Aquat Ecol
https://doi.org/10.1007/s10452-023-10044-8

MFLD-net: a lightweight deep learning network for fish morphometry using landmark detection

Alzayat Saleh · David Jones · Dean Jerry · Mostafa Rahimi Azghadi

Received: 1 December 2022 / Accepted: 26 June 2023

Abstract Monitoring the morphological traits of farmed fish is pivotal in understanding growth, estimating yield, artificial breeding, and population-based investigations. Currently, morphology measurements mostly happen manually and sometimes in conjunction with individual fish imaging, which is a time-consuming and expensive procedure. In addition, extracting useful information such as fish yield and detecting small variations due to growth or deformities, require extra offline processing of the manually collected images and data. Deep learning (DL) and specifically convolutional neural networks (CNNs) have previously demonstrated great promise in estimating fish features such as weight and length from images. However, their use for extracting fish morphological traits through detecting fish keypoints (landmarks) has not been fully explored. In this paper, we developed a novel DL architecture that we call Mobile Fish Landmark Detection network (MFLD-net). We show that MFLD-net can achieve keypoint detection accuracies on par or even better than some of the state-of-the-art CNNs on a fish image dataset. MFLD-net uses convolution operations based on Vision Transformers (i.e. patch embeddings, multi-layer perceptrons). We show that MFLD-net can achieve competitive or better results in low data regimes while being lightweight and therefore suitable for embedded and mobile devices. We also provide quantitative and qualitative results that demonstrate its generalisation capabilities. These features make MFLD-net suitable for future deployment in fish farms and fish harvesting plants.

Keywords Fish morphology · Automated phenotyping · Computer vision · Convolutional neural networks image and video processing · Machine learning · Deep learning

Communicated by Télesphore Sime-Ngando.

A. Saleh · D. Jones · D. Jerry · M. R. Azghadi (✉)
College of Science and Engineering, James Cook University, Townsville, QLD, Australia
e-mail: mostafa.rahimiazghadi@jcu.edu.au

D. Jones · D. Jerry · M. R. Azghadi
ARC Research Hub for Supercharging Tropical Aquaculture through Genetic Solutions, James Cook University, Townsville, QLD, Australia

Introduction

Morphology is an important metric in the production of farmed fish because it can be used to determine the weight and overall size of a fish. These variables are key to animal health and welfare and are used for phenotype analyses in advanced breeding programmes. In aquaculture, determining the morphology is a frequent task crucial for selecting fish for culture as well as developing and testing novel fish strains. Furthermore, fish morphological traits are valuable resources for artificial breeding (Castrillo et al. 2021), functional gene mapping (Figueroa et al.

Published online: 08 July 2023

2018), and population-based investigations (Powers et al. 2020). Morphology helps identify when fish are mature enough to produce eggs or sperm. When determining the growth and maturity of fish, a number of specific morphological traits may be evaluated including the distance from the tip of the mouth to the posterior midpoint of the caudal fin, or the depth of the body from the posterior base of the dorsal fin to anterior of the anal fin (Jerry and Cairns 1998). However, traditional manual fish morphology measurement methods are inefficient and time-consuming. A typical fish measuring process includes measuring the fish's weight using a digital scale, measuring its body lengths with a ruler and then recording these values. Not only is this process inefficient and labour-intensive, but it is also prone to human error.

An automatic tool can help aquaculturists and animal health and welfare authorities to save time and reduce costs by quickly characterising fish morphology and predicting their overall quality in a fast, accurate, and cost-effective manner. Furthermore, the tool would improve the quality of information available to fish farmers and may unlock niche information, because the system can be used at scale. A promising technique to automate this measurement process is computer vision used along with machine learning to capture fish images and automatically extract fish morphology.

A few previous works (Sanchez-Torres et al. 2018; Mathiassen et al. 2011; Islamadina et al. 2018) have used computer vision and traditional image processing techniques to segment (Sanchez-Torres et al. 2018; Islamadina et al. 2018) or make a 3D model (Mathiassen et al. 2011) of the fish body to then use classical machine learning methods, e.g. regression (Sanchez-Torres et al. 2018; Mathiassen et al. 2011) for weight and/or length extraction. Although these studies have achieved significant results, they have involved a complex image processing and feature engineering process to suit their experimental conditions. In contrast, more recent research has been motivated by the outstanding performance of deep learning (DL)-based convolutional neural networks (CNNs) in processing images, without the need for complex image processing and/or feature engineering steps (Konovalov et al. 2018, 2019; Fernandes et al. 2020). In (Konovalov et al. 2018, 2019), the authors have used a CNNs to predict fish body weight by feeding fish images to a segmentation CNNs to extract the fish body area. These studies have utilised both the entire fish body (*i.e.* fish outline) and excluded fins and tails for weight estimation through mass–area estimation models. In (Fernandes et al. 2020), the authors have also explored the use of CNN for estimating the weight and length of fish but they have utilised different CNN architectures. Specifically, the authors have implemented a SegNet-like CNN architecture with an image resolution of 512×512 to estimate the correlation between body measurements and body weight, carcass weight, and carcass yield.

Additionally, researchers have used CNNs for predicting morphological characteristics such as overall length and body size by detecting keypoints on the fish body (Suo et al. 2020; Tseng et al. 2020), similar to the proposed method in this research. However, (Tseng et al. 2020) have proposed a CNN classifier to detect only two keypoints, the fish head, and tail fork regions, to measure the fish body length. On the other hand, (Suo et al. 2020) have used two neural networks, i.e. a faster R-CNN (Ren et al. 2017) to first detect the fish in the image and then a Stacked Hourglass (Newell et al. 2016) to detect specific keypoints on the initially-detected fish, which makes the proposed method complex and expensive. In a more recent study, (Li et al. 2022) proposed a CNN for marine animal segmentation with good results on a self-curated dataset. However, the 207.5 million trainable parameters of their network make it unsuitable for usage in embedded systems or on mobile computing devices for easy deployment in fish farms. To the best of our knowledge, previously published studies that use deep learning to predict fish's morphological traits are complex and large. This renders them unsuitable for use within embedded and mobile devices for commercial use at scale and for easy integration into fish farms. This is because these devices, which are usually designed for resource-constrained environments, have limited computational and power budgets making them incapable of running large networks such as the one proposed in (Li et al. 2022). To address the lack of a lightweight but efficient fish morphological measurement tool, we develop a new deep learning (DL) model for fish body landmark detection using CNNs.

CNNs have dominated the design of DL systems used for computer vision tasks for many years. However, architectures based on emerging transformer models, such as Vision Transformer (ViT)

(Dosovitskiy et al. 2020), are shown to outperform standard convolutional networks in many of these tasks, especially when large training datasets are available. These recent advances motivated us to explore transformer-based architectures for developing lightweight but efficient Fish Landmark Detection networks for automatic fish morphometric analyses.

Vaswani *et al.* first (Vaswani et al. 2017) suggested transformers for machine translation, and they have subsequently become the standard solution for many Natural Language Processing (NLP) applications. Since then, there have been several attempts to incorporate convolutional network characteristics into transformers, making the Vision Transformers (ViT). Recently, the use of patch embeddings for the first layer of the network has spawned a new paradigm of "isotropic" designs, i.e. those having identical sizes and shapes across the network. These models resemble repeating transformer's encoder blocks, but instead of self-attention and multi-layer perceptrons (MLP) operations, alternative operations are used. For example, Bello *et al.* (Bello et al. 2019) introduced a two-dimensional relative self-attention mechanism replacing convolutions as a stand-alone computational primitive for image classification. ResMLP (Touvron et al. 2021) built upon multi-layer perceptrons for image classification by a simple residual network that alternates between a linear layer and a two-layer feed-forward network.

Because of its capacity to capture long-distance interactions, self-attention has been widely adopted as a computational module for modelling sequences (Bahdanau et al. 2015). For example, Ramachandran *et al.* (Ramachandran et al. 2019) replaced all instances of spatial convolutions with a form of self-attention applied to a CNN model to produce a fully self-attentional model that outperforms the baseline on ImageNet classification.

Inspired by the strong performance of Vision Transformers, we investigated utilising some of ViT's architectures using convolution operations. Specifically, we studied the use of patch embeddings (Dosovitskiy et al. 2020), multi-layer perceptrons (MLP-Mixer) (Tolstikhin et al. 2021), and isometric architectures (Sandler et al. 2019). In order to apply a transformer to greater image sizes, patch embeddings aggregate together small areas of the image into single input features. Then, MLP-Mixer works directly with the patches as input, separating the mixing of spatial and channel dimensions, while keeping the network's size and resolution constant (i.e isometric). In this work, we utilise these techniques to modify a standard CNN's architecture to a simple model that is similar in spirit to the ViT using convolutions operations, but does not need a pre-trained model and can generalise well when trained on a small dataset.

