DeepDive: On Estimating Fish Biomass Using Monocular Unconstrained Underwater Videos

Usman Jalil
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Sector H-12, Islamabad, 44000, Pakistan
ujalil.bee20seecs@seecs.edu.pk

Syed Muhammad Abubakar
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Sector H-12, Islamabad, 44000, Pakistan
sabubakar.bee20seecs@seecs.edu.pk

Muhammad Saad
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Sector H-12, Islamabad, 44000, Pakistan
msaad.bee20seecs@seecs.edu.pk

Ahmad Salman
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Sector H-12, Islamabad, 44000, Pakistan
ahmad.salman@seecs.edu.pk

December 4, 2024

Abstract

Today, vast amounts of underwater imagery and video data are available, offering new opportunities for marine scientists and conservationists to estimate fish abundance and biomass using AI and computer vision. However, the predominance of monocular video data presents significant challenges for automatic biomass estimation, as modern tools typically require stereo vision to calculate depth, size, and mass. Additional obstacles include poor lighting, dynamic backgrounds, murky water, and fish camouflage, which complicate effective detection and classification. This study proposes a hierarchical deep learning pipeline to address these issues, employing a monocular RGB-to-depth imagery translation module within the generative AI domain. The approach combines a pixel-wise probabilistic foreground change estimator with a YOLOv8 deep neural network to accurately detect and classify fish using temporal and spatial features. A neural autoencoding algorithm, supported by fish tracking and segmentation, calculates size and mass, leading to species-specific biomass estimates. The benchmark LifeCLEF 2015 monocular dataset is used to estimate biomass for 15 species in Taiwan's coral reefs. The proposed model achieves a mean square error of 5.67% with a standard deviation of 3.27 when compared to standard sizes of fish. This research promotes advanced computer vision and machine learning techniques for conservation and habitat monitoring.

Keywords: Fish biomass; Fish detection; Fish classification; Depth estimation; and Deep learning

1 Introduction

The automated estimation of fish populations through the analysis of underwater video footage holds significant importance for marine scientists as it enables them to gauge the relative abundance and biomass of different species across various marine ecosystems. This critical information is instrumental in safeguarding endangered species from the perils of overfishing and environmental shifts. Furthermore, determining the maximum count of a specific fish species aids marine scientists in establishing environments conducive to fostering greater fish biodiversity in specific regions.

The advent of rapid data acquisition through underwater camera systems has made thousands of hours of video data available from diverse regions around the world. However, manually observing and sampling such vast volumes of data presents a labor-intensive and cost-prohibitive challenge for marine biologists and conservationists, despite the undeniable merits of its non-destructive nature [26, 27]. In contrast, the automatic sampling of fish using modern machine learning and computer vision tools is increasingly garnering attention from marine and fisheries communities as an essential requirement.

Video-based fish detection is an essential precursor to their species classification and then mass estimation, as each video frame may contain multiple fish belonging to various species. To address this challenge, numerous machine learning (ML) algorithms are available; however, they grapple with the considerable variability within species, non-rigid deformations, alterations in orientation, reduced visibility, and intricate lighting conditions.

In the past, marine scientists have relied on textural features to detect and classify fish, utilising classical methods such as Principal Component Analysis (PCA), hierarchical decision trees with Support Vector Machines (SVM), and Gaussian Mixture Models (GMM). These techniques aimed to automate fish sampling [19, 31, 41, 18]. However, given the intricate nature of the problem, recent research endeavors have shifted towards Deep Neural Networks (DNN) for tasks like fish detection, tracking, and classification. For example, fish classification based on Convolutional Neural Networks (CNN) is applied to the LifeCLEF 2014 and 2015 datasets, which featured an extensive class distribution [50]. Similarly, YOLO, a renowned object detector, is used to achieve a remarkable 93% detection accuracy for a dataset comprising 839 fish samples [56]. Another study [67] also leveraged YOLO for fish detection across multiple datasets, achieving an impressive mean average precision score of 54%. In a different approach, a multi-class SVM on AlexNet CNN features are utilised for the LifeCLEF 2015 fish detection task, securing a 74% F-score [20]. In another work, image enhancement strategy tailored for the task of coral reef fish detection is introduced for the LifeCLEF 2015 dataset [55]. Their method employ saliency maps generated through a Siamese network to effectively reconstruct input images, markedly improving visual clarity despite challenges such as variations in luminosity, the presence of moving background objects, and image blurriness. Subsequently, they applied Cascade-RCNN to the refined dataset, attaining an F-score of 81.7%.

Taking a distinct route, authors in [68] propose pre- and post-processing techniques applied to deep learning to extract fish patches for detection. GoogleNet [8] for fish detection and classification achieves an F-score of 84% for 15 species on the same dataset. On the other hand, a combination of GMM features and pixel-wise posterior analysis for fish detection in complex backgrounds is proposed, attaining an average F-score of 84.28% in the LifeCLEF 2014 fish detection challenge [51]. Furthermore, a study in [21] presents promising results on LifeCLEF 2015 dataset by adopting a hybrid approach combining temporal and CNN features. They utilise GMM and optical flow features over raw sequential images and applied the Resnet-50 [15] fish classifier for fish identification. Moreover, they employed the YOLOv3 architecture [44] on raw images in parallel, ultimately achieving F-scores of 95.47% and 91.64% for fish detection and classification, respectively. Further advancements have involved the incorporation of dense optical flow as temporal features, along with CNN analysis of fish grazing behavior on specific fish species [9]. This integration has demonstrated improvements in classifica-

tion performance, a finding endorsed by [64] in a controlled environment, achieving over 95% accuracy in a dataset of more than 1,000 videos. Additionally, [62] applied optical flow and 3D CNN for fish feeding intensity estimation, achieving 95% accuracy in their collected dataset of 24 videos under constrained conditions.

In a recent contribution by [22], a novel application of semi-supervised learning for fish detection has been introduced, specifically focusing on the FishInTurbidWater dataset. Their approach harnesses the potential of weakly-labeled datasets through an ensemble of two deep neural networks, resulting in a high level of accuracy with reduced turnaround time. The methodology involves training a self-supervised model on unlabeled data, followed by fully-supervised incremental learning using weakly-labeled data. Furthermore, the researchers implement a pioneering weakly-supervised XGBoost ensemble, incorporating pre-trained DNNs such as EfficientNet and Vision Transformer, with fine-tuning on the FishInTurbidWater dataset. This multifaceted approach yields an impressive 93.6% F-score, offering an innovative perspective on semi-supervised learning techniques for fish detection.

The application of unsupervised learning for fish tracking and segmentation has been explored using optical flow and CNN on the DeepFish video dataset, with an average precision and recall of 50% and 72%, respectively [49]. This approach, which combines temporal features with learning-based solutions, extends beyond fish fauna detection in underwater imagery [53], showcasing the benefits of feature combination in both temporal and spatial contexts.

As widely recognized, the utilization of more intricate and deeper neural architectures, as mention above, necessitates a substantial volume of training data and may still be susceptible to overfitting, as acknowledged by [32]. Additionally, despite their improved generalization capabilities and promising outcomes, these complex models struggle in capturing fish-related information in complex and dynamic underwater scenarios where fish exhibit both moving and static postures, sometimes camouflaged in the background. This generated a research gap to device end-to-end pipeline for fish detection and their species classification in a way that the information extracted by these tasks may be utilised further to more advanced problems like size and biomass estimation using available video data.

Fish biomass estimation traditionally involves multiplying the total count of fish within a designated aquatic region by the average weight of sampled fish [14]. In the last decade, many biomass estimation studies relied on taking the fish out of water for measuring it's size or other parameters like length or area. However, this method has been critiqued for its reliance on manual weighing techniques, which are not only labor-intensive and time-consuming but also prone to inaccuracies [28]. Additionally, manual handling of fish during weighing procedures has been shown to induce stress, adversely impacting their health and well-being [3]. Consequently, the conventional approach to

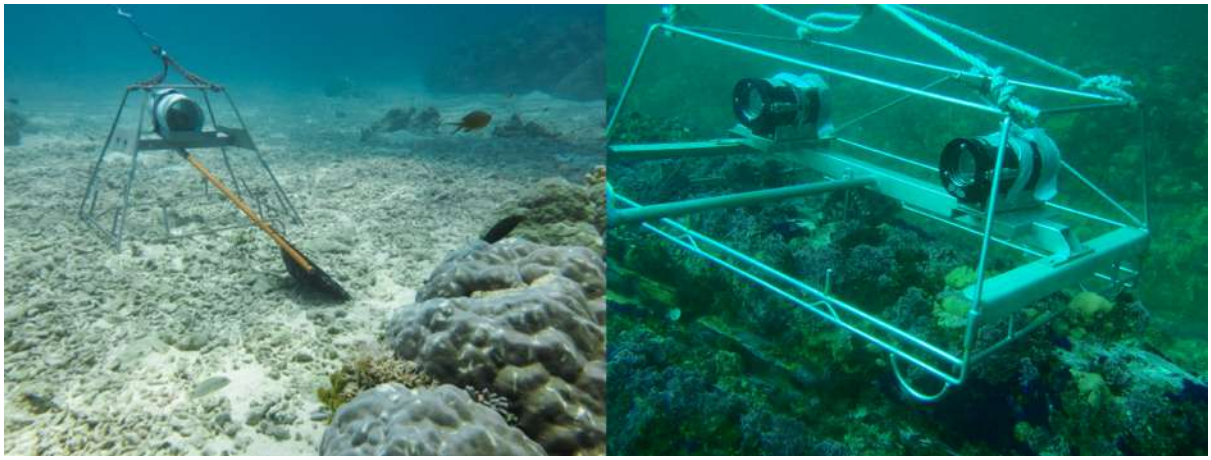

Figure 1: Monocular (left) and stereo (right) BRUVS, ©AIMS.

biomass calculation presents notable limitations.

In recent years, considerable attention has been devoted to the exploration of automatic and non-invasive methodologies for estimating underwater fish biomass [34, 63]. These methods include Machine Vision [65, 48], Acoustics [13] and environmental DNA [57]. Using advanced Computer Vision techniques can help in calculation of fish mass without the need of human intervention[73, 70]. Many of these methodologies rely on the utilisation of 2D computer vision systems and external devices, such as polystyrene boards [63], conveyor belts [23], tanks [1], and boxes [4], to facilitate the estimation of biomass alongside other parameters such as length, width, area, and weight. The utilisation of stereo-vision technology has gained popularity in addressing the challenges posed by the variability of distance and angle in the bodies of free-swimming fish, which cannot be effectively managed by 2D computer vision systems. This approach has been highlighted in recent studies [33, 47, 40].

