October University for Modern Sciences and Arts

Faculty of Computer Science

Graduation Project

Shrimp disease classification and behavior detection

Supervisor: Dr Ayman Ezzat

Name: Abdelaziz Ashraf

ID: 191223

Table of Contents

Table of Figures

Figure 48 Our experimental Tank recorded video 50

Abstract

Shrimp is very important aquaculture species in the world. Usually some of viruses affect them. Over the past fifty years, there has been a steady increase in shrimp production worldwide. Shrimp production reached 5.5 tons in 2021, and that many countries tend to increase their CAGR and production. Some major problems and challenges persist in shrimp production, such as feed quality and availability, production cost, seed quality, and diseases. Shrimp disease is a difficult issue because of its capacity to spread quickly through water to neighboring aqua farms. Along these areas, fast and exact diagnosis is required to control such diseases. Generally, these diseases will be analyzed by utilizing the collected experience of fisher man or fishery departmental expertise. In any case, the exactness of such last conclusion eventually relies on upon individual skill and experience and the time spent on analysis of every symptom There are types of diseases such as black gill and white spot disease. Any delay in the detection of the diseases can lead to the loss of shrimps and infection of other shrimps. Furthermore, some external conduction and diseases make shrimps behave certain behaviors and that maximize the molarity of the shrimp in tanks. Some of this behavior's is tend to swim around the tank, irregular speed then sudden stop, erratic swimming and others' behaviors. So, we need to detect the shrimp disease and detect abnormal behaviors in tanks. Our aim is to use machine learning /deep learning model to classifies if the shrimp diseased, and also, we aim to detect any abnormal behaviors in shrimp tank to minimize the molarity in the tanks. For the classification of shrimp diseases MobileNetv1 was the best. Also, Quadratic discriminate analysis was the best at Behavior classification.

Chapter 1: **Introduction**

1.1 Introduction

Shrimp is very important aquaculture species in the world. Usually some of viruses affect them. Over the past fifty years, there has been a steady increase in shrimp production worldwide. Shrimp production reached 5.5 tons and the CAGR reached 5.4 % in 2021, and that many countries tend to increase their CAGR and production. Some major problems and challenges persist in shrimp production, such as feed quality and availability, production cost, seed quality, and diseases. There are types of diseases such as black gill and white spot disease. Any delay in the detection of the diseases can lead to the loss of shrimp and infection of other shrimps. We aim to use transfer Machine learning models to detect two types of shrimp disease (white spot disease and black gill), and to detect diseased shrimp from normal shrimp. Furthermore, we also aim to detect any abnormal behaviors of shrimps in tank and notify the farmer**.** Current situation that there is multiple study that detect the shrimp diseases and can classifies the type of the disease and their also studies that track down the shrimp and its behaviors to know if its behaviors is normal or not. Furthermore, it also sends to the farmer alert to notifies are their abnormal behaviors. The accuracy of classification is not that high and may misclassifies the disease, and there may be false detection of shrimp in the tank and can wrongly count as the shrimp object, when changing the illumination condition, their system can false detect the shrimp and count as shrimp, however we may achieve a better accuracy in both classification and detection of shrimp in tank. To reduce disease risks, shrimp farmers buy quality products from reputable hatcheries and prepare and disinfect ponds regularly. The loss of an entire shrimp crop can be severe if disease detection and treatment are delayed or the farmer didn't know if the shrimps show abnormal behaviour.

So, we will use deep learning and Convolution neural network to extract diseased features from the shrimp image, moreover well use data augmentation on the data and image processing for pre-processing.

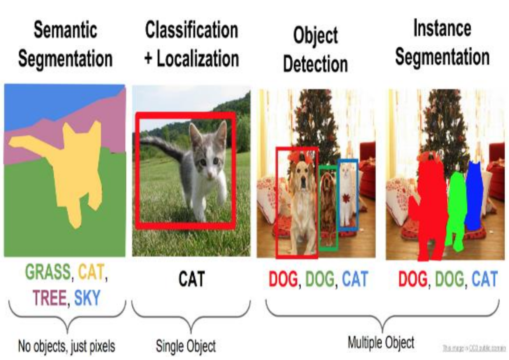


Figure 1:Object detection methods

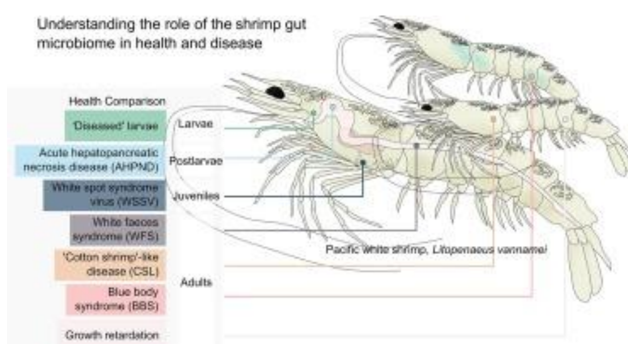


Figure 2: Shrimp diseases

Furthermore, we will use object detection to detect shrimp under water, also image segmentation and object tracking to get the trajectory of the shrimp and classifies this trajectory if it abnormal or normal using deep learning / machine learning

Figure 3 Object tracking

1.2 Problem statement:

One of the machine learning or deep learning biggest problem is data, whether the data is images or a sequence of frames of numbers. Because it's necessary for the machine to learn the features of the data so that it can predicted the outcome. and also, for object tracking it's hard to detect the object if there are many surroundings specially if it underwater, we need to enhance the detect and segmenting objects, Recently, there are many promising attempts in this area of detection of shrimp disease and to detection of any abnormal behaviors in shrimp tanks.

1.3 Objective:

This project should present classification of shrimp disease whether its diseased or not and for behaviors detection, this project should present whether the shrimp behavior underwater are an abnormal behavior or not using its trajectory.

1.4 Motivation:

Working on type of project that has a lot of market need as in Egypt, shrimp production in last 5 years there are steady increase of production and farming, as framers and companies invest in shrimp farming. We currently working with **Nutrivet Misr** for developing such a project, which they personally asked us to do it. Which in real world a lot of companies and framers loses have a high molarity in shrimp tanks which leads to a lot lose money, and to work on and applied computer science project and see the results of your work helps and benefits other is a good feeling

1.5 Thesis layout:

In this thesis, the first chapter will provide an introduction about the project and its aim.Then,the second chapter will provide a literature review and a background of the previous work in the same area of research.

Chapter 2: **Background and Literature Review**

2.1 Background:

A number of industries are adapting technology such as Farming industry and most people are becoming more technology orientated. Now farmers and feeding companies are not considering only how farm and make money from framing, but also how easy it is to use technology. That is why using image processing, deep learning and computer vision techniques are used in order to improve the usability of technology in farmers and feeding fields. Many farms use system to monitor their tanks and aquariums. A fish farm may use a intelligence system to monitor thee behaviors of fish in tanks. on the same scale we need to install a system that detect the shrimp disease and detect shrimp behaviors. As there are many diseases that affect the shrimp as black gill, white spot, bacterial disease, nutritional deficiency, red head. Also, as for behaviors shrimp show certain behaviors when stressed or lack oxygen or hungry as we need to detect and classifies these behaviors. Disease is considered the most challenging issue that faces shrimp owners and farmers. To reduce disease risks, shrimp farmers buy quality products from reputable hatcheries and prepare and disinfect ponds regularly. The loss of an entire shrimp crop can be severe if disease detection and treatment are delayed or the farmer didn't know if the shrimps show abnormal behaviors. So, we will use deep learning and machine learning to extract features from the shrimp image, moreover well use data augmentation on the data and image processing for pre-processing on data. Furthermore, we will use object detection to detect shrimp under water, also image segmentation and noise reduction, as now there are 3 approaches in this field using machine leering and deep learning and computer vision

2.1.1 Transfer Neural Network:

The CNN architecture is used in the transfer learning models. A novel approach to machine learning procedures is the concept of knowledge transmission. It creates a knowledge transferability mechanism in one or more source tasks and then applies it to a new task to increase prediction accuracy. It's similar to knowledge propagation from a well-developed domain with a large amount of learning data to a less-developed area with less data. Machine learning models may now be applied to fresh data obtained from distributions that are completely different from the old data sources using this technology. By exploiting already trained models on predetermined big datasets, machine learning techniques avoid cold start difficulties and boost generalization.

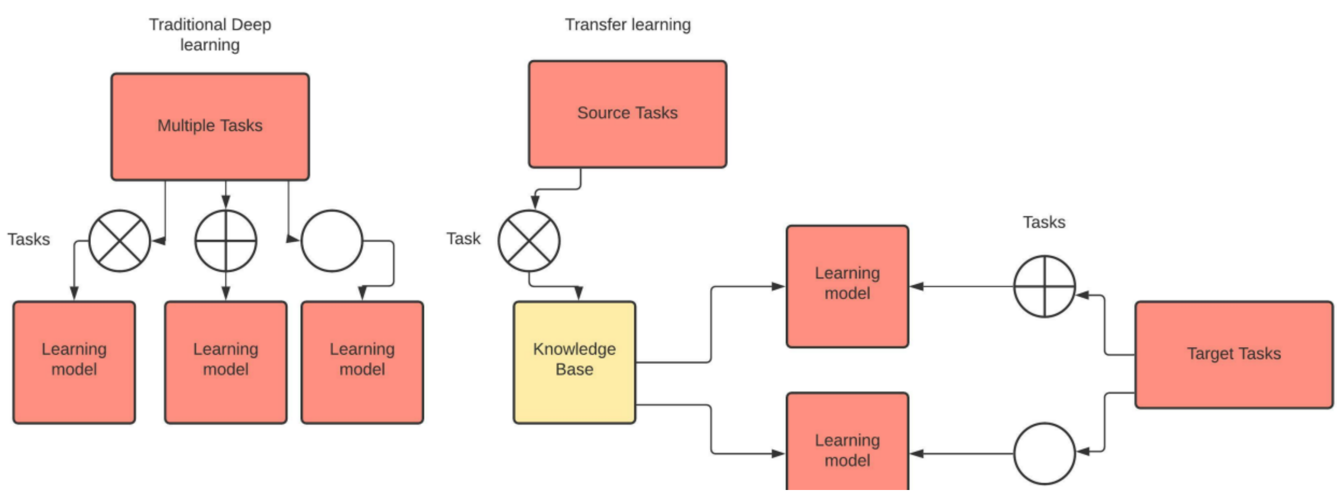


Figure 4:Transfer learning concept

2.1.2 Neural Network and fuzzy logic:

To determine the accuracy and reliability rate of the tool in detecting the disease, a hybrid algorithm that includes an Artificial neural network (Feed Forward Back Propagation Neural Networks) and fuzzy logic is used. The Logical Decision-Making Phase will process the ANN model's output, which will include a fuzzy logic model with 200 fuzzy rules. This stage aids the HNN model's output accuracy in identifying outputs.

2.1.3 Image Processing:

The video's consecutive frames (RGB images) are used as input data. To begin, we will transform the RGB color image to grayscale. In the converted grey scale picture, the object region has a deeper color than the other background areas. As a consequence, we compute the threshold value and apply it to the object area to detect it. The found object area is next morphologically processed in order to restrict the object range. The thinning process can be used to minimize the number of related pixels. As a consequence, counting object has never been easier or more convenient. To monitor the object, the moving distance between the object center of gravity is utilized. To begin, we calculate the gravitational center of each object zone detected in each frame. We next compute the Euclidean distance between the object region's center of gravity and all object regions in the previous frame to assign the tracking id to the recognized objects in the current frame. The tracking id is subsequently assigned to the item with the shortest Euclidean distance. After assigning the tracking id to all objects in the current frame, we analyze their movement. we analyze the moving distance of the object that obtained from the tracking process

2.1.4 Object tracking using Deep learning:

Object tracking uses deep learning in which an algorithm monitors an object's motion. it is the challenge of estimating or forecasting the locations of moving objects in a video and other pertinent information. tracking is frequently preceded by the detection of an object. Here's a short rundown of the steps. Detection, in which the algorithm classifies and detects the object by enclosing it in a bounding box. Creating a unique identifier for each object (ID). Tracking the identified item as it travels across frames and saving pertinent data.

2.2 Previous Work:

2.2.1 Research 1: Towards classification of shrimp diseases using transferred convolutional neural networks, 2020

2.2.1.1 Strategy & Structure

The major limitation in this filed that there are not many algorithms and model to detect and classifies the disease. The researchers develop a deep learning to detect and classifies shrimp disease, they used to detect six types of shrimp disease, they used transfer learning models (inceptionV3, MoblieNets-V1), they are CNN architecture based. That are pretrained on big datasets

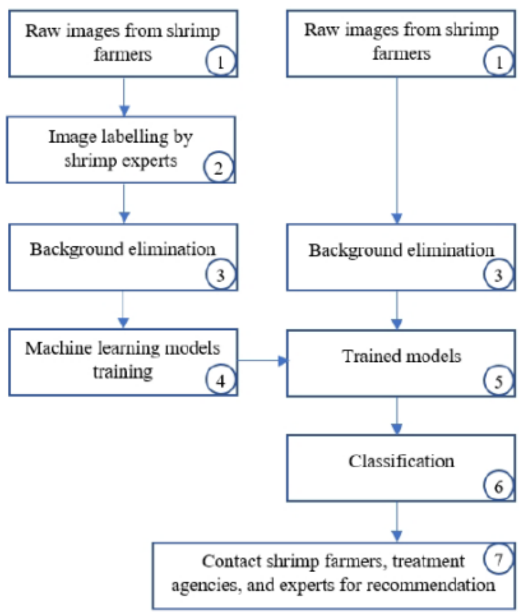


Figure 5 System overview

2.2.1.2 Data

Collection of data set was 348 images, they gathered it by launching a community hub to collect shrimps' images from farmers by uploading the image on the forums. The image was sent to experts to label them, and a background elimination was used on the shrimp images.

