SEAFLOOR SEDIMENT CLASSIFICATION OF SONAR IMAGES

Mrs K.S.Jeen Marseline[1], Dr.C.Meena[2]
[1]*Assistant Professor, Sri Krishna Arts & Science College, Coimbatore*
[2]*Center Head Avinashilingam Institute For Home Science And Higher Education For Women Coimbatore*

ABSTRACT

As the images collected by sonar are high-dimensional and difficult to characterize, it is very challenging to detect a specific kind of sediment, for example the rocks that can be used as landmarks for image registration during underwater navigation. In these situations, a classification system of seafloor sediments is considered as one of the key areas of interest by sonar image applications. This paper concentrates on designing and developing feature extraction and proposed a enhanced classification model SVM-ELM classifier to identify the different objects, like sand, rock, silt, ripple, cobbles and other objects. Various experiments were conducted to evaluate the proposed algorithm to identify different sediments of sonar images.

Keywords: *sediments, feature extraction, classification, high-dimensional, seafloor, underwater navigation, SVM-ELM*

I. INTRODUCTION

After preprocessing the sonar image, the next step is feature extraction and selection. The main goal of this phase is (i) Feature Extraction: Converts the image data into a representation that allows classification of image data into sediment regions (ii) Dimensionality reduction: Selects optimal subset of features from the extracted features. The task of the feature extraction and selection methods is to obtain the most relevant information from the original data and represent that information in a lower dimensionality space. Developing low-dimensional discriminative features is crucial for any classification system. Feature selection is an important task that allows the determination of the most relevant features for seafloor sediment grouping. The objective of feature selection is three-fold:

- Improving the performance
- Providing a faster and more cost-effective classification
- Providing a better understanding of the underlying process that generated the data

The goal of feature selection is to reduce the dimensionality of vectors associated to patterns by selecting a subset of attributes smaller than the original. The algorithms used in this paper works to find the optimal feature subspace where the “relevant” and “irrelevant” image sets can be best separated, the similarity between these two sets are used as the feature selection criterion. The performance of the subsequent steps like clustering, indexing and matching is often improved by eliminating redundant features. The second stage is the extraction and selection of texture features from sediment region obtained from the first step. These features are later used for sediment region classification (Stage III). An image texture feature is a set of metrics calculated to provide information about the spatial arrangement of color or intensities in an image or selected region of an image [1]. Texture features can be extracted using either statistical approaches or transformation approaches. Haralick [2] presented one of the most widely used statistical approaches to texture analysis, namely, the spatial gray level dependence matrix (SGLDM) or the gray level co-occurrence matrix (GLCM) approach. They listed 14 texture features that utilize the spatial relationship amongst gray level values of pixels with in a region. A comprehensive review on statistical algorithms is provided by [3][4]. However, the GLCM has one major issue, namely, the computation cost [5] [6] [7] [8] Another most frequently used area for texture feature extraction is the Transformation-based techniques. Transformation of images based on Discrete Wavelet Transformation (DWT), Fourier Transformation (FT), Gabor Transformation (GT) have also been extensively used for texture analysis. GT based feature extraction method are computationally complex [9] [10] and it has been proved that DWT are more efficient than FT based techniques [11] [12]

II. CLASSIFICATION

In classification domain, Support Vector Machines (SVMs) are state-of-the-art models which have been used to solve many problems. Their success is due to the following three factors,

(i) They are maximum-margin classifiers

(ii) The dual form of the SVM optimization problem is quadratic and its computational complexity depends on the number of data, not on their dimensionality

(iii) Since the optimization of their dual form only needs the inner products of data points and not the data points themselves,kernels can easily be plugged into SVMs.

Scholtena [13] proposed an optimized SVM classifier for object classification which used a K-Means algorithm to enhance the operation of SVM classifier. The performance of the algorithm is directly dependent on the performance of K-Means algorithm, which varied according to the selection of K. Since the incorrect estimation of K degrades the performance, in this study, the K-Means algorithm is omitted and SVM classifier is enhanced by combining it with ELM (Extreme Learning Machine) [14]. This model uses the segmented result from the previous Phase and extracts wavelet packet-based texture features. The proposed enhanced classifier is trained and tested to recognize six types of sea floor sediment areas, namely, sand, rock, ripples, silt, cobbles and other objects.

