Underwater Image Classification using ML Techniques

Periseti Sai Ram Mohan Rao, Mahesh Vasimalla
Department of Computer Science Engineering
Indian Institute of Information Technology, Raichur,
Raichur, 584135, Karnataka, India
{maheshvasimalla333, sairam1911}@gmail.com

Neha Agarwal
Department of Computer Science Engineering
Indian Institute of Information Technology Raichur,
Raichur 584135, Karnataka, India
neha@iiitr.ac.in

***Abstract*— Underwater image classification is a compelling field within computer vision and marine science, leveraging the power of Machine Learning (ML) and Deep Learning (DL) to decipher the hidden realms of our oceans. This underwater image classification is essential for measuring the water bodies' health and quality and protecting endangered species. Further, it has applications in oceanography, marine economy and defense, environment protection, underwater exploration, and human-robot collaborative tasks. In this paper, we explore the significance and challenges of classifying underwater images, highlighting the unique characteristics of this environment, including low visibility, color distortion, and complex marine life diversity. This paper presents various types of deep learning and ML techniques for performing underwater image classification.**

***Index Terms*— PCA, GAN, ReLU, SVM-CNN, Transfer Learning, Neural Networks**

I. INTRODUCTION

In today's digital age, the ubiquity of images and visual data has revolutionized the way we interact with technology. In the age of visual data, where images are the language of machines, image classification serves as a bridge between the visual world and computational intelligence. Image classification, a cornerstone of this technological revolution, plays a pivotal role in the analysis of visual data.

This paper focuses on the underwater image classification. Underwater image classification aims to decode the secrets of the deep by enabling the automated analysis of images captured in aquatic environments. The underwater environment introduces factors such as reduced visibility, color distortion, and the presence of diverse marine species, all of which complicate the process of image analysis.

In recent years, the fusion of Machine Learning (ML) and Deep Learning (DL) techniques has emerged as a transformative force in this field. These methodologies have the capacity to not only identify and classify underwater objects, species, and habitats but also to enhance our understanding of marine ecosystems and contribute to conservation efforts. By leveraging ML and DL, we can efficiently process large volumes of underwater imagery, accelerating scientific discovery and environmental protection.

This paper delves into the world of underwater image classification, emphasizing the role of ML and DL in addressing the unique challenges posed by aquatic environments. We explore the state-of-the-art techniques, including Convolutional Neural Networks (CNNs) and transfer learning, which have propelled the field forward. Furthermore, we underscore the significance of robust and diverse underwater image datasets, as well as the importance of preprocessing methods tailored to this context.

II. CHALLENGES IN THE AUTOMATED CLASSIFICATION OF UNDERWATER IMAGES:

The automated classification of underwater images is faced with numerous challenges. The intensity of light is diminished due to energy loss during its propagation, resulting in low and variable illumination and visibility, particularly in deeper waters. Furthermore, ocean currents in contrast to still waters like ponds or swimming pools contribute to changes in luminosity. These alterations, along with impurities and suspended solids, give rise to intricate noise in underwater images, particularly those captured in the ocean. Additionally, these images exhibit low contrast and degraded edges and details. Moreover, the non-uniform spectral propagation causes color distortion depending on the distance. In order to address some of these limitations, the use of sophisticated yet expensive cameras is necessary. Here are some key challenges in automated underwater image classification datasets:

Limited Labeled Data- Collecting labeled data for underwater image classification is a labor-intensive task. The limited availability of diverse and well-labeled datasets hinders the training of robust machine learning models. Insufficient data can lead to overfitting, where the model

performs well on training data but fails to generalize to new, unseen data.

Species and Object Recognition-Identifying underwater species and objects is challenging due to the diversity of marine life and the potential for occlusions. Some species may have similar appearances, and certain objects may be partially hidden, making accurate classification difficult.

Dynamic Environmental Condition-Underwater conditions can change rapidly, including variations in water currents, temperature, and light levels. Models must be robust to these dynamic environmental factors to ensure consistent performance across different scenarios.

Sensitivity to Illumination Conditions- Illumination conditions underwater can vary significantly depending on the time of day, weather conditions, and water depth. Models need to be robust to changes in lighting for consistent performance. Addressing these challenges requires a combination of advanced computer vision techniques, domain-specific knowledge, and the availability of high-quality, diverse datasets. Hence we use preprocessing techniques to solve these challenges and data augmentation is a necessary and crucial strategy in underwater image classification to overcome challenges related to limited labeled data, enhance model robustness, and improve generalization to diverse underwater conditions.