Although the length of fish (or any other visible sizes) can be estimated directly from the underwater images, determining the fish mass can only be done approximately [4, 63]. Contemporary investigations, exemplified by the work of [63], have expanded the scope of estimation techniques, demonstrating the feasibility of estimating the mass of certain fish species, such as *Jade perch* (*Scortum barcoo*), through the measurement of fish area from a lateral perspective. Furthermore, methodological innovations, as illustrated in [38], have incorporated advanced technologies such as image processing. Their approach entails the automated collection of images of swimming fish via cameras, subsequent measurement of fish length through image processing techniques, and finally, estimation of actual fish length via approximation using third-order regression curves.

By integrating underwater stereo vision with Deep Learning methods [72], there is a potential for enhancing the efficiency and accuracy of biomass measurements, while simultaneously reducing the need for manual labor-intensive tasks. However, stereo vision based systems, as depicted in Figure 1, are relatively expensive to install on a Baited

Remote Underwater Video System (BRUVS) systems and require complex methods and heavy computations to find the exact point in both images for fish measurement task [42]. Since most of the underwater video data captured around the world for fish sampling is monocular, our primary focus in this research centers on enhancing contemporary state-of-the-art to estimate the biomass of fish by utilising monocular vision. This enhancement is achieved through a meticulously crafted algorithm designed to capture fish motion-related features while concurrently eliminating irrelevant noise and background interference. Then, by accurate fish detection in video frames followed by their species classification, biomass is estimated using estimated depth images (disparity maps) which technically require stereo vision. Our evaluation draws upon the LifeCLEF 2015 dataset, which pose challenges stemming from the underwater environment's inherent variability, characterized by poor visibility and aquatic background ambiguity. Out contributions are summarised as follows,

- A spatio-temporal hierarchical pipeline is proposed for fish detection where a pixel-wise Gaussian foreground change estimator is utilised to capture fast moving fish while YOLOv8 extract static fish candidates.
- The detected fish candidates are classified by the YOLOv8 and then tracked across the video frames to omit the missed detection and enhance the recall rate yielding better F-Score.
- A CNN-based autoencoder network is devised with residual inter-block connection enriched with multi-head attention mechanism to project monocular underwater imagery to depth maps. These depth maps help in estimating the fish size and consequently biomass of the classified species by keeping the the account of fish tracks.

2 Materials and Methods

This section discusses comprehensive details about the dataset used in this study, followed by end-to-end pipeline which comprises fish detection, species classification, image depth estimation that finally leads to biomass estimation.

2.1 Dataset

LifeCLEF 2015, a few samples of which are shown in Figure 2, has been used which includes a total of 93 videos of flv format, each containing instances of 15 distinct fish species. This dataset is a subset of a more extensive collection of underwater videos known as Fish4Knowledge, as documented by [24]. Videos in LifeCLEF 2015 are captures at

Figure 2: Sample images of LifeCLEF-2015

640 × 480 and 320 × 240 pixels. The dataset is manually annotated with fish locations (as bounding boxes) in each frame with corresponding species name which adds up to 14,000 annotated frames and 19,583 bounding boxes in total. In addition to the videos, 20,000 sample images if 15 different fish species are also provided to support training algorithms for species classification. The dataset is unbalanced in terms of number of appearance of fish species in frames as shown in the table. The LifeCLEF 2015 dataset comprises environmental variations and video distortions which presents challenges for machine learning and computer vision algorithms in yielding acceptable performance. These include blurriness, complex background, low resolution, scenes with crowded fish and pixelated noise. Table 1 details the fish species distribution, showing the total number of images available for each species in the dataset..

2.2 Methodology

Utilising deep neural networks (DNN) remains the primary strategy for fish detection and species classification in both images and videos. However, improving DNN capabilities can be achieved by integrating traditional computer vision algorithms. Given recent advancements in spatio-temporal feature extraction, our proposed methodology is inher-

Species name	**No. of images**
Abudefduf vaigiensis	434
Acanthurus nigrofuscus	2,770
Amphiprion clarkii	3,265
Chaetodon lunulatus	3,544
Chaetodon speculum	162
Chaetodon trifascialis	704
Chromis chrysura	3,859
Dascyllus aruanus	1,749
Dascyllus reticulatus	5,327
Hemigymnus melapterus	361
Myripristis kuntee	3,231
Neoglyphidodon nigroris	213
Pempheris vanicolensis	906
Plectroglyphidodon dickii	3,102
Zebrasoma scopas	343

Table 1: Fish species distribution in LifeCLEF 2015 dataset

ently hybrid as show in in Figure 3. We incorporate temporal features obtained from segmenting moving fish and eliminating static background using pixl-wise Gaussian foreground detector. Moreover, a specialized branch, powered by a modified YOLOv8 fish detector, is further reinforced by merging the outcomes of our motion-based fish segmentation algorithm. This integrated approach aims to enhance fish detection and species classification performance, particularly in addressing the distinctive challenges presented by dataset like LifeCLEF 2015. Afterwards, a deep CNN based autoencoder branch estimates the depth maps of video frames which helps in approximating fish biomass.

2.2.1 Motion-Based Detection

The first branch branch in the proposed architecture (see Figure 3 is based on motion based detection called Foreground Change Estimator (FCE), where each pixel location in a collection of video frames is represented as a distinct Gaussian probability distribution function (PDF). By combining the distribution functions for each pixel in RGB channels of a few video frames, a background is estimated. Next, the PDFs for upcoming frames are compared with the background to detect any change due to motion highlighting moving fish candidate regions in the foreground. To assign a bounding box for each segmented

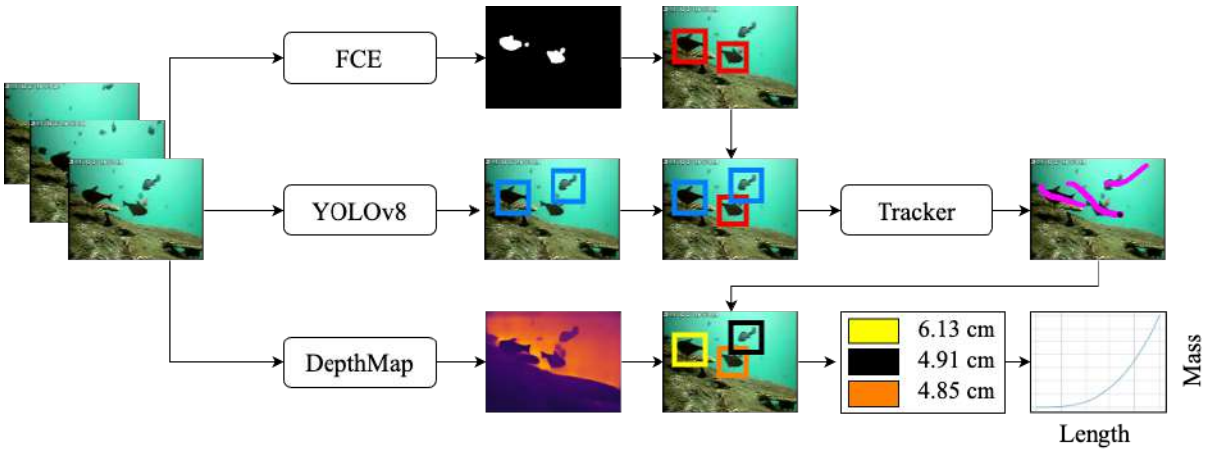

Figure 3: Proposed architecture. A temporal branch employs foreground change estimator (FCE) to capture fast moving fish. and filter noise due to dynamic light beams and motion of aquatic plants. The fish candidates are added to the outcome of YOLOv8 static fish detector. Working on the final fish detection result, a tracker module records the distinct fish tracks across the video frames. Finally, the depth map is generated using monocular RGB frames to estimate fish size and consequently fish mass, refined by unique fish tracks.

foreground output within binary images, we computed it's coordinates and mapped these bounding boxes onto the original RGB frames. This process is illustrated in Figure 4 and mathematical fomulation is given in (1).

$$p = \sum_{n=1}^{N} w_n p_n, \tag{1}$$

where p is overall PDF which is a weighted sum of constituent pixel probabilities p_n with heights w_i which is a tunable parameter with expectation-maximization algorithm [39].

Effective utilisation of FCE module necessitates a substantial dataset for training and estimating parameters such as mean and covariance of the background. This enables the model to distinguish fish from non-fish entities, encompassing diverse elements like kelp, coral reefs, sea grass beds, and other aquatic flora, as well as sessile invertebrates such as sponges, gorgonians, and ascidians, along with the physical seabed structure. Statistical patterns exhibited by stationary structures, such as coral reefs and seabed formations, differ distinctly from the movement patterns of fish. Additionally, objects with constrained movement, like swaying kelp and aquatic plants, possess statistical signatures divergent from fish movement. Leveraging temporal changes in frames, FCE can also identify unannotated fish instances. However, subsequent filtering based on Intersection over Union (IoU) criteria ensures retention of only annotated fish detection.

