Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

Applications of deep learning in fish habitat monitoring: A tutorial and survey

Alzayat Saleh [a], Marcus Sheaves [a], Dean Jerry [a,b], Mostafa Rahimi Azghadi [a,b,*]

[a] *College of Science and Engineering, James Cook University, 1 James Cook Drive, Townsville, 4811, QLD, Australia*
[b] *ARC Research Hub for Supercharging Tropical Aquaculture through Genetic Solutions, James Cook University, 1 James Cook Drive, Townsville, 4811, QLD, Australia*

ARTICLE INFO

Keywords:
Marine Science
Computer Vision
Convolutional Neural Networks
Image and video processing
Machine Learning
Deep Learning
Deep Neural Networks

ABSTRACT

Marine ecosystems and their fish habitats are becoming increasingly important due to their integral role in providing a valuable food source and conservation outcomes. Due to their remote and difficult to access nature, marine environments and fish habitats are often monitored using underwater cameras to record videos and images for understanding fish life and ecology, as well as for preserve the environment. There are currently many permanent underwater camera systems deployed at different places around the globe. In addition, there exists numerous studies that use temporary cameras to survey fish habitats. These cameras generate a massive volume of digital data, which cannot be efficiently analysed by current manual processing methods, which involve a human observer. Deep Learning (DL) is a cutting-edge Artificial Intelligence (AI) technology that has demonstrated unprecedented performance in analysing visual data. Despite its application to a myriad of domains, its use in underwater fish habitat monitoring remains under explored. In this paper, we provide a tutorial that covers the key concepts of DL, which help the reader grasp a high-level understanding of how DL works. The tutorial also explains a step-by-step procedure on how DL algorithms should be developed for challenging applications such as underwater fish monitoring. In addition, we provide a comprehensive survey of key deep learning techniques for fish habitat monitoring including classification, counting, localisation, and segmentation. Furthermore, we survey publicly available underwater fish datasets, and compare various DL techniques in the underwater fish monitoring domains. We also discuss some challenges and opportunities in the emerging field of deep learning for fish habitat processing. This paper is written to serve as a tutorial for marine scientists who would like to grasp a high-level understanding of DL, develop it for their applications by following our step-by-step tutorial, and see how it is evolving to facilitate their research efforts. At the same time, it is suitable for computer scientists who would like to survey state-of-the-art DL-based methodologies for fish habitat monitoring.

1. Introduction

Proper understanding of our planet and its ecosystems is not possible unless suitable tools are developed to explore and learn about our largest ecosystem, the marine environment. Computer Vision (CV) technology through deployment of its underwater cameras can help us better comprehend and manage remote marine fish habitats. However, due to the sheer volume of their visual data, manual processing is time- and cost-prohibitive, requiring a new radical shift in data analysis, through advanced technologies such as Deep Learning (DL).

DL is at the frontier of computer vision. Its deep neural network architectures are capable of learning complex mappings from high-dimensional data to interpretable feature representations, hence, DL has been successfully applied to various challenging computer vision tasks such as semantic image segmentation (Chuang et al., 2011; Jing et al., 2020; Laradji, Rodriguez, et al., 2021; Pathak et al., 2015; Qi et al., 0000), visual object detection (Kim et al., 2016; Pathak et al., 2018; Villon et al., 2016; Wang et al., 2018), and tracking (Duan & Deng, 2019; Garcia et al., 2016; Kang et al., 2018; Lumauag & Nava, 2019). These applications have the potential to radically alter the way we interact with the world through computers. Recently, the applications of DL and its underlying Deep Neural Networks (DNNs) for underwater visual processing have received significant attention (Chuang et al., 2016; Laradji, Saleh, et al., 2021; Mandal et al., 2018; Naseer

[*] Corresponding author at: College of Science and Engineering, James Cook University, 1 James Cook Drive, Townsville, 4811, QLD, Australia.
E-mail addresses: alzayat.saleh@my.jcu.edu.au (A. Saleh), marcus.sheaves@jcu.edu.au (M. Sheaves), dean.jerry@jcu.edu.au (D. Jerry), mostafa.rahimiazghadi@jcu.edu.au (M. Rahimi Azghadi).

https://doi.org/10.1016/j.eswa.2023.121841
Received 11 April 2022; Received in revised form 13 March 2023; Accepted 25 September 2023
Available online 1 October 2023

et al., 2020; Nilssen et al., 2017; Saleh et al., 2020a; Salman et al., 2020; Siddiqui et al., 2018; Villon et al., 2018).

The main advantage of deep learning is its ability to learn features in different data types, such as underwater fish images, through end-to-end training. Training of DNNs is often thought to be easy. Many frameworks take delight in providing few lines of code that solve some CV tasks, providing the misleading impression that all that is needed is then plug and play, using some general Application Programming Interfaces (APIs). In these APIs, the developers have lifted the burden from us and, in doing so, disguised the complexity behind a few lines of code needed to achieve the task at hand. The framework developers have achieved the purpose of "providing a few lines of code" but we, the end-users, have been fooled into believing we need to spend only a few hours learning the intricacies of the provided APIs.

However, when it comes to training a DL algorithm, things become more complicated. The task of training a DNN is actually as complicated as the problem it is intended to solve. In fish monitoring, for example, the number of input images you use, how you pre-process your images, how you build your models, how you fine-tune the model (using dropout or regularisation, for example), how you extract the features, how you combine them to produce final predictions, what metric you use to report your model performance, and your choice of which layer to extract features from to feed to your classifier, are among some of the many variables to consider when training a DNN. You can include any number of variations on these factors to further optimise your model and achieve the best possible accuracy.

Due to the above intricacies, most of the time DNNs are not simply an "off-the-shelf" technology that works with all kinds of datasets, even those similar to the one that has been meticulously customised for it. The fact that training a customised high-performance DNN is rigorous and challenging is now widely accepted. However, this challenging process can be facilitated by being patient, paying attention to details, and working systematically. Developing customised DNNs with a specific application, for example, for underwater fish monitoring, should follow the same systematic steps of developing any other computer vision applications (e.g. detection of vehicles in traffic). The only difference lies in the type of data being fed to the DNN.

In this paper, we first present a tutorial that covers the background of DL to help understand the above-mentioned common DL terminologies. The tutorial also provides a comprehensive overview of the essential systematic steps to help better develop a supervised DL model, with a focus on underwater fish habitat monitoring.

In the second part of the paper, we survey state-of-the-art research and development on the use of DL for fish monitoring. We synthesise the literature into four main categories covering the common CV tasks of classification, counting, localisation, and segmentation of fish images. We investigate different deep learning architectures and their performance. We also survey publicly available underwater fish image datasets. Finally, we provide a comprehensive overview of the challenges in applying DL to marine fish monitoring domains. We also draw a roadmap for future research works.

Although a number of previous relevant review articles (Goodwin et al., 2022; Li & Du, 2021; Li et al., 2021; Moniruzzaman et al., 2017; Saleh, Sheaves, & Rahimi Azghadi, 2022; Yang, Liu, et al., 2021; Zhao et al., 2021) exist, our paper has a different approach and motivation that compliments prior surveys. Compared to Goodwin et al. (2022), which provides a survey of the general domain of ecological data analysis, covering a wide array of studies on plankton, fish, marine mammals, pollution, and nutrient cycling, we focus only on fish monitoring. We also provide a detailed analysis of fish datasets and comprehensively review the literature on four key tasks in underwater fish video and image processing. This detailed analysis and review are not provided in Goodwin et al. (2022), or any of the previous works, making our paper useful for readers who would like to study fish monitoring using DL in more detail and depth, while seeing a comprehensive literature review.

In addition, Li and Du (2021) provides a review of studies on fish condition, growth, and behaviour monitoring in aquaculture settings. It briefly covers and reviews various DL architectures and their aquaculture applications, unlike the present communication that is focused mainly on Convolutional Neural Network (CNN) and provides a detailed survey and analysis of the underwater fish monitoring literature.

The work presented in Zhao et al. (2021) covers the general domain of Machine Learning, as opposed to the specific domain of DL in our paper. This is done for aquaculture applications as wide as fish biomass and behaviour analysis to water quality predictions, while also briefly covering and reviewing fish classification and detection methods.

A survey of computer vision models for fish detection and behaviour analysis in digital aquaculture is provided in Yang, Liu, et al. (2021). An interested reader should study Yang, Liu, et al. (2021) before reading our paper, due to the background technical details provided on image acquisition, which are key to developing effective DL datasets and models, as we discussed in our paper.

Furthermore, the DL-based studies presented in Li et al. (2021), Moniruzzaman et al. (2017) are mainly around the two specific tasks of underwater fish tracking, and underwater object detection, respectively. These applications are different to our study. However, since our underwater fish monitoring task are related to these applications, our paper can complement these works.

In Saleh, Sheaves, and Rahimi Azghadi (2022), we have provided a historical survey of fish classification methods between the years 2003–2021. These methods cover traditional CV techniques and modern DL methods, only for fish classification in underwater habitats and not for the general domain of underwater fish habitat monitoring.

This paper covers the use of deep learning in underwater fish monitoring. Section 2 covers the basics of deep learning, including neural networks, convolutional neural networks, and supervised learning. Section 3 provides an overview of the development process of deep learning models, from training to deployment. Section 4 discusses the applications of deep learning in underwater fish monitoring, including classification, counting, localisation, and segmentation. Section 5 discusses the advantages and disadvantages of the application of DL to fish habit monitoring. Section 6 explores the challenges of underwater fish monitoring, such as environmental factors, model generalisation, and limitations of available datasets. Section 7 presents potential opportunities for deep learning in underwater fish monitoring, including knowledge distillation, merging image data from multiple sources, automatic fish phenotyping, and visual monitoring of fish behaviour and movements. Finally, Section 8 summarises the study's main findings and provides concluding remarks.

2. Deep learning

This section discusses the basics of deep learning (Saleh, Sheaves, & Rahimi Azghadi, 2022), a sub-field of machine learning, and its utilisation of multi-layered neural networks to automatically learn input features. It also introduces Convolutional Neural Networks (CNNs) and their efficient learning of deep features for image processing, making them suitable for underwater fish monitoring (Saleh, Sheaves, & Rahimi Azghadi, 2022).

2.1. Neural networks

Neural networks are a type of computational model that are inspired by the structure and function of biological neural systems in animals. They consist of basic processing units called neurons that take input signals, apply a function to them, and produce an output. In a neural network, the neurons are organised into layers, with each layer performing a specific type of computation. The layers are typically arranged in a hierarchical fashion, with the input layer receiving raw data and the output layer producing the final result.

The activation function of a neuron is the mathematical function that determines whether or not the neuron "fires" or produces an output signal based on the input signals it receives. One common activation function is the sigmoid function, which is a non-linear function that maps the input to a value between 0 and 1. This function is useful for classification tasks, such as image classification, where the output of the neuron can be interpreted as a probability.

Bias nodes are another important component in neural networks. These nodes are like neurons, but they do not receive input signals. Instead, they have a fixed input value of 1 and a weight associated with them. The bias value is added to the sum of the input-weight products to increase the flexibility of the model. In other words, bias nodes allow the neural network to adjust the output even when all input features are equal to zero.

Different types of loss functions are used for different types of tasks. For classification tasks, such as image classification, the cross-entropy loss is a common choice. This loss function measures the difference between the predicted probability of the correct class and the actual probability. Hinge loss is another type of loss function that is commonly used for classification tasks, where the correct class score should be higher than the sum of the scores for all other classes by some margin.

Regularisation is a technique used in neural network learning to prevent overfitting by discouraging complex mapping functions or models. This technique involves adding a regularisation term to the general model loss function, which takes into account the loss function value for all the training dataset examples. The two most common forms of regularisation are L1 and L2, with L2 being the sum of the square of the weights, and L1 being the sum of the weights.

2.1.1. Optimisation

In supervised learning, the learning task can be reduced to an optimisation problem in the form of

$$\theta^* = \arg\min_\theta g(\theta), \tag{1}$$

where θ is a parameter vector, at which the loss function $g(\theta)$ that usually represents the average loss for all training examples, reaches its minimum. g can be represented as

$$g(\theta) = \frac{1}{n}\sum_{i=1}^{n} L\left(f_\theta\left(x_i\right), y_i\right), \tag{2}$$

where (x_i, y_i) represents a (input, desired output) training pair.

Similarly, in DL, an optimisation method is used to train the neural network by minimising the error function E that is defined as

$$E(W, b) = \sum_{i=1}^{m} L\left(\hat{y}_i, y_i\right) \tag{3}$$

where W and b are the weights and biases of the network, respectively. The value of the error function E is thus the sum of the mean squared loss L between the predicted value $\hat{y}$ and true value y, for m training examples. The value of $\hat{y}$ is obtained during the forward propagation step and makes use of the previously-mentioned weights and biases of the network, which can be initialised in different ways. Optimisation minimises the value of the error function E by updating the values of the trainable parameters W and b. The error function E is usually minimised by using its gradient slopes for the parameters. The most commonly used optimisation method is *Gradient Descent* (Sun et al., 2019), in which the gradient is optimised by calculating a matrix of partial derivatives (computed using backpropagation, as detailed in the next subsection). These derivatives provide the slope of g simultaneously at each dimension of θ. Therefore, the gradient-based optimiser is used to iteratively update the network weights in the direction of the steepest descent of the loss function, with the aim of reducing the training loss to as low a value as possible. This is achieved by subtracting a small quantity from each weight in the direction of the negative gradient of the loss function. While the ultimate goal is to find a good local minimum of the loss function, the non-convexity of the loss function makes it difficult to search for the global optimum directly. Instead, the optimiser seeks to improve the network's performance on the training data, while also ensuring that the validation loss remains low, which indicates that the network is generalising well to new data.

2.1.2. Backpropagation

Backpropagation is probably the most important part of learning in neural networks. It is performed after a forward propagation or pass, in which a subset of the training dataset (named a batch) $\left\{\left(x_i, y_i\right)\right\}_{i=1}^{m}$ and the current network parameters θ are used to calculate the final layer output and the loss. During the forward pass, the data input is passed to the first layer to process according to its activation function and their values are passed on to the next layer, hence the term "forward pass". After the forward pass and calculating the final layer loss, backpropagation happens, through which we start to calculate the loss backwards, layer by layer, and the layer derivatives are then "chained" by the local gradients to minimise the overall loss, g.