This paper is structured as follows:SectionIII Proposed methodology .Section IV- Image Pre-processing techniques. In Section V- Data Augmentation. In Section VI -Transfer Learning Models. In SectionVII -Compare the results with other models and finally in SectionVIII- The conclusion.

III. Proposed Methodology

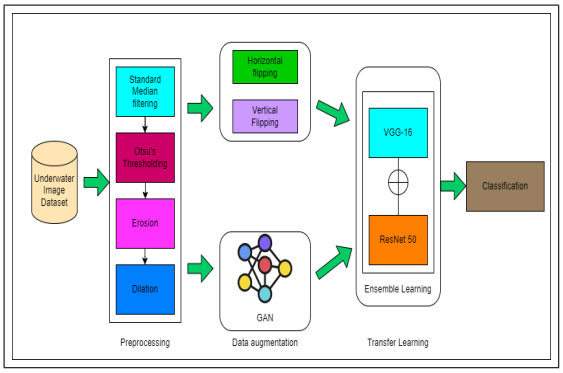

Fig. 1. Architecture

IV. Image Pre-processing Techniques:

Why is there a need for Image pre-processing? - Underwater images often suffer from issues such as poor visibility, color distortion, and environmental noise, therefore pre-processing them is absolutely essential. Pre-processing helps address these challenges and enhances the quality of images, leading to improved performance in image classification models. The lack of underwater image datasets requires the use of data augmentation and transfer learning. Transfer learning also reduces computational demands during training[2]. [13]For underwater computer vision, the image preprocessing is the most important procedure for object detection. Because of the effects of light scattering and absorption in the water, the images obtained by the underwater vision system show the characteristics of uneven illumination, low contrast, and serious noise and more which are mentioned in the II i.e., Challenges in the underwater image classification.

Fig. 2. Original Image before Pre-processing

We now discuss the techniques used to solve these challenges-

Standard Median filtering- it is a commonly used image preprocessing technique that aims to reduce noise and preserve edges in an image. Here in this proposed model, we used a 5x5 kernel and this kernel moved on the image such that the center of the kernel traverses all input image pixels. Features that are smaller than half the size of the median filter kernel are completely removed by the filter. It operates by replacing each pixel's intensity with the median value of the pixel intensities in its neighborhood, which is used to reduce the salt pepper noise i.e., (0 or 255 values) in the image we use standard median filtering[1]. When median filtering is applied to an underwater image it reduces the noise of the image but increases blurring in the image. To reduce these effects we can use dehaze, this technique is designed to reduce or remove the haze and blur effects and it aims to enhance the clarity and visibility of the image.

Fig. 3. Resultant image after applying Standard Median Filter on original image

We also used the **Otsu Thresholding,**which is an image segmentation technique used to find an optimal threshold for binarization. A criterion function is computed for intensity and that which maximizes this function is selected as the threshold. Otsu's thresholding picks the threshold value to minimize the intra-class variance of the thresholded black and white pixels. This technique separates an image into two classes: foreground and background or object and non-object according to the threshold values. This method is particularly effective when there is a bimodal distribution of pixel intensities in the image.

Algorithm[2]:
Step 1: Compute histogram for a 2D image.
Step 2: Calculate foreground and background variances (measure of spread) for a single threshold.
i) Calculate the weight of background pixels and foreground pixels.
ii) Calculate the mean of background pixels and foreground pixels.
iii)Calculate the variance of background pixels and foreground pixels. **Step 3**: Calculate "within class variance"

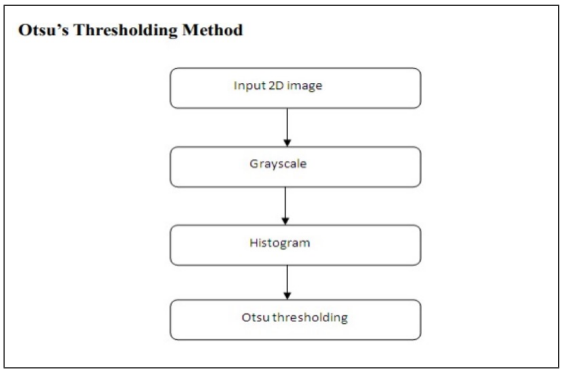

Fig. 4. Otsu's Thresholding Work Flow[2]

Fig. 5. Otsu's Thresholding Work Flow[2]

Erosion and **dilation** are the basic operations of morphological image processing The erosion process shrinks the foreground structures while the dilation process enlarges them. The performance of both operations depends on their structuring element shape. In this paper, the erosion and dilation operations are programmed using the **OpenCV** tools.

