Vol 8. No. 2 June, 2015
African Journal of Computing & ICT

www.ajocict.net

Fish Classification Using Support Vector Machine

S.O. Ogunlana, O. Olabode , S.A. A. Oluwadare & G. B. Iwasokun
Department of Computer Science
Federal University of Technology
Akue, Nigeria

ABSTRACT

Fish recognition is presently a very complex and difficult task despite its commercial and agricultural usefulness. Some of the challenges facing accurate and reliable fish recognition include distortion, noise, segmentation error, overlap and occlusion. Several techniques, which include K-Nearest Neighbor (KNN), K-mean Clustering and Neural Network, have been widely used to resolve these challenges. Each of these approaches has inherent limitations, which limit classification accuracy. In this paper, a Support Vector Machine (SVM)-based technique for the elimination of the limitations of some existing techniques and improved classification of fish species is proposed. The technique is based on the shape features of fish that was divided into two subsets with the first comprising 76 fish as training set while the second comprises of 74 fish as testing set. The body and the five fin lengths; namely anal, caudal, dorsal, pelvic and pectoral were extracted in centimeter (cm). Results based on the new technique show a classification accuracy of 78.59%, which is significantly higher than what obtained for ANN, KNN and K-mean clustering-based algorithms.

Keywords: Support Vector Machine, Image Recognition/ Classification

African Journal of Computing & ICT Reference Format:
S.O. Ogunlana, O. Olabode , S.A. A. Oluwadare & G. B. Iwasokun (2015). Fish Classification Using Support Vector Machine. Afr J. of Comp & ICTs. Vol 8, No. 2. Pp 75-82.

I. INTRODUCTION

Fish recognition is the act of recognizing or identifying fish species based on their features. It is also a process of identifying fish targets to species based on similarity to images of representative specimens [1]. Fish recognition is necessary for a number of reasons, which include pattern and contour matching, feature extraction, determination of physical or behavioral trait and statistical and quality control of fish species [2]. Fish recognition is also beneficial to fish counting and population assessments, description of fish associations and monitoring ecosystems [3]. Accurate recognition of fish species is important, as there are often legal restrictions on fishing practices when their existence is considered threatened or endangered.

Fish recognition is a challenging and worthy task judging from its high demand for commercial and agricultural purposes. Some of the challenges militating against accurate fish recognition include distortion, noise, segmentation error, overlap and occlusion [4]. Traditionally, marine biologists identify fish from their ichthyologic characteristics such as meristics and mophometrics, scale morphology and so on [5]. Statistical classification methods such as Principal Component Analysis (PCA), Discriminant Function Analysis (DFA) and classification tree have been used in fish recognition with their attendant limitations [6] which have prompted the shift to Machine Learning (ML) which provides tool for identifying structures in complex and nonlinear data as well as generating accurate predictive models.

ML method consists of a range of approaches that rely on Artificial Neural Network (ANN), Fuzzy logic, K-Nearest Neighbor (KNN), K-means clustering and Support Vector Machine (SVM) [7]. SVM (also referred to as Maximum Margin Classifiers (MMC)) consists of a group of learning algorithms, originally developed by Vapnik [8]. It performs simultaneous minimization of the empirical classification error and maximization of the geometric margin. A SVM performs classification by constructing an N-dimensional hyper-plane that optimally separates data into two categories based on an algorithm that finds the maximum-margin hyper plane with the greatest separation between classes.

The instance at the minimum distances from the maximum-margin hyper plane constitutes the support vectors. In this paper, a SVM-based platform for fish recognition and classification is presented. Section II presents a review of relevant literature on fish classification and SVM, Section III focuses on the summary of some existing fish classification techniques while Section IV presents the design of the proposed system. Sections V and VI focus on the experimental study and discussion respectively.

2. RELATED LITERATURE

The authors in [9] presented a computer vision-based system that reliably classifies different fish species based on length

African Journal of Computing & ICT

www.ajocict.net

measurement and weight determination. The system has capability for using vision-based catch registration for automatic classification of fish species but requires great computer power and very expensive computation. Furthermore, it is only applicable to fish length and weight measurements. In [10], an artificial neural network-based platform for fish species identification is presented. The platform uses several statistical methods such as discriminate function analysis and principal component analysis. Its limitation is its high false identification rate due to its reliance on overtraining of fish species.