Our proposed network, which we named MFLD-net is implemented to estimate landmarks (keypoints) on the fish body to better understand and estimate its morphology. MFLD-net can assist ecologists and fisheries managers with the fast, efficient, accurate, and non-invasive prediction of the size and other morphological aspects of the fish. This provides them with the capacity to make informed management decisions. To evaluate our model, we use an image dataset of Barramundi (*Lates calcarifer*), also known as Asian seabass. We also compare our results to several baseline models to show the performance of MFLD-net.

In summary, the contributions of this work are as follows:

(1) We propose a simple CNN network that estimates the position of known keypoints in a fixed-size fish image.
(2) Due to our architectural innovations, our proposed model is fast and compact, while requiring small training data. These make our system suitable for deployment in aquaculture farms.
(3) We compare our results with several baselines, including U-net (Ronneberger et al. 2015), ResNet-18 (He et al. 2015), ShuffleNet-v2 (Zhang et al. 2018), MobileNet-v2 (Sandler et al. 2018), and SqueezeNet (Iandola et al. 2016).
(4) We provide an evaluation of our model on 60% of our fish image dataset to quantify its generalisation and robustness.

The rest of the paper is organised as follows. Sect. 2 presents our method for training and evaluating our model. Our model's framework is described in detail in Sect. 2-A. The experimental setup and results are presented in Sect. 3, while detailed discussions of our results are presented in Sect. 5. Finally, Sect. 6 concludes our paper.

Materials and methods

We ran three main experiments to test and optimise our proposed model. First, we trained our network (MFLD-net) on only 40% of our dataset. Next, we tested its predictive performance on the dataset test subset described below. Finally, we compared our MFLD-net to five models from (Ronneberger et al. 2015; He et al. 2015; Zhang et al. 2018; Sandler et al. 2018; Iandola et al. 2016). We assessed both the inference speed and prediction accuracy of each model as well as their training time and generalisability. When comparing these models we incorporated the number of model parameters, the model size on the hard disk, and the model image throughput per second. We applied the same configuration for each of the six investigated models in order to hold the training routine the same for all models. The models are also trained using the same data augmentations, without affecting their performance.

Figure 1 shows a high-level flow diagram that outlines the key steps involved in our proposed method. The flow diagram consists of eight main steps: data collection, annotation, data splitting, model development, training, validation, testing, and evaluation. The flow diagram illustrates how we developed and tested our novel deep learning network for fish morphometry using landmark detection. The following sections describe in detail the materials and methods used in this work.

Model architecture

We propose the Mobile Fish Landmark Detection network (MFLD-net), a novel end-to-end keypoint estimation model designed as a lightweight architecture for mobile devices. We apply the architecture

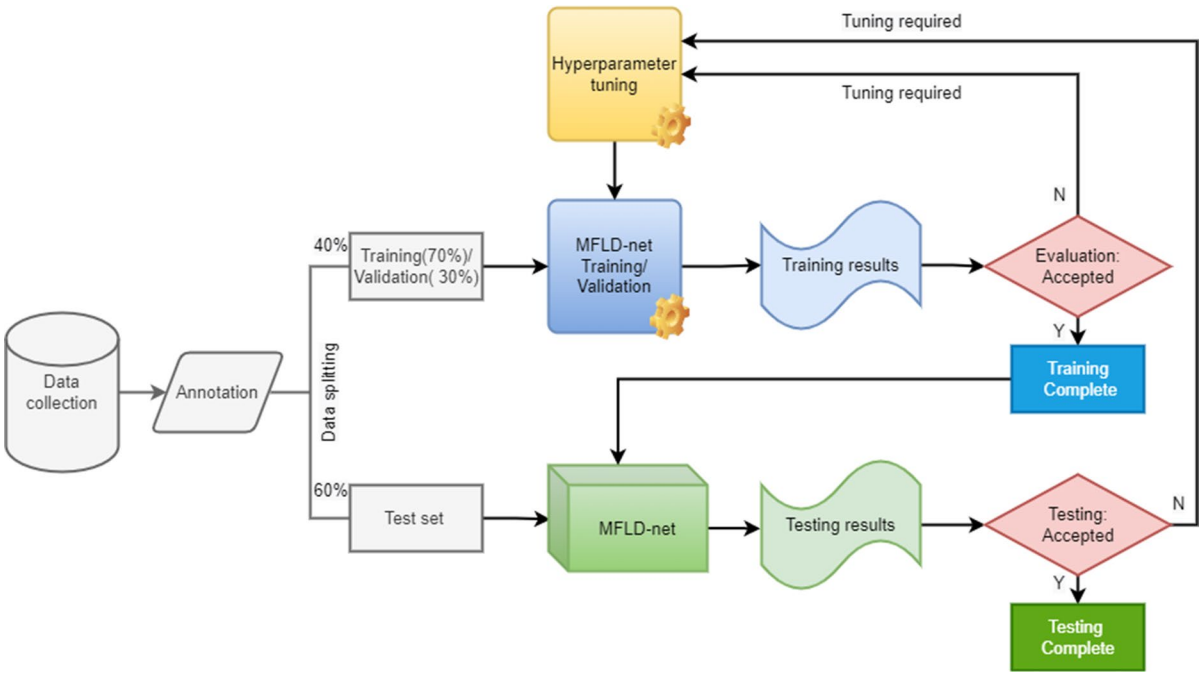

Fig. 1 A flow diagram that outlines the key steps involved in our proposed method. The flow diagram consists of eight main steps: Data collection: Collection of fish images using a high-performance CMOS industrial camera. Annotation: Manual annotation of the images for 16 keypoints per fish. Data splitting: Random splitting of the annotated dataset into training and validation sets (70% and 30%, respectively) and a test set (60%). Model development: Development of the Mobile Fish Landmark Detection network (MFLD-net) using convolution operations based on Vision Transformers, including patch embeddings and multi-layer perceptrons. Training: Training of the MFLD-net on the training set. Validation: Validation of the MFLD-net on the validation set to ensure that it is not overfitting to the training set. Testing: Testing of the MFLD-net on the test set and comparison of its performance to five other state-of-the-art baseline models. Evaluation: Evaluation of the MFLD-net's performance, including detection accuracies and generalisation capabilities

to address some of the main issues of current methods such as accuracy and efficiency on mobile and static keypoint estimation. The detailed architecture of MFLD-net is shown in Fig. 2. It builds upon convolutional neural networks (CNNs) (Sandler et al. 2019), Vision Transformer architecture (Dosovitskiy et al. 2020), and multi-layer perceptrons (MLP-Mixer) (Tolstikhin et al. 2021). Additionally, MFLD-net adapts a hybrid method for processing confidence maps and coordinates that provides accurate detection for estimating keypoint locations.

To achieve higher robustness and efficiency, our architecture leverages the use of patch embedding (Dosovitskiy et al. 2020), spatial/channel locations mixing (Tolstikhin et al. 2021), as well as a combination of CNNs that have the same size and shape throughout the network, i.e. are isometric (Sandler et al. 2019).

Isometric rrchitecture

Our model architecture is based on isometric convolutional networks (Sandler et al. 2019), which are made up of several similar blocks with the same resolution across the model. Architectures that are "Isometric" have the same size and shape throughout the network and maintain a fixed internal resolution throughout their entire depth (see Fig. 2).

Sandler *et al.* (Sandler et al. 2019) have demonstrated that the resolution of the input picture has only a minimal impact on the prediction quality of modern CNNs. Instead, the trade-off between accuracy and the number of multiply-adds required by the model is mostly determined by the internal resolution of intermediate tensors. Also, model accuracy can be improved further without the use of additional parameters given a fixed input resolution.

Therefore, our model has two main attributes: (1) No pooling layers while still maintaining a high receptive field. (2) Isometric networks have a high degree of accuracy while needing relatively little inference memory. These attributes make our model lightweight, hence suitable for edge processing on mobile and low-power devices, such as drones and robots, which are commonplace across various industries ranging from agriculture (Lammie et al. 2019) to marine sciences (Jahanbakht et al. 2022). This lightweight design does not, however, compromise accuracy due to its use of an isometric architecture. In an era of mobile processing (Jahanbakht et al. 2021), there is a significant need for lightweight, yet powerful, and effective keypoint estimation models.