2.1 Support Vector Machine

The second classifier used is Support Vector Machine (SVM), which given a set of input data and predicts, for each given input, which of two possible classes the input is a member [15]. This makes SVM a non-probabilistic binary linear classifier. SVM classifier is well known for its generalization performance and ability to handle high dimensional data. Support Vector Machines (SVMs) were introduced by [16] and have proved to be fast effective classifier. Considering the binary classification case, let $((x_1, y_1) \ldots (x_n, y_n))$ be the training dataset where xi are the feature vectors that represent the observations and $y_i \in (-1, +1)$ be the two labels that each observation can be assigned to. From these observations, SVM builds an optimum hyperplane (a linear discriminant in the kernel transformed higher dimensional feature space) that maximally separates the two classes by the widest margin by minimizing the following objective function (Figure 1, Equation 1).

$$\min_{w,b,\xi_i} \frac{1}{2} w.w^{T} + C\sum_{i=1}^{N}\xi_i \qquad (1)$$

wherew is the norm of the hyperplane, b is the offset, $y(x_i)$ are the labels and $\xi_i$are the slack variables that permit the non-separable case by allowing misclassification of training instances.

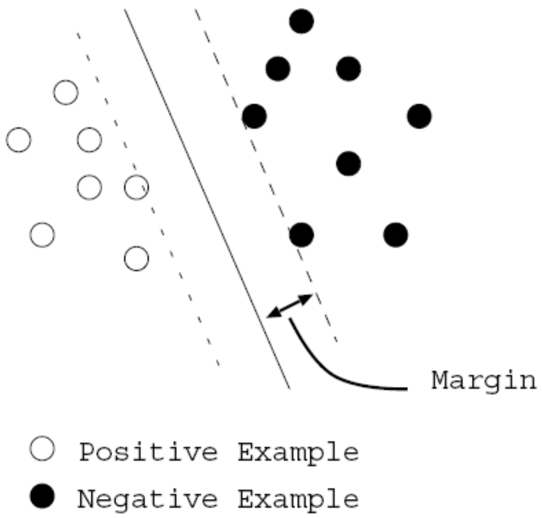

Fig. 1: Support Vector Machine Hyperplane

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

2.2 Extreme Learning Machine

ELM is a single hidden layer feed forward neural network where the input weights are chosen randomly and the output weights are calculated analytically. ELM employs any continuous / discontinuous non-linear function (example sigmoidal or Gaussian functions) as an activation function in the hidden layer and a linear activation function in the output layer. This discussion uses a Gaussian function as an activation function in the hidden layer. The architecture of ELM network is shown in Fig 2.

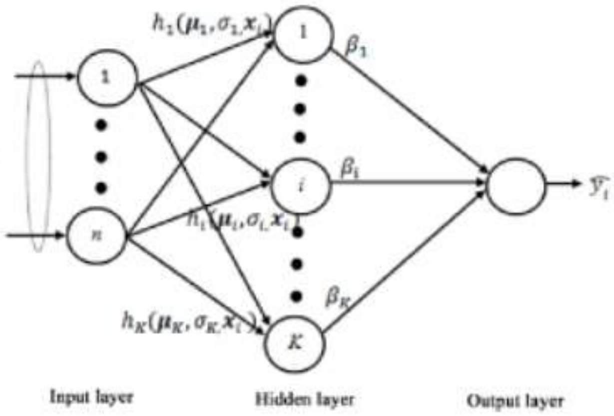

Fig . 2: ELM Network

A multi-category classification problem can be stated in the following manner. Let there are N observation samples $\{X_i, Y_i\}$, where $X_i = \{x_{i1}, \ldots, x_{in}\} \in R^n$is an n-dimensional features of sample i and $Y_i = \{y_{i1}, y_{i2},$