The authors in [11], presented a system for acoustic identification of small pelagic fish species using support vector machines and neural networks. Though the system out-performs the statistical methods, it experiences data imbalance and greater error for less represented fishes. In [12], an ANN and Decision Tree-based platform for fish classification based on feature selection, image segmentation and geometrical parameter techniques is presented. The platform suitably recognized fish image and analyze their individual impact but is procedurally extensive, time consuming and cumbersome. Automated techniques for detection and recognition of fishes using computer vision algorithms are proposed in [13, 14]. The techniques are highly noted for automated process of fish detection and recognition from video or still camera source. Their limitations include lack of support for fish in different positions and illuminations. A fish classification system that is based on color texture measurements is proposed in [15]. The system uses the gray level co-occurrence matrix (GLCM) method for the extraction of feature that promotes robust classification based on colour textures. The method records significantly high time for network training with tendency for the neighboring pixels of the fish texture to converge to each other thereby, making smooth and accurate fish recognition difficult.

In [16], a hierarchical approach to recognize live fish from underwater video is proposed. The approach focuses on feature extraction, hierarchical classification and tree construction as its core function. Although, the work achieved better accuracy compared to some other techniques, it is time consuming, computationally bulky and used one to one classifier and imbalanced datasets which are not sustainable for large number datasets. Lee et al. in [17] carried out shape analysis of fish and developed an algorithm for removing edge noise and redundant data point base on nine species with similar shape features. Decision tree was presented as a suitable method for high accuracy while the number of shape characters needed and how to use them depend on the number of species and the kind of species required. Experiments conducted on a given number of fish images of various species recorded very significant classification success.

Larsen et al. in [18] presented a shape and texture based fish classification method with several images and species. Shape and texture features were separated using active appearance model that is based on principal component scores and linear discriminate analysis. Rova et al. in [19] applied SVM algorithm to fish recognition and constructed a texture-based mechanism that distinguishes between the Striped Trumpeter and the Western Butterfish species. Two templates (one per specie) were built and each query image was warped to both templates for the texture-based classifier.

3. EXISTING FISH CLASSIFICATION ALGORITHMS

Fish classification is necessary for identification, marketability, pricing, consumption, scientific research and so on and the summary of some of the existing fish classification algorithms are presented below:

Artificial Neural Networks (ANNs) Algorithm

ANNs comprised of simple neurons with three basic elements: a set of synaptic weight, integration and activation function. The mathematical model of the neuron K is expressed as follows [1, 10, 12]:

$$U_k = \sum_{j=1}^{P} w_{kj} x_j + b_k \tag{1}$$

$$y_k = \varphi(U_k) \tag{2}$$

x_j is the input signal, w_{kj} is the weight from the jth to kth neuron, b_k is the bias of the kth neuron, φ(.) is the activation function, and y_k is the output of the kth neuron. Several types of activation functions are used in ANN, but the sigmoid function is used as follows:

$$\varphi(u) = \frac{1}{1 + e^{-u}} \tag{3}$$

The sigmoid function generates a continuous valued output between 0 and 1 as the neuron's net input goes from negative to positive infinity. Training the network involves the back-propagation algorithm and the goal is to find a set of connection weights that minimizes an error function. The back-propagation algorithm consists of following four steps:

- Compute the error derivative (EA) as follows:

$$EA_j = \frac{\partial E}{\partial y_j} = y_j - d_j \tag{4}$$

  E represents Error, y_j is the activity level of the jth unit and d_j is the desired output of the jth unit.

