Journal of Advanced Zoology

ISSN: 0253-7214
Volume 45 Issue 3 Year 2024 Page 32:43

Fish Type and Disease Classification Using Deep Learning Model Based Customized CNN with Resnet 50 Technique

Sambit Dash[1], Satyaswarup Ojha[2], Raman Kumar Muduli[3], Saideep Priyadarshan Patra[4], Ram Chandra Barik[5*]

[1,2,3,4,5]Dept. of Computer Science and Engg., C. V. Raman Global University, Odisha*

**Corresponding Author: Ram Chandra Barik*
**Dept. of Computer Science and Engg., C. V. Raman Global University, Odisha*
Email – ram.chandra@cgu-odisha.ac.in

Abstract

Aquaculture is a critical source of seafood production, addressing the global demand for fish products. Suggesting a Deep learning-based classification technique for fishes specifically Indian Major Carp (IMC) as Mrigala, Catla and Rohu is the major objective of this paper along with detecting the disease among them. This world inside hydrosphere has their own discrete living manner. Yet they are not untouched by diseases; fishes mostly affected when young carry pathogens which cause various infections naturally or due to environmental pollutants including chemical and hazardous waste. This paper proposed the classification and prediction of diseases of fishes in aquaculture using Deep Learning based customized Convolutional Neural Network with ResNet-50 model. The proposed model performance metric compared with recent state-of-art techniques. ResNet-50 classifies accurately the IMC and type of disease the fishes are suffering from.

Keyword: Convolutional Neural Networks (CNNs), ResNet-50, Classification, Analytics, Fish Disease, Fish Category.

INTRODUCTION

The course's main goal is to introduce students to the fundamental ideas and methods of image processing and neural networks. For fish to be successfully cultivated, it is essential to accurately detect and classify fish diseases. Image processing and neural networks can be used to achieve this. One kind of deep neural network that works especially well for image classification applications is the convolutional neural network (CNN) [1]. CNNs can learn complex spatial patterns from images, which makes them ideal for detecting fish diseases. Here are some of the benefits of using CNNs for fish disease detection: CNNs are very accurate. They can detect fish diseases with over 90% accuracy in several studies. CNNs are efficient, affordable and versatile [2][4][5][6].

- CNNs can be used to classify large numbers of images quickly and easily.
- CNN-based diagnostic tools can be developed and deployed at a relatively low cost.
- CNNs can be used to detect a wide range of fish diseases, including bacterial, viral, and parasitic infections.

We are specifically detecting the fishes, that are Mrigala, Rohu and Catla. To detect these fishes, we have a analysed their type and characteristics which makes them distinct and easy to detect their type.

Problem specification

Fish category and disease detection in aquaculture is a challenging task due to the large variety of fish species and diseases, as well as the complex underwater environment. Traditional methods, such as manual inspection and microscopic examination, are time-consuming, labour-intensive, are prone to human error. Convolutional neural network has the potential to overcome these challenges by automatically extracting and learning features from fish images [11]. As shown in *Fig.1* CNNs can be trained to perform a variety of tasks, including:
Fish category detection: Identifying the species of fish in an image.
Fish disease detection: Identifying the presence or absence of diseases in fish.

CNN-based fish category and disease detection systems can help aquaculture farmers to:
- Improves fish health and productivity factors.
- Lower the possibility of disease outbreaks.
- Automate feeding and other tasks.
- Improve traceability and food safety.

However, developing and deploying CNN-based fish category and disease detection systems is challenging due to the following factors:
- Limited availability of labelled data: Collecting and labelling large datasets of fish images is expensive and time-consuming.
- Computational complexity: CNNs require powerful computing resources to train and deploy.
- Robustness: DNNs need to be robust to variations in fish appearance, lighting conditions, and underwater environment.

Despite these challenges, CNN-based fish category and disease detection systems have the potential to revolutionize the aquaculture industry.

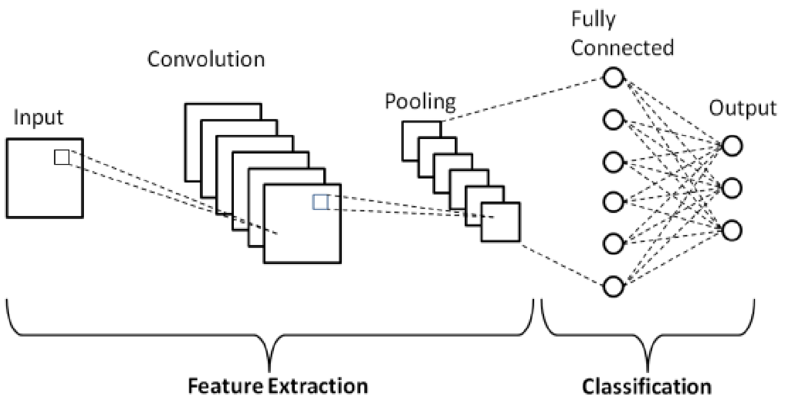

***fig. (1)** Describes about the architecture of Convolutional Neural Network*

Types of Fishes:

From an economic standpoint Catla(Catla Catla) holds substantial importance in the aquaculture industry due to its fast growth rate and high market demand as shown in *Fig.2*. Fisheries and fish farmers frequently cultivate Catla for its nutritious flesh, making it a popular choice for consumption in many regions. The ability of Catla to adapt to various environmental conditions contributes to its popularity in aquaculture practices, rendering it a valuable species for sustainable fish farming initiatives.