Morphological erosion is applied to binary images to reduce the size of foreground objects or to grayscale images for various image enhancement tasks. The operation involves the use of a structuring element (also known as a kernel) to modify the shape or size of objects in an image. Erosion is effective in reducing small-scale noise or thin structures in binary images and is also used to modify the shape or size of objects in an image. This technique is often followed by dilation in morphological operations.

Fig. 6. Resultant image after applying Morphological Erosion on Otsu's Thresholded image

Morphological Dilation is a versatile technique used in various image processing applications, including object fusion, boundary expansion, noise removal, and feature extraction. It is an essential operation in morphological processing pipelines, contributing to tasks such as image enhancement, segmentation, and pattern recognition. This process helps join the broken parts of the objects with a particular technique which further helps in modelling. Dilation is often followed by erosion in morphological operations, and the combination of these operations is known as closing.

Fig. 7. Resultant Image after applying Morphological dilation on the previous result image

V. Data Augmentation:

It is a technique commonly used in machine learning and computer vision to artificially increase the diversity of a training dataset by applying various transformations to the existing data. The main goal of data augmentation is to enhance the generalization ability of a machine learning model, making it more robust to variations and improving its performance on unseen data.

We used various data augmentation techniques like-

- Rotation: Rotating the image by a certain angle.
- Flip: Flipping the image horizontally or vertically.
- Shift: Shifting the image horizontally or vertically.
- noise injection.
- Random cropping.

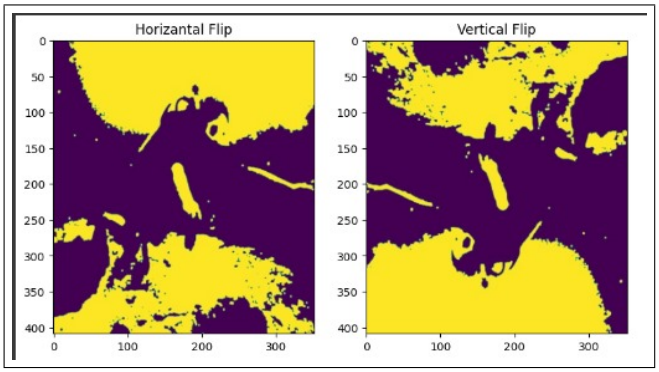

Fig. 8. Horizontal and Vertical flip of the preprocessed image

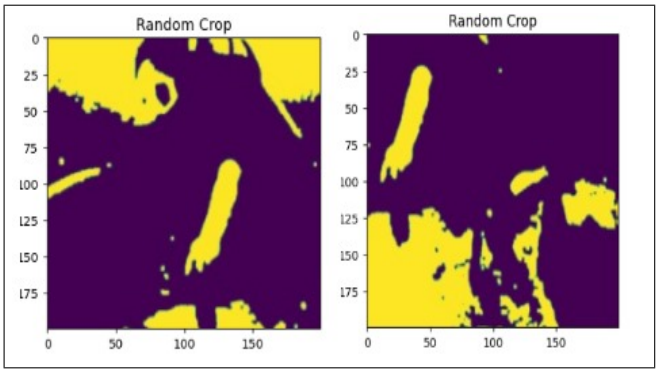

Fig. 9. Random cropped images of the preprocessed image

A. GANs-Generative Adversarial Networks

Both data augmentation and GANs contribute to enhancing the diversity and generalization of machine learning models, they serve different purposes and operate in distinct ways. Data augmentation focuses on expanding the training dataset by applying various transformations to the existing data whereas GANs are a class of generative models designed to generate entirely new and realistic data samples that resemble the training data.