- Computes how fast the error change as the total input received by an output is changed as follows:

$$EI_j = \frac{\partial E}{\partial x_j} = \frac{\partial E}{\partial y_j} \times \frac{\partial y_i}{\partial x_j} = EA_j y_j \ (1 - y_j) \tag{5}$$

- Computes how fast the error changes as the weight on the connection into an output unit is changed as follows:
-

$$EW_{ij} = \frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial x_j} \times \frac{\partial x_i}{\partial w_{ij}} = EI_j y_j \tag{6}$$

Vol 8. No. 2 June, 2015
African Journal of Computing & ICT

www.ajocict.net

- Computes how fast the error changes as the activity of a unit in the previous layer is changed as follows:
-

$$EA_i = \frac{\partial E}{\partial y_i} = \frac{\partial E}{\partial x_j} \times \frac{\partial x_j}{\partial y_j} = \sum_j EI_j w_{ij} \quad (7)$$

Principal Component Analysis (PCA) Algorithm

PCA is used to reduce the dimensionality of a dataset. Given that x^T is an $n \times p$ observation mean-centered data matrix with n–observation of p-variables, the mean centering is expressed as $x_i \leftarrow \bar{x}_I$, x_i is the ith elements of the vector x and $\bar{x}$ is the mean of its elements. The covariance matrix S of X is defined as [20]:

$$S = COV(X) \equiv \frac{1}{n-1} XX^T \quad (8)$$

N is the number of observation, X is the matrix and X^T is transpose matrix of X. The first linear function is defined as $Z_1 = \alpha_1^T x$. In PCA, the variance of the linear function Z_1 is maximize as follows:

$$Z_1 = \alpha_1^T x = \alpha_{11}x_1 + \alpha_{12}x_2 + \ldots + \alpha_{1p}x_p$$
$$= \sum_{j=1}^{P} \alpha_{ij} x_j \quad (9)$$

α is the eigenvector of covariance.
The variance is maximize using the Lagrange multiplier γ as follows:

$$\frac{\delta}{\delta x}[f(x) - \gamma g(x) - 1] = 0 \quad (10)$$

$$f(x) = \alpha_1^T S \alpha_1 \quad (11)$$
$$g(x) = \alpha_1^T \alpha_1 \quad (12)$$

The differential gives the eigenvalue for the covariance matrix as follows:

$$s\alpha_1 - \gamma\alpha_1 = 0 \text{ or } (s - \gamma I_p)\alpha_1 = 0 \quad (13)$$

I_p is a $p \times p$ identity matrix and γ is the variance. The percentage variation by the corresponding principal component is calculated from the respective eigenvalues as follows:

$$P\% = \frac{\gamma_1}{\gamma_1 + \ldots + \gamma_p} 100\% \quad (14)$$

K-Nearest Neigbour (KNN) Algorithm

Given a training set $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ where x_i *is d*-dimensional feature vector of real numbers, for all i, y_i is the class label for all *I*, then the task is to find y_{new} from x_{new} . KNN algorithm involves finding k closest training points to x_{new} with respect to the Euclidean distance. The Euclidean distance is defined as [21, 22]:

$$E = \sqrt{[a_1(x_j - x_{new})^2 + \ldots + a_d(x_j - x_{new})^2} \quad (15)$$

$a_1, a_2, \ldots, a_d$ represent the scaling factor for different dimensions. During testing, for new test data $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$, the classifier generates label $y_{n+1}^{knn}, \ldots, y_{n+m}^{knn}$. The percentage of accuracy, A is obtain as follows:

$$A = \frac{a}{b} \times 100 \quad (16)$$

a is number of correctly classified items and b is the number of items respectively.

K-means Clustering Algorithm

K-means clustering is an algorithm that classifies or groups object into k number of group based on attributes or features. The grouping is based on minimization of the sum of squares of the distances between data and the corresponding cluster centroid.

The k-mean clustering algorithm is presented as follows [23, 24]:

- Initialize cluster centroid $\mu_1, \ldots, \mu_k \in R^n$ randomly.
- Repeat until convergence: {

$$\text{for every set } i,$$
$$c_i = argmin\|x_i - \mu_j\|^2 \quad (17)$$
$$\text{for every set } j,$$
$$\mu_i = \frac{\sum_{i=1}^{m} 1\{c_i = j\} x_i}{\sum_{i=1}^{m} 1\{c_i = j\}} \quad (18)$$

}

k is the number of cluster, μ_i is cluster centroids.
The initialization is by randomly picking k training set while the inner-loop repeatedly assign each training set x_i to the closest cluster centroid μ_j and move each cluster centroid μ_j to the mean of the points assigned to it. The centroid coordinate is determined thus:

Given x^k, the ith coordinate of x^{k+1} is given by x_i^{k+1} =argmin $f(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, y, x_{i+1}^{k}, \ldots, x_n^{k})$; for y $\in R$. Thus, an initial guess x^0 is used for a local minimum of F, and a sequence x_1^0, x_1^1, x_1^2,...*is obtained* iteratively.
Using line search in each iteration, then $F(x^o) \geq F(x^1) \geq F(x^2), \ldots$ while the K-means convergence is achieved by defining the objective function:

$$j(c, \mu) = \sum_{i=1}^{m} \|x_i - \mu_{ci}\|^2 \quad (19)$$

www.ajocict.net

The inner-loop of k-means repeatedly minimizes j with respect to c while holding µ fixed and minimizes j with respect to µ while holding c fixed and thus, the value of j converges.

4. PROPOSED FISH RECOGNITION SYSTEM

The architecture of the proposed system is presented in Figure 1. The system uses the support vector machine (SVM) process (shown in Figure 2) to analyze and then classify fish species according to their features or characteristics by constructing an N-dimensional hyper-plane that optimally separates fish species into categories. The hyper-plane is based on a predictor variable and a vector of predictor values (which is the set of values assigned to the different fields in the dataset) is formed. Most importantly, SVM modeling is used to find the optimal hyper-plane that separates clusters of vector in such a way that cases with one category of the target variable (classified as the training subset) are on one side of the plane and cases with the other category (classified as testing set) are on the other side [4]. For this research, the training set is a collection of the features of the fish and it is used to discover the predictive relationship among the fish species while the testing set consists of the features actually extracted from the fish species and is used to access the strength of the SVM algorithm for correct classification. The SVM algorithm comprises of the following components:

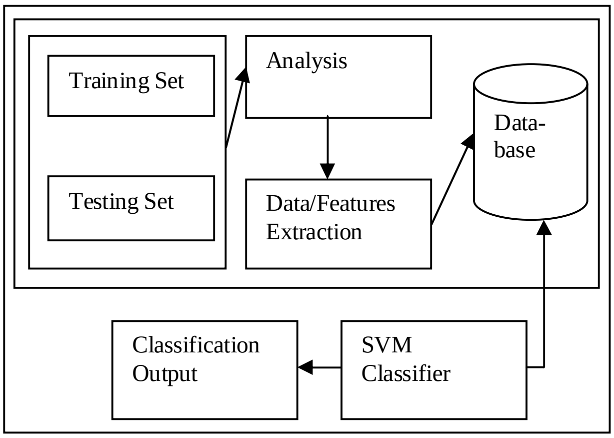

Figure 1: Proposed Fish Recognition Model

Optimal Separating Hyper-plane (OSH)

With OSH, there is assumption that a set S of points $X_n \in R^n, n = 1,2,3,\dots,N$ *exists such that each* point X_n belongs to one of the two classes and thus is given a label $Y_n \in \{-1,1\}$. The goal is to establish the equation of a hyper-plane that divides S leaving all the points of the same class on the same side while maximizing the distance between the two classes and the hyper-plane. The set S is linearly separable if there exist $W \in R^n$ for dataset of size T and $b \in R$ such that the separating hyper-plane is defined as follows [25]*:*

$$y_n[W^T X_n + b] \ge 1, for\ n = 1,2,\dots,N, \quad (20)$$

the pair *(w,b)* defined a hyper-plane of the equation:

$$[W^T X_n + b] = 0 \quad (21)$$

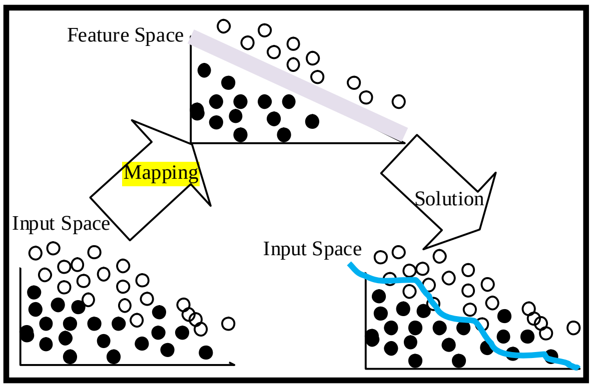

Figure 2: Overview of SVM process (http://www.dtreg.com)