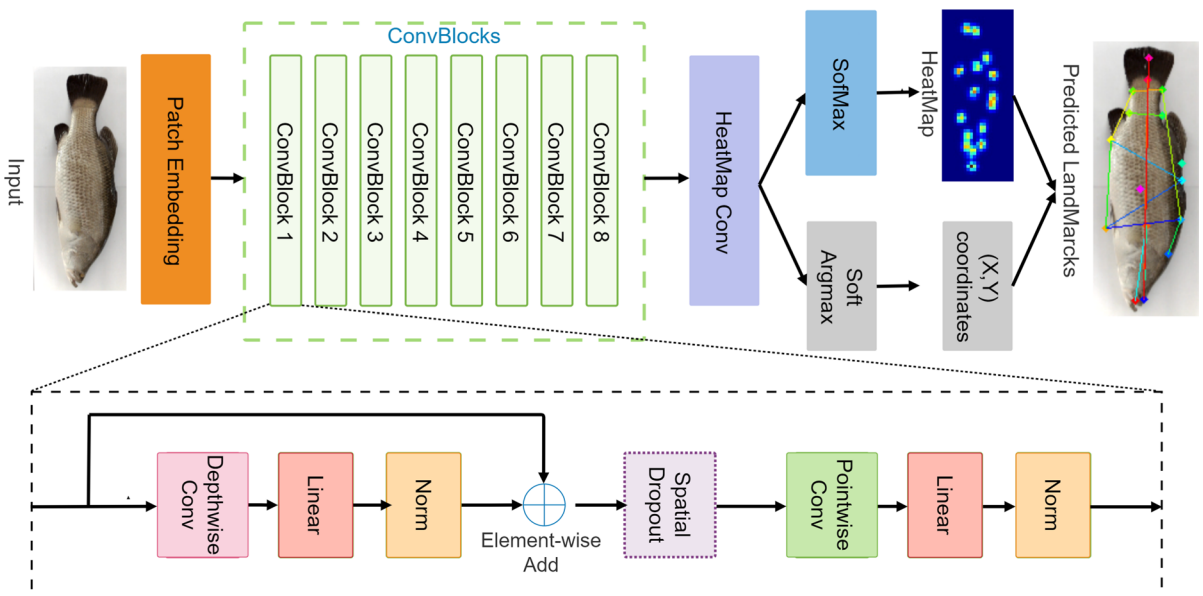

Fig. 2 Proposed MFLD-net architecture, which is similar in spirit to the ViTs, but uses convolutions operations for keypoints estimation

Patch embedding

Inspired by the Vision Transformer architecture (Dosovitskiy et al. 2020), we experiment with applying patch embeddings directly to a standard CNN. To do so, we divide an image into patches and feed a CNN tensor layout patch embeddings to preserve locality. In an NLP application, image patches are processed similarly to tokens (words). Patch embeddings enable all downsampling to occur simultaneously, lowering the internal resolution and therefore increasing the effective receptive field size, making it simpler to combine sparse spatial information. The key advantage of using CNN instead of transformer is the inductive bias of convolution (Li et al. 2018; Cohen and Shashua 2017) such as translation equivariance and locality. Therefore, CNN is well-suited to vision tasks because it generalises well when trained on a small dataset. We implemented patch embeddings as convolution with 3 input channels, 256 output channels, kernel size of 4, and stride of 4, followed by 8 ConvBlocks, as can be seen in Fig. 2.

ConvBlock

Our architecture is made of 8 ConvBlocks, each consisting of depthwise convolution (i.e. mixing spatial information) as in multi-layer perceptrons (MLP-Mixer) (Tolstikhin et al. 2021), and spatial dropout (Lee and Lee 2020) for strongly correlated pixels, followed by pointwise convolution (i.e. "mixing" the per-location features). Each of the convolutions is followed by Gaussian error linear units (GELU) (Hendrycks and Gimpel 2016) activation and BatchNorm. We found that for the task of keypoint estimation, architectures with a fewer number of layers result in better performance. We also added residual connections between Conv layers. We use a dropout with a rate of 0.2 to prevent overfitting. The structure of our ConvBlock can be seen in the bottom panel of Fig. 2.

Hybrid prediction and a multitask loss function

Fully convolutional networks (FCNs) 2 are good at transforming one image to produce another related image, or a set of images while preserving spatial information. Therefore, for our keypoint task, instead of using FCN to directly predict a numerical value of each keypoint coordinate as an output (i.e. regressing images to coordinate values), we modified FCN to predict a stack of output heatmaps (i.e. confidence maps), one for each keypoint. The position of each keypoint is indicated by a single, two-dimensional, symmetric Gaussian in each heatmap in the output, and the scalar value of the peak reflects the prediction's confidence score.

Moreover, our network not only predicts heatmaps but also predicts scalar values for coordinates of each keypoint. Therefore, during the training process, we have a multitask loss function, which consists of two losses, i.e. Jensen–Shannon divergence for heatmaps and Euclidean distance for coordinates (see Fig. 3). The first loss measures the distances between the predicted heatmaps and the ground truth heatmaps, while the second loss measures the distances between

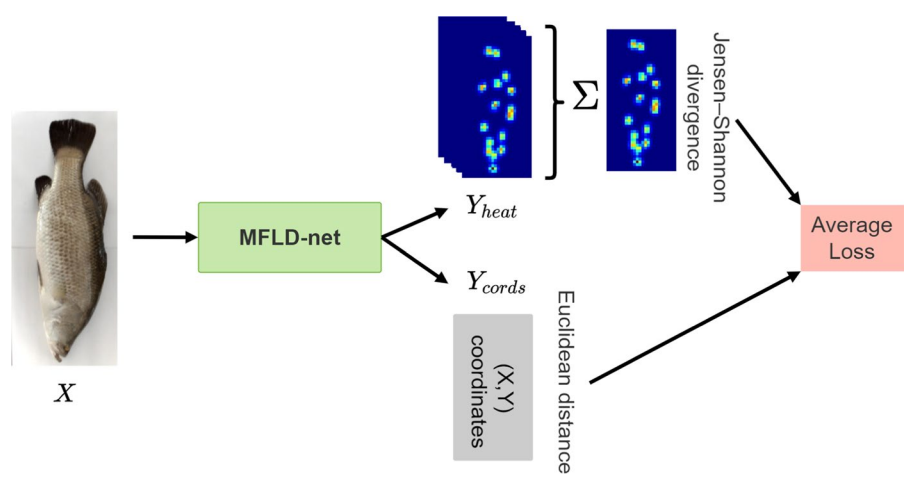

Fig. 3 A schematic diagram of the multitask loss function used for training MFLD-net

the predicted coordinates and the ground truth coordinates. Then, we take the average of the two losses as the optimisation loss.

Datasets

We performed experiments using a dataset of Barramundi (*Lates calcarifer*), also known as Asian seabass. These fish were photographed in a laboratory setting. The dataset was collected in four data collection sessions using the same experimental data collection setup but under four different environmental, i.e. lighting, conditions. Figure 4 demonstrates a sample image from each of the four data collection trials. In total, 2500 images were collected, each of which was photographed on a conveyor belt with normal ambient lighting. The images were recorded from above using a high-performance CMOS industrial camera (see Fig. 4). All barramundi were provided by the aquaculture team from James Cook University, Townsville, Australia.

To demonstrate the robustness of our network, we trained and validated our network on only 40% of the dataset. This training subset was further split into randomly selected training and validation sets, with 70% training examples and 30% validation examples. The other 60% of the collected dataset was used only for testing the model and comparing its performance to five other state-of-the-art baseline models (Ronneberger et al. 2015; He et al. 2015; Zhang et al. 2018; Sandler et al. 2018; Iandola et al. 2016). The images were manually annotated for 16 kepoints as shown in Fig. 5-middle. For each fish, ground truth keypoints have the form $[(x_1, y_1), ..., (x_k, y_k)]$, where (x_i, y_i) represents the ith keypoint location. Each ground truth object also has a scale s which we define as the square root of the object segment area. For each fish, our developed keypoint detector model outputs keypoint locations (see Fig. 5-right). Predicted keypoints for each fish have the same form as the ground truth, i.e. $[x1, y1, ..., xk, yk]$.

We recognise that the imaging setup used in our study was in a laboratory setting and may not fully represent the conditions of a real-world fishery. Factors such as lighting, background, and fish movement may vary significantly between a laboratory setting

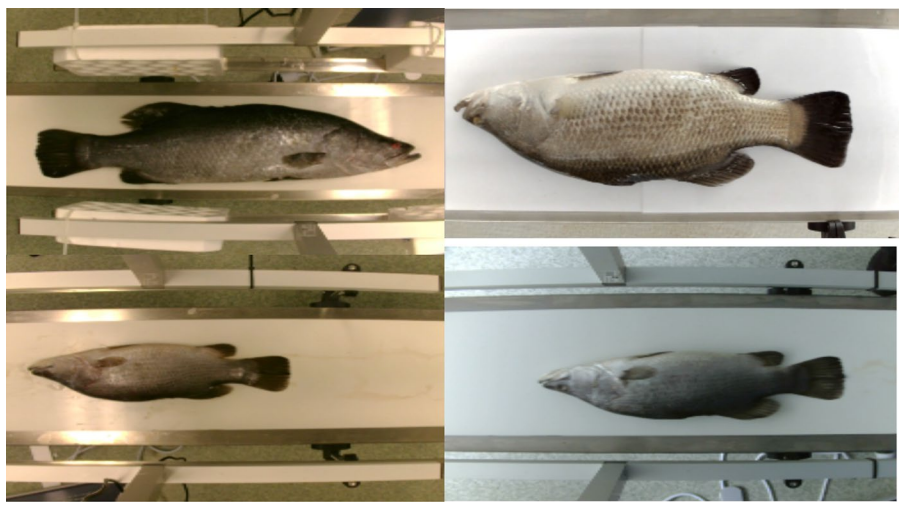

Fig. 4 Sample images from the four data collection sessions, which are all used in our experiments

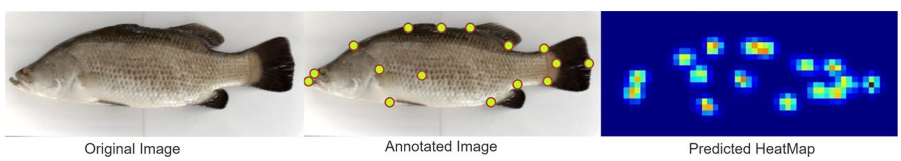

Fig. 5 Point annotations in a sample fish image (**X**) (left). The points in the training (**Y**) are inflated and highlighted for visibility, but only the centre pixel and its class label are collected and used (middle). The predicted heatmap of the model (right)

and a fishery. We chose to use a laboratory setting for our data collection to have a controlled and consistent environment, which reduces noise and variability in the images and improves the quality of the data for effective model development. This setup also facilitates the important annotation process, which requires manual labelling of 16 keypoints for each fish image to train the deep learning models.

