12-2023

CLASSIFICATION OF LARGE SCALE FISH DATASET BY DEEP NEURAL NETWORKS

Priyanka Adapa

CLASSIFICATION OF LARGE SCALE FISH DATASET BY DEEP NEURAL NETWORKS

A Project

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

In

Computer Science

---

by

Priyanka Adapa

December 2023

CLASSIFICATION OF LARGE SCALE FISH DATASET BY DEEP NEURAL NETWORKS

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

by

Priyanka Adapa

December 2023

Approved by:

Dr. Jennifer Jin, Advisor, School of Computer Science

Dr. Yan Zhang, Committee Member

Dr. Ronald Salloum, Committee Member

ABSTRACT

The development of robust and efficient fish classification systems has become essential to preventing the rapid depletion of aquatic resources and building conservation strategies. A deep learning approach is proposed here for the automated classification of fish species from underwater images. The proposed methodology leverages state-of-the-art deep neural networks by applying the compact convolutional transformer (CCT) architecture, which is famous for faster training and lower computational cost. In CCT, data augmentation techniques are employed to enhance the variability of the training data, reducing overfitting and improving generalization. The preliminary outcomes of our proposed method demonstrate a promising accuracy level of 100% over the Large-scale fish dataset, with the potential for real-time deployment in aquatic monitoring systems. Furthermore, this work proposes avenues for future research in the domain of fish classification.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to the individuals who have played a pivotal role in the completion of my graduation project titled "Classification of large-scale fish dataset by deep neural networks"

First and foremost, I thank the almighty for providing me with the strength, determination, and opportunities to pursue my academic goal. His blessings have been a guiding light throughout this endeavor.

I am writing to express my gratitude to Dr. Jennifer Jin my Advisor, for her continuous support to create this research work as the best version. Moreover, I want to thank my committee members Dr. Yan Zhang and Dr. Ronald Salloum for believing in me and agreeing to join the committee.

I would also like to acknowledge the collective influence of all the faculties and mentors who have motivated and inspired me over the years. Their dedication to nurturing our knowledge and curiosity has been a driving force behind this project's success.

My deepest gratitude goes out to my parents, whose unconditional love and encouragement have been the foundation of my academic pursuits. Their belief in me has been an endless source of motivation, and I'm profoundly thankful for their support. Lastly, but certainly not least, I want to acknowledge my friends who have stood by me with their endless support and encouragement.

In conclusion, the successful completion of this project is a result of the collective support, guidance, and encouragement of all those mentioned above. I

am deeply appreciative of their contributions and feel fortunate to have such a strong support network.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

Background

Over the years, the revelation of diverse aquatic life comprising thousands and thousands of species has left the human race amazed. These species are observed across the world in different kinds of water bodies like ponds, lakes, rivers, and oceans. Among these species, fish occupies a significant place. Fish are indeed an important part of the aquatic ecosystem, but they are also relied upon for sustenance by millions of people who are omnivores by nature. The ability of a machine to precisely recognize and classify fish is beneficial for the conservation of biodiversity and the management of fisheries. Illegal fishing practices involving endangered fish species are also preventable with efficient fish classification techniques. Apart from that, it will be helpful for researchers studying aquatic ecosystems. The conventional approach for fish classification was to use manual methods that were dependent on morphological characteristics. But, the execution of these methods is not only prolonged and tedious but also less accurate and efficient. The exploration of artificial intelligence to study ecology is still in its initial stages. But, with the advent of technology, employing machine learning and deep learning approaches has proven fruitful for fish classification tasks. The growth of cutting-edge technologies like big data and artificial intelligence in recent times has ensured efficient classification with accurate prediction outcomes.

Fish are vertebrates found in water bodies, and are available in a large variety of shapes, sizes, color, and behavior. This heterogeneity extends to their ecological roles, habitats, and life histories. The first challenge encountered in studying these characteristics of fish is collecting quality data in the form of images or videos in large numbers. The traditional way is by filming underwater videos by humans. This strategy is invasive and, therefore, not capable of expressing the normal state underwater. Also, the data collected in this manner is not sufficient for analysing and comprehending the behavior of the aquatic environment. The installation of high-definition cameras capable of remotely recording videos underwater in the absence of human intervention is now possible in the age of technology [1]. This has helped marine scientists study the underwater environment and conduct research on methods of managing and conserving aquatic ecosystems [2]. In the beginning, image frames are extracted from the videos captured underwater, and then each frame goes through the steps of preprocessing. The first part of fish detection and classification deals with acknowledging the presence of fish in an image frame. Edge detection algorithms are employed for this purpose. Once a fish is detected, the region of interest is segmented from the background.

Further features are extracted from the fish images to proceed with classification. In the conventional way, the features are generally geometrical features like shape and size, color and texture features, or a hybrid of both. Machine learning (ML) and deep learning (DL) technologies have gained massive popularity in recent times, promising to revolutionize the field of fish classification by addressing the challenges of traditional methods. Machine learning and deep learning are subfields of artificial intelligence that make use of various algorithms to learn patterns hidden in data. Machine learning algorithms rely on handcrafted features.

On the other hand, deep learning often relies on neural networks inspired by the structure and workings of the human brain. It is capable of automatically learning meaningful features from raw data. Deep learning architectures such as CNN, which takes images as input, are now trending because of their impressive performance in image classification and the computer vision domain.

CNN is widely used in fish classification tasks as it has proven to be a prevalent tool in computer vision. CNN efficiently learns the complex relationship between input and output with the help of convolution and pooling layers. Transfer learning is a relatively new yet very impactful concept in the area of deep learning, where a model trained for a particular task is adapted or fine-tuned to carry out other related tasks. In addition to boosting the performance of object detection and classification models, transfer learning effectively reduces the training time and computational resources. In most of the work done till date, fish classification is done after segmenting the fish images from the ground truth. This goal is achieved by utilizing the deep learning-based architecture or hand crafted segmentation tool. The aim of this section is to render an overview of fish classification and its importance, along with challenges associated in this area of research.

