Nile Tilapia Size Estimation and Tracking Using YOLOv5 and DeepSORT

Angelica Ann Romero*[*1], Kathy Dela Cruz [1], Daniela Ocampo [1], Gideon Palabasan [1], Mary Anne Salac [1], Vincent Kyle Sison [1], Emmanuel Trinidad [2]

[1]*Student, Department of Electronics Engineering, Don Honorio Ventura State University, Bacolor Pampanga, Philippines.*

[2]*Faculty, Department of Electronics Engineering, Don Honorio Ventura State University, Bacolor Pampanga, Philippines.*

[*]*Corresponding Author: 2019990953@dhvsu.edu.ph*

Abstract: - The growing industry of Nile tilapia aquaculture evolved along with the current developments in technology. The manual grading or measurement of fish remained an inconvenience in terms of time consumption and labor. This paper compared the performances of three of the known state-of-art one-stage object detectors; You Only Look Once (YOLO) v5, RetinaNet, and EfficientDet, by training them on Nile tilapia dataset. The fish detection results show that YOLO with 88.1% mean average precision (mAP) and 83% F1-score at 80 epochs, outperforms the other two algorithms. The YOLO algorithm was then deployed and used as a detector of Nile tilapia and integrated with DeepSORT for real-time fish identification and tracking using a single web camera in an experimental controlled environment. The resulting system measurement produced accuracies of 70.06% and 52.10% for length and height measurements, respectively. Unfortunately, DeepSORT shows inconsistent and frequent ID switching due to occlusion. Even so, the fish detection was done successfully and can be instrumental in improving aquaculture for Nile Tilapia monitoring.

Key Words: — ***Aquaculture, DeepSORT, YOLOv5, Fish Detection.***

I. INTRODUCTION

One of the world's expanding food production businesses because of the huge need for protein from animals is aquaculture. Aquaculture refers to the raising of aquatic species in artificial environments or cages that are either controlled or semi-controlled [1]. The Philippines has been acknowledged as one of the major contributors in terms of supply in fisheries. Nearly 2 million Filipino fishermen rely on the country's fisheries industry for their living, which also contributes significantly to the national economy. The most widely cultivated freshwater fish in the Philippines is the Nile tilapia (Oreochromis Niloticus) which accounts for 26.84% of inland fish catch and is the highest among the species [2], [3].

Manuscript revised July 31, 2023; accepted August 01, 2023. Date of publication August 02, 2023.

This paper available online at www.ijprse.com

ISSN (Online): 2582-7898; SJIF: 5.59

Furthermore, the tilapia business supports the country's growing population and depends on farming and fishing for a living (FAO). However, in a report on the decline of tilapia production [4] in the Philippines, the major causes include high temperature within fish ponds, poor quality breeding, high cost, lack of government assistance, and lack of capital. Unhealthy living environments were also a consideration for their survival [5].

Fish length is one of the primary metrics used in fisheries to determine fish reproduction, recruitment, growth, and mortality. Since the fish must be acquired in large quantities and measured one at a time, the current method for collecting these length samples, which is traditionally done by manual measurement is time-consuming and inconvenient. The aquaculture industry has been revolutionized by electronic technologies, which offer real-time monitoring capabilities with minimal human intervention. Monitoring of fish in systems is essential to ensure sustainable growth and efficient use of resources.

One of the methods used in monitoring fish in aquaculture is image processing [6]. Image processing is composed of different operations in images to extract data about the subject in which the machine can make decisions such as classification

and detection also known as computer vision [7]. By using computational hardware and algorithmic techniques, the fish measurement process in aquaculture can be automated. A fish monitoring system employing image processing can use a variety of cameras, including SONAR cameras, Pentax cameras, Canon digital cameras, and underwater cameras [8].

