CHAPTER 3

SYSTEM DESIGN

3.1.GENERAL

The design and the working of the whole system is organized into four modules which includes:

- Image Preprocessing
- Dataset Preparation
- Integration of GAN
- Ensemble Model Training
- Web App Development

3.2.PROPOSED SYSTEM

3.2.1. SYSTEM FLOW DIAGRAM

A flowchart is often used to manage, analyze, design a process in many different fields. It is a useful tool that everyone should learn to help solve problems easier and more efficiently. System flowchart is one of the common variations of the flowchart.

System flowcharts are the diagram type that shows you the flow of data and how decisions can affect the events surrounding it.

The NeuroFish has 6 major component:

- Data Collection: The data are collected from various sources for training purpose.
- Data Preprocessing and Augmentation: The image in fine tuned and enhanced for better results in training phase.
- Feature Extraction: The minute detailing and unique features are extracted from the fine-tuned image.
- Model Integration: As NeuroFish uses ensemble learning multiple ML models are integrated for better training accuracy.

- Adaptive Learning and Improvement: The NeuroFish uses Transfer Learning and improvement technique to improve the dataset and recognition accuracy.
- Output: The result is displayed in a web based user interface .

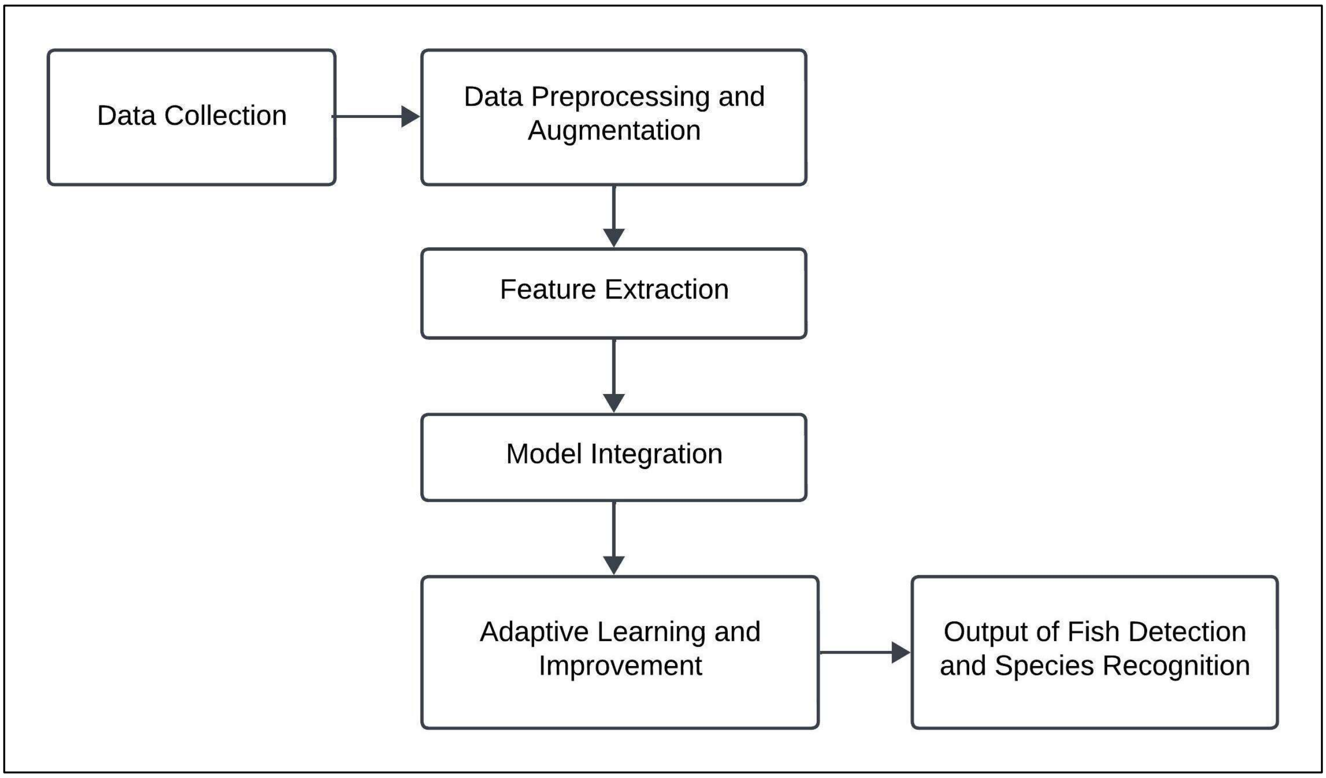


Figure 3.1. Block diagram of proposed system

3.2.2. ARCHITECTURE DIAGRAM

The software architecture diagram is a visual presentation of all the aspects that constitute a system, either in part or whole. It is a depiction of a set of concepts that comprise architecture, such as its principles, components, and materials. It is also a system diagram used to abstract the general layout of the software system as well as the interactions, limitations, and limits between parts.

The NeuroFish uses Data Lake as dataset for training purpose. The dataset is passed to Ensemble learning component which integrates multiple ML algorithm to perform operation such as feature extraction, data augmentation, identification and probability score. The data is transferred to Training Module. The prediction is done by incorporating the image inputted via user interface and the test result is sent back to user interface for displaying output.