If the norm of **w is** denoted with *w*, then the signed distance d_n of a point $\mathbf{x}_n$ from the separating hyper-plane (**w**, *b*) is defined by:

$$d_n = \frac{w^T x_n + b}{w} \quad (22)$$

The integration of Equations 1 and 3 for all $X_n \in S$ *gives:*

$$y_n X_n \ge \frac{1}{w} \quad (23)$$

$\frac{1}{w}$ is the lower bound on the distance between point x_n and the separating hyper-plane (**w**, *b*). ***Given a separating hyper-plane (w, b) for the linearly separable set S, the canonical representation of the separating hyper-plane is obtained by rescaling the pair (w, b) into the pair*** (w', b') ***in such a way that the distance of the closest point, say*** x_j***, equals*** $\frac{1}{w}$ ***[25]***)

$$min_{xj \in X_n}\{y_n[w'X_n + b]\} = 1 \quad (24)$$

A separating hyper-plane is the canonical representation and given a linearly separable set S, the optimal separating hyper-plane is the one with the closest distance (which in most cases is equal to $\frac{1}{w}$) to S.

The Margin of Separation (MS)

The margin of separation is the lower bound of the minimum distance between points of different classes. MS can be coupled with OSH to form a separating hyper-plane that maximizes the margin which can be thought of as a measure of the difficulty of the problem. In general, the quadratic optimization problem with linear constraints is expressed based on MS as:

African Journal of Computing & ICT

www.ajocict.net

$$Minimize\ \frac{1}{2}\|w\|^2 = \frac{1}{2}w^T w$$

Subject to $y_n[w^T x_n + b] \geq 1 \quad n = 1,2,3, \dots, N$ (25)

$$with\ respect\ to\ w \in R^d, and\ b \in R$$

w is the Euclidean space R and b is a scalar belonging to real number.

The optimization problem has a convex quadratic objective and only linear constraints. Its solution, obtained by using the Lagrange Multipliers Technique (LMT), gives the optimal margin classifier as well as the separating hyper-plane with the best possible margin. The LMT for a multiplier α_n is defined as follows:

$$Minimize\ L(w, b, \alpha_n) = \frac{1}{2}w^{T+1} - \sum_{n=1}^{N} \alpha_n[y_n(w^T X_n + b) - 1] \quad (26)$$

$$subject\ to\ y_n[(w^T X_n + b)] \geq 1, \ \ n = 1,2,3, \dots, N$$

$\alpha_1, \dots, \alpha_n$ is a vector of non-negative Lagrange Multipliers.

The solution is obtained by solving the standard quadratic programming:

$$w = \sum_{n=1}^{N} \alpha_n y_n x_n \quad (27)$$

$$b = \sum_{n=1}^{N} (w^T x_n - y_n) \quad (28)$$

$$\sum_{n=1}^{N} \alpha_n y_n = 0 \quad (29)$$

N is the number of support vectors and only a few α_n is greater than 0, corresponding to the support vectors. The solutions to *w*, *b*, and α_n *are* still unknown. To solve for α_n, $w = \sum_{n=1}^{N} a_n y_n x_n$ and $\sum_{n=1}^{N} a_n y_n = 0$ are sunstituted into $L(w, b, \alpha_n)$ to get:

$$L(\alpha_n) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{m,n} \alpha_n \alpha_m y_n y_m x_n^T x_m \quad (30)$$

Equation 7 becomes:

$$L(\alpha_n) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{m,n} \alpha_n \alpha_m y_n y_m k(x_n, x_m) \quad (31)$$

$k(x_n, x_m) = x_n^T x_m$ is a Kernel function.