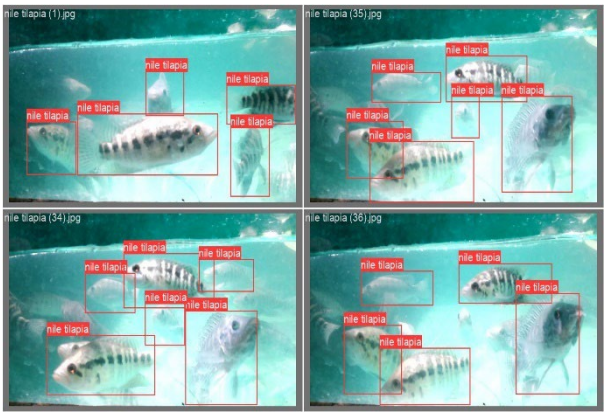


Fig.1. Annotated and Predicted Sample

Recently, neural networks are used for object identification to find and identify anything like a car, person, bike, animal, and more inside an image or video frame. Its purpose is to locate every instance of classification or category in an image and create bounding boxes around it. Various methods, including Convolutional Neural Networks (CNNs), Region-based CNNs (R-CNNs), You Only Look Once (YOLO) [9], RetinaNet, EfficientDet, and Single Shot Detectors (SSDs) are used as object detection algorithms.

This paper aims to implement and evaluate object detection and tracking of Nile tilapia using a single web camera. Additionally, the measurement estimation in terms of height and length using the bounding box predicted by the detection.

II. METHODOLOGY

In this section, the procedures of the study are discussed in detail. The approach of detection starts by obtaining a training dataset, training object detector architectures, and comparing the performances using standard evaluation metrics. A controlled experimental setup has been deployed to sustain the fishes throughout the study. The object tracking was also trained and integrated with YOLOv5. Finally, the system has been deployed to measure the fish sizes in real time and evaluated the accuracy of the measurements.

2.1 Dataset for Fish Detection

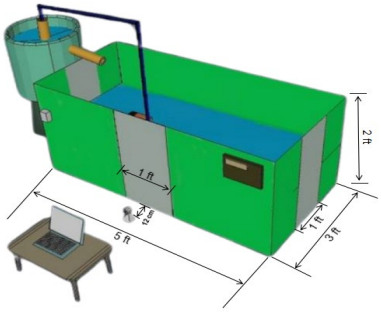
A standard infrastructure for computer vision applications was created with OpenCV to speed up the incorporation of artificial intelligence into products. Its vast and diverse library includes applications such as image detection, face recognition, size measurements, and medical diagnosis. The versatile library of OpenCV allows the integration and deployment of image recognition and processing algorithms. Furthermore, to allocate faster training and compare the performance of the three models, Google Collaboratory was used to train the models.

Initially, a total of 80 images were taken from raw videos and trained the models. However, the validation metrics show poor performances even after 100 epochs of training. To overcome this, additional images were taken, and performed data augmentation to increase the dataset to a total of 200 images. The dataset was split into 150 for training and 50 for validation. Samples of annotated images using MakeSenseAI are illustrated in Figure 1. The bounding box is shown and is used for pixel-to-ratio conversion using the 14 cm distance from the pool.

The dataset was fed to YOLOv5, RetinaNet, and EfficientDet. Comparisons were made between the three models by evaluating their performances for every 20 epochs to see the optimal epochs and avoid overfitting.

2.2 Experimental Setup

An Azolla pool 3D design is illustrated in Figure 2a with dimension $3 \times 5 \times 2$ ft was used as the setup to house the 10 Nile tilapia. Furthermore, a filter was utilized to maintain the water clarity and provide sufficient oxygen. The distance of the camera to the viewing pool was set to 12 cm which was also used for the conversion of pixel-to-ratio for the size determination. A viewing window with a size of 1 ft to further limit the viewing angle of the camera for the measurement as seen in Figure 2b.



a)

b)

Fig.2. Experimental a.) 3D Design of the Setup b.) Actual Setup

2.3 DeepSORT

DeepSORT is a deep learning extension of Simple Online Realtime Tracking (SORT) which uses the detection output from a detector like YOLO and tracks the object further. The detection from frame to frame is solved using the Kalman filter and the long-term occlusion is solved using the cosine distance metric. Usually, DeepSORT is implemented using YOLO and its advanced versions for tracking the object detected by the YOLO.

2.4 Evaluation Metrics

To evaluate the performance for object detection, the models were evaluated using four metrics, mean Average Precision (mAP), precision, recall, and the F score. The mAP is the average precision over multiple intersection over union (IoU) thresholds. The precision is calculated using (1)

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

where TP and FP are true positive and false positive, respectively. Furthermore, Recall (2) is the number of TP over the total number of TP and false negatives FN.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Lastly, the F1 score combines both the precision and recall scores of the model given in (3).