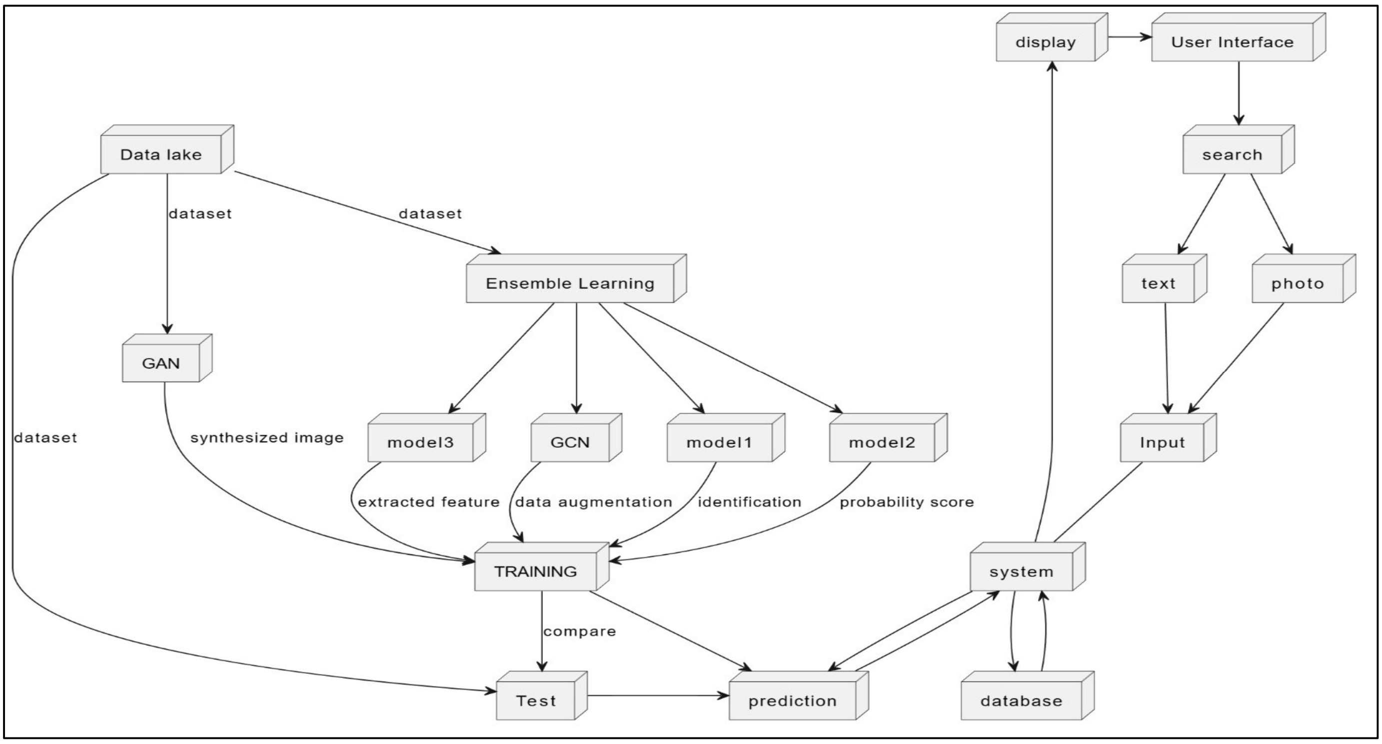


Figure 3.2. Architecture diagram of the proposed system.

3.2.3. USECASE DIAGRAM

A use case is a methodology used in system analysis to identify, clarify and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. The method creates a document that describes all the steps taken by a user to complete an activity.

Every use case contains three essential elements:

- The actor: NeuroFish has service engineer as a sole actor
- The goal: The goal is to recognize the fish species from the input image
- The system: The NeuroFish consists of various steps, starting with Data Collection, followed by Estimation which includes SoC, SoH and RUL. Then Operation involves the functioning of the system and User Interface handles the user-system interaction with a web based UI.

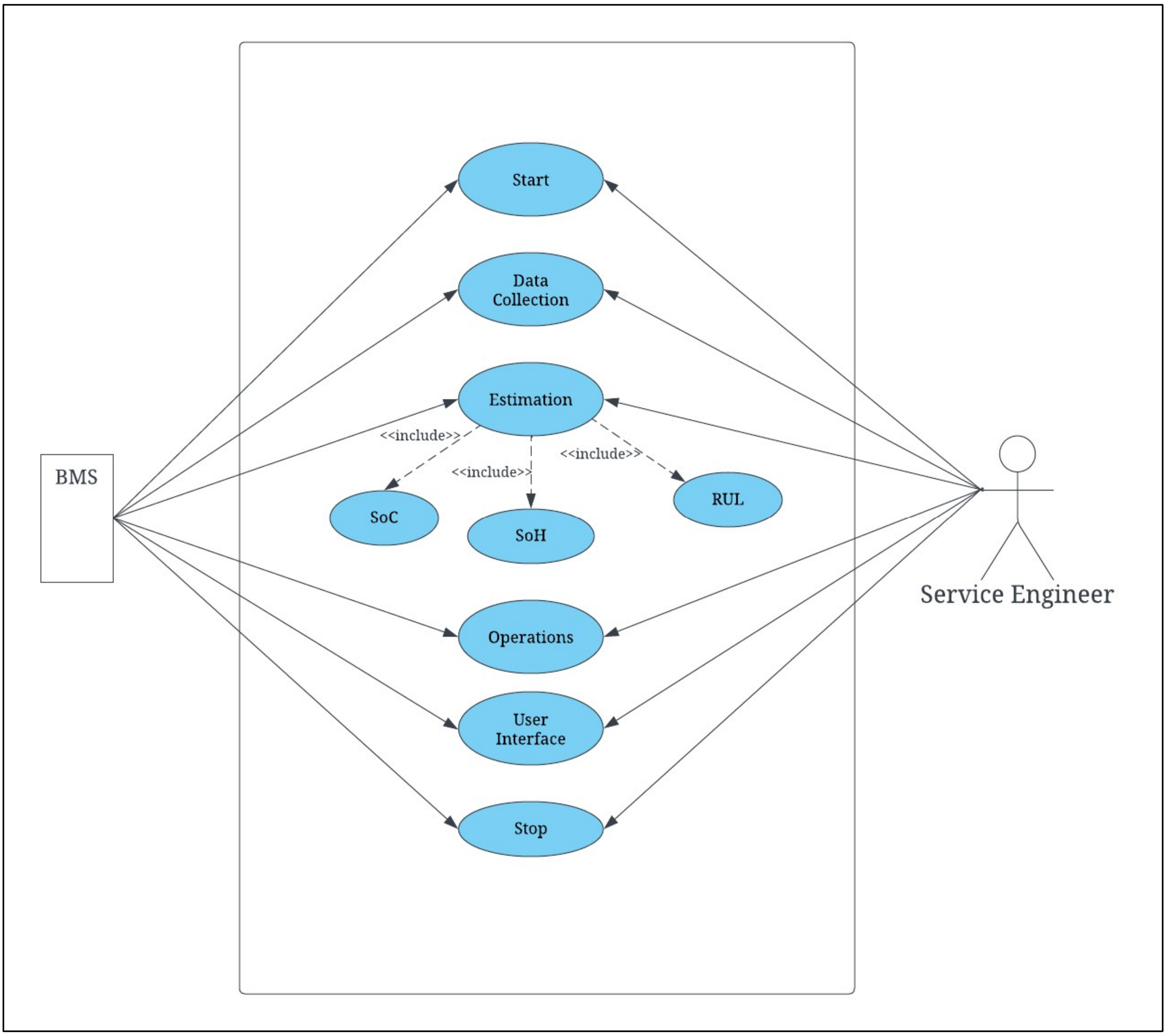


Figure 3.3. Use Case diagram of the proposed system.