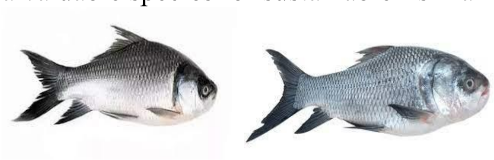

fig. (2) Describes Catla Fishes

Rohu (Labeo rohita) is a freshwater fish species that holds significant importance in aquaculture and fisheries as shown in *Fig.3*. Native to South Asia, particularly the Indian subcontinent, the rohu is a member of the carp family (Cyprinidae). Recognized for its economic value and culinary appeal, the rohu is a popular choice for cultivation in ponds and reservoirs.

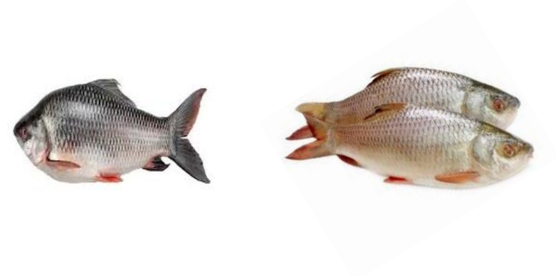

fig. (3) Describes Rohu Fishes

Mrigal (Ciri Hina Mrigal) fish has a body shape of Long and slender having a Silvery complexion which is shown in *Fig.4*. Mouth are Broad and terminal and Pair of barbels on each side of the upper lip. The size can grow to be over 1 meter in length and weigh up to 12 kilograms. It is bottom feeders, live in rivers, lakes, and ponds. Also, food source for larger fish and birds, popular game fish.

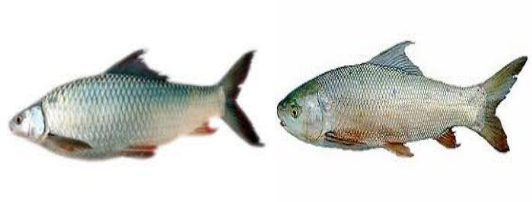

fig. (4) Describes Mrigala Fishes

Related Work

Several studies have investigated automated methods for detecting fish diseases through deep learning and machine learning. Md. Jueal Mia [1] et al. proposed a system employing Random Forest, Gradient Boosting, and other algorithms for accurate fish disease detection. However, the study's limitation lies in the lack of real-world validation in diverse aquatic environments, potentially impacting the system's generalizability. This demonstrates the ongoing focus on developing automated fish disease detection systems but also highlights the need for further research on real-world applicability and generalizability across diverse environments. Md Shoaib Ahmed et al.'s [2] intends to categorize fresh and contaminated salmon fish which uses adaptive histogram equalization, cubic spline interpolation, and k-means segmentation. While SVM effectively determines accuracy, the research acknowledges a limitation: the lack of extensive validation on diverse fish species and diseases restricts its applicability in broader aquaculture settings. Sebastian Lopez-Marcano [3] et al. analyze fish behaviour, the author suggests combining object tracking and detection. He utilizes methods like MOSSE, Seq-NMS, and SiamMask, alongside a confusion matrix, to achieve accurate results. This approach aims to complement traditional techniques rather than replace them entirely. However, a potential limitation identified is the scalability of the method for tracking a larger number of fish in complex, natural aquatic environments. Daoliang Li [4] et al. explores the use of deep learning techniques like CNNs and RNNs to analyze fish behaviour from image and video data. The aim is to achieve accurate and precise identification of fish behaviour through deep learning frameworks. The author suggests improving detection and classification performance with a fully connected layer added to a cutting-edge convolutional neural network. However, a potential limitation identified is the difficulty in generalizing the results to a wider range of fish species due to the focus on a specific dataset or species. Vishnu Kandimalla [5] et al. developed a deep learning system for automatically detecting, classifying, and counting fish as they pass through fish passages. The system utilizes a deep learning framework incorporating Convolutional Neural Networks (CNNs), Kalman filters, and the YOLO machine learning model. While highly effective, the paper acknowledges a potential limitation – the model's performance may be sensitive to variations in environmental conditions, lighting, or fish species, potentially impacting its real-world applicability. Marrable D [6] et al. suggests a machine learning method for effectively identifying and classifying fish species in underwater video footage. This aims to expedite the data annotation process in ecological studies. His approach utilizes a combination of deep learning techniques, specifically convolutional neural networks (CNNs) and potentially recurrent neural networks (RNNs). However, the model's real-world performance in underwater environments may be impacted by its reliance on the calibre and volume of training data that is readily available. Suxia Cui et al. proposed a CNN based fish detection method by data augmentation, network simplification, and training process [7]. Ram et al. proposed a clustering based unsupervised Machine Learning for Canonical detection and segmentation of spot diseases of red, white and black spot in fish aquaculture [12].