Overall, neural networks are a powerful and flexible tool for a wide range of machine learning tasks, and their components, including neurons, activation functions, bias nodes, and loss functions, are essential to their success.

2.2. Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are a type of Deep Neural Network that are particularly powerful for computer vision tasks. They work by applying a convolution (filtering) operation on the input data through several convolution layers. This extracts useful features from the input data by sliding convolution filters across the input image represented to the network as matrices. One of the first and most successful examples of CNNs in computer vision was AlexNet proposed in 2012 (Krizhevsky et al., 2012) . Since then, many different variations of CNNs have been proposed, revolutionising image processing in different domains.

A typical CNN architecture consists of convolutional layers, pooling layers, non-linear activation layers, and final output layers, as shown in Fig. 1. The building blocks and layers of a typical CNN include Convolutional Layers, Batch Normalisation, Activation Layer, Pooling Layer, Dropout, and Fully Connected Layers. Convolutional Layers apply a filtering operation on input matrix data to generate a feature map. Batch Normalisation is used to normalise the learning of the network across the current set of training data to improve the speed of learning and convergence of the deep learning model. Activation Layers increase the non-linearity of the convolutional layer output to learn complex data. Pooling Layers reduce the size of the feature map and improve the efficiency of computation. Dropout is used to avoid overfitting the training data. Fully Connected Layers contain a small number of neurons and are the second-last layer of a CNN, before the output layer.

2.3. Supervised learning

There are two main approaches to learning in general DL. These include unsupervised and supervised learning. Unsupervised learning is often used to discover the structure and composition of the input and output domains without explicit and supervised target domain. This approach enables generalisation from one input domain to another by transforming data representations that are not directly related to the data distribution of target domain.

The supervised learning approach, on the other hand, is designed to explicitly map data from the input domain to its output domain via training pairs that exhibit matching representations. These pairs are carefully crafted by a human (supervisor), hence the name. The training process of supervised learning can suffer from instability and is less

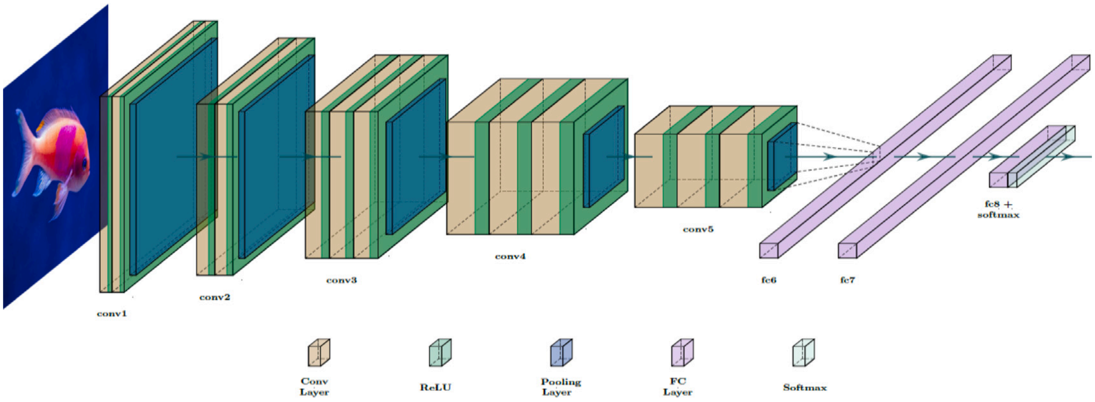

Fig. 1. Schematic diagram of a CNN architecture used for the classification of fish images. The architecture consists of five convolutional layers that include the batch norm operation within them, followed by pooling layers (conv1–conv5). In this model, the feature maps from convolutional layers are pooled through pooling layers and then flattened through two fully connected layers (fc6 and fc7). The classification output is the result of a fully connected layer and a softmax activation layer (fc8+softmax).

effective than the unsupervised learning method, because it learns with an accurate target distribution without domain-specific knowledge.

Supervised deep learning uses a subtle deep neural network mechanism to extract useful features from large amounts of input training data that are labelled to show their desired output domain. The learning is done by using the repetitive backpropagation process (Rojas & Rojas, 1996) explained earlier, to adjust the DL architecture parameters (such as weights and biases) while keeping fixed its hyperparameters (such as the shape, number, and size of convolutional, pooling, and fully connected layers). The goal is to optimise the function f, which maps the input domain X to the output domain Y. While the architecture of the network is typically fixed during training, the optimiser adjusts the internal parameters of the network to achieve the best possible mapping of the input training data to their desired output.

2.4. Deep learning and fish monitoring

One of the applications of DL is fish monitoring, which is the process of observing and measuring fish populations and their habitats. Fish monitoring is important for understanding the ecology and biodiversity of aquatic ecosystems, as well as for managing fisheries and aquaculture. DL can help with fish monitoring by providing accurate and efficient methods for fish classification, detection, counting, tracking, behaviour analysis, health assessment, and so on. DL can also handle complex underwater environments that pose challenges for traditional image processing techniques, such as low visibility, noise, distortion, illumination variation, etc.

The data for DL-based fish monitoring can come from various sources, such as underwater cameras, sonar, drones, satellites, etc. The data can be collected in different scenarios, such as shallow or deep water, fresh or marine water, natural or artificial habitats, etc. To improve the performance and robustness of DL-based fish monitoring systems, domain knowledge such as fish biology, ecology, and aquaculture management can be integrated and other technologies can be combined with DL algorithms (Li & Du, 2021). These include hardware technologies: such as sensors, communication devices, storage devices, etc. Software technologies: such as data augmentation, feature extraction, model optimisation, etc.

The goals and effects of applying DL to fish monitoring are manifold (Yang, Zhang, et al., 2021). Some examples of these goals are:

- Enhancing scientific understanding of aquatic ecosystems and their dynamics
- Improving fisheries management and conservation by providing reliable data on fish stocks and their distribution.
- Increasing aquaculture productivity and profitability by optimising feeding strategies, reducing disease outbreaks, and preventing escapes and predation.
- Reducing human intervention and labour costs by automating fish monitoring tasks.
- Promoting public awareness and education on aquatic biodiversity and sustainability.

In summary, DL is a promising technique for fish monitoring that can provide automated solutions for various tasks related to fish identification, measurement, localisation, and segmentation. By combining DL with other technologies such as sonar or drones, fish monitoring can be performed more effectively and efficiently in different underwater environments.

3. Developing deep learning models

A comprehensive overview of the essential systematic steps for training a DL model is summarised in Fig. 2. Even though these steps are general in DL training, we included useful tips arising from our experience in developing DL applications in various domains from medical imaging to marine science applications. Nevertheless, we put an emphasis on the development of DL for underwater fish habitat monitoring.

3.1. Training dataset

The available training data is essential for developing an efficient DL model. Datasets are becoming increasingly crucial, even more so than algorithms. Perhaps, the most important factor when considering a supervised learning dataset is its size. The requirement for a large training dataset to achieve high accuracy is often a big obstacle. Because visual algorithms are trained by pairs of images and labels, in a supervised manner, they can only identify what has already been given to them. As a result, depending on the project, the number of objects to identify, and the required performance, training datasets might contain hundreds to millions of images. However, smaller training datasets with only a few hundred samples per class may also achieve good results (Konovalov, Saleh, Bradley, et al., 2019; Konovalov et al., 2018; Konovalov, Saleh, Efremova, et al., 2019; Saleh et al., 2020b). Nevertheless, the larger the training dataset, the greater the recognition accuracy.

Because of the scarcity of datasets and the difficulty of acquiring reliable data, approaches for boosting the accuracy rate from small samples will inevitably become a focus of future studies. The problem of limited sample data can be also alleviated by transfer learning (Lee et al., 2018; Mathur et al., 2020; Molchanov et al., 2016). Furthermore, data augmentation will become increasingly critical. Section 6.3 covers some challenges of limited data and some approaches to address these challenges.

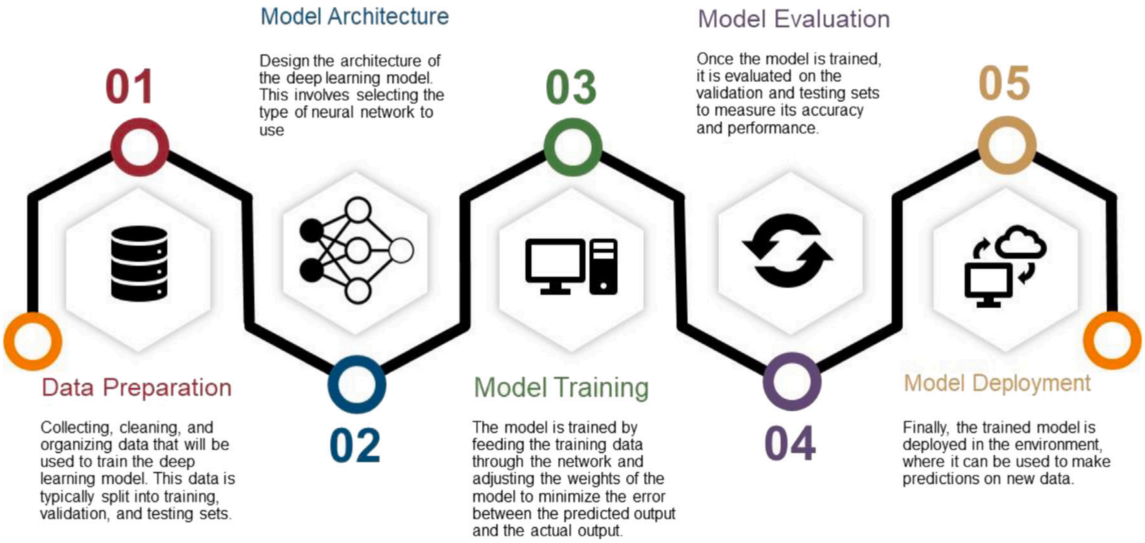
Model Architecture
Design the architecture of the deep learning model. This involves selecting the type of neural network to use
Model Evaluation
Once the model is trained, it is evaluated on the validation and testing sets to measure its accuracy and performance.
01
03
05
Data Preparation
Collecting, cleaning, and organizing data that will be used to train the deep learning model. This data is typically split into training, validation, and testing sets.
02
Model Training
The model is trained by feeding the training data through the network and adjusting the weights of the model to minimize the error between the predicted output and the actual output.
04
Model Deployment
Finally, the trained model is deployed in the environment, where it can be used to make predictions on new data.

Fig. 2. A schematic diagram showing the steps and components required for training a deep learning model.

The second factor to consider when preparing a dataset for DL training is having a balance. This is critical to ensure that each class to be identified contains a sufficient number of instances to minimise class imbalance biases. These biases happen when the DL favours one or more classes due to seeing them more often when being trained.

Also, the training dataset is typically divided into two subsets, the training subset for efficiently training the model and the validation/test subset for assessing the trained model's performance. For the training subset, a subset of the training dataset is reserved for training the model. If the training subset is too large, it can prolong the model training. If, on the other hand, the training subset is too small, the resulting model may not generalise well to unseen inputs. The validation/test subset is typically used to avoid overfitting, which is a common problem in machine learning and happens when the developed model simply memorises the inputs rather than properly learning them. Cross-validation is another widely used methodology for testing a DL model's training performance, by splitting the training dataset into multiple mutually exclusive subsets of training and testing data. One method of cross-validation is called $k-fold$ cross-validation, in which the training dataset is split into k equally sized subsets. In this method, $k-1$ folds are used for training the model, while the remaining fold is used to test the learning performance. This process is repeated until all the folds have been used once as a test/validation set.

In addition to the above, it is usually vital to, initially and before embarking on code development, perform a comprehensive inspection of the dataset. This will help to clean the dataset, for instance by finding and removing duplicate data instances. It also helps identify imbalances and biases, as well as data distribution, trends, or outliers, which will help in better model design and understanding of possible wrong DNN predictions.

Fortunately, in the domain of fish habitat monitoring, researchers currently have access to a variety of datasets. Table 1 lists publicly available underwater fish datasets, their sources, and where to get them, in addition to a summary of their features, their labels, and their sizes. The main point to note about these datasets is that they differ in both size and number of features. Although the number of these fish datasets is still small (17), the diversity of aquatic species they cover is already quite wide. They cover a large number of aquatic species, as indicated in Fig. 3. Moreover, each dataset features a different number of images that have varying resolutions. For each image, there is also a ground truth annotated by a human expert, which make them very useful. For instance, these datasets can be used by researchers to test their DL models or to pre-train them, as the first step, for their more specific fish monitoring tasks.

After preparing the training dataset or utilising alternative approaches to addressing insufficient data challenge, one can start developing their DL model using a machine-learning development framework.

3.2. Development framework

The rapid evolution of DL has led to the creation of a vast number of development libraries and packages that enable the setting up of DNNs with insignificant effort. Usability and availability of resources, architectural support, customisability, and hardware support are all various benefits of using existing machine-learning frameworks. The most commonly used frameworks are PyTorch, TensorFlow, MATLAB, Microsoft Cognitive Toolkit (CNTK) and Apache MXNET. In the context of DL for marine research, as will be shown later in Tables 3 to 5, PyTorch and TensorFlow are the dominant frameworks, while Matlab and Caffe have been used only in a few works. Overall, details such as the project needs and the programmer and developer preference should be taken into account, when choosing the development framework.

When the development framework is chosen, the next step is to find the most suitable network architecture for the task at hand. This sometimes depends on the framework, as some recent methods may not immediately be supported by all frameworks.

3.3. Network architecture

Network architecture is the structure of the DL model, which depends on what it intends to achieve and its expected input and output. Therefore, the type of training dataset and the expected outcome influence the architecture's choice and its performance. DL network architectures can differ in a variety of ways such as the type and number of layers, their structure, and their order. Before selecting a network architecture, it is critical to understand the dataset you have and the task you are going to complete. For example, convolutional neural networks or CNNs are known to learn higher-order features, such as colours and shapes, from data within their convolution layers. Therefore, they are ideally adapted to image-based object recognition. On the other hand, Recurrent Neural Networks (RNNs) have the capability of processing temporal information or sequential data, such as the order of words in a sentence. This feature is ideal for tasks such as handwriting or speech recognition.

