Autoencoder Based Classification of Three Popular Carps

A thesis submitted in partial fulfillment of the requirement for the

Degree of Master of Computer Application

Of

Jadavpur University

By

Payel Seth

Registration Number: 137317 of 2016-2017

Examination Roll Number: MCA196008

Under the Guidance of

Dr. Nibaran Das

Associate Professor

Department of Computer Science and Engineering

Jadavpur University, Kolkata - 700032

May 2019

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled **AUTOENCODER BASED CLASSIFICATION OF THREE POPULAR CARPS** has been satisfactorily completed by Payel Seth (University Registration No.: 137317 of 2016-17, Examination Roll No.: MCA196008). It is a bonafide piece of work carried out under my guidance and supervision and be accepted in partial fulfillment of the requirement for the Degree of Master of Computer Application, Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, Kolkata.

_____

Dr. Nibaran Das (Thesis Supervisor)
Associate Professor
Department of Computer Science and Engineering
Jadavpur University, Kolkata-700032

Countersigned

_____

Prof. Mahantapas Kundu
Head, Department of Computer Science and Engineering,
Jadavpur University, Kolkata-700032.

_____

Prof. Chiranjib Bhattacharjee
Dean, Faculty of Engineering and Technology,
Jadavpur University, Kolkata-700032.

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled **AUTOENCODER BASED CLASSIFICATION OF THREE POPULAR CARPS** is a bonafide record of work carried out by Payel Seth in partial fulfillment of the requirements for the award of the degree of Master of Computer Applications in the Department of Computer Science and Engineering, Jadavpur University during the period of January 2019 to May 2019. It is understood that by this approval the undersigned does not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

_____

Signature of Examiner :
Date :

_____

Signature of Supervisor :
Date :

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ETHICS

I hereby declare that this thesis entitles **AUTOENCODER BASED CLASSIFICATION OF THREE POPULAR CARPS** contains a literature survey and original research work by the undersigned candidate, as part of her Degree of Master of Computer Application.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Payel Seth
University Registration No. : 137317 of 2016 -17
Examination Roll No. : MCA196008

Thesis Title: AUTOENCODER BASED CLASSIFICATION OF THREE POPULAR CARPS.

Signature:

_____

Date:

_____

ACKNOWLEDGEMENT

First and foremost, I would like to start by thanking God Almighty for showering me with the strength, knowledge and potential to embark on this wonderful journey and to persevere and complete the embodied research work satisfactorily. I am pleased to express my deepest gratitude to my thesis guide, Dr. Nibaran Das, Department of Computer Science and Engineering, Jadavpur University, Kolkata for his invaluable guidance, constant encouragement and inspiration during the period of my dissertation.

I am highly indebted to **Jadavpur University** for providing me the opportunity and the required infrastructure to carry on my thesis. I am also grateful to the **Faculty of Fishery Sciences** for supplying the valuable data sets of this project work.

I am thankful to all the teaching and non-teaching staff whose helping hands have smoothed my journey through the period of my research.

Last but not the least; I would like to thank my family members, classmates, seniors (specially, ph.D students Arnab Banerjee, Swarnendu Ghosh and project fellow, Soumyajyoti Dey) and friends for giving me constant encouragement and mental support throughout my work.

Name: Payel Seth
University Registration No.: 137317 of 2016-17
Examination Roll No. : MCA196008
Master of Computer Application
Department of Computer Science and Engineering
Jadavpur University

TABLE OF CONTENTS

List of Tables

CHAPTER ONE

INTRODUCTION

The act of fish recognition is one of the frequently discussed areas nowadays. Fishes are very useful for mankind, as it consists of high protein and low fat, which provides various health benefits. Billions of people eat fish as the main ingredient of their daily meal. It also gives many useful products and helps in the economy development of a country. Many people can be employed in fishery sectors, and it also helps to grow income in many fields. For these benefits, we can see that the demand for fishes is continuously increasing. It also helps to get rid of many health problems, the problems of malnutrition, etc. and also provides economic balance to the nation.

Globally, if we see the report, more than 300 million people for their livelihoods, depend on fisheries and aquaculture on a daily basis and there are millions of people employed in this area. There are also many people who are engaged in marketing or processing works. Many poor people who directly depend on this fishing as a primary source of their income. In rural and as well as in urban areas, these fisheries are equally important [1].

Now, for a variety of reasons, fish recognition is really important, like for fishery, fishery biological research, and fishery independent stock assessment etc [2]. Also, perfect identification of fish species is necessary, the work is challenging and worthy.

Image recognition and object classification is the core of the task of classification of fishes. Image recognition is a step by step process of detecting and identifying an object or a feature in a digital image or video. This is used in many areas like for

security purpose, face recognition, etc. This act of recognition of images is basically based on their features. Some of the algorithms used for this task are HOG (Histogram oriented Gradient), SIFT (Scale-Invariant Feature Transform), SURF (Speeded Up Robust Features), PCA (Principal Component Analysis), etc.

Now, in India, Bangladesh, Nepal, Sri-Lanka, Nepal, the use of three carps, Rohu, Catla, and Mrigal is so common. As they belong to the same species, there would occur many problems to identify these fishes. In this study, we try to propose an algorithm which can classify these above mentioned popular carps. As this is an image classification based approach, feature extraction is very crucial part of it. In this study, the statistical and structural features have been used for feature extraction. Also, the autoencoders may be used to extract features. In the present work, the combination of statistical and structural features with autoencoder based features is used and the better accuracy of 81.9% is achieved than the individuals.

The remaining part of this article proceeds as follows. In the next chapter, we will discuss the theoretical background of this work. In chapter three, the methodology of our proposed work which deals with the data collection, feature extractions and classification is presented. Then in chapter four, we will discuss the experimental protocol and the result. Finally, we will conclude this article in chapter five.

CHAPTER TWO

THEORETICAL BACKGROUND

The art of detecting a particular specimen's species is one of the many applications that are done by image processing and machine learning. There are vast numbers of motivations for this work. This identification process is applicable to the plant, animal, or bacteria, etc. In many fields like in science, engineering, environmental, agricultural this is really worthy and helpful as well. There are a lot of works already have been done in this area.