We acknowledge that using a laboratory setting for our data collection has some limitations. One of the main challenges is transferring our model to a fishery setting, where the imaging conditions may vary significantly from our laboratory setup. For example, a fishery setting may have different lighting conditions, backgrounds and environments, as well as different fish species, sizes and shapes. These factors may affect the performance and accuracy of our model and the baseline models.

To address these issues and make our model as generalisable as possible, we used four different lighting conditions for our data collection, which simulate some of the variations that may occur in a fishery setting. In addition, we collected data from various fish sizes and in different orientations to augment our data collection. Furthermore, all our data were collected using a high-performance CMOS industrial camera, which is a common choice for other monitoring activities at fisheries (Jiang et al. 2017).

Data augmentation

To improve the training of our network and examine its robustness to rotation, translation, scale, and noise, we apply spatial and pixel level augmentation to our training data for all models using Albumentations library (Buslaev et al. 2020). In particular, we apply the following image transformations:

(1) Randomly flip an image horizontally with a probability of 0.5.
(2) Randomly flip an image vertically with a probability of 0.5.
(3) Randomly shift and scale an image with shift limit of 0.0625°, scale limit of 0.20° with a probability of 0.5.
(4) Randomly rotate an image with a rotation limit of 20° with a probability of 0.5.
(5) Randomly blur an image with blur limit of 1 with a probability of 0.3.
(6) Randomly RGB-shift an image with R-shift limit of 25, G-shift limit of 25, B-shift limit of 25 with a probability of 0.3. These augmentations help to further ensure robustness to shifts in lighting.

We did not apply any of the image transformation operations to our validation or test sets.

Performance metrics

The following metrics were used to optimise and evaluate the model and to compare the quality of the predicted keypoint locations:

Euclidean distance measures the distance of the keypoints based on their coordinates (i.e the line segment between the two points), and does not depend on how the ground truth has been determined (Wang et al. 2005). The best value of 0 indicates that the predicted keypoint is exactly at the same coordinate of the ground truth keypoint.

We calculate the sum of the squared Euclidean distance of the difference between two feature vectors, i.e. the predicted feature vector and the ground truth feature vector. This represents the total difference between the two feature vectors. The Euclidean distance is

$$d(g,p) = \sqrt{\sum_{i=1}^{n}(v_i^g - v_i^p)^2}, \tag{1}$$

where g and p are two sets of points in Euclidean n-space for ground truth and prediction, respectively. v_i^g, v_i^p are Euclidean vectors, starting from the origin of the space (initial point) for the ground truth and prediction, respectively. n is the number of keypoints.

Jensen–Shannon divergence is a distance measure between two distributions, such as the difference between the predicted and ground truth point distributions (Nielsen 2020). It can therefore be used to quantify the accuracy of the predicted keypoints. The lower this value is, the better the model performs.

This distance is calculated based on the Kullback–Leibler divergence (KLD) (Contreras-Reyes and Arellano-Valle 2012), where the inputs for the summation are probability distribution pairs. The *KLD* for two probability distributions, P and Q and when there are n pairs of predicted p, and ground truth g, can be expressed as:

$$KLD(P||Q) = \sum_{i=1}^{n} p_i(x) log\left(\frac{p_i(x)}{q_i(x)}\right), \quad (2)$$

to measure the difference between two probability distributions over the same variable x and indicate the dissimilarity between the distributions. The best value is 0. Utilising *KLD*, *JSD* can be expressed as follows:

$$JSD_M(P||Q) = \sqrt{\frac{KLD(p \parallel m) + KLD(q \parallel m)}{2}}, \quad (3)$$

where m is the pointwise mean of p and q.

This is a measure of the difference between two probability distributions P and Q. As can be seen from the formula, the best value of 0 indicates no difference between the distributions.

Object Keypoint Similarity (OKS) OKS keypoints estimation serves the same purpose as Intersection over Union (*IoU*) as in object detection. It is determined by dividing the distance between expected and ground truth points by the object's scale (Lin et al. 2014). This gives the similarity between the keypoints (or corners) of the two detected boxes. The result is between 0 and 1, where 0 means no similarity between the keypoints, while perfect predictions will have OKS=1. The equation is as follows:

$$OKS = \frac{\sum_i \exp\left(-d_i^2/2s^2k_i^2\right)\delta\left(v_i > 0\right)}{\sum_i \delta\left(v_i > 0\right)}, \quad (4)$$

where d_i is the Euclidean distance between the detected keypoint and the corresponding ground truth, v_i is the visibility flag of the ground truth, s is the object scale, while $k_i s$ represents a per-keypoint constant that controls falloff.

To compute OKS, we pass the d_i through an unnormalized Gaussian with standard deviation $k_i s$. For each keypoint, this yields a keypoint similarity that ranges between 0 and 1. These similarities are averaged over all labelled keypoints. Given the OKS, we can compute Average Precision (*AP*) and Average Recall (*AR*) just as the *IoU* allows us to compute these metrics for box/segment detection.

Both equations 1 and 3 have been used for model training and optimisation, and also used to compare different models' performance as in Table 1. Equation 4 was used as a final evaluation metric for all the models used in this study.

Model training

We trained six different models on the training subset. The models used for training are U-net (Ronneberger et al. 2015), ResNet-18 (He et al. 2015), ShuffleNet-v2 (Zhang et al. 2018), MobileNet-v2 (Sandler et al. 2018), SqueezeNet, (Iandola et al. 2016) and our proposed lightweight network MFLD-net. For each experiment, we set our model hyperparameters to the same configuration for all models. All the models were trained with 224×224 resolution input and 56×56 heatmap resolution output except U-net (Ronneberger et al. 2015) with 224×224 resolution for both input and output (see Fig. 6). Each model has two outputs (heatmap and coordinates), where two losses were applied as shown in Fig. 3.

We found that for this problem set, a learning rate of 1×10^{-3} works the best. It took around 50 epochs for all models to train on this problem and the learning rate was decayed by $\gamma = 0.1$ every 30 epochs. Our networks were trained on a Linux host with a single NVidia GeForce RTX 2080 Ti GPU using Pytorch framework (Paszke et al. 2019). The batch size we used was 64. We used Adam optimiser (Kingma et al. 2014) with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1.0 \times 10^{-08}$. We applied the same hyperparameter configuration for all six models. The optimum model configuration will depend on the application, hence, these results are not intended to represent a complete search of model configurations.

Because we only used the training subset ($n = 1000$ images) for training and validation, during optimization, we heavily augmented our training set, challenging the model to learn a much broader data distribution than that in the training set. We applied several image transformations for data augmentation as specified in Sect. 2.3.

We regarded the model to be converged when the validation loss stopped improving after 50 epochs. Only for the best performing version of the models, we calculated validation error as the Euclidean distance between predicted and ground truth picture coordinates and Jensen–Shannon divergences between heatmaps and centres of the target Gaussians, which we assessed at the end of each epoch during optimization. Figure 7 shows the training and validation losses for our proposed network.

