ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2022.0386

FRESHWATER FISH SPECIES CLASSIFICATION USING DEEP CNN FEATURES

Jayashree Deka[1], Shakuntala Laskar[2] and Bikramaditya Baklial[3]
[1]Department of Electrical and Electronics Engineering, Assam Don Bosco University, India
[2,3]Department of Zoology, Bahona College, India

Abstract

Deep-Learning and image processing have shown excellent performance in automated fish image classification and recognition task in recent years. In this research paper, we have come up with a novel deep-learning method based on CNN features extracted from deeper layer of a pretrained CNN architecture for automatic classification of eleven (11) indigenous fresh water fish species from India. We have utilized top three layers of a pretrained Resnet-50 model to extract features from fish images and an "ones for all SVM" classifier to train and test images based on the CNN features. This paper reports an exceptional result in overall classification performance on Fish-Pak dataset and on our own dataset. The proposed framework yields overall classification accuracy, precision and recall of 100% on our own data and a maximum of 98.74% accuracy on Fish-Pak dataset which is the best till date.

Keywords:

Automatic Fish Detection, Fish Classification, Fish Species Recognition, Fish Database, Feature Extraction

1. INTRODUCTION

Detection and classification of fish species are important for the assessment of fish abundance, distribution and diversity in aquaculture, commercial fisheries and marine protected areas. Apart from these, it is also important to correctly identify edible fish from wild, poisonous fish, as consuming the wrong fish can lead to serious health issues. The traditional manual method of identifications/classifications are tedious, inaccurate and destructive as it involves net casting and catching. Also, manual identification of fish species needs appropriate taxonomic knowledge of fish, so a lack of subject expertise may cause potential mistakes. Contrary to these, automatic classification is faster, accurate, efficient and does not cause any physical harm to the fish. Some of the other techniques include use of otoliths, scales, echo-sounders with multiple discreet frequencies (acoustic method), DNA barcodes, internal and external tags. However, these methods are labour-intensive and exorbitant.

In recent years, image processing and deep-learning methods in computer vision have shown dramatic improvement in fish classification task in partially or fully uncontrolled environments [1]-[5]. Image based fish classification involves extraction of shape [1]-[3] [9] [11], texture [3] [4] [7] [9] or colour features [1] [4] [6] [7]. Different low-level hand-crafted methods like PCA [10], SIFT [8] [10] [45], HAAR [6], and SURF [8] features have been applied in recent years which require domain knowledge for effective feature extraction. However, these short comings are easily overcome by the use of deep network that learns feature for multi-level representation of data. Deep learning-based fish classification techniques integrate fundamental image processing steps where learning of image features and classification is possible without any manual intervention [14]-[20]. Yet these methods demand huge datasets and involves enormous computational cost while the network from scratch. However, this too can be taken care of by the use of already pre-trained deep learning models. Pre-trained CNNs models are good feature extractors and a new classifier can be used to classify objects based on those extracted features. This approach has been successfully implemented for freshwater fish classification by using pre-trained ResNet-150[15], ResNet-152[16], Alex-Net [14] [25], VGG-16 [14] [22] and SPP-Dense-net [26].

We have observed that previous works showed better classification result when species under investigation are fewer. The problem arises when the recognition model has to deal with multiclass fish species with only few data available per species. While India has variety of fish species with different names in different localities, till date we can't trace any significant research work that dealt with automatization of Indian fish species. Only, Mattew et al. [12] attempted to classify Sardinella Longiceps, Rastrelliger Kamasutra, Ethynes Affinis and Stolephorus Indicus which are native to Kerala, India. These are different varieties of Tuna. In another work reported by Jose et al. [13], three classes of Tuna species are classified using ensemble methods. But, all of these works studied only marine fish. As per FishBase, India has 1,035 varieties of fresh water fish and many of us, do not even know all of these fish names. Industrialization of fresh water fish is yet to be taken seriously by all the stake holders of fisheries department in India. Therefore, an image based affordable recognition model of small indigenous fresh water fish can give a boost to this un-organised sector of fisheries in our country (India).

Taking this into account, we use deep-learning methods for automatic classification of 11 indigenous fresh-water fish species that are easily available and popular in Assam, India, as a cheap source of protein. The correct and automated fish classification can help the fish farmers in sorting harvested fish according to species which promotes better marketing for polyculture fish farming. Also, by proper identification of fish species, farmers can plan feeding strategies properly which yields better productivity giving a boost to the local economy. Lastly, the correct automatic classification of fish species will help sellers in fixing a fair price for each fish species.

The major contributions of this research work are:

- This paper presents a simple and cost-effective method to automatically classify fish images. Images are captured from collected samples with the help of mobile phone cameras and same are labelled immediately with the help of our third author who has been actively involved with fish fauna study of Assam.
- The work investigates the best feature layer to extract deep CNN features from the proposed ResNet-50 model to achieve the best classifier accuracy and recall rate. While most of the previous works use pre-trained CNNs for feature

extraction, none of them studied the model performance for different layers.

- The proposed model has been tested and it performed well on all the three sets of fish images: QUT dataset, Fish_pak dataset and our own sets of collected images.
- To our understanding, this is the first time any of these contributions have been presented in previous works.

2. METHODOLOGY

The Fig.1 provides the details of the proposed fish classification model that has been followed throughout our experiment. In this section we have discussed about the Fish datasets, Transfer learning methods and various performance measure used for assessing multi-class classification task.

2.1 DATA PRE-PROCESSING AND AUGMENTATION

CNN requires a huge data set. A CNN with smaller datasets increases the risk of over-fitting. As our dataset is not large enough so, in our proposed technique, we have augmented the data without actually bothering to collect new data. In the data augmentation method, the number of samples is increased by applying geometric transformations to the image data sets using simple image processing techniques. We have undergone color processing, scaling, 90° rotation clockwise and anti-clockwise and flipping.