Now, to perform fish species identification based on images, selecting suitable features are really important. The main motive of feature selection is:

1. Improving the accuracy of the prediction.
2. Providing more efficient and lower cost predictors.
3. To run the following methods successfully.

The result of feature extraction resembles the attributes of an image that any human vision system can identify. A lot of classification works has been done using pattern recognition and machine learning approaches. Now, we discuss all the work related to fish species identification that has been done before.

In [3], they discussed the necessity of extracting colour and shape based features. They have proposed a model which can perform scale and rotation invariant matching between any species of fishes. Then in the foreground/background separation step, a bounding rectangle selects a target object, and the object is processed. Thereafter, the target object is converted into a CSS (Curvature Scale Space) map. After that, CSS map is converted into a CV (circular vector) and then,

its representative vector based on the concept of force equilibrium is found out. This step was done to perform rotation invariant matching. After rotating the representative vector based on the concept of force equilibrium is found out. After performing a rotation of the representative vector into the canonical orientation, every unknown object can be compared with the model objects efficiently [4]. In Dudani [5], Zernike velocity moments are developed to describe an object using its shape and its motion through an image is claimed by Mercimekem [6]. In [4], a fish recognition using back-propagation classifier is introduced where robust features are extracted from colour, texture, etc. A fish classification recognition system using support vector machine is developed in [7], where the technique is based on the shape features of the fish. They measure the shape of the fishes by measuring the length in centimeter.

Autoencoders can also be used for feature extraction. The importance of autoencoder is increasing day by day as it has lots of benefits in image processing and deep learning. In [8], they have used stacked denoising autoencoder to extract feature as it is robust to the noise. They used logistic regression approach for the supervised fine-tuning and used ReLU as the activation function. They have shown that their method can achieve better accuracy than SVM (Support Vector Machine). A fishing activity detection from AIS data is introduced in [9]. Here the result of the autoencoders is compared with SVM and random forests. They show that autoencoder gives a better result than the other two.

Now, here in this study, we have used statistical and structural methods as well as autoencoders for feature extraction. Then, we combine the features to make a strong feature set and use the SVM classification algorithm. This leads to better recognition accuracy.

CHAPTER THREE

METHODOLOGY

There are several steps to perform this task. The steps are shown below in a simple diagram figure 1.

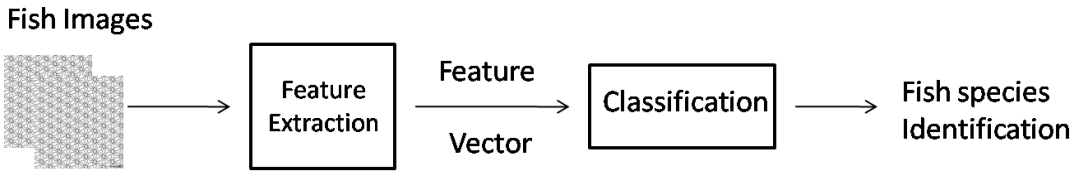

Figure 1: Algorithm of the proposed work.

In figure 1, we have shown how we can identify fish species based on the images. First, we need to prepare datasets for this task, which is the collection of fish images and preprocessing them. Then, feature extraction is performed. There are several algorithms for it. As a result, we get feature vectors. We pass these vectors to the classification step. Finally, after all these steps mentioned above, we can identify the species of fishes correctly.

3.1 Preparation of Datasets

Preparing dataset is the very first step. To move further, we need to perform this step. Below we discuss how we prepare the datasets.

3.1.1 Introduction to Raw Data

The raw data comprises of 209 Catla, 91 Mrigal and 170 Rohu fish items. The photos are captured by normal RGB cameras. Sometimes, the image of a particular fish is taken two or three times, that is, from above, from below side, etc. The photos are taken from the local vendors, fisheries and fish markets. Figure 2 shows the raw data of each category of fishes.

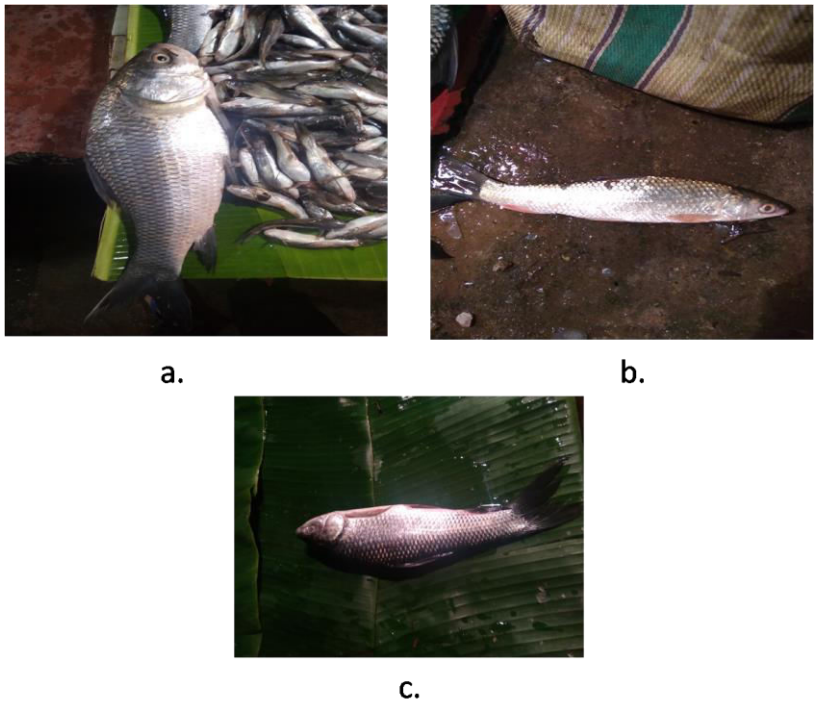

Figure 2: Examples of raw data of three categories of fishes.

a. Catla, b. Mrigal, c. Rohu .

By inspecting data, we find some issues with the images:

- The fishes of different categories look similar to some extent.