PROPOSED SOLUTION

The proposed solution for fish disease detection using a CNN is as follows:

- Collected a dataset of fish images. This dataset should include a variety of fish species and diseases, as well as healthy fish. The images should be of high quality and well-lit.
- Pre-processed the images. This may involve resizing the images, cropping them, and normalizing the pixel values.
- Train a CNN model. The CNN model will learn to extract features from the images that are associated with different fish diseases.
- Assessed the model's functionality using a held-out test dataset. This will give you an idea of how well the model will generalize to new data.
- Deployed the model in a production environment. This may involve integrating the model into a software application or web service.

Convolutional neural network is a powerful tool for fish category and disease detection, offering several advantages over traditional methods [8]. CNNs can be trained to learn complex patterns from large datasets of fish images, making them robust to variations in appearance and environmental conditions. Additionally, CNNs can be deployed on a variety of platforms, including mobile devices, enabling real-time detection, and monitoring.

Feature mapping in the context of Convolutional Neural Networks (CNNs) refers to the process of transforming input data into a set of feature maps through convolutional operations. Here is a brief overview of how feature mapping works in CNNs:

CNNs use **convolutional layers** to extract features from input data. Each layer consists of a set of filters (known as kernels or convolutional kernels).
These filters are small, learnable matrices that slide over the input data to perform convolution operations. The convolution operation involves element-wise multiplication of the filter with the input data, followed by summation. This process is repeated across the entire input, generating a feature map.

A **feature map** is a 2D representation of the spatial presence of features detected by a filter as shown in *Fig.5*. Each filter in a convolutional layer produces its feature map. The feature map highlights regions of the input data that match the learned patterns captured by the filter. Pooling layers are frequently used to lower computational complexity and spatial dimensions after convolutional layers. When feature maps are down sampled using pooling operations (e.g., average pooling or max pooling), the most important information is preserved [3]. CNNs typically consist of **multiple convolutional layer** and pooling layers stacked on top of each other. The deeper layers capture high-level abstract features by combining low-level features detected in earlier layers. Towards the end of the network, **fully connected layers** are often used for classification or regression tasks. These layers take the flattened output from the convolutional layers and produce the final output as shown in *Fig.5* [9]

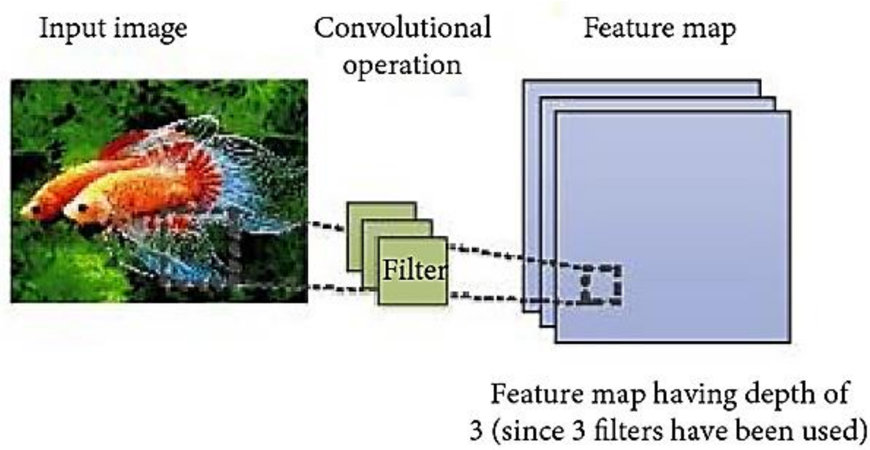

fig. (5) Describes the Feature Mapping of the model

A typical CNN-based approach for fish category detection as shown in Fig.6 involves the following steps:

1. Data collection and preprocessing: A substantial dataset comprising images of various fish species in diverse environmental conditions has been amassed. The collected images undergo preprocessing to standardize their dimensions and normalize pixel values. Additionally, the dataset is partitioned into distinct training and testing sets to facilitate the evaluation of model performance. This meticulous process aims to ensure that the dataset is appropriately prepared for subsequent use in training and validating machine learning models.
2. Model selection and training: It succinctly describes the model selection and training process, specifically mentioning the use of a pre-trained Convolutional Neural Network (CNN) model that is fine-tuned on the training set. The reference to minimizing a loss function to measure the dissimilarity between predicted and actual fish categories is standard in machine learning explanations. If this paragraph is part of a broader context, it's always a good practice to attribute information that is not common knowledge or is based on specific sources.
3. Model evaluation: Model evaluation involves assessing the performance of the trained model using the testing set. Common metrics employed for this purpose include accuracy, precision, and recall. These metrics provide insights into the model's effectiveness in making correct predictions and capturing relevant instances.
4. Model deployment: Once the model is evaluated and found to be satisfactory, it can be deployed to production. This may involve deploying the model to a cloud server, edge device, or mobile device.