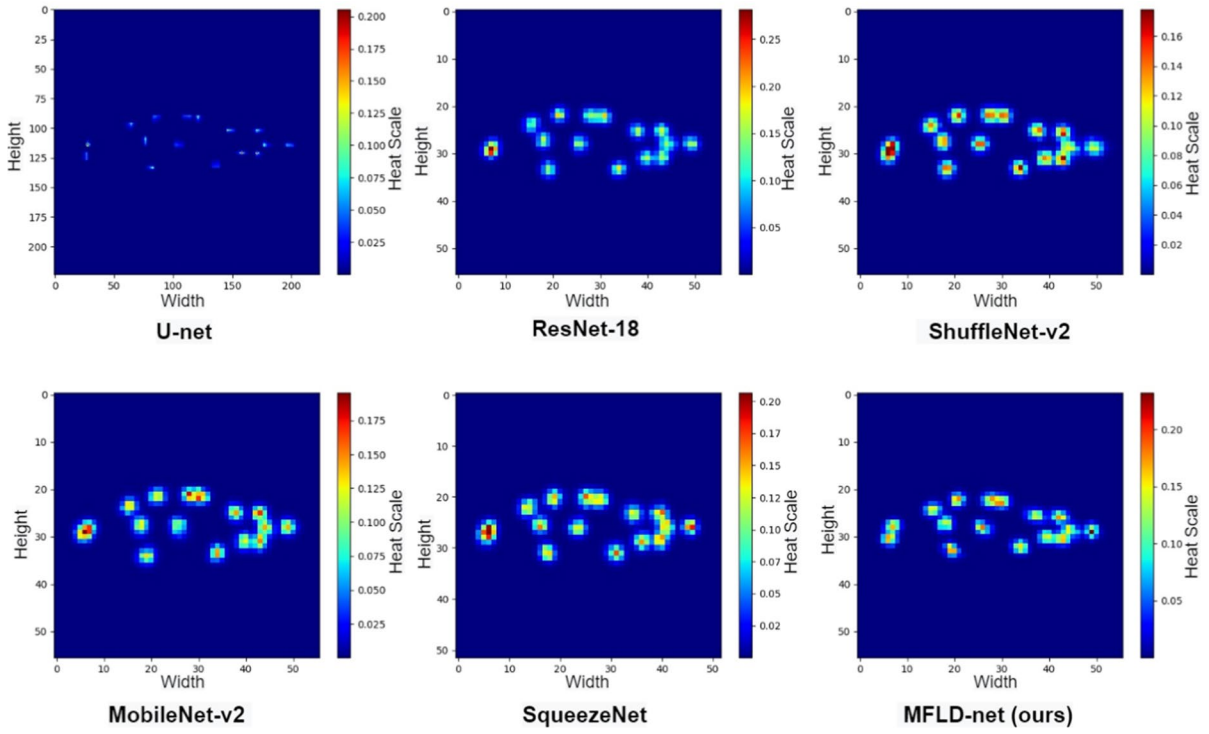

Fig. 6 Sample output heatmap from each of the 6 networks used in this work

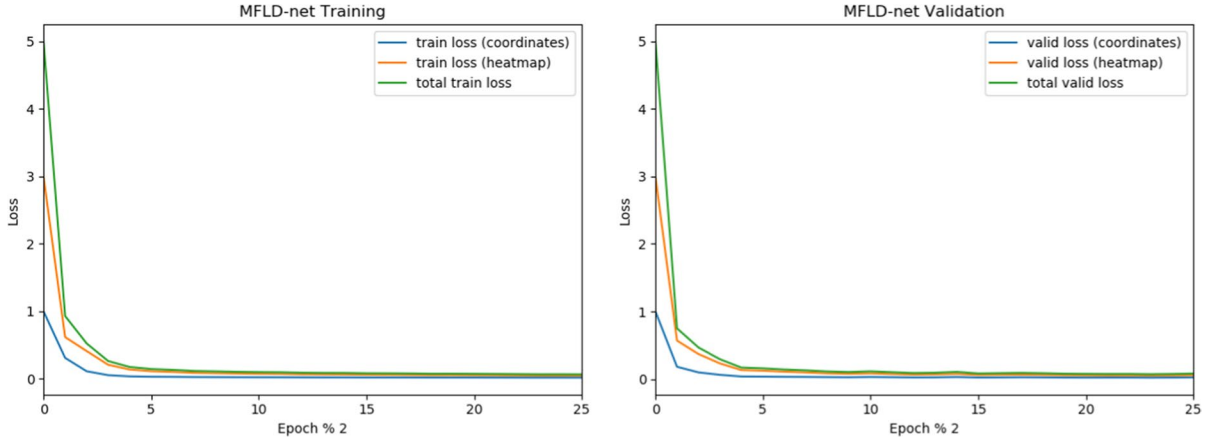

Fig. 7 The two different losses, i.e. coordinate (Eq. 1) and heatmap (Eq. 3) prediction losses are shown along with the total loss for both training and validation

Model evaluation

Deep learning (DL) models are typically evaluated for their predictive performance (i.e. ability to generalise to new data), using a sub-sample of annotated data (test set) that is not used for training or validation. A test set is typically used to avoid overfitting the model hyperparameters to the validation set, which can result in biased performance measurements. Therefore, we used only 40% of our dataset for

training and left the other 60% of its images for testing the model's predictive performance using metrics described in Sect. 2.4.

Precision and Recall Object Keypoint Similarity (OKS) (Lin et al. 2014) was used as a performance metric (see Sect. 2.4 for more details). As explained, the following 6 metrics are usually used for characterising the performance of a keypoint detector model. We, therefore, used them.

- Average Precision (*AP*):
  - AP (at $OKS = .50 : .05 : .95$ (primary metric))
  - $AP^{.50}$ (at $OKS = .50$)
  - $AP^{.75}$ (at $OKS = .75$)
- Average Recall (*AR*):
  - AR (at $OKS = .50 : .05 : .95$)
  - $AR^{.50}$ (at $OKS = .50$)
  - $AR^{.75}$ (at $OKS = .75$)

Results

To fully evaluate our model and compare it with other methods, we ran experiments to optimise our approach and compared it to the five aforementioned models in terms of image throughput (speed), accuracy, inference time, and generalisation ability. We benchmarked these models using the test subset (see Sect. 2.2 for details).

We applied the same training configuration for all of the six models, meaning that the models are all trained using the same dataset and data augmentations as explained in Sect. 2.5.

Performance comparison

Table 1 shows comparative results based on the number of parameters of a model, the model size on the hard disk, and the model throughput in image per second. In addition, the coordinates loss (Eq. 1), heatmap loss (Eq. 3), and the average of both losses are shown. All the tests were conducted on a desktop computer with a single NVidia GeForce RTX 2080 Ti GPU.

Overall, the results summarised in Table 1 show that our network (MFLD-net) outperforms other networks, achieving the lowest number of parameters (47x fewer parameters than U-net (Ronneberger et al. 2015)), the smallest size on the hard disk, and the second-highest throughput after SqueezeNet (Iandola et al. 2016). Also, our model has a lower average loss than U-net (Ronneberger et al. 2015), ShuffleNet-v2 (Zhang et al. 2018), and MobileNet-v2 (Sandler et al. 2018). The small number of parameters as well as the very compact size of our model while having a high throughput makes it an appealing solution for many problems such as real-time mobile fish video processing and portable autonomous systems (Saleh et al. 2022).

To examine the efficacy of our model generalisation, we compared its performance with randomly initialised weights, against the five benchmark models with randomly initialised weights to provide a direct comparison. We show in Table 2 that our MFLD-net model achieves good generalisation with few training examples and without the use of transfer learning when combined with strong data augmentation. Overall, the results summarised in Table 2 show that our network (MFLD-net) outperforms ShuffleNet-v2 (Zhang et al. 2018), MobileNet-v2 (Sandler et al. 2018), and SqueezeNet (Iandola et al. 2016) achieving $AP = 0.967$, while being competitive with U-Net and ResNet, despite having substantially fewer parameters. This shows the effectiveness and generalisability of our MFLD-net model.

Qualitative results

To further confirm our model generalisation on the unseen images, we perform a qualitative experiment on the test subset, with sample results shown in Fig. 8. This figure clearly shows that our network performs better than the previous methods. The other methods have the problem of misclassifying the pixels with a similar intensity of one colour as the other colour, whereas our method shows a strong ability to differentiate pixels with similar intensity. We can also clearly see that the proposed method can work on images with different lighting conditions (Fig. 9).

Fish morphometry

Morphometry is the study of the size and shape of organisms and their variation. Fish morphometry is

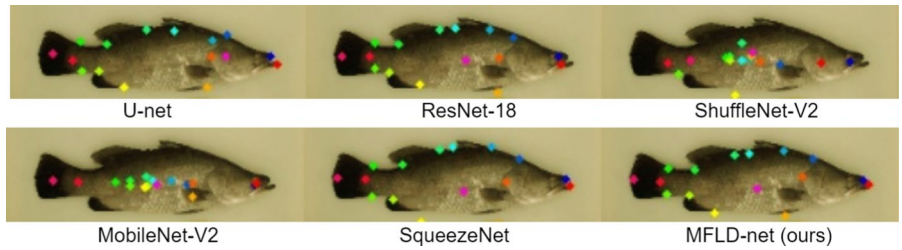

Fig. 8 Example keypoints estimation predicted by the proposed network and a state-of-the-art CNNs

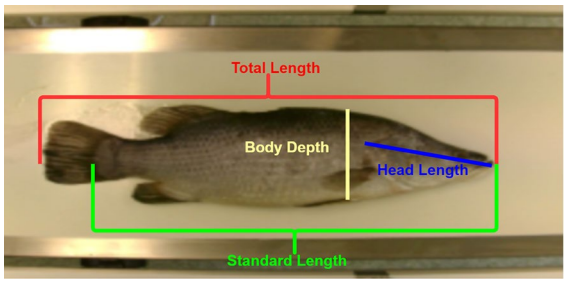

Fig. 9 Fish body measurement used in this study

a useful tool for fishery science, as it can help identify different species, populations, stocks, and growth patterns of fish (Tripathy 2020). Fish morphometry can be performed using traditional methods, such as measuring various body parts with a ruler or a caliper, or using advanced methods, such as image analysis, and deep learning. These methods provide an efficient approach to extract more information on the shape and variation of fish, automatically and cost-effectively.