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{3}$$

In measuring the accuracy of the measurements, 5 fish measured that were part of the trained dataset, then 5 additional fish that were not part of the dataset were added to asses the actual performance. The manual measurement was done using a caliper and served as the true measurements. Furthermore, since the measurements are continuous using the software, the measurements are stored and extracted automatically in an Excel file. The mean absolute error (MAE) (4) and mean relative error (MRE) (5) are computed.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{4}$$

where y_i is the predicted measurement, x_i is the true measurement, and n is the total number of data points.

$$MRE = \frac{\sum_{i=1}^{n} \left(\frac{|y_i - x_i|}{x_i} \right)}{n} \tag{5}$$

The MRE refers to the measure of the difference between the true value and the estimated value of a quantity, expressed as a percentage or a ratio of the absolute error to the actual value. It is a way of measuring the accuracy of an estimate relative to the magnitude of the actual value.

2.5 Hardware

The study utilizes the A4Tech PK-925H web camera with a resolution of 1920×1080 pixels and a viewing angle of $70°$ a focus range of 60 cm and a 30 fps frame rate. A laptop with 12 GB RAM was used for coding and tabulating the results and acquiring data from the sensor modules. The pH and temperature sensors are implemented using Arduino UNO. The schematic diagram for the monitoring circuit is illustrated in Figure 3. The readings are initially displayed using an LCD screen but were further integrated into the code to store the readings.

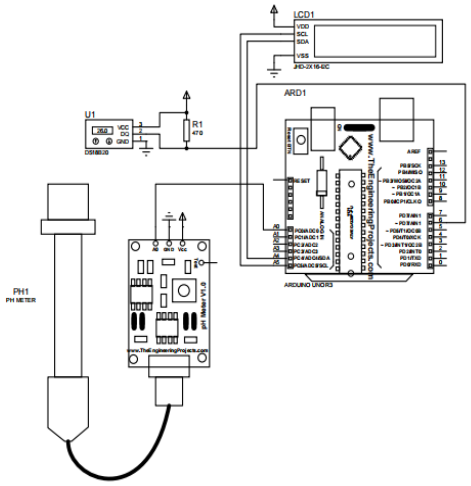


Fig.3. Temperature and pH Sensor schematic diagram

III. RESULTS AND DISCUSSION

The training images were fed in the three detection algorithms that were extracted and labeled from the raw video frames of 21 Nile tilapia. The training process began with 20 epochs, which resulted in an mAP of 77.1%, Precision of 76.9%, 71.4% Recall, and 74.05% F1 score. With increasing epochs by 20, it was observed that the metrics increased until 80 epochs and decreased at 100 epochs, except for the recall. The highest mAP, Precision, and F1 scores were recorded at 80 epochs. This implies that increasing epochs does not guarantee an increase in mAP and other metrics. At this rate, the training needs to be stopped due to the overfitting of data. It is also observed that at increased epochs, the precision decreased but the recall increased. Considering this result, it is concluded that precision and recall have an inverse relationship. However, the model needs to have high results on both metrics mentioned. To further interpret and balance these two metrics, the harmonic mean of the two called F1 scores is considered. Table 1 summarizes the percentage scores for every 20 epochs of the three detectors

Table.1. Performance of Efficient Det, RetinaNet, and YOLOv5 per Epochs

EfficientDet					
Metrics (%)	**Epochs**				
	20	**40**	**60**	**80**	**100**
mAP50	15.83	32.36	35.2	46.7	37.53
Precision	13.9	30.9	33.7	42.6	36.1
Recall	28.4	27.1	30.7	36.2	23.4
F1 Score	18.66	28.88	32.13	39.14	28.39
RetinaNet					
Metrics (%)	**Epochs**				
	20	**40**	**60**	**80**	**100**
mAP50	35.53	57.5	68.6	64.34	58.45
Precision	34.1	53.68	66.43	62.7	54.78
Recall	28.49	49.4	62.7	59.27	53.20
F1 Score	31.04	51.45	64.51	60.94	53.98
YOLOv5					
Metrics (%)	**Epochs**				
	20	**40**	**60**	**80**	**100**
mAP50	77.1	83.57	85.4	88.1	84.7
Precision	76.9	80.47	81.63	86.8	78.4
Recall	71.4	74.34	78.83	79.5	81
F1 Score	74.05	77.28	80.21	83	80