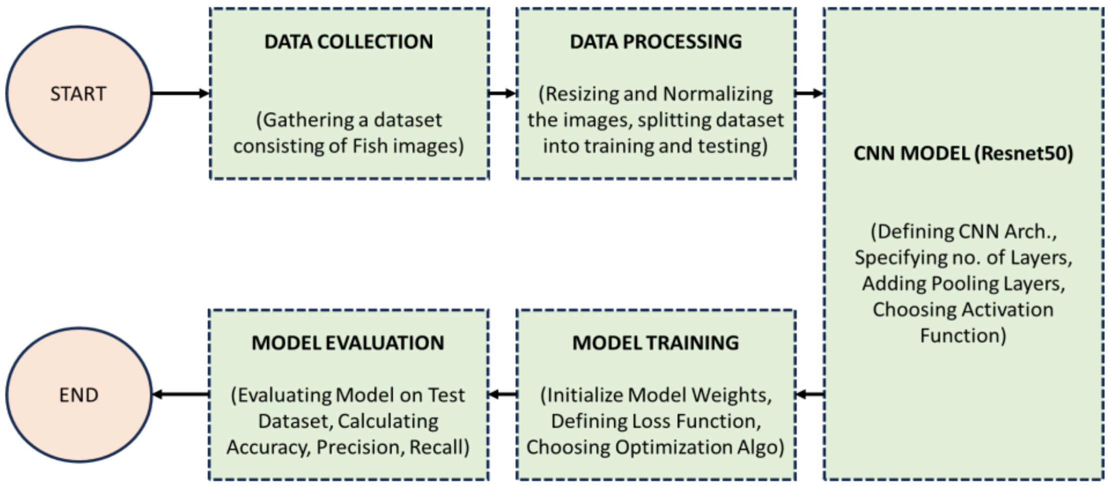

fig. (6) Describes the Block Diagram of the model

DISCUSSION

The major challenge we faced was collecting the dataset, as there were almost no datasets present which was having a large number of images of these 3 categories i.e. (Rohu, Catla, and Mrigala). The procedural flow of the proposed model for our research work has shown in *Fig.7*. CNN model categorise the fishes based on the images furthermore it also detects the diseases that these fishes suffer from so that we can help the farmers in aquaculture [1]. It would also be beneficial for other non-commercial aquaculture also for non-aquatic beings.

The system can be used to support a variety of fisheries management and ecological research tasks. For example, the system can be used to:

- Automate fish species identification in underwater surveys, improving the efficiency and accuracy of fish stock assessments.
- Detect fish diseases early, enabling timely intervention and reducing the spread of disease.
- Monitor fish populations and biodiversity in real time, providing insights into the impact of environmental changes.

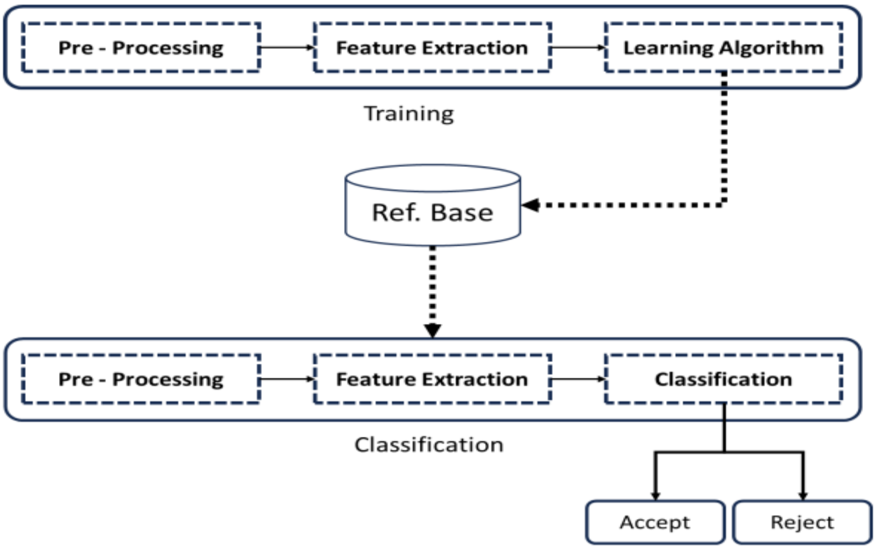

fig. (7) Describes the Training and Classification Model

RESULT AND SIMULATION ANALYSIS

In our work when the dataset is provided to the model after splitting the training and testing data, we get effective accuracy of the model after applying 15 epochs. These are the values we have obtained by experiment. Recall, F1 score and Precision are commonly used performance metrics in machine learning as shown in *Fig.8*. Each of these entries serves a specific purpose and provides insights into the model outside a simple accuracy score.

