la direction d'insonification et normalisée par le premier paramètre (notée $Max/Moy(stdGab)$),
- l'écart-type local de la luminance des images (noté $std(lum)$).

3 Extraction-réduction de paramètres

Il y a deux raisons pour faire de l'extraction de paramètres avant l'étape de la classification : ou bien les paramètres ne sont pas tous pertinents, ou bien il y a une redondance dans ces paramètres. Par conséquent, l'extraction permet de simplifier la représentation des données et aussi la tâche de la classification.

Les méthodes d'extraction de paramètres déterminent un espace approprié (dans un cas linéaire ou non linéaire) à partir des données de départ. Les transformations linéaires comme l'analyse en composantes principales ou l'analyse linéaire discriminante sont largement utilisées. L'analyse en composantes principales (ACP) calcule les plus grandes valeurs propres de la matrice de covariance des données de départ. La transformation des données est définie par $Y = HX$, où X est la matrice de données de départ, Y est matrice des données transformées et H est la matrice de la transformation linéaire où ses colonnes sont les vecteurs propres. Au contraire de l'ACP qui est une méthode non supervisée, elle n'utilise pas l'information de l'appartenance des données dans les classes, l'analyse linéaire discriminante (ALD), utilise cette information pour la projection. L'ALD utilise le critère de Fisher qui cherche les vecteurs propres de $S_w^{-1}S_b$ (le produit de l'inverse de la matrice de covariance inter-classe, S_w , et de la matrice de covariance intra-classe S_b).

Il existe différentes façons pour définir des techniques d'extraction de paramètres non linéaires. Parmi ces méthodes, nous trouvons l'ACP à noyau [38], qui utilise un noyau pour représenter les données dans un espace de grande dimension (éventuellement infini) où on effectue une projection par l'ACP classique. Une autre méthode non linéaire est l'analyse en composantes curvilinéaires (ACC) introduite par Démartines [7] et sa version supervisée que nous avons développée [22, 24]. L'ACC cherche à préserver la topologie locale des données de départ. En d'autres termes, elle préserve les petites distances entre les données de départ. La version supervisée utilise l'information sur la classe des individus ; elle cherche à préserver la topologie locale des individus considérées par classe ; pour deux individus $x_1, x_2 \in \mathbb{R}^M$ de même classe, on cherche $y_1, y_2 \in \mathbb{R}^m$ tels que $d_1(x_1, x_2) = \psi d_2(y_1, y_2)$, où d_1, d_2 sont deux distances définies, respectivement, sur \mathbb{R}^M et \mathbb{R}^m et ψ une constante utilisée pour contrôler le degré de regroupement des individus de chaque classe.

Notons que les paramètres ainsi transformés, en utilisant des méthodes linéaires ou non linéaires pour l'extraction, peuvent donner naissance à des paramètres plus discriminants que les paramètres de départ (qui ont une signification physique), cependant ces paramètres après la transformation n'ont plus de signification physique.

4 Classification et évaluation

Nous présentons dans cette partie la méthode des machines à vecteurs de support (SVM) sur laquelle est fondée notre système de classification, nous donnons aussi, à titre comparatif, le principe des k -plus proches voisins. Ces deux méthodes sont supervisées et n'utilisent que des classes dures (*i.e.* l'appartenance des données est choisie d'une manière unique parmi un nombre fini de classes), contrairement aux méthodes de classification floue ou crédibiliste qui utilisent la notion des classes floues, fondées sur les théories de l'incertain [20, 18].

L'évaluation des résultats de ces méthodes est un problème à résoudre vu que la vérité terrain n'est pas connue avec certitude. Nous présentons à la fin de cette partie la méthode d'évaluation que nous avons utilisé dans cet article [31].

4.1 Machines à vecteurs de support

L'approche des machines à vecteurs de supports, initiées par Vapnik [45], est une méthode de classification binaire. Elle tente de séparer des individus issus de deux classes (notées $+1$ et -1) en cherchant l'hyperplan optimal qui sépare les deux classes en garantissant une marge maximale entre les éléments des deux classes. Intuitivement, cela garantit un bon niveau de généralisation car de nouveaux individus pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être tout de même situés d'un côté ou de l'autre de la frontière. Un autre intérêt est la sélection de vecteurs de support grâce auxquels est déterminé l'hyperplan optimal. Les données utilisés lors de la recherche de l'hyperplan ne sont alors plus utiles et seuls ces vecteurs de support sont utilisés pour classer un nouveau cas. Cela en fait une méthode très rapide. Ainsi, la force des SVM tient à leur simplicité de mise en œuvre face à des problèmes complexes et à des fondements théoriques solides. Dans le cas où les données sont linéairement séparables, on cherche l'hyperplan $y = w \cdot x + b$ qui maximise la marge entre les deux ensembles où $w \cdot x$ est le produit scalaire de w et x . Ainsi w est la solution du problème d'optimisation convexe :

$$\text{Min}_{w,b} \quad \|w\|^2/2 \quad (18)$$

sous les contraintes :

$$y_t(w \cdot x_t + b) - 1 \geq 0 \quad \forall t = 1, \dots, l, \quad (19)$$

où les $x_t \in \mathbb{R}^p$ représentent les N données d'apprentissage, et $y_t \in \{-1, +1\}$ la classe. Ce problème d'optimisation se résout par la méthode de Lagrange.

Dans le cas où les données ne sont pas linéairement séparables, les contraintes (19) sont relâchées par l'introduction de termes positifs ξ_t . Nous cherchons alors à minimiser :

$$J = \frac{1}{2} \|w\|^2 + C \sum_{t=1}^l \xi_t, \quad (20)$$

sous les contraintes données pour tout t :

$$\begin{cases} y_t(w \cdot x_t + b) \geq 1 - \xi_t \\ \xi_t \geq 0 \end{cases} \quad (21)$$

où C est une constante choisie par l'utilisateur. Le problème se résout alors de manière similaire au cas linéairement séparable par la méthode de Lagrange.

Afin de classer un nouvel élément x , il suffit d'étudier la fonction de décision donnée par :

$$f(x) = \text{sign}\left(\sum_{t \in SV} y_t \alpha_t^0 x_t \cdot x - b_0\right), \quad (22)$$

où SV est l'ensemble des vecteurs de support donné par $SV = \{t ; \alpha_t^0 > 0\}$ pour le cas séparable et par $SV = \{t ; 0 < \alpha_t^0 < C\}$ pour le cas non séparable, et $\alpha_t \geq 0$ sont les multiplicateurs de Lagrange.

Dans les cas non linéaires, le principe des SVM est de projeter, par une fonction noyau, les données de départ dans un espace de grande dimension (éventuellement infinie). Ainsi la classification d'un nouvel élément x est donnée par la fonction de décision :

$$f(x) = \text{sign}\left(\sum_{t \in SV} y_t \alpha_t^0 K(x, x_t) - b_0\right) \quad (23)$$

où K est la fonction noyau. Les fonctions noyau les plus utilisées sont soit du type polynomial $K(x, x_t) = (x \cdot x_t + 1)^d$, $d \in \mathbb{N}$, soit du type gaussien $K(x, x_t) = e^{-\gamma \|x - x_t\|^2}$, $\gamma \in \mathbb{R}^+$. Le choix du noyau et l'optimisation des paramètres de celui-ci reste délicat selon l'application.

Il y a différentes approches pour généraliser les SVM pour plusieurs classes : d'une manière directe où on essaye de résoudre un problème d'optimisation général ou bien en combinant les résultats des classifieurs binaires par SVM, c'est le cas pour les deux approches un-contre-un et un-contre-reste. Notons qu'il existe différentes approches fondées sur le un-contre-un [21]. Nous avons utilisé, dans cet article, l'approche du un-contre-un après la comparaison faite par [4].

4.2 Les k -plus proches voisins

Soit X un ensemble de l individus répartis en N classes notées C_q ($q = 1, \dots, N_c$). Chaque individu $x \in X$ est caractérisé par un vecteur d'attributs à p composantes (x_1, \dots, x_p) . Ainsi, il sera représenté comme un point de \mathbb{R}^p .

Soit x_0 un nouvel individu. Le problème de classification consiste à affecter l'individu x_0 à une classe parmi les N classes (x_0 sera affecté à la classe la plus proche).