Fish body measurement used in this study

In this study, we used four body measurements to describe the morphometry of fish: total length, standard length, body depth, and head length. These important morphological measurements are widely used in monitoring fish, for example, its growth (Jerry et al. 2022). The four measurements automated using our approach are depicted in 9 and are defined as follows:

Total length is the overall length of the fish, measured from the tip of the snout to the end of the tail fin. This measurement is important for determining the overall size of the fish, which is relevant for various ecological and management purposes, such as estimating growth rates, biomass, and abundance.

Standard length is the length of the fish from the tip of the snout to the end of the vertebral column, excluding the caudal fin. This measurement is more appropriate for comparing the body proportions and shape of fish among different species or populations, as it removes the variation introduced by the size of the tail fin.

Body depth is the maximum vertical distance between the dorsal and ventral body surfaces, usually measured at the midpoint of the body length. This measurement reflects the thickness or robustness of the fish body, which can be related to its feeding habits, swimming ability, and reproductive strategy.

Head length is the distance from the tip of the snout to the posterior margin of the operculum, which covers the gills. This measurement is relevant for assessing the size and shape of the head, which can provide information on the feeding behaviour, sensory perception, and phylogenetic relationships of the fish.

Therefore, the combination of these four measurements can provide a comprehensive description of the size, shape, and body structure of fish, which can be useful for various research and management applications. Figure 10 shows these measurements' distribution pair plots based on the automatic measurements captured by MFLD-Net. These plots are essential to show the distribution and correlation of these measurements, which can provide insights into the fish's morphometry and body structure (Fig. 11).

We are presenting these plots to demonstrate the effectiveness of our approach in automatically extracting these important morphological measurements from fish images. The automatic measurements captured by MFLD-Net can provide accurate and consistent results, which can save time and effort compared to manual methods.

Quantitative comparison

In the quantitative comparison of fish morphometry, it is important to compare the accuracy and precision

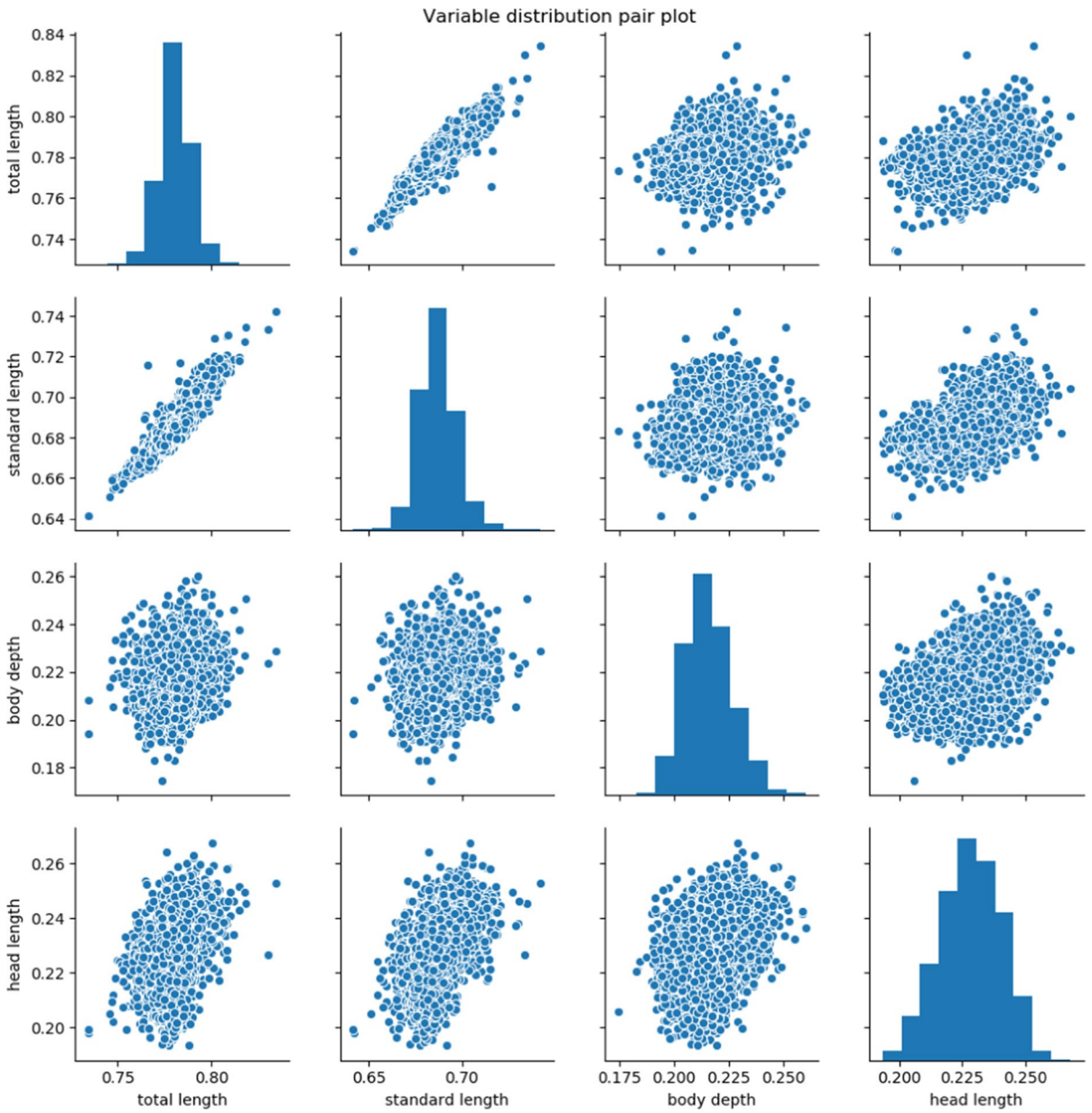

Fig. 10 The distribution pair plots for the four body measurements (total length, standard length, body depth, and head length) used to describe the morphometry of fish

of the DL measurements. Here, accuracy refers to how close the DL measurements are to their true manual measurements, while precision refers to the degree of consistency or reproducibility of results.

To assess accuracy and precision, the following metrics have been used in this study: the mean absolute difference (MAD), and the standard deviation of the difference (SDD) between the manual and DL measurements. For both of these measures, a lower value indicates better performance.

MAD is the average absolute difference between two values (accuracy). It is calculated by taking the sum of the absolute differences between each value and dividing it by the number of values. The formula for calculating MAD is:

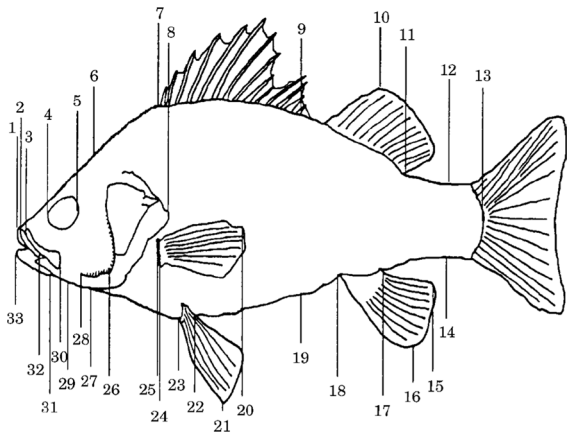

Fig. 11 Position of most of the landmark points used to describe shape variation in *M. novemaculeata* from seven geographically distinct rivers. See (Jerry and Cairns 1998) for an explanation of variables measured. Figure is from (Jerry and Cairns 1998)

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

where n is the total number of observations, x_i and y_i are the values of the $i-th$ observation in two different samples (here manual and DL measurements), and the vertical bars indicate absolute value.

SDD is a statistical measure that describes the amount of variation or dispersion between two sets of data. Specifically, it measures how spread out the differences between the two sets of data are (i.e. precision). SDD is calculated by taking the square root of the variance of the differences using the following formula:

$$SDD = \sqrt{\frac{\sum_{i=1}^{n}(x_i - y_i - \bar{x} + \bar{y})^2}{n-1}}$$

where x and y are the two sets of data, n is the number of observations in each set, and $\bar{x}$ and $\bar{y}$ are the means of the two sets.