2.2.1.3 Method Evaluation

Using deep learning pretrained transfer learning models to detect the diseases of the shrimps

Which avoid cold start and improves generalization, hoverer, they used manually background remover.

2.2.1.4 Results Evaluation

The pretrained transfer learning models (moblienet_V1 and inception_V3) with extracted CNN features is trained on the collected dataset and achieve accuracy 90% with learning rate 0.01 and 3000 epoch.

2.2.2 Research 2: Sugpo: A white spot disease detection in shrimps using hybrid neural networks with fuzzy logic algorithm, 2018

2.2.2.1 Strategy & Structure

The researchers used Artificial neural network model to detect WSSV location and discoloration of the shrimp body and also to make the model more reliable they used fuzzy logic to make sure to the results

2.2.2.2 Data

Collection of data set was 300 images, the researcher gathers it and contains WSSV and healthy shrimp, the images went through edge detection, blob extraction and covert int to gray scale

2.2.2.3 Method Evaluation

Using machine learning hybrid model that contains ANN and hybrid to detect the diseases of the shrimps and fuzzy logic put by expert but the model accuracy isn't high and training with ANN does not get the features of the image

2.2.2.4 Results Evaluation

The ANN with fuzz logic model is trained on the collected dataset and achieve accuracy 90% with 0.8 Test-Retest Reliability.

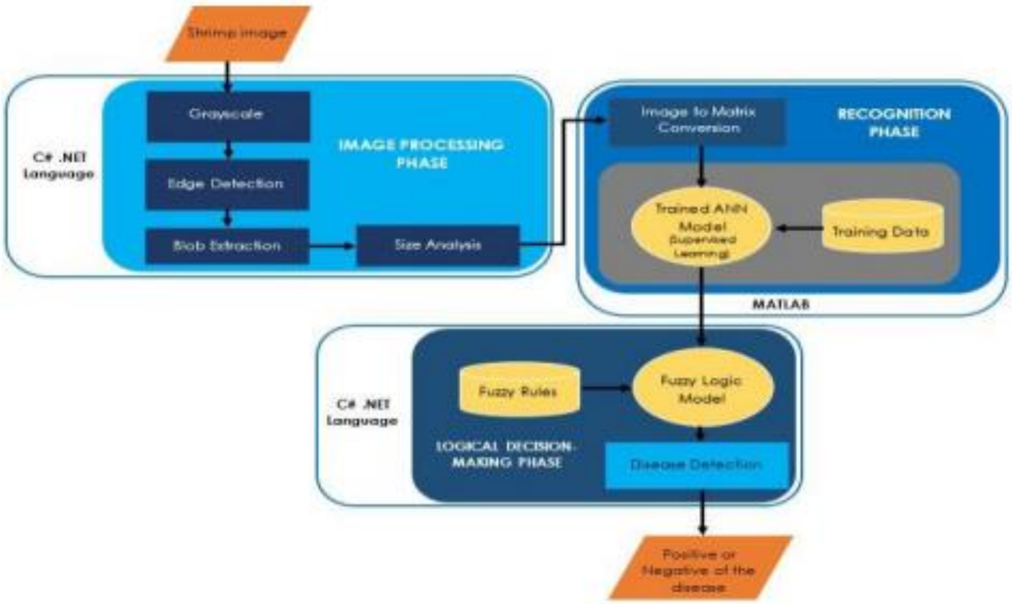


Figure 6 Proposed system

2.2.3 Research 3: A study on abnormal behavior detection of infected shrimp," in 2018 IEEE

2.2.3.1 Strategy & Structure

In this research, first the observed objects are recognized as shrimp and counting the number of shrimps in the original image. Second is for tracking the shrimp movements to study the moving distances. Finally, distinguishes the infected shrimp from the normal shrimp by the moving distance during feeding.

2.2.3.2 Data

They put the infected shrimp in the left tank and the non-infected shrimp is put in the right tank. They took the data performed on experiments using the shrimp videos data.

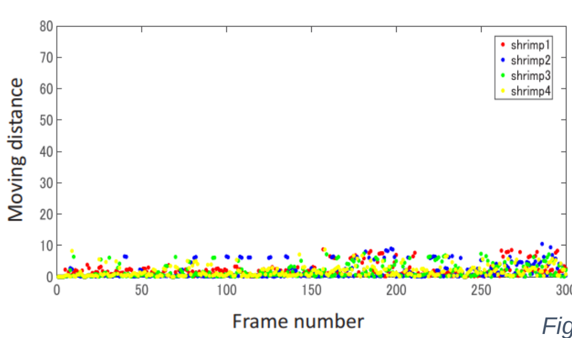
2.2.3.3 Method Evaluation

We convert the RGB color image into the gray scale image and detect the shrimp area. We then perform morphological processing on the detected shrimp area in order to thin the shrimp's region. The thinning operation can reduce the number of connected pixels, which is very effective and useful for counting the number of shrimps. Tracking is performed by using the moving distance of the shrimp. Same tracking id is given to the shrimp object which has the shortest Euclidean distance. After assigning the tracking id to each shrimp object, we analyze the movement of each shrimp objects. we can classify the infected shrimp and on-infected shrimp by analyzing the moving distance feature but it may false detection of the shrimp and there is any noise removal method

Figure 7 Results of image Segmentation

2.2.3.4 Results Evaluation

They used their technique to detect the shrimps and then used Euclidean distance as figure below



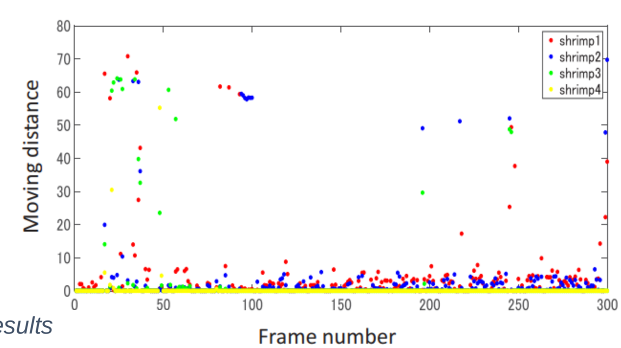


Figure 8 Results

2.2.4 Research 4: Detecting Abnormal Fish Behavior Using Motion Trajectories In Ubiquitous Environments, 2020

2.2.4.1 Strategy & Structure

Monitoring fish farms is one of the most expensive and difficult tasks for fish farmers, as it involves controlling water quality and abnormal fish behaviors within the pond. In this paper, a method for detecting fish behaviors is presented, which involves identifying the fish and analyzing their trajectories in a difficult water environment. To begin, we used an image enhancement algorithm to color-enhance water images as well as improve fish detection. The fish were then identified using an object detection algorithm. Finally, we used a classification algorithm to spot fish that were acting strangely.

2.2.4.2 Data

2000 images of gold fish were captured in order to create the model. The regions of interest are then labeled in each image using a labeling software. The labeling software produces a text file with coordinates of each box for each image in the dataset. The dataset for fish trajectories used for abnormal behaviors classification contains 96 images, each of which represents the trajectories of all detected fish over a 10-second timeframe. The 96 images were created by extracting their trajectories from a recorded video of normal behaviors and another video of abnormal behaviors. Both videos were shot at a frame rate of 30 frames per second. The images were divided into two categories: normal fish trajectories (55 images) and abnormal fish trajectories (41 images).

2.2.4.3 Method Evaluation

The main steps in our procedure. To begin, we use the Multi-Scale Retinex color enhancement algorithm (MSR) [26] to enhance unclear water images in the pre-processing phase. Following that, processing begins, with the enhanced input video sequence being loaded into our trained YOLO model, which detects the fish. The fish trajectories are then extracted and used to track fish in the pond. Finally, we detect abnormal fish behaviors in order to notify fish farmers.

2.2.4.4 Results Evaluation

the highest accuracy was 89% in the 10 seconds images and the best option for KNN was when the K=3

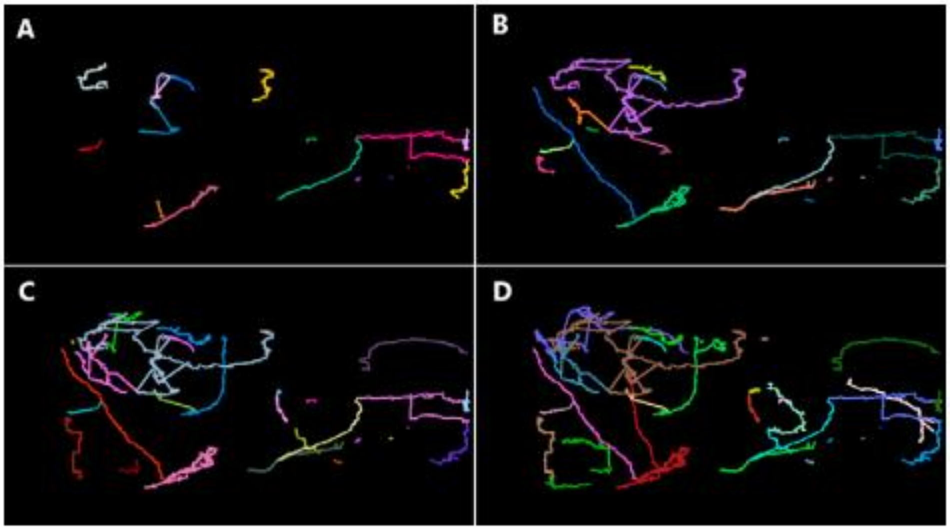


Figure 9 Fish trajectory

2.2.5 Research 5: YOLO fish detection with Euclidean tracking in fish farms, 2020

2.2.5.1Strategy & Structure

Managing fish farms, such as fish ponds surveillance, is one of the most difficult and costly missions for fish farmers. These activities are typically performed manually, wasting time and money for fish farmers. In this paper, a method is presented for improving fish detection and trajectories in challenging water conditions. Algorithm for image enhancement is first used to improve hazy images. The enhanced images are then subjected to an object detection algorithm to detect fish. In Finally, features such as fish count and trajectories are extracted from the detected object's coordinates.

2.2.5.2 Data

A dataset was made for the YOLO model to detect fish. It contained 2000 images of golden fish that were collected from our setup.

2.2.5.3 Method Evaluation

(MSR-YOLO) is explained in detail. In order to get better fish detection our algorithm integrates the Multi-Scale Retinex (MSR) color enhancement algorithm with the YOLO algorithm. After detecting fish, they are tracked to extract features like fish trajectories. Finally, fish trajectories are combined with YOLO to get the different fish movements

2.2.5.4 Results Evaluation

as figure below show results for enhancement and figure 2 show with a Optical flow and YOLO combination. b Trajectories extracting and YOLO combination

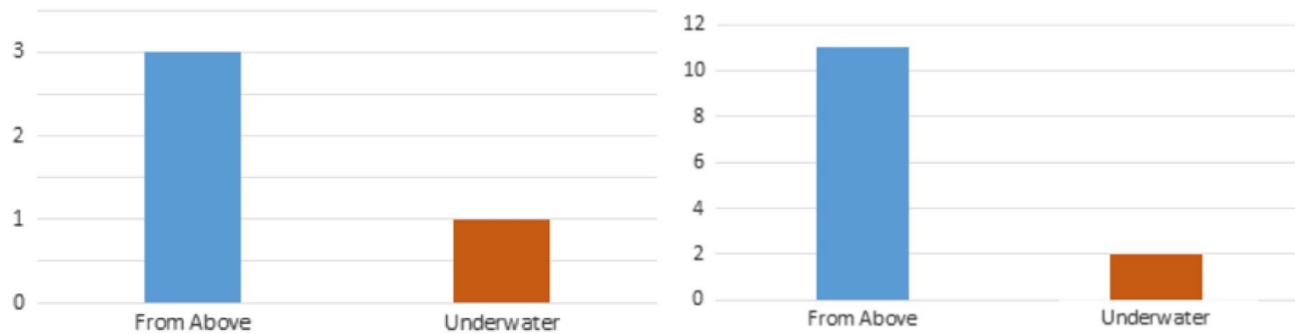


Figure 11 Results after enhancement

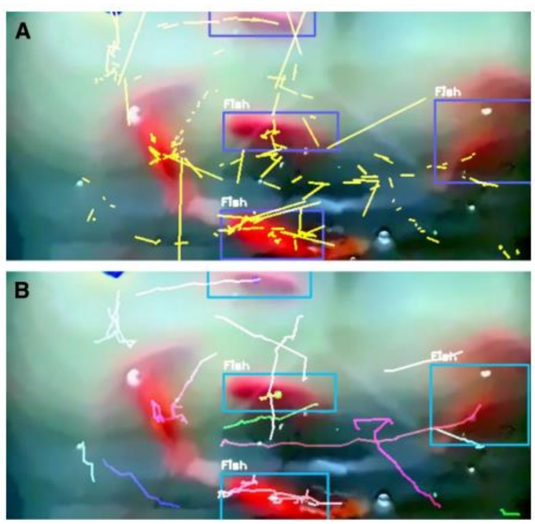


Figure 10 Trajectories and YOLO combination

Chapter 3: **Materials and Methods**

3.1 Materials:

In this section we will explain the used materials int this project which variedbetween data & tools & environments

3.1.1 Data:

This part is most challenging part in the filed as not many people work in this filed and publish their paper. Data gathering of the diseased shrimps is challenging, as paper [] and [] used to collect their dataset and collected 348 images and 300 image respectively from community forums and shrimp farmers and websites. As there are no available data on the internet or on website such as Kaggle or GitHub. So, it a challenging part of the process to whom want to work on it and a lot of people tend to stay away from this filed due to lack of data. We Contact two researchers in Polytechnic University of the Philippines (PUP) and in Can Tho University to use their dataset that their work on it. But unfortunately, the first researcher was working with a company so he couldn't share his dataset. So, we contact the second researchers of p in Polytechnic University of the Philippines but they sent a small sample of their dataset as their does not have their dataset any more, so we collected our dataset by scraping image from community google images and community website and Facebook community forums and YouTube videos. We collected 200 images

Figure 12:Our dataset samples

from this website. And for behavior detection we Build an experimental shrimp tank to conduct tests on a controlled environment and get the trajectory of abnormal behavior of the shrimp as we bought a tank A 15-litre shrimp tank (35x20x23) was provided to the house and treated to regular morning sunshine and tourch lighting at night. Furthermore, 5 red shrimp were acquired for our research A camera was placed into the tanks to take images and video. 1.8 hours video was recorded of shrimp behaviors and motions then split to a 30 seconds video length. Which give us 216 videos Then from these videos we manually get the trajectory of shrimps in which the abnormal and hungry behaviors appear. We got 140 paths / trajectory of the shrimp and saved it in excel file.