In the context of fish habitat monitoring, if you are working on a task that requires you to learn temporal information of the input

Table 1
Summary of some publicly available datasets containing fish for training and testing deep learning models.

Dataset	Summary	Labels	Dataset size	Website
A - Deepfish	Videos from coastal habitats in north-eastern and western Australia	fish/no fish	40k classification labels, 3.2k images with point-level annotations, 310 segmentation masks	github.com/alzayats/DeepFish
B - Croatian Fish Dataset	12 species of fish found in Croatian waters	species names	794 classification labels	www.inf-cv.uni-jena.de/fine_grained_recognition.html#datasets
C - Fish in seagrass habitats	RUV taken in Australian seagrass habitat of 2 species	species	9k classification labels, bounding boxes and segmentation masks	github.com/globalwetlands/luderick-seagrass
D - Fish4Knowledge	Fish detection and tracking dataset, 17 videos at 10 min long, rate of 5 fps.	fish/no fish	3.5k bounding boxes	groups.inf.ed.ac.uk/f4k/index.html
E - Fish-Pak	Image dataset of 6 different fish species from 3 locations in Pakistan	species	1k classification labels	data.mendeley.com/datasets/n3ydw29sbz/3
F - Labelled Fishes in the Wild	Rockfish (Sebastes spp.) and other species (non-fish) near the seabed	fish/non-fish	1k bounding boxes (fish), 3k (non-fish)	swfscdata.nmfs.noaa.gov/labeled-fishes-in-the-wild/
G - OzFish	Large data set comprising of 507 species of fish.	species, fish/no fish	80k labelled cropped images, 45k bounding box annotations (fish/no fish)	github.com/open-AIMS/ozfish
H - QUT Fish Dataset	468 species in varying ex-situ and in-situ habitats.	species name	4k classification images	www.dropbox.com/s/e2xya1pzr2tm9xr/QUT_fish_data.zip?dl=0
I - Whale Shark ID	543 individual whale sharks (Rhincodon typus)	individuals	7.8k bounding boxes	http://lila.science/datasets/whale-shark-id
J - Large Scale Fish Dataset	9 different seafood types collected from a supermarket in Izmir, Turkey	species name	For each class, there are 1000 augmented images and their pair-wise augmented ground truths	www.kaggle.com/crowww/a-large-scale-fish-dataset
K - NCFM	Image dataset of 8 different fish species	species name	~16000 classification images	www.kaggle.com/c/the-nature-conservancy-fisheries-monitoring/data
L - Mugil liza sonar	Sonar-based underwater videos of schools of migratory mullets (Mugil liza)	number of fish	500 counting images	zenodo.org/record/4751942#.YKzfUKgzayk
M - MSRB Dataset	Real underwater images without marine snow and synthesised with marine snow	NA	~6000 images	github.com/ychtanaka/marine-snow
N - WildFish	1,000 fish categories	species name	~54000 classification images	github.com/PeiqinZhuang/WildFish
O - SUIM	Image dataset of 8 different underwater objects	object name	~1500 annotated images semantic segmentation mask	github.com/xahidbuffon/SUIM
P - DZPeru fish-datasets	Several species in varying ex-situ and in-situ habitats.	species name	~17000 annotated images segmentation mask	github.com/DZPeru/fish-datasets
Q - LifeCLEF	10 different fish species	species name	~1000 annotated videos	www.imageclef.org/

sequence, for example fish image sequence analysis, the DL architecture you choose can be very important. For example, a CNNs architecture is more suited for *image-based* object recognition such as fish classification, while the RNN architecture is more suitable for tasks where the input sequence is temporal in nature such as generating fish habitat descriptions.

To find a suitable architecture, you first need to define your problem. This problem is defined by two questions: (1) What features will you extract? (2) How will you label these features? The features you extract are defined by your data. In other words, you are interested in the representation of the data you have. The number of features you choose to extract is defined by the task you are trying to solve. As described above, the DL architectures can learn features such as colours and shapes from image-based object recognition. Before trying to construct your network, you first need to decide what data type you will use and how will you encode the information. After you have defined your task, you should think about what features are important for the task. You will need to define this in order to construct your network. For example, if the features you want to extract are fish shape and fish location, then you could define a convolutional architecture. The features you choose to define should be a subset of all the features in the data. For example, for an image-based object recognition network, you would extract features such as fish species. However, your extracted features will also need to cover all the data. For example, you will also need features of the type of water or the type of background. It is important to take all these features into account when defining your network. For a complete discussion on different DL architectures see Khan et al. (2020).

3.4. Network model

When a general network architecture is selected, the next step is to select, or sometimes develop, a network model of that architecture. For instance, when you decided to use a CNN, you can use different varieties of CNN models. The rule of thumb for selecting a CNN is to choose a model that results in a satisfactory training loss for your dataset. Creating an exotic and creative model is not recommended at this stage. It is usually recommended to avoid the temptation and choose a model big enough to overfit your dataset, and then regularise it properly to improve the validation loss.

For example, one may pick a well-known CNN model, e.g. ResNet, which can be used out-of-the-box, if their task is simple, e.g. fish classification. In later stages, they can customise their model to adequately capture their dataset. We show in Tables 3 to 5 in the next section that ResNet is the most commonly used model for fish counting (Table 3), fish localisation (Table 4), and fish segmentation (Table 5).

3.5. Training the model

After choosing the best model is time to set up a full train/validation pipeline. The below steps are recommended at this stage of development.

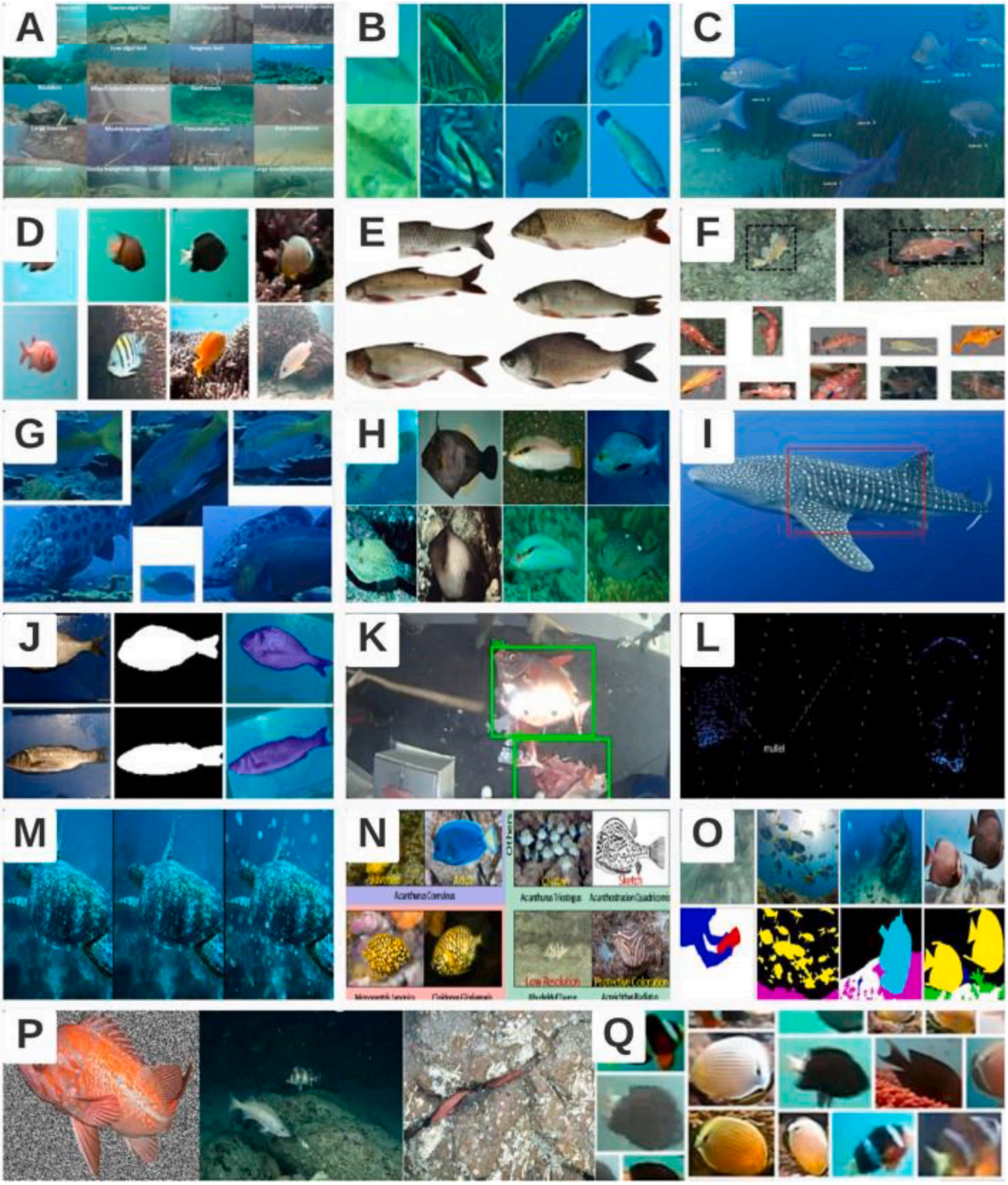

Fig. 3. Sample images from publicly available datasets detailed in Table 1.

- Start with a simple model (i.e. a small number of convolutional layers) that can hardly go wrong and visualise the model performance metrics. Do not use an out-of-the-box large model like ResNet, just yet. It is recommended to plot training loss to see how the network is progressing during learning and if the loss is getting smaller. This also shows the speed of learning.
- To better understand the process, it is recommended to use a fixed random seed (for randomly initialising the network parameters) to ensure that the same results can be achieved when running the code twice.
- Do not perform any data augmentation at this stage as it may introduce errors. You can do data augmentation at a later stage after confirming that your network works properly. You can see a brief introduction to data augmentation and other methods at Section 6.2.
- Use ADAM algorithm (Kingma & Ba, 2014), which helps the learning by applying adaptive optimisation to the learning rate of the network.
- The learning rate is an important hyperparameter of a deep learning model. It is usually the most crucial value during training and should be configured using trial and error. Depending on the size of your dataset, a specific learning rate decay may be needed. The learning rate decay is a technique that allows the learning rate to fall during successive training epochs until it

Table 2
Performance metrics used to compare various surveyed works.

Performance Metric	Symbol used	Description
Classification Accuracy	CA	The percentage of correct predictions. For multi-class classification, CA is averaged among all the classes. $CA = (TP + TN)/(TP + TN + FP + FN)$
Precision	P	The fraction of true positives (TP), to the sum of TP and false positives (FP). $P = TP/(TP + FP)$
Recall	R	The fraction of true positives (TP) to the sum of TP and false negatives (FN). $R = TP/(TP + FN)$
F1 score	F1	The harmonic mean of precision and recall. $F1 = 2 \times (P \times R)/(P + R)$
Mean Square Error	MSE	Mean of the square of the errors between predicted and observed values
Root Mean Square Error	RMSE	Is the square root of the mean of the square of all of the errors.
Mean Relative Error	MRE	The mean error between predicted and observed values, in percentage
L2 error	L2	Root of the squares of the sums of the differences between predicted counts and the actual counts
Intersection over Union	IoU	A metric that evaluates how similar the predicted bounding box is to the ground truth bounding box. by dividing the area of overlap between the predicted and the ground truth boxes, by the area of their union.
The maximum number	MaxN	MaxN, the maximum number of the target species in any one frame.
Mean average precision	mAP	Depending on the detection difficulty, the mean AP across all classes and/or total IoU thresholds are used.
Classification Error	CE	Is how often is the classifier incorrect and also known as “Misclassification Rate”. $CE = (FP + FN)/(TP + TN + FP + FN)$

converges. A high learning rate at the start prevents the network from memorising noisy data, whereas decaying the learning rate improves complex pattern learning.

- Implement early stopping and monitor the learning process by looking at the training loss plot to prevent overfitting.
- Add complexity to your model gradually, e.g. add more layers or use off-the-shelf CNN models, and obtain a performance improvement over time.

3.6. Testing the model

When the model is trained, its accuracy and performance should be tested using the test subset of the training dataset. A test set can also be independent of the training dataset to evaluate the model performance. The main point to remember is that the test set should not have been used for the training or evaluation of the model, at all.

The model's performance should be measured by computing appropriate metrics suitable to the task at hand. A list of the most common metrics used in testing fish monitoring models is given in Table 2. For classification tasks, Classification Accuracy (CA), Precision and Recall rates are appropriate metrics, while F1-score, which is a combination of precision and recall, can provide a better measure of model performance and is used in fish counting and localisation tasks as shown in Tables 3 and 4. The Intersection-Over-Union (IoU) is the appropriate metric for segmentation tasks, while the mean average precision (mAP) metric suits pixel-wise localisation of fish in images. Looking at Tables 3 to 5, other metrics such as Mean Square Error (MSE) and Root MSE (RMSE) have also been used in the marine fish monitoring literature. These can be considered and used if required.

3.7. Fine tuning the model

The performance and accuracy of the model could be improved if needed. The amount of this improvement is, though, strongly influenced by its current accuracy. This step may quickly become complicated, since increasing the model accuracy might require several steps such as adjusting the learning rate, collecting new data, or fully modifying the model's architecture. You should keep this fine-tuning step to a reasonable level. Otherwise, the model might overfit the data.

3.8. Deploying the model

Finally, the model deployment mode should be chosen. This depends on the application and the deployment requirements. The model can be deployed to run on a local or remote device (on a web server, a docker container, a virtual private server (VPS), etc.). This will determine whether the results can be accessed remotely or only within the local network. It is recommended to use a cross-platform deployment method to avoid issues such as input/output data format, or the type of files used for storing data.