…, $y_{iC}\} \in R^C$ is its coded class label. If the sample X_i is assigned the class label c_k, then the k^{th} element of Y_i is one (y_{ik}= 1) and other elements are –1. Here, it is assumed that the samples belong to the C distinct classes. The function describing the useful information on probability of predicting the class label with the desired accuracy is called as classifier function and is defined as

$$Y = F(X) \tag{2}$$

The objective of the classification problem is to estimate the functional relationship between the random samples and its class label from the known set of samples. Using universal approximation property, it can be said that the single layer feedforward network with sufficient number of hidden neurons H can approximate any function to any arbitrary level of accuracy . It implies that for bounded inputs to the network there exist optimal weights (not necessarily unique) to approximate the function. Let W_i be H x n input weights, B be H x 1 bias of hidden neurons and W_o be C x H output weights. The output (Y^*) of the ELM network with H hidden neurons has the following form:

$$F^*(X_i) = \sum_{j=1}^{H} w_{o_{kj}} G_j(W_i, B, X_i) \tag{3}$$

where k = 1, 2, …, C and $G_j(.)$ is the output of the j^{th} hidden neuron and G(.) is the activation function. For the sigmoidal hidden neurons, the output of the j^{th} hidden neuron $G_j(.)$ is defined using Equation (4).

$$G_j = G\left(\sum_{k=1}^{n} W_{i_{jk}} x_{ik} + b_j \right) \tag{4}$$

where j = 1, 2, …, H. In case of the radial basis function (RBF), the output of the j^{th} Gaussian neuron $G_j(.)$ is defined as

$$G_j = G(b_j \, \|X-W_i\|) \tag{5}$$

with j = 1, 2, …, H and W_i and $b_j (b_j \in R^+)$ are the center and width of the RBF neuron, respectively. Equation (6) can be written in matrix form as

$$Y^* = W_o \, Y_H \tag{6}$$

where Y_H is given by Equation (7).

$$Y_H = \begin{bmatrix} h_1(\mu_1,\sigma_1,x_1) & h_1(\mu_1,\sigma_1,x_2) & \cdots & h_1(\mu_1,\sigma_1,x_N) \\ \vdots & \vdots & & \vdots \\ h_k(\mu_k,\sigma_H,x_1) & h_k(\mu_k,\sigma_H,x_2) & \cdots & h_k(\mu_k,\sigma_H,x_N) \end{bmatrix} \tag{7}$$

Here, Y_H(dimension H x N) is called the hidden layer output matrix of the neural network and the i^{th} row of Y_H is the i^{th} hidden neuron outputs for the entire training input X. For most of the practical problems, it is assumed that the number of hidden neurons is always less than that of training samples. In the ELM algorithm, for a given number of hidden neurons, it is assumed that the input weights W_i and the bias B of hidden neurons are selected randomly. By assuming the predicted output Y^* is equal to the coded labels Y, the output weights are estimated analytically as $W_o^* = YY_H^+$ where Y_H^+ is the Moore-Penrose generalized pseudo-inverse of Y_H . In summary, the following are the steps involved in the ELM algorithm:

- For a given training samples $\{X_i, Y_i\}$, select the appropriate activation function G(.) and the number of hidden neurons;
- Generate the input weights W_i and the bias values B randomly.

- Calculate the output weights $W_o$analytically: $W_o = YY_H^+$.

ELM classifiers are fast, but they do not search for maximum-margin hyperplanes. Instead, they minimize a sum of squared errors between the class labels and the MLP output.

2.3 SVM-ELM Classifier

When using SVMs, three choices must be made, namely, the kernel type, the kernel parameters and the regularization parameter. These choices are critical for the quality of the results and are usually done using cross-validation. However, this process can be computationally intensive since many models have to be built. To solve this issue, the study uses ELM algorithm which are then used as kernel to SVM. This model offers two important gains, namely, fast to train (obtained through the use of ELM) and maximum-margin classifiers (obtained through the use of SVM). This proposed method merges extreme learning machines (ELMs) with SVM to obtain classification models which are fast to train and maximum-margin classifiers. ELM allows focusing on the regularization. It proposes to build an explicit feature space using random neurons, as in extreme learning. Then, the corresponding Gram matrix is computed and used with a standard SVM. The proposed SVM-ELM classifier is shown in Fig 3.