Also, for the object tracking we used YOLOv5 to track the shrimps we gather 448 images to train the YOLOv5 model with 80% for training and 20% for validation.

Figure 13:Our experimental Tank

Abdelaziz Ashraf Wahideldin Hussein <abdelaziz.ashraf@msa.edu.eg> Sat, Aug 21, 2021, 3:35 PM
to ncngon, aezzat

This is Abdelaziz Ashraf from Egypt, October University for modern sciences and arts, faculty of computer science . I am a senior student, doing my graduation project, writing down my graduation project thesis and writing a research paper about shrimp disease Classification that is expected to be submitted very soon. I'm expected to graduate next year 2022 and I am going to write a several papers.
I hope you have a good day.
I was wondering if I can get access to the dataset of your research paper Towards Classification of Shrimp Diseases Using Transferred Convolutional Neural Networks, so that I can use it in my university graduation project.
It's a very hard task to gather the data in our country, Due to covid-19 circumstances it's hard to visit the fields. It will really help with my graduation project. And I will cite your publication and add you to my acknowledgement of my paper, and this is very important to finalize my research paper.
Thank you for your help and time.

Ayman Ezzat Atia Wed, Aug 25, 2021, 11:45 AM
Dear Dr ,Nguyen Chi-Ngon This is Dr Ayman Ezzat, Vice dean of the faculty of CS at MSA, and i am supervising this project with my student AbdelAziz. I would be

Nguyen Chi Ngon 001062 <ncngon@ctu.edu.vn> Wed, Aug 25, 2021, 11:49 AM
to Ayman, me

Dear Dr Ayman Ezzat
My project supported by a company
I do not have the right to share the dataset.
So sorry
BR
Ngon
...
--

========================
sent from my phone

Assoc.Prof.Dr.Nguyen Chi Ngon,
Can Tho University
Campus 2, 3/2 Street, Ninh Kieu District
94000 Can Tho, Vietnam
https://orcid.org/0000-0002-9638-7259

---------- Forwarded message ---------
From: **Alet Fabregas** <aletfabregas@gmail.com>
Date: Tue, Sep 7, 2021 at 5:10 AM
Subject: Re: Shrimp disease classification
To: Ayman Ezzat Atia <aezzat@msa.edu.eg>

Dear Dr. Atia,
Good day. Since our paper was presented and published last 2018 and 2019 respectively, we are trying to retrieve the image of shrimps included in the training and experiment. Unfortunately, we found very few samples, but we are glad to share them. Initially I was asking our co-authors, our graduates now, to share these samples. We are very honored that you did notice our paper. Hoping that our paper will be recognized in your very good research. And as an associate professor in the university, hoping that we can also ask help from your university as part of collaboration to improve the quality of our research.

Attached are a few samples of our findings. Hoping this will be of help to your research endeavor.

Thanks and best regards.

Dr. Aleta C. Fabregas
Associate Professor, Polytechnic University of the Philippines(PUP)

MSA University
The First British Higher Education in Egypt

26th July Mehwar Road Intersection with
Wahat Road, 6th of October City, Egypt

Tel: +202-383-71114-6 Fax: +20238371543
Postal Code: 12451
Hotline: 16672

SUGPO dataset.docx

Figure 15:Contacting other researchers

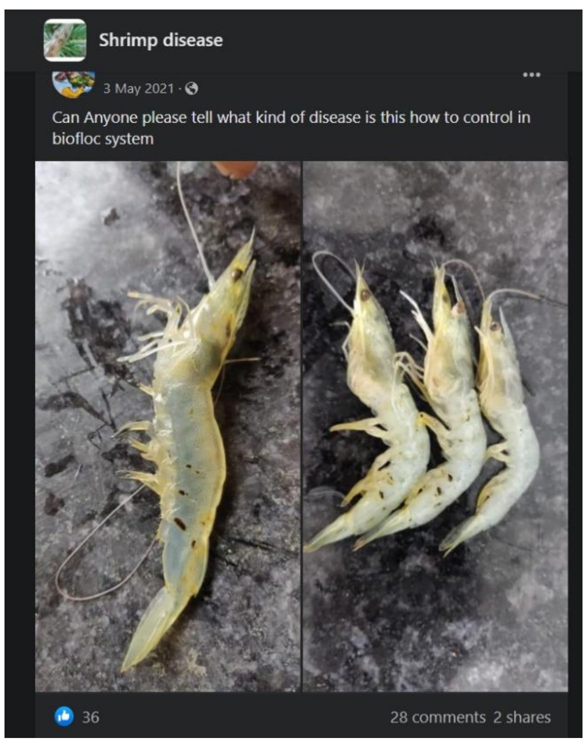


Figure 14:Comunity forums

3.1.2 Tools:

- Anaconda: is a platform for efficient developing and applying AI and machine learning models.

- Python 3: An open-source programming language that enables developers to work and integrate their systems quickly and effectively

- Jupiter notebook: Data scientists can use this open-source web application to create and share documents that include live code, equations, computational output, visualizations, and other multimedia resources**.**

- Pandas: is a Python library that provides data structures that are quick, dynamic, and powerful.
- NumPy: Arrays can be used to conduct a wide range of mathematical calculations.
- Matplotlib: Python module for data visualization and graphical graphing that is cross-platform.

- TensorFlow: It is an open-source artificial intelligence library

- OpenCV: Image processing and computer vision tasks are made easier with this tool. It's an open-source library for tasks like face detection, objection tracking, landmark detection, and more.

- Pycaret: It is a complete machine learning and model management solution that accelerates the trial cycle and increases productivity.

3.1.3 Environment:

- Local GPU, Nvidia 4GB RAM with Memory type GDDR5 and used google colab for training.

- Local as cloud will cost a money to get the resources

3.2 Methods:

3.2.1 System architecture Overview:

- First of all, we have two path first classification of shrimp disease and second behaviors detection, for classification of shrimp disease our dataset is the input of our model, but it goes for some pre-processing such as noise removal as it goes through gusaain filter and background removal to only have the shrimp not any additional surrounding and data augmentation (zooming , rotation, flip) and rescaling the image to 224 *224 , then it goes to transfer learning model (MobileNetV1) to detect the shrimp disease and its confidence level percentage as reported output. And for behaviors detection we put a camera Infront of the tank to, so the video frames go through pre-processing noise removal gusaain filter and shrimp detection using Object tracking model (YOLOv5) for better detection and trajectory extraction as input to the model (LSTM, SVM, KNN,.) then the output is reported of the abnormal behaviors analysis. We used 5 CNN transfer learning models (VGG16, resnet50, inception_V3, moblienet_V1, moblienet_v2) and one traditional model to with 300 epochs compare between them and to see the best accuracy between them. We didn't include top layers in the models (freeze it) and the input image with differ size and channel and added to each model multiple layers to its architecture. Used Adam optimizer. added to each model 3 dense layers and dropout and batch normalize layer and avarege_pooling layer, to increase the accuracy of the models. And for behaviors detection. We used gaussian blur for minimize noise and YOLO to detect the shrimps in frame. We used YOLO bounding box to get the trajectory of the shrimp every 10 seconds then takes this trajectory to classifies the trajectory for 2 or more behavior.

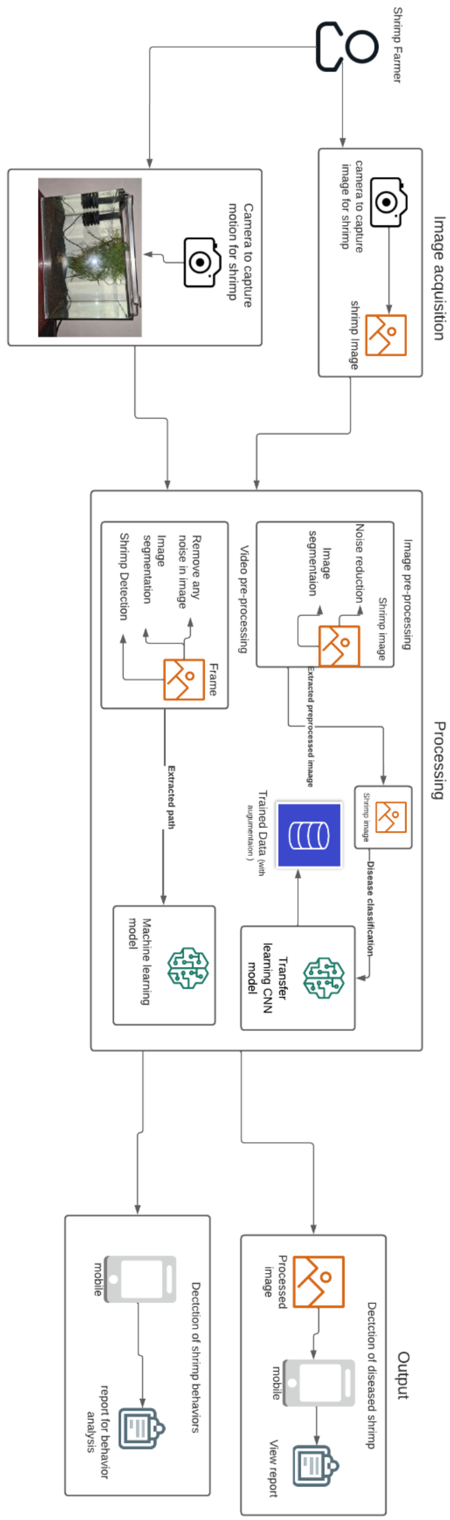


Figure 16:System overview

NUTRIVET

إتفاقية تعاون لدعم مشروع التخرج

إلى عميد كلية علوم الحاسب

تحية طيبة وبعد،

تتشرف شركة نيوتريفت بالتعاون مع **كلية علوم الحاسب بجامعة أكتوبر للعلوم الحديثة الآداب**

لدعم مشروع التخرج المقدم من الطالب/ عبد العزيز أشرف

وتحت إشراف دكتور/ أيمن عزت

بعنوان : "**التعرف على الأمراض الأنماط السلوكية للجمبري**"

وتقوم شركة نيوتريفت بتقديم الدعم الفني والتقني الإستشارات اللازمة لتجربة النموذج الأولي خلال مراحله التجريبية بمعرفتها ، كما سيتم التنسيق لعمل زيارات متبادلة بين القائمين على المشروع و الأطراف المعنية لدينا.

شركة نيوترفيت

توقيع

NUTRIVET MISR

Villa 8, Elnabatat St., West Somid, 6 of October, Giza, Egypt.
+2 (02) 383 67 978 info@nutrivetmisr.com www.nutrivetmisr.com

Figure 17:Cooperation Document

Chapter 4: **System Implementation**

4.1 System Development

Because the image consists of simply a shrimp, the first model was constructed to classify the diseased shrimps' photos using a gathered dataset. The dataset images were not in a fixed dimension. So, after reading the photos and their labels, the initial approach is to resize the input images using the OpenCV library. A simple classification model and CNN architecture Transfer learning models were constructed using Keras/Tensorflow APIs to verify that the models are able to distinguish the difference between the two categories and deliver the correct label for each class, which is 0 for Normal and 1 for Diseased. The best at distinguishing the two groups and assigning the correct label was MoblieNetV1. Also, for classifying the behaviors of the shrimps. As for behaviors path consists of only x and y coordinates by using a collected dataset from controlled experiential tank, The dataset contains the paths of shrimp where the paths length was not fixed, and the first solution was to fix the length by making them the all-same length. By padding the empty cells with the last two x and y's. or fill it was -1, We also employ Minmax, which scales and translates each characteristic based on the given range (0,1). Which max the values closer to 0 and help max it more stable, the Minmax scaler frequently produces superior results. By using Pycaret a classification machine learning models was developed to ensure that the models are able to recognize the difference between the 3 categories and give the right label for each class which is 0 for Normal, 1 for Abnormal, 3 for Hungary. The Quadratic Discriminant Analysis model was the best for classifying 3 behaviors.