The training results from the EfficientDet algorithm yielded very low metric performance. As the training progressed with increasing epochs, the highest scores were recorded at 80 epochs, having an mAP of 46.7%, 42.6% Precision, 36.2% Recall, and 39.14% F1 Score. At 100 epochs, the mAP already declined by 9% while all the other metrics also decreased significantly. The confusion matrix for the EfficientDet trained model achieved 36% of True Positives, or the correct predictions made, and 64% False Negatives which represents the missing detections. Sample predictions of the EfficientDet are shown in Figure 4.

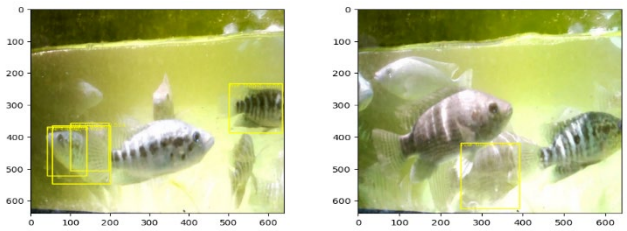


Fig.4. Sample Predictions of EfficientDet

On the other hand, the training result of the RetinaNet algorithm attained 68.6% mAP, 66.43% Precision, 62.7% Recall, and 64.51% F1 Score at 60 epochs as shown in Table 1. This demonstrates that the algorithm can detect the Nile tilapia, indicating that its performance is around average as seen with the sample predictions in Figure 5. From the given table of labeled versus predicted detections produced by the RetinaNet trained model, we can infer that it can detect the Nile tilapia but there are a lot of missing detections or false negatives. The 64.51% F1 measure is only able to perform detections that, most likely, only half of the expected detections.

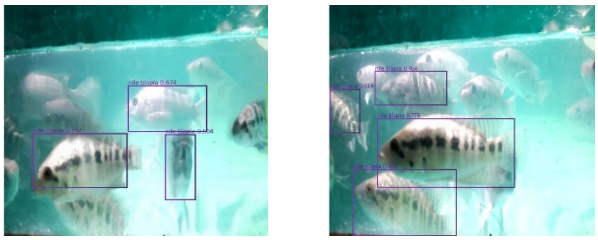


Fig.5. Sample Prediciton of RetinaNet

Using the YOLOv5, the training process began with 20 epochs, which resulted in an mAP of 77.1%, Precision of 76.9%, 71.4% Recall, and 74.05% F1 score. With increasing epochs by 20, it was observed that the metrics increased until 80 epochs and decreased at 100 epochs, except for the recall. The highest mAP, Precision, and F1 scores were recorded at 80 epochs. This implies that increasing epochs does not guarantee an increase in mAP and other metrics. At this rate, the training needs to be stopped due to the overfitting of data. The labeled test dataset is used to validate and evaluate the performance of the resulting model which is shown in the predicted test dataset. As observed from the predicted, there were cases of false positives and false negatives as seen from the sample predictions in Figure 6. From these test images, a total of 49 over 63 detections are true positive or those that are detected correctly. On the other hand, a total of 14 false negatives were counted which equates to missing detections, while only 4 detections were recognized as false positives.