2.2 FEATURE EXTRACTION AND TRANSFER LEARNING

As our dataset is small, we used transfer learning to train our model. In transfer learning approach, a network is initially trained on a base dataset and then, from the learned features, the network can be re-used to train on a different target dataset [29]. In practice, there are two ways to utilize a pretrained network: fixed feature extraction and fine-tuning. In fixed feature extraction method, the fully connected layers are to be removed from the pre-trained network that is trained on ImageNet dataset. In fine tuning approach, fully connected layers of pre-trained network are replaced with new sets of layers that are fine-tuned to take care of the convolutional base.

2.2.1 Activation_49_RELU Layer:

Technically activation_49_RELU is not a layer as no weights/parameters are learned inside any activation layer. An activation layer accepts an input volume of size $W_{input} \times H_{input} \times D_{input}$ and then applies the given activation function (RELU in this case) in an element-wise manner, so, the output dimensions of an activation layer will be same as that of an input dimension, i.e. $W_{input} = W_{output}$, $H_{input} = H_{output}$, $D_{input} = D_{output}$. The activation_49_RELU layer in resnet-50 network contains 2048 activation or feature maps, each with dimensions of 7×7. Let f_k represent the k^{th} activation map, where $k \in 1,2,3\ldots,2048\}$.

2.2.2 Average Pooling Layer:

Pooling layer in CNN down samples the feature values obtained from the previous convolutional layer. This strategy retains the main features of the input image by reducing the size of the feature map. The pooling filter or kernel usually slides the image horizontally and vertically with a window size of K and step size of s(stride) and choose the average of the pixels in the image. For an N*N image, size of the output after pooling operation will be given by Eq.(1).

$$\left(\frac{(N+2p-K)}{s}\right)+1\times\left(\frac{(N+2p-K)}{s}\right)+1 \tag{1}$$

where p is the padding used.

The global average pooling layers have no learnable parameters, so they are less prone to overfitting problem. Let's consider for any given image $f_i(x,y)$ represent the activation of unit i in the last convolutional layer at spatial location (x, y). Then, for unit i, the result to performing global average pooling is represented by Eq.(2)

$$F^i = \frac{1}{N}\sum_{x,y} f_i(x,y) \tag{2}$$

The 2D Global average pooling block takes a feature vector of size (Height)×(Width)×(Depth) and takes the average value of all the values across the entire (input width)×(input height) matrix for each of the (input channels) and outputs 1×1×Depth shown in Fig.2. The use of 2D global average pooling layer is useful in replacing the fully connected blocks of a CNN. Use of global average pooling layer helps in reducing the overall execution time [30]. These networks can also be more robust to spatial translations of input data.

2.2.3 Fully Connected Layer:

It is simpler version of feed forward neural network. Its purpose is to flatten the output of a pooling layer or last convolution layer. It transforms an M-dimensional feature map into 1-D feature vector. In a CNN model, there might be more than one fully connected layer but the last layer will have equal number of output nodes as the number of classes.

2.3 SVM CLASSIFIER

SVM is a preferred choice of classifier in most of the classification problem. It has been extensively used in numerous supervised classification problems like character recognition [17] -[19], hand-written digit recognition, face detection [22]-[23], marine object recognition [27]-[30]. Although SVMs were originally designed for binary classification problem, it can be successfully implemented for multi-class problem too. Basically, two techniques are available for multiclass SVM. One is "one vs. all" SVM where M-number of classes use M-number of binary classifiers. Another one is "one vs. one" where M-number of classes use $M(0.5(M\text{-}1))$-number of classifiers. One vs. all SVM classifier performs well with smaller datasets.

2.4 CLASSIFICATION PERFORMANCE MEASURE

As our classification task involves more than two classes, so we are using performance metrics that are especially helpful for multi-class classification. There are many performance evaluation metrics that come in handy while testing the ability of a multi-class classifier and they are described below:

2.4.1 Confusion matrix:

It is a perfect measure to inspect the performance of a classifier. It works well for n-class classification problem. The correctly classified outcomes are represented diagonally. Incorrect predictions are placed in off diagonal cells of the matrix. True Positive, TP, represents the classes that are positive, also classified as positive by the classifier whereas False Positive are incorrect predictions by the classifier when the actual class is negative. Likewise, True Negative, TN are the values that are classified negative when the class is negative too and False Negative are wrong predictions by the classifier when class is negative.

It is the most popular metrics in classification problem. It gives the idea of overall measure of how much the classification model is able to correctly predict on the entire set of data. Accuracy calculation is based on the number of items correctly identified as either truly positive or truly negative out of the total number of items