The most commonly used cross-platform model deployment method is Docker (Abdul et al., 2019; Potdar et al., 2020), which is a virtualisation software that allows setting up and running other software environments on top of a base Linux distribution without the need to set-up virtual machines. Docker helps build, configure, and run applications using the same Docker file. Typically, Docker is the recommended approach for web applications. In this method, you can use Docker container or Docker host on your development machine. Docker container may be the easiest option for web applications. You can also deploy your network to a remote machine via Docker. The advantage of using a container is that you can share the development environment and run tests of your model using multiple docker containers. You can also install the Docker tool on your local machine to manage containers, so it is convenient.

4. Applications of deep learning in underwater fish monitoring

Deep learning has been widely used in marine environments with applications spanning from deep-sea mineral exploration (Juliani & Juliani, 2021) to automatic vessel detection (Chen et al., 2019). However, we confine the scope of this paper to only marine fish image processing, which typically includes four tasks of classification, counting, localisation, and segmentation of underwater fish images, as shown in Fig. 4.

Here, the goal is to assist the reader in understanding the similarities and differences across these tasks and their relevant DL models and techniques. We provide a background of what each task involves, what previous works have been published towards addressing it using deep learning, and synthesise the literature on each task.

4.1. Classification

As its name infers, in visual processing, classification is the task of classifying images into different categories. There can be only two categories, i.e. a binary classification, in which the images are classified into two groups, e.g. “fish” and “no fish”, depending on the presence or absence of fish in an image (e.g. Deepfish dataset described in the first row of Table 1). The classification can also involve multiple “classes”

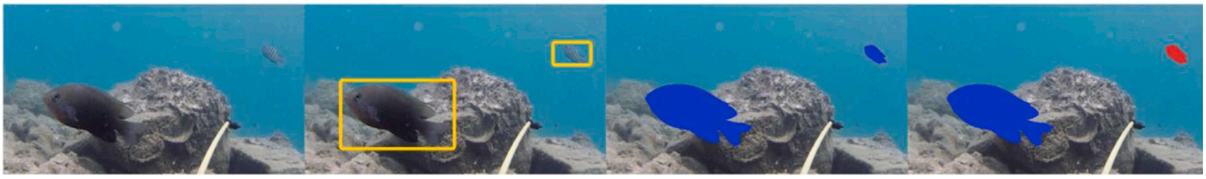

Fig. 4. Illustration of four typical fish monitoring tasks. From left: Fish Classification (*i.e.* is there a fish in the image, or what type (class) of fish is in the image?); Fish Detection/Localisation/Counting; Fish Semantic Segmentation, and Fish Instance Segmentation.

or groups. For instance, consider assigning different underwater fish images into different groups based on the species (e.g. FishPak dataset in Table 1) present in them.

Consider a manual procedure, in which images in a dataset are compared and relative ones are classified based on similar features, but without necessarily knowing what you are searching for in advance. This is a difficult assignment as there could be thousands of images in the dataset. Moreover, many image classification tasks involve images of different objects. It rapidly becomes clear that an automatic system, such as a DNN, is required to complete this task quickly and efficiently.

Classification is the most widely-used and -studied underwater image processing task using DL. In a previous work, we have covered the use of DNNs specifically for the task of underwater fish classification. We refer the reader to Saleh, Sheaves, and Rahimi Azghadi (2022) for a comprehensive review of prior art on classification.

4.2. Counting

The purpose of the counting task is to predict the number of objects existing in an image or video. Object counting is a key part of the workflow in many major CV applications, such as traffic monitoring (Khazukov et al., 2020; Zhang et al., 2017). In the context of marine applications and fish monitoring, counting may be used to map distinct species and monitor fish populations for effective conservation. With the use of commercially available underwater cameras, data gathering can be done more comprehensively. It is, however, difficult to correctly count fish in underwater habitats. To perform effective counting, models must understand the diversity of the items in terms of posture, shape, dimension, and features, which makes them complex. Meanwhile, manual counting is very time-consuming, costly, and prone to human error.

DL affords a faster, less expensive, and more accurate alternative to the manual data processing methods currently employed to monitor and analyse fish counts. Table 3 lists several of the recent DL techniques used for fish counting. Saleh et al. (2020a) created a novel large-scale dataset of fish from 20 underwater habitats. They used Fully Convolutional Networks (FCNs) for several monitoring tasks including fish counting and reported a Mean Average Error (MAE) of 0.38%. DL has the potential to be a more accurate method for assessing fish abundance than humans, with results that are stable and transferable between survey locations. Ditria et al. (2021), Ditria, Lopez-Marcano, et al. (2020), Ditria, Sievers, et al. (2020) compared the accuracy and speed of DL algorithms for estimating fish population in underwater pictures and video recordings to human counterparts in order to test their efficacy and usability. In single image test datasets, a DL method performed 7.1% better than human marine specialists and 13.4% better than citizen scientists. For video datasets, DL was better by 1.5% and 7.8% compared to marine and citizen scientists, respectively.

Despite this high potential, DL has not been thoroughly investigated for counting underwater fish. One possible reason for the lack of comprehensive research on fish counting is the scarcity of large publicly available underwater fish datasets. In addition, properly annotating fish datasets to train robust DL models is time-prohibitive and expensive. Although underwater fish counting is limited in the literature, several previous works have advanced the field in this area. For instance, Tarling et al. (2021) created a novel dataset of sonar video footage of mullet fish labelled manually with point annotations and developed a density-based DL model to count fish from sonar images. They counted fish by using a regression method (Xue et al., 2016) and achieved a MAE of 0.30%. Other researchers (Liu et al., 2018; Schneider & Zhuang, 2020) used sonar images as well because they present substantially different visual characteristics compared to natural images. Counting fish in sonar images, however, is substantially different from counting fish in underwater video surveillance (Mandal et al., 2018). Unlike natural images, sonar images present unique visual characteristics and are in lower resolution due to the specific imaging forming principle.

Using DL, a computer can be taught to identify fish in underwater images, thus eliminating the subjectivity of humans in counting fish. However, its use for fish population and count analysis is dependent on the model performance on a set of well-defined performance metrics and parameters, which is in itself a challenge. In Section 3, we discussed how one can train high-performance DL models, how the use of the current DL pipeline (and other methodologies) can be improved, and how future DL models can be designed for better assessing fish population including their abundance and their location, which is the subject of the next subsection.

4.3. Localisation

Object localisation is an essential task in CV, where the goal is to locate all instances of specified objects (e.g. fish, aquatic plants and coral reef) in images. Marine scientists assess the relative abundance of fish species in their environments regularly and track population variations. Various CV-based fish sample methods in underwater videos have been offered as an alternative to this tedious manual assessment. Though, there is no perfect method for automated fish localisation. This is mostly owing to the difficulties that underwater videos bring, such as illumination fluctuations, fish movements, vibrant backgrounds, shape deformations, and a variety of fish species.

To address these issues, several research works have been carried out, which are listed in Table 4. Saleh et al. (2020a) have developed a fully convolutional neural network that performs localising of fish in realistic fish-habitat images with high accuracy. Jalal et al. (2020) introduced a hybrid method based on motion-based feature extraction that combines optical flow (Beauchemin & Barron, 1995) and Gaussian mixture models (Zivkovic & van der Heijden, 2006) with the YOLO deep learning technique (Chaudhari et al., 2020) to identify and categorise fish in unconstrained underwater videos using temporal information. They achieved fish detection F-scores of 95.47% and 91.2% on LifeCLEF 2015 benchmark (Joly et al., 2014) and their own dataset, respectively. Gaussian mixture is an unsupervised generative modelling approach that may be used to learn first and second-order statistical estimates of input data features (Zivkovic & van der Heijden, 2006). Within an overall population, this is used to indicate Normally Distributed subpopulations. The weakness of Gaussian mixture is when trained on videos with some fish but no pure background, the fish are modelled as background as well, resulting in misdetections in subsequent video frames (Salman et al., 2019). In order to compensate for the Gaussian mixture's weakness, optical flow can be used to extract features that are solely caused by underwater video motion. The pattern of apparent motion of objects, surfaces, and edges in a visual scene

Table 3
Summary of recent DL research works performing the task of fish counting.

Article	DL model	Framework	Data	Annotation/Pre-processing/Augmentation	Classes and labels	Perf. metric	Metric value	Comparisons with other methods
A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis (Saleh et al., 2020a)	ResNet-50 CNN	Pytorch	Authors-created database containing 39,766 images for 20 habitats from remote coastal marine environments of tropical Australia and split to sub-dataset for four computer vision tasks: classification, counting, localisation, and segmentation.	Each image was annotated by point-level and semantic segmentation labels	20 classes of 20 different fish habitat.	MAE	0.38	NA
Annotated Video Footage for Automated Identification and Counting of Fish in Unconstrained Seagrass Habitats (Ditria et al., 2021)	ResNet-50 CNN	Pytorch	The dataset consists of 4,281 images and 9,429 annotations (9,304 luderick, 125 bream) at the standard high resolution (1920 x 1080 p).	Each image was annotated by drawing a bounding box and segmentation mask	2 classes of fish	F1	92%	NA
Automating the Analysis of Fish Abundance Using Object Detection Optimising Animal Ecology With Deep Learning (Ditria, Lopez-Marcano, et al., 2020)	Mask R-CNN ResNet50	Pytorch	Authors-created database containing 6,080 fish images from 20 habitats from Tweed River Estuary in southeast Queensland	Each image was annotated by segmentation mask	1 class of fish	F1	Image (95.4%) Video (86.8%)	The computer's performance in determining abundance was 7.1% better than human marine experts and 13.4% better than citizen scientists in single image test datasets, and 1.5% and 7.8% higher in video datasets, respectively.
Deep learning for automated analysis of fish abundance: the benefits of training across multiple habitats (Ditria, Sievers, et al., 2020)	Mask R-CNN ResNet50	Pytorch	Authors created five datasets, each consisting of 4700 annotated luderick, total of 23500 images	Each image was annotated by drawing a Polygonal segmentation masks around the region of interest (ROI)	1 fish class	F1	87%–92%	NA
Deep learning with self-supervision and uncertainty regularisation to count fish in underwater images (Tarling et al., 2021)	ResNet50 CNN	Tensorflow	Authors created a data set of 500 labelled sonar images from video sequences	Each image was annotated by dot annotation	3 classes of fish according to number of fish	MAE	0.30%	Comparison between DeepFish dataset 0.38% and authors' benchmark result and their model 0.30%.
Counting Fish and Dolphins in Sonar Images Using Deep Learning (Schneider & Zhuang, 2020)	CNN	NA	Authors created a data set of 143 labelled sonar images from the Amazon River	Each image was annotated by counting number of fishes	35 classes for fish and 4 for dolphin	MSE	Fish 2.11% Dolphins 0.133%	Comparing four Network Architectures, DenseNet201, InceptionNetV2, Xception, and MobileNetV2
Counting Fish in Sonar Images (Liu et al., 2018)	CNN	NA	Authors created a dataset of 537 labelled sonar images from video sequences	Each image was annotated by dot annotation	1 class of fish	RMSE	16.48%	Comparison with other state-of-the-art approaches
Assessing fish abundance from underwater video using deep neural networks (Mandal et al., 2018)	Faster R-CNN	Caffe	Authors created a dataset of 4909 labelled images from video sequences	Each image was annotated by drawing a bounding box	50 classes from 50 Different fish habitat.	mAP	82.4%	NA

generated by the relative motion of an observer and a scene is known as optic flow (Beauchemin & Barron, 1995).

Knausgård et al. (2021) also implemented YOLO (Chaudhari et al., 2020) for fish localisation. To overcome their small training samples, they employed transfer learning (explained in the next Section). The YOLO technique achieved Mean Average Precision (mAP) of 86.96% on the Fish4Knowledge dataset (Giordano et al., 2016). YOLO-based object detection systems have been also used in several other research to robustly localise and count fish (Jalal et al., 2020; Knausgård et al., 2021; Xu & Matzner, 2018). To test how well Yolo could generalise to new datasets, Xu and Matzner (2018) used it to localise fish in underwater video using three very different datasets. The model was trained using examples from only two of the datasets and then tested on examples from all three datasets. However, the resulting model could not recognise fish in the dataset that was not part of the training set.

Other CNN models have also been adapted to robustly detect fish under a variety of benthic background and illumination conditions. For instance, Villon et al. (2016) and Choi (2015) used GoogLeNet (Szegedy, Liu, et al., 2015), while Labao and Naval (2019a) used an ensemble of Region-based Convolutional Neural Networks (Ren et al., 2015) that are linked in a cascade structure by Long Short-Term Memory networks (Hochreiter & Schmidhuber, 1997). In addition, Inception (Szegedy, Vanhoucke, et al., 2015) and ResNet-50 (He et al., 2015) were examined in Zhuang et al. (2017) for fish detection and recognition based on weakly-labelled images. Furthermore, Han et al. (2020) and Li et al. (2015) used Fast R-CNN (Region-based Convolutional Neural Network) (Ren et al., 2015) to detect and count fish.

Table 4 demonstrates that state-of-the-art methods (e.g. YOLO and Fast R-CNN) can achieve high accuracy in localisation tasks. These methods generally train object detectors from a wide variety of training images (Felzenszwalb et al., 2010; Girshick et al., 2014) in a fully supervised manner. The drawback is that these models depend on instance-level annotations, e.g. tight bounding boxes need to be drawn around fish in training datasets. This is time-consuming and labour-intensive and makes the use of DL in marine research very challenging, if not impossible. In Section 6.3.4 we discuss how this critical issue can be addressed using weakly supervised localisation of objects, where only binary image-level labels showing the existence or absence of an object type are needed for training.

Similar to fish classification, counting, and localisation, fish segmentation, i.e. detecting the entire body of fish in an image is a critical task

Table 4
Summary of recent DL research works performing the task of fish localisation.