3.2.4. ACTIVITY DIAGRAM

In UML, the activity diagram is used to demonstrate the flow of control within the system rather than the implementation. It models the concurrent and sequential activities.

The activity diagram helps in envisioning the workflow from one activity to another. It put emphasis on the condition of flow and the order in which it occurs. The flow can be sequential, branched, or concurrent, and to deal with such kinds of flows, the activity diagram has come up with a fork, join, etc.

The NeuroFish has linear flow mechanism, firstly the input image is fed into the system via user interface and series of ML algorithm is applied such as YOLO v7, GAN, GCN to perform image recognition, feature extraction and augmentation respectively. The features are inputted to Ensemble Learning for training and the trained model performs the recognition. The result is displayed at the user console.

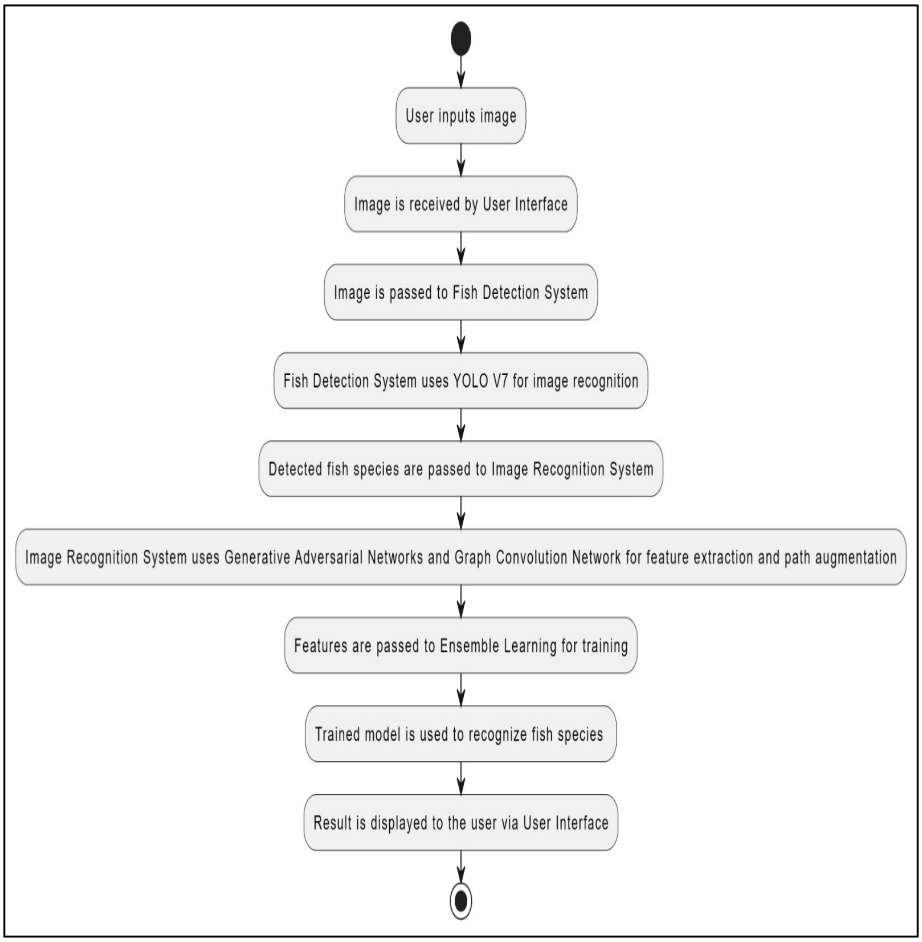


Figure 3.4. Activity diagram of the proposed system.

3.2.5. CLASS DIAGRAM

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

The NeuroFish consists of 11 class namely UserInterface, FishDetectionSystem, Fish, GAN, GCN, EnsembleLearning, Yolo v7, FishBoundingBox, Model, Image, Species.

The purpose of class diagram is to model the static view of an application. The GAN, GCN, EnsembleLearning and YOLO v7 collectively belongs to training module. The Fish, FishBoundingBox, Species, Image and UserInterface performs the user side functionality.

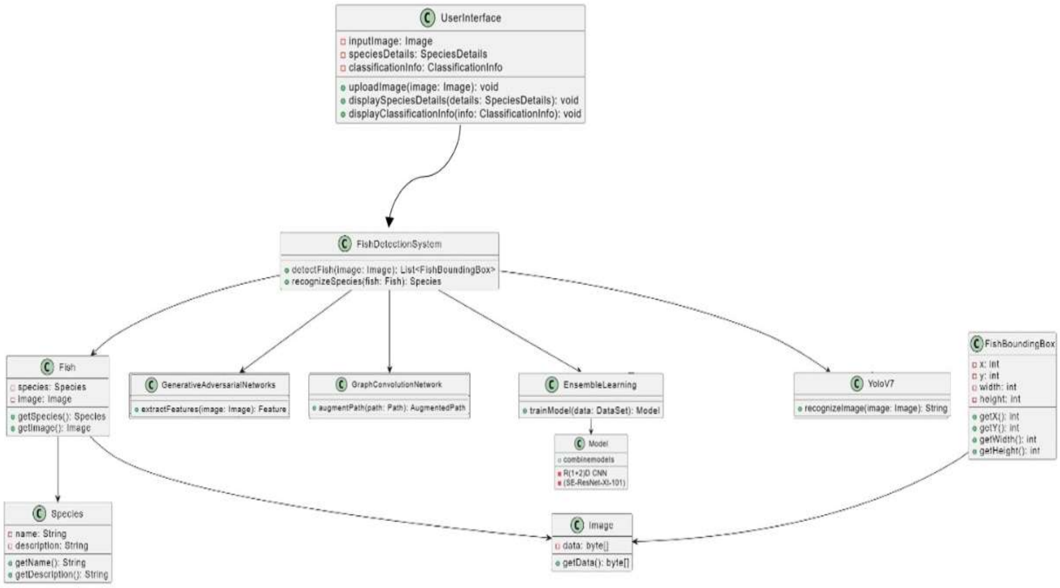


Figure 3.5. Class diagram of the proposed system.

3.2.6. SEQUENCE DIAGRAM

The sequence diagram represents the flow of messages in the system and is also termed as an event diagram. It helps in envisioning several dynamic scenarios. It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time.