Figure 19 Original Image (574 x 440) before resizing

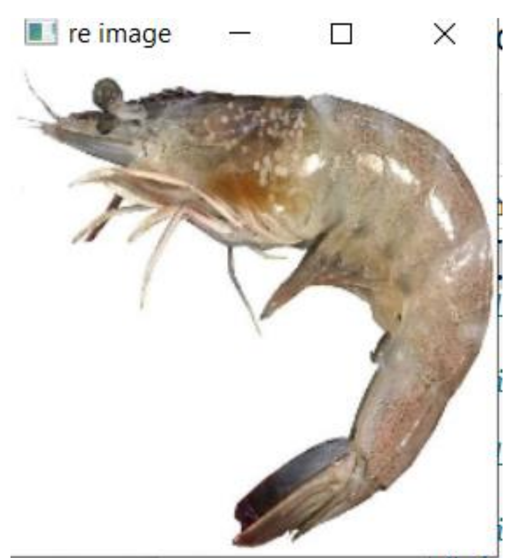


Figure 18 Resized image to (224 x 224)

Figure 21 Data after fixing the length for behaviors classification

121	Hungray	300	536	297	536	290	536	289	536	286	536	284	536	283	536	282	536	281	536	279	536	278	536	276	536	275	536	275	536	272	536	272	536	269	536	268	536	267	536	265	536	261
122	Hungray	1298	649	1297	650	1283	651	1278	651	1252	651	1225	651	1220	651	1197	651	1181	651	1180	651	1168	648	1165	647	1152	646	1149	644	1136	639	1132	638	1123	634	1121	634	1108	629	1104	629	1091
123	Hungray	299	547	299	546	301	540	302	532	302	531	301	523	300	523	296	518	294	517	287	512	284	511	283	511	275	507	268	507	265	506	246	504	229	502	221	501	220	501	214	501	209
124	Hungray	1363	682	1362	683	1347	683	1312	683	1253	683	1238	683	1183	683	1170	683	1099	678	1088	676	1047	668	1040	668	1006	664	990	662	988	661	985	661	982	661	980	661	977	660	976	656	974
125	Abnorma	1169	498	1169	494	1170	471	1170	456	1171	456	1202	486	1205	487	1228	474	1233	466	1245	445	1246	443	1254	438	1286	434	1289	433	1288	418	1288	415	1288	413	1288	406	1289	402	1289	402	1289
126	Abnorma	1094	683	1106	658	1109	649	1117	608	1118	603	1120	586	1121	581	1125	581	1137	598	1140	600	1156	613	1168	612	1172	610	1189	583	1192	580	1207	569	1207	568	1220	558	1223	555	1240	531	1254
127	Abnorma	990	673	1016	651	1025	643	1089	612	1093	611	1102	624	1103	634	1111	668	1119	666	1121	659	1136	635	1136	634	1137	633	1138	633	1135	626	1134	618	1134	617	1134	611	1134	609	1133	608	1133
128	Abnorma	730	614	761	614	786	613	786	613	789	613	791	613	792	613	802	627	802	631	803	644	803	645	804	647	821	643	829	641	853	639	856	639	857	639	860	639	861	639	861	639	863
129	Abnorma	744	501	719	468	717	458	704	433	694	442	690	447	673	474	669	449	652	412	635	403	635	403	634	406	633	408	632	413	632	413	631	413	630	413	629	413	627	413	627	413	609
130	Hungray	356	709	348	708	344	707	324	701	321	699	302	698	297	697	276	697	269	696	247	694	243	694	230	694	226	694	188	686	180	684	164	682	162	682	157	681	154	681	145	676	140
131	Hungray	1263	691	1261	691	1243	668	1241	665	1233	649	1233	648	1227	637	1226	636	1222	627	1221	624	1212	612	1210	610	1200	599	1200	598	1192	588	1191	587	1184	580	1177	567	1176	564	1173	556	1173

Figure 20 Original data before fixing the length for behaviors classification

122	Hungray	1298	649	1297	650	1283	651	1278	651	1252	651	1225	651	1220	651	1197	651	1181	651	1180	651	1168	648	1165	647	1152	646	1149	644	1136	639	1132	638	1123	634	1121	634	1108	629	1104	629	1091
123	Hungray	299	547	299	546	301	540	302	532	302	531	301	523	300	523	296	518	294	517	287	512	284	511	283	511	275	507	268	507	265	506	246	504	229	502	221	501	220	501	214	501	209
124	Hungray	1363	682	1362	683	1347	683	1312	683	1253	683	1238	683	1183	683	1170	683	1099	678	1088	676	1047	668	1040	668	1006	664	990	662	988	661	985	661	982	661	980	661	977	660	976	656	974
125	Abnorma	1169	498	1169	494	1170	471	1170	456	1171	456	1202	486	1205	487	1228	474	1233	466	1245	445	1246	443	1254	438	1286	434	1289	433	1288	418	1288	415	1288	413	1288	406	1289	402			
126	Abnorma	1094	683	1106	658	1109	649	1117	608	1118	603	1120	586	1121	581	1125	581	1137	598	1140	600	1156	613	1168	612	1172	610	1189	583	1192	580	1207	569	1207	568	1220	558	1223	555	1240	531	1254
127	Abnorma	990	673	1016	651	1025	643	1089	612	1093	611	1102	624	1103	634	1111	668	1119	666	1121	659	1136	635	1136	634	1137	633	1138	633	1135	626	1134	618	1134	617	1134	611	1134	609	1133	608	1133
128	Abnorma	730	614	761	614	786	613	786	613	789	613	791	613	792	613	802	627	802	631	803	644	803	645	804	647	821	643	829	641	853	639	856	639	857	639	860	639	861	639	861	639	863

For the image classification, A 5 transfer learning models and a CNN model was developed to see which model will be able to predict the best. But preprocessing and Data augmentation were used to help the model train. Each transfer learning model has its own preprocessing for the data, So the dataset images go to each preprocessing of the models (ResNet50, VGG16, InceptionV3, MobileNetV1, MobileNetV3) when training each of them. Preprocessing of the data is like converting the images to Gray Scale or BGR or scale data pixels between -1 and 1. Also the data after going to preprocessing, it goes through data augmentation to confusion the models. Also, the data is rescaled. The base model trainable was set to False to save the features extracted from previous. And add some layers to train the data on it. 5 Transfer learning models and one CNN model used to see which is the best to predict the 2 categories dataset, The best was MobileNetV1. Because our dataset is tiny, the model has a minimal number of parameters, which may contribute to greater classification accuracy. Furthermore, the model performs better with latency, size, and low-power models customized to match the resource limits of a variety of use cases. To develop MobileNetV1, depth wise separable convolutions are used instead of normal convolutions. Two new global hyperparameters have been added to MobileNets (width multiplier and resolution multiplier).

Figure 22 MoblieNetV1 preprocesing input

For the behavior classification, 14 model was used to see which model can predict the behavior of shrimp. the paths length was not fixed, and the solution was to fix the length by making them the all-same length.

By padding the empty cells with the last two x and y's or fil it with -1. Using Stratified K-Fold and use Minmax scaler to scale the data. The best was Quadratic Discriminant Analysis using the last two x and y's as fill for the missing cells.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
qda	Quadratic Discriminant Analysis	0.8929	0.9092	0.8613	0.9074	0.8884	0.8198	0.8291	0.1380
dt	Decision Tree Classifier	0.8429	0.8624	0.8038	0.8653	0.8369	0.7306	0.7458	0.1470
gbc	Gradient Boosting Classifier	0.8429	0.9507	0.8141	0.8663	0.8354	0.7357	0.7521	11.2660
lda	Linear Discriminant Analysis	0.8286	0.7965	0.7855	0.8565	0.8209	0.7016	0.7232	0.2020
rf	Random Forest Classifier	0.8214	0.9398	0.7877	0.8624	0.8160	0.6901	0.7178	1.0700
lightgbm	Light Gradient Boosting Machine	0.8143	0.9171	0.7766	0.8386	0.8099	0.6813	0.6939	2.7080
et	Extra Trees Classifier	0.8071	0.9359	0.7833	0.8274	0.8029	0.6676	0.6822	1.0420
ada	Ada Boost Classifier	0.7429	0.8143	0.6976	0.7514	0.7309	0.5622	0.5815	1.0060
ridge	Ridge Classifier	0.6643	0.0000	0.5313	0.6431	0.5983	0.3463	0.4266	0.0510
lr	Logistic Regression	0.6429	0.7601	0.5022	0.5791	0.5676	0.3096	0.3702	0.9480
dummy	Dummy Classifier	0.5357	0.5000	0.3333	0.2883	0.3745	0.0000	0.0000	0.0180
knn	K Neighbors Classifier	0.5286	0.6631	0.4641	0.5275	0.5129	0.2029	0.2123	0.1750
svm	SVM - Linear Kernel	0.4786	0.0000	0.4556	0.5409	0.4277	0.1949	0.2514	0.0840
nb	Naive Bayes	0.2429	0.5254	0.2893	0.1713	0.1762	-0.0641	-0.0902	0.0690

Figure 23 Behavior classification models

4.2 System Structure

Here we will explain on the coming subsections the overview of the final system we developed by explaining the flow between its components, and in the second subsection we will illustrate the class diagram and Tensorborad for the used classes in this system.

4.2.1 System Overview

The system composed of 2 main Phase, at the first we have shrimp disease classification, As the input of this phase is that shrimp farmer captures an image of a shrimp / shrimps using camera to then send it to the processing phase, the first step in processing phase is resize the image to the required dimension (224*224)

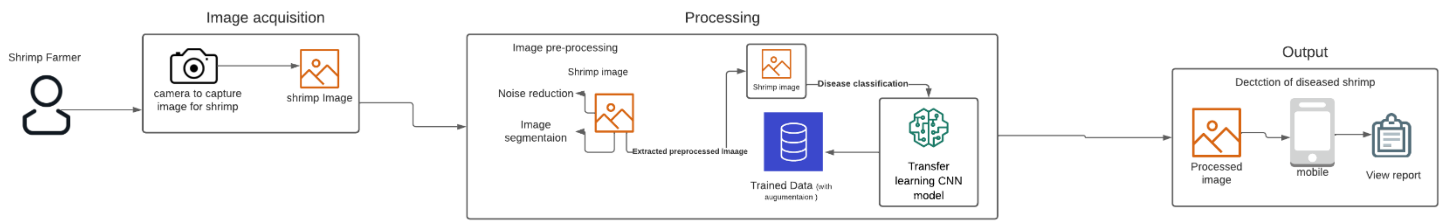
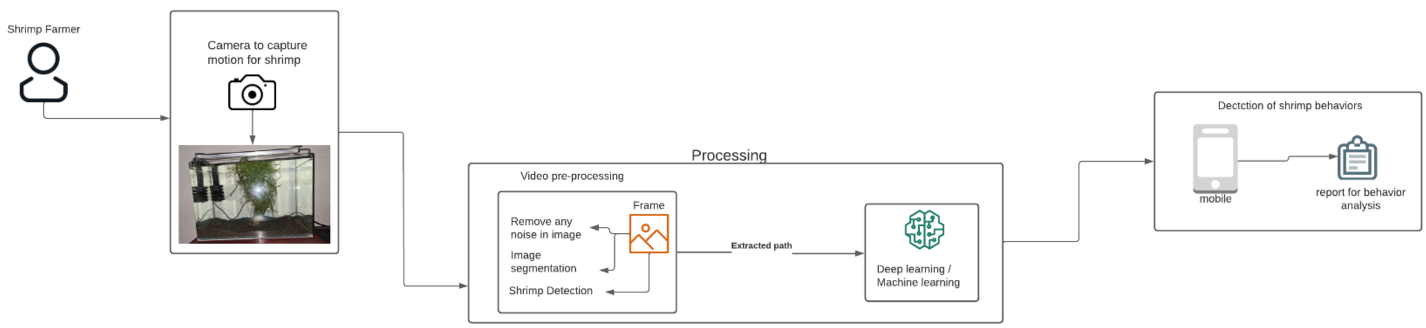


Figure 24 Image classifying phase

as the model is trained on this dimension. Second is to reduce noise in the image as the image can contain noises that effect the model to predict and classifies. Third step is removing background of the images, as background may affect the prediction of the model and send to the model unwanted features. Forth step is segmentation if there are multiple shrimps in one image, we segment the shrimps and then send it separately to the model to predict. After these 4 steps the preprocessing step is finished and now, we can send the image/images to the model to predict. The processing step here is that the image then passed to a trained transfer learning CNN model (MobileNetV1) to classifies the shrimp if it diseased or normal. Then the output of the model and the confidence score of the prediction with the image is sent to the shrimp farmer to his phone to see the if the shrimp diseased or not and which type of diseased.

As we said the system composed of 2 main Phase, at the seconds phase we have shrimp behaviors detection, As the input of this phase is that there a camera fixed in-front of the tank of capture the path / trajectory of the shrimp as live fed, Then the video goes to a pre-processing phase. The first step is reduce noise in video to help in detection of shrimp in tank, also an image enhancement technique is used to enhance the frame of the video to also help in detection of the shrimp in tank, as the water in the tank may



be unclear or impurities exits so we need image enhancement technique to help in detection. Then a deep

Figure 25 Behaviors classifying phase

learning tacker is applied to track the shrimp/shrimps in tank and draw a bounding box and id for each shrimp, then extract trajectory / path of the shrimp every 30 seconds. The extracted trajectory / path contains x and y of the shrimp path in the 30 seconds. Then the extracted trajectory / path is passed to and Machine learning model / time series analysis algorithm to see if the passed extract trajectory / path of the shrimp is normal or abnormal or hungry. Then output of the Machine learning model / time series analysis algorithm is sent in report to the frame to act open it if its abnormal or hungry.