The FCE has its limitations if relied upon exclusively. It struggles to detect fish with minimal temporal changes, such as those that remain stationary in a video sequence, as

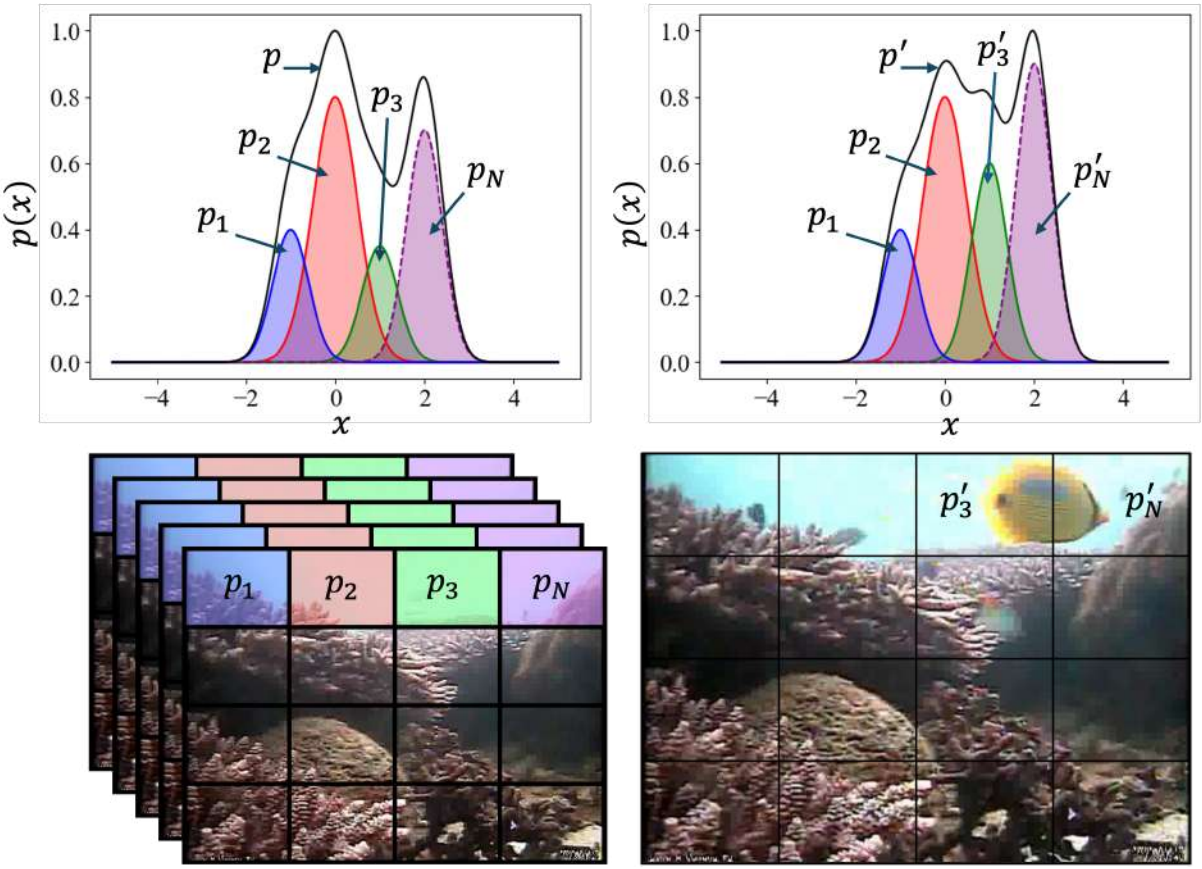

Figure 4: Illustration of FCE algorithm for foreground change detection. The pixel probabilities $p_1, p_2, p_3, ..., p_N$ may alter on appearance of any change i.e., fish in our case fish instance to get $p_1, p_2, p_3^{'}, ..., p_N^{'}$

it is designed to model temporal changes in pixel distributions, making it effective for detecting moving fish. However, due to the lack of significant changes in pixel intensities over time, stationary fish are not detected by FCE. In contrast, the static fish detector based on YOLOv8 effectively addresses the detection of stationary fish with high accuracy.

2.2.2 Static Image-Based Detection

This branch employs the capabilities of YOLOv8, an advanced real-time object detection model renowned for its efficiency and accuracy [58, 54]. YOLOv8 operates by dividing the input image into a grid and predicting bounding boxes and class probabilities for each grid cell. Unlike traditional object detection models that use a sliding window approach, YOLOv8 predicts objects in a single pass through the network, making it significantly faster. In our case, YOLOv8 predicts the bounding box coordinates on a 2D image plane, including the width and height of the box region (x, y, W, H). Additionally, it provides a confidence score for each detected fish and classifies its species from the 15 annotated species in the dataset. The working illustration of YOLOv8 is depicted in Figure 5

YOLOv8 architecture consists of multiple convolutional blocks. Each convolutional

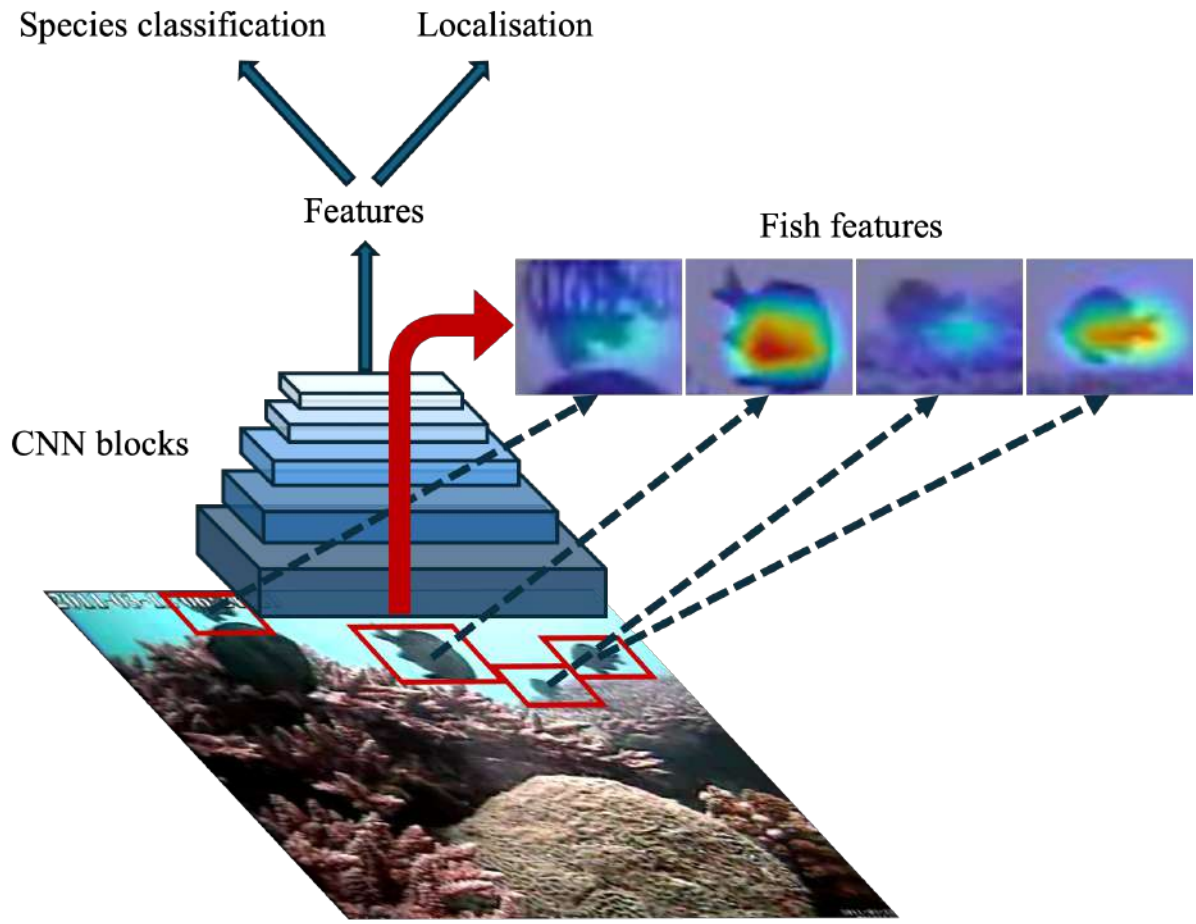

Figure 5: YOLOv8 architecture. CNN blocks extract fish-dependent textural and color-based features which are localised in an image while the classification head recognises the species.

block comprises one 2D convolutional layer, one 2D batch normalization layer which helps suppress spurious non-fish information while extracting fish-related textural features. This is followed by bottleneck blocks to sift out highly relevant features in lower dimensions. The bottleneck block consists of two 2D convolutional layers with a residual connection which helps in combining basic shape and color dependent features with fine-grained features of fish. Moreover, C2f block consists of two Convolutional blocks with multiple number of bottleneck blocks. Feature maps extracted from the first convolutional block are split, and some are fed into bottleneck blocks while the leftovers are concatenated with the output of all the bottleneck blocks. The output of the concatenation layer is then fed into the next convolutional block. This is followed by spatial pyramid pooling fast (SPPF) block. The SPPF block has two convolutional blocks along with three 2D max-pooling layers. The output of the first convolutional block, along with the output of all the 2D max-pooling layers, is concatenated before feeding it into the next convolutional block. The main purpose of SPPF is to generate a fixed feature representation of fish of various sizes in an image without resizing and introducing spatial

information loss. The detection block consists of two tracks with the same layers. The first track is for bounding box localisation with two convolutional blocks and a single 2D convolutional layer at the end. The same goes for the second track, which is for species classification.

The architecture begins with the backbone. It takes an RGB ($640 \times 640 \times 3$) image as input. It starts with two convolutional b locks and then one C2f block with residual connections enabled in 1 bottleneck block inside it. The output shape will be ($128 \times 128 \times 32$). This output is fed again into 3 repetitions of Conv Block and C2f with an output shape of ($20 \times 20 \times 256$). This output is fed into the SPPF Block. The output of the SPPF block is then fed into the neck of YOLOv8. It is first upscaled and then concatenated with the output of one C2f block from the backbone and then fed again into the C2f block. The output shape will be ($40 \times 40 \times 128$). The same process from above will be repeated again with another upscaling layer, concatenation layer, and C2f. The output of the C2f block will be fed into one Detection block with a shape of ($80 \times 80 \times 128$). The first Detection block is used for small-sized objects. Secondly, the same output will pass through the 2 blocks of (convolutional block – concatenation layer – C2f block). The output of the first block will be fed into the second detection block with a shape of ($40 \times 40 \times 128$). The second detection block is for detecting medium-sized objects. Similarly, the output of the second block (consisting of two sub-blocks and one concatenation layer) will be fed into the third Detection Block, which is used for detecting large-sized objects.

In numerous instances, both the YOLOv8 and FCE models produce bounding boxes for the same fish, leading to ambiguity in selecting the more precise bounding box. To address this issue, YOLOv8's decision is adopted preferentially, as it better avoids noise in dynamic scenes. In cases where FCE and YOLOv8 detect different fish instances, both are included in the detection count. The unified output image is then fed into the tracking module, which is explained next.