Pour les k -plus proches voisins, on choisit tout d'abord une distance dans \mathbb{R}^p , on choisit ensuite les k -plus proches voisins de x_0 à l'aide de cette distance. Parmi ces k individus, on identifie la classe C_{N_0} à laquelle appartient le plus grand nombre d'entre-eux. C'est à cette classe C_{N_0} que l'individu x_0 est enfin affecté.

Cette approche s'applique sur des données avec des classes dures, et peut être généralisée pour des données incertaines et imprécises en se fondant sur les théories de l'incertain. Nous citons par exemple les deux approches des k -plus proches flous [17] et les k -plus proches voisins crédibilistes [8].

4.3 SVM floue et crédibiliste pour la régression

Différentes tentatives ont été proposées pour intégrer le flou dans les SVM. Ainsi l'apprentissage peut être réalisé à partir d'une pondération issue de fonctions d'appartenance [13]. Dans [44], une fonction d'appartenance est introduite pour lever l'ambiguïté dans les zones d'ombre où la classification multi-classes est problématique. Une autre approche proposée dans [2] consiste à modéliser le bruit sur les entrées $x_i, i = 1, \dots, l$ par des Δx_i qui ne dépassent pas en norme un certain $\delta_i, i = 1, \dots, l$. Enfin [15] propose

une régression sur des nombres flous triangulaires qui présente des différences avec notre approche sur lesquelles nous reviendrons plus loin (*cf.* section 4.3.3).

Nous proposons dans cette partie une méthode de classification à partir de données floues ou crédibilistes fondée sur le principe de régression des SVM [40]. En effet les fonctions d'appartenance et les fonctions de croyance possèdent des propriétés similaires qui sont introduites en tant que contraintes dans l'approche d'optimisation.

Les machines à vecteurs de support offrent la possibilité de procéder à une régression linéaire pour non plus prédire une classe, mais une fonction quelconque [45]. Dans le cas de fonctions à valeurs dans \mathbb{R}^N différentes solutions ont été apportées, par exemple dans [12, 41] une régression simple est effectuée sur chacune des dimensions. Dans le cas des fonctions d'appartenance ou des fonctions de croyance la condition de normalisation impose de considérer chacune des dimensions de la fonction à prédire conjointement et de manière indépendante [34, 37]. Les contraintes classiques que l'on peut considérer identiques des fonctions d'appartenance et des fonctions de croyances que nous décrivons ci-dessous (à valeurs dans $[0, 1]$ et dont la somme est 1), nous permettent de réécrire la régression linéaire multiple tout en généralisant les travaux de [34, 37], comme nous le montrons ci-dessous.

Hong [15] propose une régression sur des nombres flous triangulaires, il se place ainsi dans le cas d'une régression multiple en dimension 3, uniquement, mais la différence fondamentale avec notre approche vient des contraintes sur les fonctions d'appartenance et les fonctions de croyance décrites dans les sections 4.3.1 et 4.3.2.

Nous avons situé cette approche dans la littérature [20]. Ainsi elle est novatrice par la prise en compte des contraintes similaires de normalisation des fonctions de croyance et d'appartenance dans le problème de régression multiple. De plus nous proposons ici d'employer une résolution du problème d'optimisation pouvant gérer de grandes quantités de données. Cette approche proposée fait l'objet de la section 4.3.3.

4.3.1 Fonctions d'appartenance

Les fonctions d'appartenance permettent de décrire une appartenance floue à une classe. Ainsi l'appartenance d'une observation x à une classe C_i parmi N_c classes, est donnée par une fonction $\mu_i(x)$ telle que :

$$\begin{cases} \mu_i(x) \in [0, 1] \\ \sum_{i=1}^n \mu_i(x) = 1. \end{cases} \quad (24)$$

Dans ce cas, nous considérons les classes floues. Dans le cas de classes nettes, il est possible de considérer les distributions de possibilité. Ces fonctions à valeurs dans $[0, 1]$ ont une contrainte de normalisation différente de la somme précédente, nous ne les étudions pas ici [49].

4.3.2 Fonctions de croyance

La théorie des fonctions de croyance est fondée sur la manipulation des fonctions de masse. Les fonctions de masse sont définies sur l'ensemble de toutes les disjonctions du cadre de discernement $\Theta = \{C_1, \dots, C_{N_c}\}$ et à valeurs dans $[0, 1]$, où C_i représente l'hypothèse "l'observation appartient à la classe i ". Généralement, il est ajouté une condition

de normalité, donnée par :

$$\sum_{A \in 2^\Theta} m_j(A) = 1, \quad (25)$$

où $m(\cdot)$ représente la fonction de masse. La première difficulté est donc de définir ces fonctions de masse selon le problème. A partir de ces fonctions de masse, d'autres fonctions de croyance peuvent être définies, telles que les fonctions de crédibilité, représentant l'intensité que toutes les sources croient en un élément, et telles que les fonctions de plausibilité représentant l'intensité avec laquelle on ne doute pas en un élément.

Afin de conserver un maximum d'informations, il est préférable de rester à un niveau crédal (*i.e.* de manipuler des fonctions de croyance) pendant l'étape de combinaison des informations pour prendre la décision sur les fonctions de croyance issues de la combinaison. Si la décision prise par le maximum de crédibilité peut être trop pessimiste, la décision issue du maximum de plausibilité est bien souvent trop optimiste. Le maximum de la probabilité pignistique, introduite par [39], reste le compromis le plus employé. La probabilité pignistique est donnée pour tout $X \in 2^\Theta$, avec $X \neq \emptyset$ par :

$$\text{betP}(X) = \sum_{Y \in 2^\Theta, Y \neq \emptyset} \frac{|X \cap Y|}{|Y|} \frac{m(Y)}{1 - m(\emptyset)}. \quad (26)$$

4.3.3 Principe du SVM flou et crédibiliste pour la régression

Soient les vecteurs d'apprentissage $x_t \in \mathbb{R}^p$ et les fonctions associées $y_t \in \mathbb{R}^N$, où $N = N_c$ le nombre de classes dans le cas des fonctions d'appartenance et $N = 2^{N_c}$ dans le cas des fonctions de masse. Ainsi par la régression multiple linéaire, nous cherchons une fonctionnelle $f = (f_1, \dots, f_N)$ où les f_n sont linéaires, de forme $f_n(x) = w_n \cdot x + b_n$. Nous cherchons à déterminer cette fonctionnelle telle que pour les (x_t, y_t) de la base d'apprentissage $|y_{tn} - w_n \cdot x_t + b_n|$ ne dépassent pas un certain ϵ fixé pour tout n . Nous supposons ainsi que tous les points sont à l'intérieur du cylindre défini par ϵ . Afin de généraliser, nous associons un facteur C pour les points qui sont à l'extérieur du cylindre défini par ϵ . Le problème d'optimisation convexe revient donc à celui exposé dans la section 4.1, et le critère à minimiser est :

$$J = \frac{1}{2} \sum_{n=1}^N \|w_n\|^2 + C \sum_{n=1}^N \sum_{t=1}^l (\xi_{tn} + \xi_{tn}^*), \quad (27)$$

sous les contraintes données pour tout t et tout n :

$$\left\{ \begin{array}{l} y_{tn} - w_n \cdot x_t - b_n \leq \epsilon + \xi_{tn}, \\ w_n \cdot x_t + b_n - y_{tn} \leq \epsilon + \xi_{tn}^*, \\ \sum_{n=1}^N (w_n \cdot x_t + b_n) = 1, \\ w_n \cdot x_t + b_n \geq 0, \\ w_n \cdot x_t + b_n \leq 1, \\ \xi_{tn}, \xi_{tn}^* \geq 0. \end{array} \right. \quad (28)$$

Le lagrangien est donc donné par :

$$L = \frac{1}{2} \sum_{n=1}^N \|w_n\|^2 + C \sum_{n=1}^N \sum_{t=1}^l (\xi_{tn} + \xi_{tn}^*) - \sum_{n=1}^N \sum_{t=1}^l (\eta_{tn} \xi_{tn} + \eta_{tn}^* \xi_{tn}^*) \quad (29)$$

$$\begin{aligned}
& - \sum_{n=1}^N \sum_{t=1}^l \alpha_{tn} (\epsilon + \xi_{tn} - y_{tn} + w_n \cdot x_t + b_n) \\
& - \sum_{n=1}^N \sum_{t=1}^l \alpha_{tn}^* (\epsilon + \xi_{tn}^* + y_{tn} - w_n \cdot x_t - b_n) \\
& - \sum_{n=1}^N \sum_{t=1}^l \beta_{tn} (w_n \cdot x_t + b_n) - \sum_{n=1}^N \sum_{t=1}^l \beta_{tn}^* (1 - w_n \cdot x_t - b_n) \\
& - \sum_{t=1}^l \gamma_t \left(1 - \sum_{n=1}^N (w_n \cdot x_t + b_n) \right)
\end{aligned}$$

où les η , α , β et γ sont les multiplicateurs de Lagrange et sont positifs.