Equation 12 is solved by quadratic programming to get global optimal of α_n by maximizing $L(\alpha_n)$ as follows:

$$Maximise\ L(\alpha_n) = \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{m,n} \alpha_n \alpha_m y_n y_m x_n^T x_m$$

$$subject\ to\ \alpha_n \geq 0 \text{ for } n = 1, 2, \dots, N \text{ and } \sum_{n=1}^{N} \alpha_n y_n = 0 \quad (32)$$

5. EXPERIMENTAL STUDY

The experimental study of the proposed algorithm took place on a Brian System with Dual Core T5900 at 2.20 GHz processor, 2GB RAM and Window Vista 32-bit Operating System. MATLAB 2000b and Microsoft Access Database Management System featured as the frontend and backend engines respectively. The study was based on **s**hape feature and image texture datasets obtained from the selected species through collaboration between Fishery Departments of the Federal University of Technology, Akure (FUTA), Nigeria and Adekunle Ajasin University, Akungba-Akoko (AAUA), Nigeria. Six features; namely body length, anal fin length, caudal fin length, dorsal fin length, pelvic fin length and pectoral fin length (see Figure 3) were extracted. The fish texture dataset comprises of extracted texture from the two species**.**

Each fish image was in JPEG format of 20x20 pixels at 256 grey levels per pixel. Fish classification involves separating data into training and testing sets with each instance of the training set containing one target value (class label) and several attributes (the features). Based on the SVM, a model that is based on the training data is produced for predicting the target values of the test data when only the test data attributes is given. The testing set was used to assess the strength of the SVM classification model. The sequence of operations on the data required by SVM for classification includes data preparation (which ensured that two classes of data were collected), data conversion to SVM format and determination of the class of the input feature based on the algorithm presented in Figure 4.

