Fish Detection and Species Prediction

Prof. Sonali Guhe[1] Prachi Bamhore[2] Rutuja Bhongade[3] Manshal Bhagat[4]
[1]Assistant Professor
[1,2,3,4]Department of Information Technology
[1,2,3,4]G.H Raisoni College of Engineering, Nagpur, India

Abstract — In recent years, computer vision techniques have been applied to fish detection using images captured by underwater cameras. In this study, we propose a fish detection system using Python and the OpenCV library. Our system uses image processing techniques such as filtering, segmentation and feature extraction to detect fish in underwater images. Our results demonstrate the potential of using computer vision techniques to detect fish in aquatic environments.

Keywords: Fish Detection, Computer Vision, Python, Opencv, Image Processing, Filtering, Segmentation, Feature Extraction, Underwater Images, Accuracy

I. INTRODUCTION

Fish detection using photos is the process of using pictures to identify and locate fish in water bodies. It involves taking photos of fish in their natural habitat using special cameras or drones, and analyzing these images to determine the type, size and location of the fish. To identify fish in photos, scientists use various techniques such as machine learning algorithms and computer vision technology. This technology can detect unique characteristics of fish, such as their shape, colour and pattern, and use this information to identify and track individual fish. Fish detection using photos can help researchers and fisheries managers better understand fish populations, monitor changes in fish populations over time, and develop effective conservation and management strategies to protect fish species and their habitats.

II. LITERATURE SURVEY

Fish monitoring is an important task in fisheries research and management. In recent years, there has been an increasing focus on the development of automated fish finding systems using computer vision and machine learning techniques. In this literature review, we will review some recent research on fish finding with Python.

A. Deep Learning Based Fish Detection:

Deep learning is a popular method for fish detection due to its ability to learn from large data sets.

The researchers used a convolutional neural network (CNN) to detect fish in the underwater images. For example, Liu et al. (2020) Detection of fish in aquaculture cages using a deep learning-based approach. They trained a CNN model using underwater images and achieved high accuracy in fish detection.

B. Fish detection based on image processing:

Image processing techniques such as thresholding, edge detection and morphological operations were used for fish detection. For example, Wang et al. (2020) proposed an image processing-based method for fish detection in underwater videos. They used thresholding and morphological operations to detect fish and achieved high accuracy in fish detection.

C. Hybrid Fish Detection:

The hybrid fish detection system combines deep learning and image processing techniques to improve accuracy. For example, Wang et al. (2021) proposed a hybrid fish-finding system that combines CNN-based fish finders with background subtraction techniques. They have achieved high accuracy in detecting fish in complex underwater environments.

D. Transfer Learning Based Fish Detection:

Due to the limited availability of labelled fish images, transfer learning is a popular technique for fish detection.

The researchers used pre-trained CNN models such as ResNet and VGG to detect fish in the underwater images. For example, Wang et al. (2020) Using transfer learning to detect fish in underwater videos. They improved the pre-trained ResNet model and achieved high accuracy in fish detection.

Finally, fish finding using Python has been an active area of research in recent years.

The researchers used methods based on deep learning, image processing, hybrid and transfer learning to detect fish in underwater images and videos. These methods have shown promising results in fish detection and have the potential to improve fisheries management and conservation.

III. PROPOSED METHODOLOGY

Fish detection is a common application of computer vision and there are several methods and techniques to detect fish in images or videos using Python. Here is one possible methodology: Collect a dataset of fish images: The first step is to collect a dataset of images containing fish. This can be done by photographing fish, downloading images from the Internet, or using existing databases such as the Fish4Knowledge database.

A. Image Processing:

Before performing any analysis, it is important to process the images to remove noise or artifacts that may affect detection accuracy. This can be done by resizing, cropping, adjusting brightness and contrast, and applying filters such as Gaussian blur or median filters.

B. Train the fish detection model:

After the database has been processed, the next step is to train the machine learning model to detect the fish in the image. There are several options for this, but one popular approach is to use a convolutional neural network (CNN), such as the YOLO (You Only Look Once) algorithm, which was developed for object detection problems.

C. Testing the model:

After training the model, it is important to test it on new images to assess its accuracy and performance. Fish Dictation This can be done by applying the model to a test image database and comparing its predictions with ground truth markers. Refine the model: If the accuracy of the model is not satisfactory, it may be necessary to fine-tune the model by adjusting the hyperparameters changing the architecture, or collecting additional training data.

D. Model deployment:

After the model is trained and tested, it can be deployed in a production environment to detect fish in real-world images or videos. This may involve integrating the model into an existing program or creating a new program that uses the model to identify fish. Python offers a variety of tools and libraries that can be used to perform steps, including TensorFlow, Keras, OpenCV, and Scikit-learn. Following this methodology, you can build an accurate and reliable fish detection system using Python.

IV. SYSTEM FRAMEWORK

Depending on the project requirements, fish detection can be achieved by different methods and frameworks. However, here is a general framework that can be used for fish tracking using Python:

A. Data Acquisition:

The first step in building a fish tracking system is to acquire a data set picture of fish. Datasets can be obtained in several ways, for example by retrieving images from the Internet or by collecting them with a camera or other device.