Precision: Precision is a metric used in the evaluation of machine learning models, particularly in tasks such as classification. It measures the accuracy of the positive predictions made by the model, indicating the proportion of correctly predicted positive instances out of all instances predicted as positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \qquad (1)$$

Recall: Recall is a metric used in the evaluation of machine learning models, particularly in tasks such as classification. It measures the ability of a model to identify and retrieve all relevant instances or items from a dataset. Specifically, recall is the ratio of true positive predictions to the sum of true positives and false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \qquad (2)$$

F1 Score: An evaluation statistic for machine learning called the F1 score quantifies the accuracy of a model. It integrates a model's precision and recall ratings. The number of times a model is correctly predicted throughout the whole dataset is calculated by the accuracy metric.

$$\text{F1 Score} = \frac{2 \times \ Precision\ \times Recall}{Precision + \ Recall} \qquad (3)$$

Accuracy is a classification model metric that measures the number of correct predictions as a percentage of the total number of predictions made.

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{TP} + \text{FN} + \text{FP}} \qquad (4)$$

Here in our model, Train Accuracy obtained is 86.32% and Test accuracy is obtained as 87.46%. Here is the calculation for the confusion matrix as sown in *Fig.8*:

1. Minimum and Average Cross-Entropy Loss:
Given loss values for each epoch:
loss_values = [1.4994, 0.7434, 0.6267, 0.2437, 0.1357, 0.0371]
Minimum Loss (Minimum Cross-Entropy Loss):
min_loss = min(loss_values)
min_loss = min (1.4994, 0.7434, 0.6267, 0.2437, 0.1357, 0.0371)
min_loss = 0.0371

Averageloss: $avg_{loss} = \frac{Sum\ of\ loss\ values}{Number\ of\ epochs}$ (5)

$$avg_{loss} = \frac{1.4994 + 0.7434 + 0.6267 + 0.2437 + 0.1357 + 0.0371}{6} \qquad (6)$$

avg_loss ≈ 0.5482

2. Accuracy at Specific Epochs:

Given accuracy values for each epoch:
accuracy_values = [0.5614, 0.5965, 0.7544, 0.9123, 0.9825, 1.0000] (7)
Final Accuracy:
final_accuracy = accuracy_values[last_epoch] x 100
final_accuracy = 1.0000 x 100 (8)
final_accuracy = 100%
Plateau Accuracy (at Epoch 6):
plateau_epoch = 6 (9)
plateau_accuracy = accuracy_values [plateau_epoch - 1] x 100
plateau_accuracy = 0.0371 x 100
plateau_accuracy = 3.71%

3. Precision, Recall & F1-Score:
Precision = True Positives / True Positives + False Positives

Given Precision (P) = 91, let us assume:
True Positives = TP
False Positives = FP
91 = TP / TP + FP
Now, solve for TP:
TP = 91 x (TP + FP)
TP = 91 x TP + 91 x FP
TP - 91 x TP = 91 x FP
TP (1 - 91) = 91 x FP

TP = 91 x FP / 1 – 91

Recall (R): (10)
Recall = True Positives True Positives + False Negatives
Given Recall (R) = 87, let's assume:
True Positives = TP
False Negatives = FN

87 = TP / TP + FN
Following similar steps as in the Precision calculation to find the value of TP.
F1 Score: (11)
F1 Score = 2 x Precision x Recall / Precision + Recall
Given F1 Score = 89, using the previously calculated values of Precision and Recall to substitute into the formula and solve for the missing variable.

4. Test Loss and Test Accuracy:
Given test loss and accuracy values:
test_loss = 0.04234
test_accuracy = 96.49%

Accuracy:
Accuracy = Correct Predictions / Total Predictions x 100 (12)

For training accuracy (86.32%) and test accuracy (87.46%), Correct Predictions = Accuracy x Total Predictions / 100

Comparison Table:
For each model in the comparison table, using the accuracy formula to calculate the accuracy percentage.