- As the images are taken by normal RGB cameras, it leads to a low-quality image.
- Different illumination effects.
- Sometimes shape and size differ from one category to another category of fishes.

3.1.2 Data preprocessing

The purpose of this stage is to convert the raw data (i.e. RGB camera images) to appropriate images such that we can apply feature extraction techniques of machine learning. The process can be summarised in three steps:

1. Segmentation of images.
2. Data augmentation.
3. Dividing the dataset for training, testing, and validation.

1) Segmentation of images:

Image segmentation is the operation of dividing a whole image into some connected sets of pixels. The segmentation is generally based on some measurements, which are texture, depth or motion based on the images. This is an initial and the foremost step for an image processing related tasks. Some applications of image segmentation are:

- For object-based measurements, identification of an object in an image.
- For identification of objects in a moving scene in a video.
- For identification of different objects which are at different distances from a sensor.

We have performed the segmentation of the images manually using the GIMP software. It leads to the following result:

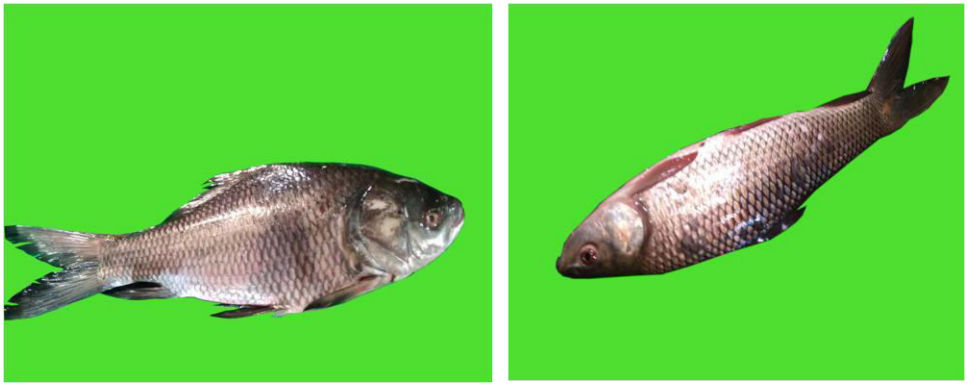

Figure 3: Example of Segmented pictures

2) Data augmentation:

In machine learning or deep learning, for the high-dimensional elements, data augmentation is the essential part. In this process, new samples of the original dataset are created by keeping the levels intact as the original one. The new samples are generated artificially. It is really helpful for generating more training data easily. Data augmentation is a frequently performed task in image processing and machine learning.

In this study, this substep is essential, as, the three categories of fishes belong to the same species, carps. So, the different category of fishes might have the same orientation, size or same category of fishes are different in shapes, etc., which gives a very poor result. To overcome these problems, we augment data in the actual data set.

Here, we have taken help of augmentor package of python. It is a data augmentation tool. In this tool, first, we need to create an empty pipeline. Then, we have to apply operations on the pipeline one by one in whatever order we want. These operations will be applied to each image of the dataset which passes through the pipeline. We have to mention the probability in each operation. We can also specify the freedom of movement of each operation. For example, we can specify the rotation operation can operate within the range of -30° to 30°. Then we will have to mention the sample of images we want to produce. Now every time an image passed through the

pipeline, the pipeline will produce different image data of that particular image. This stochastic approach allows for a potentially very large amount of images to be generated from even a small initial data set [10].

Main features of this tool are:

- Elastic distortions
- Perspective skewing
- Rotating
- Shearing
- Mirroring
- Cropping

In this paper, to perform the augmentation with the help of augmentor tool, after creating the pipeline, we have performed five operations on each image, namely, horizontal flip flop, vertical flip flop, rotate 90◦ and 270◦ and resizing an image. This is how we create 300 samples of each category of fishes and add those data to the original data.

2) Dividing the dataset for training, testing and validation :

First, we resize each image in shape $64 \times 64 \times 3$. Then, we randomly shuffle the whole dataset. We performed this as shuffling data helps to reduce variance and for this, the models overfit less and remain general. Then, we divide the whole dataset into 50% training and 50% test data. After that, for validation, again, we divide the training dataset into a 4:1 ratio.

3.2 Feature extraction

The feature of an object are the function of one or more measurements, where the measurements measurements are some quantifiable properties of that object and denote to some important characteristics of that object.

The features can be classified into:

1. **General feature**: These are basically independent features like color, texture, and shape. This can be further divided into the following:

- Local features: Based on the subdivision of the image band of an image segmentation or edge detection, features are measured.
- Global features: Features are detected over the whole image or just regular sub-area of an image is considered.
- Pixel-level features: At each pixel, features are calculated, for example, colour feature, etc.

2. **Domain-specific feature**: The features which depend on the applications. For example, human faces, fingerprints, etc.

Apart from that, all features can be classified into low high-level features and low-level features. We can extract low-level features directly from the original images and the high-level feature extraction depends on low level features [11].

Now, the objective of feature extraction is to find significant properties of the image such that it can be distinguished from one image to another image. These extracted features should have the following criteria:

- The features should contain enough information about the image and may not require the domain-specific knowledge for their extraction.
- In case, there is a large image dataset, then the features should be computed easily.
- The features should match with the characteristics of the human detected features of an image [11].

3.2.1 Structural and Statistical Feature Extraction methods

Here, in this study, for these three carps, we have this following distinguished features in table 1.

Table 1: Distinguished features of three carps

Features	Rohu	Catla	Mrigal
Head	Moderate	Big	Pointed than Rohu
Body Shape	Streamline	Body Depth more(less than common carps)	Slender
Mouth	Middle	Upper	Lower
Eye position	Moderate distance from mouth tip. Middle position	Moderate distance from mouth tip. Middle position	Close to mouth tip.slide upper side
Body colour	Upper side dark golden and lower side whitish	Upper side black and ventral side whitish	Whitish
Fin Colour	Blackish	blackish	Reddish
Fin position	Dorsal fin moderate length started from the middle of body length	Dorsal fin slight long started from the slight front	Dorsal fin slight short started from middle body length
Scale	Medium size	Medium size	Medium size

Analyzing table 1, we extract three structural features, colour, shape, texture and one statistical feature, which is, histogram oriented gradient (HOG). We will discuss them in brief below:

Colour Feature- This features is used in most of the cases in image processing and classification. Colour feature has some benefits. Those are:

- Effectiveness: The main image and the extracted colour matching image have many similarities between them.
- Robustness: If rotated or scaled, in any condition, the colour histogram changes a very little.