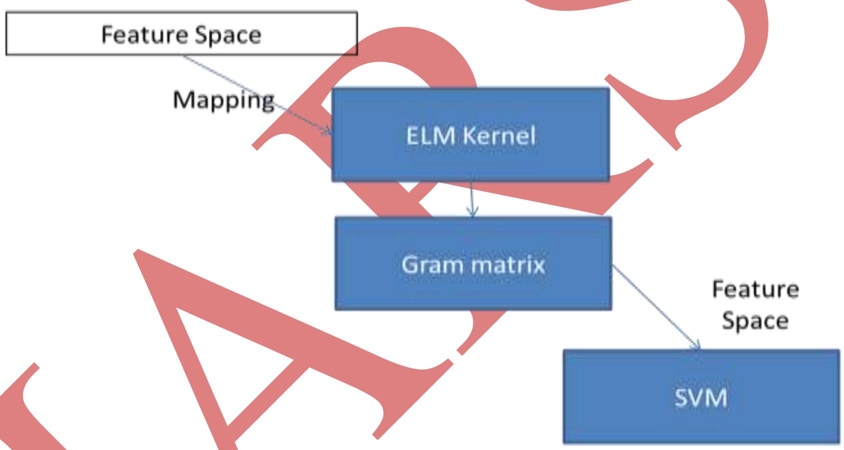

Fig. 3: Proposed SVM-ELM Classifier

In the ELM framework, the hidden layer is treated as a mapping ϕ from the feature space $R^d$to a feature space $R^p$where a linear is to be solved. This is very similar to the idea behind the use of kernels: statistically, the data points are easier to separate in a higher dimensional space. Hence, the first layer of an ELM can be thought as defining some kind of randomised kernel, which will be subsequently called the ELM kernel. Given two feature points x and z and an ELM with p neurons defining a mapping ϕ from $R^d$to R^p, the corresponding ELM kernel function is defined as

$$k(x, z) = \phi(x) \,.\, \phi(z) \; \frac{1}{p} \phi(x) \,.\, \phi(z) \qquad (8)$$

Therefore, the Gram matrix corresponding to this ELM kernel and a set of data points can be computed as

$$G = \frac{1}{p} \Phi\Phi' \qquad (9)$$

The enhanced SVM is thus formed by using the above kernel into the standard SVM.

III. PERFORMANCE EVALUATION

This section presents the results of the performance evaluations conducted with the conventional and proposed models, for the six types of sediments. The accuracy % , Error rate in % and Speed in seconds are compared with the conventional SVM and ELM classifiers.

Table 1. Classification models

S.No.	Description	Code
1	Support Vector Machine	SVM
2	Extreme Learning Machine	ELM
3	Proposed MODEL	ASSCS

As mentioned previously, the study considers six types of sediements, namely, sand, rock, ripple, silt, cobbles and other regions. Table 2 presents the accuracy of the ASSCS classifiers in identifying each of the selected sediment type. The results are compared with its conventional counterparts, SVM and ELM.

Table 2. Classification accuracy (%)

Sediment Type	SVM	ELM	ASSCS
Sand	83.56	85.81	91.62
Rock	84.21	85.44	89.93
Ripple	82.63	83.72	88.39
Silt	80.92	82.56	88.04
Cobbles	81.29	83.05	87.10
Others	80.88	81.45	86.25

Table 3 Presents The Error Rate Of The Classifiers

Table 3. Classification error rate (%)

Sediment Type	SVM	ELM	ASSCS
Sand	8.52	7.47	4.31
Rock	9.04	7.16	4.58
Ripple	9.51	7.86	4.93
Silt	9.57	9.02	6.25
Cobbles	9.99	9.48	7.58
Others	8.48	7.43	4.30

The results pertaining to error rate shows a trend that is similar to accuracy. The proposed model reduces the error rate.