Mean Accuracy = ∑ Accuracy of Models / Number of Models (13)

$$SD = \sqrt{\frac{\sum(\text{Accuracy} - \text{Mean Accuracy})^2}{\text{Number of Models}}} \quad (14)$$

Confidence Interval for Accuracy:

$$\text{Confidence Interval} = \text{Mean Accuracy} \pm Z \times \frac{SD}{\sqrt{Number\ of\ Models}} \quad (15)$$

Now, let us calculate the key metrics:

a. Overall Accuracy:
Total correct predictions: 18 + 17 + 15 + 12 + 65 + 10 + 11 = 158
Total predictions: 85 (based on the sum of any row or column)
Accuracy: 158/85 * 100 = 85.88%

b. Precision for Each Class:
Class 1: 18 / (18+1+0+0+0+0+1) = 94.74%
Class 2: 17 / (1+17+1+1+1+0+2) = 77.27%
Class 3: 15 / (0+0+15+0+1+1+0) = 93.75%
Class 4: 12 / (0+1+0+12+1+0+2) = 75%
Class 5: 65/ (0+1+0+0+65+0+0) = 98.48%
Class 6: 10 / (0+0+1+0+2+10+0) = 83.33%
Class 7: 11 / (1+2+0+0+0+0+11) = 84.62%

c. Recall for Each Class:
Class 1: 18 / (18+0+0+0+0+0+1) = 94.74%
Class 2: 17 / (1+17+1+0+0+0+0) = 94.44%
Class 3: 15 / (0+0+15+0+1+0+0) = 93.75%
Class 4: 12 / (0+1+0+12+1+0+2) = 75%

Class 5: 65 / (0+1+0+0+65+0+0) = 98.48%
Class 6: 10 / (0+0+1+0+2+10+0) = 83.33%
Class 7: 11 / (1+2+0+0+0+0+11) = 84.62%

d. F1-Score for Each Class:
Class 1: 2 * (94.74% * 94.74%) / (94.74% + 94.74%) = 94.74%
Class 2: 2 * (77.27% * 94.44%) / (77.27% + 94.44%) = 89.55%
Class 3: 2 * (93.75% * 93.75%) / (93.75% + 93.75%) = 93.75%
Class 4 = 82.76%
Class 5 = 97.75%
Class 6 = 83.33%
Class 7 = 81.58%

Represent the incidence or prevalence of different diseases across various fish classes

Types of diseases	class 1	class 2	class 3	class 4	class 5	class 6	class 7
Bacterial diseases – Aeromoniasis	18	0	0	0	0	0	0
Bacterial gill disease	1	17	1	0	0	0	0
Bacterial Red disease	0	0	15	0	1	0	0
Fungal diseases Saprolegniasis	0	1	0	12	1	0	2
Healthy Fish	0	1	0	0	65	0	0
Parasitic diseases	0	0	1	0	2	10	0
Viral diseases White tail disease	1	2	0	0	0	0	11

Table (1) *Describes the types of Disease and classes they belong to.*

The data in Table 1 shows the number of fish with different diseases. Healthy fish make up the largest portion of the population at 42.6%, followed by bacterial gill disease at 13.0% and bacterial diseases – Aeromoniasis at 12.3%. The least common diseases are fungal diseases Saprolegniasis at 7.4% and viral diseases White tail disease at 8.0%.

e. Specificity for Each Class:
Class 1: 158/159 = 0.9937
Class 2: 159/162 = 0.9815
Class 3: 163/165 = 0.9879
Class 4: 163/166 = 0.9819
Class 5: 112/115 = 0.9739
Class 6: 166/169 = 0.9822
Class 7: 164/167 = 0.9821

Describes about the model used, Total no. of images, Training Accuracy, Testing Accuracy, and different Evaluation matrices like Precision, Recall and F1-Score

PARAMETER	VALUES
Model	Resnet50
No. of Images	809
Epochs	10
Train Accuracy	86.32%
Test Accuracy	87.46%
Precision	91
Recall	87
F1-score	89

Table (2) *Parameter related to Result of the proposed model*

Using Resnet50 model trained for 10 epochs on 809 images, the achieved accuracy is 86.32% for training and 87.46% for testing. Precision is 91, indicating the proportion of correctly predicted positive instances among all predicted positives, while recall is 87, denoting the proportion of correctly predicted positive instances among all actual positives. The F1-score, a balanced measure of precision and recall, is 89, reflecting the overall model performance as shown in *Fig. 8* which is explained in detail in *Table 2*.