Au point selle du lagrangien L , on a pour tout t et tout n , $\partial L / \partial b_n = 0$, $\partial L / \partial w_n = 0$, $\partial L / \partial \xi_{tn} = 0$ et $\partial L / \partial \xi_{tn}^* = 0$. Ainsi :

$$\left\{ \begin{array}{l} \sum_{t=1}^l \sigma_{tn} = 0, \\ w_n = \sum_{t=1}^l \sigma_{tn} x_t, \\ \eta_{tn} = C - \alpha_{tn}, \\ \eta_{tn}^* = C - \alpha_{tn}^*, \end{array} \right. \quad (30)$$

avec $\sigma_{tn} = \alpha_{tn} - \alpha_{tn}^* + \beta_{tn} - \beta_{tn}^* - \gamma_t$.

En intégrant utilisant l'équation (30) dans le lagrangien (équation (29)), le problème revient à maximiser la quantité :

$$-\frac{1}{2} \sum_{n=1}^N \sum_{t,t'=1}^l \sigma_{tn} \sigma_{t'n} x_t \cdot x_{t'} - \sum_{n=1}^N \sum_{t=1}^l \beta_{tn}^* + \frac{\gamma_t}{N} - \sum_{n=1}^N \sum_{t=1}^l \alpha_{tn}^* (\epsilon + y_{tn}) - \sum_{n=1}^N \sum_{t=1}^l \alpha_{tn} (\epsilon - y_{tn}) \quad (31)$$

sous les contraintes :

$$\left\{ \begin{array}{l} \sum_{t=1}^l \sigma_{tn} = 0, \\ \alpha_{tn} \in [0, C], \\ \alpha_{tn}^* \in [0, C], \\ \beta_{tn} \geq 0, \\ \beta_{tn}^* \geq 0, \\ \gamma_t \geq 0. \end{array} \right. \quad (32)$$

Enfin, pour prédire la $n^{\text{ème}}$ sortie \tilde{y}_n , d'un nouvel élément x , on calcule :

$$\tilde{y}_n = \sum_{t=1}^l \sigma_{tn} x_t \cdot x + b_n, \quad (33)$$

où b_n est déduite des conditions de Kuhn, Karush et Tucker :

$$\begin{cases} \alpha_{tn}(\epsilon + \xi_{tn} - y_{tn} + w_n \cdot x_t + b_n) = 0, \\ \alpha^*_{tn}(\epsilon + \xi^*_{tn} + y_{tn} - w_n \cdot x_t - b_n) = 0, \\ (C - \alpha_{tn})\xi_{tn} = 0, \\ (C - \alpha^*_{tn})\xi^*_{tn} = 0, \\ \beta_{tn}(w_n \cdot x_t + b_n) = 0, \\ \beta^*_{tn}(1 - w_n \cdot x_t - b_n) = 0. \end{cases} \quad (34)$$

Si pour un t_0 , $\alpha_{t_0 n} \in]0, C[$ alors, $\xi_{t_0 n} = 0$, ainsi $b_n = y_{t_0 n} - w_n \cdot x_{t_0} - \epsilon$, un raisonnement identique sur α^* donne $b_n = y_{t_0 n} - w_n \cdot x_{t_0} + \epsilon$.

Si on suppose que la relation entre les x_t et les sorties \tilde{y}_t est non linéaire, nous pouvons représenter les données de départ en utilisant un noyau. Ainsi le produit scalaire entre les données de la base d'apprentissage peut être substitué par un noyau : le produit scalaire $x \cdot x'$ devient $K(x, x')$. Une régression linéaire peut alors s'appliquer dans l'espace de représentation. Pour une observation x , la sortie \tilde{y} se prédit en considérant les N valeurs :

$$\tilde{y}_n = \sum_{t=1}^l \sigma_{tn} K(x, x_t) + b_n. \quad (35)$$

A partir de cette approche de régression sur les fonctions d'appartenance ou les fonctions de croyance, nous obtenons un classifieur en prenant la décision *via* le maximum des fonctions d'appartenance ou le maximum de la probabilité pignistique.

Notons que cette approche est équivalente à l'approche directe de SVM pour la classification avec plusieurs classes si $\forall t = 1, \dots, N, \exists n \in \{1, \dots, N_c\} ; y_{tn} = 1$ et $y_{tj} = 0$, pour $j \neq n$.

La résolution du système d'optimisation de la régression par SVM pour des problèmes de grande dimension nécessite des mémoires de stockage de grande taille. Ainsi, l'application des algorithmes d'optimisation classiques est difficile. Ces limites ont été constatées dans [20]. Une solution est d'utiliser des méthodes d'optimisation itératives, où on essaye de résoudre des sous-problèmes du problème principale [33]. Pour notre problème d'optimisation, nous avons adapté la résolution par SMO (*Sequential Minimal Optimization*) développée par [36] pour les machines à vecteurs de support. Il résout des sous-problèmes de dimension deux d'une manière analytique. Nous pouvons ainsi résoudre des problèmes de grande taille avec une vitesse remarquable.

Le système d'optimisation ainsi défini peut être écrit sous la forme :

$$\begin{cases} \text{Min} & \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ & \mathbf{A} \mathbf{x} = 0 \\ & 0 \leq \mathbf{x} \leq \mathbf{C} \end{cases} \quad (36)$$

La matrice \mathbf{Q} est de grande dimension, les méthodes classiques pour la résolution de tels systèmes sont difficiles vue la difficulté de stocker de telles matrices. Les méthodes de décomposition sont utilisées pour pallier à ce genre de problème [36]. A la différence de la plupart des méthodes d'optimisation où, à chaque itération, le \mathbf{x} est mis à jour, les méthodes de décomposition utilisent seulement un sous ensemble des \mathbf{x} pour chaque itération. Ce sous-ensemble noté B donne un sous-problème à résoudre pour chaque itération. Le cas extrême est celui du SMO où B contient deux éléments.

4.4 Evaluation de la classification

L'évaluation est utilisée pour valider le résultat de la classification. Elle se place généralement après un résultat de segmentation et de classification. L'évaluation des résultats de la classification consiste à confronter ses résultats avec une vérité. Elle peut être « objective » (comme la vérité terrain) ou subjective (dépendant de l'évaluation d'un expert). Un résultat de classification est valide s'il ne remet pas en cause cette vérité, qu'elle soit objective ou subjective.

On distingue les méthodes d'évaluation en deux catégories.

- **Evaluation absolue** : L'évaluation ne dépend que de l'objet à évaluer. Pour le résultat, on utilise celui de la classification.
- **Evaluation relative** : L'évaluation dépend d'autres mesures que celles évaluées (on utilise une vérité de terrain).

La méthode d'évaluation proposée dans cette section, peut être appliquée pour l'évaluation d'un algorithme de classification des données où des étiquettes incertaines et imprécises sont fournies. Nous ne considérons pas ici le problème de l'apprentissage sur les étiquettes incertaines et imprécises que nous avons introduit dans la partie 4.3.3 : la classification peut être faite par ce type d'algorithmes ou par d'autres.