$$ACC=(TP+TN)/(TP+TN+FP+FN) \tag{3}$$

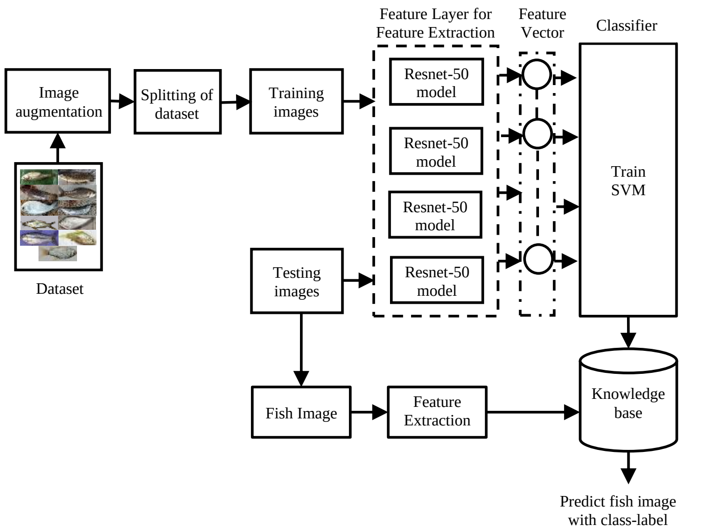


Fig.1. Proposed fish Classification model

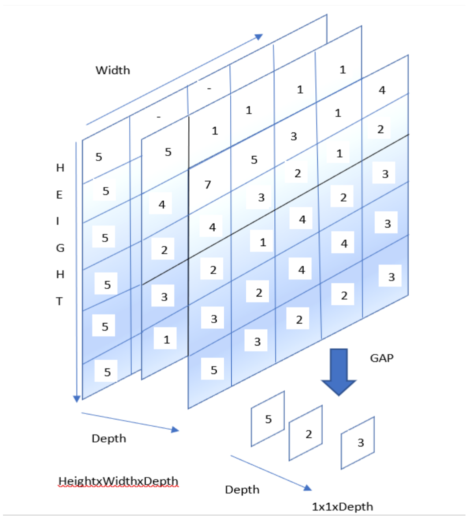


Fig.2. Global Average pooling

True positive and True negative values are obtained from the confusion matrix.

However, Accuracy can’t be considered as a useful measure if we don’t have the same number of samples per class. Moreover, when two or more classes are involved in a dataset, a higher accuracy is not sufficient enough to know, whether all classes are being predicted correctly or whether some of the classes are neglected by the classifier model. In that case, we need to use precision and recall to assess the classifier performance.

2.4.2 Precision:

It refers to the number of items correctly identified as positive out of the total items identified as positive.

$$\text{Precision}=TP/(TP+FP) \tag{4}$$

2.4.3 Recall:

It is one of the most important parameters and sometimes known as sensitivity too. It is defined as number of items correctly identified as positive out of the total actual positives.

$$\text{Recall}=TP/(TP+FN) \tag{5}$$

2.4.4 F-score:

It is the harmonic mean of the precision and recall values, which states the effectiveness of classification.

$$F\text{-}Score=(2*(Precision*Recall))/(Precision+Recall) \tag{6}$$

2.5 ALGORITHM FOR FISH CLASSIFICATION

Algorithm for Fish image classification using Deep CNN features

Input: Natural images of fresh-water Fish.

Output: Class of the fish to be recognised

Step 1: For each image I=1:N where N is the total number of images; perform the following

a. Pre-process the images using shifting, colour processing and rotating
b. Segregate the images into training data and validation data.
c. Find CNN features from relu-49, average pool and f_c-1000 layers of Resnet-50.
d. Input the feature vectors into the classifier for training

Step 2: End for

a. Take a test image.
b. Extracts the feature of the test image from the feature layer
c. Classifies the image
d. Evaluate the performance of the classifier

3. EXPERIMENTAL RESULTS AND DISCUSSIONS

3.1 FISH DATASET

For initial simulations, we have used QUT dataset for training and testing of classification model. This dataset consists of 3,960 fish images collected from 468 different species, full of real-world fish images captured in various conditions defined as "controlled", "out-of-the water" and "in-situ". The "controlled" images consist of fish specimens, with their fins spread, taken against a constant background with controlled illumination. The "in-situ" images are underwater images of fish in their natural habitat and so there is no control over background or illumination, in addition there is the challenge of the unique underwater imaging environment. The "out-of-the-water" images consist of fish specimens, taken out of the water with a varying background and limited control over the illumination conditions.

To check our proposed model's success rate with other datasets, we have used body images of Fish_Pak [20] dataset (Table.1) and also, our own fish images collected from different parts of Assam, India.Fish-Pak dataset contains images of 6 major carps that are native to South-East Asian countries. We have collected 54 fish samples of eleven (11) varieties of small indigenous fresh water fish from different local ponds of Assam, India, using fishing nets and some of these are shown in Fig.3.

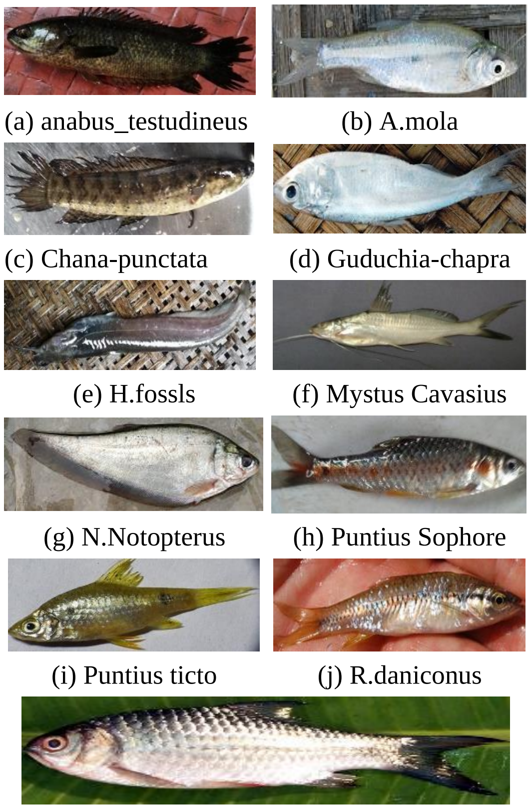
The samples are collected from site that lies between 26.1445° N to 26.4446° N in latitude and 91.7362° E to 91.4411° E longitude. The visual data(images) from the collected samples are captured in natural lighting conditions using a SamsungM30s Galaxy mobile phone camera with a specification of 48-megapixel (f/2.0) + 5-megapixel (f/2.2)+8-megapixel (f/2.2) and are shown in Table.2.