The NeuroFish has user as an actor. The Fish Detection System, Image Recognition System, GAN, GCN and Ensemble Learning are the objects. The lifeline of object are active throughout the process.

The purposes of the sequence diagram are:

- To model high-level interaction among active objects within a system.
- To model interaction among objects inside a collaboration realizing a use case.
- It either model's generic interactions or some certain instances of interaction.

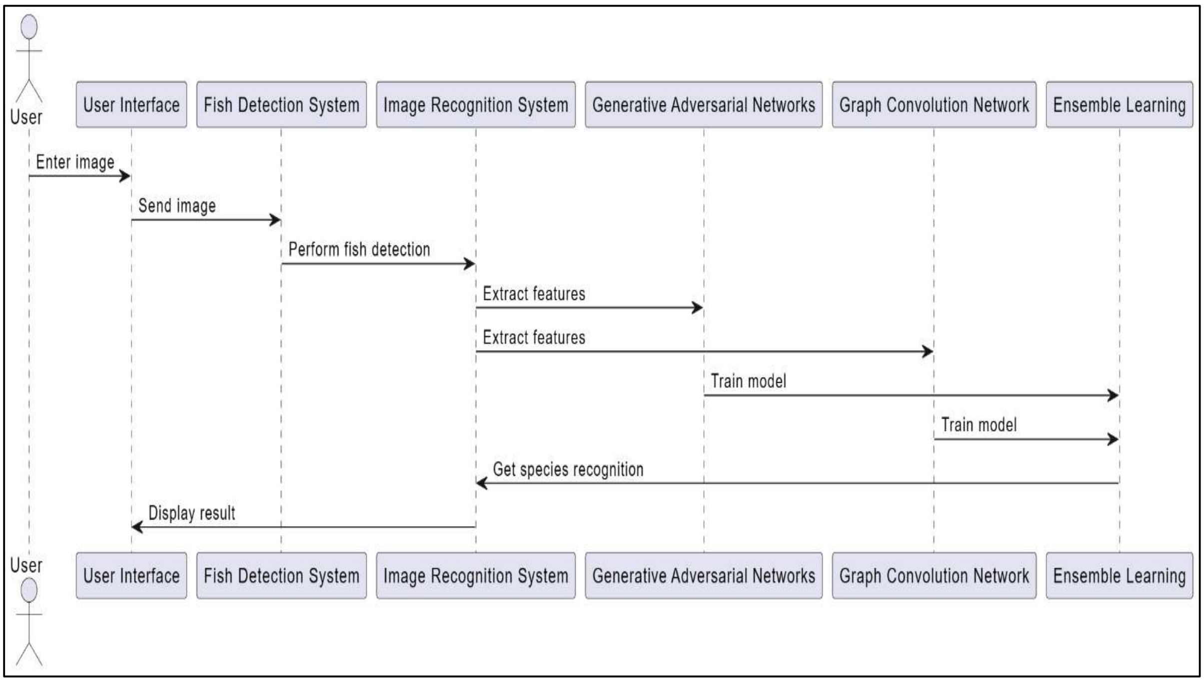


Figure 3.6. Sequence diagram of the proposed system.

3.1.7. COMPONENT DIAGRAM

Component diagrams are different in terms of nature and behavior. Component diagrams are used to model the physical aspects of a system. The NeuroFish consists of 6 component i.e User Interface, Fish Detection System, Image Recognition System, GAN, GCN, Ensemble learning.

Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

The User Interface component handles the user side interaction with the system, while the Fish Detection System performs the functionality by incorporating the Image Recognition System, which processes the image with use of GAN and GCN. The Details are processed as input to the Ensemble learning Component.

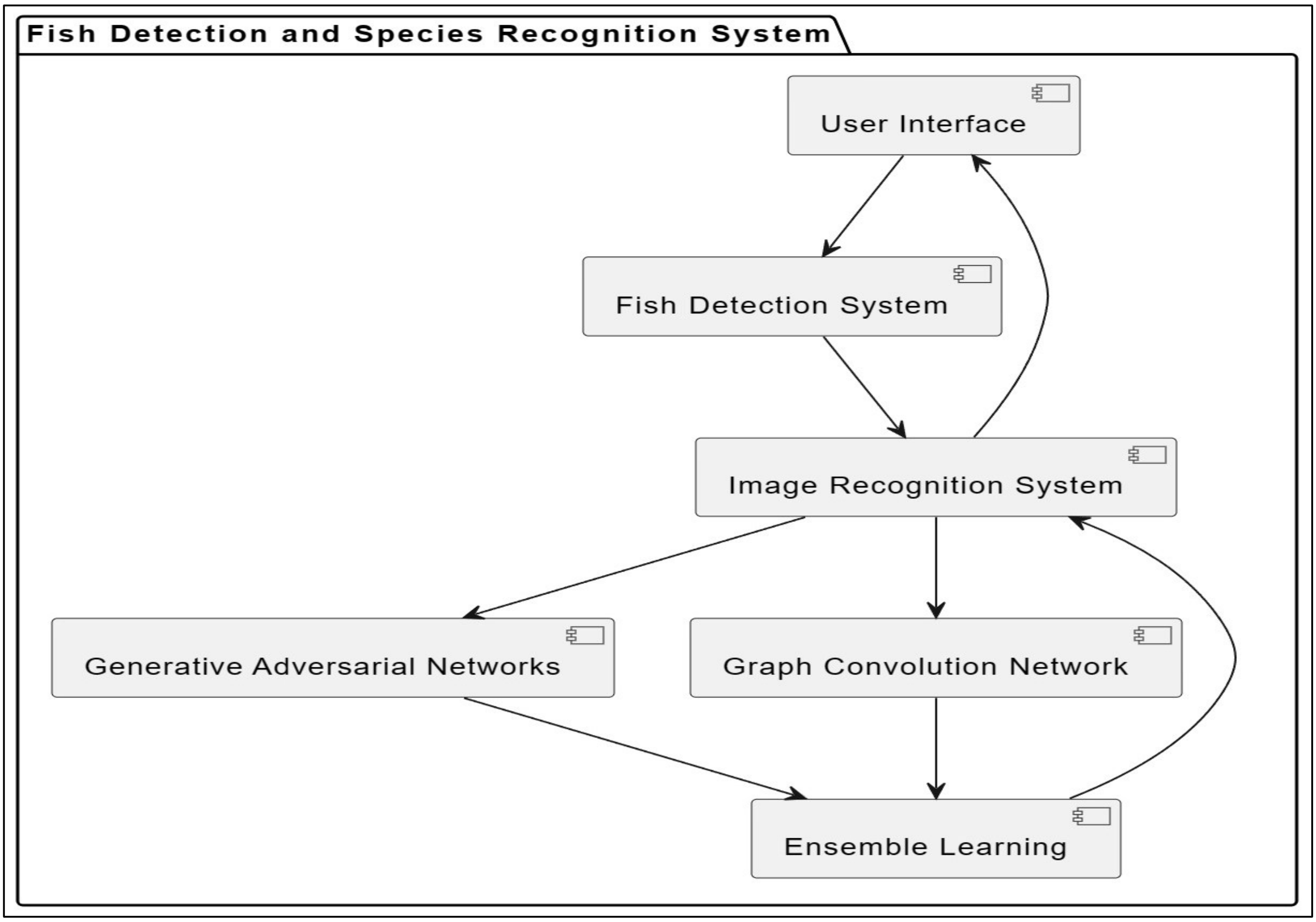


Figure 3.7. Component diagram of the proposed system.