Motivation

Fish classification is important to study aquatic ecology and deal with certain real-world problems, including the extinction of endangered fish species, the sustenance of humans, and so on. However, to design an efficient fish classification system, there are numerous challenges that are necessary to be addressed. The general approach to fish classification explores the hand-engineered features that are not efficient enough. There are limited investigations in this domain incorporating deep learning architectures that are capable of learning patterns directly from the data and producing high accuracy for fish classification. This work is motivated to look out for CNN-based deep learning solutions and improve the performance of fish classification over the existing methods. The prime goal is to do classification without the use of segmentation. This work also aims to reduce the computational cost without lowering the accuracy of the system. The idea is to built a convolutional transformer oriented model such that the self attentional layer of the transformer identifies different features of the fishes without compromising the accuracy of the network. This transformer based model would also help to reduce the computational overhead and enhance speed and accuracy of the model.

Problem Statement

In this work, the objective is to explore and evaluate deep learning solutions with the goal of boosting the accuracy and efficiency of fish classification, surpassing the current literature. Considering the high variability observed in fish species, learning intricate features is a major challenge. This challenge is addressable through the application of advanced deep learning techniques. This work not only attempts to improve the accuracy of fish classification but also contributes to the development of more robust and adaptable solutions for the broader field of species identification in diverse ecosystems.

This section is dedicated to providing a detailed description of the fish classification task. The hierarchical steps involved in this task are thoroughly discussed. The challenges associated with this problem in the domain of computer vision are highlighted. Furthermore, potential solutions to these challenges are also conferred. Based on this discussion, a problem statement is outlined. This work aims to perform classification of fishes without segmenting the images. The primary objective is to classify various classes of fishes with less overhead in the computation cost and also maintaining speed and accuracy. In the next chapter, a detailed study is made on the existing work.

CHAPTER TWO

OVERALL DESCRIPTION

System Requirements

This section describes the System Requirements in term of software specification and hardware specifications.

Hardware :

1. RAM 12 GB + (At least)
2. A100, P100, Tesla K10, NVDIA RTX 3050 and above GPU required.
3. GPU memory: 12 GB +
4. CPU: i5 13th gen or higher
5. Operating System: windows, ubuntu (any linux) or MacOS

Software:

1. Jupyter Notebook
2. Python 3.7+ (language for the development)
3. Tensorflow 2.0 and above (Neural network backbone)
4. Keras (the model is developed on the base of tensorflow and keras)
5. Tensorflow addons (for optimizer)
6. Numpy (for scientific and complicated array operations)
7. Matplotlib (visualization library)
8. Seaborn (visualization library)

9. Google cloud platform for resource constraints.

    a. Google Colab

       This platform offers rich resources in terms of memory RAM and GPU also most of the libraries are pre-installed.

    b. Google Drive

       The best way to store data and retrieve data from and to google colab is the drive.

CHAPTER THREE

LITERATURE SURVEY

In this chapter, a detailed survey is made to investigate the existing works done on the field of classification of marine fishes. The limitations of the existing works made can be eliminated by proposing a better and novel architecture. Hence, this chapter deals with the work done by different researchers in the field of fish classification.

In fish classification, the first step to carry out on the dataset is pre-processing. Pre-processing includes adjusting contrast, grayscale conversion, scaling, noise removal, and emphasizing the foreground area of interest of the image. Further enhancing the fish images [3] as a part of pre-processing increases the reliability of the classification process.

Geometrical feature extraction from fish images becomes easier with the anchor points method, which is useful in taking measurements of length and orientation [8]. Besides statistical features like variance, standard deviation, homogeneity, etc., texture-based features also became popular for fish classification tasks. Gray-level co-occurrence matrix (GLCM) is one such texture feature extraction technique employed by researchers to attain a higher degree of accuracy [9,10]. Many researchers attempted to perform fish classification using hybrid features obtained by combining shape, texture, and color-based features with the expectation of achieving higher classification accuracy. Feature selection algorithms are also proposed to determine effective feature sets [11].

However, hand-engineered features are not always reliable, as they are seldom able to represent the complex patterns hidden in data. In a traditional machine learning approach by [12] fish classification is performed using support vector machines. The proposed model overperformed K-NN, ANN, and k-means classification models with 74.32% recognition accuracy. Six geometrical features were extracted manually to discriminate between two classes. Multiple research studies have been conducted for fish image classification, claiming the superiority of SVM and decision tree classifiers over other traditional classification models [7,10,13]. However, these models suffer from overfitting issues while trained with a large number of samples. Scalability became a problem in backpropagation algorithms used for fish classification [15,16]. P.I. Santosa & R.A. Pramunendar [17] emphasized on the quality of image feature extraction and proposed Grey Wolf Optimizer. Their proposed method was faster than Genetic Algorithm and Particle Swarm Optimizer based classification techniques.

In recent years, researchers have observed a shift from conventional machine learning approaches to deep learning approaches to solve numerous problems related to the computer vision domain. The reason behind this is the ability of deep learning architecture to mimic human cognitive ability. Fish segmentation and recognition tasks have also attained significant improvement in performance with deep learning architectures. Convolutional neural network-based architectures gained popularity for recognition tasks due to their efficacy in learning local features from data that requires the least pre-processing. The multiple layers of abstraction help CNN learn sophisticated patterns hidden in image data that are not possible with handcrafted feature extraction methods. In some instances, CNN architecture trained with a large number of image samples even surpassed the accuracy of human recognition [18]. However, high-end processing units are essential to deal with huge data costs effectively. Some researchers experimented with important parameters like activation functions, filter size, and the number of convolutional layers [19-20]. Transfer learning algorithms pre-trained on ImageNet are found to be beneficial in dealing with situations when limited training samples are available [21-23]. A lightweight AlexNet [24] architecture, implemented by Iqbal et al. for fish classification, achieved promising classification accuracy.

A publicly available large scale fish dataset was created [31] to investigate the performance of several handcrafted features compared to CNN based features. Apart from fish class recognition, segmentation is also performed. The manually extracted contrast and energy features attained remarkable result with SVM classifier.

The YOLO neural network architecture was adopted by Minsung Sung et al. [25] for fish detection and classification. Their approach resulted in 93% classification accuracy. Later, Mengfan Wang et al. [26] developed YOLOv2, a state-of-the-art detector, and claimed to attain better performance compared to YOLO. They have modified and performed optimization on the original Darknet-

19 architecture. The impotence of the original architecture to detect objects that are smaller compared to the background is dealt with in this model. This model attained a mean average precision (MAP) value of 0.912 with faster speed and optimal computational cost. Autoencoder models were developed by A. Banerjee et al. [27] to differentiate three varieties of carp fish that look similar in terms of size and shape. Three different autoencoder architectures, namely the Simple Autoencoder, the Deep Autoencoder, and the Deep Convolutional Autoencoder, were implemented. Their Deep Convolutional Autoencoder technique with SVM's latent representation overperformed well-known transfer learning models by achieving an accuracy of 97.33%. Despite impressive performance, their architectures are limited to very few classes of fish and incur higher computational costs.