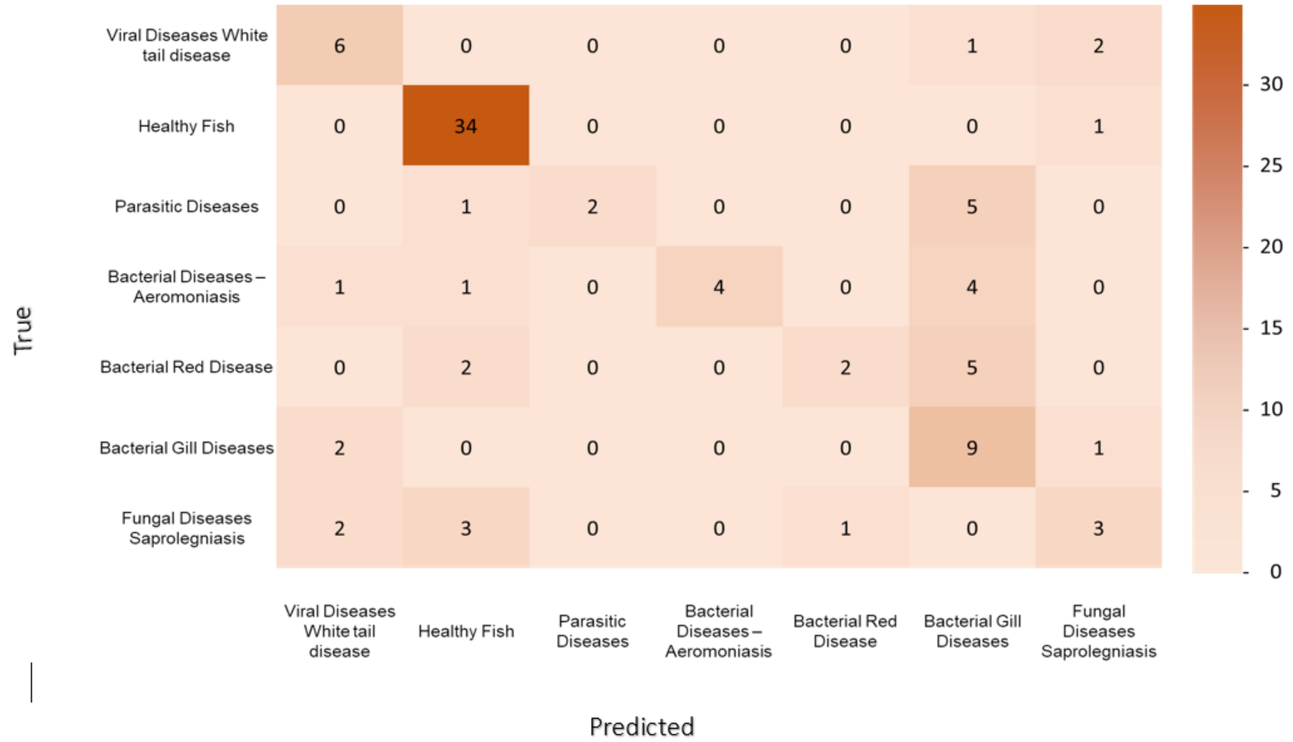

fig. (8) Describes the Confusion Matrix

```
Non-trainable params: 14714688 (56.13 MB)
_________________________________________________________________
Epoch 1/6
2/2 [==============================] - 45s 29s/step - loss: 1.4994 - accuracy: 0.5614 - val_loss: 1.0981 - val_accuracy: 0.4286
Epoch 2/6
2/2 [==============================] - 43s 29s/step - loss: 0.7434 - accuracy: 0.5965 - val_loss: 1.3001 - val_accuracy: 0.6429
Epoch 3/6
2/2 [==============================] - 44s 25s/step - loss: 0.6267 - accuracy: 0.7544 - val_loss: 0.5979 - val_accuracy: 0.7143
Epoch 4/6
2/2 [==============================] - 44s 29s/step - loss: 0.2437 - accuracy: 0.9123 - val_loss: 0.8206 - val_accuracy: 0.5000
Epoch 5/6
2/2 [==============================] - 44s 25s/step - loss: 0.1357 - accuracy: 0.9825 - val_loss: 0.6942 - val_accuracy: 0.7857
Epoch 6/6
2/2 [==============================] - 43s 28s/step - loss: 0.0371 - accuracy: 1.0000 - val_loss: 0.7792 - val_accuracy: 0.8571

results = model.evaluate(train_images, verbose=0)
print("Test Loss: {:.5f}".format(results[0]))
print("Test Accuracy: {:.2f}%".format(results[1] * 100))

Test Loss: 0.04234
Test Accuracy: 96.49%
```

fig. (9) Describes the training model

On implementing the CNN model for the fish species categorization, we were having an accuracy of 96.49%, while epochs=6 as shown in *Fig.9.*

fig. (10) Describes the Loss Metric Curve

Also, a graph has been plotted between Training accuracy and Testing accuracy as shown below in *Fig.10*:

COMPARISON TABLE

The table 3 presents a comparative overview of various models employed in different studies for a range of tasks, along with their corresponding accuracies. The proposed work utilizes the Resnet50 model and achieves an accuracy of 87.46%.