CHAPTER 4

PROJECT DESCRIPTION

4.1.METHODOLOGIES

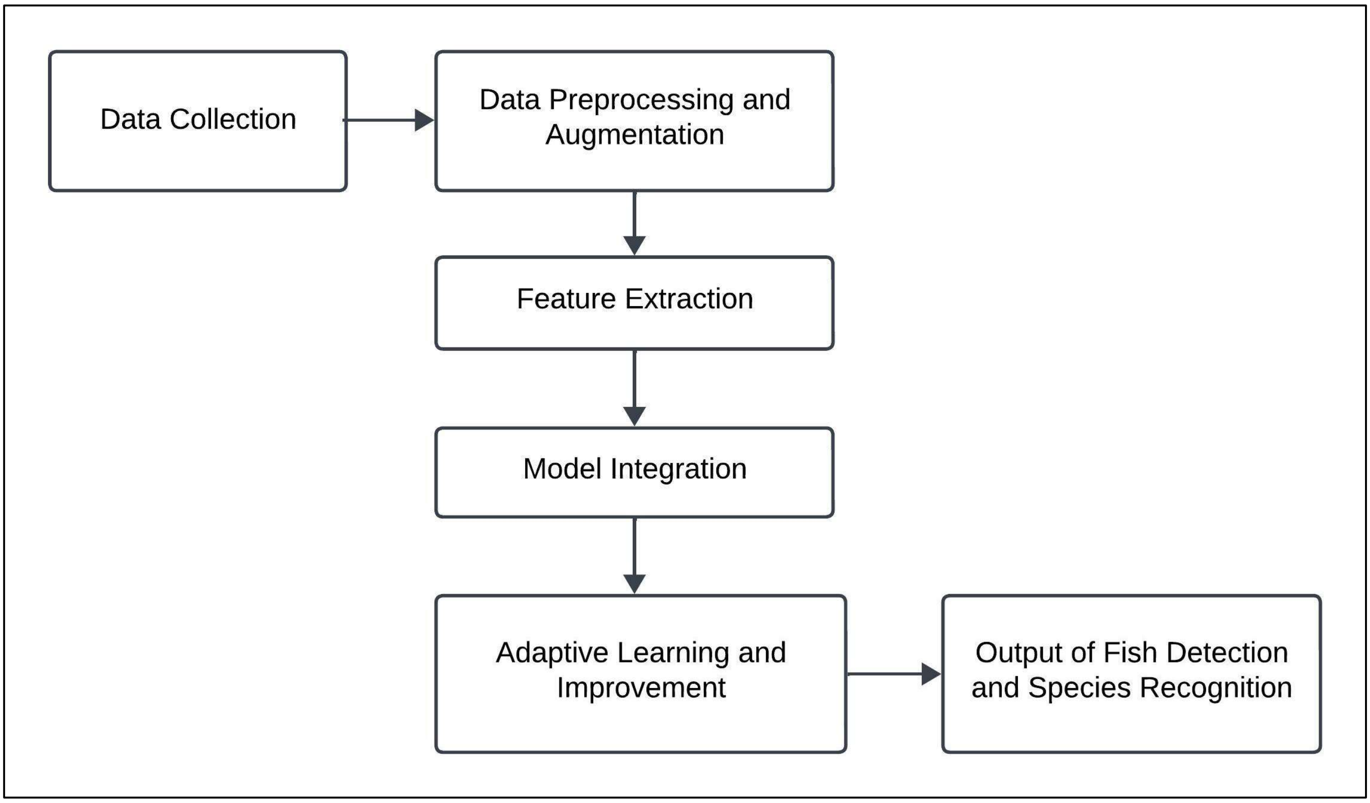


Figure 4.1. Flow of the system

Figure 4.1 represents the flow of the proposed methodology,

- Collecting the data of underwater imagery
- Raw data undergoes preprocessing to ensure uniformity and relevance.
- Data augmentation techniques are applied to artificially expand the dataset, enhancing model robustness.
- CNNs are employed for feature extraction.
- Pre-trained models, optimized for image analysis, are utilized to capture discriminative features from underwater imagery.
- Extracted features serves as inputs to Ensemble Learning Model.
- Continuous improvement mechanisms, such as model retraining is used
- Advanced image enhancement techniques are integrated.

- The system generates output reports containing information on detected fish and their species.

4.1.1. MODULES

1. Image Preprocessing

The preparation and cleaning of the image data is done for deep learning models, using techniques such as noise reduction, cropping, grayscale conversion, and extreme point detection.

2. Dataset Preparation

The dataset preparation module involves loading grayscale images, applying a bilateral filter to reduce noise, and introducing a pseudo-colored effect using the "Bone" colormap. Subsequently, the images are resized to a standardized 150x150 pixel dimension, aiming to enhance uniformity and feature representation in the dataset.

3. Integration of GAN

In the third module, the ResNet-50 model is employed for training, utilizing its deep convolutional architecture renowned for image classification tasks. The training process optimizes model parameters to enable accurate predictions on new, unseen data, leveraging ResNet-50's hierarchical feature representations.

4. Ensemble Model Training

In the fourth module, Grad-CAM (Gradient-weighted Class Activation Mapping) is integrated. This involves overlaying visualizations on the input images to highlight the regions influencing the model's predictions. Grad-CAM provides insights into the model's decision-making process by emphasizing important image features, aiding in the interpretation of neural network outputs.

5. Web App Development

The web app development module utilizes Python Flask. The front-end, built with HTML, CSS, and JavaScript, enables easy medical image uploads. Flask handles back-end tasks, integrating the ResNet-50 model for accurate predictions with robust error handling and Grad-CAM visualizations.