A large scale dataset called FishNet [4] encompassing more than 94000 images was created and introduced in [5] for performing fish detection, fish class prediction and fish habitat prediction tasks. Behavioural characteristics of fish were also studied. They employed ResNet and vision transformer based architectures along with ConvNeXt for fish class prediction and habitat recognition. The ConvNeXt architecture outperformed other approaches. Furthermore, they finetuned the pre-trained models YOLOF [6] and TOOD for detecting presence of a fish in an image.

A deep CNN architecture is proposed [7] where principal component analysis, binary hashing and histogram are used in consecutive layers followed by spatial pyramid pooling for extracting efficient features from fish images. Finally, SVM classifier is employed to achieve better performance and reduce false positives.

The problem of class imbalance has been attempted to be solved using the class-aware loss function in [14]. They implemented MobileNetv3 and VGG16-based fish classification techniques over multiple datasets to show the efficiency of their proposed method.

IsVoNet8 [28], a CNN based architecture, was proposed to perform fish classification using the large scale fish dataset. Their model overperformed ResNet50, ResNet101, and VGG16 models with an accuracy of 98.62%.

In this chapter, a detailed literature survey is done. All the works done till date uses either statistical or deep learning based tool for the purpose of recognition of fish data. These architectural designs consume more computational resources compromising the speed. These big architectural designs not only compromise speed but also generates a greater number of false positives. In this project, the goal is to optimize the speed and accuracy such that the number of false positives reduces. In the next chapter, the dataset and uses in this project is detailed.

CHAPTER FOUR

ALL ABOUT DATA USED.

The data used in this fish classification project is described in this section. The major steps are involved in this project are data and its uses, the architecture development and model evaluation.

Dataset

The choice of the dataset is very much crucial in any data science project. The dataset [31] used here in this implementation is the “Large Scale Fish dataset” which is a publicly available dataset. This dataset can be downloaded from Kaggle. The dataset contains two broad categories- augmented and non-augmented. The augmented category contains 1000 images per class along with their segmented ground truth fish image.

There are 9 classes in total. The original size of the images is (2832, 2128) and (1024, 768). The images are then resized into (590, 445) dimension respectively. The resized images are flipped and rotated in order to generate augmented images. Also, the images are 3-channel RGB images. Image augmentation or data augmentation has become widely popular due to the increased diversity of class data. The augmentation technique ensures that the machine learns from various orientations and diverse class data.

Data Selection

In this project, the data is selected from the augmented category in order to classify different classes of fish. This is done due to increase the diversity in the dataset. The colored fish images are taken and processed for the next step. The ground truth segmented images are not considered in this project

This is the text for the test chapter. This is the text for the test chapter. This is the text for the test chapter. This is the text for the test chapter. This is the text for the test chapter. This is the text for the test chapter.

Data for Training

In the fish data classification project, training data is taken from the data pool taken into consideration in the data selection stage. The training data is randomly picked up from the pool of data. The percentage of training data used in this project is in the ratio of 80% of the selected data.

The data for training plays an important role while optimizing the architecture's performance. Also, the training data is selected in such a way that there are no duplicate samples. The total samples used for training is 7200.

Another method for selection of training data is using k-fold cross validation. This k-fold cross validation ensures better optimisation and is more accurate than other method. But due to limited GPU resources in google colab, in this project random selection of data is done.

Data for Testing

The test data helps to determine the model's performance after the training phase. To evaluate model's performance and visualize the rate of false positives, the test data is used. In this experiment random 20% of the overall data is considered as test dataset. The test and train set are mutually exclusive to ensure unbiased evaluation of the model's throughput.

Validation Data

The validation data is taken from the pool of training data. It is done to validate the training data over each and every epoch. The validation data taken for this experiment is 10% of the training data. Also, the validation is used to visualize if there exists overfitting or not.

This chapter describes the use of large scale fish dataset. The first section details the dataset, its dimension and size. In the remaining section, the data distribution is depicted. The dataset is resized and split into different sets. Each of these sets are used for different purposes i.e. training, testing and validation. Further, data augmentation is performed to ensure that there are enough data for the implementation. The data augmentation is discussed in the chapter 5 along with the necessary concepts that are used in this implementation.

CHAPTER FIVE

ARCHITECTURE PROPOSED

Before we dive into the used architecture, we will investigate some of the core components of the architecture and its original form and how it varies from the proposed method.

Transformer

Transformer [29] based networks are widely used nowadays. It is a special type of neural network which is widely used in large language models. Due to the tremendous capability of transformers, the computation cost and speed of the network are optimized when compared to conventional convolutional neural networks. The popularity of the transformer network is due to its self-attentional layers.

The transformer network relies on an encoder-decoder framework. In the encoder block, embeddings of the multi-head self-attentional layer with a fully connected layer and residual shortcuts with layer normalization. On the other hand, the decoder block also uses a self-attentional mechanism on the output embedding generated by the encoder block. Further, these attentional layers do the mapping from a set of (key, value) to the corresponding output. The resultant is the weighted addition of all the values.

The transformer encoder generates tokens that are sequential. These tokens are passed to the decoder section, where the multi-head self-attentional mechanism, with the help of the SoftMax activation function, does the task of recognition.

In a nutshell, the transformer model takes the input and converts it into a vector. These vectors are embedding vectors and are formed by combining the input with the positional encodings. Further, these vectors are fed to the encoder block, where all the input sequences are attended by the attentional layer. The decoder also takes the positional vectors and attends the input sequence. The resultant embedding from the decoder is passed through the multi-layered neural network with a SoftMax activation function at the output layer, which does the task of prediction. The transformer network was designed for the purpose of neural language translation. However, due to the need for a more powerful tool to deal with complicated problems, different researchers tried to introduce convolution along with the concept of transformers. In the next sub-section, the use of convolution along with a self-attention based transformer is detailed. Further, in this project, the convolution-based self-attentional transformer network is used with certain modifications.