Table.1. Fish-pak Dataset

Species Name	Images/species
Cirrinhus Mrigal	22
Cyprinus Carpio	50
Catla	20
Ctenopharyngodon Idella (grass-carp)	11
Silver carp	47
Labeo-Rohita (Rou)	73
Total	223

Table.2. Own dataset indigenous fresh-water fish

Species	Images/species
Amblypharyngodon_mola	97
Mystus Cavasius	77
Guduchia Chapra	109
N.Notopterous`	96
R.daniconus	79
Systomus Sarana	98
Puntius Ticto	71
Heteropneustes_fossils	147
Anabas_testudineus	122
Puntius Sophore	97
C.punctata	80
Total	1073



(a) anabus_testudineus (b) A.mola (c) Chana-punctata (d) Guduchia-chapra (e) H.fossls (f) Mystus Cavasius (g) N.Notopterus (h) Puntius Sophore (i) Puntius ticto (j) R.daniconus (k) Systomus Sarana

Fig.3. Sample fish image of own dataset

4. RESULTS AND DISCUSSIONS

The simulation is performed in MATLAB environment in a system with an Intel Core i5 7300HQ Processor of 2.50GHz turbo up to 3.50GHz with NVIDIA GeForce GTX 1050 Ti with a computation capability of 6.1.

We use a pre-trained Resnet-50 architecture which accepts an input size of 224×224.As all the images are of different size, we have used 'augmentedImageDatastore' function from MATLAB to resize all the manually augmented images.

Simulation based experiment is conducted first on QUT dataset. Out of 468 species,395 species are chosen in our experiment. The average pool layer of ResNet-50 was used to gather features from the un-augmented original QUT data. As all the images are of different size, we had to resize the images to fit into the first layer of the network. So, we used 'augmentedImageDatastore' function to automatically resize the training and testing data. We have used the output of the last convolutional block of ResNet-50 to extract the features. The output of the Conv5 block is a 7×7×2048-dimensional array. This large array is however, first converted to a 2048-dimensional vector by using the 'avg-pool' layer function from the Deep Learning Toolbox. This layer extracts higher level features using the low-level features of earlier layers.

The CNN features extracted from the deeper ResNet-50 model is used to train the classifier. The classifier used in this experiment is a multiclass linear SVM. We have used a stochastic gradient solver('sgd') optimizer for the linear learning. To extract the training features, we have used a 'mini batch size of 32' so that it fits into our GPU.

The classification result with various training data using the original QUT dataset is shown in Table.3. As it can be seen from Table.3, that classification performance is not up to the mark with original QUT dataset. However, with manually augmented data set of 20,071 images, the classifier shows a considerable improvement in the overall classification performances which can be found in Table.4. Here, we see that true positive rate of classification post image augmentation reaches to 92.11% which is quite an improvement.

Table.3. Classification performance measure on original QUT dataset

Training data	**80%**	**70%**	**60%**	**50%**
Accuracy	54%	52%	49%	47%
Precision	0.5363	0.5193	0.4899	0.4697
Recall	0.51	0.5213	0.5027	0.4861
F-Score	0.523	0.52	0.50	0.4778

Table.4. Classification performance measure on augmented QUT dataset

Training data	**Accuracy**	**Precision**	**Recall**	**F-score**
70%	88%	89.46%	88.17	88.81
80%	91.63%	91.63%	92.60%	92.11%

As CNN features extracted from average pool layer shows improvement in overall classification parameters with QUT dataset, so, we use same methodology to Fish-Pak dataset and our own Indigenous fish image sets. We have experimented with different layers of ResNet-50 as feature extractors. First, we use "activation_49_ReLU" layer as "feature layer". It gave a highest classification accuracy of 97.9% on Fish-Pak and 98.55% on our dataset with 80% training data. In the next set of experiment, we have applied the 'average-pool' layer and skipped the fully connected layer. Maximum accuracy of 100% is achieved on our own dataset with 'average-pool' layer. This skipped Resnet-50 with GAP showed outstanding classification performance on both the datasets. In the last experiment, the last fully connected layer, "fc-1000", is used as feature layer and we observe that the highest accuracy achieved with this layer is only 98.32% on Fish-Pak and 98.59% on our data. Resnet-50 with a skipped global average pooling layer outperforms all other feature layer in terms of accuracy, precision and recall. This gives true positive rate of classification of 100% with a training data of 80% on our dataset. Table.5 shows all the performance parameters of the classifier for the three sets of feature layers on Fish-Pak datasets while Table 6 represents the performance parameters on our own dataset.

Table.5. Classification performance measure on Fish-Pak data

Training Data (%)	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1_score (%)**
Feature layer=activation_49_relu, learning rate=0.01				
80	97.90	97.40	97.45	97.42
70	96.64	95.51	95.70	95.61
Feature layer=avg_pool, learning rate=0.01				
80	98.74	98.85	98.87	98.86
70	98.04	97.89	97.97	97.93
Feature layer=f_c-1000, learning rate=0.01				
80	98.32	98.25	98.28	98.27
70	97.76	97.70	97.71	97.71

Confusion matrices on the Fish-Pak and our own datasets are shown in Fig.4-Fig.9. The confusion matrix gives us the actual numbers of prediction where the x-axis represents the name of the predicted species (predicted class) and y-axis represents the name of the actual class (target class) that it belongs. From Fig.4 and Fig.7, we can observe that with 'activation_49_relu" as feature layer, the total mis-classification with 20% of testing dataset is 11 in Fish-pak dataset and 6 on our dataset. There is no mis-classification with "average pool" layer on our dataset and only 3 with Fish-pak data. Whereas with "fc-1000" layer, the proposed model predicted inaccurately 6 from Fish-pak and 3 from our dataset. From Fig.4 and Fig.6, it is observed that mainly fish images are misclassified as Rohu species due to its similarity with other carp species. Similarly in our dataset also,some of A.mola images are predicted wrongly as Mystus-cavasius and H. fossils which can be observed in Fig.7 and Fig.9. This happens with P.ticto too. As small indigenous fish are tiny compared to other fish, so, the difficulty arises where the model become inefficient in distinguishing it from other species. However, the designed model performed excellent with average pooling layer where there are only 3 misclassifications with Fish-pak dataset and all are predicted correctly in our own dataset. Total number of misclassifications for the three feature layers of ResNet-50 on both the sets of data is represented in Table.7.