4.4.1 Evaluation classique

Le résultat de classification d'une image peut être évaluée visuellement en le comparant avec la vérité terrain (image de référence). Mais, pour évaluer l'algorithme de classification, nous devons considérer plusieurs configurations possibles. Généralement, les algorithmes de classification sont évalués en utilisant la matrice de confusion. Cette matrice (MC) est composée par les nombres mc_{ij} des éléments de la classe i qui sont classifiés en classe j . Nous pouvons normaliser cette matrice pour obtenir des taux qui sont faciles à interpréter :

$$Nmc_{ij} = \frac{mc_{ij}}{\sum_{k=1}^{N_c} mc_{ik}} = \frac{mc_{ij}}{N_i}, \quad (37)$$

où N_c est le nombre de classes considérées et N_i est le nombre des éléments de la classe i . Nous pouvons calculer à partir de cette matrice de confusion normalisée le vecteur des taux de bonne classification (TC) :

$$TC_i = Nmc_{ii}, \quad (38)$$

un vecteur de probabilités d'erreur de classification (PE) :

$$PE_i = \frac{1}{2} \left(\sum_{j=1, j \neq i}^{N_c} Nmc_{ij} + \sum_{j=1, j \neq i}^N \frac{Nmc_{ji}}{N-1} \right), \quad (39)$$

et la probabilité moyenne pondérée des PE_i :

$$PE_m = \frac{\sum_{i=1}^{N_c} N_i PE_i}{N}. \quad (40)$$

Cette erreur de classification est la moyenne des deux erreurs, l'erreur qui correspond aux éléments de classe i et classifiés en une autre classe (premier terme de (39)) et l'erreur qui correspond aux éléments classifiés en j et qui appartiennent à la classe i (deuxième terme de (39)). Ces erreurs sont appelées pour la première erreur des éléments « faux positifs » (ou erreur du premier type) et erreur des éléments « faux négatifs » (ou erreur du deuxième type).

4.4.2 Evaluation avec connaissance incertaine de la vérité terrain

Nous adoptons ici l’approche que nous avons développé [31]. Supposons que l’information est donnée par un expert sur chaque pixel d’une imagerie de taille $n \times n$. Dans ce cas, cette imagerie peut contenir plus d’une classe et l’algorithme de classification ne donnera qu’une seule classe. Dans ce cas, si $M = (mc_{ij})_{i,j=1,\dots,n}$ représente la matrice de confusion, mc_{ii} sera $mc_{ii} + N_i/n^2$ où N_i est le nombre de pixels i dans l’imagerie, et $mc_{ij,i \neq j} = mc_{ij} + N_j/n^2$, où N_j est le nombre de pixels j de l’imagerie.

Dans le cas où l’expert donne un degré de certitude sur les classes considérées, nous affectons des poids pour chaque degré de certitude (par exemple $2/3$ pour sûr, $1/2$ pour moyennement sûr et $1/3$ pour pas sûr). Ainsi, si on a une imagerie de taille $n \times n$ de classe i avec une certitude moyenne, alors, si l’algorithme de classification trouve la classe i , le terme mc_{ii} sera $mc_{ii} + 1/2$ sinon, si l’algorithme de classification trouve une autre classe j alors $mc_{ij} = mc_{ij} + 1/2$.

5 Expérimentation

Nous appliquons les différentes approches étudiées des différentes phases du processus ECD, en utilisant différentes combinaisons entre les méthodes d’analyse de la texture, d’extraction-réduction de paramètres et de classification. Nous donnons les résultats de la classification des images sonar entières par les deux nouvelles approches floue et crédibilistes des SVR. Ces deux approches sont comparées dans un premier temps avec les SVM et les k -plus proches voisins sur des images sonar puis dans un second temps avec les SVM sur des images sonar entières.

5.1 Base de données

La base de données est constituée de 42 images sonar fournies par le GESMA (Groupe d’Etudes Sous-Marines de l’Atlantique) ayant été obtenues à partir d’un sonar Klein 5400 au large des côtes finistériennes et de résolution de 20 à 30 cm en azimuth et 3 cm en range. La profondeur des fonds se situe entre 15 m et 40 m. Les images ont été constituées à partir des données brutes *i.e.* signaux réverbérés sur le fond et captés par le sonar. Ces images ont été labellisées, par trois experts, à partir d’un logiciel développé spécialement en spécifiant le type du sédiment présent (sable, ride, vase, roche, cailloutis ou ombre) (*cf.* figure 4) et le degré de certitude de l’expert (sûr, moyennement sûr ou non sûr). Parmi ces sédiments, nous avons considéré trois classes distinctes, particulièrement importantes pour la navigation sous-marine et les sédimentologues.

Le tableau 1 dresse l’effectif des sédiments par expert sur la base de données utilisées.

Dans un premier temps sont utilisées des images homogènes issues de la base de données des 42 images sonar. Nous considérons 3 classes choisies de façon à ce que les textures des sédiments qu’elles contiennent se ressemblent : une classe pour le sable et la vase, une classe pour les rides (pour différentes orientations) et une classe pour les roches et les cailloutis. Nous n’avons pas considéré les deux autres classes ombre et autre puisque elles sont en trop faible nombre (*cf.* tableau 1).

Nous avons utilisé, pour les différents tests, 20 tirages aléatoires pour avoir une meilleure estimation des résultats avec 6000 images pour les bases d’apprentissage et 3000 images pour les bases de test. La taille des fenêtres de la base de données est de 32×32 pixels (soit environ 640×640 cm).

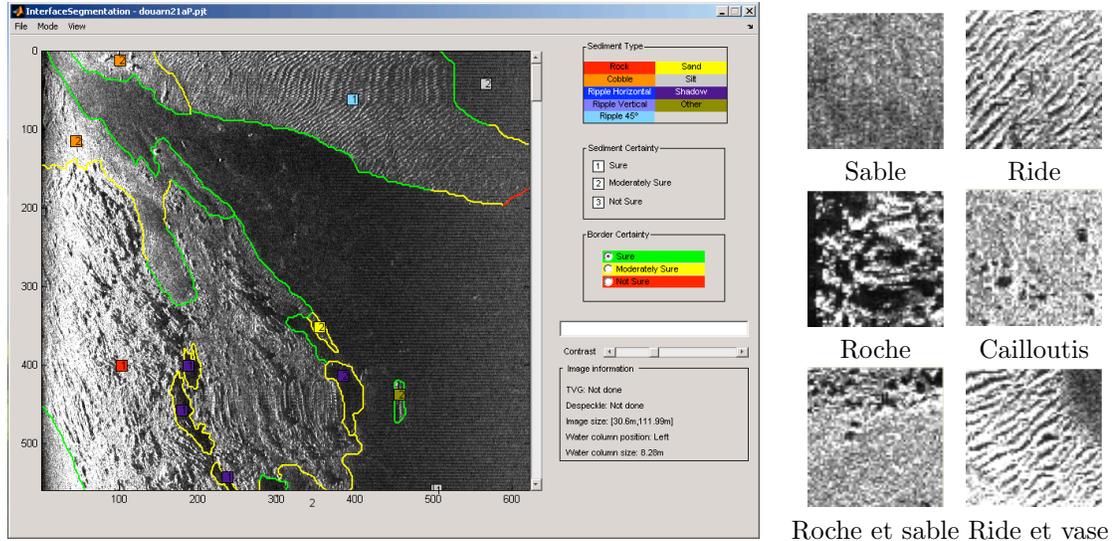


FIG. 4 – Exemple d’image sonar (fournie par le GESMA) et d’imagettes étiquetées.

Imagettes	Expert 1	Expert 2	Expert 3
Roche	3134	4300	4192
Cailloutis	1446	2358	2612
Ride	3569	5835	3540
Sable	8521	11999	9062
Vase	13749	11632	11476
Ombre	44	1449	33
Autre	1	1424	395
Inhomogène	8533	7374	7687

TAB. 1 – Nombre d’imagettes de taille 32×32 de la base de données des trois experts.

La deuxième étape où on classe des images sonar entières est conduite sur la base de données des 42 images sonar du premier expert. Cette base de données est divisée en deux parties : une base de données pour l’apprentissage de 24 images sonar et une deuxième base de données pour le test de 18 images sonar. Nous effectuons 10 tirages aléatoires de ces deux bases de données pour avoir des résultats cohérents et significatifs. Le résultat de classification est la moyenne sur les résultats de classification des 10 tirages utilisés. Nous utilisons seulement les imagettes homogènes des images de la base d’apprentissage pour l’apprentissage du classifieur.

5.2 Modélisation des fonctions floues et crédibilistes

Nous avons utilisé l’approche développée dans [17] pour calculer la fonction d’appartenance des vecteurs d’apprentissage que nous utilisons pour l’apprentissage du SVR flou, et l’approche donnée dans [8] pour estimer les fonctions de masse que nous utilisons pour l’apprentissage du SVR crédibiliste.