Compact Convolutional Transformer

The idea of a compact convolutional transformer [30] came from a transformer when there is a need to deal with smaller datasets with less overfitting, maintaining speed and accuracy. The compact convolutional transformer works on the principle of tokenization and prediction with the multi-head self-attentional mechanism.

The input is a 3-channel RGB image, which is fed to a convolutional block with 64 numbers of kernels along with max pooling to extract features. The above step is again repeated with 128 numbers of kernels and max pool. The extracted features are then reshaped to form 1-D vectors. These vectors are also called tokens. The process is called the positional embedding of vectors. These embeddings are then passed to the transformer encoder with a multi-head self-attentional layer and sequence pooling followed by a linear layer of multi-layer perceptron with SoftMax to perform prediction. The sequence pooling pools the whole range of tokens that are generated by the encoder.

The use of a convolutional tokenizer enables the generation of more dominant tokens while preserving the local semantics. The transformer encoder uses layer normalization at each level with gelu as an activation function and also the dropout of unutilized neurons. The number of transformer heads used are 2, along with 128 hidden layer dimension. In the convolutional blocks, relu is used as an activation function. The optimizer used for training is the AdamW algorithm, which uses the adaptive method of estimation in order to compute the moments. The number of transformer layers is 7. This architecture was designed to perform better when there is a limited resource in terms of computations (GPU) or when the data pool is too small for a neural network task. In the next sub-section, the modified version of the compact convolutional transformer is presented.

Proposed Architecture

In this section, the above architecture is modified with certain pre-processing steps, and the architecture level changes. The proposed architectural diagram is shown in the figure 1 below along with the overall system pipeline in the figure 2. The figure 1 shows the difference between original CCT and modified CCT. The figure 2 shows the different stages of the implementation of the architecture.

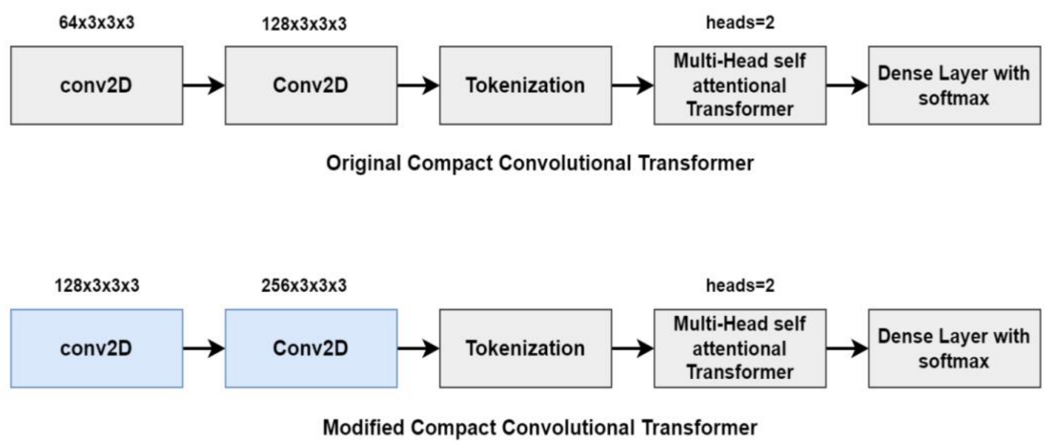

Figure 1 The Proposed Architecture

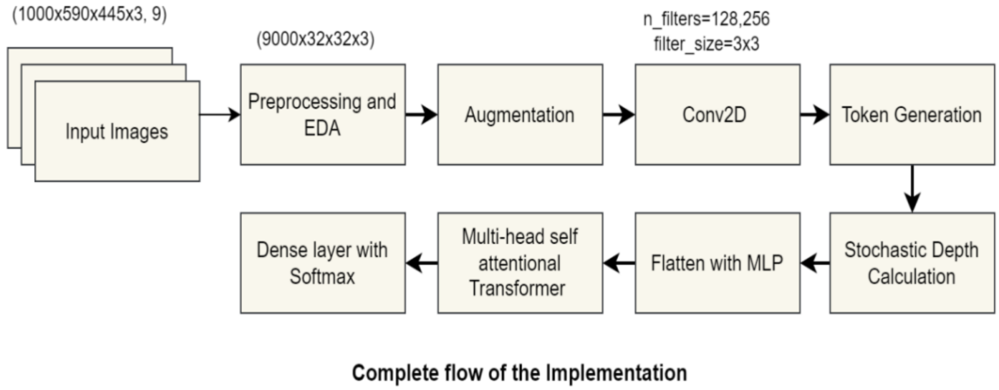

Figure 2 The Flow of the Project

The implementation starts with the pre-processing step, and the employment of data augmentation is done. The process of increasing class data by mean of a special technique called sampling. There are two types of sampling- random over sampling and under sampling. These types of techniques are employed when there are few samples in a class compared to other classes. The oversampling technique states that replicas of the samples are created to ensure a uniform distribution of data across all the classes. In under sampling process, the samples are deleted randomly from the majority classes. Another instance is when the data pool is balanced but not enough to represent diverse information. In these cases, there comes the role of data augmentation.

Image data augmentation is the process of replicating images with different parameters such that there is sufficient samples in the data pool for training. The need for data augmentation is to prevent any chances of overfitting or the original data pool is too small; it is also used to enhance the model's efficiency; finally, it is used to reduce the cost of operation by means of cleaning raw image data. But if there is bias associated with the raw data, then the bias persists in the new dataset, too. There are various transformations associated with image augmentation, such as geometric and color space transformations, blending images, and introducing noise or sharpness in the image. In this project, we would limit our discussion only on geometric transformation. The transformation done is scaling the image, flipping the image horizontally, and cropping the image.

The next step is the architecture. In the above architecture, the first modification done is in the convolutional blocks. The number of kernels is changed to 128 and 256. Secondly, the transformer projection dimension is changed to 256. The base compact convolutional transformer with the above modification is utilized. Further, stochastic depth estimation is done to ensure that the training network's depth is reduced and, hence, the computation cost.

Summarizing the above, the principles of transformers and their capabilities are discussed at first. Secondly, by employing transformers, how compact convolutional transformers evolved and their capabilities. Thirdly, the proposed architecture, where we saw how data augmentation works and its needs. Finally, the modifications were done on the convolutional transformer. In the next chapter, we will concentrate our discussion on the detailed implementation of the code.