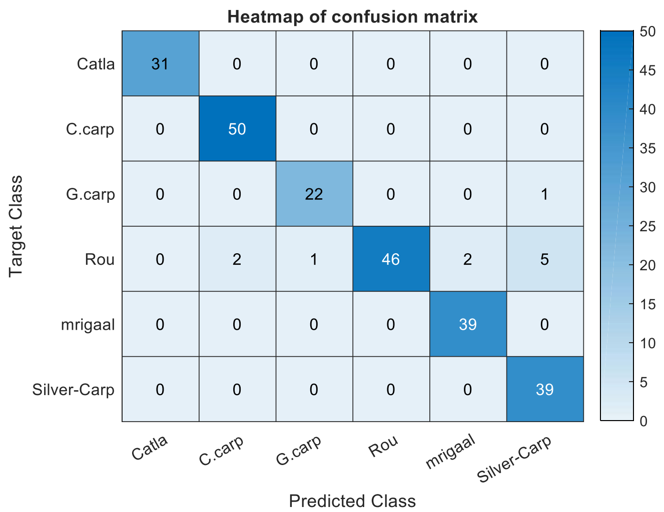


Fig.4. Confusion matrix from Relu-7 layer on Fish-Pak dataset with 20% testing data

Table.6. Classification performance measure on own dataset

Training Data (%)	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1_score (%)**
Feature layer=activation_49_relu, learning rate=0.01				
80	97.18	96.44	96.81	96.62
70	97.83	97.53	97.66	97.59

Feature layer=avg-pool, learning rate=0.01				
80	100	100	100	100
70	99.38	98.31	99.33	99.32
Feature layer = f_c-1000, learning rate=0.01				
80	98.59	98.56	98.68	98.62
70	98.14	98.06	98.19	98.12

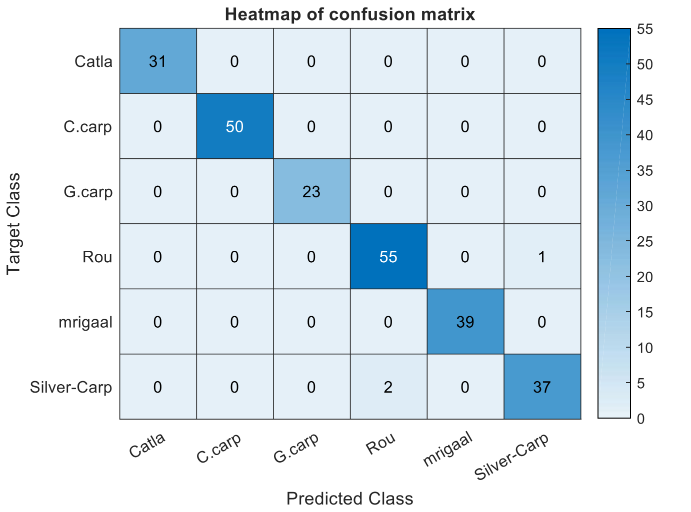


Fig.5. Confusion matrix from avg-pool layer on Fish-Pak dataset with 20% testing data

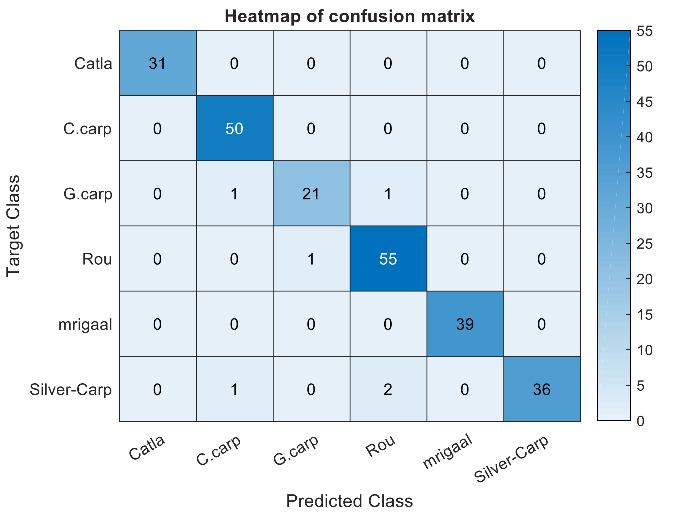


Fig.6 Confusion matrix from f_c-1000 layer on Fish-Pak dataset with 20% testing data