- Simplicity in implementation: The process is really simple, it just takes some steps. First, we have to scan the image, then to the resolution of the histogram, colours are assigned and lastly, we need to build the histogram using color components.
- Computational simplicity: The complexity of the histogram computation is O(x,y) where the size of the images is x × y. The complexity for a single image is O(n), where n is the number of different colours or the resolution of the histogram.
- Low storage requirements: It requires lower storage than the actual image, assuming color quantization.

Shape Features- It is one of the important visual features. The shape content descriptor cannot be defined by measuring the similarity between two images as it is difficult to do. So, it is done another way. There are mainly two steps to perform this task, extracting features, and measuring the similarity between extracted features. Shape descriptor is of two types: region based and contour based. The region based use the whole area of an object for shape description and the other use local features as boundary segments.

Texture Features- This feature is another essential property of the image. For the retrieval related problems, a texture feature is a potent tool. The similarity matching between the images cannot be performed using it but it can classify textured images from non-textured ones. If we combined this feature with other features like shape and colour, we could get better results. Texture features are

- Statistical measures
  - Entropy
  - Homogeneity
  - Contrast
- Wavelets
- Fractals

[11]

HOG feature- HOG feature is histogram oriented feature. It is basically an edge orientation histograms based on the orientation of the gradient in the localized region that is called cells. Therefore, the rough shape of the object can be easier to express, and also, it is robust to illumination changes and variations in geometry. Again, it does not support rotation and scale changes.

So, we will use the following descriptors for feature extraction:

- Colour using Color Histogram
- Shape using Hu Moments
- Texture using Haralick feature
- Others using HOG feature

Then, combine those features, Which leads to the following result.

Two Features Combination :

- Colour Histogram and Haralick feature
- Hog and Colour feature
- Hog and Haralick feature
- Hog and Hu Moments
- Hu Moments and Colour Histogram
- Hu Moments and Haralick feature

Three Features Combination :

- Hog, Colour Histogram, and Haralick feature
- Hog, Hu Moments and Colour feature
- Hog, Hu Moments and Haralick feature
- Hu Moments, Colour Histogram and Haralick feature

Four Features Combination :

- Hog feature, Hu Moments, Colour Histogram and Haralick feature

3.2.2 Feature Extraction Using Autoencoder

First, we will discuss the autoencoder in brief, and then we will discuss the feature extracting method using it.

Autoencoder is one type of artificial neural network. It is trained to produce output as same as the input, that is, it tries to reconstruct the input in its output, so the label of the original one remains the same. That is why it is unsupervised learning method. The main feature of this network is that it is used in dimensionality reduction by training to ignore signal noise. It has three layers, and the middle layer is smaller than the other two. It can be used in many applications like for feature extraction, etc. For image recognition, stacked sparse autoencoder or convolutional autoencoders, etc. are used in many cases.

The functions of each layer are different. The very first layer, where we pass the input, learns to encode easy features of the inputs. The second layer learns to encode less local features by analyzing the output of the first layer. The final output layer tries to reconstruct back its input. This autoencoder can be used for the generative model also. For example, if we make the system learn with the pictures of fish and flying both images, then it can produce an image of flying fish in its output, even if it is not possible in reality or the autoencoder was not fed with this input before.

The architecture of the autoencoder – Architecturally, the simplest form of the autoencoder is something like multiplayer perceptron (MLP). It has one input layer, one or more hidden layers, and one output layer where the output layer has the same number of nodes as the input layer. The autoencoder tries to reconstruct the input in its output in an unsupervised learning method.

The encoder and decoder parts of the autoencoder can be defined using α and β such that:

$\alpha: \boldsymbol{I} \rightarrow \boldsymbol{F}$ ..... Equation 1

$\beta: \boldsymbol{F} \rightarrow \boldsymbol{I}$..... Equation 2

$\alpha, \beta = \arg\min_{\alpha,\beta} \|\boldsymbol{I} - (\beta \circ \alpha)\boldsymbol{I}\|^2$..... Equation 3

In the simplest form, an autoencoder has only one hidden layer and if the input is x and the output is y, then,

$\mathbf{y} = \gamma(\mathbf{W}\mathbf{x} + \mathbf{b})$..... Equation 4

where, $\mathbf{x} \in R^d = \boldsymbol{I}$, which maps it to $\mathbf{y} \in R^p = \boldsymbol{F}$, $\mathbf{W}$ is a weight matrix**,** $\mathbf{b}$ is a bias and γ is an element-wise activation function.

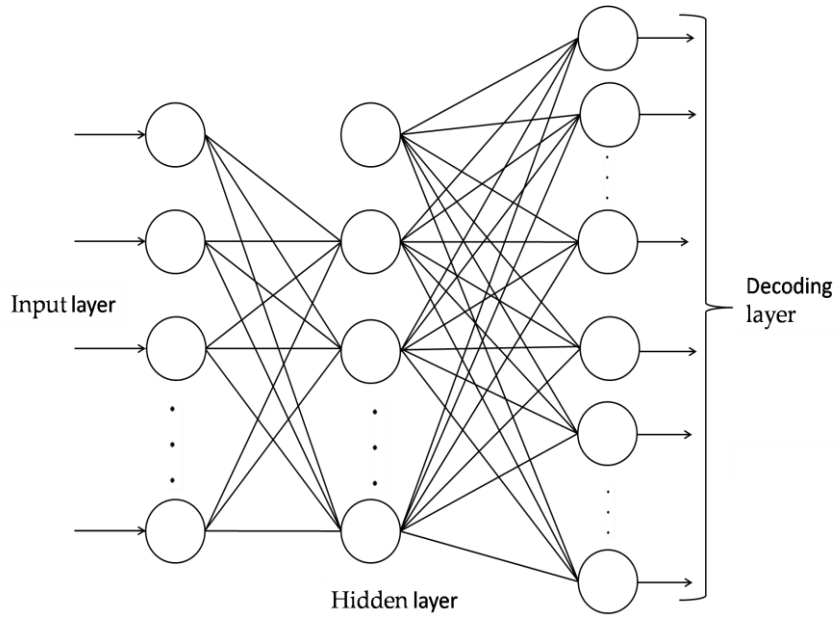

Figure 4: General structure of the autoencoder.