This chapter describes the necessary concepts used for the implementation of the fish classification project. The concept of transformer was used along with convolution to fulfil the aim of the project. The use of a transformer, along with stochastic depth calculation, makes the network better in terms of using resources and reduces the computation cost and complexity of the network. The network not only performs better but also it reduces the chances of overfitting. The network also performs efficiently in comparison to the state-of-the-art methodologies and traditional deep learning techniques. In the next chapter, the code implementation is detailed, along with snapshots of the necessary outputs.

CHAPTER SIX

DETAILED IMPLEMENTATION

The step wise implementation is detailed in this section. In this project, the Keras library is used in order to use the neural network framework. The whole implementation is divided into different phases.

The phases are Loading data and visualizing the data, Selecting data for training purposes, encoding of multi-class data, model building, data augmentation, stochastic depth estimation, training, testing, confusion matrix evaluation, and classification report generation.

Let’s discuss each of them in detail. At first, the dataset is stored in Google Drive and then loaded into Google Colab with the help of the glob library. The glob library is used to load data from any local or global storage. After loading the data from the drive, the images are resized into 32x32x3 with the help of the Keras library. Also, the loaded data is stored into a list along with the corresponding target to another list. The list of data and target values are then converted into a numpy array. Furthermore, the data is visualized class wise, along with a bar plot representing the data distribution across all the classes.

The target variables are one hot encoded for categorical classification. Further, the data and the target are split into train and test sets. Hence, the data is ready to be trained and tested.

The architecture pipeline is initialized with parameters with 2 numbers of convolutional layers, with the number of filters being 64 and 128; the projection dimension of the transformer network is 256 along with 2 multi-head attentional mechanisms; 2 transformer layers along with stochastic depth rate of 10% is also added. The learning rate of the network is initialized to 0.001, along with a weight decay rate of 0.0001. The batch size of 30, along with 200 iterations, is also added for parameter tuning.

The pre-processing step includes data augmentation with cropping of the image, horizontally flipping the image, and rescaling the image. The augmented data are passed to the tokenization network.

The implementation of architecture begins with the tokenization phase. Rich convolutional tokens are generated with the help of cascading convolutional layers with max pooling and then flattening the feature vectors into 1-D vectors. The sequence length and position of embeddings are calculated. Furthermore, the transformer layer is initialized with a multi-head attention layer along with layer normalization. Also, to reduce the depth of the network such that the complexity in terms of computation cost is also reduced, stochastic depth estimation is done. Finally, using a multi-layered perceptron and gelu activation function, the representational features are extracted. These representational features, along with weights of the attention mechanisms, are then multiplied with matrix multiplication. The attention mechanisms use SoftMax. In the end, 9 perceptrons corresponding to different classes along with SoftMax are used for the prediction of classes.

Furthermore, the model is stored and also the training parameters, along with snapshots of each and every epoch where the accuracy of validation and training data are stored using the history variable. With this history variable, the loss incurred at every epoch with time is graphically represented.

The one-hot encoded test target data is taken along with predicted test target data, and a confusion matrix is plotted with the help of Seaborn. Further, a classification report is also generated stating class-wise recall, precision, accuracy, and f1-score. The above evaluation metrics will again be elaborated in Chapter 7.

In this chapter, the detailed code implementation is presented. The data visualization, as well as data distribution, is done. The use of data augmentation for pre-processing is discussed. The transformer model is implemented with the help of a convolutional tokenizer. The network's complexity is reduced with the help of stochastic depth computation. The training set is validated with the validation set. SoftMax activation is used at the output for the categorical classification. In the next chapter, the results are elaborated with the help of different metrics.

CHAPTER SEVEN

EVALUATION

This section describes the evaluation of the results of the fish classification project. This chapter is divided into 4 sub-categories: a) The data representation and distribution, b) Metrics for performance evaluation, c) Proposed result vs state-of-the-art results.

The Data Representation and Distribution

The data is represented by plotting graph where the plot shows each of the rows corresponds to different classes. Also, the data distribution is uniform as the bar plot shows that the data in each and every class are equal. Hence, there is less chance of overfitting and bias. The class data and data distribution are shown in the figure 3 and figure 4 below.