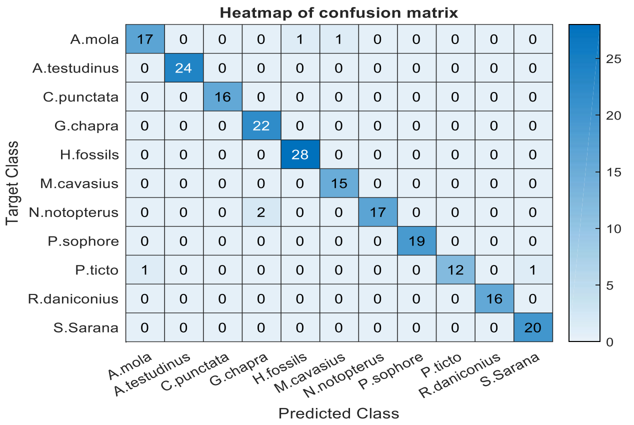


Fig.7. Confusion matrix from Relu-7 layer on own dataset with 20% testing data

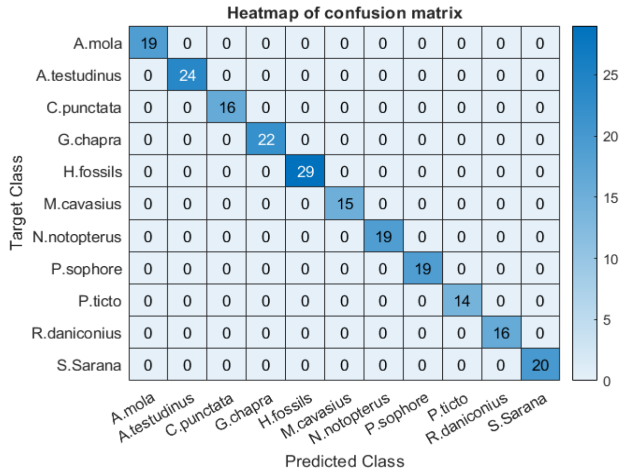


Fig.8. Confusion matrix from avg-pool layer on own dataset with 20% testing data

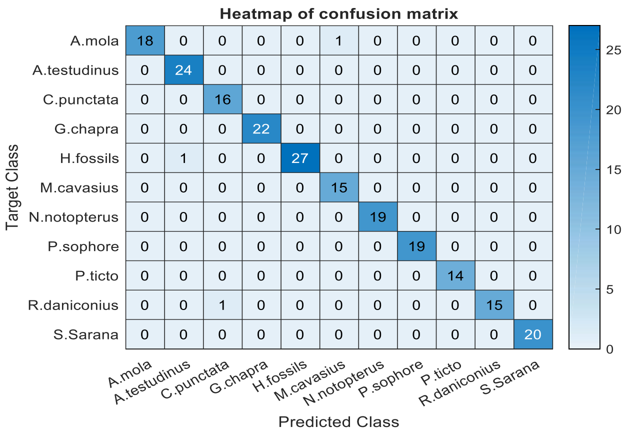


Fig.9. Confusion matrix from fc-1000 layer on own dataset with 20% testing data

Table.7. Misclassification of species for the three feature layers

Parameters	Relu7	Avg_pool	fc-1000
For Fish_pak dataset			
misclassifications	11	3	6
For own indigenous fish dataset			
misclassifications	6	0	3

We have also recorded the recall rate for each of these species for the three feature layers under different training data conditions on our dataset and it is shown in tabular form in Table.8. Recall specifically tells us, out of total observations, how many observations are actually predicted correctly by the model. From this Table.8, it is clear that A.mola images are worst affected where actual positivity rate is lowest among all the 11 fish species. From the confusion matrices and Table.8, it can be summarised that due to similarity in sizes, a few A.mola fish images are missed out while predicting correctly so, the recall value for A.mola is affected. The bar diagram of recall per species is shown in Fig.8 for 70% and 80% training data for the Resnet-50 model with global average pooling layer. From Fig.10, we have observed that except A. mola and C. punctata, all other species are 100% correctly classified for the two sets of training data.

Table.8. Recall per species for different feature layers on own dataset

Species/ parameters	Average_ Pool (0.7)	Average_ Pool (0.8)	_49_ _relu (0.7)	activation_ 49-relu (0.8)	fc-1k (0.7)	fc-1k (0.8)
A.mola	96.6	100	96.6	94.7	89.7	84.2
A.testudinus	100	100	97.3	100	97.3	100
C.punctata	95.8	100	100	100	91.7	100
G.chapra	100	100	100	100	100	100
H.fossils	100	100	100	100	100	100
M.cavasius	100	100	91.03	100	100	100
N.notopterus	100	100	100	100	100	100
P.sophore	100	100	100	100	100	100
P.ticto	100	100	95.2	78.6	100	100
R.daniconius	100	100	95.8	87.5	100	100
S.Sarana	100	100	96.6	100	100	100

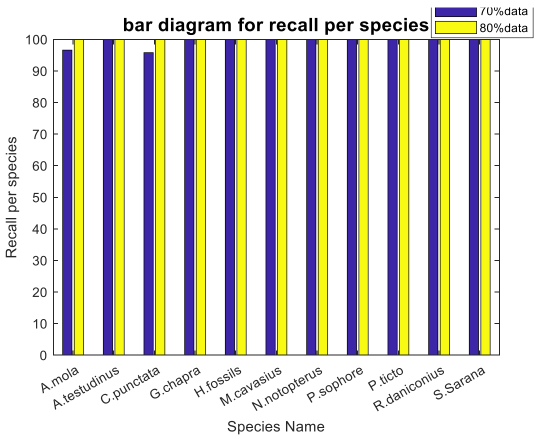


Fig.10. Recall per species on our dataset

4.1 FEATURE MAP

Visualization of feature map of 1st convolutional layer,1st RELU layer, 49th RELU layer and average pooling layer for anabus_testidineus fish is shown in Fig.11-Fig.14. We have extracted visual feature maps from different layers by using MATLAB 'activation' function. These feature maps are extracted from each of the image for further processing. As it is obvious that bottom layers in CNN typically extract fewer shallow features like edge and blob, so they show high resolution and also, number of activations are large in total as shown in Fig.11 and Fig.12. Since, activation-49-RELU and average pool are the top layers, so they tend to extract high level features, as shown in Fig.13 and Fig.14. These features are more conjectured in nature so it is extremely difficult for human eye to interpret.