Table 3 compares the performance of our MFLD-net model to other models in measuring fish morphometric traits such as total length, standard length,

Table 1 Performance comparison to other models

Network	# Params (x10^6)	Size (MB)	Throughput (img/sec)	Coords[1]	Losses HeatMap[2]	Avg.[3]
U-net (Ronneberger et al. 2015)	31.04	124.3	201	0.024	0.355	0.190
ResNet-18 (He et al. 2015)	12.85	51.5	404	0.028	0.090	0.059
ShuffleNet-v2 (Zhang et al. 2018)	3.06	12.5	170	0.047	0.153	0.100
MobileNet-v2 (Sandler et al. 2018)	4.10	16.7	205	0.041	0.137	0.089
SqueezeNet (Iandola et al. 2016)	2.33	9.4	551	0.027	0.078	0.052
MFLD-net (ours)	**0.65**	**2.7**	480	0.039	0.120	0.080

[1] This loss corresponds to the coordinates loss (Eq. 1)

[2] This loss corresponds to heatmap loss (Eq. 3)

[3] This loss corresponds to the average of both losses (Eq. 1 and Eq. 3)

Table 2 Performance comparison using the *OKS* metric on the **test** datasets

Network	AP	$AP^{.50}$	$AP^{.75}$	AR	$AR^{.50}$	$AR^{.75}$
U-net (Ronneberger et al. 2015)	0.968	0.990	0.990	0.983	0.999	0.999
ResNet-18 (He et al. 2015)	0.970	0.990	0.990	0.985	0.999	0.999
ShuffleNet-v2 (Zhang et al. 2018)	0.949	0.990	0.990	0.968	0.999	0.999
MobileNet-v2 (Sandler et al. 2018)	0.952	0.990	0.989	0.967	0.999	0.996
SqueezeNet (Iandola et al. 2016)	0.964	0.990	0.990	0.975	0.999	0.999
MFLD-net (ours)	0.967	0.990	0.990	0.983	0.999	0.999

Table 3 Performance comparison of various DL models in measuring four important fish morphological traits

Network	Total length		Standard length		Body depth		Head length	
	MAD	SDD	MAD	SDD	MAD	SDD	MAD	SDD
U-net (Ronneberger et al. 2015)	10.57	10.67	**08.00**	**08.64**	**06.23**	**06.04**	*06.58*	*06.77*
ResNet-18 (He et al. 2015)	*10.07*	10.64	09.28	09.82	09.06	09.31	12.96	12.03
ShuffleNet-v2 (Zhang et al. 2018)	14.11	14.95	12.31	12.05	11.90	11.21	15.48	15.70
MobileNet-v2 (Sandler et al. 2018)	15.45	15.04	14.33	14.78	16.73	16.87	11.80	12.89
SqueezeNet (Iandola et al. 2016)	*10.07*	*10.04*	09.28	*09.64*	09.06	09.20	12.96	11.38
MFLD-net (ours)	**09.25**	**07.60**	*08.27*	09.74	*07.62*	*07.34*	**06.57**	**06.57**

The mean absolute difference (MAD), and the standard deviation of the difference (SDD) between the manual and DL measurements in (mm) are shown. The best two results are shown in bold and italics, with bold corresponding to the best value and italics corresponding to the second best

body depth, and head length. MAD and SDD between manual and deep learning (DL) measurements are reported in mm for each model. The top two results are highlighted in red and blue, with red corresponding to the best value and blue corresponding to the second best.

The MFLD-net model, proposed in this study, outperforms the others with MAD values of 9.25 and 6.57 for total length and head length. It also performs well for standard length and body depth with MAD values of 8.27 and 7.62, highlighted in blue. In addition, its SDD values are competitive with other models.

The U-net model shows the second-best performance and the ResNet-18 model performs well for standard length and body depth with MAD values of 9.28 and 9.06 but performs poorly for head length. The ShuffleNet-v2 and MobileNet-v2 models have higher MAD and SDD values for all traits. The SqueezeNet model performs well for total length but has high MAD values for standard length and head length.

Discussion

Fish morphology determination is required for both selecting and evaluating novel fish strains for cultivation. The most widely used method to characterise fish is by observation of their overall appearance. An experienced observer can determine a fish's size, weight, possibly sex, and even its condition. The traditional observation method to evaluate fish morphology includes weighing fish, measuring lengths with a ruler or callipers and or some other aspect of the fish, and then recording these observations. This observation process is slow, labour-intensive, and highly prone to human error.

A possible solution could automate the fish observation process if an accurate mobile system is developed that can be deployed in the field and in fish farms. This fish morphometric tool could quickly measure various fish features and morphological traits from fish images captured online or offline using a camera. The tool also collects the morphological data, and then uses it for analysis and producing a final report. Such a tool is very useful to aquaculture and fish farms and could provide a new way to select, evaluate, and analyse fish and other aquaculture animal products.

In this paper, we developed a novel deep learning algorithm for accurate fish morphometric measurements from fish images. To efficiently measure various fish morphological traits, we developed a fish-specific landmark detection model that could accurately localise keypoints (landmarks) on the fish body (for example see Fig. 8). These landmarks can be then used to rapidly measure various fish traits including their weight, length, head shape, and body shape. In addition, the fish landmarks can be used to describe shape variation, deformation and differential development in various fish species (Jerry and Cairns 1998; Jerry et al. 2022).

We build our landmark detection model upon the most widely used deep learning variant, i.e. CNN. A number of factors can significantly influence CNNs performance. These factors include the size of the network (including the number of layers, number of

kernels, and their width), the number of input features, and the size of the training set. In addition, the use of the convolution layers affects the size and complexity of the network but can help to decrease the error rate and improve prediction accuracy. However, there is no clear mechanism to arrive at the optimal convolutional architecture for a specific task. The architecture selection involves choosing important hyperparameters such as the network structures and the training time.

Through experimentation and using our experience in developing deep learning algorithms, we designed a lightweight CNN with a short training time and high generalisability to make it suitable for fast deployment and real-time mobile applications in fish farms. Our experiments showed that MFLD-net best performances can be achieved by (i) increasing the size of the kernel to 9, (ii) including more input dimensions by patch embeddings, and (iii) reducing the number of convolution layers to 8. The reduction in the number of convolution layers resulted in a model with fewer parameters that achieved better generalisation capabilities compared to the state-of-the-art models.

To train and evaluate our model performance, we collected a dataset containing 2500 harvested or sedated fish images. These images were manually annotated for important landmarks on the fish body. We used a combination of data augmentation techniques to improve the network's performance in a low data regime. In our experiments, the input images were scaled to a size of 224×224, and the output was the position of each fish landmark. These landmarks (keypoints) were indicated by a single, two-dimensional, symmetric Gaussian heatmap, where a scalar peak value reflects the prediction's confidence score. The quantitative and qualitative experimental results showed that our proposed model while being significantly lighter, can outperform some and be competitive with other state-of-the-art models. We also showed that our model has a high generalisation capability and does not need transfer learning even when using a small training dataset.

To deploy our model to real-world fisheries setting, one approach is to perform site-specific model tuning using our baseline MFLD-Net. This means that, before deployment to a new setting, we collect some new data and retrain our model to adjust it to the new environment as well as task conditions. This adjustment is much faster and more efficient than developing a new model for the new setting. The newly added data can diversify the model's generalisation capabilities and gradually improve its performance in a wider set of environments. To perform this adjustment quicker, one approach is to use self-supervised learning techniques to shorten the time required for a large amount of data labelling, and to add more data from other sources to improve our model.

The main limitation of our study is that all the samples for training and testing are taken from a similar source, even though, they were slightly different, due to being collected in different conditions and by different operators. Another limitation is the use of a single fish species in our dataset. Since there are a variety of different species and sizes of fish in the aquaculture industry, there is a need to test the model for more than one species. However, our aim in this study was to build a proof of concept, which can be extended in future works to other species. Our presented results indicate that our developed MFLD-net model trained using images from a single species could be generalised to detect fish of different species and in different environments. This could be the subject of future research.

Furthermore, we should emphasise that our model is not designed to classify fish species, but rather to detect landmarks on fish bodies that can be used for morphometric analysis. However, it is possible to extend our model to handle different fish species by using techniques such as multitask learning or domain adaptation. For example, multitask learning could be used to train our model to simultaneously detect landmarks and classify fish species, while domain adaptation techniques could be used to adapt our model to new fish species with minimal additional training data. These are potential avenues for future research and development of our model.

In addition, in future work, the model can be trained with images of other objects, or images captured from different fish species. It is worth noting that, collecting new fish images and annotating them is a time-consuming and expensive exercise. This was the case, even in our data collection trials, where fish images were collected when the fish passed on a conveyor belt and under a camera capturing videos.

In addition, developing new low-cost, low-power, and high-speed mobile devices has been an evolving research area in many applications such as agriculture (Lammie et al. 2019), and marine science (Jahanbakht

et al. 2021, 2022). These devices need a lightweight and fast network, such as the proposed model in this work. Therefore, an interesting future research project is to develop a low-cost mobile device to perform fish morphology estimation using the proposed network.

Conclusion

In conclusion, our research demonstrated the potential of using a vision transformer-inspired CNN for fish landmark detection and morphology measurement. Our proposed model outperforms existing deep learning models in terms of accuracy and speed while using fewer parameters, making it suitable for deployment on mobile and resource-constrained devices. This advancement brings us closer to practical applications in the rapidly growing aquaculture and fisheries industries. Future research will focus on testing the model on a wider range of aquaculture animals and exploring other CNN architectures for fish landmark detection.