After that, in the decoder portion of the autoencoder, it tries to map $\mathbf{y}$ to $\mathbf{x}^j$ such that:

$\mathbf{x}^j = \gamma^j(\mathbf{W}^j\mathbf{y} + \mathbf{b}^j)$..... Equation 5

where, $\mathbf{x}^j$ is the reconstruction, having the same shape as that of input $\mathbf{x}$. γ^j, $\mathbf{W}^j$, and $\mathbf{b}^j$ for the decoder may differ in general from the corresponding γ, $\mathbf{W}$, and $\mathbf{b}$ for the encoder, depending on the design of the autoencoder.

Autoencoders are also trained to minimize reconstruction errors (such as squared errors), which is called loss:

$$L(\mathbf{x}, \mathbf{x}^j) = \|\mathbf{x} - \mathbf{x}^j\|^2 = \|\mathbf{x} - \gamma^j(\mathbf{W}^j(\gamma(\mathbf{W}\mathbf{x} + \mathbf{b})) + \mathbf{b}^j)\|^2 \quad \text{Equation 6}$$

where **x** is usually averaged over some input training set [8].

Variations of Autoencoder - There exists a number of varieties of an autoencoder. We discuss them below one by one:

Simple Autoencoder - This is the simplest form of the autoencoder. It is also called as vanilla autoencoder. In this form, the autoencoder has three layers. That is one input, one output, and only one hidden layer. The input and outputs are the same. We use the following architecture of the simple autoencoder described in table 2.

Table 2: Architecture of simple autoencoder

Layer (type)	Output Shape	Parameter number
Input Layer	(None, 12288)	0
Dense	(None, 512)	6291968
Dense	(None, 12288)	6303744
Total params: 12,595,712		
Trainable params: 12,595,7112		
Non-trainable params: 0		

Convolutional Autoencoder - This is a type of Convolutional Neural Networks (CNN). CAE is mainly used where the inputs are of image type. This type of autoencoder gives satisfactory result for images. The CAE first uses several convolutions and pooling layers to transform the input into a high dimensional feature map representation and then reconstructs the input using strided transposed convolutions. CAEs are general purpose feature extractors which are different from autoencoders that ignore entirely the 2D image structure. The summary of the convolutional autoencoder is shown in table 3.

Table 3: Architecture of convolutional autoencoder

Layer (type)	Output Shape	Parameter number
Input Layer	(None, 64, 64, 3)	0
Conv2D	(None, 64, 64, 32)	896
MaxPooling	(None, 32, 32, 32)	0
Conv2D	(None, 32, 32, 64)	18496
MaxPooling	(None, 16, 16, 64)	0
Conv2D	(None, 16, 16, 64)	36928
MaxPooling	(None, 8, 8, 64)	0
Conv2D	(None, 8, 8, 64)	36928
UnSampling	(None, 16, 16, 64)	0
Conv2D	(None, 16, 16, 64)	36928
UnSampling	(None, 32, 32, 64)	0
Conv2D	(None, 32, 32, 32)	18464
UnSampling	(None, 64, 64, 64)	0
Conv2D	(None, 64, 64, 3)	867
Total params: 914,507		
Trainable params: 149,507		
Non-trainable params: 0		

Denoising autoencoder - This type of autoencoder is trained with the partially corrupted input, and it can recover the original undistorted input. This was introduced with a specific approach to good representation. Proper representation can be obtained from a corrupted input and will be useful for recovering the corresponding clean input.

To train an autoencoder to denoise data, fist the stochastic mapping $\mathbf{x} \rightarrow \tilde{\mathbf{x}}$ is performed, in order to corrupt the input data. Then this $\tilde{\mathbf{x}}$ is passed to the normal autoencoder as input. Here, the loss should be $\mathbf{L}(\mathbf{x}, \tilde{\mathbf{x}}')$.

Here, we will train the autoencoder to map noisy digits images to clean digits images. We generate synthetic noisy digits by applying a Gaussian noise matrix and clip the images between 0 and 1 [12]. Here, we add noise factor of 0.3. Examples of noisy images are shown in figure 7.

Figure 5: Example of noisy input images.

For this autoencoder, we will use convolutional autoencoder of the same architecture as described in table 3.

Sparse Autoencoder - When the first autoencoder was invented in the 1980s, it was difficult to train initially as the encoding had to compete to set the same small set of bits. This problem was resolved by introducing sparse autoencoder. In SAE, there is a more number of hidden layers than inputs. However, all the hidden layers are not active at the same time, but only a few stay active. During the training period, the sparsity can be achieved by comparing the probability distribution of the hidden unit activations with the low desired value. Else if we manually do zeros all except the few strongest hidden unit activations (referred to as a k-sparse autoencoder), it can also be performed.

The architecture of the network is the same that of the simple autoencoder.

Deep Autoencoder - A deep autoencoder consists of two symmetrical networks. The first network is the encoding part which has four or five shallow layers and the second half consists the four or five layers which are known as the decoder part. Restricted Boltzmann Machine (RBM) is the basic building block of the deep belief network [13]. The final encoding layer is compact and fast.

The summary of the autoencoder is shown in table 4.