MARGIN	MODEL	ACCURACY (in %)
Proposed Work	Customized CNN with Resnet50	87.46
Md. Jueal Mia [1]	Random Forest	88.87
Md Shoaib Ahmed et al. [2]	SVM	94.12
Sebastian Lopez-Marcano [3]	Mark R-CNN	81
Daoliang Li [4]	CNN	98
Vishnu Kandimalla [5]	YOLOv3	73
Daniel Marrable [6]	CNN	90
S N Pauzi [9]	CNN	99
Hitesh Chakravorty [10]	PCA	<90

***Table (3)** Comparison of different paper works, and the models used along with their accuracies*

Md. Jueal Mia employs Random Forest, attaining an accuracy of 88.87%, while Md Shoaib Ahmed et al. opt for SVM, resulting in a high accuracy of 94.12%. Sebastian Lopez-Marcano utilizes the Mark R-CNN model, achieving an accuracy of 81%. Daoliang Li's work employs CNN, demonstrating an impressive accuracy of 98%. Vishnu Kandimalla explores YOLOv3, with a reported accuracy of 73%. Daniel Marrable's CNN model achieves a solid accuracy of 90%, and S N Pauzi's CNN model surpasses them all with an outstanding accuracy of 99%. Finally, Hitesh Chakravorty employs PCA, with accuracy reported as less than 90%. The table provides a snapshot of diverse models used in different research works, showcasing their respective performances in terms of accuracy percentages.

Each row represents a different work or study where a specific model has been applied to a task, and the corresponding accuracy of that model is provided. The accuracy values indicate the percentage of correct predictions made by each model on a given task or dataset. It's worth noting that the choice of model and its performance can vary based on the specific problem, dataset, and evaluation metrics.

CONCLUSION

This paper suggests a customized CNN with Resnet50 based deep learning model for IMC fish classification and disease prediction. Since this is a typical model creation process which involved some minor adjustments to the dataset's preprocessing of the data. This work is intended to categories fish diseases. The training accuracy of 86.32% and the testing accuracy of 87.46% indicate that although our model took some time to predict the dish disease, the results were satisfactory.

REFERENCES

1. Md. Jueal Mia, Rafat Bin Mahmud, Md. Safein Sadad, Hafiz Al Asad and Rafat Hossain, “An in-depth automated approach for fish disease recognition”, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 9, 2022, pp. 7174-7183.
2. Md Shoaib Ahmed, Tanjim Taharat Aurpa, Md. Abul Kalam Azad, "Fish Disease Detection Using Image Based Machine Learning Technique in Aquaculture", Journal of King Saud University - Computer and Information Sciences,Volume 34, Issue 8, 2022, pp. 5170-5182,
3. Lopez-Marcano, Sebastian, Eric L Jinks, Christina A. Buelow, Christopher J. Brown, Dadong Wang, Branislav Kusy, Ellen M Ditria, and Rod M. Connolly. "Automatic detection of fish and tracking of movement for ecology." Ecology and Evolution, 11, No. 12, 2021, pp.8254-8263.
4. Iqbal, Usama, Daoliang Li, and Muhammad Akhter. "Intelligent Diagnosis of Fish Behavior Using Deep Learning Method", Fishes, 7, No. 4, 2022, 201, pp. 1-9.
5. Kandimalla V, Richard M, Smith F, Quirion J, Torgo L and Whidden C, “Automated Detection, Classification and Counting of Fish in Fish Passages with Deep Learning”, Frontiers in Marine Science Volume 8, 2022, 823173, pp. 1-15.
6. Marrable D, Barker K, Tippaya S, Wyatt M, Bainbridge S, Stowar M and Larke J, “Accelerating Species Recognition and Labelling of Fish from Underwater Video with Machine-Assisted Deep Learning”. Frontiers in Marine Science Volume 9, 2022, pp.1-11
7. Cui, Suxia, Zhou, Yu, Wang, Yonghui, Zhai, Lujun, “Fish Detection Using Deep Learning”. Applied Computational Intelligence and Soft Computing, Volume 2020, 2020, pp. 1-13.
8. Ahsan Jalal, Ahmad Salman, Ajmal Mian, Mark Shortis, Faisal Shafait, “Fish detection and species classification in underwater environments using deep learning with temporal information”, Ecological Informatics, Volume 57, 2020, 101088.
9. Pauzi, S. N., M. G. Hassan, N. Yusoff, N. H. Harun, AH Abu Bakar, and B. C. Kua. "A review on image processing for fish disease detection." In Journal of Physics: Conference Series, Volume 1997, No. 1, IOP Publishing, 2021, p. 012042.
10. Chakravorty, Hitesh, Paul, Rituraj and Das, Prodipto., “Image Processing Technique to Detect Fish Disease”, Volume 9, Issue 2, 2015, pp. 121-131
11. Wang, Zhen, Haolu Liu, Guangyue Zhang, Xiao Yang, Lingmei Wen, and Wei Zhao, "Diseased Fish Detection in the Underwater Environment Using an Improved YOLOV5 Network for Intensive Aquaculture", Fishes 8, No. 3, 2023, pp. 169.
12. Ram Chandra Barik, Lavin A Kanuga, Lopamudra Mishra, Ankit Kumar Panda and Samarendra Nath Panda, “Spot Disease Identification using unsupervised Machine Learning based Image Segmentation with its Remedial Solution in Aquatic Fauna", Journal of Survey in Fisheries Sciences, 10 (2), 2023, pp. 912-922.