2.2.3 Fish Tracking

For fish tracking, we use a traditional and simpler computer vision-based tracking approach. For each video clip, we initialize an empty set of tracked fish T. Each frame is passed through the detection module (YOLOv8) to get the detected fish D. If T is empty, all of the detected fish are added to T. Otherwise, pairwise distances d_{ij} are computed for each $T_i \in T$ and $D_j \in D$ with i and j being the consecutive video frames. If d_{ij} is greater than a specific threshold, it is set to infinity to avoid spurious assignments across the frame. The computed distances are arranged in the form of an affinity matrix and passed to Hungarian algorithm [5] for optimal assignments of detected fish to the tracked fish. The positions of the tracked fish are then updated accordingly. Any unassigned detected fish at this point are added to T as new fish instance. We also maintain the age (frames

Algorithm 1 Fish Tracking Algorithm

Require: V: Video clip
Ensure: T: Set of tracked fish
$T \leftarrow \emptyset$ {Initialize empty set of tracked fish}
for each frame f_i in V **do**
  $D \leftarrow \text{DetectedFish}(f_i)$ {Get detected fish in frame f_i}
  if T is empty **then**
    $T \leftarrow D$ {Add all detected fish to T}
  else
    $d_{ij} = \text{Distance}(T_i, D_j)$
    if $d_{ij} >$ threshold **then**
      $d_{ij} \leftarrow \infty$
    end if
    AffMatrix $\leftarrow$ ArrangeIntoMatrix(Distances)
    Assignments $\leftarrow$ HungarianAlgo(AffMatrix) {Update positions of tracked fish}
    $T_{\text{unassigned}} \leftarrow T_{\text{unassigned}} \cup \text{UnassignedDetectedFish}$
  end if
  for each entity e in T **do**
    Update e_{age}
    if $e_{age} >$ threshold **then**
      Remove e from T
    end if
  end for
end for

since the last updated) of each entity in T and once the age exceeds a certain threshold (15 frames), the instance is removed from T. The role of tracker module in the main pipeline is explained as follows, consider the current frame f_i, with the preceding and the next frames being f_{i-1} and f_{i+1} respectively. The detector's false alarm in the current frame is reduced by calculating distances d_{ij} between all entries of T belonging to f_{i-1} and f_{i+1}. If there is an extra instance in f_i it is credited as an actual false alarm, hence removing such instances will increase precision. Similarly, tracking algorithm aids in reducing miss-classification rate by calculating pairwise distances d_{ij} between f_{i-1} and f_{i+1}, and if there is missing fish instance in frame f_i, we interpolate the fish box in that frame to reduce false negatives hence improving recall rate of the detection system. Algorithm 1 summarises the tracking process.

2.2.4 Biomass Estimation

Fish biomass is a critical measure in aquatic ecology, representing the total mass of all fish in a particular ecosystem, habitat, or population at a specific time. It helps scientists understand how many fish are present and where they are located within water environments. This information is crucial for managing fisheries effectively and protecting aquatic ecosystems. To calculate fish biomass, researchers typically use the following

relation which is widely used and accepted in fisheries science:

$$\text{Fish Biomass} = \text{Number of fishes} \times \text{Total mass observed}$$

Our primary objective is to determine the amount of fish biomass in an underwater setting. To achieve this, we require both the quantity and mass of the observed fish species. Utilising the YOLOv8 + FCE model, we identify and document the count of fish from 15 distinct species in the dataset. Previous studies extensively explored the connection between the size in square centimeters, length in centimeters, and other factors such as fish height in centimeters, with its mass in kilograms. Among the historical precedents, [61] proposed a influential method rooted in the relationship between fish length and mass, which was aimed to establish mathematical relationships among these factors, enabling the estimation of fish biomass using any of the mentioned parameters.

$$\text{Fish mass} = a \times \text{Length}^b, \tag{2}$$

where a and b are constants derived from regression techniques [7, 37].

The relationship between the length and mass of 15 fish species in the LifeCLEF 2015 dataset is examined to estimate their combined biomass. Consistency in the calculations is maintained with this approach. It is demonstrated in (2) that the length of the fish is needed to estimate its weight. Estimating length is challenging, particularly in a monocular dataset. To address this, a scale was developed to convert the pixel length of a detected fish into real lengths using depth data. A straightforward scale was created see (3), utilising the fish's pixel length and depth (which will be elaborated on later) to determine the fish's length in centimeters (4).

$$\text{Scale} = \frac{\text{Pixel length} \times \text{Distance from camera}}{\text{Actual length}}, \tag{3}$$

$$\text{Actual length} = \frac{\text{Pixel length} \times \text{Distance from camera}}{\text{Scale}}, \tag{4}$$

To construct the scale, the species of fish depicted in a given video are first determined using known factual information. Next, the average length of each fish species is calculated using data from FishBase [10], an online database. The only assumption made is that the fish in the dataset are healthy and represented an average size for their respective species. Using this methodology, the average lengths for each fish species are derived and can be verified by referencing a corresponding table 3.

The fish length is determined by measuring from the tip of its head to the middle of its tail. The proposed pipeline comprising YOLOv8 + FCE model is utilised in this study, demonstrating high precision and recall. Consequently, segmentation is not employed for fish length estimation. Instead, the bounding boxes acquired by the proposed pipeline

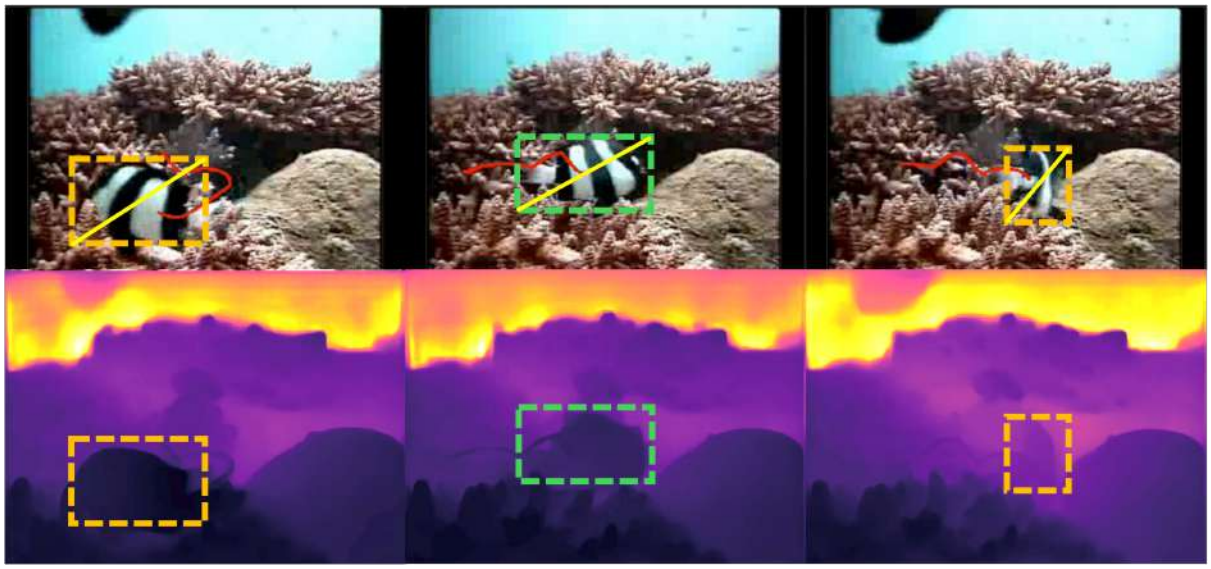

Figure 6: A tracked fish in three consecutive frame with corresponding depth estimates (second row). In the left frame, the fish is nearer to the camera with decent side profile where length of fish is calculated by estimating the depth, bounding box's length and diagonal. In the middle frame, the fish is farther to the camera but exhibits perfect side profile hence, with a good estimate of length. In the right frame, fish takes a turn. The middle frame is kept for estimating fish length while other bounding boxes are discarded in calculation.

are used and their pixel lengths are relied upon to approximate the fish's size from head to tail midpoint. Due to the continuous movement of some fish, accurately measuring their length is challenging, as they align with the camera for only a brief period (1-2 frames) with a complete side profile. To address this issue, a record of fish lengths corresponding to their unique tracker IDs is maintained throughout the entire video. The maximum recorded length is then used to estimate the fish's actual length in centimeters. Additionally, due to uncertainty about the fish's orientation and to prevent including its width when it moves vertically in the video, the mean value of the bounding box's length (horizontal) and its diagonal is computed in a each frame. The bounding box in a particular frame having maximum mean ensures maximum possible availability of fish's side profile. The length of the bounding box is then taken as an estimate of fish's actual length (see Figure 6 first row). Indeed, this needs another parameter to cater i.e., depth.

2.2.5 Depth Estimation from Monocular Videos

Depth estimation is achieved through a modified U-Net architecture [46] that incorporates an encoder-decoder pipeline for image regression. An RGB image is used as input to the U-Net, while the output is the corresponding depth image representing a disparity map (see Figure 6, second row). Within the encoder of the U-Net, a multi-head self-attention mechanism is included in each convolutional block to emphasize the most relevant spatial

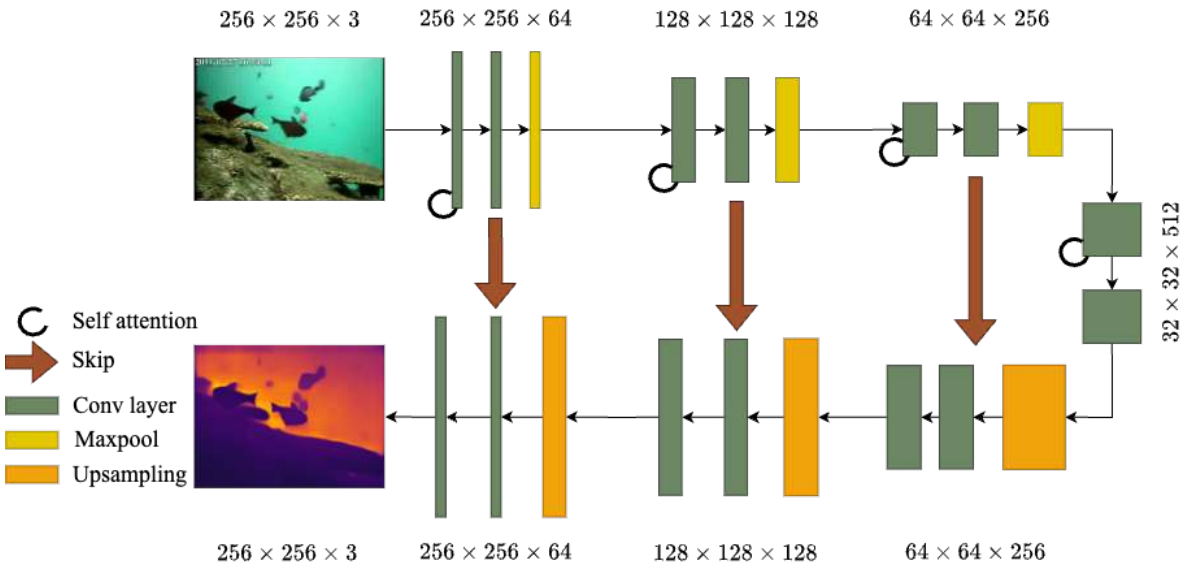

Figure 7: The Proposed U-Net architecture. There are three convolutional blocks in encoder and decoder. The dimension for each block represents feature size after the second convolutional layer of each block.

features of objects in the RGB image, which are translatable to a certain distance from the camera. As depicted in Figure 7, four convolutional blocks, each containing two convolutional layers, are included in the U-Net. The attention mechanism is applied to the features of the first convolutional layer in each block. Skip connections are used to link the encoder-decoder layers, excluding the bottleneck block.