4.2.2 Class Diagram and Tensor Board

The following figure is the TensorBoard about the whole main components of the CNN architecture for the classification task. It describes its convolutional layers, max-pooling layers, batch normalization layers, the drop out, the dense layers, and the activation functions. Also, the added layers on the pretrained models.

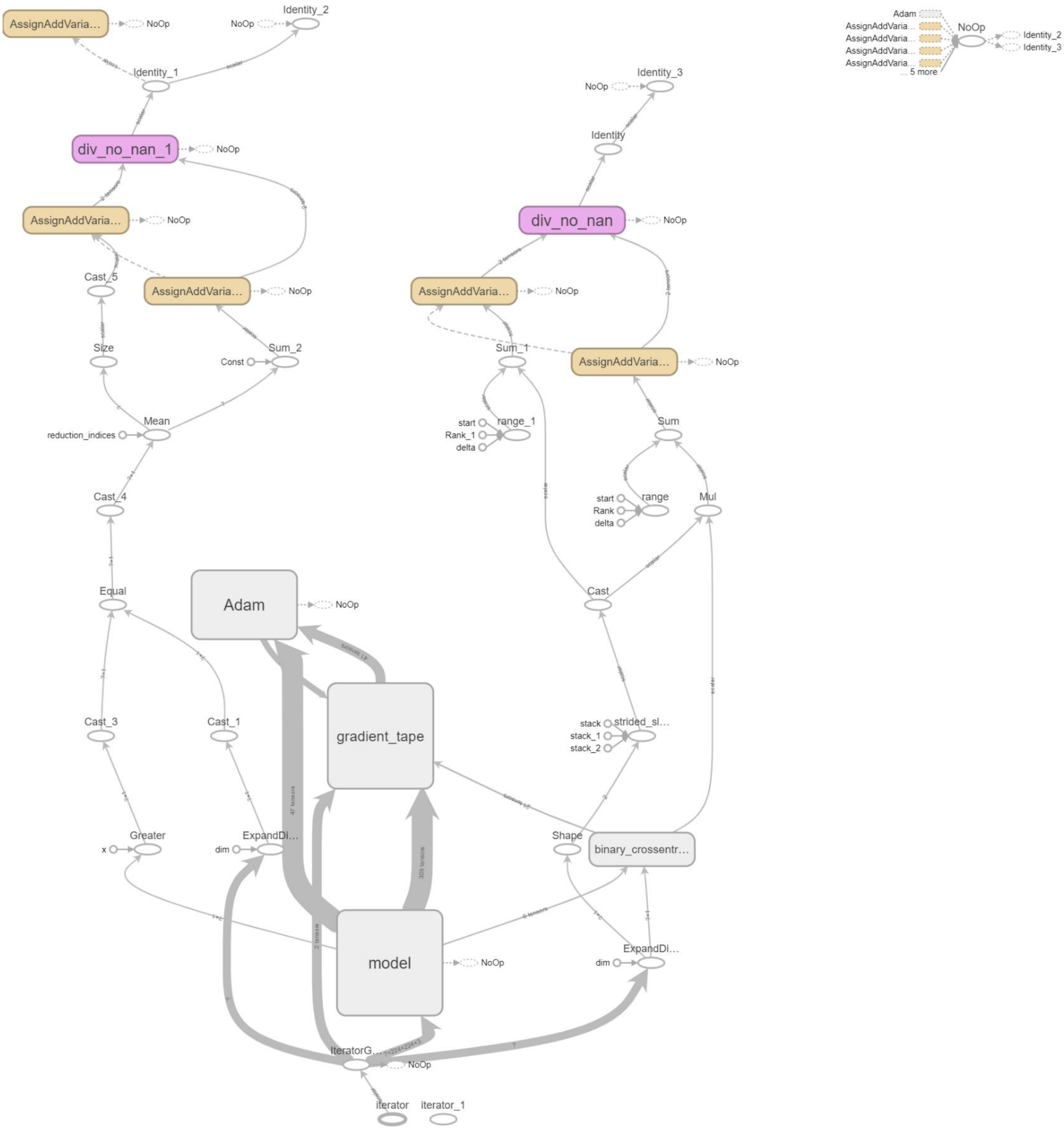


Figure 26 Tensorboard

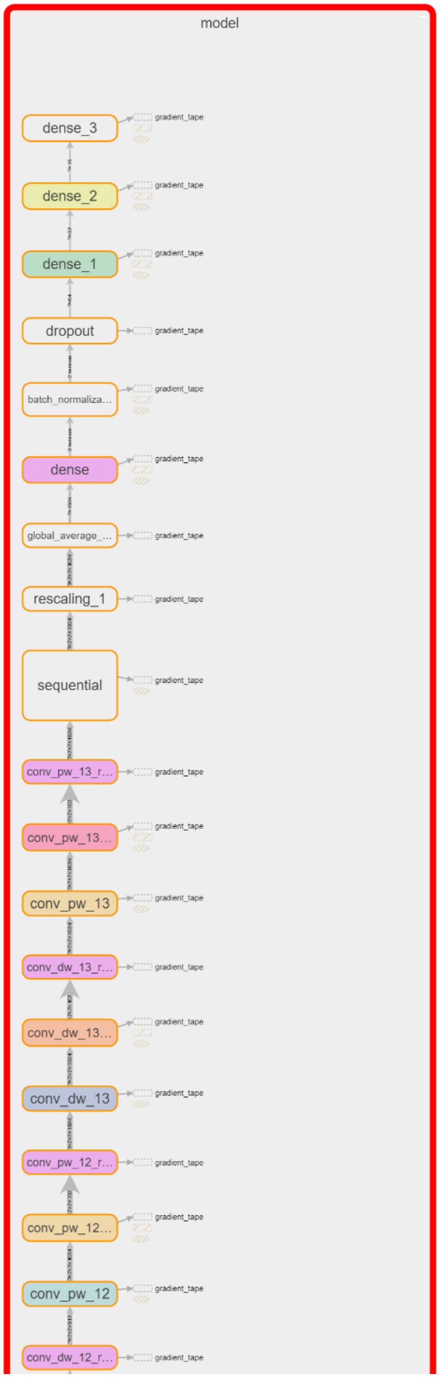


Figure 27 MoblieNeV1 last layers with fine tune layers added

Figure 28 Model layers

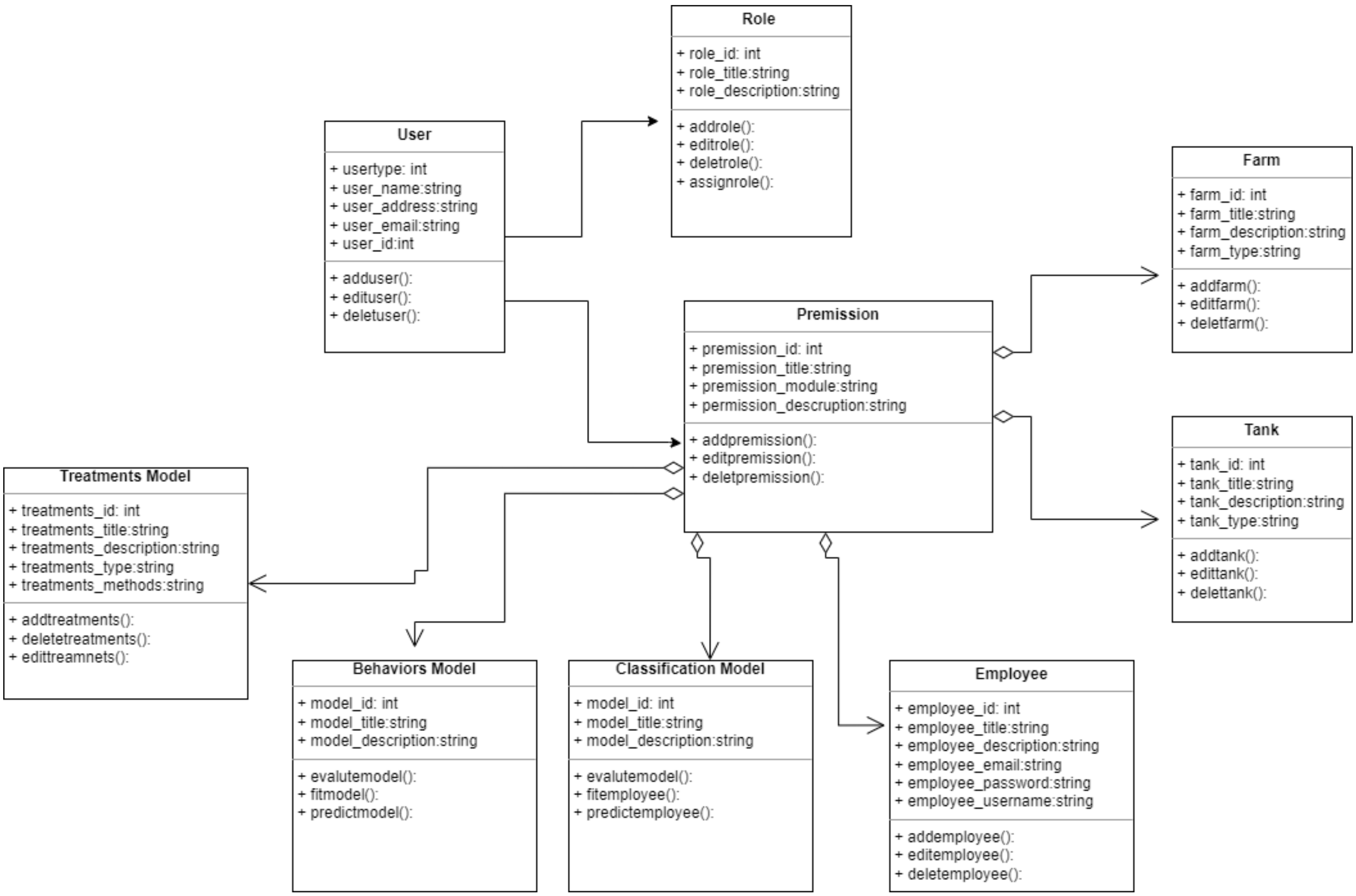


Figure 29 Class diagram

4.3 System Running

For the disease classification, First the input of training, resizing function resize the original image dataset, but we resize the image to (224* 224) and then it goes through the MobileNetv1 preprocessing

Figure 30 Original image

Figure 31 Resized image

input. The output goes to data augmentation that produce different images generated by data augmentation function to increase the small number of the dataset images and avoid model overfitting. Data

Figure 35 Original Image for image classification

Figure 34 flipped image

augmentation function uses Randomflip to flip the input images horizontality then uses Randomrotation to rotate the images by 0.1 the positive value means rotating counter clock-wise. So, the images will be

Figure 33 Original Image

Figure 32 Rotated image

rotated by 0.1 to the right.

Then Randomzoom to zoom the images by 0.1 the positive value means zooming out. So, the images will be zoomed out by 0.1. then we rescale the images using Rescaling

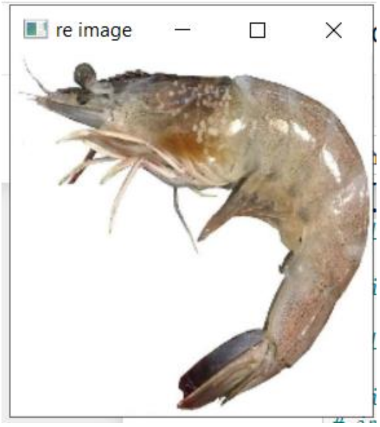


Figure 37 Original image

Figure 36 Zoomed image

function that rescales every value of an input by multiplying by scale and adding offset. We rescale an input in the [0, 255] range to be in the [0, 1] range, we would pass scale = 1. /255. All these processes are for the training data, as in inference we only will resize the image to 224*224.