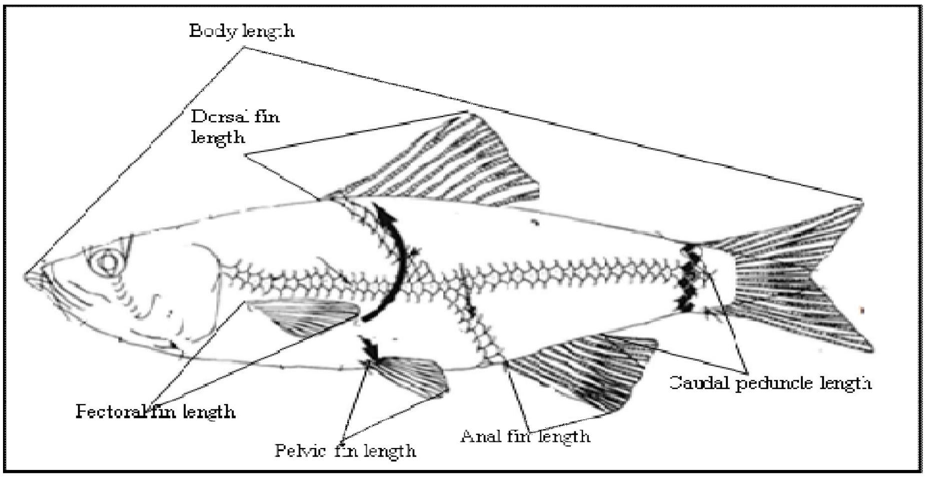

Figure 3: Extracted Feature Lentghs

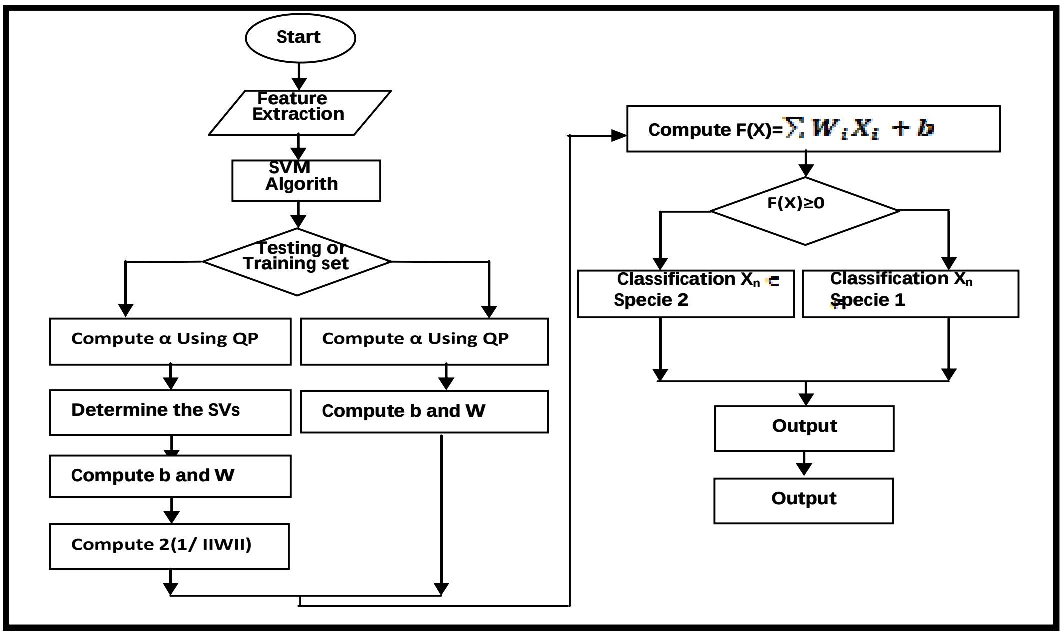

Figure 4: Flowchart of SVM Process

The summary of the parameters and the results for different algorithms with equal threshold is presented in Table 1. The use of these features is premised on their commonness and measurability. The system was trained using 76 fish consisting 38 Ethmalosa fimbriata and 38 Scomberomorous tritor. The Class Ethmalosa fimbriata was assigned 1 while the class Scomberomorous tritor was assigned 2. Seventy-four (74) fish with 37 Ethmalosa fimbriata and Scomberomorous tritor species apiece were used for testing. SVM method was based on a classifier with linear kernel due to the linearly separable nature of the data and a recognition accuracy of 78.59% was recorded. For Classification using neural network algorithm, the network was trained using back-propagation algorithm and a recognition accuracy of 60.01% was returned. The K-Nearest Neighbour (K-NN) classification algorithm used k=7 and a recognition accuracy of 52.69% was returned. The choice of k depends on the data and it must be odd number to avoid ties. Smaller k resulted in higher variance (less stable), while larger k resulted in higher bias (less precise).

Finally, the recognition accuracy for Classification using K-Means Clustering Algorithm was 50.97%. These results

African Journal of Computing & ICT

www.ajocict.net

indicate superior performance for SVM algorithm in the classification of fish species.

Table 1: Experimental results for Fish Classification Based on different methods

S/N	Parameters	Experiment			
		SVM	**K-NN**	**ANN**	**K-Means**
1	No. of Fish/ Observation	150	150	150	150
2	No. of training Set	76	76	76	76
3	No. of testing Set	74	74	74	74
4	No. of Fish Family	2	2	2	2
5	No. of input Features	6	6	6	6
6.	Recognition Accuracy	74.32%	52.69%	60.01%	50.97%

CONCLUSION

This paper contributed to knowledge by formulating an SVM-based fish classification algorithm. Six shape features; namely body length, anal fin length, caudal fin length, dorsal fin length, pelvic fin length and pectoral fin length were extracted from 150 fish (divided into 76 training and 74 testing sets) and the extraction formed the basis for the classification. The classification results exhibited the potential of the new algorithm for reliable and adequate fish classification and placed it at comparative advantage over some existing techniques such as ANN, K-NN and K-Means Clustering. However, the obtained recognition accuracy of 78.59% reveals there is still a lot of room for improvement. Future research therefore aims at improving the classification rate and performing fish classification on larger datasets and species

REFERENCES

[1] Benson B., Cho J., Gosorn D. and Kastner R. (2010). “Field Programmable Gate Array (FPGA) based fish detection using haar classifiers”. American Academy of Underwater Sciences, Atlanta, Georgia, USA.

[2] Bermejo S. (2007). “Fish age classification based on length, weight, sex and otolith morphological features”. Fish. Res. 84.

[3] Cabreira A. G., Tripode M. and Madirolas A. (2009). “Artificial neural networks for fish-species identification”. – ICES Journal of Marine Science, 66: 1119–1129.

[4] K.A. Mutasem, B.O. Khairuddin, N. Shahrulazman and A. Ibrahim (2010). “Fish Recognition Based on Robust Features Extraction from Size and Shape Measurements Using Neural Network”. Journal Computer Science, Volume 6, Issue 10

[5] X. H. Phoenix (2014). Balance-Guarantee Optimized Tree with Reject option for live fish recognition. Ph.D thesis, Institute pf Perception, Action and Behaviour, School of informatics, University of Edinburgh.