Disparity maps are generated by applying the proposed U-Net architecture to the LifeCLEF 2015 dataset via transfer learning. In general, large datasets are required for CNN-based architectures to converge, and the LifeCLEF 2015 dataset is sparse for optimal training. To mitigate this challenge, two additional datasets, KITTI [12] and NYU [6], are used. These datasets provide RGB images and corresponding depth/disparity maps calculated through a stereo-camera setup. The proposed U-Net is trained using monocular RGB images from these datasets as inputs, with the corresponding disparity maps as target images. This transfer learning setup, utilising more than 1.5 million annotated images from the KITTI and NYU datasets for scratch training, results in superior RMSE performance on LifeCLEF 2015 dataset compared to previous methods.

To determine the depth of a solitary fish, the bounding box provided by YOLOv8 is utilized, and the depth value from the depth array generated by the U-Net model is obtained at the center of this bounding box. It is reasonably assumed that the fish will consistently occupy the center of the bounding box, allowing for the calculation of the fish's distance from the camera.

3 Results

This section presents performance evaluation on fish detection, species classification and their biomass estimation on videos available in LifeCLEF 2015 dataset. For this purpose, F1 score and standard accuracy metrics are used. The F1 score (see (5)) takes into account both precision and recall, providing a balanced assessment of a model's ability to correctly classify instances across multiple classes. In other words, higher F1 scores confirms detector or classifier model's ability to correctly identify relevant instances and while keeping the missed detections and flase alarms low. On the other hand Accuracy (see (6)) depends on true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}. \tag{5}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{6}$$

3.1 Fish Detection and Classification

As mentioned earlier, the fish detection and classification pipeline comprises two branches i.e., FCE branch to which acts as temporal model to detect fast moving fish and YOLOv8 branch to capture static fish instances.

The proposed FCE model is trained for foreground (moving fish) extraction in videos. The training process involves the following parameters. Firstly, the threshold of variance in pixel intensities is fixed at a value of 127, indicating the minimum variance required for a pixel to be considered part of the foreground. Secondly, the background ratio that determines the model's sensitivity to classify pixels as foreground is set to 0.7, indicating that a pixel must have at least a 70% probability of belonging to the background to be classified as such. Additionally, the number of Gaussians in FCE is set to 20, enabling the model to capture complex background distributions effectively. These hyperparameters are crucial for fine-tuning the background subtraction process, balancing sensitivity to changes with computational efficiency, and ensuring accurate extraction of foreground objects from video sequences. Adjusting these parameters allows for optimised performance tailored to specific video data characteristics and application requirements.

For training YOLOv8, already provided labels in LifeCLEF 2015 dataset are utilised. The label file contains information about fish candidate locations and their corresponding species class information, where legitimate fish instances are annotated along with their bounding box coordinates. Images without fish instances are represented by empty vectors. During training, YOLOv8's detector branch focuses on regression tasks for coordinates using mean square error, while the classifier branch is trained on labels using

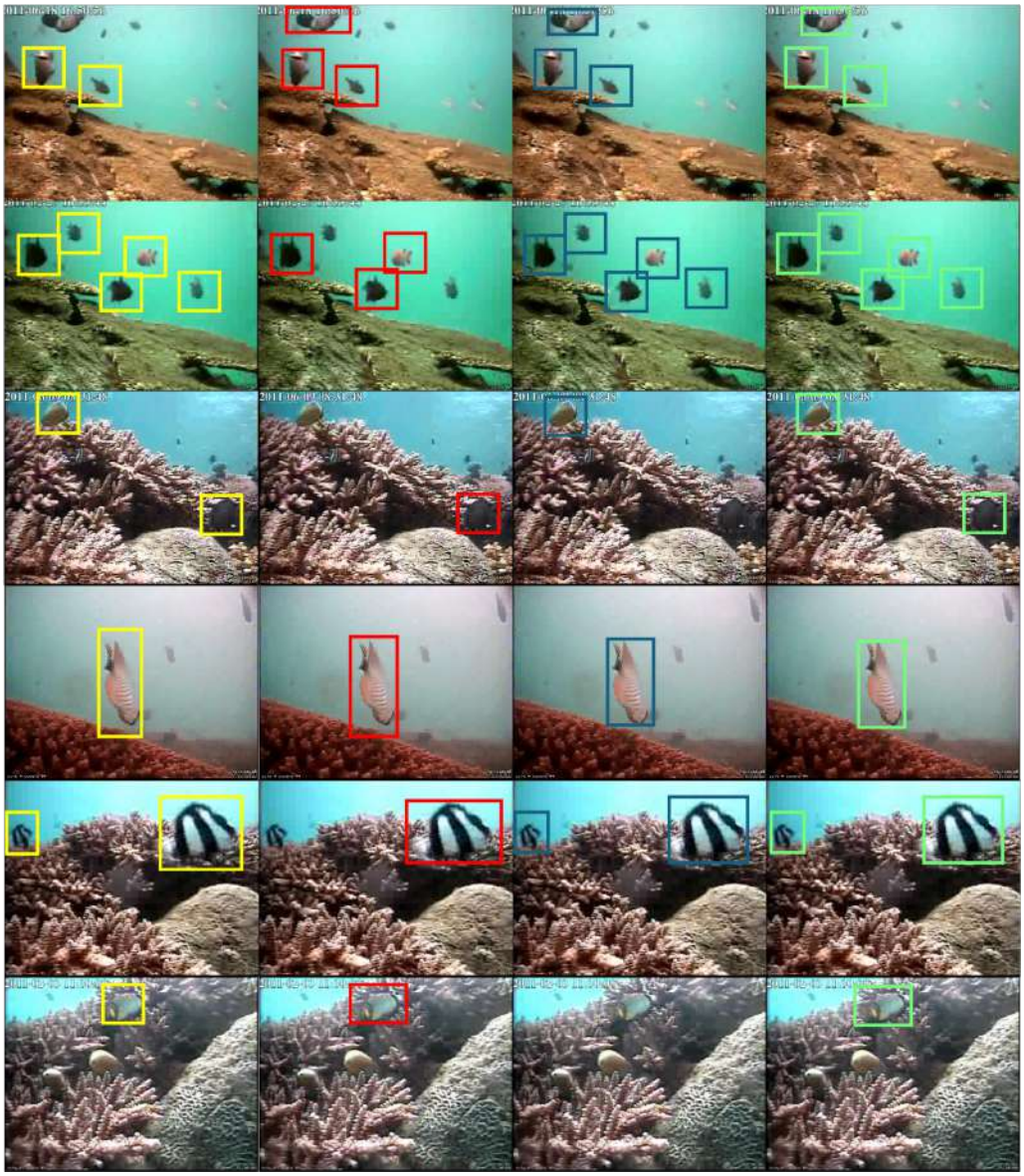

Figure 8: Fish detection performance: Ground truths (first column), FCE (second column), YOLOv8 (third column), and the proposed solution (fourth column).

cross-entropy loss. The training process involves various hyperparameter settings, such as training for 500 epochs, a batch size of 16, an image size of 640×480 pixels, a learning rate of 0.01, an optimizer weight decay of 0.0005, and warm-up momentum set to 0.8. To enhance model generalization, various data augmentation techniques are applied, including hue-saturation-value adjustments (HSV = [0.015, 0.7, 0.4]), translation by 0.1, mosaic augmentation at a factor of 1.0, and horizontal flipping with a probability of 0.5. The detection outcomes are filtered out below a confidence score of 0.6 during evaluation stage. The training process for YOLOv8 took approximately 18 hours, with the larger amount of training data provided by the Detection Set contributing to a more stable training curve and resulting in a significant improvement in the F-score on the test set.

Figure 8 depicts outcome of the fish detection task by the proposed pipeline and its

Table 2: Fish detection and their species classification results. The F1 score is calculated using 5 represents detection metric while Accuracy is calculated using 6 is the species-wise classification scores.

Species	**Precision**	**Recall**	**F1 Score**	**Accuracy**
Abudefduf vaigiensis	0.949	0.833	0.914	0.877
Acanthurus nigrofuscus	0.967	0.983	0.994	0.975
Amphiprion clarkii	0.954	0.985	0.985	0.969
Chaetodon lunulatus	0.988	0.999	0.992	0.993
Chaetodon speculum	0.988	1.000	0.995	0.994
Chaetodon trifascialis	0.989	0.981	0.993	0.985
Chromis chrysura	0.964	0.906	0.946	0.934
Dascyllus aruanus	0.988	0.999	0.992	0.993
Dascyllus reticulatus	0.960	0.945	0.980	0.952
Myripristis kuntee	0.943	0.981	0.992	0.962
Neoglyphidodon nigroris	0.937	0.971	0.989	0.954
Pempheris vanicolensis	0.930	0.938	0.971	0.934
Plectroglyphidodon dickii	0.942	0.920	0.953	0.931
Zebrasoma scopas	0.985	1.000	0.995	0.992
Average	**0.965**	**0.959**	**0.975**	**0.963**

constituent components i.e., FCE and YOLOv8 model. As FCE and YOLOv8 models are responsible for temporal and spatial modeling of the fish instances in videos, their outcomes are combined to get the final output when they detect exclusive bounding boxes. In case of same detection outcome by FCE and YOLOv8, preference will be give to the fish localisation (coordinates) by the later due to better confinement and tightness in encompassing the fish. Table 2 presents the fish detection and species classification scores. The overall fish detection F1 score is 97.5% while accuracy of species classification is 96.3% after combining YOLOv8 and FCE models in the proposed pipeline. The *Hemigymnus melapterus* has no bounding box annotations available in the dataset, so it was not detected by our models. Therefore, it is not included in Table 2 or Table 3.