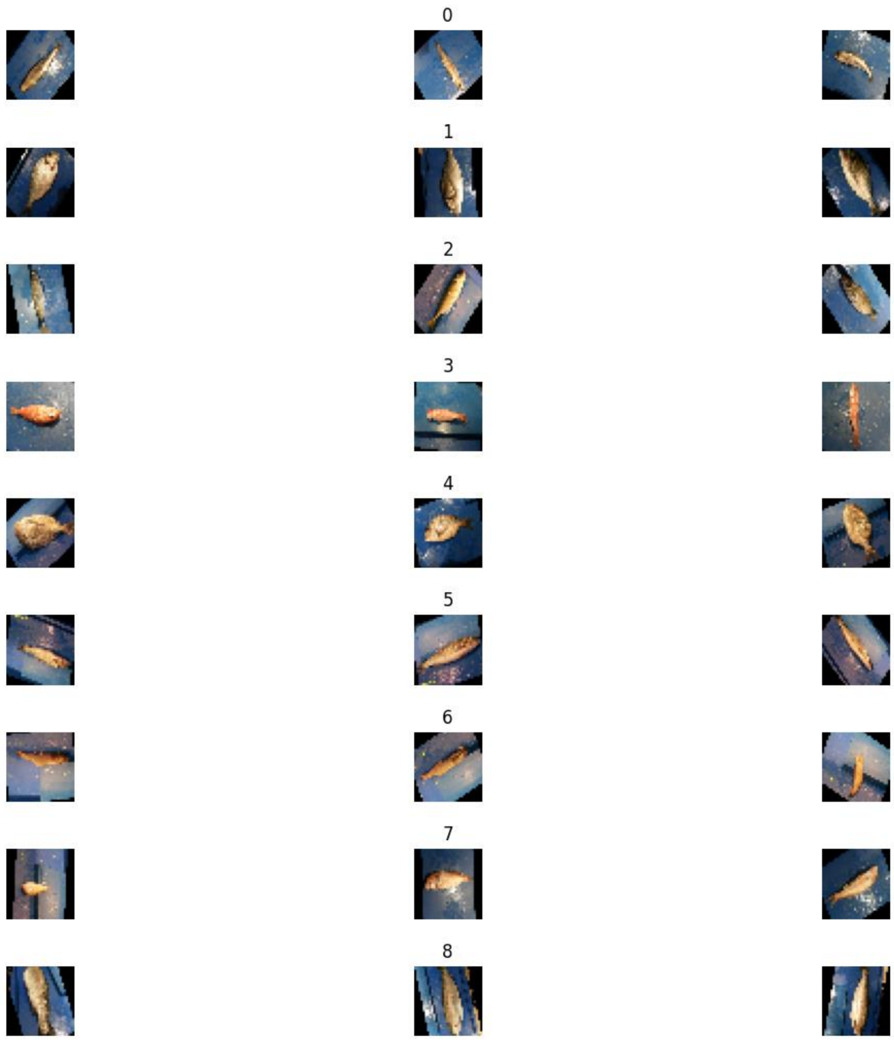

Figure 3 The Class Data

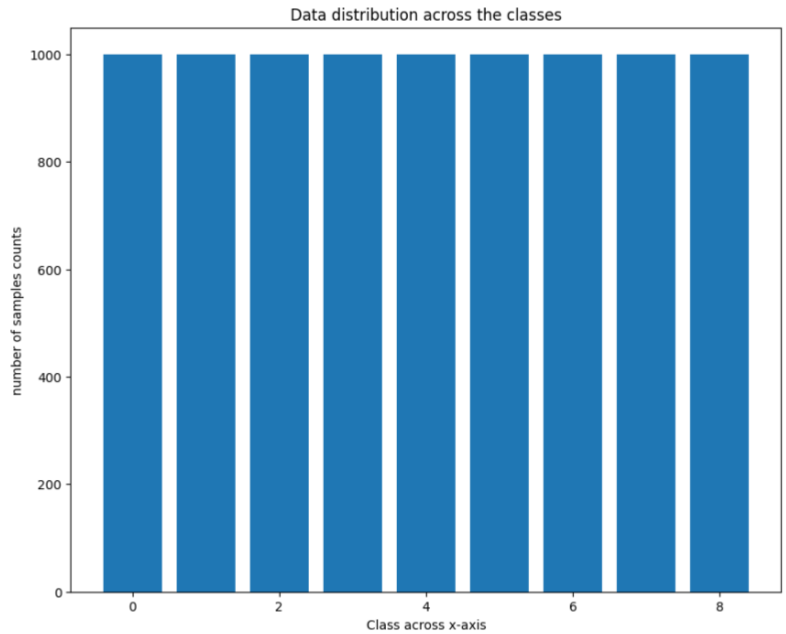

Figure 4 The Data Distribution

Metrics for Performance Evaluation

The metrics used for the evaluation of the architecture's performance are accuracy, precision, recall, f1-score. Accuracy Ac_{fish}, is the metric that determines how close a neural network forecasts the outcome. Recall Rc_{fish}, is the rate at which samples are identified correctly. Precision Pr_{fish}, is the count of the rate at which samples are identified correctly. F1-score fS_{fish}, is the harmonic mean of recall and precision.

All of the above metrics are computed with the help of confusion matrix. It is a matrix or a table to evaluate the efficiency of a recognition algorithm. This table summarizes and visualizes the number of true positives, false positives, true negatives, and false negatives. True positives are the cases if actual and predicted outcomes are same in terms of positive class. True negatives are the cases if actual and predicted outcomes are same in terms of negative class. False positives are the cases if actual and predicted outcomes are not same in terms of positive class. False negatives are the cases if actual and predicted outcomes are not same in terms of negative class. These true positive, true negatives, false positives and false negatives are also denoted by TP, TN FP, FN. Mathematically, the metrics are represented as follows:

$$Ac_{fish} = \frac{TP_{fish} + TN_{fish}}{Total\ test\ cases}$$

$$Pr_{fish} = \frac{TP_{fish}}{TP_{fish} + FP_{fish}}$$

$$Rc_{fish} = \frac{TP_{fish}}{TP_{fish} + FN_{fish}}$$

$$fS_{fish} = \frac{Pr_{fish}.Rc_{fish}}{Pr_{fish} + Rc_{fish}}$$

The accuracy obtained for the classification of fish data with large scale fish dataset is 100% for the test dataset. The confusion matrix and classification report are shown in figure 6 and figure 7. The loss graph is shown in figure 5.

The loss graph in figure 5 shows the change of training and validation loss with time and epochs.