Acknowledgements This research is supported by an Australian Research Training Program (RTP) Scholarship and Food Agility HDR Top-Up Scholarship. D. Jerry and M. Rahimi Azghadi acknowledge the Australian Research Council through their Industrial Transformation Research Hub program.

Author contributions AS contributed to conceptualisation, data curation, data analysis, software development, DL algorithm design, writing—original draft and is the principal author. DJ contributed to data curation, data analysis, and editing the draft. DJ contributed to conceptualisation, PhD supervision, and reviewing/editing the draft. MRA contributed to conceptualisation, data curation, data analysis, PhD supervision, and reviewing /editing the draft.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions. No funding was received to assist with the preparation of this manuscript. The authors have no relevant financial or non-financial interests to disclose.

Data availability The datasets generated during and analysed during the current study are not publicly available due to data protection reasons. However, the underlying data are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Alzayat Saleh, David Jones, Dean Jerry, Mostafa Rahimi Azghadi have patent pending to James Cook University.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors. Informed consent was obtained from all individual participants included in the study.

References

Bahdanau D, Cho KH, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. In: 3rd international conference on learning representations, ICLR 2015 - Conference Track Proceedings

Bello I, Zoph B, Le Q, Vaswani A, Shlens J (2019) Attention augmented convolutional networks. In: Proceedings of the IEEE international conference on computer vision, volume 2019-Octob, pp 3285–3294

Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA (2020) Albumentations: Fast and flexible image augmentations. Information (Switzerland), 11(2)

Castrillo PA, Varela-Dopico C, Bermúdez R, Ondina P, Quiroga MI (2021) Morphopathology and gill recovery of Atlantic salmon during the parasitic detachment of Margaritifera margaritifera. J Fish Dis 44:1101–1115

Cohen N, Shashua A (2017) Inductive bias of deep convolutional networks through pooling geometry. In: 5th International conference on learning representations, ICLR 2017 - Conference Track Proceedings

Contreras-Reyes JE, Arellano-Valle RB (2012) Kullback–Leibler divergence measure for multivariate skew-normal distributions. Entropy 14(9):1606–1626

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2020) An image is worth 16 x 16 Words: Transformers for Image Recognition at Scale. IEEE

Fernandes AFA, Turra EM, de Alvarenga R, Passafaro TL, Lopes FB, Alves GFO, Singh V, Rosa GJM, de Alvarenga ER, Passafaro TL, Lopes FB, Alves GFO, Singh

V, Rosa GJM (2020) Deep Learning image segmentation for extraction of fish body measurements and prediction of body weight and carcass traits in Nile tilapia. Comput Electron Agric 170:105274

Figueroa RI, De Bustos A, Cuadrado A (2018) A novel FISH technique for labeling the chromosomes of dinoflagellates in suspension. PLoS ONE 13:e0204382

He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. Comput Vis Pattern Recogn

Hendrycks D, Gimpel K (2016) Gaussian error linear units (GELUs). ICCV

Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K (2016) SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and<0.5MB model size. Comput Vis Pattern Recogn

Islamadina R, Pramita N, Arnia F, Munadi K (2018) Estimating fish weight based on visual captured. In: 2018 international conference on information and communications technology, ICOIACT 2018

Jahanbakht M, Xiang W, Hanzo L, Azghadi MR (2021) Internet of underwater things and big marine data analytics - a comprehensive survey. IEEE Commun Surv Tutor 23(2):904–956

Jahanbakht M, Xiang W, Waltham NJ, Azghadi MR (2022) Distributed deep learning in the cloud and energy-efficient real-time image processing at the edge for fish segmentation in underwater videos. IEEE Access, pp 1–1

Jerry DR, Cairns SC (1998) Morphological variation in the catadromous Australian bass, from seven geographically distinct riverine drainages. J Fish Biol 52(4):829–843

Jerry DR, Jones DB, Lillehammer M, Massault C, Loughnan S, Cate HS, Harrison PJ, Strugnell JM, Zenger KR, Robinson NA (2022) Predicted strong genetic gains from the application of genomic selection to improve growth related traits in barramundi (Lates calcarifer). Aquaculture 549:737761

Jiang B, Pan Z, Qiu Y (2017) Study on the key technologies of a high-speed CMOS camera. Optik 129:100–107

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980

Konovalov DA, Saleh A, Domingos JA, White RD, Jerry DR (2018) Estimating mass of harvested Asian seabass Lates calcarifer from images. World J Eng Technol 6(03):15

Konovalov DA, Saleh A, Efremova DB, Domingos JA, Jerry DR (2019) Automatic weight estimation of harvested fish from images. In: 2019 digital image computing: techniques and applications, DICTA 2019. Institute of Electrical and Electronics Engineers Inc

Lammie C, Olsen A, Carrick T, Rahimi Azghadi M (2019) Low-power and high-speed deep FPGA inference engines for weed classification at the edge. IEEE Access

Lee S, Lee C (2020) Revisiting spatial dropout for regularizing convolutional neural networks. Multimedia Tools Appl 79(45–46):34195–34207

Li L, Dong B, Rigall E, Zhou T, Dong J, Chen G (2022) Marine animal segmentation. IEEE Trans Circ Syst Video Technol 32(4):2303–2314

Li X, Grandvalet Y, Davoine F (2018) Explicit inductive bias for transfer learning with convolutional networks. In: 35th International conference on machine learning, ICML 2018, vol 6, pp 4408–4419

Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft COCO: Common objects in context. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Mathiassen JR, Misimi E, Toldnes B, Bondø M, Østvik SO (2011) High-speed weight estimation of whole herring (Clupea harengus) using 3D machine vision. J Food Sci 76(6):E458–E464

Newell A, Yang K, Deng J (2016) Stacked Hourglass Networks for Human Pose Estimation. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol 9912 LNCS, pp 483–499. Springer Nature

Nielsen F (2020) On a generalization of the Jensen–Shannon divergence and the Jensen–Shannon centroid. Entropy 22(2):221

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) PyTorch: an imperative style, high-performance deep learning library. Adv Neural Inform Process Syst

Powers AK, Garita-Alvarado CA, Rodiles-Hernández R, Berning DJ, Gross JB, Ornelas-García CP (2020) A geographical cline in craniofacial morphology across populations of Mesoamerican lake-dwelling fishes. J Exp Zoology Part A Ecol Integr Physiol 333:171–180

Ramachandran P, Bello I, Parmar N, Levskaya A, Vaswani A, Shlens J (2019) Stand-alone self-attention in vision models. Adv Neural Inform Process Syst, volume 32

Ren S, He K, Girshick R, Sun J (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149

Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention, pp 234–241

Saleh A, Sheaves M, Rahimi Azghadi M (2022) Computer vision and deep learning for fish classification in underwater habitats: a survey. Fish Fish 23(4):977–999

Sanchez-Torres G, Ceballos-Arroyo A, Robles-Serrano S (2018). Automatic measurement of fish weight and size by processing underwater hatchery images. Eng Lett, 26(4)

Sandler M, Baccash J, Zhmoginov A, Howard A (2019) Non-discriminative data or weak model? On the relative importance of data and model resolution. In: Proceedings - 2019 international conference on computer vision workshop, ICCVW 2019, pp 1036–1044

Sandler M, Howard AG, Zhu M, Zhmoginov A, Chen LC (2018) inverted residuals and linear bottlenecks: mobile networks for classification, detection and segmentation. CoRR, abs/1801.0

Suo F, Huang K, Ling G, Li Y, Xiang J (2020) Fish keypoints detection for ecology monitoring based on underwater visual intelligence. In: 16th IEEE international conference on control, automation, robotics and vision, ICARCV 2020

Tolstikhin I, Houlsby N, Kolesnikov A, Beyer L, Zhai X, Unterthiner T, Yung J, Steiner A, Keysers D, Uszkoreit J,

Lucic M, Dosovitskiy A (2021) MLP-Mixer: An all-MLP Architecture for Vision. In: Proceedings - 2021 international conference on computer vision workshop, ICCVW 2021

Touvron H, Bojanowski P, Caron M, Cord M, El-Nouby A, Grave E, Izacard G, Joulin A, Synnaeve G, Verbeek J, Jégou H (2021) ResMLP: feedforward networks for image classification with data-efficient training. IEEE

Tripathy SK (2020) Significance of traditional and advanced morphometry to fishery science. J Hum Earth Future 1(3):153–166

Tseng CH, Hsieh CL, Kuo YF (2020) Automatic measurement of the body length of harvested fish using convolutional neural networks. Biosyst Eng

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. arXiv preprint arXiv:1706.03762

Wang L, Zhang Y, Feng J (2005) On the Euclidean distance of images. IEEE Trans Pattern Anal Mach Intell 27(8):1334–1339

Zhang X, Zhou X, Lin M, Sun J (2018) ShuffleNet: an extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, pp 6848–6856