CHAPTER 5

CONCLUSIONS AND WORK SCHEDULE FOR PHASE II

5.1.CONCLUSION

In conclusion, the Fish Detection and Species Recognition project represents a significant stride towards revolutionizing fisheries management through the integration of ML and computer vision technologies. By addressing the inefficiencies of manual monitoring methods in underwater environments, characterized by variable conditions and diverse species, the automated system offers a scalable, accurate, and efficient solution. The project's success in mitigating labour-intensive processes and reducing errors underscores its potential impact on fisheries management, conservation efforts, and environmental monitoring. As technology continues to advance, the fusion of ML and computer vision holds promise for sustaining aquatic ecosystems and enhancing our understanding of the delicate balance within these vital ecosystems.

5.2.TIMELINE CHART FOR PHASE 11

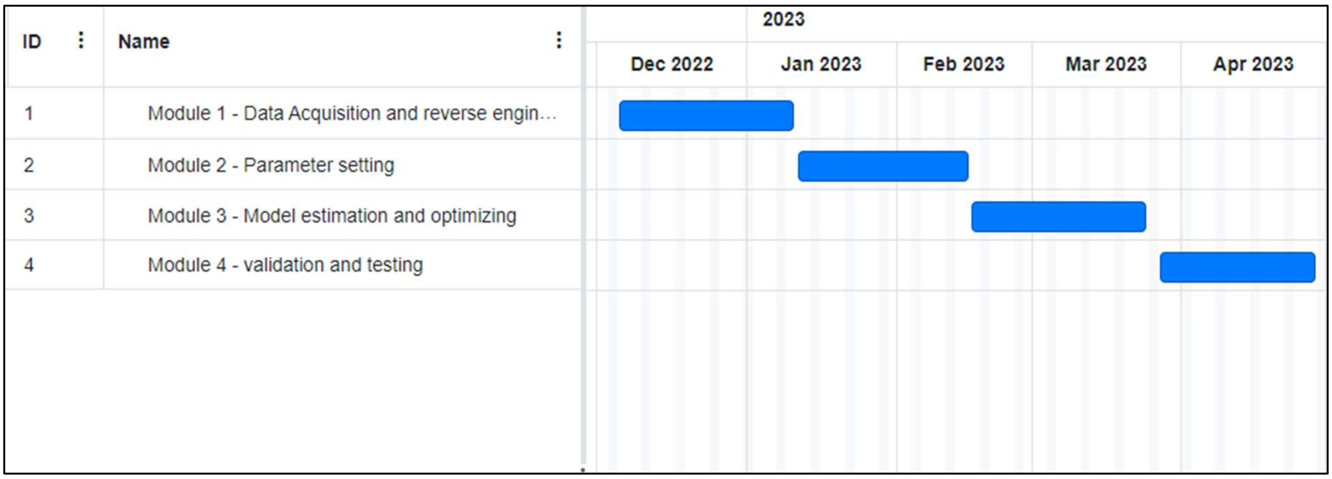


Figure 5.1. Timeline chart for PHASE II

REFERENCES

[1] M. Zhang, S. Xu, W. Song, Q. He, and Q. Wei, “Lightweight underwater object detection based on yolo v4 and multi-scale attentional feature fusion,” Remote Sens (Basel), vol. 13, no. 22, Nov. 2021, doi: 10.3390/rs13224706.

[2] Y. Cui, D. Shi, Y. Zhang, and Q. Sun, “IDNet: A Single-Shot Object Detector Based on Feature Fusion,” in Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI, IEEE Computer Society, Nov. 2020, pp. 1137–1144. doi: 10.1109/ICTAI50040.2020.00173.

[3] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, “DeepFish: Accurate underwater live fish recognition with a deep architecture,” Neurocomputing, vol. 187, pp. 49–58, Apr. 2016, doi: 10.1016/j.neucom.2015.10.122.

[4] J. Liu, L. Zhang, Y. Li, and H. Liu, “Deep residual convolutional neural network based on hybrid attention mechanism for ecological monitoring of marine fishery,” Ecol Inform, vol. 77, p. 102204, 2023, doi: https://doi.org/10.1016/j.ecoinf.2023.102204.

[5] K. M. Knausgård et al., “Temperate fish detection and classification: a deep learning based approach,” Applied Intelligence, vol. 52, no. 6, pp. 6988–7001, Apr. 2022, doi: 10.1007/s10489-020-02154-9.

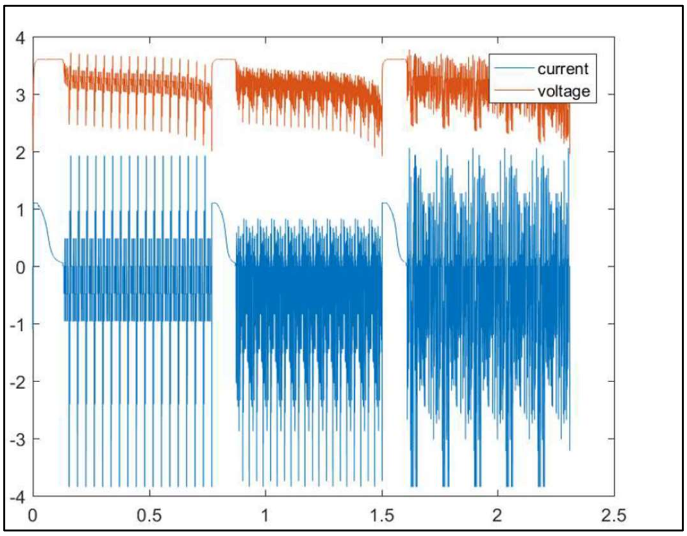
[6] C. H. Yeh et al., “Lightweight Deep Neural Network for Joint Learning of Underwater Object Detection and Color Conversion,” IEEE Trans Neural Netw Learn Syst, vol. 33, no. 11, pp. 6129–6143, Nov. 2022, doi: 10.1109/TNNLS.2021.3072414.