L’approche développée dans [17] est celle d’un k -plus proches voisins flou. Les fonctions

d'appartenance d'un vecteur d'apprentissage x_t sont estimées dans un premier temps par :

$$\mu_i(x_t) = \frac{k_i(x_t)}{k_f}, \quad (41)$$

où k_f est le nombre de plus proches voisins choisi pour le voisinage flou V_{K_f} et $k_i(x_t) = |C_i \cap V_{K_f}(x_t)|$. Dans un second temps, nous calculons la fonction d'appartenance pour un vecteur x à classifier :

$$\mu_i(x) = \frac{\sum_{j=1}^k \frac{\mu_i(x_{t_j})}{\|x - x_{t_j}\|^2}}{\sum_{j=1}^k \frac{1}{\|x - x_{t_j}\|^2}}. \quad (42)$$

où $x_{t_j}, j = 1, \dots, k$ sont les k -plus proches vecteurs de x et la norme employée ici est la norme euclidienne.

L'approche proposée dans [8] calcule une estimation des fonctions de masses à partir d'un modèle de distance :

$$\begin{cases} m_k(C_i|x^{(t,k)})(x) = \alpha_i e^{\gamma_i d^2(x, x^{(t,k)})} \\ m_k(\Theta|x^{(t,k)})(x) = 1 - \alpha_i e^{\gamma_i d^2(x, x^{(t,k)})} \end{cases} \quad (43)$$

où C_i est la classe associée à $x^{(t,k)}$, qui sont les k vecteurs d'apprentissage les plus proches de la valeur x et la distance employée est la distance euclidienne. α_i et γ_i sont des coefficients d'affaiblissement, et de normalisation. Les k fonctions de masse ainsi calculées pour chaque x sont combinées par la règle orthogonale normalisée de Dempster-Shafer. Cette règle est donnée pour deux experts et pour tout $A \in 2^\Theta$, $A \neq \emptyset$ par :

$$m(A) = \frac{\sum_{B \cap C = A} m_1(B)m_2(C)}{1 - \sum_{B \cap C = \emptyset} m_1(B)m_2(C)}, \quad (44)$$

et $m(\emptyset) = 0$.

5.3 Résultats

Nous commençons par donner les résultats des expérimentations sur des imassettes homogènes en utilisant 20 tirages aléatoires pour avoir des résultats significatifs. Nous avons utilisé 6000 imassettes pour la base d'apprentissage et 3000 imassettes pour la base de validation. Les résultats sont présentés sous forme de matrices de confusion classiques normalisées. Nous avons utilisé les SVM avec les paramètres par défaut de *libSVM* [3] ($C = 1$ et un noyau gaussien avec $\gamma = 1/l$ où l est le nombre de vecteurs de la base d'apprentissage) comme classifieur de référence pour comparer entre les différentes approches d'analyse de texture.

5.3.1 Comparaison entre les méthodes d'analyse de texture

Nous avons effectué des tests pour les différentes approches d'analyse de texture, avec $\delta = 2$ et $\theta = 0^\circ, 45^\circ, 90^\circ$ et 135° pour les matrices de cooccurrence et un niveau de

décomposition égale à 2 pour les ondelettes. Nous avons utilisé aussi le regroupement des attributs texturaux de ces méthodes. La figure 5 donne les taux de classification obtenus pour les 20 bases de données tirées aléatoirement. Les matrices de confusions moyennes, les taux de classification globaux moyens (*cf.* section 4.4.2) et les probabilités d'erreurs associées sont présentées dans le tableau 2 pour le classifieur de référence SVM.

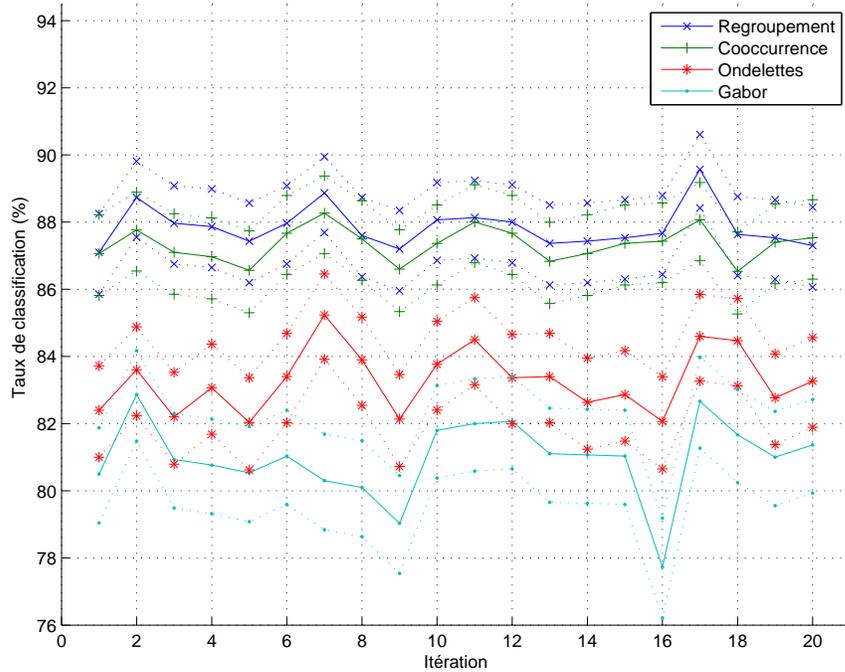


FIG. 5 – Classification par le classifieur de référence SVM sur les différents attributs texturaux.

	MC	PE	TC	PE _m
Cooccurrence	$\begin{pmatrix} 73.92 & 0.69 & 25.39 \\ 23.74 & 35.77 & 40.50 \\ 1.87 & 0.05 & 98.09 \end{pmatrix}$	$\begin{pmatrix} 19.44 \\ 32.30 \\ 17.43 \end{pmatrix}$	87.34 ± 1.24%	23.06±1.47%
Ondelettes	$\begin{pmatrix} 52.04 & 3.85 & 44.12 \\ 11.08 & 17.38 & 71.54 \\ 0.45 & 0.00 & 99.55 \end{pmatrix}$	$\begin{pmatrix} 26.86 \\ 42.27 \\ 29.14 \end{pmatrix}$	83.28 ± 1.38%	32.76±1.66%
Gabor	$\begin{pmatrix} 50.11 & 0.21 & 49.68 \\ 16.84 & 2.00 & 81.16 \\ 0.28 & 0.04 & 99.68 \end{pmatrix}$	$\begin{pmatrix} 29.22 \\ 49.06 \\ 32.87 \end{pmatrix}$	80.98 ± 1.44%	37.05±1.71%
Regroupement	$\begin{pmatrix} 68.37 & 1.76 & 29.87 \\ 17.20 & 34.72 & 48.08 \\ 0.82 & 0.09 & 99.10 \end{pmatrix}$	$\begin{pmatrix} 20.32 \\ 33.10 \\ 19.94 \end{pmatrix}$	87.85 ± 1.22%	24.45±1.50%

TAB. 2 – Performances de classification par le classifieur de référence SVM pour les différentes approches d'analyse de texture.

Ces résultats montrent que les paramètres calculés sur les matrices de cooccurrence sont meilleurs en terme de taux de classification avec une différence significative avec les ondelettes de Haar et les filtres de Gabor. Mais en terme de reconnaissance classe par classe, on remarque que les trois approches arrivent à détecter une partie importante de

la classe des sables et des vases (classe 3). Les matrices de cooccurrence donnent une meilleure détection de la classe des roches et des cailloutis (classe 1) par rapport aux ondelettes de Haar et aux ondelettes de Gabor, mais les trois approches n'arrivent pas à séparer la classe des rides (classe 2) des deux autres classes. Ceci est peut être dû à la non-équirépartition des classes lors de l'apprentissage du classifieur. En effet, la classe des sables et des vases est majoritaire par rapport aux deux autres classes (*cf.* tableau 1), le classifieur a ainsi tendance à avoir une bonne capacité de généralisation pour la classe majoritaire au dépit des deux autres classes.

Le regroupement des différents attributs texturaux améliore le taux de classification dû à la bonne détection de la classe sable, la classe majoritaire de la base de données. Mais, il a dégradé le taux de détection de la classe roche cailloutis (classe 1) et de la classe ride (classe 2) par rapport aux matrices de cooccurrence qui donnent les meilleurs taux. Nous utilisons ainsi pour les tests qui suivent le regroupement des attributs des trois approches d'analyse de texture.