For behaviors classification first the camera live capture the movements of shrimps inside the tank by detecting the shrimp and draw a bounding box around the detected shrimp then we calculate the centroid of the shrimp to get the path/trajectory every 30 seconds,

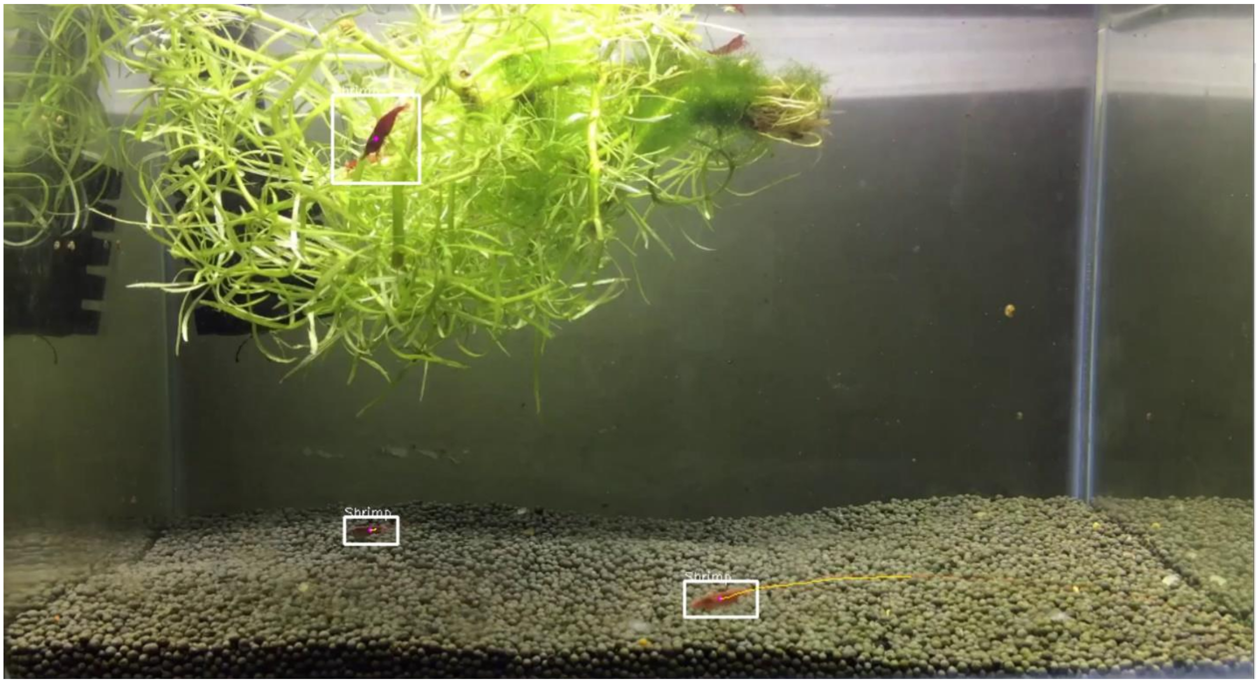


Figure 38 Shrimp detected and tracked

as the behaviors path/ trajectory consists of only x and y coordinates. There are steps for processing in this phase. The paths of shrimp length were not fixed, and the first solution was to fix the length by making first the path into array then put each x point and y point in Colum to make all-same length.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
0	Normal	1216.0	685.0	1216.0	684.0	1216.0	684.0	1216.0	683.0	1216.0	682.0	1217.0	682.0	1220.0	682.0	1222.0	682.0	1223.0	682.0	1
1	Normal	1310.0	671.0	1310.0	670.0	1310.0	669.0	1310.0	669.0	1310.0	668.0	1310.0	667.0	1311.0	668.0	1313.0	668.0	1314.0	668.0	1
2	Normal	1248.0	627.0	1247.0	627.0	1243.0	627.0	1239.0	626.0	1238.0	626.0	1236.0	625.0	1236.0	625.0	1234.0	624.0	1233.0	624.0	1
3	Normal	1042.0	566.0	1041.0	566.0	1040.0	568.0	1039.0	568.0	1038.0	568.0	1038.0	569.0	1037.0	569.0	1036.0	570.0	1035.0	571.0	1
4	Normal	971.0	641.0	970.0	641.0	967.0	641.0	966.0	641.0	959.0	642.0	958.0	642.0	952.0	642.0	949.0	642.0	945.0	642.0	
5	Normal	930.0	610.0	929.0	609.0	928.0	608.0	928.0	607.0	928.0	606.0	928.0	605.0	928.0	604.0	928.0	604.0	927.0	602.0	
6	Normal	1314.0	668.0	1313.0	668.0	1311.0	668.0	1310.0	668.0	1308.0	668.0	1307.0	668.0	1302.0	668.0	1300.0	668.0	1300.0	668.0	1

Figure 39 Path point x and y transferred in each Colum

By padding the empty cells with the last two x and y's. As by padding with the last two x and y's may make the machine learning learn that the object is standing still. And if padding last two x and y make all rows same length. And we used Minmax scaler.

121	Hungray	300	536	297	536	290	536	289	536	286	536	284	536	283	536	282	536	281	536	279	536	278	536	276	536	275	536	275	536	272	536	272	536	269	536	268	536	267	536	265	536	261
122	Hungray	1298	649	1297	650	1283	651	1278	651	1252	651	1225	651	1220	651	1197	651	1181	651	1180	651	1168	648	1165	647	1152	646	1149	644	1136	639	1132	638	1123	634	1121	634	1108	629	1104	629	1091
123	Hungray	299	547	299	546	301	540	302	532	302	531	301	523	300	523	296	518	294	517	287	512	284	511	283	511	275	507	268	507	265	506	246	504	229	502	221	501	220	501	214	501	209
124	Hungray	1363	682	1362	683	1347	683	1312	683	1253	683	1238	683	1183	683	1170	683	1099	678	1088	676	1047	668	1040	668	1006	664	990	662	988	661	985	661	982	661	980	661	977	660	976	656	974
125	Abnorma	1169	498	1169	494	1170	471	1170	456	1171	456	1202	486	1205	487	1228	474	1233	466	1245	445	1246	443	1254	438	1286	434	1289	433	1288	418	1288	415	1288	413	1288	406	1289	402	1289	402	1289
126	Abnorma	1094	683	1106	658	1109	649	1117	608	1118	603	1120	586	1121	581	1125	581	1137	598	1140	600	1156	613	1168	612	1172	610	1189	583	1192	580	1207	569	1207	568	1220	558	1223	555	1240	531	1254
127	Abnorma	990	673	1016	651	1025	643	1089	612	1093	611	1102	624	1103	634	1111	668	1119	666	1121	659	1136	635	1136	634	1137	633	1138	633	1135	626	1134	618	1134	617	1134	611	1134	609	1133	608	1133
128	Abnorma	730	614	761	614	786	613	786	613	789	613	791	613	792	613	802	627	802	631	803	644	803	645	804	647	821	643	829	641	853	639	856	639	857	639	860	639	861	639	861	639	863

Figure 42 Padding with last two x and y

	X0	Y0	X1	Y1	X2	Y2	X3	Y3	X4	Y4	...	Y506	X507	Y507	X508	Y508
0	0.884525	0.941329	0.886558	0.939811	0.898134	0.941023	0.910665	0.943250	0.910302	0.941831	...	0.936385	0.935312	0.936385	0.935312	0.936385
1	0.958366	0.917170	0.959596	0.915735	0.971229	0.915004	0.984652	0.919175	0.983940	0.917879	...	0.838635	0.907915	0.838635	0.907915	0.838635
2	0.747840	0.735979	0.750583	0.736887	0.761275	0.739809	0.771350	0.745486	0.770858	0.746792	...	0.903801	0.741248	0.903801	0.741248	0.903801
3	0.692066	0.865401	0.695416	0.865864	0.704510	0.866435	0.713892	0.871023	0.708970	0.873396	...	0.829325	0.652968	0.829325	0.652968	0.829325
4	0.926159	0.856773	0.928516	0.857266	0.942457	0.856028	0.956316	0.860705	0.956522	0.859709	...	0.775019	0.521309	0.775019	0.521309	0.775019
5	0.853888	0.810181	0.857032	0.810834	0.869362	0.810928	0.882330	0.815993	0.882883	0.818648	...	0.781226	0.866058	0.781226	0.866058	0.781226
6	0.870385	0.760138	0.871018	0.760963	0.881804	0.758890	0.893349	0.764402	0.892284	0.763901	...	0.824670	0.890411	0.824670	0.890411	0.824670
7	0.896308	0.803279	0.898213	0.803955	0.910575	0.805724	0.924046	0.810834	0.924403	0.813516	...	0.937936	0.958904	0.937936	0.958904	0.937936
8	0.293794	0.677308	0.301476	0.678418	0.312597	0.677363	0.316411	0.683577	0.318057	0.685201	...	0.716059	0.242009	0.716059	0.242009	0.716059
9	0.303221	0.742882	0.310800	0.743766	0.321151	0.743278	0.325069	0.748925	0.327458	0.750214	...	0.711404	0.354642	0.711404	0.354642	0.711404
10	0.515318	0.720449	0.522145	0.721410	0.535770	0.720729	0.545455	0.726569	0.552291	0.727973	...	0.722265	0.843988	0.722265	0.843988	0.722265
11	0.831893	0.716997	0.832168	0.717971	0.842924	0.717259	0.853207	0.721410	0.852331	0.722840	...	0.731575	0.828767	0.731575	0.828767	0.731575
12	0.555381	0.410699	0.571873	0.392089	0.585537	0.393755	0.601338	0.412726	0.594595	0.420873	...	0.363072	0.968798	0.363072	0.968798	0.363072
13	0.745483	0.716135	0.749806	0.717111	0.760498	0.716392	0.772137	0.722270	0.768508	0.701454	...	0.713732	0.756469	0.713732	0.756469	0.713732
14	0.673213	0.684211	0.677545	0.685297	0.692068	0.684302	0.702873	0.690456	0.708970	0.693755	...	0.905353	0.510654	0.905353	0.510654	0.905353
15	0.518460	0.656601	0.521368	0.656062	0.513997	0.647875	0.515545	0.650903	0.494320	0.649273	...	0.694337	0.537291	0.694337	0.537291	0.694337

Figure 41 Minmax scaling

Chapter 5: **Results and Evaluation**

5.1 Testing Methodology

5.1.1 image classification for shrimp disease

First, we have a methodology for image classification for shrimp disease. we have 6 models to test if we can predict the disease, the 6 models are MobileNetV1, MobileNetV2, VGG16, RESNET50, INCEPTIONV3, CNN. The models architectures of images of size (224* 224). To develop the models, we used an Adam optimizer and loss Binary - Crossentropy as we have only two classes to predict. We tested our six models on the 200 photos from two classes in the dataset we gathered (Diseased shrimp, Healthy shrimp). Backgrounds were eliminated from the photographs. On the training dataset, we used data augmentation to re-scale the photos to 1. /225 and randomly horizontally flip them and Random flip and Random zoom and also, while on the validation dataset, we used data augmentation to re-scale the images to 1. /225 and randomly horizontally flip them and Random flip and Random zoom. Also, we have used 16 batch size on all models. The photos from the dataset were divided into two categories: training and validation. Due to the tiny number of photos, we employed an early stopping strategy to get the most out of the models and reduce overfitting.

5.1.2 Behaviors detection

And for behaviors detection methodology, need first to build an experimental tank in which we can test our algorithm in a supervised condition. In a testing setup, a - 15 liters shrimp tank (35x20x23) was delivered to home and subjected to regular sunlight in the morning and tourch illumination at night. In addition, 5 red shrimp were purchased for our studies. A camera was put Infront the tanks to capture photographs and videos. so, our first methodology is testing the traditional trackers to track the shrimps in tank we used a 3 trackers CSRT, BOOSTING, MIL. These tracking put around the shrimps a bounding box so we get the centroid of the shrimp to get the trajectory of the shrimp, the algorithms were evaluated on 50 video length are 30 seconds. we take these trajectories to a 2 trajectory classifiers $ 1 Unistroke Recognizer and F-DTW to see which is better in classifying the trajectory. The 2 algorithms were evaluated on 21 video length are 30 seconds each. Then we used a deep learning/machine learning methodology to track the shrimps, we have the same setup as before but this time we used a YOLO model to track the shrimp and used a 14 different machine learning models (SVM, Logistic regression, Random Forest, etc.) to see which model is better to classify the behaviors of the shrimp. We also used deep learning (LSTM, BLSTM, GRU) models to see which is better but it didn’t give us the required results.

5.2 Results

<u>Case study for 3 cases for diseases classification.</u>

1 case for perfect result for the model to predict as this image belong to Diseased class as the image is in perfect case no background to affect the model, the image is noise free, a white background to help the model to extract only wanted features from the image, also the lighting and exposure is good, also the image is similar to what the model trained by.

Figure 44 Best results

1 case for acceptable result for the model to predict as this image belong to Diseased class as the image is in acceptable case there are background that may affect the model, the image contains some noise, not a completely white background to help the model to extract only wanted features from the

Figure 45 Acceptable results

image instead the background may confuse the model, also the lighting and exposure is good, also the image somehow similar to what the model trained by.

1 worst case result for the model to predict as this image belong to Diseased class as the image is in worst case even if there are no background that may affect the model, the image contains high noise some parts are missing, also the lighting and exposure is not good, also the image not similar to what the model trained by.

Figure 46 Worst case

Case study for behaviors classification cases

1 case for behaviors classification result for the model to predict as this path / trajectory belong to normal / abnormal / hungry classes as the path / trajectory contains x and y of the shrimp centroid. As the different is the algorithm for detection and tracking of the shrimp across of the tank in the best and acceptable case as the shrimp is visible for the algorithm to detect and track even if there are noise or light or exposure the model uses deep learning to help relocate and detect it.