Table 4 presents the speed of the proposed ASSCS and the conventional SVM and ELM classifiers while grouping seafloor image pixels into the selected six categories. The speed measured in seconds is the testing time only.

Table 4. Classification speed (seconds)

Sediment Type	SVM	ELM	ASSCS
Sand	9.73	6.58	4.33
Rock	9.77	6.63	4.81
Ripple	9.80	5.90	5.33
Silt	9.83	6.55	6.72
Cobbles	9.87	6.04	7.29
Others	9.91	6.50	4.53

IV.CONCLUSION

The proposed model for sediment classification of sonar images produces better results compared to the conventional methods. The accuracy and error rate table shows the better performance compared to the conventional model. As evident from the table results, the speed of the enhanced SVM algorithm has increased by the inclusion of ELM algorithm.

REFERENCES

[1] Shapiro, L.G. and Stockman, G.C. (2001) Computer Vision, New Jersey, Prentice-Hall, Pp 279-325.

[2] Haralick, R. M., Shanmugam, K. and Dinstein, I. (1973) Textural features for image classification, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 3, No. 6, Pp. 610-621.

[3] Haralick, R.M. (1979) Statistical and structural approaches to texture, Proceeding of the IEEE, Vol. 67, Pp. 786- 804.

[4] Clausi, D.A. and Jernigan, M.E. (1998) A fast method to determine co-occurrence texture features, IEEE Trans on Geoscience & Rem. Sens., Vol. 36, No. 1, Pp. 298-300.

[5] Clausi, D.A. and Zhao, Y. (2002) An advanced computational method to determine co-occurrence probability texture features, IEEE Int. Geoscience and Rem. Sens. Sym, Vol. 4, Pp. 2453-2455.

[6] Svolos, A.E. and Todd-Pokropek, A. (1998) Time and space results of dynamic texture feature extraction in MR and CT image analysis, IEEE Trans. on Information Tech. in Biomedicine, Vol. 2, No. 2, Pp. 48-54.

[7] Kiranyaz, S. and Gabbouj, M. (2007) Hierarchical Cellular Tree: An Efficient Indexing Scheme for Content-based Retrieval on Multimedia Databases, IEEE Transactions on Multimedia, Vol. 9, No. 1, Pp. 102-119.

[8] Saraswat, M., Goswami, A.K. and Tiwari, A. (2013) Object Recognition Using Texture Based Analysis, International Journal of Computer Science and Information Technologies, Vol. 4, No. 6, Pp. 775-782

[9] Mokji, M.M. and Abu Bakar, S.A.R. (2007) Gray Level Co-occurrence Matrix Computation Based on Haar Wavelet, Computer graphics, Imaging and Visualization (CGIV) IEEE, pp1-8

[10] Al Wadi, S., Ismail, M.T. and Karim, S.A.A. (2010) A comparison between haar wavelet transform and fast fourier transform in analyzing financial time series data, Research Journal of Applied Sciences, Vol. 5, Issue 5, Pp. 352-360.

[11] Sifuzzaman, M., Islam, M.R. and Ali, M.Z. (2009) Application of wavelet transform and its advantages compared to Fourier transform, Journal of Physical Sciences, Vol13,Pp.121-134.

[12] Frénay, B.andVerleysen, M. (2010) Using SVMs with randomized feature spaces: an extreme learning approach, Proceedings, European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning, Belgium, Pp. 315-320.

[13] Gondra, I. (2008) Applying machine learning to software fault-proneness prediction, The Journal of Systems and Software, Science Direct, Vol. 81, Pp.186–195.

[14] Vapnik, V. (1998a) Statistical Learning Theory, New-York: John Wiley

[15] Suresh, S., Venkateshbabu,R. and Kim, H.J. (2009) No-reference image quality assessment using modified extreme learning machine classifier, Applied Soft Computing, Vol. 9, Pp.541–552.

[16] Penrose, R. (1955) A generalized inverse for matrices, Cambridge Philosophy Society, Vol. 51, Pp. 406–413.