Les attributs extraits après l'analyse de texture peuvent être en grand nombre, contenir des redondances, et ne pas être tous discriminants. Nous en avons 25 à partir des trois méthodes d'analyse de texture vues précédemment sur les images sonar. Ces problèmes peuvent affecter la qualité de la classification et le temps nécessaire pour cette tâche. On peut résoudre ces problèmes de deux façons différentes : ou bien on calcule de nouveaux paramètres à partir des attributs originaux, ou bien on fait une sélection sur ces attributs. La première approche ne garde pas la signification physique des attributs, au contraire de la seconde approche qui fait une sélection sur les attributs originaux.

Nous avons présenté en section 3 les méthodes classiques d'extraction-réduction de paramètres. Nous les appliquons et étudions ici dans le cas des images sonar.

Le tableau 3 donne les matrices de confusion, les probabilités d'erreur associées aux meilleurs taux de classification pour les différentes approches d'extraction-réduction de paramètres et pour les différentes dimensions de projection.

	MC	PE	TC	PE _m	Nb. Par.
ACP	$\begin{pmatrix} 71.97 & 1.88 & 26.15 \\ 17.38 & 42.05 & 40.57 \\ 1.06 & 0.23 & 98.72 \end{pmatrix}$	$\begin{pmatrix} 18.62 \\ 29.50 \\ 17.32 \end{pmatrix}$	88.43 ± 1.19%	21.81±1.44%	8
ALD	$\begin{pmatrix} 66.54 & 4.62 & 28.84 \\ 18.65 & 41.51 & 39.83 \\ 0.87 & 0.22 & 98.91 \end{pmatrix}$	$\begin{pmatrix} 21.61 \\ 30.45 \\ 17.71 \end{pmatrix}$	87.93 ± 1.21%	23.26±1.48%	12
ACC	$\begin{pmatrix} 76.62 & 14.83 & 8.56 \\ 21.56 & 65.17 & 13.27 \\ 4.19 & 4.63 & 91.18 \end{pmatrix}$	$\begin{pmatrix} 22.19 \\ 25.13 \\ 15.48 \end{pmatrix}$	87.27 ± 1.24%	20.93±1.42%	24
ACC-sup	$\begin{pmatrix} 20.24 & 4.33 & 75.43 \\ 18.22 & 3.28 & 78.50 \\ 17.06 & 6.17 & 76.78 \end{pmatrix}$	$\begin{pmatrix} 48.70 \\ 50.98 \\ 50.09 \end{pmatrix}$	58.76 ± 1.77%	49.92±1.57%	2

TAB. 3 – Extraction de paramètres après regroupement.

Rappelons que nous avons obtenu un taux de classification de 87.85±1.22% avec les SVM sans extraction de paramètres avec des probabilités d'erreur [20.35 33.10 20.06] (*cf.* tableau 2 p19). Les taux trouvés avec l'ACP et l'ALD sont supérieurs à celui trouvé sans extraction de paramètres. Par contre l'ACC a donné un taux inférieur et l'ACC supervisée donne un taux de classification faible. En effet, nous avons obtenu un taux de classification de 88.43±1.19% en utilisant l'ACP avec seulement 8 paramètres, 87.93±1.21% en

utilisant l'ALD avec 12 paramètres, $87.27 \pm 1.24\%$ en utilisant l'ACC et enfin, avec l'ACC supervisée, nous avons obtenu un taux de $58.76 \pm 1.77\%$ avec 2 paramètres. Nous remarquons toujours la faible détection de la 2^{ème} classe (classe des rides) et la bonne détection de la 3^{ème} classe (classe du sable et de la vase). L'ACC donne une meilleure détection, par rapport à l'ACP et à l'ALD, des deux premières classes ; les classes minoritaires de la base de données. Nous remarquons pour l'ACC supervisée qu'une grande partie de la première classe (la classe des roches et des cailloutis) et de la deuxième classe (la classe des rides) est classé en sable et vase, qui correspond à la classe majoritaire de cette étude.

Les paramètres extraits après les deux étapes de l'analyse de texture et l'extraction-réduction de paramètres sont utilisés par des méthodes de classification pour étiqueter les imageries associées à ces paramètres. Nous utilisons par la suite le regroupement des attributs texturaux des différentes méthodes d'analyse de texture étudiées puisque elles donnent les meilleurs taux de classification avec les SVM. Nous n'utilisons pas d'extraction-réduction de paramètres puisque les approches étudiées n'améliorent pas d'une manière significative les résultats de la classification.

5.3.2 Comparaison entre les classifieurs

Le regroupement des paramètres calculés sur les images sonar sont utilisés comme entrées des classifieurs.

Nous présentons ici une comparaison entre les quatre classifieurs considérés : Les SVM, les k -plus proches voisins et les deux nouvelles approches, les SVR flous (f -SVR) et les SVR crédibilistes (b -SVR). Nous utilisons ces classifieurs avec des paramètres que nous avons optimisé par les algorithmes génétiques [19] pour les SVM, les SVR flous et crédibilistes et avec une recherche directe pour les k -plus proche voisins.

Nous donnons dans cette partie une comparaison entre les quatre classifieurs (k -plus proches voisins, SVM, b -SVR et f -SVR) avec leurs meilleurs paramètres respectifs :

1. pour les SVM avec un noyau gaussien et $C = 2^{13.37}$, $\gamma = 2^{-3.6563}$ pour le noyau gaussien.
2. pour les k -plus proches voisins : $k = 15$.
3. pour les SVR flous avec un noyau gaussien et $C_f = 27864.85$, $\gamma_f = 0.123493$, $\epsilon_f = 0.000008$.
4. pour les SVR crédibilistes avec un noyau gaussien et $C_b = 20188.34$, $\gamma_b = 0.088573$ et $\epsilon_b = 0.000083$

Notons que nous avons utilisé une version modifiée de *libSVM* [3] pour implémenter les deux approches f -SVR et b -SVR.

Le tableau 4 dresse les matrices de confusions normalisées moyennes, les taux de classification moyens et les probabilités d'erreur moyennes sur les 20 tirages aléatoires.

Nous remarquons que les SVM donnent le meilleur taux de classification ($90.57 \pm 1.10\%$). Ce taux est significativement meilleur que ceux obtenus avec les k -plus proches voisins ($88.74 \pm 1.18\%$), $87.66 \pm 1.23\%$ pour le f -SVR et $88.02 \pm 1.21\%$ pour le b -SVR. En terme de détection pour chaque classe, nous remarquons que les SVM arrivent à avoir une très bonne détection de la classe sable et vase (classe 3) et une détection assez bonne des deux autres classes. Les deux approches b -SVR et f -SVR donnent une meilleure détection de la classe vase et sable (classe 3) et une bonne détection de la classe roche et cailloutis (classe 1) mais une classification moyenne de la classe ride (classe 2). Les probabilités d'erreurs

	MC	PE	TC	PE _m
<i>k</i> -ppv	$\begin{pmatrix} 73.83 & 4.47 & 21.71 \\ 12.30 & 53.53 & 34.17 \\ 1.40 & 0.74 & 97.86 \end{pmatrix}$	$\begin{pmatrix} 16.51 \\ 24.54 \\ 15.04 \end{pmatrix}$	88.74± 1.18%	18.70±1.35%
SVM	$\begin{pmatrix} 75.25 & 4.94 & 19.81 \\ 11.46 & 62.75 & 25.79 \\ 0.87 & 0.82 & 98.30 \end{pmatrix}$	$\begin{pmatrix} 15.46 \\ 20.07 \\ 12.25 \end{pmatrix}$	90.57 ± 1.10%	15.93±1.27%
<i>f</i> -SVR	$\begin{pmatrix} 65.59 & 6.15 & 28.26 \\ 12.53 & 45.65 & 41.83 \\ 0.91 & 0.45 & 98.64 \end{pmatrix}$	$\begin{pmatrix} 20.56 \\ 28.83 \\ 18.20 \end{pmatrix}$	87.66 ± 1.23%	22.53±1.46%
<i>b</i> -SVR	$\begin{pmatrix} 71.19 & 4.69 & 24.13 \\ 11.90 & 49.84 & 38.26 \\ 1.15 & 0.85 & 98.01 \end{pmatrix}$	$\begin{pmatrix} 17.67 \\ 26.46 \\ 16.59 \end{pmatrix}$	88.02 ± 1.21%	20.24±1.40%

TAB. 4 – Résultats de classification avec différents classifieurs.

moyennes montrent que les SVM commettent moins d'erreurs suivis des *k*-plus proches voisins puis les deux approches floues et crédibilistes.