Table 4: Architecture of deep autoencoder

Layer (type)	Output Shape	Parameter number
Input Layer	(None, 12288)	0
Dense	(None, 512)	6291968
Dense	(None, 256)	131328
Dense	(None, 512)	131584
Dense	(None, 12288)	6303744
Total params: 12,858,624		
Trainable params: 12,858,624		
Non-trainable params: 0		

Feature extraction algorithm using autoencoder[14] - The autoencoders can be used for feature extraction. For this, let **x** be the training input, **y** be the training labels, **h** be the number of hidden layers and **f** be the number of extracted features. Then the method of extracting features is:

1. First based on the inputs the hidden layers **h** are pre-trained such that it contains **f** neurons.
2. The deep network is then fine-tuned according to labels **y** with back-propagation and optimizer.
3. Extract the **f** node values in the middle hidden layers.

Then, the features got from various autoencoders are combined using permutation. Hence we get the following combinations of autoencoder's extracted feature.

Two Combinations -

- Simple and Deep autoencoder
- Simple and Convolutional autoencoder
- Simple and Denoising autoencoder
- Simple and Sparse autoencoder
- Deep and Convolutional autoencoder
- Deep and Denoising autoencoder

- Deep and Sparse autoencoder
- Convolutional and Denoising autoencoder
- Convolutional and Sparse autoencoder
- Denoising and Sparse autoencoder

Three Combinations -

- Simple, Deep and Convolutional autoencoder
- Simple, Deep and Denoising autoencoder
- Simple, Deep and Sparse autoencoder
- Deep, Convolutional and Denoising autoencoder
- Deep, Convolutional and Sparse autoencoder
- Convolutional, Denoising and Sparse autoencoder
- Convolutional, Denoising and Simple autoencoder
- Denoising, Sparse and Simple autoencoder
- Denoising, Sparse and Deep autoencoder

Four Combinations -

- Simple, Deep, Convolutional and Denoising autoencoder
- Simple, Deep, Convolutional and Sparse autoencoder
- Simple, Convolutional, denoising and Sparse autoencoder
- Simple, Deep, Denoising and Sparse autoencoder
- Deep, Convolutional, Denoising, and Sparse autoencoder

Five Combinations -

- Simple, Convolutional, Deep, Denoising and Sparse autoencoder

3.2.3 Combining Structural, Statistical and Autoencoder extracted features -

Here, we combine the structural, statistical and autoencoder extracted features to check whether this gives a better accuracy or not. Then we will use all these features for classification.

3.3 Classification

Classification is a technique which learns to categorize the given dataset into a desired and distinct number of classes in a supervised manner such that some labels can be assigned to each class. There are two types of classifiers:

- Binary classifiers: Classification with only 2 distinct classes or with 2 possible outcomes.
- Multi-Class classifiers: Classification with more than two distinct classes.

In this study, we classify using the SVM method, as it can be used for both classification and regression. This algorithm can determine the best decision boundary between vectors that belong to a given group (or category) and vectors that do not belong to it. It is applicable to any kind of vectors which encode any kind of data.

Support vector machine basically represents the space or gap between different categories of the training points in the space. Then the new data points are mapped to the same space and determine in which category they belong based on which side of the gap they fall.

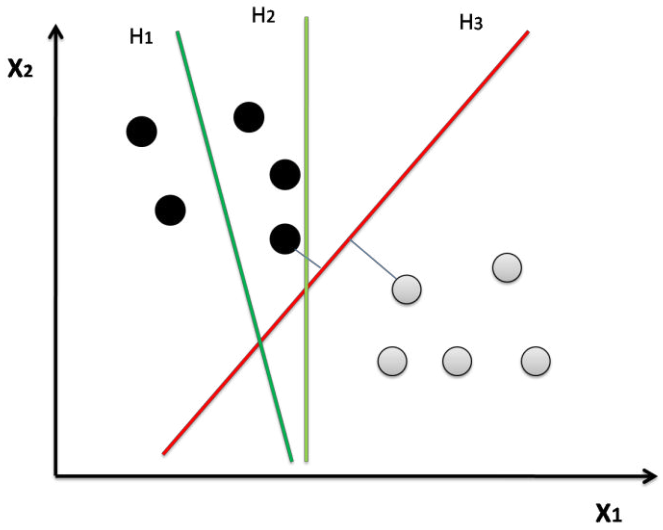

Figure 6: Diagram of SVM.

Depending on the category labels are assigned to them. The advantage of SVM is that it is memory efficient as it is useful in high dimensional spaces and in decision function, it uses a subset of training points. Disadvantages of this algorithm are it does not directly provide probability estimates, these are calculated using a five-fold cross-validation, which is very expensive. There are three main parameters of the SVM classifier:

- Type of kernel
- Gamma value
- C value

In figure 6, H2 separates the classes but only a few margins and H1 does not. They are separated by H3 with the maximal margin [15].

CHAPTER FOUR

EXPERIMENTAL PROTOCOL & RESULT

We have now 509 Catla, 391 Mrigal and 470 Rohu fishes and each image of fishes is of shape 64 × 64 × 3. We have used Keras [16] library in Python for the implementation of the autoencoder. We use the RMSprop optimizer implemented in Keras. The optimizer uses a learning rate of 0.01, a momentum of 0.001, and a batch size of 128 is used to train the autoencoder. For the classification algorithm, we utilized the scikit-learn library [17] of Python. The parameters that we have used for SVM (Support Vector Machine) are : C = 1.0, cache size = 200, degree = 3, gamma = auto, kernel = linear, max iter = −1, tol = 0.001, class weight = None.

First, after extracting structural (colour, shape, and texture) and statistical (HOG) features, we applied SVM, and we got the result in table 5.