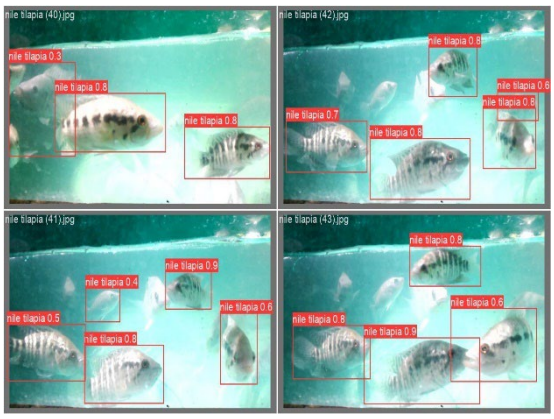


Fig.6. Sample Prediction of YOLOv5

Table 2 displays a comparison of three detection algorithms based on their metric results. Among the trained algorithms, YOLO recorded the highest scores, achieving an mAP of 88.1%. In contrast, EfficientDet and RetinaNet scored lower with 46.7% mAP and 68.6% mAP respectively. Notably, the EfficientDet model had the poorest detection results and the lowest metric scores. The selection of trained models for comparison was based on the epochs at which they achieved their highest metric scores. YOLO and EfficientDet both reached their peak at 80 epochs, while RetinaNet reached its highest score at 60 epochs. It is important to understand the role of training epochs as increasing them does not automatically guarantee high accuracy and can potentially lead to overfitting. Overfitting occurs when the model becomes too focused on the training dataset and struggles to recognize and detect new, unseen data from the test dataset. To address this, the researchers employed early stopping and augmented the dataset using existing data to artificially expand the training set. Considering that YOLO performed the best across the four-standard metrics for object detection, it was chosen as the algorithm to be implemented in the measurement system. The trained model was deployed in the PyCharm IDE and integrated with DeepSORT for fish ID integration during the coding process. The results of the system measurement were then saved in an Excel file.

Table.2. Performance of Detection Algorithms

Metrics	YOLOv5	EfficientNet	RetinaNet
mAP50	88.1 %	46.7 %	68.6 %
Precision	86.8 %	42.6 %	66.43 %
Recall	79.5 %	36.2 %	62.7 %
F1 Score	83 %	39.14 %	64.51 %

The manual measurements were done using a caliper as shown in Figure 7. Since the ID of the fish using DeepSORT is frequently changing due to successive frames, each fish was first isolated using a plastic container. This is to compare the manual measurement properly to the ID given by DeepSORT. In the first trial, 5 Nile Tilapia are first measured, then in the second trial, new 5 fishes were added.

Fig.7. Manual measurements using caliper

The true measurements for fish 1 are 19 cm and 7cm in length and height while the system measured values are 16.06cm by length and 9.03 cm by height. The accuracy for fish 1 based on

their MAE and MRE is 84.53% for length and 71.00%. In fish 2, the measured length and height were 16.06cm and 9.74cm respectively. The accuracy of the system was 89.23% and 37.51% compared to its true value of 18cm by 16cm. Fish 3 has a length of 17 cm but the average system measured based on the data was 16.58 cm, the accuracy of the length is 96.43%. The height of fish 3 is 6.5cm as the average of the system measured is 10.46cm which is higher compared to its actual height thus the accuracy was 39.02%. For fish 4, the system measured are 17.95cm and 10.46cm for length and height but the true values are 11 cm by 3 cm. Thus the accuracy for the fish 4 measurements was 36.49% and 98.89%. Lastly, fish 5 average mean length and height are 19.85cm and 9.28cm but the true values are 14.5cm and 4.5cm respectively thus its accuracy is 58.92% and -6.28%.

In the second trial with 10 fish in the pool, fish 1 with a true measurement in length is 19 cm and its height is 7 cm. While the system average measurement obtained 20.30 cm in length and 10.14 cm in height. The accuracy of fish 1 is 89.22% and 54.06% based on the absolute and relative error. Next, for fish 2 the true length and height are 21 cm and 7.5 cm while its average system measurements are 16.25 cm and 9.636 cm respectively. The accuracy attained for fish 2 is 98.57% and 75.33%. Moreover, the 3rd fish's true values are 22 cm in length and 7.2 cm in height. At the same, the system average measurements are 19.70 cm and 9.44 cm for both length and height. The accuracy based on the MAE and MRE are 89.56% and 68.77% subsequently. Moving forward, fish 4 seized true measurements of 19 cm and 7 cm for length and height. Compared to the system measurements which are 16.78 cm and 10.66 cm. Resulting in 88.32% and 47.65% in accuracy. Fish 5 achieved true values of 13 cm and 4.5 cm in length and height. The system measurements are 19.98 cm in length and 8.80 cm. Acquiring 45.89% and 4.41% accuracy. And for fish 6 has a 7 cm by 2 cm in terms of true values of length and height while it gained 110.95 cm in length and 10.60 cm using the system. Through the MAE and MRE, the accuracy is 43.51% and 310.16%. Moving to fish 7, its true values are 14.5 cm by 5.5 cm while its true values are 15.15 cm and 10.92 cm resulting in 83.06% and 1.30% accuracy. Furthermore, system measurements for fish 8 are 11.57 cm and 11.38 cm compared to its true values of 12.5 cm and 3.5 cm the accuracy based on the absolute and relative error are 70.84% and 125.17%. Next, fish 9 actual measurements are 15 cm in length and 5 cm in height its average system measurements are 23.40 cm and 9.32 cm in length and height. Gaining an accuracy of 43.99% and 13.46%.