3.2 Fish Species Biomass

Videos in LifeCLEF 2015 dataset representing various locations in Taiwan coral reef [24] is used to estimate overall fish species biomass. It is worth mentioning that ground truth about the fish size and their weight (or mass) isn't provided with the dataset.

Table 3: Biomass estimation with monocular data. The average RMSE value is **5.69** while the average standard deviation is **3.27** for overall dataset. Length and weight is denoted as L and W respectively while GT is the ground truth range on length.

Species	**Avg.** L **(cm)**	**GT** L **(cm)**	**Avg.** W **(g)**	**Biomass (g)**	**RMSE (cm)**	**SD (cm)**
Abudefduf vaigiensis	4.94	3.4 - 11.5	13.10	2806.29	4.76	2.51
Acanthurus nigrofuscus	23.72	4.5 - 18.7	54.20	22166.04	14.05	12.12
Amphiprion clarkii	4.48	8.5 - 9.5	6.74	5677.83	4.54	4.41
Chaetodon lunulatus	16.54	7.5 - 26.7	2.03	3781.29	9.62	0.56
Chaetodon speculum	7.56	2.0 - 11.5	403.58	55243.35	4.82	0.81
Chaetodon trifascialis	13.96	3.0 - 14.0	16.93	27699.94	7.75	5.46
Chromis chrysura	5.92	5.5 - 12.0	25.69	7416.31	4.31	2.83
Dascyllus aruanus	7.29	2.3 - 9.0	4.99	14215.75	3.73	1.64
Dascyllus reticulatus	6.99	3.5 - 8.0	10.76	74409.92	2.57	1.24
Myripristis kuntee	6.22	6.0 - 14.0	5.18	1758.79	5.50	3.77
Neoglyphidodon nigroris	5.96	6.5 - 10.6	13.10	8984.17	3.29	2.58
Pempheris vanicolensis	6.49	7.7 - 15.5	4.99	3185.57	6.42	5.11
Plectroglyphidodon dickii	5.35	2.8 - 11.6	2.96	2057.84	4.77	1.85
Zebrasoma scopas	5.64	3.1 - 10.1	6.75	1726.23	3.63	0.95

Therefore, biomass estimation using the proposed pipeline that requires fish species detection/classification and tracking is made by employing (2-4). The outcome is compared by taking into account average mass of each species as catalogued in the benchmark FishBase. Table 3 lists the species-wise average length, weight (mass) and overall biomass in the LifeCLEF 2015 dataset. The root mean square error (RMSE) using (7) and standard deviation using (8) are mentioned when average estimated length of a fish species is compared with its corresponding minimum and maximum values ($x_{\min}$ and $x_{\max}$) in the length range at FishBase taken as ground truth. The average RMSE and average SD of the entire dataset is 5.69 and 3.27 respectively.

$$\text{RMSE} = \sqrt{\frac{(y - x_{\min})^2 + (y - x_{\max})^2}{2}}. \tag{7}$$

$$\text{SD} = \left| y - (x_{\min} + x_{\max})/2 \right|. \tag{8}$$

For our experimentation, we use Intel®Core i5$^{\text{TM}}$ processor with 32GB RAM and

Table 4: Comparison on LifeCLEF 2015 fish detection task

Approach	**F1 Score**
GoogleNet [8]	0.840
FishNet [71]	0.804
Pixel-level Image Enhancement + Cascaded-RCNN [55]	0.817
AlexNet [20]	0.740
GMM + Optical Flow [52]	0.843
GMM + Optical flow + YOLOv3 [21]	0.954
Ours (YOLOv8 + FCE)	**0.975**

Nvidia®1080Ti 11GB GPU for training and other calculations.

3.3 Discussion

In recent times, many computer vision and machine learning algorithms have surfaced, displaying potential across various fish-related tasks such as automatic fish detection, species classification, tracking, and biomass estimation. These innovative techniques predominantly lean on sophisticated deep learning algorithms ([25]; [29]. However, a notable research gap remains in understanding the influence of environmental factors on the performance of these systems. While state-of-the-art object detection algorithms excel at pinpointing static fish based on textural patterns, they often falter in detecting moving fish, particularly in challenging underwater settings with poor visibility [21].

This gap has been addressed in this study by achieving state-of-the-art performance in the fish detection task. A novel algorithm for temporal or motion-based feature extraction is integrated seamlessly with spatial features through the YOLOv8 deep architecture in the hybrid machine vision pipeline (illustrated in Figure 3). The temporal features crucial for fish detection are derived through simultaneous motion segmentation and background subtraction, allowing the filtering out of irrelevant background pixels with lower motion magnitudes while retaining potential fish candidates.

The YOLOv8, in its standard configuration, struggles to accurately detect fast-moving fish instances, particularly when lacking clear texture or shape due to factors like water murkiness or camouflage in the background. As an example, see Figure 8 third and last row of third column where fish textural features are totally merged with the background therefore, YOLOv8 fails to detect the fish instances. The temporal branch (second column) effectively addresses this limitation, significantly boosting the overall F-score of and paving the way for potential applications such as assemblage and biomass estimation.

In our empirical evaluation aimed at identifying the most suitable deep architec-

ture, we trained several popular backbone networks including ResNet-18, ResNet-50 [15], ResNeXt [66], and MobileNetV3 [17]. Despite exploring this wide range of architectures, we found that these models were consistently outperformed by the YOLOv8 detector. YOLOv8 not only offers faster inference times but also excels in detection accuracy compared to Single-Shot Detector (SSD)[36] and Region Proposal Network (RPN)[45] models. The initial layers capture high-level features like fish edges and overall structural patterns, while middle and top layers meticulously focus on features unique to the fish shape, including the snout, tail, and fins. The first and second rows of Figure 8 show the standard and zoomed-in images, respectively, while the third row depicts the fish-less background where the network's activations are random without any informative pattern. These static or very slowly moving fish are well detected by the YOLOv8 system. Conversely, relatively fast-moving fish instances with unclear textural patterns due to motion blurriness in the images are successfully captured by our proposed motion-based FCE algorithm while suppressing underwater dynamic backgrounds. Table 4 compares some latest work on LifeCLEF 2015 dataset employing various architectures and hence advocating the proposed scheme of unconstrained underwater fish detection.

Given the dataset's complexity and image quality, as depicted in Figures 1, detecting and classifying fish faces numerous challenges. Various sources of variation, particularly dynamic backgrounds and unclear fish textures, exacerbate the nonlinearity of the data [32]. Conventional machine learning and computer vision algorithms, which tend to be shallow and linear, often struggle to effectively address this issue. Therefore, our ideal approach involves highly nonlinear deep neural networks augmented with temporal information to achieve desired results. The novelty of our approach primarily revolves around the specially designed architecture dedicated to fish detection which will be a prerequisite to biomass estimation, directly impacting the ecosystem's overall health in the specific sampling region.

The choice of LifeCLEF 2015 dataset and proposed scheme in fish detection and classification task can be justified by studying similar recent approaches. For example, in [22], the FishInTurbidWater dataset focuses on turbid water conditions, lacking challenges like deep underwater environments with coral reefs, moving aquatic plants, algae, and illumination variation. These factors increase false alarms and reduce accuracy in fish detection. In contrast, the LifeCLEF 2015 dataset offers a broader range of unconstrained underwater scenarios. Our approach, combining a novel temporal detector with a YOLOv8 spatial feature extractor, has been tested on the LifeCLEF 2015 dataset, which differs visually from the FishInTurbidWater dataset. In contrast, [55] enhanced the LifeCLEF 2015 dataset for better fish visualization and used cascade RCNN for detection, achieving an F-score of 81%. However, image enhancement alone introduces background noise, reducing accuracy and overall F1 score. Our hybrid approach, by contrast, effectively identifies fast-moving fish, suppresses background noise, and eliminates outliers.