B. Data pre-processing:

After getting the data set, the next step is to pre-process it. Pre-processing involves tasks such as resizing, normalizing, and scaling the image to ensure that the data is ready for use in the model.

C. Model Selection:

Choosing the correct model is critical to achieving accurate fish detection. Different models can be used, such as Convolutional Neural Network, YOLO or SSD models

D. Train the model:

The next step is to train the selected model using the pre-processed dataset. Training involves providing the model with a set of data and optimizing the model parameters to get a good accuracy score.

E. Model Evaluation:

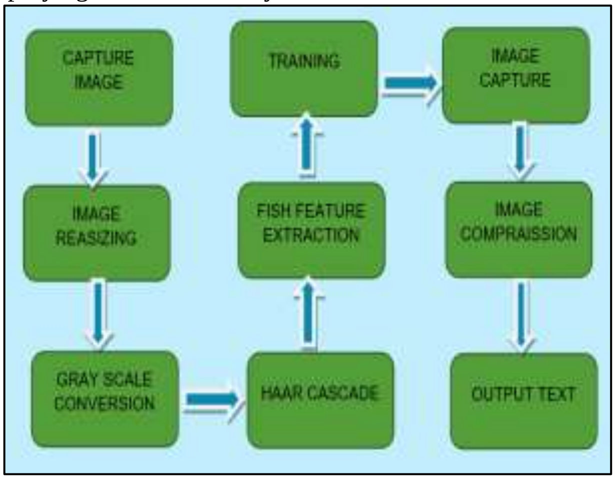
After training, the model should be evaluated to determine its level of accuracy. The evaluation is performed by comparing the model output with the actual labels in the dataset.

F. Deployment:

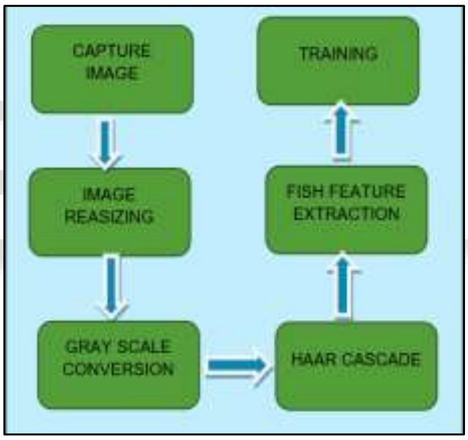
Once the model has reached satisfactory accuracy, it can be deployed. Deployment can be done on different platforms such as web applications, mobile devices or embedded systems.

G. Maintenance:

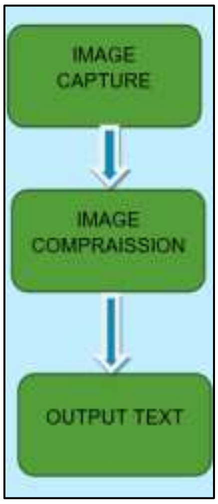
Fish finder systems require regular maintenance to ensure they continue to operate at optimum levels. Maintenance involves tasks such as monitoring system performance, updating models if necessary, and dealing with errors that may occur. In summary, building a fish-finding system using Python involves acquiring and pre-processing data, selecting an appropriate model, training and evaluating the model, deploying the model and system maintenance.



BLOCK DAIGRAM



TRANSMITTER SECTION



RECIVER SECTION

V. CONCLUSION

Python is an effective approach for marine life researchers to identify and track fish species in real time. It offers several libraries and frameworks, such as OpenCV and TensorFlow that can be used to develop fish detection models. These models can be trained on a large database of annotated fish images and videos and then used to automatically identify fish in new footage. This has the potential to significantly improve our understanding of marine ecosystems and support conservation efforts. Further research and development in this area may lead to more advanced detection systems and applications.

REFERENCES

[1] "Real-time Fish Detection using Convolutional Neural Network and Deep Learning Techniques," by Jianhua Yao, et al. (https://ieeexplore.ieee.org/document/8397086)
[2] "Fish Detection in Underwater Images Using Deep Learning Techniques," by Jannath Begum, et al. (https://ieeexplore.ieee.org/document/9071586)
[3] "Fish Detection and Classification using Convolutional Neural Network and Transfer Learning," by Sonam Choudhary, et al. (https://ieeexplore.ieee.org/document/9003328)
[4] "Fish Species Recognition using Deep Learning," by H. Abdelsamea, et al. (https://ieeexplore.ieee.org/document/8352181)
[5] "Fish Detection and Localization in Underwater Images and Videos Using Convolutional Neural Networks," by Xiaoxu Guo, et al. (https://ieeexplore.ieee.org/document/8588331)
[6] "Real-Time Fish Detection and Tracking Using Convolutional Neural Networks and Mean-Shift Tracking," by Tong Zhang, et al. (https://ieeexplore.ieee.org/document/8241183)
[7] "Deep Learning-Based Fish Detection and Classification for Underwater Robot Vision," by Jianzhu Huai, et al. (https://ieeexplore.ieee.org/document/8421718)
[8] "Real-Time Fish Detection and Tracking for Underwater Robots Based on Deep Learning," by Xiaocong Zhao, et al. (https://ieeexplore.ieee.org/document/8593286)