Article	DL model	Framework	Data	Annotation/Pre-processing/Augmentation	Classes and labels	Perf. metric	Metric value	Comparisons with other methods
Marine Animal Detection and Recognition with Advanced Deep Learning Models (Zhuang et al., 2017)	ResNet-10 CNN	NA	The dataset is made of 73 videos from the public datasets Fish4Knowledge	Each image was annotated by drawing a bounding box	1 class of fish	F1	0.07%	NA
Fish detection and species classification in underwater environments using deep learning with temporal information (Jalal et al., 2020)	Yolo - CNN	TensorFlow	The dataset is made of two datasets 93 videos from LifeCLEF 2015 fish dataset And an authors-created database containing 4418 videos	Each image was annotated by drawing a bounding box and species name	15 classes of 15 different fish species.	F1	LCF-15 95.47% UWA 91.2%	Comparison with other state-of-the-art approaches
Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system (Salman et al., 2019)	ResNet-152 CNN	TensorFlow	The dataset is made of 110 videos from two public datasets Fish4Knowledge and LifeCLEF 2015 fish dataset	Each image was annotated by drawing a bounding box	15 classes of 15 different fish species.	F1	87.44% and 80.02% respectively	NA
Temperate fish detection and classification: a deep learning based approach (Knausgård et al., 2021)	YoloV3 - CNN	Pytorch	total of 27230 images catalogued into 23 different species from the public datasets Fish4Knowledge	Each image was annotated by drawing a bounding box	23 classes of 23 different fish species.	mAP	86.96%	NA
Underwater Fish Detection Using Deep Learning for Water Power Applications (Xu & Matzner, 2018)	YoloV3 - CNN	Keras - TensorFlow	Authors-created database of underwater video sequences for a total of 70000 train/test frame	Each image was annotated by drawing a bounding box	3 classes of fish	mAP	54.74%	NA
Coral Reef Fish Detection and Recognition in Underwater Videos by Supervised Machine Learning: Comparison Between Deep Learning and HOG+SVM Methods (Villon et al., 2016)	GoogLeNet CNN	NA	Authors-created database containing 13000 fish thumbnails from videos	Each image was annotated by drawing a bounding box	11 classes of 8 different fish species.	F1	98%	Compare HOG+SVM With Deep Learning
Fish identification in underwater video with deep convolutional neural network (Choi, 2015)	GoogLeNet CNN	NA	20 videos from LifeCLEF 2015 fish dataset	Each image was annotated by drawing a bounding box	15 classes of 15 different fish species.	AP	81%	NA
Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild (Labao & Naval, 2019a)	RNN-LSTM	NA	Authors-created database containing 18 underwater video sequences for a total of 327 train/test frame	Each image was annotated by drawing a bounding box and species name	1 class of fish	F1	67.76%	Comparison with R-CNN Baseline
A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis (Saleh et al., 2020a)	ResNet-50 CNN	Pytorch	Authors-created database containing 39,766 images for 20 habitats from remote coastal marine environments of tropical Australia and split to sub-dataset for classification, counting, localisation, and segmentation.	Each image was annotated by point-level and semantic segmentation labels	20 classes of 20 different fish habitat.	MAE	0.38	NA
Marine Organism Detection and Classification from Underwater Vision Based on the Deep CNN Method (Han et al., 2020)	VGG16-RCNN	NA	The dataset is obtained from the video provided by the Underwater Robot Picking Contest, test set contains 8800 images.	Each image was annotated by drawing a bounding box	3 classes of fish	mAP	91.2%	NA

in marine research and applications. In the next subsection, we discuss how DL can be used to perform fish segmentation and how it is useful in marine research.

4.4. Segmentation

Semantic segmentation task is to predict a label from a set of predefined object classes for each pixel in an image (Shelhamer et al., 2017). In the context of marine research, fish segmentation provides a visual representation of fish contour, which might be helpful for human expert visual verification or to estimate fish size and weight. Table 5 lists a number of research addressing the task of fish segmentation.

Saleh et al. (2020a) developed a FCN model that performs fish Segmentation in realistic fish-habitat images with a high accuracy. Labao and Naval (2019b) proposed a DL model that can simultaneously localise fish, estimate bounding boxes around them and segment them using a unified multi-task CNN in underwater videos. Unlike previous approaches (Qian et al., 2016; Wang & Kanwar, 2021) that relied on motion information to identify fish body, their proposed method predicts fish object spatial coordinates and per-pixel segmentation using just video frames independent of motion information. Their suggested approach is more resilient to camera motions or jitters since it is not dependent on motion information, making it more suitable for processing underwater videos captured by Autonomous Underwater Vehicles (AUVs). Region Proposal Networks (RPN) (Ren et al., 2017) have been also used for fish segmentation in underwater videos (Alshdaifat et al., 2020). RPN is a FCN that generates boxes around identified objects and gives them confidence scores of belonging to a specific class, simultaneously.

Computational efficiency is essential in the autonomy pipeline of visually-guided underwater robots. For this reason, Islam et al. (2020) developed SUIM-Net, a fully-convolutional encoder–decoder model that balances the trade-off between performance and computational efficiency. On the other hand, for higher performance, Zhang et al.

Table 5
Summary of recent DL research works performing the task of fish segmentation.

Article	DL model	Framework	Data	Annotation/Pre-processing/Augmentation	Classes and labels	Perf. metric	Metric value	Comparisons with other methods
A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis (Saleh et al., 2020a)	ResNet-50 CNN	Pytorch	Authors-created database containing 39,766 images from 20 habitats from remote coastal marine environments of tropical Australia and split to sub-dataset for classification, counting and localisation, and segmentation.	Each image was annotated by point-level and semantic segmentation labels	20 classes of 20 Different fish habitat.	mIoU	0.93%	NA
Weakly supervised underwater fish segmentation using affinity LCFCN (Laradji, Saleh, et al., 2021)	ResNet-CNN	Pytorch	Public DeepFish dataset (Saleh et al., 2020b)	Each image was annotated by segmentation labels	20 classes of 20 Different fish habitat	mIoU	0.749%	NA
Simultaneous Localisation and Segmentation of Fish Objects Using Multi-task CNN and Dense CRF (Labao & Naval, 2019b)	ResNet-CNN	TensorFlow	Authors-created dataset containing 1525 images from ten 10 different sites in central Philippines	Each image was annotated by drawing a bounding box and segmentation labels	1 class of fish	AP	93.77%	NA
Semantic Segmentation of Underwater Imagery: Dataset and Benchmark (Islam et al., 2020)	VGG16-CNN	Keras - TensorFlow	Authors-created dataset containing 1525 images of 8 object categories	Each image was annotated by segmentation labels	8 classes of 8 different object categories.	mIoU	84.14%	NA
DPANet: Dual Pooling-aggregated Attention Network for fish segmentation (Zhang et al., 2022)	ResNet-50 CNN	Pytorch,	Two public datasets DeepFish (Saleh et al., 2020b) and SUIM (Islam et al., 2020)	Each image was annotated by segmentation labels	20 classes: 20 Different fish habitat.	mIoU	91.08%, 85.39%	Comparison with other state-of-the-art approaches
Weakly-Labelled semantic segmentation of fish objects in underwater videos using a deep residual network (Labao & Naval, 2017)	ResNet-FCN	TensorFlow	Authors-created dataset containing several underwater videos from six different sites in Verde Island Passage, Philippines.	Each image was annotated with weakly-labelled ground truth derived from a motion-based background subtraction (BGS)	1 class of fish	AP	65.91%	NA
Improved deep learning framework for fish segmentation in underwater videos (Alshdaifat et al., 2020)	ResNet-CNN	TensorFlow	Two datasets extracted from the Fish4Knowledge to produce 2000 frames	Each image was annotated by drawing a bounding box and segmentation labels	15 classes of 15 different fish species.	AP	95.20%	NA

(2022) proposed Dual Pooling-aggregated Attention Network (DPANet) to adaptively capture long-range dependencies through a computationally friendly manner to enhance feature representation and improve not only the segmentation performance, but also its computational resources and time.

All previously discussed models use fully-supervised methods that require a large amount of pixel-wise annotations, which is very time-consuming and expensive, because a human expert must segment and label, for example, each fish in an image. To overcome this serious issue, weakly-supervised semantic segmentation models are used. These models do not need to be trained with pixel-wise annotation (Rajchl et al., 2016). However, due to a lower level of supervision, training weakly-supervised semantic segmentation models is often a more challenging task. Applying weakly labelled ground truth derived from motion-based adaptive Mixture of Gaussians Background Subtraction, Labao and Naval (2017) managed to get an average precision of 65.91%, and an average recall of 83.99%. Recently, several other weakly-supervised methods have been introduced to overcome the cost of a large amount of pixel-wise annotations. These new methods include bounding boxes (Dai et al., 2015; Khoreva et al., 2017), scribbles (Lin et al., 2016), points (Bearman et al., 2016; Laradji, Saleh, et al., 2021), and even image-level annotation (Ahn & Kwak, 2018; Huang et al., 2018; Pathak et al., 2015; Wang et al., 2018; Wei et al., 2018). Since weakly-supervised methods are integral to the success of important DL-based segmentation tasks, in Section 6.3, we discuss them further.

In the previous subsections, we discussed how DL is useful in a number of key applications in fish habitat monitoring. In the following Section, we discuss the many challenges on the way of developing DL models for such applications.

4.5. Acoustic and sonar data

Acoustic and sonar data are valuable sources of information for monitoring fish habitats and behaviours. Acoustic methods use sound waves to detect, identify, and quantify fish in various aquatic environments (Mancusi et al., 2022). Sonar systems emit sound pulses and receive echoes from objects in the water, such as fish. By analysing the characteristics of the echoes, such as frequency, intensity, and shape, sonar systems can provide information about fish size, shape, orientation, density, and movement.

Acoustic methods have several advantages over other techniques for fish monitoring, such as visual observation or net sampling (Muñoz et al., 2020). Acoustic methods can cover large areas and depths quickly and efficiently; they can operate in turbid or dark waters where visual methods are ineffective; they can provide continuous data over long periods of time; they can minimise disturbance to fish and their habitats; and they can be integrated with other sensors or platforms for multidisciplinary studies.

Acoustic and sonar data can be combined with other technologies such as GPS and environmental sensors to provide a more complete picture of fish behaviour and their habitat. For example, the combination of acoustic and sonar data with GPS allows researchers to track fish movements and habitat use, while the integration of environmental sensors can provide information on water temperature, salinity, and other important environmental factors that may influence fish behaviour.

One of the challenges of acoustic methods is to accurately classify fish species based on their acoustic signatures (Benoit-Bird & Lawson, 2016). Different species may have similar acoustic characteristics due to their morphology or behaviour. Moreover, environmental factors such as noise, reverberation, or multipath effects may degrade the quality of the acoustic data. Therefore, advanced signal processing and machine learning techniques are needed to improve the performance of acoustic classification. In addition, the deployment of acoustic and sonar sensors in natural environments can be challenging and expensive, which may limit the availability of data for the training and validation of DL models (McCann et al., 2018).

Acoustic and sonar data combined with DL techniques offer a powerful tool for monitoring fish habitats and behaviours in a non-invasive and efficient way (Zhou et al., 2022). By using this tool, fisheries scientists and managers can gain insights into fish ecology, distribution, abundance, migration patterns etc., which can help them make informed decisions for sustainable fisheries management.

4.6. Automatic fish phenotyping from underwater images

Automatic fish phenotyping, i.e. extracting their weight, size, and length, in their natural habitats can provide invaluable information in better understanding marine ecosystems and fish ecology (Goodwin et al., 2022). Although many studies have addressed fish monitoring in aquaculture and fish farm settings (Li & Du, 2021; Zhao et al., 2021), monitoring fish for measurement in natural habitats remain mostly unexplored, and can be investigated in future research. This research should address problems such as low visibility and light, fish occlusion and overlap, which are shared with aquaculture monitoring. However, other problems unique to natural habitats such as cluttered background environments and underwater distance measurement should be addressed too. One study addresses fish species identification in an underwater video for marine monitoring applications, using a hierarchical CNN model that incorporates targeted data augmentation techniques (Ben Tamou et al., 2022). Automated imaging has also been used to obtain phenotypic data on growth and body colour (Fu & Yuna, 2022).

4.7. Visual monitoring of fish behaviour and movements

Although some telemetry and satellite tracking devices can be used in limited settings (Lennox et al., 2017), fish monitoring in their natural habitats over a period of time is not achievable using these techniques mainly due to the hostile underwater signal communication medium (Jahanbakht et al., 2021). For instance for tracking fish movements, schooling, and behaviour, new visual monitoring techniques should be devised. A possible direction for future studies is to devise a better understanding of fish vision characteristics (Boudhane & Nsiri, 2016) and their implications in the current and next generation of automated DL-based tracking systems (Li et al., 2021) and marine object detection (Moniruzzaman et al., 2017). An example of an alternative tracking method is presented in Zhao et al. (2019), where the image-based identification and tracking method for fish is designed based on biological water quality monitoring. To improve the fish tracking task, some techniques can also be combined with visual image enhancement algorithms. For instance, when the image enhancement methods are used, the underwater images can be corrected for distortion and noise, and the fish tracking task can be easily performed. In Saberioon and Cisar (2016), the authors studied the potential of underwater fish monitoring by using visual and underwater sensing methods.

Another challenging research area is developing novel underwater fish tracking algorithms, using DL or other technologies, with low power consumption and real-time speed. For this, various hardware technologies and techniques used in other domains such as biomedical applications (Azghadi et al., 2020) can be explored. Of course, any automated vision-based tracking system should be validated through real-world trials, which is a significant undertaking requiring many resources, in order to ensure the accurate and real-time tracking of fish.

There have been several recent studies on the visual monitoring of fish behaviour and movements. For example, some studies surveyed the application of computer vision technology in analysing fish behaviour and fish monitoring (Li et al., 2022; Niu et al., 2018; Zhou et al., 2019). Another study demonstrated an integrated object detection and tracking pipeline as a noninvasive and reliable approach to studying fish behaviour by tracking their movement under field conditions (Lopez et al., 2021). Another study explores how fish behaviour can be used as a proxy to measure the physiological states of fish under different environmental stressors, such as pollutants, temperature changes, and social interactions (Fu et al., 2022).

5. Advantages and disadvantages of the application of DL to fish habit monitoring

Deep learning has been applied to various fields, including fish habitat monitoring (Saleh, Sheaves, & Rahimi Azghadi, 2022). The application of DL in fish habitat monitoring has several advantages that make it an attractive option for researchers and practitioners. One of the main advantages is its ability to handle complex data. DL models can learn complex patterns and relationships in the data, making them ideal for analysing large datasets with numerous variables (Ditria, Sievers, et al., 2020). This ability is particularly useful in fish habitat monitoring, where numerous variables such as water temperature, dissolved oxygen, and water quality can influence the fish's behaviour and habitat.