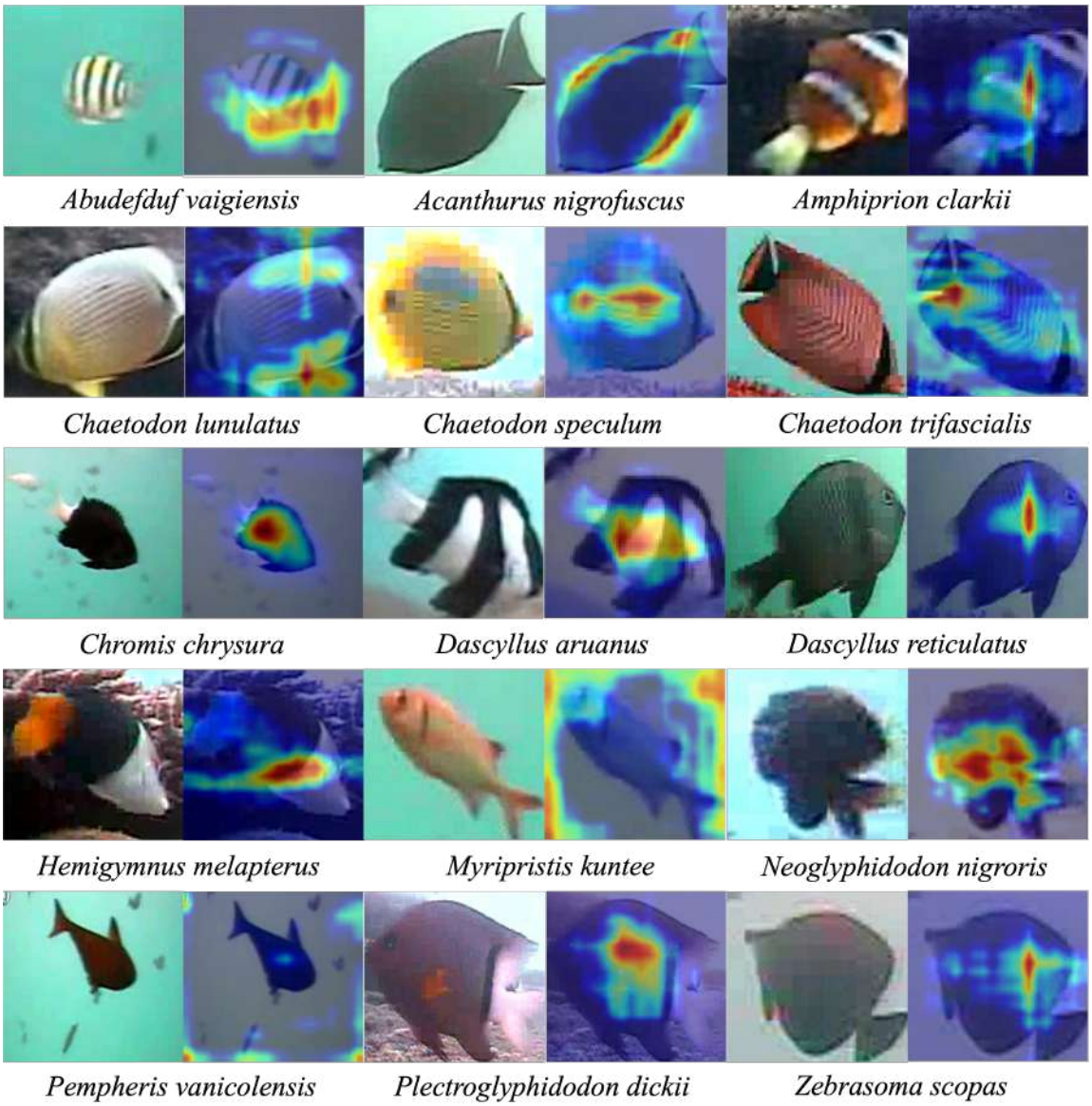

Figure 9: Visualization of features learned by the proposed scheme for various fish species in LifeCLEF 2015 dataset.

Deep learning-based fish segmentation techniques are increasingly used for underwater videos. For instance, [11] utilises Mask R-CNN, and [2] employed ResNet, YOLOACT, and Cascade R-CNN models. These studies, often using controlled environment datasets, do not address challenges like illumination and contrast variations, blurriness, turbidity, camouflage, occlusion, jitter, and moving background objects present in the LifeCLEF 2015 dataset, leading to false alarms and missed classifications. In another work [30] A-LCFCN (Affinity-based Fully Convolutional Neural Network) model with two branches is introduced, evaluated on the DeepFish dataset. Despite an impressive IoU (intersection over union) of 0.99 for background objects, the low IoU of 0.73 for foreground objects indicates excessive false alarms and missed classifications.

A robust fish detection and classification system is key to a successful biomass esti-

mation. In other words, fish instances should be accurately identified for their species classification. The proposed pipeline of YOLOv8 and FCE yields favourable outcome due to its ability to capture species-specific information from images. Figure 9 depicts images of fish species in LifeCLEF 2015 dataset with their corresponding feature maps. These maps, in general, show the parts of fish where the proposed pipeline put emphasis for better recognition. Fish species with clear and exclusive texture and colour pattern are focused more on main bodies e.g., *Abudefduf vaigiensis* with black and white stripes, *Amphiprion clarkii* with orange and white bands, and *Dascyllus aruanus* with black and white bands. Fish species where texture is not clear the attention is made on shape features including tail, gills and fins. The examples are *Acanthurus nigrofuscus*, *Myripristis kuntee*, *Neoglyphidodon nigroris*, *Dascyllus reticulatus* and *Zebrasoma scopas*. In addition, fish species with similar shape and texture, features on main body as well as tail, gills and fins are equally emphasised as in the case of *Chaetodon lunulatus*, *Chaetodon speculum*, and *Chromis chrysura*.

For biomass estimation, previous studies often use stereo fish datasets for calculating fish parameters like length and area [59, 60, 35]. However, these methods face limitations with monocular datasets like LifeCLEF 2015, making fish length estimation from monocular images challenging and necessitating innovative solutions incorporating depth analytics. Moreover, the fact that most of the underwater data that is being collected around the globe is monocular due to cheaper hardware and handling, necessitates the availability of computer vision and machine learning algorithms to fill this research gap. Nevertheless, the use of generic stereo vision data to train the deep regression model for generating RGB to depth image is inevitable. In this study, a special U-Net architecture with attention mechanism in the encoder is proposed. The netowrk is trained on very large KITTI and NYU stereo image datasets to learn to extract depth map from RGB images. The attention mechanism introduced in the encoder helps in the utilization of the proposed U-Net for underwater imagery with visual challenges. Expanding upon this paradigm shift, we have leveraged the prevalent power model relationship between fish length and weight, a staple in the biomass estimation domain [43, 59, 16]. Prior to the introduction of our novel approach, there existed no established method for estimating biomass within monocular datasets like LifeCLEF 2015. Consequently, the introduction of this new method marks a significant contribution to the field, representing a novel and pioneering advancement in fish biomass estimation techniques. It is evident in Table 3 that the proposed approach successfully estimates the average length of fish species as compared to ground truth ranges. This is a beneficial achievement especially when exact measurement of each fish instance is not available as ground truth. Indeed, in future, a comprehensive dataset with labelled fish instances, species and depth values will be useful is fine-tuning on accurate length estimation of fish and consequently biomass.

4 Conclusion

In this study, a Convolutional Neural Network (CNN) known as YOLOv8 is utilised for the fish detection and species classification task. YOLOv8, known for its high accuracy and speed, is chosen as the primary model however, its limitations to miss fast moving fish is mitigated by the introduction of FCE probabilistic modelling to suppress background and capture moving foreground (fish). Furthermore, a specialised regression U-Net model is presented to convert RGM underwater images to epth maps for object distance calculation from monocular camera. This calculation together with fish tracking are used to estimate fish species biomass in the LifeCLEF 2015 dataset. In future, we aim to employ convolutional neural networks with temporal layers and vision transformers on larger underwater datasets to improve the biomass estimation.

References

[1] Al-Jubouri, Q., Al-Nuaimy, W., Al-Taee, M. and Young, I., 2017. An automated vision system for measurement of zebrafish length using low-cost orthogonal web cameras. Aquacultural Engineering, 78, pp.155-162.

[2] Alshdaifat, N.F.F., Talib, A.Z. and Osman, M.A., 2020. Improved deep learning framework for fish segmentation in underwater videos. Ecological Informatics, 59, p.101-121.

[3] Ashley, P.J., 2007. Fish welfare: current issues in aquaculture. Applied animal behaviour science, 104(3-4), pp.199-235.

[4] Balaban, M.O., Ünal Şengör, G.F., Soriano, M.G. and Ruiz, E.G., 2010. Using image analysis to predict the weight of Alaskan salmon of different species. Journal of food science, 75(3), pp.E157-E162.

[5] Bewley, A., Ge, Z., Ott, L., Ramos, F. and Upcroft, B., 2016, September. Simple online and realtime tracking. In 2016 IEEE international conference on image processing (ICIP) (pp. 3464-3468). IEEE.

[6] Couprie, C., Farabet, C., Najman, L. and LeCun, Y., 2013. Indoor semantic segmentation using depth information. arXiv preprint arXiv:1301.3572.

[7] Copernicus, 2013 ¡https://bg.copernicus.org/articles/10/529/2013/bg-10-529-2013-supplement.pdf¿

[8] Choi, S., 2015, September. Fish Identification in Underwater Video with Deep Convolutional Neural Network: SNUMedinfo at LifeCLEF Fish task 2015. In CLEF (Working Notes) (pp. 1-10).

[9] Ditria, E.M., Jinks, E.L. and Connolly, R.M., 2021. Automating the analysis of fish grazing behaviour from videos using image classification and optical flow. Animal Behaviour, 177, pp.31-37.

[10] Froese, R. and D. Pauly. Editors. 2024.FishBase. World Wide Web electronic publication. www.fishbase.org, (02/2024).

[11] Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., Vågstøl, H. and Løvall, K., 2020. Automatic segmentation of fish using deep learning with application to fish size measurement. ICES Journal of Marine Science, 77(4), pp.1354-1366.

[12] Geiger, A., Lenz, P., Stiller, C. and Urtasun, R., 2013. Vision meets robotics: The kitti dataset. The International Journal of Robotics Research, 32(11), pp.1231-1237.

[13] Giorli, G., Drazen, J.C., Neuheimer, A.B., Copeland, A. and Au, W.W., 2018. Deep sea animal density and size estimated using a Dual-frequency IDentification SONar (DIDSON) offshore the island of Hawaii. Progress in Oceanography, 160, pp.155-166.

[14] Harvey, E., Cappo, M., Shortis, M., Robson, S., Buchanan, J. and Speare, P., 2003. The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (Thunnus maccoyii) with a stereo–video camera system. Fisheries Research, 63(3), pp.315-326.

[15] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[16] Hong, H., Yang, X., You, Z. and Cheng, F., 2014. Visual quality detection of aquatic products using machine vision. Aquacultural Engineering, 63, pp.62-71.

[17] Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V. and Le, Q.V., 2019. Searching for mobilenetv3. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1314-1324).

[18] Huang, P.X., Boom, B.J. and Fisher, R.B., 2015. Hierarchical classification with reject option for live fish recognition. Machine Vision and Applications, 26(1), pp.89-102.

[19] Huang, X. and Huang, P.X., 2014. Balance-Guaranteed Optimized Tree with Reject option for live fish recognition.

[20] Jäger, J., Rodner, E., Denzler, J., Wolff, V. and Fricke-Neuderth, K., 2016, September. SeaCLEF 2016: Object Proposal Classification for Fish Detection in Underwater Videos. In CLEF (working notes) (pp. 481-489).

[21] Jalal, A., Salman, A., Mian, A., Shortis, M. and Shafait, F., 2020. Fish detection and species classification in underwater environments using deep learning with temporal information. Ecological Informatics, 57, p.101088.

[22] Jahanbakht, M., Azghadi, M.R. and Waltham, N.J., 2023. Semi-supervised and weakly-supervised deep neural networks and dataset for fish detection in turbid underwater videos. Ecological Informatics, 78, p.102303.