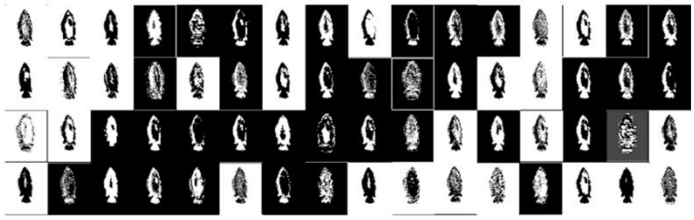


Fig.11. Feature map of anabus-testidineus fish image from 1st convolution layer

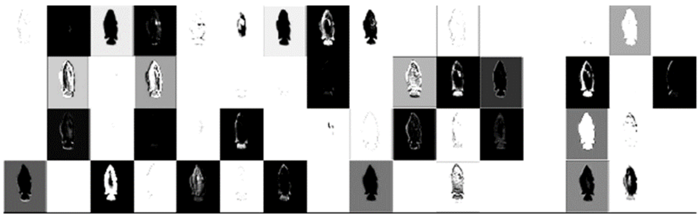


Fig.12. Feature map of anabus-testidineus fish image from 1st relu layer

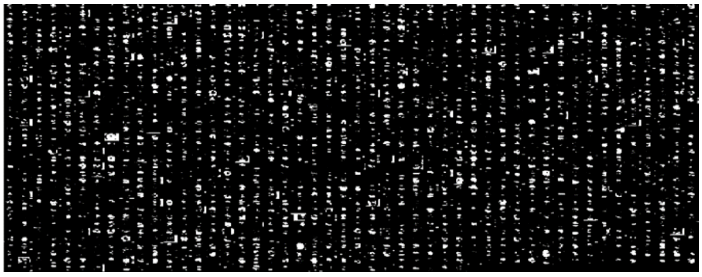


Fig.13. Feature map of anabus-testidineus fish image from 49th-RELU layer

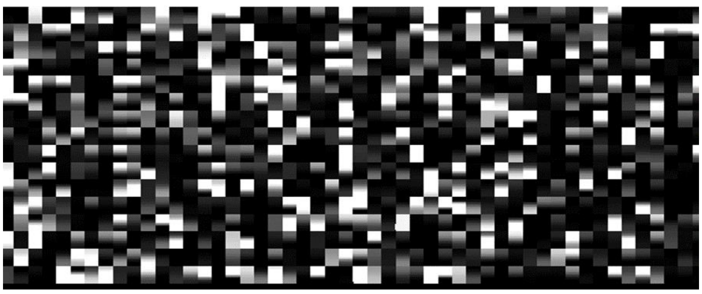


Fig.14. Feature map of anabus-testidineus fish image from average-pool layer

Table.9. Performance comparison of proposed system with state-of-the-art models

Papers	Algorithm	Accuracy (%)	Precision (%)	Recall (%)	Dataset	No. of species
Ma et al. [15]	Pretrained ResNet-152 and SVM	97.19%	Not available	Not available	fish images from Internet	6
Salman et al. [16]	Pretrained ResNet-152 and SVM	94.73%	Not available	Not available	UWA dataset	16
Islam et al. [30]	HLBP feature with SVM	90	Not mentioned	Not mentioned	BDIndigenousFish2019	8
Rauf et al. [22]	32-layer VGGNet,	98.5%	94.83	95.67	Fish_Pak	6
Banan et al. [23]	VGG-16	100	Not mentioned	Not mentioned	Own images of major carps	4
Wang et al. [26]	SPP-densenet	97	97.62	99.2	Google Images	6
Dey et al. [24]	5-layer CNN	99	99.01	99.01	BDIndigenousFish2019	8
Abinaya et al. [25]	AlexNet with fuse Naive Bayesian layer	98.64	99.80	98.99	Fish_Pak	6
Proposed Work	Fine-tuned Resnet-50 and SVM classifier	98.74	98.68	98.78	Fish_Pak	6
Proposed Work	Fine-tuned Resnet-50 and SVM classifier	100	100	100	Own dataset	11

4.2 QUALITIVE COMPARISON

From our thorough background search, we have narrowed down that the work methodologies followed by [21]-[26] are relevant to our model with regard to perspectives and they have used different fresh water fish dataset for fish detection, recognition and species classification. It has been observed that, in those works, features of fish species were extracted only by a single feature layer of pre-trained CNN models. So, therefore, we have made a comparison of our design with all of these state-of-the-art approaches, which heavily counted on deep learning and almost all them exploited deep CNN modules. In Table.9, we have made a comparison of our work with the some of the previous state-of the-art work in fish classification using deep learning techniques. The highest accuracy achieved till date is 100% on four major carps. On Fish_Pak dataset, maximum accuracy achieved was 98.64%. However, their confusion matrix and no. of mis-classification on the testing data is not available to justify the available frequency, whereas, our proposed method attained accuracy of 98.74% on Fish_Pak dataset and misclassification is only 3. The proposed deep CNN feature-based image classification technique achieves 100% classification accuracy, precision and recall on our dataset comprising of locally captured fish images.

5. CONCLUSION AND FUTURE SCOPE

In this paper, we have utilized deep CNN features from the three top layers of a pre-trained Resnet-50 model for classification of different fish species using SVM classifier. This approach shows outstanding classification performance on all the three datasets we have used in the simulation process. We have achieved classification accuracy of 100% on our own dataset of small indigenous fish from North-Eastern parts of India. The accuracy is highest till date in case of multi-class fish classification. In future, many improvements could be made to the proposed scheme. The most important would be applying the proposed model to a large set of local fish from our region and observe whether there are any significant changes in the performance of the model with increasing number of species.

REFERENCES

[1] N.J.C. Strachan, "Recognition of Fish Species by Colour and Shape", *Image and Vision Computing*, Vol. 11, pp. 2-10, 1993.

[2] S. Cadieux, F. Michaud and F. Lalonde, "Intelligent System for Automated Fish Sorting and Counting", *Proceedings of International Conference on Intelligent Robots and Systems*, pp. 1279-1284, 2000.

[3] A. Rova, G. Mori and L. Dill, "One Fish, Two Fish, Butterfish, Trumpeter: Recognizing Fish in Underwater Video", *Proceedings of APR Conference on Machine Vision Applications*, pp. 404-407, 2007.

[4] C. Spampinato and Chen-Burger, "Automatic Fish Classification for Underwater Species Behavior Understanding", *Proceedings of International Conference on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams*, pp. 1-8, 2008.