Another advantage of using DL in fish habitat monitoring is the potential to automate the monitoring process. Traditional fish monitoring methods involve manual data collection and analysis, which can be time-consuming, labour-intensive, and expensive. With DL, data can be automatically collected and analysed in real time, allowing for faster and more efficient monitoring. This automation can also reduce the likelihood of human error, leading to more accurate and reliable results.

However, the application of DL to fish habitat monitoring also has some disadvantages. One of the main disadvantages is the need for large amounts of high-quality data to train the DL models effectively. The quality of the data can have a significant impact on the performance of the model, and the lack of high-quality data can lead to inaccurate results.

Another disadvantage is the complexity of the DL models themselves. DL models are often complex and difficult to interpret, making it challenging to understand how the model arrived at its conclusions. This lack of transparency can make it difficult for researchers and practitioners to verify the accuracy of the model's results.

In addition, DL models require significant computing power and storage, which can be expensive and require specialised infrastructure. This requirement can be a barrier for some researchers and practitioners who do not have access to the necessary resources.

Overall, while the application of DL to fish habitat monitoring has several advantages, it also has some drawbacks that need to be considered. To maximise the benefits of DL in fish habitat monitoring, it is crucial to address these challenges and develop strategies to overcome them.

6. Challenges in underwater fish monitoring

Underwater fish monitoring presents a series of challenges for DL, which have been the focus of many research works. In this section, we first introduce the major environmental challenges faced when developing underwater fish monitoring models. We then show that one of the approaches to properly address these environmental challenges is to use DL. However, DL training for fish monitoring has its own challenges, which will be discussed in detail.

6.1. Environmental challenges

In order to work in underwater environments, monitoring models must be able to recognise objects and scenes in complex, non-trivial backgrounds. This presents both a challenge in the development and training of these models and in robustly testing them. The main environmental challenges in underwater visual fish monitoring can be categorised as follows:

1. The environment is noisy including very large lighting variation. An object viewed from a distance is much less bright than a close-up object. These problems become more acute when the background is not uniform.

2. Underwater scenes are highly dynamic, i.e. the scene's content and objects change very quickly. The background can change from being completely occluded to being visible and vice versa.
3. Depth and distance perception can be incorrect due to refraction. This is more severe for short distances.
4. Images are affected by water turbidity, light scattering, shading, and multiple scattering.
5. The image data are frequently under-sampled due to low-resolution cameras and power constraints underwater.

One of the main approaches used in literature to address these challenges is for the monitoring models to use hand-crafted features (Chuang et al., 2016; Fouad et al., 2014; Hossain et al., 2016; Hu et al., 2012; Huang et al., 2014; Islam et al., 2019; Ogunlana et al., 2015; Rova et al., 2007; Wang et al., 2017). Hand-crafted features are defined by a human to describe a fish image. For example, a low-level feature can be the histogram of a texture or a Gabor filter response. As a more complex and representative feature, a mid-level feature can be a Scale-Invariant Feature Transform (SIFT) (Lindeberg, 2012), or a Histogram of Oriented Gradient (HOG) (Dalal & Triggs, 2005). However, human-defined features cannot be applied to other datasets, and the definition of a human-defined feature is a time-consuming task, which restricts real-time detection and requires manual effort. Moreover, hand-crafted features are limited by human experiences, which may contain noise and are difficult to design. For example, a SIFT descriptor does not work well with lighting changes and blur.

Therefore, a fish image is transformed into a feature space that a computer can understand. The feature space is often based on a combination of low-level image features (for example, colour distribution and gradient), and other features in the image such as edges, shapes, and textures. Models using hand-crafted features, however, do not perform well under varying environmental conditions, and the feature space cannot be easily or robustly created. Additionally, the features created are too low-level and cannot be easily used for processing images from different sources.

An alternative way to build prediction models capable of working in the presence of these significant environmental challenges is to use DNNs. However, training effective DNNs require resolving some other challenges, which we discuss in the below subsections. We also describe some of the approaches in literature addressing them. The reviewed approaches in addressing these common challenges can provide a quick reference for future researchers developing DL-based fish monitoring models.

6.2. Model generalisation

Improving the generalisation abilities of DNNs is one of the most difficult tasks in DL. Generalisation refers to the gap between a model's performance on previously observed data (i.e training data) and data it has never seen before (i.e testing data). This is a fundamental problem, with implications for any applications using deep neural networks to process image data, videos, etc. This challenge is even more pronounced when more difficult tasks such as fish recognition in underwater environments.

Generalisation problem happens usually because during training the network over-fits to the training data. In other words, the weights of the network are adapted to produce a response that is best suited for reproducing the training examples. During testing, the network produces a response that is a compromise between the different training examples. This mismatch is a common cause of poor performance on test data, which is often referred to as a network over-fitting to the training data, even when the network has been trained for many epochs. The reason it occurs is that the network "memorises" the training data during the training. The training data can become quite large, consisting of hundreds of thousands or millions of examples. This makes the issue of network over-fitting quite significant. In the last few years, there have been significant research efforts towards solving the problem of over-fitting to improve model generalisation.

Previous works have shown that it is possible to prevent the network from over-fitting using techniques called regularisation (Kukačka et al., 2017). There are also some theoretical techniques to make the network more robust to training data. Below, we provide a brief overview of some of these techniques and how they have been applied to solve the problem of deep network over-fitting to training data, to improve generalisation in DL.

- *Regularisation Term*: It is hypothesised that neural networks with fewer weight matrices can result in simpler models with the same capability as the complete model. A regularisation term is, therefore, added to the model loss function to remove some of the weight matrices components. The most popular methods of regularisation are L1 and L2. For example, Tarling et al. (2021) showed that incorporating uncertainty regularisation improves performance of their multi-task network with ResNet-50 (He et al., 2015) backend to count fish in underwater images.
- *Batch normalisation*: Introduced in Section 2.2 as part of the convolutional layer in CNNs, batch normalisation was first introduced by Ioffe and Szegedy (2015) to decrease the effect of internal covariate shift. Internal covariate shift is the shift in the mean and covariance of inputs and network parameters across a batch of examples. Internal covariate shift can impede the training of deep neural networks. Batch normalisation is used in almost any DL model training, to improve the model generalisation. In the fish monitoring domain, for instance, Islam et al. (2020) proposed an optional residual skip block consisting of three convolutional layers with batch normalisation and ReLU non-linearity after each convolutional layer to perform effective semantic segmentation of underwater imagery.
- *Dropout*: Introduced in Section 2.2 as a common operation in CNNs, dropout reduces the network dependency on a small selection of neurons and encourages more useful and robust properties and features of the dataset to be learnt. When working with a complex neural network structure, dropout is frequently recommended to introduce additional randomisation, which helps with the generalisation capability of the network. For example, Iqbal et al. (2021) claimed that the inclusion of dropout layer has enhanced the overall performance of their proposed model for automatic fish classification.

6.3. Dataset limitation

Preparing training datasets is one of the central and most time-consuming bottlenecks in developing DL models, which require a large amount of data, e.g. a variety of underwater fish images in different environmental conditions, which should also be labelled and analysed by humans for supervised learning. Due to these requirements, making a large dataset is most of the time, very challenging, which makes the datasets limited and small. However, when compared with DL models trained with a large dataset, the convergence speed and training accuracy of the models trained with small datasets are much lower. Generally, increasing the size of training datasets by adding more data to them is the classic way to accelerate the training and improve the accuracy of DL models, but it is expensive. Therefore, in recent years, researchers have tackled the dataset limitation challenge by devising new ways described below.

6.3.1. Data augmentation

Data augmentation is a technique to increase the number of labelled examples required for DL training. It artificially enlarges the original training dataset by introducing various transformations such as translation, rotation, scaling, and even noise, to the original data instances, to make new instances. It is particularly relevant to the challenge posed when the quantity or quality of labelled data is insufficient to train a DL model. At the same time, data augmentation can be used to reduce the probability of overfitting and increase model generalisability. In contrast to the techniques listed above for improving model generalisation, data Augmentation addresses overfitting from the source of the problem (*i.e.* the original dataset). This is done under the notion that augmentations can extract additional information from the original dataset by artificially increasing the size of the training dataset. It is also critical to consider data augmentation's "safety" (*i.e.* the possibility of misleading the network post-transformation). For example, rotation and horizontal flipping are typically safe data augmentation techniques for fish classification tasks (Saleh et al., 2020a; Sarigül & Avci, 2017) but not safe on digit classification tasks, due to the similarities between 6 and 9. A data augmentation technique is to use the super-resolution reconstruction method (Ledig et al., 2017) based on Generative Adversarial Network (GAN) (Goodfellow et al., 2014) to enlarge the dataset with high-quality images. This has been previously used to improve small-scale fine-grained fish classification (Qiu et al., 2018), and to increase the model's predictive performance (i.e. ability to generalise to new data) (Konovalov, Saleh, Bradley, et al., 2019) for underwater fish detection and automatic fish classification (Chen et al., 2018).

Using augmentation techniques such as cropping, flipping, colour changes, and random erasing together can result in enormously inflated dataset sizes. For example, Islam et al. (2020) used rotation, width shift, height shift, shear, zoom and horizontal flip for semantic segmentation of underwater imagery to significantly increase their dataset size. Another data augmentation technique used during training DL models are scale jittering, which has been used in Mandal et al. (2018) for assessing fish abundance in underwater videos. Gaussian filtering to blur images and different degrees of rotation for fish recognition in underwater-drone with a panoramic camera is another augmentation technique used in the marine monitoring domain (Meng et al., 2018).

However, augmentation is not always favourable, as it might lead to large overfitting in cases with very few data samples. As a result, it is critical to determine the best subset of augmentation techniques to train your DL model using a limited dataset.

6.3.2. Transfer learning

Transfer Learning is preserving information obtained while solving one problem, and transferring the learned knowledge to another similar problem. For instance, one may initially train a network on a large object dataset, such as ImageNet that includes 1000 different object classes, and then utilise the learned network parameters from that training as the initial learning parameters in a new classification task, e.g. fish classification. In most cases, just the weights in convolutional layers are transferred, rather than the complete network, including fully connected layers. This is extremely useful since many image datasets have low-level spatial features and properties that are better learnt in massive datasets. For example, Zurowietz and Nattkemper (2020) presented unsupervised knowledge transfer to use their limited amount of training data in order to avoid time-consuming annotation for object detection in marine environmental monitoring and exploration.

6.3.3. Hybrid features

DL architectures have demonstrated excellent capabilities in capturing semantic knowledge that is latent in image features. Handcrafted features, on the other hand, can provide specific physical descriptions if they are carefully chosen. In addition, attributes of natural images have been demonstrated to be described differently by CNN features and hand-crafted features. This means a feature's discriminative ability may behave differently on different datasets. Therefore, these two types of features may complement each other for better learning.

However, increasing feature dimensions by fusing hand-crafted and DL-generated features can result in increased computational requirements. One way to avoid this is to initially utilise DL features for a particular dataset, and later add hybrid features to enhance the performance. As a result, when working with difficult datasets, such as uncommon and rare marine species, more sophisticated algorithms and techniques based on hybrid features may be required. In fact, several research groups have used such strategies to improve the performance of marine species recognition tasks.

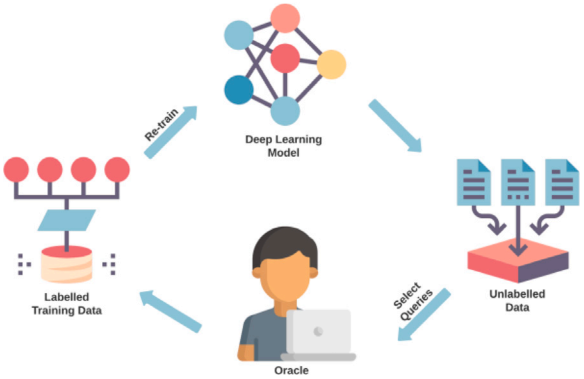

Fig. 5. Schematic diagram of Active Learning.

For instance, Mahmood et al. (2016) used texture- and colour-based hand-crafted features extracted from their CNN training data to complement generic CNN-extracted features and achieved a classification accuracy higher than when using only generic CNN features when classifying corals. A combination of CNN and hand-designed features have also been used in Cao et al. (2016) for marine animal classification, again showing that their method achieves higher accuracy than applying CNN alone. In another work, Blanchet et al. (2016) showed that aggregation of multiple features outperforms models using single feature-extraction techniques, for automated coral annotation in natural scenes.

6.3.4. Weakly-supervised learning

DL methods (LeCun et al., 2015) have consistently achieved state-of-the-art results in a variety of applications, specifically in fully supervised learning tasks like classification and regression (Li et al., 2009; Lin et al., 2014). Fully supervised learning methods create predictive algorithms by learning from a vast amount of training patterns, where each pattern has a label showing its ground-truth output (Kotsiantis, 2007). Although the current fully supervised methods have been very successful in certain activities (De Vos et al., 2017; Mader et al., 2018; Wörz & Rohr, 2006), they come with a caveat of requiring a large portion of the data to be labelled, and it is sometimes difficult or extremely time-consuming to obtain ground-truth labels for the dataset. Thus, it is desirable to develop learning algorithms that are able to work with less labelled data (*i.e.* weakly supervised) (Oquab et al., 2015; Zhou, 2018).

Weak supervision in particular can be very useful in underwater fish monitoring, where the limited dataset size and the time- and cost-prohibitive nature of labelling limits achieving a useful dataset for developing effective, smart, and automated habitat monitoring tools and techniques. A number of works in literature have already used weak supervision for underwater fish habitat monitoring. For example, Laradji et al. (2020) proposed a segmentation model that can efficiently train on underwater fish images, not manually segmented for training, but only labelled with simple point-level supervision. This work demonstrated that in the marine monitoring context, weakly-supervised learning can effectively improve the accuracy and speed of model development with limited dataset sizes and limited labelling budget.