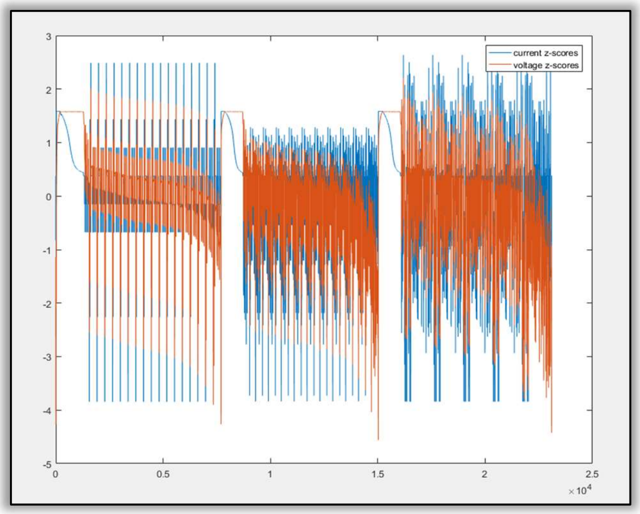
Appendix

Data_Point	Test_Time(s)	Date_Time	Step_Time(s)	Step_Index	Cycle_Index	Current(A)	Voltage(V)	Charge_Capacity(Ah)	Discharge_Capacity(Ah)	Charge_Energy(Wh)	Discharge_Energy(Wh)	dV/dt(V/s)	Internal_Resistance(Oh	Is_FC_Data	AC_Impedance(Ohm)	ACI_Phase_Angle(Deg)	Temperature (C)_1
1	3.057882145	08-13-2012 11:00:24	3.005927223	1	1	0	2.846107244	0	0	0	0	0.000123119	0	0	0	0	1.256173849
2	8.065273792	08-13-2012 11:00:29	5.007092111	2	1	-1.09918463	2.468466282	0	0.001528987	0	0.003830817	-0.00701728	0	0	0	0	1.256173849
3	13.07254414	08-13-2012 11:00:34	10.01436246	2	1	-1.09937251	2.403833389	0	0.003058049	0	0.007555851	-0.00670953	0	0	0	0	1.332008719
4	18.07782617	08-13-2012 11:00:39	15.01964449	2	1	-1.09937251	2.333968401	0	0.004586512	0	0.011177808	-0.00744815	0	0	0	0	1.256173849
5	23.07808717	08-13-2012 11:00:44	20.01990549	2	1	-1.09937251	2.250561237	0	0.006113414	0	0.014680505	-0.0092948	0	0	0	0	1.301033378
6	28.08240103	08-13-2012 11:00:49	25.02421973	2	1	-1.09956038	2.134221792	0	0.007641574	0	0.018036859	-0.01378837	0	0	0	0	1.452703238
7	31.76451946	08-13-2012 11:00:53	28.70633778	2	1	-1.09937251	1.999416113	0	0.008765985	0	0.020366863	-0.02049785	0	0	0	0	1.345893025
8	36.77421013	08-13-2012 11:00:58	5.008227742	3	1	0	2.548487902	0	0.008765985	0	0.020366863	0.004616594	0	0	0	0	1.332008719
9	41.78348919	08-13-2012 11:01:03	10.01750681	3	1	0	2.566954613	0	0.008765985	0	0.020366863	0.001600456	0	0	0	0	1.376867175
10	46.79269088	08-13-2012 11:01:08	15.02670811	3	1	0	2.577726603	0	0.008765985	0	0.020366863	0.001169538	0	0	0	0	1.332008719
11	51.79890221	08-13-2012 11:01:13	20.03291944	3	1	0	2.58449769	0	0.008765985	0	0.020366863	0.000615549	0	0	0	0	1.376867175
12	56.79930487	08-13-2012 11:01:18	25.03332249	3	1	0	2.590345621	0	0.008765985	0	0.020366863	0.000492477	0	0	0	0	1.301033378
13	61.79969176	08-13-2012 11:01:23	30.03370976	3	1	0	2.59496212	0	0.008765985	0	0.020366863	0.00049243	0	0	0	0	1.345893025
14	66.80469893	08-13-2012 11:01:28	35.03871654	3	1	0	2.598655462	0	0.008765985	0	0.020366863	0.00036931	0	0	0	0	1.345893025
15	71.80997749	08-13-2012 11:01:33	40.04399473	3	1	0	2.602041006	0	0.008765985	0	0.020366863	-6.156E-05	0	0	0	0	1.314918995
16	76.81415431	08-13-2012 11:01:38	45.04817154	3	1	0	2.604195356	0	0.008765985	0	0.020366863	0.000246191	0	0	0	0	1.345893025
17	81.81853438	08-13-2012 11:01:43	50.05255199	3	1	0	2.606965303	0	0.008765985	0	0.020366863	0.000246191	0	0	0	0	1.132279992
18	86.82483078	08-13-2012 11:01:48	55.05884839	3	1	0	2.608812094	0	0.008765985	0	0.020366863	0.000246239	0	0	0	0	1.239085674
19	91.82868461	08-13-2012 11:01:53	60.06270184	3	1	0	2.610966444	0	0.008765985	0	0.020366863	0.000246191	0	0	0	0	1.239085674
20	96.82994458	08-13-2012 11:01:58	65.06396181	3	1	0	2.612505436	0	0.008765985	0	0.020366863	0.000615549	0	0	0	0	1.314918995
21	101.8314455	08-13-2012 11:02:03	70.06546277	3	1	0	2.614044189	0	0.008765985	0	0.020366863	0.000184631	0	0	0	0	1.239085674
22	106.8398981	08-13-2012 11:02:08	75.07391571	3	1	0	2.615583181	0	0.008765985	0	0.020366863	0.000123119	0	0	0	0	1.208113551
23	111.8409386	08-13-2012 11:02:13	80.07495587	3	1	0	2.616814137	0	0.008765985	0	0.020366863	0.000123072	0	0	0	0	1.270059466
24	116.8433212	08-13-2012 11:02:18	85.07733877	3	1	0	2.618660927	0	0.008765985	0	0.020366863	0.000246239	0	0	0	0	1.163251758
25	121.8513976	08-13-2012 11:02:23	90.08541484	3	1	0	2.618968725	0	0.008765985	0	0.020366863	0.000123119	0	0	0	0	1.163251758
26	126.8587242	08-13-2012 11:02:28	95.09274139	3	1	0	2.62019968	0	0.008765985	0	0.020366863	0.000184631	0	0	0	0	1.177142978
27	131.8590675	08-13-2012 11:02:33	100.0930852	3	1	0	2.620815277	0	0.008765985	0	0.020366863	6.15597E-05	0	0	0	0	1.163251758
28	136.8655834	08-13-2012 11:02:38	105.099601	3	1	0	2.623585224	0	0.008765985	0	0.020366863	0.000738668	0	0	0	0	1.132279992
29	141.8744475	08-13-2012 11:02:43	110.1084647	3	1	0	2.622662067	0	0.008765985	0	0.020366863	0.000123119	0	0	0	0	1.163251758
30	146.8806965	08-13-2012 11:02:48	115.1147137	3	1	0	2.622969627	0	0.008765985	0	0.020366863	0	0	0	0	0	1.132279992
31	151.7679726	08-13-2012 11:02:53	120.0019902	3	1	0	2.623585224	0	0.008765985	0	0.020366863	0	0	0	0	0	1.163251758
32	156.7754252	08-13-2012 11:02:58	5.007144466	4	1	1.099753737	3.065243483	0.001529914	0.008765985	0.004635929	0.020366863	0.005601549	0	0	0	0	1.163251758
33	161.7798211	08-13-2012 11:03:03	10.01154032	4	1	1.099753737	3.102484226	0.003059009	0.008765985	0.009353279	0.020366863	0.003631735	0	0	0	0	1.163251758