[6] G. L. Lawson, M. Barange and P. Freon (2001). “Species identification of pelagic fish schools on the South African continental shelf using acoustic descriptors and ancillary information”. ICES Journal of Marine Science, 58: 275–287.

[7] H. Hoang, K. Lock, A. Mouton and P.L.M. Goethals (2010). Application of classification trees and support vector machines to model the presence of macro invertebrates in rivers in Vietnam, Ecol. Inform. 140–146, http://dx.doi.org/10.1051/kmae/2011037, 4/17/2013 4:45PM.

[8] V.N. Vapnik (1995). The Nature of Statistical Learning Theory. Springer-Verlag, New York,

[9] Svellingen C., Totland B., White D. and Øvredal J. T. (2006). Automatic Species Recognition, Length Measurement and Weight Determination, using the CatchMeter Computer Vision System.

[10] Cabreira A. G., Tripode M. and Madirolas A. (2009). “Artificial neural networks for fish-species identification”. – ICES Journal of Marine Science, 66: 1119–1129.

[11] Hugo R., Paul B., Juan C. G., Jorge C. and Inmaculada P. (2009). “Acoustic identification of small pelagic fish species in Chile using support vector machines and neural networks”. Fisheries Research 102 (2010) 115–122. www.elsevier.com/locate/fishres

[12] Mutasem K.A., Khairuddin B.O., Shahrulazman N. and Ibrahim A. (2009). “Fish Recognition Based on the Combination Between Robust Features Selection, Image Segmentation and Geometrical Parameters Techniques using Artificial Neural Network and Decision Tree”. International Journal of Computer Science and Information Security, Volume 6, Issue 2.

[13] Matai J., Kastner R., Cutter G. R. and Demer D. A. (2010). Automated Techniques for Detection and Recognition of Fishes using Computer Vision Algorithms

[14] Rova A., Mori G. and Dill L.M. (2012). One Fish, Two Fish, Butterfish, Trumpeter: Recognizing Fish in Underwater Video

[15] Mutasem K.A., Khairuddin B.O., Shahrulazman N. and Ibrahim A. (2010a). “Fish Recognition Based on Features Extraction from Colour Texture using Back-Propagation Classifier”. In Journal of Theoretical and Applied Information Technology. 2005 - 2010 JATIT www.jatit.org, 7/18/2014 7:10PM.

[16] Phoenix X. H., Bastiaan J. B. and Robert B.F. (2012). Hierarchical Classification for Live Fish Recognition.

[17] D.J. Lee,, R. Schoenberger, D. Shiozawa, X. Xu and P. Zhan (2008). Contour matching for a fish recognition and migration monitoring system. Stud. Comput. Intell., 122: 183-207.

[18] R. Larsen, H. lafsdottir, and B. Ersbøll (2009). “Shape and texture based classification of fish species”. In

African Journal of Computing & ICT

www.ajocict.net

Proceedings of the Scandinavian Conference on Image Analysis, page 745–749.

[19] A. Rova, G. Mori, and L. M. Dill (2007). One fish, two fish, butterfish, trumpeter: Recognizing fish in underwater video. In IAPR Conference on Machine Vision Applications, pages 404–407.

[20] Marco T. A. Rodrigues, Flavio L. C. Padua, Rogerio M. Gomes, Gabriela E. Soares, Automatic Fish Species Classification Based on Robust Feature Extraction Techniques and Artificial Immune Systems

[21] M. P. Sampat, A. C. Bovik, J. K. Aggarwal, K. R. Castleman (2005), Supervised Parametric and Non-Parametric Classification of Chromosome Images, Elsevier Science

[22] Padraig Cunningham and Sarah Jane Delany (2007), k-Nearest Neighbour Classifiers, Technical Report UCD-CSI

[23] Ali Salem Bin Samma and Rosalina Abdul Salam (2009), Adaptation of K-Means Algorithm for Image Segmentation, World Academy of Science, Engineering and Technology, Vol. 50

[24] Soumya.D.S1,Arya.V (2013), Chromosome Segmentation Using K-Means Clustering, International Journal of scientific research and management, Volume 1, Issue 1, 51-54

[25] Massimiliano P. and Alessandro V. (1998). “Support Vector Machines for 3D Object Recognition”. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 6. June 1998