6.3.5. Active learning

Active learning is a sub-field of Machine Learning (ML) and, more broadly, of AI. In active learning, the proposed algorithm is allowed to be “inquisitive”, that is, it is allowed to pick the data to learn, which in theory means the algorithm can do more with less guidance, similar to weak supervision. Active learning systems are seeking to solve the constraint of labelling by posing a questionnaire in the context of unlabelled examples to be labelled by an oracle (e.g. a human annotator). In this manner, the goal of the active learner is to attain high precision by using as few labelled examples as possible, thus minimising the expense of acquiring labelled data; see Fig. 5.

In many cases, the labels come for little or no cost, like the “spam” label that is used to mark spam emails, or the five-star rating that a user could post for a movie on a social networking platform. Learning methods use these labels and scores to help screen your spam email and recommend movies that you might enjoy. In these cases, certain labels are given free of charge, but for more sophisticated supervised learning tasks, such as when you need to segment a fish in an underwater environment, this is not the case. For example, in Nilssen et al. (2017) active learning has been used for the classification of species in underwater images from a fixed observatory. The authors proposed an active learning method that assigns taxonomic categories to single patches based on a set of human expert annotations, making use of cluster structures and relevance scores. This active learning method, compared to traditional sampling strategies, used significantly fewer manual labels to train a classifier.

6.3.6. Few-shot learning

The scarcity of rare species images in training datasets is one of the main limitations when addressing the automatic processing of wildlife images, especially in fish habitat monitoring. Such limitations lead researchers to explore few-shot learning.

Few-shot learning is another sub-field of ML. It is closely related to active learning since it aims to infer relationships between data from very few data samples. The central concept is how one can learn from a small number of examples and apply this knowledge to unlabelled data (Wang et al., 2021; Zhao, Jin, & Wang, 2021). For example, you want to do animal identification in wildlife camera trap image datasets. However, since you have only a few labelled examples of rare species, with only a few images in training datasets, you cannot train your model to recognise these animals because you only have a few examples. In this case, few-shot learning can be used to learn how to use the previously learned classifier to recognise other features of objects on the image (e.g. shape) that might help you complete the task. However, training on these new features should be done in a few-shot manner (Liu et al., 2019; Villon et al., 2022). The idea is to have a pre-trained model trained on a much larger dataset of different species. Then, once a new species appears in the dataset of unlabelled images, you can use this pre-trained model to find similarities between the new image and those that are already in the dataset and label those that are similar to the target species.

In a pioneering study of using few-shot learning in processing underwater videos, Villon et al. (2021) used it to discriminate 20 coral reef fish species with a range of training datasets from 1 image per class to 30 images per class. Few-shot object detection has been also used to localise wildlife using a camera trap in Feng and Xiao (2022). In another study, Feng and Li (2022) proposed a data augmentation method that applies constraints on the mixture of foreground and background images based on species distributions. Therefore, after training a convolutional neural network for species classification, the model can localise a new image to a species with the help of the species distribution constraints in the mixture of foreground and background images. Similar techniques can be used in addressing the scarcity of sample data for rare marine species in underwater videos.

6.3.7. Adaptive loss

The cross-entropy loss can be overwhelmed by the large class imbalance between foreground and background classes in the dataset during the training of dense detectors. This is because it is based on an implicit assumption of equal class priorities and does not differentiate between easy or hard examples. Therefore, Lin et al. (2017) proposed to use a weighted cross-entropy loss, which assigns higher weights to the loss of hard samples and down-weight easy examples, thus focusing the training on hard negatives. The adaptive focal loss $FL(p_t)$ is derived from the entropy loss.

$$\mathrm{FL}\left(p_{\mathrm{t}}\right) = -\alpha_{\mathrm{t}}\left(1 - p_{\mathrm{t}}\right)^{\gamma}\log\left(p_{\mathrm{t}}\right)$$

where α balances the importance of positive and negative examples, p_t is predicted probability, $\left(1 - p_{t}\right)^{\gamma}$ is a modulating factor to the cross-entropy loss, and γ is a tunable focusing parameter. It has been shown in Lin et al. (2017) that adaptive focal loss improves the accuracy compared to other losses for object detection on COCO test-dev (Lin et al., 2014).

In marine and fish habitat monitoring applications, it is very likely that strong class imbalance happens when datasets are being collected. This is mainly because the collected videos will have more examples of specific backgrounds such as coral reef, compared to various species of fish of interest. To address these issues, in addition to techniques such as adaptive loss mentioned above, other techniques developed for dealing with the problem of long-tailed distribution of training data can be explored and adopted. These include techniques such as those proposed in Cui et al. (2019) where the authors proposed a class-balanced loss to re-weight loss inversely with the adequate number of samples per class, or by replacing the standard cross-entropy in Cao et al. (2019) with label-distribution-aware margin loss.

6.4. Biodiversity challenges

In a recent article, Villon et al. (2022) have discussed some challenges beyond dataset limitation, focusing on biodiversity and how it can affect the deep learning-based automatic monitoring of marine and fish habitats through computer vision. Specifically, they consider the implications of three major universal rules of biodiversity, i.e. the distribution of species abundance, species rarity, and ecosystem openness (Villon et al., 2022). The authors discuss how these rules bring about three main issues affecting the performance of deep learning algorithms for underwater monitoring. They also discuss promising solutions to these issues, some of which were already discussed in Section 6.3. Due to the importance of these issues and the challenges they pose to fish habitat monitoring, we briefly discuss them here. However, the reader is encouraged to refer to Villon et al. (2022) for further details.

The first issue discussed is the imbalance of long-tail datasets, which is due to the abundance of some species in the collected videos and datasets, while some other groups may only be represented occasionally. Similarly, the second discussed issue is scarce data due to species rarity, which is a prominent biodiversity issue. Both the “long-tail datasets” and the "scarce data" issues, can cause a classifier to overfit the majority classes and fail to detect or predict the minority classes (Cui et al., 2019). One way to tackle this issue is by data augmentation (see Section 6.3.1) or Few-shot learning (see Section 6.3.6). The other way is in the training algorithms itself by modifying the loss function with respect to dataset imbalances (see Section 6.3.7).

The third challenge discussed is the “open world” issue that deals with an open ecosystem creatures. This results in the challenge of always having a new species that the “closed world” application is not trained on. This leads the model to misclassify the known species especially when the goal is to detect and predict marine species at sea. Villon et al. (2022) discuss open-set learning as a way of solving such a problem. The objective of an open-set recognition model is to classify all samples belonging to the training dataset correctly while allowing it to ignore all samples of the novel classes (Bendale & Boult, 2016).

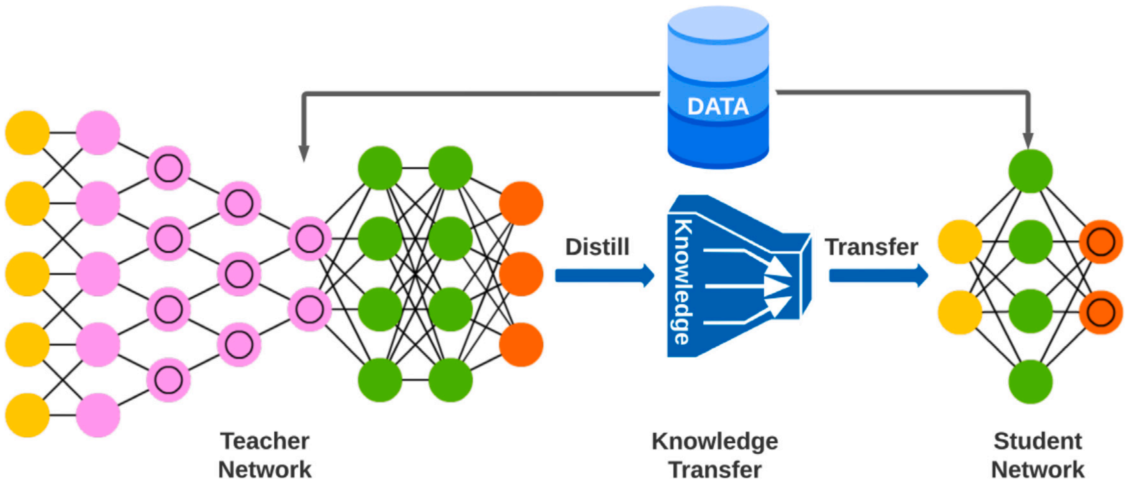

Fig. 6. Schematic diagram of knowledge distillation.

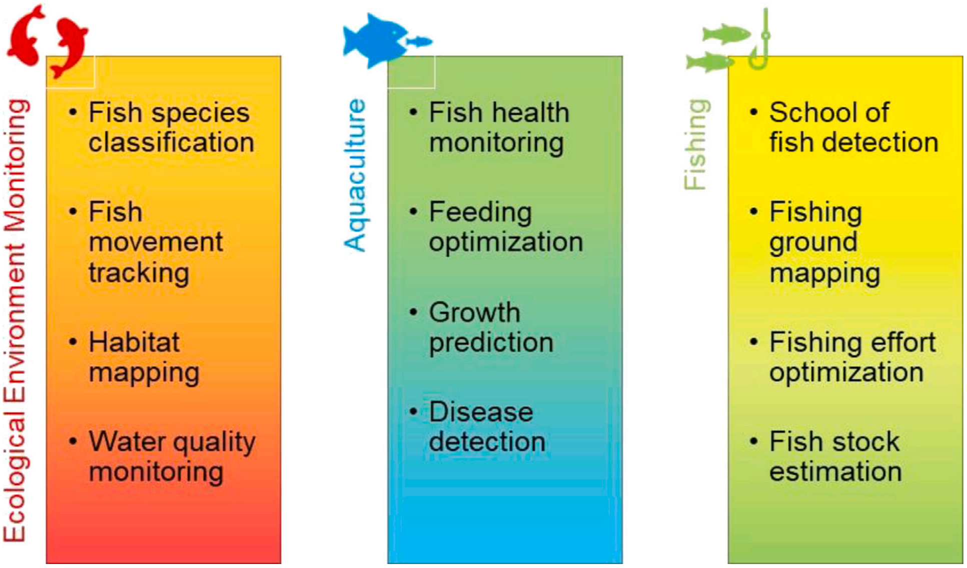

Fig. 7. Application scenarios for deep learning in underwater fish monitoring, including ecological environment monitoring, aquaculture, and fishing. Deep learning can be used to classify fish species, track their movement patterns, monitor fish health, optimise feeding schedules, and identify schools of fish for more sustainable fishing practices.

7. Opportunities in applications of DL to underwater fish monitoring

New methodologies and strategies should be developed to advance DL models for various underwater visual monitoring applications, including fish monitoring, and to bring them closer to their terrestrial monitoring equivalents. In a previous study that was focused on the task of fish classification (Saleh, Sheaves, & Rahimi Azghadi, 2022), we have discussed some of the future research opportunities including (i) utilising Spatio-temporal data to add space and time domain information to the current training algorithms that mainly learn fish images regardless of their spatial and/or temporal correlation; (ii) Developing efficient and compact DL models that can be deployed underwater for real-time parsing of the fish images at the collection edge; (iii) Combining image data from multiple collection platforms for improved multi-faceted learning; and (iv) Automated fish measurement and monitoring from underwater captured images. Fig. 7 shows application scenarios for deep learning in underwater fish monitoring, including ecological environment monitoring, aquaculture, and fishing, have been identified. Deep learning can be used to classify fish species, track their movement patterns, monitor fish health, optimise feeding schedules, and identify schools of fish for more sustainable fishing practices. In addition to the opportunities discussed in Saleh, Sheaves, and Rahimi Azghadi (2022), further research areas could include (i) Developing DL models that can handle a wider range of image quality and visibility conditions, such as those encountered in murky or low-light environments; (ii) Combining visual monitoring with other sensor modalities such as acoustic sensing to improve detection and tracking accuracy; and (iii) Developing robust data labelling and annotation methods for large-scale training datasets, which can be difficult to obtain in underwater environments.

7.1. Knowledge distillation for underwater embedded and edge processing

DL models used for fish monitoring applications are usually very large containing millions of parameters and requiring extensive computational power. To deploy these models on resource-limited devices and in resource-constrained environments such as undersea monitoring sites, different hardware-enabled compression techniques such as quantising and binarising DNN parameters (Lammie et al., 2019) can be used, as discussed in Saleh, Sheaves, and Rahimi Azghadi (2022). Another method that has seen a lot of interest and attention for compressing large-scale DL models is knowledge distillation.

Knowledge distillation is a technique for training a student (*i.e.* a small network) to emulate a teacher (*i.e.* ensemble of networks), as shown in Fig. 6. The primary assumption is that in order to achieve a competitive or even superior performance, the student model should imitate the teacher model. The main issue is, however, transferring the knowledge from a large teacher to a smaller student. To that end, Bucila et al. (2006) proposed model compression as a way to transfer knowledge from a large model into a small model without sacrificing accuracy. In addition, several other model compression approaches have been developed, and the community has shown an increasing interest in knowledge distillation, due to its potentials (Amadori, 2019; Kushawaha et al., 2021; Rassadin & Savchenko, 2017; Wang et al., 2020).

A significant research opportunity lies in applying Knowledge distillation into embedded devices and underwater video processors to achieve online and more effective surveillance with high accuracy while using limited resources. This is particularly useful because of the limitations of transferring data from underwater sensors and cameras, and due to the challenging underwater communication in the Internet of Underwater Things (Jahanbakht et al., 2021).

7.2. Merging image data from multiple sources

As discussed in Saleh, Sheaves, and Rahimi Azghadi (2022), to train more effective DNNs, multiple data collection platforms like Autonomous Underwater Vehicles (AUVs) or inhabited submarines can give varied visual data from the same monitoring subject. This can provide additional monitoring information, such as fish distribution patterns. Although it is straightforward to combine multiple data sources for training a DL network, several issues should be addressed in future research. These include possible preprocessing on part of data to make it compatible with the rest of the training dataset, class-wise weights (i.e. when you have an imbalanced dataset), and the number of outputs of a network. In addition, multiple training data sources, in particular, when using AUVs or submarines, incurs significant data collection and manual labelling cost, which is not always viable.

For this reason, some researchers have focused on learning from data with the least amount of human labelling. To reduce human-labelled data cost, several methods have been proposed to train models on data that are unlabelled (Shimada et al., 2021) or only have pseudo-labels (Wu & Prasad, 2018). Future research can advance this further by developing faster and cheaper annotating tools for underwater fish images.