Figure 47 Shrimp detection

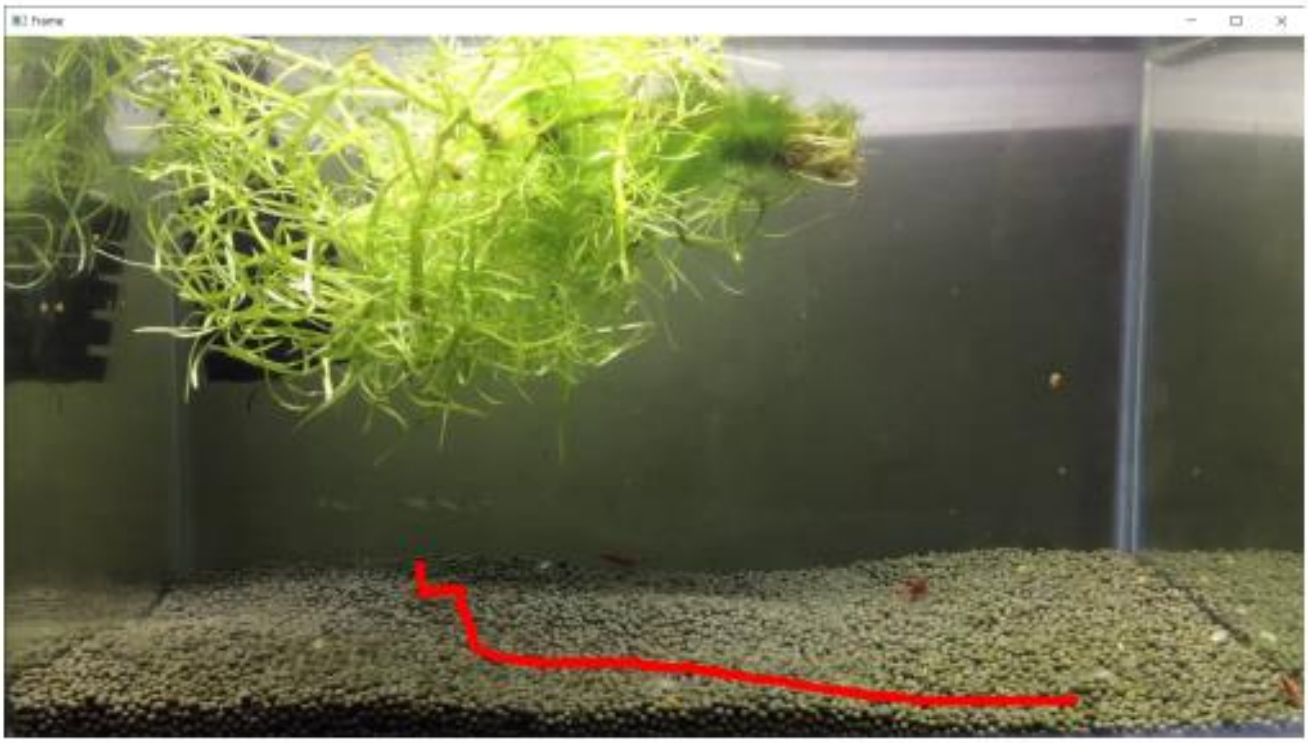


Figure 48 Shrimp path

1 case for behaviors classification result for the model to predict as this path / trajectory belong to normal / abnormal / hungry classes as the path / trajectory contains x and y of the shrimp centroid. As the different is the algorithm for detection and tracking of the shrimp across of the tank in the worst case as the shrimp is

Figure 49 Worst case for shrimp detection

not visible for the algorithm to detect and track also there are noise or the pose of the shrimp is not visible applicable for the model.

5.3 Limitations:

As for image classification for disease or behaviors classification. The dataset was the hardest the procedure to do, as there is no dataset available for this kind of topic over the internet and when we tried to reach other researchers that worked on this topic before but they cannot give us their dataset due to copyrights and some other helped us with small out of images. So, we reached out for nutrivet company to help us gather the dataset for both for image classification for disease or behaviors classification, the company sent us some images for different diseases. So, we scraped the rest of the images from the internet. We scraped form community google images and community website and Facebook community forums and YouTube videos. We collected 200 images these images These images had their background removed. Any unwanted data in the background was removed. The dataset contains images that are of different resolutions and different sizes and also the images need preprocessing as it was hard to get features form it. The dataset contains multiple diseases to make diversity to the dataset and the learning model. Using data augmentation, which improved the findings by avoiding overfitting. The quality of the images could not be improved because many of them had poor lighting or the angle of the images lost numerous details and aspects about the actual thing. And if these issues were addressed, the model may produce superior outcomes. As for behaviors classification, first we need gather a dataset for the tracking algorithms as mentioned before there is no available datasets for shrimp detection se we scraped form community google images for shrimps in different size and shapes. Also, we must build experiment shrimp tank in which to test our algorithm under supervised conditions. A - 15 liters shrimp tank (35x20x23) was brought to a residence and treated to regular sunshine in the morning and tourch lighting at night in a testing arrangement. Furthermore, 5 red shrimp were acquired for our research. A camera was placed into the tanks to take images and movies. So, we gathered 300 images to train our detection and tracking algorithm also these images need augmentation to avoid overfitting. After that we used the build experiment shrimp tank to get the trajectory of the shrimp as we recorded the shrimp behaviors in tank and asked our experts from the company or over the internet the meaning of this behavior's. They labeled the recorded behaviors to normal, abnormal, hungry. Then we extracted the trajectory or path of the shrimps from the recorded videos and trained our model with it. We receded videos that are 2 hours long in different days as to get more diversity. the data that was extracted was converted into data frame to easier to deal with.

Shrimp Diseases Classification External Inbox ×

Abdelaziz Ashraf Wahideldin Hussein <abdelaziz.ashraf@msa.edu.eg> Sat, Aug 21, 2021, 3:35 PM

to ncngon, aezzat

This is Abdelaziz Ashraf from Egypt, October University for modern sciences and arts, faculty of computer science . I am a senior student, doing my graduation project, writing down my graduation project thesis and writing a research paper about shrimp disease Classification that is expected to be submitted very soon. I'm expected to graduate next year 2022 and I am going to write a several papers.

I hope you have a good day.

I was wondering if I can get access to the dataset of your research paper Towards Classification of Shrimp Diseases Using Transferred Convolutional Neural Networks, so that I can use it in my university graduation project.

It's a very hard task to gather the data in our country, Due to covid-19 circumstances it's hard to visit the fields. It will really help with my graduation project. And I will cite your publication and add you to my acknowledgement of my paper, and this is very important to finalize my research paper.

Thank you for your help and time.

Ayman Ezzat Atia Wed, Aug 25, 2021, 11:45 AM

Dear Dr ,Nguyen Chi-Ngon This is Dr Ayman Ezzat, Vice dean of the faculty of CS at MSA, and i am supervising this project with my student AbdelAziz. I would be

Nguyen Chi Ngon 001062 <ncngon@ctu.edu.vn> Wed, Aug 25, 2021, 11:49 AM

to Ayman, me

Dear Dr Ayman Ezzat
My project supported by a company
I do not have the right to share the dataset.
So sorry
BR
Ngon

---------- Forwarded message ---------
From: **Alet Fabregas** <aletfabregas@gmail.com>
Date: Tue, Sep 7, 2021 at 5:10 AM
Subject: Re: Shrimp disease classification
To: Ayman Ezzat Atia <aezzat@msa.edu.eg>

Dear Dr. Atia,

Good day. Since our paper was presented and published last 2018 and 2019 respectively, we are trying to retrieve the image of shrimps included in the training and experiment. Unfortunately, we found very few samples, but we are glad to share them. Initially I was asking our co-authors, our graduates now, to share these samples. We are very honored that you did notice our paper. Hoping that our paper will be recognized in your very good research. And as an associate professor in the university, hoping that we can also ask help from your university as part of collaboration to improve the quality of our research.

Attached are a few samples of our findings. Hoping this will be of help to your research endeavor.

Thanks and best regards.

Dr. Aleta C. Fabregas
Associate Professor, Polytechnic University of the Philippines(PUP)

MSA University
The First British Higher Education in Egypt
26th July Mehwar Road Intersection with
Wahat Road, 6th of October City, Egypt
Tel: +202-383-71114-6 Fax: +20238371543
Postal Code: 12451
Hotline: 16672

Figure 52 Reaching out for the other researchers

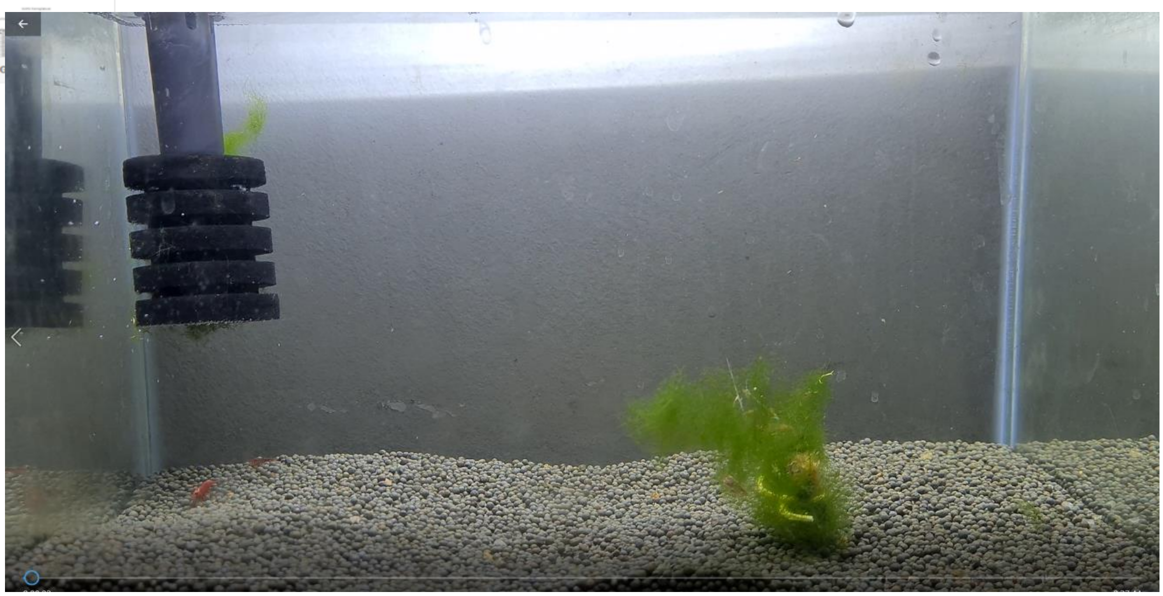


Figure 50 Our experimental Tank recorded video

5.1 Evaluation

Table 1: First Summary of the CNN Classification Models with their Accuracy

Models Number	No. of layers	Model size	Activation Function	Batch Size	Training Accuracy	Validation Accuracy
Resnet50	184	23,721,729	Sigmoid	16	100%	87.5%
Inceptionv3	318	21,936,801	Sigmoid	16	100%	85%
MobileNet V2	163	2,342,849	Sigmoid	16	100%	82.5%
VGG16	28	14,750,401	Sigmoid	16	95.63 %	90%
CNN	17	301,453	Sigmoid	16	76..68%	80%
MobileNet V1	95	3,297,345	Sigmoid	16	100%	92.5%

As shown in the previous table, different implementation to the classification CNN architecture were tried with different permutations of parameters such as the activation function, model size, and no of layers size until the best training and validation accuracy were accomplished. The first graph represents the training and validation loss and it could be noticed that the training loss is 0.9875 and the validation loss is 0.8344, and the second graphs represents the training and validation accuracy.

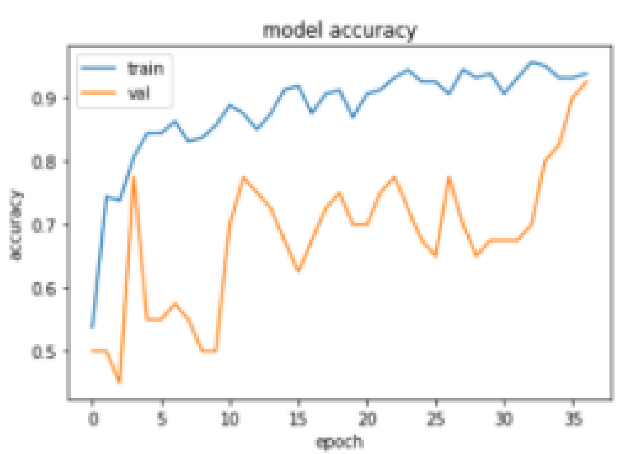


Figure 54 Model accuracy

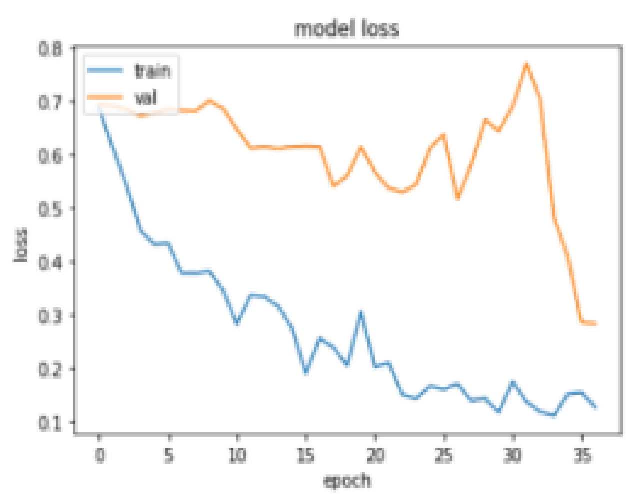


Figure 53 Model Loss

5.1.2 Behaviors classification

For the behavior classification, 14 model was used to see which model can predict the behavior of shrimp. The 14 machine learning models was tested to see the best model as well as some accuracy, f1 score, kappa. recall, prec. First, we used Bootstrap on our model as our data contains only 140 records so we used bootstrap which helped us to increase it to 189 records, furthermore we used Stratified Cross validation as to estimate the performance of the models 10 folds were used. Second, we trained the models

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
qda	Quadratic Discriminant Analysis	0.8929	0.9092	0.8613	0.9074	0.8884	0.8198	0.8291	0.1380
dt	Decision Tree Classifier	0.8429	0.8624	0.8038	0.8653	0.8369	0.7306	0.7458	0.1470
gbc	Gradient Boosting Classifier	0.8429	0.9507	0.8141	0.8663	0.8354	0.7357	0.7521	11.2660
lda	Linear Discriminant Analysis	0.8286	0.7965	0.7855	0.8565	0.8209	0.7016	0.7232	0.2020
rf	Random Forest Classifier	0.8214	0.9398	0.7877	0.8624	0.8160	0.6901	0.7178	1.0700
lightgbm	Light Gradient Boosting Machine	0.8143	0.9171	0.7766	0.8386	0.8099	0.6813	0.6939	2.7080
et	Extra Trees Classifier	0.8071	0.9359	0.7833	0.8274	0.8029	0.6676	0.6822	1.0420
ada	Ada Boost Classifier	0.7429	0.8143	0.6976	0.7514	0.7309	0.5622	0.5815	1.0060
ridge	Ridge Classifier	0.6643	0.0000	0.5313	0.6431	0.5983	0.3463	0.4266	0.0510
lr	Logistic Regression	0.6429	0.7601	0.5022	0.5791	0.5676	0.3096	0.3702	0.9480
dummy	Dummy Classifier	0.5357	0.5000	0.3333	0.2883	0.3745	0.0000	0.0000	0.0180
knn	K Neighbors Classifier	0.5286	0.6631	0.4641	0.5275	0.5129	0.2029	0.2123	0.1750
svm	SVM - Linear Kernel	0.4786	0.0000	0.4556	0.5409	0.4277	0.1949	0.2514	0.0840
nb	Naive Bayes	0.2429	0.5254	0.2893	0.1713	0.1762	-0.0641	-0.0902	0.0690