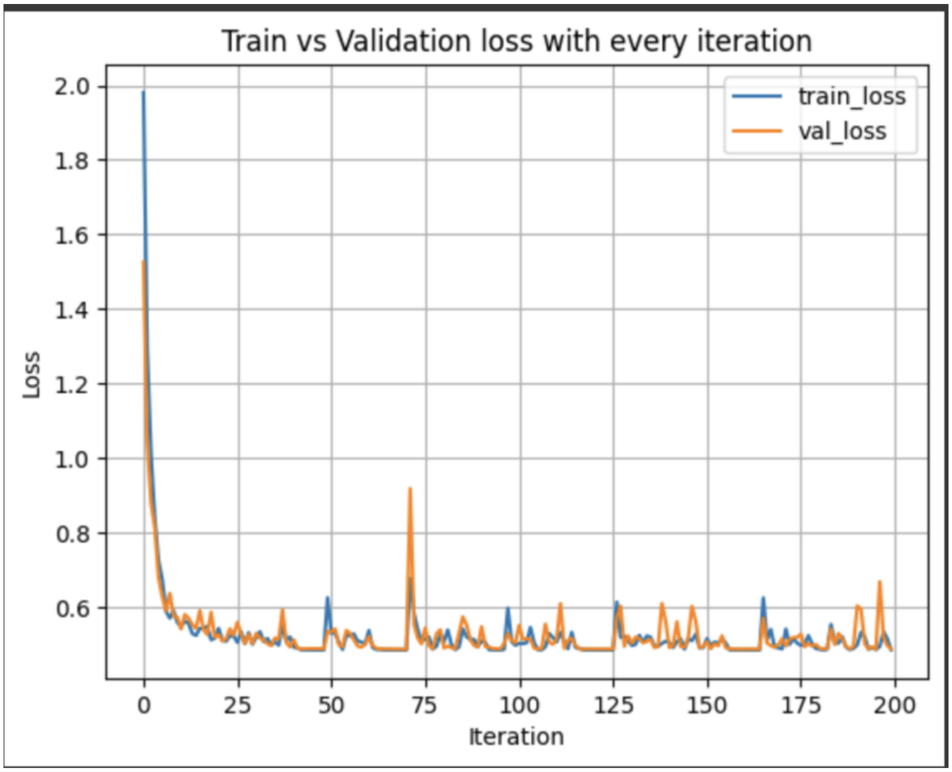

Figure 5 The Loss Graph

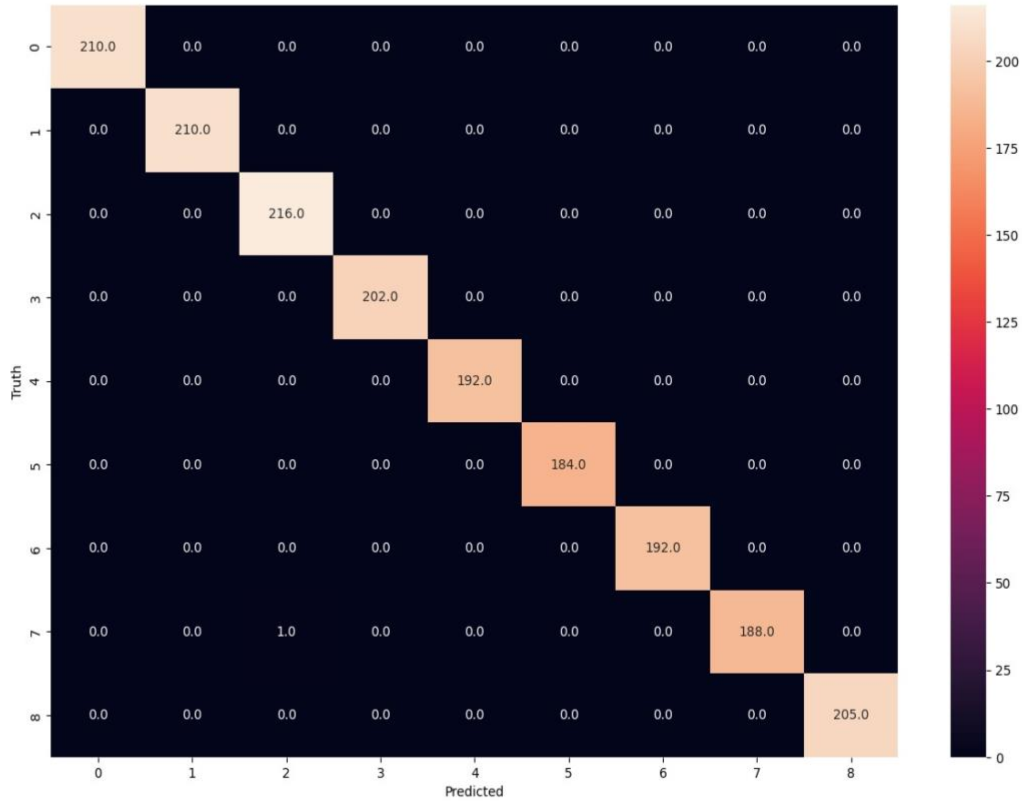

Figure 6 Confusion Matrix for Fish Classification

The diagonal elements of the figure 6 are the truth values. In the figure 7 support stands for the number of images that are test class wise and the class wise metrics are also presented.

	precision	recall	f1-score	support
0-Black Sea Sprat	1.00	1.00	1.00	210
1-Gilt-Head Bream	1.00	1.00	1.00	210
2-Hourse Mackerel	1.00	1.00	1.00	216
3-Red Mullet	1.00	1.00	1.00	202
4-Red Sea Bream	1.00	1.00	1.00	192
5-Sea Bass	1.00	1.00	1.00	184
6-Shrimp	1.00	1.00	1.00	192
7-Striped Red Mullet	1.00	0.99	1.00	189
8-Trout	1.00	1.00	1.00	205
accuracy			1.00	1800
macro avg	1.00	1.00	1.00	1800
weighted avg	1.00	1.00	1.00	1800

Figure 7 The Classification Report for the Fish Classification

Comparisons were made with the state-of-the-art methodologies, and it is found that the proposed architecture outperforms the literature as well as it is observed that the method reduces the overfitting issues associated with the dataset. The proposed novel network is better in terms of computation cost and speed and accuracy.

The results obtained from the implementation are discussed in this chapter. The system requirements are also stated in this chapter. The metrics for determining the efficiency of the model are also highlighted along with the formulation. The confusion matrix and classification report are also attached, along with comparisons. In the next chapter, the conclusion and future directions are discussed.

CHAPTER EIGHT

CONCLUSION AND FUTURE WORK

Deep learning as a cutting-edge technology has limitless possibilities when applied to real-world challenges, and it paves the way for a brighter, more informed future in marine biology and conservation. Fish classification plays an important role in learning about and conserving the aquatic ecosystem. This study aimed to explore the potential of a deep learning method in the field of fish classification that is lightweight and incurs less computational cost. The proposed fish classification system is bolstered by the power of the Compact Convolutional Transformer, a relatively new and less explored transformer architecture. Through extensive research, experimentation, and optimization, we have achieved a remarkable accuracy rate of 100% for fish classification using a Large-Scale Fish dataset, far surpassing the performance of many pre-trained models.

Looking ahead, there is potential for further refinement and expansion of this system. Incorporating additional data sources, enhancing the model's robustness in varying environmental conditions, and adapting it to real-time monitoring applications are all promising avenues for future research.

REFERENCES

[1]. Zarco-Perello, S. and Enríquez, S., 2019. Remote underwater video reveals higher fish diversity and abundance in seagrass meadows, and habitat differences in trophic interactions. Scientific reports, 9(1), p.6596.

[2]. Piggott, C.V., Depczynski, M., Gagliano, M. and Langlois, T.J., 2020. Remote video methods for studying juvenile fish populations in challenging environments. Journal of experimental marine biology and ecology, 532, p.151454.

[3]. Alsmadi, M., Omar, K. and Almarashdeh, I., 2012. Fish classification: fish classification using memetic algorithms with back propagation classifier.