Lastly, the fish 10 measurements are 14 cm by 4.5 cm in actual measurements while its system measurements are 23.46 cm and 9.85 cm in length and height respectively. Based on the MAE and MRE the obtained accuracy is 32.42% in length and 18.88% in height.

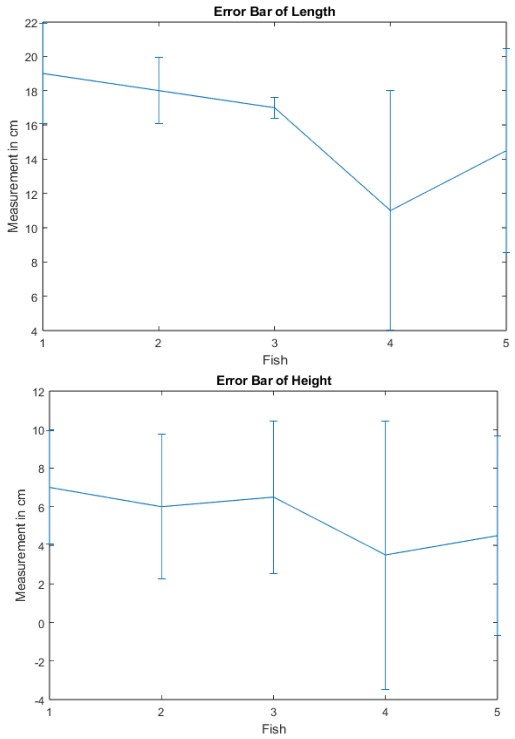


Fig.8. Error bar of length and height for the first trial

Figure 8. illustrates the error bar of length and height for the 5 fish dataset. For fish 1 the errors have distinct values while fish 2 have a lower error bar. Fish 3 on the other hand has a smaller error bar meaning it has measurements close to its actual measurements. Fish 4 has a long error bar that indicates broad data that are not close to its actual measurement. Lastly, fish 5 has sets of data with varying sizes. This is due to the multiple measurements made by the system and also due to the behavior of the fish.

In terms of length, the results based on the system measured for each fish were either accurately measured or had larger sets of measurements. In fish 1 the values measured are relative to the actual height of the fish. While Fish 2 has a significant range of data that are not close to the true value. Fish 3 similarly to Fish 2 have various sets of data that contain a high error. Fish 4 measurements were not close to the true measurements given its large portion of error.