Realtime dataset of a battery cell acquired by conducting Dynamic Stress Test (DST)

Battery type: LIFePo4





Before and after Normalization of data

PLAGIARISM REPORT OF SURVEY PAPER

A Survey on Fish Detection and Species Recognition

ORIGINALITY REPORT

16%

SIMILARITY INDEX

PRIMARY SOURCES

1	jtsiskom.undip.ac.id Internet	42 words — 1%
2	Wu-Chih Hu, Liang-Bi Chen, Bo-Kai Huang, Hong-Ming Lin. "A Computer Vision-Based Intelligent Fish Feeding System Using Deep Learning Techniques for Aquaculture", IEEE Sensors Journal, 2022 Crossref	41 words — 1%
3	Shuaijie Zhou, Jiefeng Liu, Xianhao Fan, Qi Fu, Hui Hwang Goh. "Thermal Fault Diagnosis of Electrical Equipment in Substations Using Lightweight Convolutional Neural Network", IEEE Transactions on Instrumentation and Measurement, 2023 Crossref	40 words — 1%
4	ceur-ws.org Internet	39 words — 1%
5	ijain.org Internet	37 words — 1%
6	Zhen Wang, Jianxin Guo, Leya Zeng, Chuanlei Zhang, Buhong Wang. "MLFFNet: Multilevel Feature Fusion Network for Object Detection in Sonar Images", IEEE Transactions on Geoscience and Remote Sensing, 2022 Crossref	35 words — 1%

LIST OF PUBLICATIONS AND CONFERENCES

Submitted in 2nd International Conference on Computer, Communication and Control

Publisher: IEEE [Scopus indexed]

11/4/23, 5:33 PM Conference Management Toolkit - Submission Summary

Submission Summary

Conference Name
2nd International Conference on Computer, Communication and Control

Paper ID
230

Paper Title
Fish Detection and Species Detection : Literature Survey

Abstract
Automated fish identification is essential to overcome the challenges and time constraints associated with manual identification processes. Various techniques are explored, evaluating their performance based on factors like pre-processing methods, significant characteristics, and accuracy. Fish classification is a well-studied problem with applications in diverse fields, including target marketing and government initiatives to manage fish supply and maintain ecological balance. These automated methods play a pivotal role in sectors like commercial fisheries, agriculture, marine science, and the broader industrial fish market, supporting industries such as nutrition and canning factories. One approach involves enhancing fish recognition algorithms based on models like AlexNet, IDNet, and SAFNet. Methods such as item-based soft attention mechanisms, reduced structural complexity, and transfer learning have been employed to improve accuracy and reduce training time. Similarly, researchers have applied deep learning algorithms like Mask R-CNN, MobileNet, and Fast R-CNN to tackle species identification, length estimation, and object detection in marine environments. These techniques have
shown promising results, especially when dealing with overlapping fish or elongated objects. Furthermore, lightweight models have been developed for underwater object detection, considering issues like low visibility, color distortion, and small target size. Techniques like attentional feature fusion, modified feature pyramid generation, and adaptive anchor generators have been proposed to optimize performance, achieving a balance between accuracy and processing speed.

Created on
11/4/2023, 5:33:16 PM

Last Modified
11/4/2023, 5:33:16 PM

Authors
Manoj E (Easwari Engineering College) < elumalaimanoj789123@gmail.com>
Jeba Sheela (Easwari Engineering College) < jebasheela.soft@gmail.com>
Roshan G K (Easwari Engineering College) < gkroshan10@gmail.com>
Madhan Raj (Easwari Engineering College) < 4077madhanraj@gmail.com>

Submission Files

https://cmt3.research.microsoft.com/IC42023/Submission/Summary/230 1/2

Submitted in 2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE)

Publisher: IEEE [Scopus indexed]

10/27/23, 9:59 AM Conference Management Toolkit - Submission Summary

Submission Summary

Conference Name
2024 Second International Conference on Emerging Trends in Information Technology and Engineering(ICETITE)

Paper ID
231

Paper Title
A Survey on Fish Detection and Species Recognition

Abstract
Automated fish identification is essential to overcome the challenges and time constraints associated with manual identification processes. Various techniques are explored, evaluating their performance based on factors like pre-processing methods, significant characteristics, and accuracy. Fish classification is a well studied problem with applications in diverse fields, including target marketing and government initiatives to manage fish supply and maintain ecological balance. These automated methods play a pivotal role in sectors like commercial fisheries, agriculture, marine science, and the broader industrial fish market, supporting industries such as nutrition and canning factories. One approach involves enhancing fish recognition algorithms based on models like AlexNet, IDNet, SAFNet. Methods such as item-based soft attention mechanisms, reduced structural complexity, and transfer learning have been employed to improve accuracy and reduce training time. Similarly, researchers have applied deep learning algorithms like Mask R-CNN, MobileNet, and Fast R-CNN to tackle species identification, length estimation, and object detection in marine environments. These techniques have shown promising results, especially when dealing with overlapping fish or elongated objects.
Furthermore, lightweight models have been developed for underwater object detection, considering issues like low visibility, colour distortion, and small target size. Techniques like attentional feature fusion, modified feature pyramid generation, and adaptive anchor generators have been proposed to optimize performance, achieving a balance between accuracy and processing speed.

Created on
10/27/2023, 9:58:46 AM

Last Modified
10/27/2023, 9:58:46 AM

Authors
Jeba Sheela (Easwari Engineering College) < jebasheela.soft@gmail.com>

Primary Subject Area
Computer Engineering -> Machine Learning

Submission Files

https://cmt3.research.microsoft.com/ICETITE2024/Submission/Summary/231 1/2