[23] Jeong, S.J., Yang, Y.S., Lee, K., Kang, J.G. and Lee, D.G., 2013. Vision-based automatic system for non-contact measurement of morphometric characteristics of flatfish. Journal of Electrical Engineering and Technology, 8(5), pp.1194-1201.

[24] Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.P., Planqué, R., Rauber, A., Palazzo, S., Fisher, B. and Müller, H., 2015. LifeCLEF 2015: multimedia life species identification challenges. In Experimental IR Meets Multilinguality, Multimodality, and Interaction: 6th International Conference of the CLEF Association, CLEF'15, Toulouse, France, September 8-11, 2015, Proceedings 6 (pp. 462-483). Springer International Publishing.

[25] Kandimalla, V., Richard, M., Smith, F., Quirion, J., Torgo, L. and Whidden, C., 2022. Automated detection, classification and counting of fish in fish passages with deep learning. Frontiers in Marine Science, 8, p.823173.

[26] Keitt, T.H. and Abelson, E.S., 2021. Ecology in the age of automation. Science, 373(6557), pp.858-859.

[27] Kindsvater, H.K., Dulvy, N.K., Horswill, C., Juan-Jordá, M.J., Mangel, M. and Matthiopoulos, J., 2018. Overcoming the data crisis in biodiversity conservation. Trends in Ecology & Evolution, 33(9), pp.676-688.

[28] Klontz, G.W. and Kaiser, H., 1993. Producing a marketable fish. Focus on renewable natural resources (USA), 18.

[29] Knausgård, K.M., Wiklund, A., Sørdalen, T.K., Halvorsen, K.T., Kleiven, A.R., Jiao, L. and Goodwin, M., 2022. Temperate fish detection and classification: a deep learning based approach. Applied Intelligence, 52(6), pp.6988-7001.

[30] Laradji, I.H., Saleh, A., Rodriguez, P., Nowrouzezahrai, D., Azghadi, M.R. and Vazquez, D., 2021. Weakly supervised underwater fish segmentation using affinity LCFCN. Scientific reports, 11(1), pp.1-10.

[31] Lawson, G.L., Barange, M. and Fréon, P., 2001. Species identification of pelagic fish schools on the South African continental shelf using acoustic descriptors and ancillary information. ICES Journal of Marine Science, 58(1), pp.275-287.

[32] LeCun, Y., Huang, F.J. and Bottou, L., 2004, June. Learning methods for generic object recognition with invariance to pose and lighting. In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. (Vol. 2, pp. II-104). IEEE.

[33] Letessier, T.B., Proud, R., Meeuwig, J.J., Cox, M.J., Hosegood, P.J. and Brierley, A.S., 2022. Estimating pelagic fish biomass in a tropical seascape using echosounding and baited stereo-videography. Ecosystems, pp.1-18.

[34] Li, D., Hao, Y. and Duan, Y., 2020. Nonintrusive methods for biomass estimation in aquaculture with emphasis on fish: a review. Reviews in Aquaculture, 12(3), pp.1390-1411.

[35] Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017. Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125).

[36] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. and Berg, A.C., 2016. Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14 (pp. 21-37). Springer International Publishing.

[37] Mehanna, S.F. and Farouk, A.E., 2021. Length-weight relationship of 60 fish species from the Eastern Mediterranean Sea, Egypt (GFCM-GSA 26). Frontiers in Marine Science, 8, p.625422.

[38] Miranda, J.M. and Romero, M., 2017. A prototype to measure rainbow trout's length using image processing. Aquacultural engineering, 76, pp.41-49.

[39] Moon, T.K., 1996. The expectation-maximization algorithm. IEEE Signal processing magazine, 13(6), pp.47-60.

[40] Naval, P.C. and David, L.T., 2016, October. FishDrop: Estimation of reef fish population density and biomass using stereo cameras. In 2016 Techno-Ocean (Techno-Ocean) (pp. 527-531). IEEE.

[41] Palazzo, S. and Murabito, F., 2014, November. Fish species identification in real-life underwater images. In Proceedings of the 3rd ACM international workshop on multimedia analysis for ecological data (pp. 13-18).

[42] Pérez, D., Ferrero, F.J., Alvarez, I., Valledor, M. and Campo, J.C., 2018, May. Automatic measurement of fish size using stereo vision. In 2018 IEEE international instrumentation and measurement technology conference (I2MTC) (pp. 1-6). IEEE.

[43] Petrell, R.J., Shi, X., Ward, R.K., Naiberg, A. and Savage, C.R., 1997. Determining fish size and swimming speed in cages and tanks using simple video techniques. Aquacultural Engineering, 16(1-2), pp.63-84.

[44] Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[45] Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.

[46] Ronneberger, O., Fischer, P. and Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18 (pp. 234-241). Springer International Publishing.

[47] Rosen, S., Jörgensen, T., Hammersland-White, D. and Holst, J.C., 2013. DeepVision: a stereo camera system provides highly accurate counts and lengths of fish passing inside a trawl. Canadian Journal of Fisheries and Aquatic Sciences, 70(10), pp.1456-1467.

[48] Saberioon, M., Gholizadeh, A., Cisar, P., Pautsina, A. and Urban, J., 2017. Application of machine vision systems in aquaculture with emphasis on fish: state-of-the-art and key issues. Reviews in Aquaculture, 9(4), pp.369-387.

[49] Saleh, A., Sheaves, M., Jerry, D., Azghadi, M.R., 2022. Unsupervised fish trajectory tracking and segmentation. arXiv preprint arXiv:2208.10662 .

[50] Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J. and Harvey, E., 2016. Fish species classification in unconstrained underwater environments based on deep learning. Limnology and Oceanography: Methods, 14(9), pp.570-585.

[51] Salman, A., Maqbool, S., Khan, A.H., Jalal, A. and Shafait, F., 2019. Real-time fish detection in complex backgrounds using probabilistic background modelling. Ecological Informatics, 51, pp.44-51.

[52] Salman, A., Siddiqui, S.A., Shafait, F., Mian, A., Shortis, M.R., Khurshid, K., Ulges, A. and Schwanecke, U., 2020. Automatic fish detection in underwater videos by a

deep neural network-based hybrid motion learning system. ICES Journal of Marine Science, 77(4), pp.1295-1307.

[53] Savian, S., Elahi, M., Tillo, T., 2020. Optical flow estimation with deep learning, a survey on recent advances. Deep biometrics , 257–287.

[54] Sohan, M., Sai Ram, T., Reddy, R. and Venkata, C., 2024. A review on yolov8 and its advancements. In International Conference on Data Intelligence and Cognitive Informatics (pp. 529-545). Springer, Singapore.

[55] Sun, H., Yue, J. and Li, H., 2022. An image enhancement approach for coral reef fish detection in underwater videos. Ecological Informatics, 72, p.101862.

[56] Sung, M., Yu, S.C. and Girdhar, Y., 2017, June. Vision based real-time fish detection using convolutional neural network. In OCEANS 2017-Aberdeen (pp. 1-6). IEEE.

[57] Takahara, T., Minamoto, T., Yamanaka, H., Doi, H. and Kawabata, Z.I., 2012. Estimation of fish biomass using environmental DNA. PloS one, 7(4), p.e35868.

[58] Terven, J., Córdova-Esparza, D.M. and Romero-González, J.A., 2023. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. Machine Learning and Knowledge Extraction, 5(4), pp.1680-1716.

[59] Tillett, R., McFarlane, N. and Lines, J., 2000. Estimating dimensions of free-swimming fish using 3D point distribution models. Computer Vision and Image Understanding, 79(1), pp.123-141.

[60] Torisawa, S., Kadota, M., Komeyama, K., Suzuki, K. and Takagi, T., 2011. A digital stereo-video camera system for three-dimensional monitoring of free-swimming Pacific bluefin tuna, Thunnus orientalis, cultured in a net cage. Aquatic Living Resources, 24(2), pp.107-112.

[61] TW, F., 1904. The rate of growth of fishes. Twenty-second annual report, pp.141-241.

[62] Ubina, N., Cheng, S.C., Chang, C.C. and Chen, H.Y., 2021. Evaluating fish feeding intensity in aquaculture with convolutional neural networks. Aquacultural Engineering, 94, p.102178.

[63] Viazzi, S., Van Hoestenberghe, S., Goddeeris, B.M. and Berckmans, D., 2015. Automatic mass estimation of Jade perch Scortum barcoo by computer vision. Aquacultural engineering, 64, pp.42-48.

[64] Wang, G., Muhammad, A., Liu, C., Du, L. and Li, D., 2021. Automatic recognition of fish behavior with a fusion of RGB and optical flow data based on deep learning. Animals, 11(10), p.2774.

[65] Wilson, S.K., Graham, N.A.J., Holmes, T.H., MacNeil, M.A. and Ryan, N.M., 2018. Visual versus video methods for estimating reef fish biomass. Ecological Indicators, 85, pp.146-152.

[66] Xie, S., Girshick, R., Dollár, P., Tu, Z. and He, K., 2017. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1492-1500).

[67] Xu, W. and Matzner, S., 2018, December. Underwater fish detection using deep learning for water power applications. In 2018 International conference on computational science and computational intelligence (CSCI) (pp. 313-318). IEEE.

[68] Zhuang, P., Xing, L., Liu, Y., Guo, S. and Qiao, Y., 2017, September. Marine Animal Detection and Recognition with Advanced Deep Learning Models. In CLEF (Working Notes) (pp. 166-177).

[69] Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W. and Wang, X., 2022, October. Bytetrack: Multi-object tracking by associating every detection box. In European conference on computer vision (pp. 1-21). Cham: Springer Nature Switzerland.

[70] Zhang, L., Wang, J. and Duan, Q., 2020. Estimation for fish mass using image analysis and neural network. Computers and Electronics in Agriculture, 173, p.105439.

[71] Zhao, Z., Liu, Y., Sun, X., Liu, J., Yang, X. and Zhou, C., 2021. Composited FishNet: Fish detection and species recognition from low-quality underwater videos. IEEE Transactions on Image Processing, 30, pp.4719-4734.

[72] Zhang, T., Yang, Y., Liu, Y., Liu, C., Zhao, R., Li, D. and Shi, C., 2024. Fully automatic system for fish biomass estimation based on deep neural network. Ecological Informatics, 79, p.102399.

[73] Zion, B., 2012. The use of computer vision technologies in aquaculture–a review. Computers and electronics in agriculture, 88, pp.125-132.