In GANs, there is a Generator and a Discriminator. The generator learns to create synthetic images, while the discriminator becomes adept at distinguishing between

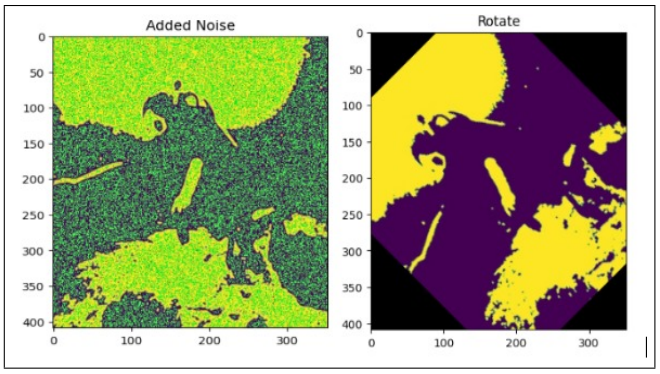

Fig. 10. Added Noise and orientation changed images of the preprocessed image

real and generated images. The training loop iteratively refines the models, and the generated images are visualized throughout the process.

B. working

We implemented a simple Generative Adversarial Network (GAN) using TensorFlow and Keras for generating images. GANs consist of a generator model and a discriminator model that are trained simultaneously in a competitive manner. The generator takes a random noise vector (`latent_dim`) as input and produces an image. It consists of fully connected layers with leaky ReLU activation functions and batch normalization.The output is a generated image with the same dimensions as the input images The discriminator takes an image as input and outputs a binary classification (real or fake). It consists of fully connected layers with leaky ReLU activation functions. The output is a probability indicating the likelihood that the input image is real, the Adam optimizer for both the generator and discriminator.

Here in fig:11,we can see the image data used in this work. Images: Each row presents high-, medium- and low-quality images, respectively, of the animals investigated in this study.We can see the different epochs obtained for each image[14].

VI. Transfer Learning Models

In transfer learning, from each pretrained and finetuned network, a set of those layers are selected that would output the “best” feature vectors for classification. For selecting the layers, they employ the “sequential floating forward selection”(SFFS) method, which uses an SVM. In plankton classification, the distributions of different classes are generally different in the training and the test sets.

We are going to use Ensemble learning is a machine learning technique that involves combining the predictions of multiple models to improve the overall performance and robustness of a system. The idea is that by aggregating the predictions of multiple models, the ensemble can often achieve better results than any

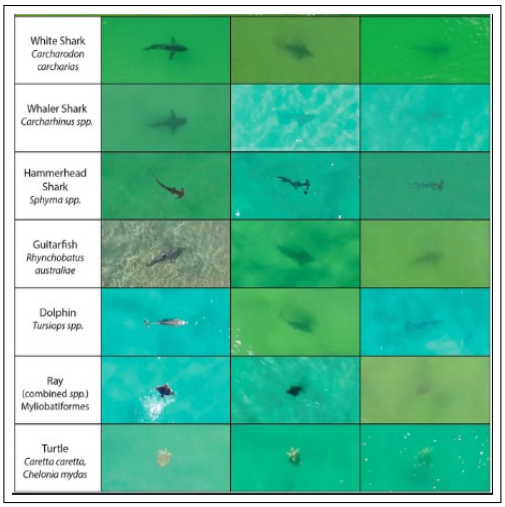

Fig. 11. Generation of GAN images at epochs 5,10,15 [14]

individual model on its own.

In this paper, we implement a multi-input neural network using the VGG16 and ResNet50 architectures for image classification. It uses transfer learning by loading pre-trained models and then adding custom layers on top for your specific classification task.

we are going to use multiple collaborative models for improved classification performance on datasets with class imbalance. This system combines pretrained CNNs followed by an additional learning phase. To mitigate class imbalance, they employ the strategies of data standardization, data augmentation, and usage of "class weights". Additionally, the authors integrate training using geometric (dimensions, area, etc.) and environmental data (temperature, salinity, season, time, etc.) into the classification system by concatenating with the extracted feature maps from CONV layers

We take VGG and ResNet models for constructing a collaborative model. The learners are trained individually and are loaded with fixed weights. And in each model last layers are concatenated and followed by softmax layer. The FC layer works as novel function for the model to learn how efficiently every learner contributes.

.