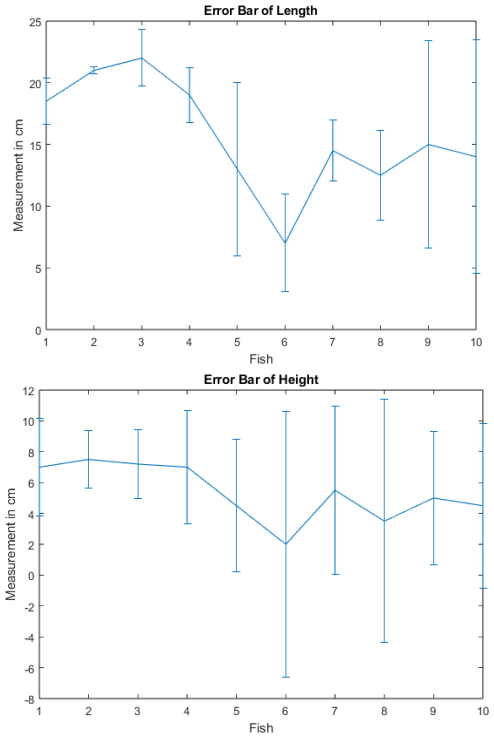


Fig.9. Error bar of length and height for the second trial

The error bar illustrated in Figure 9, the length of the 2nd test data set has different errors per fish. The first fish measured sets of data that were close to the actual value. The 2nd Fish has close data compared to its actual value. Fish 3 and 4 of the test data sets have conclusive measurements. Fish 5 has variations of data that were not close to the actual value. Fish 6 has significantly lower error values than fish 5. Fish 7 has ranges of measured value based on precise systems. Fish 8 on the other hand has a portion of data that is not close to the actual value. Fish 9 larger portions of data were not accurately measured. And Fish 10 has a high error value. In terms of height, Fish 1 has a mean error that is comparable to its actual measurement. Fish 2 has range sets of measured heights that are adjacent. Data on Fish 3 were inconclusive based on their error, Fish 4 and 5 have sets of the system measured compared to the actual value that varies. Fish 6 has large sets of data that were not close to the true value which results in high error. Values for fish 7 are significantly close compared to fish 5. While Fish 8 has a high error bar that results in a larger portion of data far from the true value. Fish 9 and Fish 10 have a high error value.

With the results of the measurement estimation in real-time, further improvements can be done by modifying and improving the equipment used. The detection and tracking are successful and can be used for behavior monitoring of fish, especially concerning the temperature and pH levels. In the sizing of the Nile Tilapia, static images can be used to capture the size instead of real-time sizing. This is due to the consistent change of the bounding boxes attributed to the fish movements and some occlusions present which remains a challenge for real-time systems. Another potential problem is the inefficiency of the pixel-to-centimeter ratio which disregards the object depth. As the Nile tilapia moves away from the foremost part of the pond, its size being read on the camera varies and the size becomes smaller compared to when it is situated on the foremost part. The use of a stereo camera for the proposed system is considered for future work to further circumvent this problem.

IV. Conclusion

In this paper, the object detection of Nile Tilapia was achieved using YOLOv5. The performance of the three algorithms was also compared in terms of standard metrics. Additionally, DeepSORT which was used for tracking purposes of Nile tilapia has shown drawbacks such as frequent ID switching and computational complexity when implemented with YOLOv5. The accuracy of the system measured varies for each fish. This means that the accuracy for individual fishes have different mean and other fish were able to achieve a high accuracy value. Nevertheless, the study suggests that it is feasible to produce smart aquaculture systems for monitoring.

Acknowledgment

The Electronics Engineering Department of Don Honorio Ventura State University is acknowledged for supporting this work.

References

[1]. Pai, K. M., Shenoy, K. A., & Pai, M. M. (2022). A Computer Vision Based Behavioral Study and Fish Counting in a Controlled Environment. IEEE Access, 10, 87778–87786.
[2]. Tahiluddin, A., & Terzi, E. (2021). An overview of fisheries and aquaculture in the Philippines. Journal of Anatolian Environmental and Animal Sciences, 6(4), 475–486.
[3]. Philippine-Fisheries-Profile2018.pdf. (n.d.).
[4]. Guerrero III, R. D. (2019). Farmed Tilapia Production in the Philippines Is Declining: What Has Happened and What Can Be Done. Philippine Journal of Science, 148(2).
[5]. Bayissa, T. N., Gobena, S., Vanhauteghem, D., Du Laing, G., Kabeta, M. W., & Janssens, G. P. J. (2021). The impact of lake ecosystems on mineral concentrations in tissues of Nile tilapia (Oreochromis Niloticus L.). Animals, 11(4), 1000.

[6]. Damanhuri, N. S., Zamri, M. F. M., Othman, N. A., Shamsuddin, S. A., Meng, B. C. C., Abbas, M. H., & Ahmad, A. (2021). An automated length measurement system for tilapia fish based on image processing technique. IOP Conference Series: Materials Science and Engineering, 1088(1), 012049.
[7]. Yoshida, S. R. (2011). Computer vision. Nova Science Publishers, Inc.
[8]. Man, M., Abdullah, N., Rahim, M. S. M., & Amin, I. M. (2016). Fish Length Measurement.
[9]. Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A Review of Yolo algorithm developments. Procedia Computer Science, 199, 1066–1073.