Table 5: Structural and statistical feature extraction

Features	Accuracy	Precision	Recall	F1 score
HOG	65.4	66.0	65.1	65.4
Hu Moments	64.5	66.9	65.9	57.9
Colour Histogram	71.6	73.5	72.9	73.2
Haralick Feature	68.6	69.4	69.9	69.6

Now, we combine these features and apply SVM. The results are as follow :

Table 6: Two combinations of structural and statistical method

Combined features	Accuracy	Precision	Recall	F1 score
Colour Histogram and Haralick feature	71.3	72.3	72.9	72.6
HOG and Colour Histogram	73.8	74.7	75.0	74.8
HOG and Haralick feature	69.9	70.9	70.1	70.4
HOG and Hu Moments	62.6	63.1	62.8	62.8
Hu Moments and Colour Histogram	74.0	75.6	75.2	75.4
Hu Moments and Haralick Feature	72.4	73.2	73.3	72.4

Table 7: Three combinations of structural and statistical method

Combined features	Accuracy	Precision	Recall	F1 score
HOG, Colour Histogram and Haralick feature	76.4	77.2	77.5	77.2
HOG, Hu Moments and Colour Histogram	74.7	75.6	75.5	75.5
HOG, Hu Moments and Haralick feature	70.2	70.7	70.4	70.5
Hu Moments, Colour Histogram and Haralick Feature	72.5	74.1	73.9	74.0

Table 8: Four combinations of structural and statistical method

Combined features	Accuracy	Precision	Recall	F1 score
Hu Moments, HOG, Colour Histogram and Haralick Feature	78.2	78.6	79.4	78.9

Now, we extract features using five autoencoders, namely, simple, convolutional, deep, sparse and denoising and apply SVM, we got the following result in table 9 :

Table 9: Features Extraction using Autoencoder

Autoencoder type	Accuracy	Precision	Recall	F1 score
Simple	68.75	67.9	70.25	68.3
Deep	64.96	64.69	66.1	60.68
Convolutional	75.0	75.91	75.96	75.93
Denoising	76	76.2	76.79	76.5
Sparse	61.4	60.4	62.8	60.9

The, we combine these features using permutations. We will show those results in table 10, 11, 12, 13.

Table 10: Two combinations of autoencoder extracted features

Combined Autoencoder type	Accuracy	Precision	Recall	F1 score
Simple and Deep	72.9	73.0	73.9	72.6
Simple and Convolutional	75.3	76.3	76.4	76.3
Simple and Denoising	73.5	74.4	74.2	74.3
Simple and Sparse	68.0	67.3	69.2	67.6
Deep and Convolutional	74.7	76.0	75.7	75.8
Deep and Denoising	75.4	75.9	76.3	76.1
Deep and Sparse	61.0	61.4	62.3	61.6
Convolutional and Denoising	76.2	76.7	77.3	76.9
Convolutional and Sparse	70.3	71.5	71.4	73.4
Denoising and Sparse	71.6	72.4	72.6	72.5

Table 11: Three Combinations of Autoencoder extracted features

Combined Autoencoder Type	Accuracy	Precision	Recall	F1 score
Simple, Deep and Convolutional	73.5	75.2	74.7	75.0
Simple, Deep and Denoising	74.1	75.6	75.0	75.3
Simple, Deep and Sparse	66.8	67.3	68.0	67.6
Deep, Convolutional and Denoising	75.1	75.7	76.3	75.9
Deep, Convolutional and Sparse	70.8	71.7	71.8	71.7
Convolutional, Denoising and Sparse	76.4	76.9	77.5	77.1
Convolutional, Denoising and SImple	75.3	75.8	76.4	76.1
Denoising, Sparse and Simple	71.5	71.8	72.5	72.1
Denoising, Sparse and Deep	74.0	74.3	75.1	74.6
Simple, Denoising and Convolutional	76.2	77.2	77.1	77.2

Table 12: Four Combinations of Autoencoder extracted features

Combined Autoencoder Type	Accuracy	Precision	Recall	F1 score
Simple, Deep, Convolutional and Denoising	73.4	74.3	74.8	74.5
Simple, Deep, Convolutional and Sparse	74.8	75.7	75.9	75.7
Simple, Convolutional, Denoising and Sparse	74.4	75.8	75.2	75.5
Simple, Deep, Denoising and Sparse	74.5	75.3	75.2	75.0
Deep, Convolutional, Denoising and Sparse	75.7	77.0	76.6	76.6

Table 13: Five combinations of autoencoder extracted features

Combined Autoencoder Type	Accuracy	Precision	Recall	F1 score
Simple, Deep, Convolutional, Sparse and Denoising	74.0	74.8	74.7	74.7

Now, in table 9, we have seen that denoising autoencoder gives the best result. So, we combine the features extracted by structural and statistical methods and the extracted features from denoising autoencoder. Thereafter, classification is done using SVM, and we get the following results in tables 14, 15, 16, 17.

Table 14: Combined features Accuracy

General Feature type	Accuracy	Precision	Recall	F1 score
HOG	75.7	77.1	76.1	76.2
Hu Moments	76.3	76.8	77.2	76.9
Colour Histogram	78.5	79.0	79.3	79.0
Haralick Feature	73.5	74.7	74.1	74.3

Table 15: Two combinations of combined features

Combined general feature type	Accuracy	Precision	Recall	F1 score
Colour Histogram and Haralick feature	72.5	74.2	73.3	73.6
HOG and Colour Histogram	77.0	78.1	77.8	77.9
HOG and Haralick feature	76.2	76.8	76.8	76.8
HOG and Hu Moments	76.4	76.6	77.0	76.7
Hu Moments and Colour Histogram	77.0	77.4	77.8	77.4
Hu Moments and Haralick Feature	75.1	76.4	75.7	76.0

Table 16: Three combinations of combined features

Combined general feature type	Accuracy	Precision	Recall	F1 score
HOG, Colour Histogram, and Haralick feature	81.9	82.1	82.3	82.1
HOG, Hu Moments and Colour Histogram	78.6	79.0	79.5	79.2
HOG, Hu Moments and Haralick feature	79.4	80.1	80.3	80.2
Hu Moments, Colour Histogram and Haralick Feature	77.0	77.8	78.1	77.9

Table 17: Four combinations of combined features

Combined features	Accuracy	Precision	Recall	F1 score
Hu Moments, HOG, Colour Histogram, and Haralick Feature	75.5	75.8	76.3	75.8

Now, if we plot the above accuracies in a graph, we have the following result:

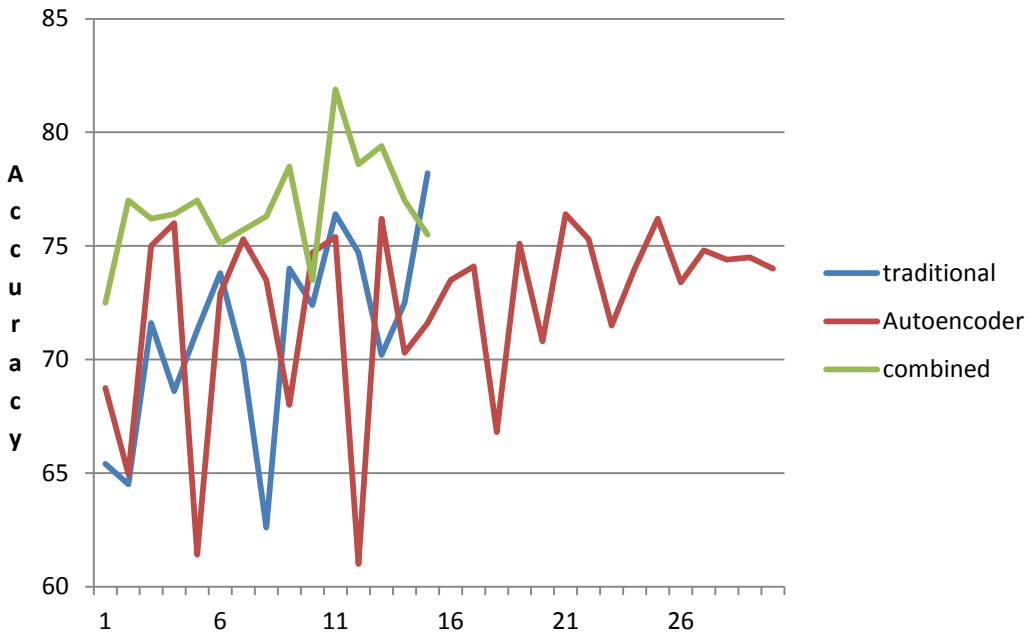

Figure 7 : Graph of three features based classification accuracy

The above chart shows that the combined method of feature extraction gives a better result than the other two. From table 16, we can see that the combination of HOG feature, Colour Histogram and Haralick feature and the feature extracted from denoising autoencoder gives accuracy the highest, which is 81.9%.

Now, to check whether our experimental protocol is correct we performed the three cross-validation on the dataset. For this, we have used the same architecture of the denoising autoencoder as mentioned in methodology.

We get the result as follows :

Table 18: Three fold cross validation result of our best performing model

Accuracy	Precision	Recall	F1 score
64.8	66.7	65.3	64.9

So, from the above table, we can see that it gives poor result than our proposed method. Hence, it proves that our protocol of performing classification task is better.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

In this study, we have proposed a method for classification for three popular carps - Rohu, Catla and Mrigal. For this, we have discussed both the structural and statistical method of feature extractions and autoencoders based feature extraction. We observed that combining both the method gives better accuracy than the other two.

This work has some limitations. First, we have tested on only three types of fishes. Second, we have not investigated the effect of using different activation functions and the inclusion of the concept of dropout during training. In the future, we plan to investigate these issues and try to get more accuracies.

REFERENCES

[1] "The Importance of Fish | WorldFish Organization." [Online]. Available: https://www.worldfishcenter.org/why-fish.

[2] R. Larsen, H. Olafsdottir, and B. K. Ersbøll, "Shape and Texture Based Classification of Fish Species," pp. 745–749, 2009.

[3] C.-J. Sze, H. Tyan, H. Liao, C.-S. Lu, and S.-K. Huang, "Shape-based Retrieval on a Fish Database of Taiwan," vol. 2, 1999.

[4] M. K. Alsmadi, K. B. I. N. Omar, S. A. Noah, and I. Almarashdeh, "Fish recognition based on robust features extraction from color texture measurements using back-propagation classifier," 2010.

[5] S. A. Dudani, K. J. Breeding, and R. B. McGhee, "Aircraft identification by moment invariants," *IEEE Trans. Comput.*, vol. 100, no. 1, pp. 39–46, 1977.

[6] M. Mercimek, K. Gulez, and T. V. Mumcu, "Real object recognition using moment invariants," *Sadhana - Acad. Proc. Eng. Sci.*, vol. 30, no. 6, pp. 765–775, 2005.

[7] S. O. Ogunlana, O. Olabode, S. A. A. Oluwadare, and G. B. Iwasokun, "Fish Classification Using Support Vector Machine," *African J. Comput. ICT Afr J. Comp ICTs*, vol. 8, no. 2, pp. 75–82, 2015.

[8] C. Xing, L. Ma, and X. Yang, "Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images," *J. Sensors*, vol. 2016, 2016.

[9] X. Jiang, D. L. Silver, B. Hu, and E. N. De Souza, "Fishing Activity Detection from AIS Data Using Autoencoders from AIS Data Using Autoencoders," no. May 2018, 2016.

[10] M. D. Bloice, C. Stocker, and A. Holzinger, "Augmentor: An Image Augmentation Library for Machine Learning," pp. 1–5, 2017.

[11] G. Caridakis, R. S. Choras, and T. Arif, "Chapter 6: feature extraction," pp. 107–121, 2008.

[12] "Building Autoencoders in Keras." [Online]. Available: https://blog.keras.io/building-autoencoders-in-keras.html. [Accessed: 16-May-2019].

[13] "Deep Learning — Different Types of Autoencoders – Data Driven Investor – Medium." [Online]. Available: https://medium.com/datadriveninvestor/deep-learning-different-types-of-autoencoders-41d4fa5f7570.

[14] R. Li, P. Wang, and Z. Chen, “A Feature Extraction Method Based on Stacked Auto-Encoder for Telecom Churn Prediction A Feature Extraction Method Based on Stacked Auto-Encoder for Telecom Churn Prediction,” no. October, 2016.
[15] Wikipedia contributors, “Support-vector machine.” 2019.
[16] F. Chollet and others, “Keras.” GitHub, 2015.
[17] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos Pedregosa, Varoquaux, Gramfort Et Al. Matthieu Perrot,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.