5.3.3 Application sur des images sonar complètes

Jusqu'ici, nous avons utilisé seulement des imagerie homogènes. Nous présentons maintenant les résultats des expérimentations sur les images sonar complètes. Pour cela nous avons utilisé la méthode développée dans [31] pour l'évaluation des résultats de la classification. La base d'apprentissage est constituée de 24 images sonar, les 18 images qui restent sont utilisées pour la validation. La figure 6 montre l'exemple d'une image segmentée manuellement et la même image segmentée automatiquement en utilisant les SVM, le *b*-SVR et le *f*-SVR. Le tableau 5 donne les codes des trois classes utilisées. La bande rouge visible sur l'image obtenue en utilisant la segmentation automatique montre l'effet de bord puisqu'on utilise des fenêtres glissantes de taille 32×32 pixels et un pas de recouvrement de 28 pixels pour la segmentation automatique.

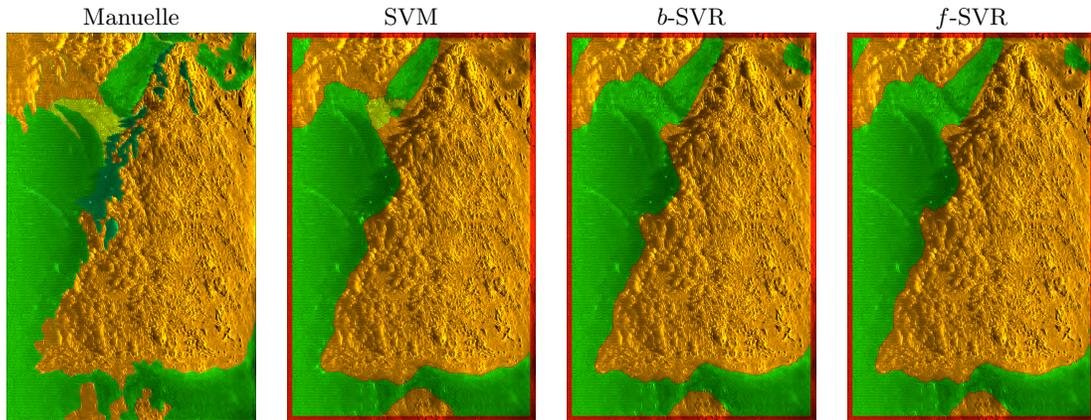
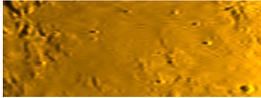


FIG. 6 – Classification automatique d'une image sonar.

Nous présentons dans le tableau 6 les résultats obtenus avec le classifieur SVM optimal avec un noyau gaussien de $\gamma = 2^{-3.6563}$ et $C = 2^{13.37}$ et avec les deux nouvelles approches les SVR floues avec un noyau gaussien et $C_f = 27864.85$, $\gamma_f = 0.123493$,

1 ^{ère} Classe	2 ^{ème} Classe	3 ^{ème} Classe
Roche et cailloutis	Ride	Sable et vase
		

TAB. 5 – Les trois classes considérées.

$\epsilon_f = 0.000008$ et les SVR crédibilistes avec un noyau gaussien et $C_b = 20188.34$, $\gamma_b = 0.088573$ et $\epsilon_b = 0.000083$.

Les résultats sont présentés sous forme de matrices de confusion modifiées pour prendre en compte le degré de certitude de l'expert comme présenté en section 4.4.2. La dernière ligne donne le taux des imagerie des autres classes (ombre et autre) qui sont classées en roche et cailloutis (classe 1), ride (classe 2) ou sable et vase (classe 3).

	SVM	b -SVR	f -SVR
MC	$\begin{pmatrix} 77.68 & 1.14 & 21.18 \\ 20.27 & 45.17 & 34.56 \\ 2.71 & 1.08 & 96.21 \\ 52.05 & 0.01 & 47.94 \end{pmatrix}$	$\begin{pmatrix} 78.39 & 0.05 & 21.56 \\ 27.75 & 30.65 & 41.60 \\ 1.40 & 0.06 & 98.54 \\ 33.85 & 0.01 & 66.14 \end{pmatrix}$	$\begin{pmatrix} 71.93 & 0.17 & 27.90 \\ 17.46 & 36.85 & 45.69 \\ 1.07 & 0.29 & 98.64 \\ 47.09 & 0.03 & 52.88 \end{pmatrix}$
PE	[23.66 27.78 19.17]	[21.31 34.69 22.28]	[26.86 31.62 22.25]
TC	85.04±0.03%	84.26±0.07%	84.97±0.01%
PE _m	23.54±0.02%	26.09±0.04%	26.91±0.05%

TAB. 6 – Performances de la classification d'images sonar en utilisant les SVM, le b -SVR et le f -SVR.

Nous avons obtenu des taux de classification de 85.04±0.03% pour les SVM, 84.26±0.07% pour le b -SVR et 84.97±0.01% pour le f -SVR. Nous remarquons la faible détection par les trois approches de la classe ride (classe 2) avec un avantage pour les SVM. Ceci peut être remarqué sur la figure 6 où les deux approches f -SVR et b -SVR n'arrivent pas à détecter la zone ride et les SVM en détectent une bonne partie. La classe sable et vase (classe 3) est bien détectée par les trois approches avec un avantage pour le b -SVR et le f -SVR : 98.54% de cette classe est bien détectée par le b -SVR et 98.64% pour le f -SVR, les SVM en détectent 96.21%. Le b -SVR et les SVM donnent la meilleure classification de la classe roche et cailloutis (classe 1) avec 77.68% pour les SVM et 78.39% pour le b -SVR et seulement 71.93% pour le f -SVR.

6 Conclusion

Nous avons proposé dans cette étude une chaîne complète de la classification des images sonar fondée sur le processus d'extraction de connaissance à partir des données. Les résultats montrent l'intérêt des méthodes développées, en particulier les deux nouvelles approches, les b -SVR et les f -SVR. Ces deux approches ont été utilisées pour remédier aux problèmes des imperfections et des incertitudes des données traitées. Nous avons montré que ces deux méthodes donnent de meilleurs résultats sur des imagerie homogènes. Nous avons soulevé le problème posé par l'automatisation des classifieurs SVM, f -SVR et b -SVR (choix des paramètres C , γ pour le noyau gaussien et ϵ pour le f -SVR et le b -SVR).

Dans cette étude des paramètres fixes pour les différentes méthodes d'analyse de texture ont été utilisées. Une étude de ces paramètres peut être envisagée pour le choix des meilleurs paramètres. Une étude envisageable est la fusion entre les différents résultats de classification, des images sonar complètes, des différentes approches étudiées vue que le résultat de classification est différent d'une classe à une autre et d'un classifieur à un autre [30].