Figure 55 Experimental models with minmax scaler

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
qda	Quadratic Discriminant Analysis	0.8857	0.9004	0.8685	0.9106	0.8814	0.8126	0.8269	0.1260
gbc	Gradient Boosting Classifier	0.8500	0.9529	0.8224	0.8711	0.8424	0.7482	0.7630	10.7840
lda	Linear Discriminant Analysis	0.8286	0.7965	0.7855	0.8565	0.8209	0.7016	0.7232	0.2080
rf	Random Forest Classifier	0.8143	0.9413	0.7766	0.8560	0.8086	0.6781	0.7046	0.9550
lightgbm	Light Gradient Boosting Machine	0.8143	0.9213	0.7730	0.8399	0.8084	0.6799	0.6949	2.4280
dt	Decision Tree Classifier	0.8071	0.8236	0.7679	0.8340	0.8014	0.6665	0.6851	0.1450
et	Extra Trees Classifier	0.8071	0.9344	0.7770	0.8273	0.8017	0.6656	0.6819	0.9280
ada	Ada Boost Classifier	0.7429	0.8150	0.6976	0.7514	0.7309	0.5622	0.5815	0.9410
ridge	Ridge Classifier	0.6571	0.0000	0.5230	0.6477	0.5895	0.3286	0.4156	0.0440
lr	Logistic Regression	0.6500	0.7496	0.5063	0.5825	0.5729	0.3200	0.3820	0.8200
knn	K Neighbors Classifier	0.5571	0.6692	0.4889	0.5434	0.5384	0.2460	0.2546	0.1480
dummy	Dummy Classifier	0.5357	0.5000	0.3333	0.2883	0.3745	0.0000	0.0000	0.0150
svm	SVM - Linear Kernel	0.4571	0.0000	0.3802	0.3413	0.3475	0.0849	0.1018	0.0740
nb	Naive Bayes	0.2429	0.5254	0.2893	0.1713	0.1762	-0.0641	-0.0902	0.0680

Figure 56 Experimental models with zscore

on the data without any preprocessing but the accuracy wasn't satisfying then we used 3 scaling techniques to which scaler will get us the best results. Zsores. Robust and Minmax scaler was used the best one was the Minmax.

For object tracking and detection we used yolov5 with SGD optimizer with 16 batch size and used yolo small weights and accuracy of 95%.

Quadratic Discriminant classifier is used in machine learning to differentiate measurements of two or more types of objects or events using a quadric surface. It is a more generalized linear classifier. It relaxed the assumption that the mean and covariance of all the classes were equal.

We also tried Deep learning approach with bootstrap using LSTN, BLSTM, RNN but the models were giving us and NAN value for Loss in both validation and train also after solving this problem by reducing our model complexity and used Regulzations the accuracy wasn't satisfying as I reached in BLSTM 100% Training but 77% validation

Models Number	**No. of layers**	**Activation Function**	**Batch Size**	**Training Accuracy**	**Validation Accuracy**
LSTM	7	Softmax	32	100%	75%
RNN	7	Softmax	32	100%	73%
BLSTM	7	Softmax	32	100%	77%

Time Performance

In terms of time, employing Google Colab GPU resources over Spyder environments and Keras APIs on the local PC CPU provided a substantial gain. When the time for the identical architectural models on the local PC CPU was 76s per epoch and 4s/step, it took 18s per epoch and 1s/step on Google Colab GPU.

Chapter 6: Conclusion and Future Work

6 Conclusion

To summaries, a fully automated system for classifying shrimp behaviors and disease This thesis was constructed. After exploring different techniques to find the most beneficial approach capable of tackling this goal problem, the choice was taken to employ the deep learning / machine learning approach and its various algorithms to develop a successful system that is considered as an ideal system for diverse farm styles. First, a classification method was implemented to discriminate the photos of healthy and ill shrimp. And the method used for this work was the pretrained MoblieNetV1 CNN-based model, which is a technique that is very known for similar classification jobs. This algorithm has proven its ability to differentiate between these objects. Second, it was the beginning by building a classification system to differentiate between normal behaviors, hungry behaviors, and aberrant behaviors. The Quadratic Discriminant Analysis was chosen as the technique for this challenge since, which is a technique that is very known approach for alike classification problems. This algorithm has proven its ability to discriminate between them behaviors. As for shrimp disease classification and behaviors classification, we believe that if the data were more homogeneous or larger in size, this approach would yield more accurate results. We believe that the system could be commercialized to benefit individuals such as shrimp farmers or companies that own tanks for shrimp breeding, allowing them to improve their productivity. and to monitor shrimp tanks and save human resources and save time.

6.1 Problem Issues

6.1.1Technical issues:

System requires great computing power and great time on the CPU, PC and it crashed by using a complex architecture of models. Furthermore, requires high power and high-end components to operate continuously without crashing.

6.1.2 Scientific issues:

A large fixed size of picture padding of 500 × 500 was chosen to integrate as many photographs as possible for model training and validation, but this affected the batch size since more than 5 batch sizes produced a system out of resources crash. The usage of some models with extremely large parameters caused the system to slow down and crash.

6.2 Future Work

In future study, deeper pre-trained models such as EfficientNetV2M and EfficientNetV2L will be employed for shrimp illness classification in order to raise the MIoU value and accuracy, as well as to try to stabilize our model using additional dataset to train and validate. Furthermore, more classes will be introduced and added to be segregated by our model by more employing cloud service provider resources such as AWS and Azur. For behaviors categorization, we propose applying advance deep learning techniques such as BIRNN, R-CNN and GRU, as well as adding additional classes and collecting more data.

6.3 ACKNOWLEDGMENT

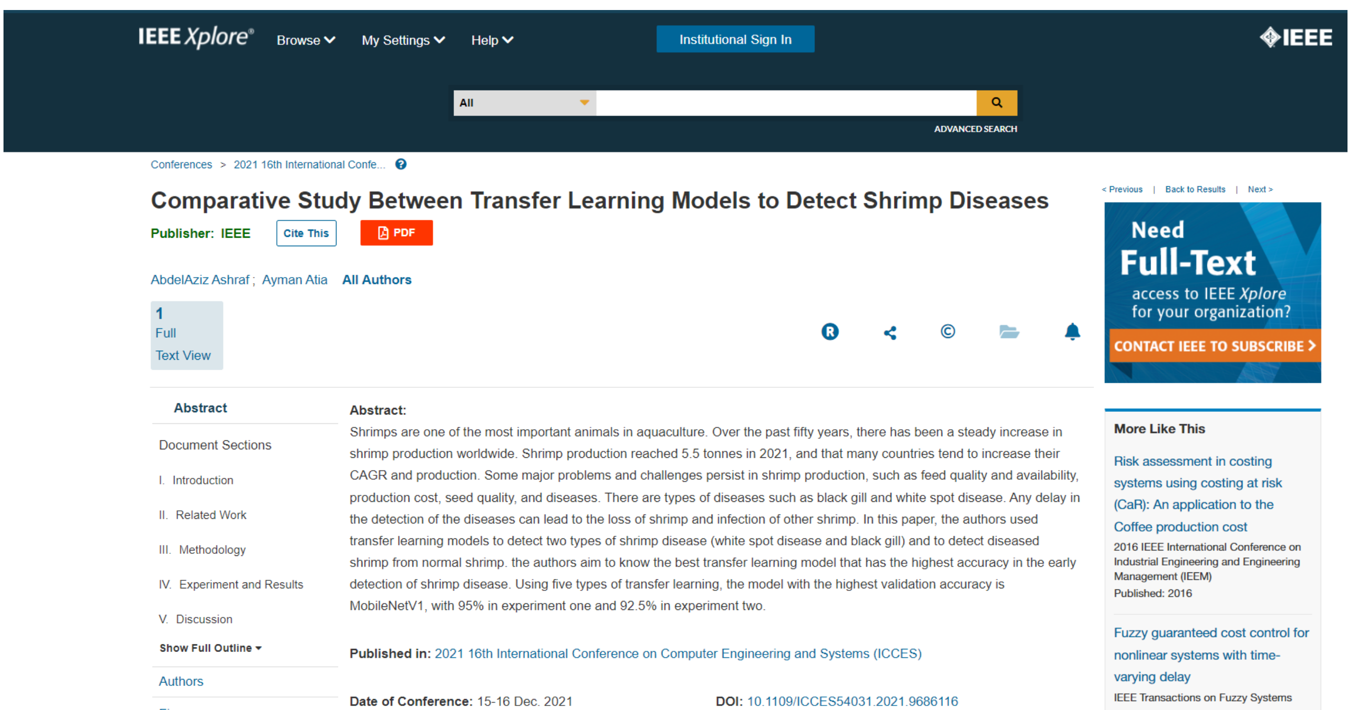
We are honored to work with the Nutrivet company. We would like to thank

them for their help and support also, who helped in this study and dataset collection.

Appendix:

We published a paper on this project named comparative study on between transfer learning models to detect shrimp diseases.

Comparative_study_between_transfer_learn



References

[1] M. S. Ahmed, T. T. Aurpa, and M. Azad, “Fish disease detection using image based machine learning technique in aquaculture,” ArXiv, vol. abs/2105.03934, 2021.

[2] R. K. Bhadoria and S. Biswas, “A model for classification of skin disease using pretrained convolutional neural network,” 2020.

[3] N. Duong-Trung, L.-D. Quach, and C.-N. Nguyen, “Towards classification of shrimp diseases using transferred convolutional neural networks.”

[4] N. Duong-Trung, L.-D. Quach, M. Nguyen, and C.-N. Nguyen, “Classification of grain discoloration via transfer learning and convolutional neural networks,” in ICMLSC 2019, 2019.

[5] N. Duong-Trung, L.-D. Quach, M.-H. Nguyen, and C.-N. Nguyen, “A combination of transfer learning and deep learning for medicinal plant classification,” in Proceedings of the 2019 4th International Conference on Intelligent Information Technology, 2019, pp. 83–90.

[6] A. C. Fabregas, D. Cruz, and M. D. Marmeto, “Sugpo: A white spot disease detection in shrimps using hybrid neural networks with fuzzy logic algorithm,” in Proceedings of the 6th international conference on information technology: IoT and Smart City, 2018, pp. 199–203.

[7] F. Jakjoud, A. Hatim, and A. Bouaaddi, “Deep learning application for plant diseases detection,” in Proceedings of the 4th International Conference on Big Data and Internet of Things, ser. BDIoT’19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3372938.3372983

[8] Z. Liu, “Soft-shell shrimp recognition based on an improved alexnet for quality evaluations,” Journal of Food Engineering, vol. 266, p. 109698, 2020.

[9] N. Minh Khiem, Y. Takahashi, D. Oanh, T. Hai, H. Yasuma, and

N. Kimura, “The use of machine learning to predict acute hepatopancreatic necrosis disease (ahpnd) in shrimp farmed on the east coast of

the mekong delta of vietnam,” Fisheries Science, vol. 86, 05 2020.

[10] T. Morimoto, T. T. Zin, and T. Itami, “A study on abnormal behavior

detection of infected shrimp,” in 2018 IEEE 7th Global Conference on

Consumer Electronics (GCCE). IEEE, 2018, pp. 291–292.

[11] L.-D. Quach, L. Q. Hoang, N. Duong-Trung, and C.-N. Nguyen,

“Towards machine learning approaches to identify shrimp diseases based

on description.”

[12] M. M. Sankar, C. N. Rao, G. Sailaja, N. Bhuvaneswary, and P. Gunasekhar, “White spot syndrome virus detection in shrimp images using

image segmentation techniques,” International Journal of Advanced

Research in Computer Science and Software Engineering, vol. 3, no. 9,

2013.

[13] R. S. Sankar, V. Jenitta, B. Kannapiran, and K. Valarmathi, “Diagnosis

of shrimp condition using intelligent technique,” in 2017 IEEE International Conference on Electrical, Instrumentation and Communication

Engineering (ICEICE), 2017, pp. 1–5.

[14] J. Sikder, K. I. Sarek, and U. Das, “Fish disease detection system: A

case study of freshwater fishes of bangladesh,” 2021.

[15] J. Wu, B. Li, and Z. Wu, “Detection of crop pests and diseases

based on deep convolutional neural network and improved algorithm,”

in Proceedings of the 2019 4th International Conference on Machine

Learning Technologies, ser. ICMLT 2019. New York, NY, USA:

Association for Computing Machinery, 2019, p. 20–27. [Online].

Available: https://doi.org/10.1145/3340997.3341010

[16] https://medium.com/ml-research-lab/what-is-object-detection-51f9d872ece7

[17] https://www.sciencedirect.com/science/article/pii/S0022201120300938