VGG16 is a convolution neural network (CNN) architecture that's considered to be one of the best vision model architectures to date. Instead of having a large number of hyper-parameters, VGG16 uses convolution layers with a 3x3 filter and a stride 1 that are in the same padding and maxpool layer of 2x2 filter of stride 2. It follows this arrangement of convolution and max pool layers consistently throughout the whole architecture. In the end it has two fully connected layers, followed by a softmax for output. The 16 in VGG16 refers to it has 16 layers that have weights. This network is a pretty large network, and it has about 138 million (approx) parameters.

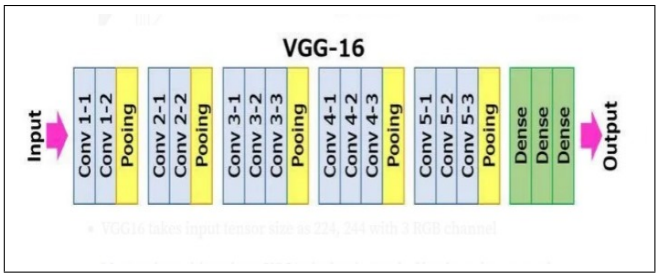

Fig. 12. VGG-16 Architecture[3]

ResNet-50 is a convolutional neural network that is 50 layers deep. You can load a pre-trained version of the neural network trained on more than a million images from the ImageNet database [1]. The trained neural network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the neural network has learned rich feature representations for a wide range of images. The neural network has an image input size of 224-by-224.

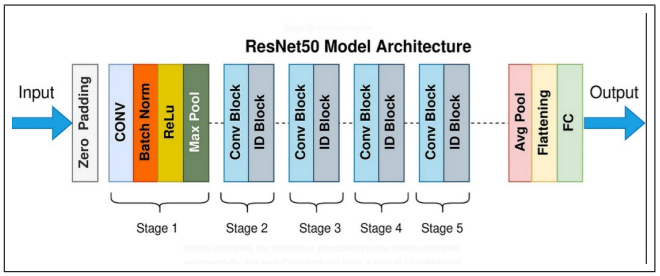

Fig. 13. ResNet Architecture[4]

Data Preparation: Training and validation data directories are specified. Image dimensions and batch size are defined. ImageDataGenerator is used for data augmentation.
Model Architecture: VGG16 and ResNet50 base models are loaded with pre-trained weights. Custom dense layers are added on top of each base model. The outputs of both models are concatenated. A final dense layer with softmax activation is added for classification. The model is created using the Model class from Keras.
Freezing Base Model Layers: The layers of both VGG16 and ResNet50 base models are set as non-trainable to preserve their pre-trained weights.
Model Compilation: The model is compiled with the Adam optimizer and categorical cross entropy loss.
Training: The model is trained using the fit method. The training data is passed as a list containing both VGG16 and ResNet50 features.
Evaluation: The model is evaluated on the validation set, and predictions are obtained. Accuracy is calculated using Scikit-learn's accuracy score.

ResNet enables the creation of very deep neural networks, which can improve performance on image recognition tasks.ResNet50 provides a novel way to add more convolutional layers to a CNN, without running into the

vanishing gradient problem, using the concept of shortcut connections. VGG16 supports the processing of a large-scale data set with deep network layers and smaller filters to produce a better performance. VGG model can have a considerable number of weight layers due to the small size of the convolution filters; of course, more layers mean better performance. However, this isn't an unusual trait. Ensembling both models creates a better model offsetting the disadvantages of the other to create a better model

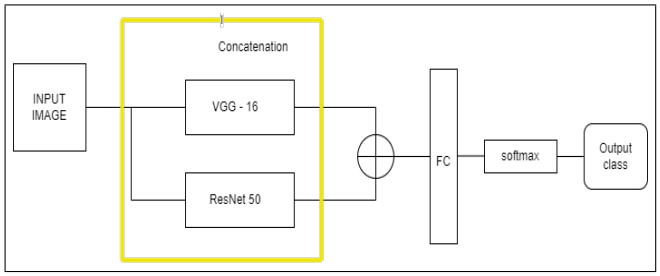

Fig. 14. Ensemble Model

VII. Comparing results with other models

SI.No	Model	Time taken	Accuracy
1	VGG16	1.4 hours	94.77%
2	ResNet	1.2 hours	89.21%
3	VGG-16 + ResNet50	1.7 hours	95.83%
4	AlexNet	1.2 hours	87.02%
5	GoogLeNet	1.5 hours	90.8%

VIII. Baseline Techniques

In the paper [1] the author has Deep Learning Techniques for Underwater Image Classification in order to find best CNN model for the classification of underwater images. The author compared them on critical parameters and highlighted their similarities and differences. The author reviewed the works related to datasets and training and those related to the design and optimization of CNNs. In this paper, we optimize the CNN model using Ensemble Learning by using VGG16 and ResNet50, and we concatenate both models and use FC classification using softmax classification. In this way, we can get better results by improving the accuracy.