Références

- [1] L. N. Atallah, Learning from sonar data for the classification of underwater seabeds. Thèse de doctorat, *University of Oxford*, 2004
- [2] J. Bi et T. Zhang, Support vector classification with input data uncertainty, *Advances in Neural Information Processing Systems*, 17 : pp 161-168, 2004.
- [3] C. C. Chang et C. J. Lin, LIBSVM : a library for support vector machines, 2001. *Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>*.
- [4] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines , *IEEE Transactions on Neural Networks*, 13(2002), 415-425.
- [5] R. W. Conners et C. A. Harlow. A theoretical comparison of texture algorithms. *PAMI*, 2(3) : pp 204-222, april 1980.
- [6] I. Daubechies, Ten Lectures on Wavelets. *CBMS, Philadelphia : Society for Industrial et Applied Mathematics*, 1992.
- [7] P. Démartines et J. Héroult, Curvilinear component analysis : a self organizing neural network for nonlinear mapping of data sets, *IEEE Transaction on Neural Networks*, vol. 8(1), pp 711-720, 1998.
- [8] T. Denoeux, A k-Nearest Neighbor Classification Rule Based on Dempster-Shafer Theory, *IEEE Transactions on Systems, Man, et Cybernetics - Part A : Systems et Humans*, 25(5) : pp 804-813, 1995.
- [9] D. Gabor, Theory of communication. *J. IEE*, 93 : pp 492-457, 1941.
- [10] D. E. Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning, *Addison-Wesley Pub. Co.* 1989.
- [11] S. Grigorescu, N. Petkov et P. Kruizinga, Comparison of texture features based on gabor filters. *IEEE Transaction on Image Processing*, 11 : pp 1160-1167, October 2002.
- [12] S. Gunn, Support vector machines for classification et regression. *ISIS Tech. Report*, University of Southampton, 1998.
- [13] H. Han-Pang et L. Yi-Hung, Fuzzy support vector machines for pattern recognition and data mining, *International Journal of Fuzzy Systems*, 14(3) : pp 25-28, 2002.
- [14] R. Haralick, Statistical et textural approaches to textures, *Proceedings of the IEEE*, Vol. 67(5), pp 786-804, 1979.
- [15] D. Hong et C. Hwang. Support vector fuzzy regression machines. *Fuzzy Sets et Systems*, 138 :271-281, 2003.
- [16] R. Haralick, K. Shanmugam, et I. Dinstein. Textural feature for image classification. *SMC*, 3(6) :610-621, November 1973.
- [17] J.M. Keller, M.R. Gray, J.A. Givens, A fuzzy k-NN neighbor algorithm, *IEEE Transactions on Systems, Man, et Cybernetics*, 15 : pp 580-585, 1985.
- [18] H. Laanaya, A. Martin, D. Aboutajdine, et A. Khenchaf, Régression floue et crédibiliste par SVM pour la classification des images sonar, *Extraction et Gestion des Connaissances (EGC)*, Namur, Belgique, 23-26 Janvier 2007.

- [19] H. Laanaya, A. Martin, D. Aboutajdine, et A. Khenchaf, Knowledge Discovery on Database for seabed characterization, *MCSEAI (Maghrebian Conference on Software Engineering and Artificial Intelligence) 2006*, Agadir, Maroc, 7-9 Decembre 2006.
- [20] H. Laanaya, A. Martin, A. Khenchaf, et D. Aboutajdine, Classification par régression floue et crédibiliste à base de machines à vecteurs de support, *LFA 2006*, Toulouse, France, 19-20 October 2006.
- [21] H. Laanaya, A. Martin, D. Aboutajdine et A. Khenchaf, Seabed classification using belief multi-class support vector machines, *Caractérisation du milieu marin*, Brest, France, 16-19 October 2006.
- [22] H. Laanaya, A. Martin, D. Aboutajdine et A. Khenchaf, A New Dimensionality Reduction Method for Seabed Characterization : Supervised Curvilinear Component Analysis, *IEEE Oceans'05 Europe*, Brest, France, 20-23 June 2005.
- [23] H. Laanaya, A. Martin, A. Khenchaf et D. Aboutajdine, Feature selection using genetic algorithm for sonar images classification with support vector machines, *European Conference on Propagation et Systems*, Brest, France, 15-18 March 2005.
- [24] H. Laanaya, A. Martin, A. Khenchaf et D. Aboutajdine, Une nouvelle méthode pour l'extraction de paramètres : l'analyse en composante curvilinéaire supervisée. *Atelier Fouille de données complexes dans un processus d'extraction de connaissance, Extraction et Gestion des Connaissances (EGC)*, Namur, Belgique, pp 81-85, 24-26 Janvier 2007.
- [25] I. Leblond, Recalage à long terme d'images sonar par mise en correspondance de carte de classification automatique des fonds, thèse de l'Université de Bretagne Occidentale, Brest, 2006.
- [26] I. Leblond, M. Legris et B. Solaiman, Use of Classification et Segmentation of Sidescan Sonar Images for Long Term Registration, *IEEE Oceans'05 Europe*, Brest, France, 20-23 June 2005.
- [27] M. Legris, K. Lebart, F. Fohanno, et B. Zerr, Les capteurs d'imagerie en robotique sous-marine : tendances actuelles et futures, *Revue Traitement du Signal*, Vol 20(2), pp 137-164, 2003.
- [28] S. Mallat, A Theory for Multiresolution Signal Decomposition : The Wavelet Representation, *IEEE Trans. Pattern Anal. et Mach. Intel.*, PAMI 11, 7, juillet, 1989.
- [29] A. Martin et G. Sévellec et I. Leblond, Characteristics vs decision fusion for sea-bottom characterization, *Colloque Caractérisation in-situ des fonds marins*, Brest, France, 21-22 October 2004.
- [30] A. Martin, Comparative study of information fusion methods for sonar images classification. *In International Conference on Information Fusion, (Philadelphia, USA)*, Juillet 2005.
- [31] A. Martin, H. Laanaya et A. Arnold-Bos, Evaluation for Uncertain Image Classification et Segmentation, *Pattern Recognition*, vol. 39, 2006.
- [32] M. Mignotte, C. Collet, P. Pérez, P. Bouthemy. Sonar image segmentation using a hierarchical MRF model. *IEEE Transactions on Image Processing*, 9(7) : pp 1216-1231, July 2000.
- [33] E. Osuna, R. Freund et F. Girosi. Improved training algorithm for support vector machines. *NNSP'97*. 1997.
- [34] F. Pérez-Cruz, G. Camps, E. Soria, J. Pérez, A. Figueiras-Vidal et A. Artés Rodriguez, Multi-dimensional function approximation et regression estimation, *International Conference on Artificial Neural Networks*, Madrid, Espagne, 2002
- [35] N. Pican, E. Trucco, M. Ross et D. M. Lane, Y. Petillot, I. Tena Ruiz, Texture Analysis for Seabed Classification : Co-occurrence Matrices vs Self-Organizing Maps, *IEEE/OES Oceans'98 conference* : Nice, France. September 1998.

- [36] J. Platt, Sequential Minimal Optimization : A fast algorithm for training Support Vector Machines, Microsoft Research Technical Report MSR-TR-98-14, 1998.
- [37] M. P. Sanchez Fernandez, M. de Prado Cumplido, J. Arenas-Garcia et F. P. Cruz, SVM multiregression for non-linear channel estimation in multiple-input multiple-output systems, *IEEE Transactions on Signal Processing*, 58(8) : pp 229-2307, 2004.
- [38] B. Schölkopf, A. Smola et K.R. Muller, Nonlinear Component Analysis as a Kernel Eigenvalue Problem, *Neural Computation*, vol. 10, no. 5, pp. 1,299-1,319, 1998.
- [39] Ph. Smets, Constructing the pignistic probability function in a context of uncertainty, *Uncertainty in Artificial Intelligence*, 5 : pp 29-39, 1990.
- [40] J. Smola, Regression estimation with support vector learning machines. Thèse de doctorat, Master's thesis, Technische Universität München, 1996
- [41] A. Smola et B.Schölkopf, A tutorial on support vector regression. *Neuro-COLT2 Technical Report* NC2-TR-1998-030, 1998.
- [42] H. Thomas, C. Collet, K. Yao, et G. Burel, Classification de textures en imagerie sonar et invariance en rotation, *Traitement du Signal*, Vol 17(1), 2000.
- [43] P. Thourel. Segmentation d'images sonar par modélisation markovienne hiérarchique et analyse multirésolution. Thèse de l'Université de Bretagne Occidentale UFR Sciences et technique mention électronique, 1996.
- [44] D. Tsujinishi et S. Abe, Fuzzy least squares support vector machines for multiclass problems, *Neural Networks*, 16 : pp 785-792, 2003.
- [45] V. N. Vapnik, Statistical Learning Theory, *John Wesley et Sons*, 1998.
- [46] M. B. Vladan et R. Gopakumar, Seabed Recognition Using Neural Networks, *D2K Technical Report - 0399-1*, March, 1999.
- [47] Weldon, T. P., Higgins, W. E. et Dunn, D. F. (1996). Gabor filter design for multiple texture segmentation, *Optical Engineering*, 35(10) : pp 2852-2863, 1996.
- [48] J. Yang, V. Honavar, Feature subset selection using a genetic algorithm. *IEEE Transaction on Intelligent Systems* 13(2) : pp 44-49, 1997.
- [49] L. Zadeh, Fuzzy Sets as the Basis for a Theory of Possibility, *Fuzzy Sets and Systems* 1 :3-28, 1978. (Reprinted in *Fuzzy Sets and Systems* 100 (Supplement) : 9-34, 1999.)