[5] M.K. Alsmadi and S.A. Noah, "Fish Classification based on Robust Features Extraction from Color Signature using Back-Propagation Classifier", *Journal of Computational Science*, Vol. 7, pp. 52-65, 2011.

[6] B. Benson and R. Kastner, "Field Programmable Gate Array (FPGA) Based Fish Detection Using Haar Classifiers", *American Academy of Underwater Sciences*, Vol. 9, No. 2, pp. 1-15, 2009.

[7] J. Hu, D. Li and X. Si, "Fish Species Classification by Color, Texture and Multi-Class Support Vector Machine using Computer Vision", *Computers and Electronics in Agriculture*, Vol. 88, pp. 133-140, 2012.

[8] M.M.M. Fouad, H.M. Zawbaa, N. El-Bendary and A.E. Hassanien, "Automatic Nile Tilapia fish Classification Approach using Machine Learning Techniques",

Proceedings of International Conference on Hybrid Intelligent Systems, pp. 173-178, 2013.
[9] C. Pornpanomchai and W. Kitiyanan, "Shape- and Texture-Based Fish Image Recognition System", *Kasetsart Journal - Natural Science*, Vol. 47, pp. 624-634, 2013.
[10] M. Rodrigues and E. Carrano, "Evaluating Cluster Detection Algorithms and Feature Extraction Techniques in Automatic Classification of Fish Species", *Pattern Analysis and Applications*, Vol. 18, pp. 1-13, 2014.
[11] P.X. Huang and R.B. Fisher, "Hierarchical Classification with Reject Option for Live Fish Recognition", *Machine Vision and Application*, Vol. 26, pp. 89-102, 2015.
[12] P. Mathew and S. Elizabeth, "Fish Identification Based on Geometric Robust Feature Extraction from Anchor/Landmark Points", *Proceedings of International Conference on Image Processing and Machine Vision*, pp. 1-14, 2017.
[13] Jisha Jose and S. Sureshkumar, "Tuna Classification using Super Learner Ensemble of Region-Based CNN-Grouped 2D-LBP Models", *Information Processing in Agriculture*, Vol. 9, pp. 1-13, 2021.
[14] S.A. Siddiqui, A. Salman and E.S. Harvey, "Automatic Fish Species Classification in Underwater Videos: Exploiting Pretrained Deep Neural Network Models to Compensate for Limited Labelled Data", *ICES Journal of Marine Science*, Vol. 75, pp. 374-389, 2017.
[15] Y. Ma, P. Pengfei and Y. Tang, "Research on Fish Image Classification Based on Transfer Learning and Convolutional Neural Network Model", *Proceedings of International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pp. 850-855, 2018.
[16] A. Salman, S. Maqbool and F. Shafait, "Real-Time Fish Detection in Complex Backgrounds using Probabilistic Background Modelling", *Ecological Informatics*, Vol. 51, No. 2, pp. 44-51, 2019.
[17] N.E. Khalifa and Aboul Ella, "Aquarium Family Fish Species Identification System Using Deep Neural Networks", *Proceedings of International Conference on Advanced Intelligent Systems and Informatics*, pp. 1-13, 2018.
[18] A. Jalal, S. Ajmal and F. Shafait, "Fish Detection and Species Classification in Underwater Environments using Deep Learning with Temporal Information", *Ecological Informatics*. Vol. 57, pp. 1-14, 2020.
[19] M.A. Iqbal and Z. Wang, "Automatic Fish Species Classification Using Deep Convolutional Neural Networks", *Wireless Personal Communications*, Vol. 116, pp. 1043-1053, 2021.
[20] S.Z.H. Shah, Malik Shahzaib Farooq and M. Muhammad, "Fish-Pak: Fish Species Dataset from Pakistan for Visual Features Based Classification", *Mendeley Data*, Vol. 3, No. 2, pp. 1-14, 2019.
[21] M.A. Islam and M.M. Rahman, "Indigenous Fish Classification of Bangladesh using Hybrid Features with SVM Classifier", *Proceedings of International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering*, pp. 1-7, 2019.
[22] H.T. Rauf and Syed Ahmad Chan, "Visual Features based Automated Identification of Fish Species using Deep Convolutional Neural Networks", *Computers and Electronics in Agriculture*, Vol. 167, pp. 1-18, 2019.
[23] A. Banan, Amin Nasiri and A. Taheri-Garavand, "Deep Learning-Based Appearance Features Extraction for Automated Carp Species Identification", *Aquacultural Engineering*, Vol. 89, pp. 1-17, 2020.
[24] K. Dey, M.M. Hassan, M.M. Rana and M.H. Hena, "Bangladeshi Indigenous Fish Classification using Convolutional Neural Networks", *Proceedings of International Conference on Information Technology*, pp. 899-904, 2021.
[25] N.S. Abinaya and R. Sidharthan, "Naive Bayesian Fusion based Deep Learning Networks for Multisegmented Classification of Fishes in Aquaculture Industries", *Ecological Informatics*, Vol. 61, pp. 1-15, 2021.
[26] H. Wang, Y. Shi and H. Zhao, "Study on Freshwater Fish Image Recognition Integrating SPP and DenseNet Network", *Proceedings of IEEE International Conference on Mechatronics and Automation*, pp. 1-8, 2020.
[27] A. Krizhevsky and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", *Neural Information Processing Systems*, Vol. 25, pp. 1-14, 2012.
[28] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition", *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
[29] Sinno Jialin and Qiang Yang, "A Survey on Transfer Learning", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 1, pp. 1345-1359, 2010.
[30] Hui-Hui and Po Ting. "The Effect of Different Deep Network Architectures upon CNN-Based Gaze Tracking", *Algorithms*, Vol. 13, pp. 127-1136, 2020.