IX. Experimental Analysis

This section describes the details of the dataset, experimental environment, parameter setting for different models and selected standard evaluation measures. The dataset is of size 1136 MB. To train the model over the entire dataset we used 12GB of RAM and 1TB storage. The generated images take nearly 12 GB of storage Running the model through the entire dataset gives a good estimation of model performance.

Dataset name: Underwater Scenes

Nature: a paired collection of images on EUVP dataset

Source: https://irvlab.dl.umn.edu/resources/euvp-dataset

Statistics:-

We are taking 2185 pairs of images for training

- Train folder for training the models.
- Validation folder to test the model and find the accuracy of the model.

X. Conclusion

A.

In this paper, we presented preprocessing techniques, and data augmentation techniques useful to make better training model techniques to classify underwater images. We compared them on critical parameters and highlighted their similarities and differences. We reviewed the works related to datasets and training and those related to the design and optimization of CNNs. We close this paper with a brief mention of future research challenges.

Deep learning models require a large amount of data to achieve high accuracy. While data augmentation overcomes the scarcity of training data also reduces the robustness of the network

Ensembling the VGG16 and ResNet50 gives a better training model for the classification of underwater images for better classification

Additionally, in future work, we integrate training using geometric (dimensions, area, etc.) and environmental data (temperature, salinity, season, time, etc.) into the classification system by concatenating with the extracted feature maps from CONV layers.

References

[1] S. Mittal, S. Srivastava, and J. P. Jayanth, "A survey of deep learning techniques for underwater image classification," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[2] S. L. Bangare, A. Dubal, P. S. Bangare, and S. Patil, "Reviewing otsu's method for image thresholding," *International Journal of Applied Engineering Research*, vol. 10, no. 9, pp. 21777–21783, 2015.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[4] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into deep learning*. Cambridge University Press, 2023.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[6] Y. Massoud, *Sensor fusion for 3D object detection for autonomous vehicles*. PhD thesis, Université d'Ottawa/University of Ottawa, 2021.

[7] T. Nguyen, T. Le, H. Vu, and D. Phung, "Dual discriminator generative adversarial nets," *Advances in neural information processing systems*, vol. 30, 2017.

[8] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *IEEE transactions on image processing*, vol. 2, no. 2, pp. 176–201, 1993.

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

[10] R. C. Gonzalez, *Digital image processing*. Pearson education india, 2009.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[12] D. Learning, "Ian goodfellow, yoshua bengio, aaron courville," *The reference book for deep learning models*, vol. 1, 2016.

[13] F. Han, J. Yao, H. Zhu, C. Wang, *et al.*, "Underwater image processing and object detection based on deep cnn method," *Journal of Sensors*, vol. 2020, 2020.

[14] C. R. Purcell, A. J. Walsh, A. P. Colefax, and P. Butcher, "Assessing the ability of deep learning techniques to perform real-time identification of shark species in live streaming video from drones," *Frontiers in Marine Science*, vol. 9, p. 981897, 2022.
[15] M. J. Islam, Y. Xia, and J. Sattar, "Fast underwater image enhancement for improved visual perception," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3227–3234, 2020.
[16] M. Jahidul Islam, Y. Xia, and J. Sattar, "Fast underwater image enhancement for improved visual perception," *arXiv e-prints*, pp. arXiv–1903, 2019.
[17] A. K. Agarwal, R. G. Tiwari, V. Khullar, and R. K. Kaushal, "Transfer learning inspired fish species classification," in *2021 8th International conference on signal processing and integrated networks (SPIN)*, pp. 1154–1159, IEEE, 2021.
[18] S. Sa'idah, A. Fany, and I. P. Y. N. Suparta, "Convolutional neural network googlenet architecture for detecting the defect tire," in *2022 International Conference on Computer Science and Software Engineering (CSASE)*, pp. 331–336, IEEE, 2022.

[1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [17] [18]