[4]. Khan, F.F., Li, X., Temple, A.J. and Elhoseiny, M., 2023. FishNet: A Large-scale Dataset and Benchmark for Fish Recognition, Detection, and Functional Trait Prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 20496-20506).

[5]. Chen, Q., Wang, Y., Yang, T., Zhang, X., Cheng, J. and Sun, J., 2021. You only look one-level feature. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 13039-13048).

[6]. Feng, C., Zhong, Y., Gao, Y., Scott, M.R. and Huang, W., 2021, October. Tood: Task-aligned one-stage object detection. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 3490-3499). IEEE Computer Society.

[7]. Qin, H., Li, X., Liang, J., Peng, Y. and Zhang, C., 2016. DeepFish: Accurate underwater live fish recognition with a deep architecture. Neurocomputing, 187, pp.49-58.

[8]. Alsmadi, M.K., Tayfour, M., Alkhasawneh, R.A., Badawi, U., Almarashdeh, I. and Haddad, F., 2019. Robust feature extraction methods for general fish classification. International Journal of Electrical & Computer Engineering (2088-8708), 9(6), pp.5192-5204.

[9]. Wishkerman, A., Boglino, A., Darias, M.J., Andree, K.B., Estévez, A. and Gisbert, E., 2016. Image analysis-based classification of pigmentation patterns in fish: A case study of pseudo-albinism in Senegalese sole. Aquaculture, 464, pp.303-308.

[10]. Saberioon, M., Císař, P., Labbé, L., Souček, P., Pelissier, P. and Kerneis, T., 2018. Comparative performance analysis of support vector machine, random forest, logistic regression and k-nearest neighbours in rainbow

trout (oncorhynchus mykiss) classification using image-based features. Sensors, 18(4), p.1027.

[11]. Hnin, T.T. and Lynn, K.T., 2016. Fish classification based on robust features selection using machine learning techniques. In Genetic and Evolutionary Computing: Proceedings of the Ninth International Conference on Genetic and Evolutionary Computing, August 26-28, 2015, Yangon, Myanmar-Volume 1 (pp. 237-245). Springer International Publishing.

[12]. Ogunlana, S.O., Olabode, O., Oluwadare, S.A.A. and Iwasokun, G.B., 2015. Fish classification using support vector machine. African Journal of Computing & ICT, 8(2), pp.75-82.

[13]. Qin, H., Li, X., Liang, J., Peng, Y. and Zhang, C., 2016. DeepFish: Accurate underwater live fish recognition with a deep architecture. Neurocomputing, 187, pp.49-58.

[14]. Alaba, S.Y., Nabi, M.M., Shah, C., Prior, J., Campbell, M.D., Wallace, F., Ball, J.E. and Moorhead, R., 2022. Class-aware fish species recognition using deep learning for an imbalanced dataset. Sensors, 22(21), p.8268.

[15]. Badawi, U.A. and Alsmadi, M.K., 2014. A General Fish Classification Methodology using Meta-Heuristic Algorithm with Back Propagation Classifier. Journal of Theoretical & Applied Information Technology, 66(3).

[16]. Boudhane, M. and Nsiri, B., 2016. Underwater image processing method for fish localization and detection in submarine environment. Journal of Visual Communication and Image Representation, 39, pp.226-238.

[17]. Santosa, P.I. and Pramunendar, R.A., 2022. A robust feature construction for fish classification using grey Wolf optimizer. Cybernetics and Information Technologies, 22(4), pp.152-166.

[18]. Villon, S., Mouillot, D., Chaumont, M., Darling, E.S., Subsol, G., Claverie, T. and Villéger, S., 2018. A deep learning method for accurate and fast identification of coral reef fishes in underwater images. Ecological informatics, 48, pp.238-244.

[19].Varalakshmi, P. and Rachel, J.J.L., 2019, February. Recognition of fish categories using deep learning technique. In 2019 3rd International Conference on Computing and Communications Technologies (ICCCT) (pp. 168-172). IEEE.

[20]. Sarigül, M. and Avci, M., 2017. Comparison of different deep structures for fish classification. International Journal of Computer Theory and Engineering, 9(5), pp.362-366.

[21]. Saleh, A., Laradji, I.H., Konovalov, D.A., Bradley, M., Vazquez, D. and Sheaves, M., 2020. A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. Scientific Reports, 10(1), p.14671.

[22]. Jin, L. and Liang, H., 2017, June. Deep learning for underwater image recognition in small sample size situations. In OCEANS 2017-Aberdeen (pp. 1-4). IEEE.

[23]. Tamou, A.B., Benzinou, A., Nasreddine, K. and Ballihi, L., 2018. Underwater live fish recognition by deep learning. In Image and Signal Processing: 8th International Conference, ICISP 2018, Cherbourg, France, July 2-4, 2018, Proceedings 8 (pp. 275-283). Springer International Publishing.

[24]. Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.

[25]. Sung, M., Yu, S.C. and Girdhar, Y., 2017, June. Vision based real-time fish detection using convolutional neural network. In OCEANS 2017-Aberdeen (pp. 1-6). IEEE.

[26]. Wang, M., Liu, M., Zhang, F., Lei, G., Guo, J. and Wang, L., 2018, May. Fast classification and detection of fish images with YOLOv2. In 2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO) (pp. 1-4). IEEE.

[27]. Banerjee, A., Das, A., Behra, S., Bhattacharjee, D., Srinivasan, N.T., Nasipuri, M. and Das, N., 2022. Carp-DCAE: Deep convolutional autoencoder for carp fish classification. Computers and Electronics in Agriculture, 196, p.106810.

[28]. Volkan, K.A.Y.A., Akgül, İ. and TANIR, Ö.Z., 2023. IsVoNet8: A Proposed Deep Learning Model for Classification of Some Fish Species. Journal of Agricultural Sciences, 29(1), pp.298-307.

[29]. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems, 30.

[30]. Hassani, A., Walton, S., Shah, N., Abuduweili, A., Li, J. and Shi, H., 2021. Escaping the big data paradigm with compact transformers. arXiv preprint arXiv:2104.05704.

[31]. Ulucan, O., Karakaya, D. and Turkan, M., 2020, October. A large-scale dataset for fish segmentation and classification. In 2020 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-5). IEEE.