7.3. Prospective research

Deep learning has proven to be an effective tool for analysing and monitoring fish habitats and behaviour. However, there are still several areas where research is needed to further advance the use of DL in fish monitoring (Saleh, Sheaves, & Rahimi Azghadi, 2022). In this section, we discuss some prospective research directions that can increase the performance and usability of DL-based visual fish monitoring tasks.

1. Spatio-temporal data utilisation: DL models mainly learn fish images regardless of their spatial and temporal correlation. Utilising spatio-temporal data can add space and time domain information to the current training algorithms, leading to improved accuracy and robustness of the models. One potential approach is to use convolutional neural networks (CNNs) with 3D convolutions to learn both spatial and temporal features from video data. Another approach is to use recurrent neural networks (RNNs) to model temporal dependencies in sequential data, such as fish movement trajectories (Saleh, Sheaves, et al., 2022).
2. Efficient and compact DL models: To deploy DL models underwater for real-time parsing of fish images at the collection edge, compact and efficient models are needed. The current state-of-the-art DL models are often computationally expensive and require large amounts of memory. Research can focus on developing lightweight architectures that can be efficiently deployed on resource-constrained devices (Jahanbakht et al., 2021). One approach is to use knowledge distillation techniques to transfer knowledge from a large pre-trained model to a smaller model while maintaining performance.
3. Multi-platform data fusion: Combining image data from multiple collection platforms, such as sonar and acoustic sensors, can improve the multi-faceted learning of DL models. However, integrating data from different sources poses several challenges, including differences in data quality and format. Developing effective techniques for data fusion, such as transfer learning and domain adaptation, can help to overcome these challenges and improve the performance of DL models for fish monitoring (see Section 4.5).
4. Automated fish measurement and monitoring: Fish size and behaviour are important indicators of ecosystem health. Manual measurement and monitoring of fish can be time-consuming and expensive. DL models can automate this process by extracting size and behaviour features from underwater captured images (Saleh, Jones, et al., 2022). Research can focus on developing DL models that can accurately measure the fish size and identify behavioural patterns, such as swimming speed and direction.

In summary, prospective research directions can expand the capabilities and effectiveness of DL-based visual fish monitoring tasks. Utilising spatio-temporal data, developing efficient and compact models, multi-platform data fusion, and automated fish measurement and monitoring are some of the areas that can lead to further advancements in the field.

8. Summary and conclusion

The goal of this article was to provide researchers and practitioners with a summary of the contemporary applications of DL in underwater visual monitoring of fish, as well as to make it easier to apply DL to tackle real challenges in fish-related marine science.

DL has progressed as a technology capable of providing unprecedented benefits to various aspects of marine research and fish habitat monitoring. We envision a future where DL, complemented by many other advances in monitoring hardware and underwater communication technologies (Jahanbakht et al., 2021), is widely used in marine habitat monitoring for (1) data collection and feature extraction to improve the quality of automatic monitoring tools; and (2) to provide a reliable means of surveying fish habitats and understanding their dynamics. We expect that such a future will allow marine ecosystem researchers and practitioners to increase the efficiency of their monitoring efforts. To achieve this, we need concentrated and coordinated data collection, model development, and model deployment efforts. We also need transparent and reproducible research data and tools, which help us reach our target sooner.

CRediT authorship contribution statement

Alzayat Saleh: Conceptualization, Writing – original draft, Investigation, Visualization. **Marcus Sheaves:** Writing – review & editing, Supervision. **Dean Jerry:** Supervision, Writing – review & editing. **Mostafa Rahimi Azghadi:** Conceptualization, Writing – original draft, Supervision.

Declaration of competing 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.

Data availability

No data was used for the research described in the article.

Acknowledgements

This research is supported by an Australian Research Training Program (RTP) Scholarship, Australia and Food Agility HDR Top-Up Scholarship, Australia. We also acknowledge the Australian Research Council, Australia for funding awarded under their Industrial Transformation Research Program.

References

Abdul, M. S., Sam, S. M., Mohamed, N., Kamardin, K., & Dziyauddin, R. A. (2019). Docker containers usage in the internet of things: A survey. *Open International Journal of Informatics (OIJI)*, *7*(2), 208–220, URL http://apps.razak.utm.my/ojs/index.php/oiji/article/view/233.

Ahn, J., & Kwak, S. (2018). Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*. http://dx.doi.org/10.1109/CVPR.2018.00523.

Alshdaifat, N. F. F., Talib, A. Z., & Osman, M. A. (2020). Improved deep learning framework for fish segmentation in underwater videos. *Ecological Informatics*, *59*, Article 101121. http://dx.doi.org/10.1016/j.ecoinf.2020.101121.

Amadori, A. (2019). Distilling knowledge from Neural Networks to build smaller and faster models. In *FloydHub blog*.

Azghadi, M. R., Lammie, C., Eshraghian, J. K., Payvand, M., Donati, E., Linares-Barranco, B., & Indiveri, G. (2020). Hardware implementation of deep network accelerators towards healthcare and biomedical applications. *IEEE Transactions on Biomedical Circuits and Systems*, *14*(6), 1138–1159. http://dx.doi.org/10.1109/TBCAS.2020.3036081.

Bearman, A. L., Russakovsky, O., Ferrari, V., Fei-Fei, L., Bearman, A. L., Ferrari, V., Li, F.-F., Russakovsky, O., Ferrari, V., & Fei-Fei, L. (2016). What's the point: Semantic segmentation with point supervision. In *ECCV*. arXiv:1506.0.

Beauchemin, S. S., & Barron, J. L. (1995). The computation of optical flow. *ACM Computing Surveys*, *27*(3), 433–466. http://dx.doi.org/10.1145/212094.212141, URL https://dl.acm.org/doi/abs/10.1145/212094.212141.

Ben Tamou, A., Benzinou, A., & Nasreddine, K. (2022). Targeted data augmentation and hierarchical classification with deep learning for fish species identification in underwater images. *Journal of Imaging*, *8*(8), http://dx.doi.org/10.3390/JIMAGING8080214, URL https://pubmed.ncbi.nlm.nih.gov/36005457/.

Bendale, A., & Boult, T. E. (2016). Towards open set deep networks. In *2016 IEEE conference on computer vision and pattern recognition, Vol. 2016-December* (pp. 1563–1572). IEEE, http://dx.doi.org/10.1109/CVPR.2016.173, URL http://ieeexplore.ieee.org/document/7780542/.

Benoit-Bird, K. J., & Lawson, G. L. (2016). Ecological insights from pelagic habitats acquired using active acoustic techniques. *Annual Review of Marine Science*, *8*, 463–490. http://dx.doi.org/10.1146/ANNUREV-MARINE-122414-034001.

Blanchet, J.-N., Déry, S., Landry, J.-A., & Osborne, K. (2016). Automated annotation of corals in natural scene images using multiple texture representations. *PeerJ*, http://dx.doi.org/10.7287/peerj.preprints.2026.

Boudhane, M., & Nsiri, B. (2016). Underwater image processing method for fish localization and detection in submarine environment. *Journal of Visual Communication and Image Representation*, *39*, 226–238. http://dx.doi.org/10.1016/j.jvcir.2016.05.017, URL https://linkinghub.elsevier.com/retrieve/pii/S1047320316300840.

Bucila, C., Caruana, R., & Niculescu-Mizil, A. (2006). Model compression. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining*. http://dx.doi.org/10.1145/1150402.1150464.

Cao, Z., Principe, J. C., Ouyang, B., Dalgleish, F., & Vuorenkoski, A. (2016). Marine animal classification using combined CNN and hand-designed image features. In *OCEANS 2015 - MTS/IEEE Washington*. http://dx.doi.org/10.23919/oceans.2015.7404375.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., & Ma, T. (2019). Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in neural information processing systems, vol. 32*.

Chaudhari, S., Malkan, N., Momin, A., & Bonde, M. (2020). Yolo real time object detection. *International Journal of Computer Trends and Technology*, http://dx.doi.org/10.14445/22312803/ijctt-v68i6p112.

Chen, G., Sun, P., & Shang, Y. (2018). Automatic fish classification system using deep learning. In *Proceedings - International conference on tools with artificial intelligence*. http://dx.doi.org/10.1109/ICTAI.2017.00016.

Chen, L., Xia, Y., Pan, D., & Wang, C. (2019). Deep learning based active monitoring for anti-collision between vessels and bridges. In *IABSE symposium, guimaraes 2019: towards a resilient built environment risk and asset management - report*. http://dx.doi.org/10.2749/guimaraes.2019.0487.

Choi, S. (2015). *Fish identification in underwater video with deep convolutional neural network*: *Technical report*, CLEF, URL http://ceur-ws.org/Vol-1391/110-CR.pdf.

Chuang, M. C., Hwang, J. N., & Williams, K. (2016). A feature learning and object recognition framework for underwater fish images. *IEEE Transactions on Image Processing*, *25*(4), 1862–1872. http://dx.doi.org/10.1109/TIP.2016.2535342.

Chuang, M. C., Hwang, J. N., Williams, K., & Towler, R. (2011). Automatic fish segmentation via double local thresholding for trawl-based underwater camera systems. In *Proceedings - international conference on image processing* (pp. 3145–3148). http://dx.doi.org/10.1109/ICIP.2011.6116334.

Cui, Y., Jia, M., Lin, T.-Y., Song, Y., & Belongie, S. (2019). Class-balanced loss based on effective number of samples. In *2019 IEEE/CVF conference on computer vision and pattern recognition, vol. 2019-June* (pp. 9260–9269). IEEE, http://dx.doi.org/10.1109/CVPR.2019.00949, URL https://ieeexplore.ieee.org/document/8953804/.

Dai, J., He, K., & Sun, J. (2015). Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV* (pp. 1635–1643).

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings - 2005 IEEE computer society conference on computer vision and pattern recognition, CVPR 2005, vol. I* (pp. 886–893). http://dx.doi.org/10.1109/CVPR.2005.177.

De Vos, B. D., Wolterink, J. M., De Jong, P. A., Leiner, T., Viergever, M. A., & Isgum, I. (2017). ConvNet-based localization of anatomical structures in 3-D medical images. *IEEE Transactions on Medical Imaging*, http://dx.doi.org/10.1109/TMI.2017.2673121.

Ditria, E. M., Connolly, R. M., Jinks, E. L., & Lopez-Marcano, S. (2021). Annotated video footage for automated identification and counting of fish in unconstrained seagrass habitats. *Frontiers in Marine Science*, *8*, http://dx.doi.org/10.3389/fmars.2021.629485, URL https://www.frontiersin.org/articles/10.3389/fmars.2021.629485/full.

Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., & Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: Optimizing animal ecology with deep learning. *Frontiers in Marine Science*, *7*, http://dx.doi.org/10.3389/fmars.2020.00429, URL https://www.frontiersin.org/article/10.3389/fmars.2020.00429/full.

Ditria, E. M., Sievers, M., Lopez-Marcano, S., Jinks, E. L., & Connolly, R. M. (2020). Deep learning for automated analysis of fish abundance: the benefits of training across multiple habitats. *Environmental Monitoring and Assessment*, http://dx.doi.org/10.1007/s10661-020-08653-z.

Duan, Z., & Deng, J. (2019). Automatic video tracking of chinese mitten crab using particle filter based on multi features. In *2019 IEEE 3rd International Conference on Electronic Information Technology and Computer Engineering, EITCE 2019*. http://dx.doi.org/10.1109/EITCE47263.2019.9095032.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, http://dx.doi.org/10.1109/TPAMI.2009.167.

Feng, J., & Li, J. (2022). An adaptive embedding network with spatial constraints for the use of few-shot learning in endangered-animal detection. *ISPRS International Journal of Geo-Information*, *11*(4), 256. http://dx.doi.org/10.3390/ijgi11040256, URL https://www.mdpi.com/2220-9964/11/4/256.

Feng, J., & Xiao, X. (2022). Multiobject tracking of wildlife in videos using few-shot learning. *Animals*, *12*(9), 1223. http://dx.doi.org/10.3390/ani12091223, URL https://www.mdpi.com/2076-2615/12/9/1223.

Fouad, M. M. M., Zawbaa, H. M., El-Bendary, N., & Hassanien, A. E. (2014). Automatic nile tilapia fish classification approach using machine learning techniques. In *13th International conference on hybrid intelligent systems* (pp. 173–178). http://dx.doi.org/10.1109/HIS.2013.6920477.

Fu, C. W., Horng, J. L., & Chou, M. Y. (2022). Fish behavior as a neural proxy to reveal physiological states. *Frontiers in Physiology*, *13*, 1420. http://dx.doi.org/10.3389/FPHYS.2022.937432/BIBTEX.

Fu, G., & Yuna, Y. (2022). Phenotyping and phenomics in aquaculture breeding. *Aquaculture and Fisheries*, *7*(2), 140–146. http://dx.doi.org/10.1016/J.AAF.2021.07.001.

Garcia, J. A., Masip, D., Sbragaglia, V., & Aguzzi, J. (2016). Automated identification and tracking of nephrops norvegicus (l.) using infrared and monochromatic blue light. In *Frontiers in artificial intelligence and applications*. http://dx.doi.org/10.3233/978-1-61499-696-5-9.

Giordano, D., Palazzo, S., & Spampinato, C. (2016). Fish4Knowledge: Collecting and analyzing massive coral reef fish video data. *Intelligent Systems Reference Library*.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*. http://dx.doi.org/10.1109/CVPR.2014.81.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., & Bengio, Y. (2014). Generative adversarial networks. arXiv, arXiv:1406.2.

Goodwin, M., Halvorsen, K. T., Jiao, L., Knausgård, K. M., Martin, A. H., Moyano, M., Oomen, R. A., Rasmussen, J. H., Sørdalen, T. K., & Thorbjørnsen, S. H. (2022). Unlocking the potential of deep learning for marine ecology: overview, applications, and outlook. *ICES Journal of Marine Science*, *79*(2), 319–336. http://dx.doi.org/10.1093/icesjms/fsab255, URL https://arxiv.org/abs/2109.14737v1.

Han, F., Yao, J., Zhu, H., & Wang, C. (2020). Marine organism detection and classification from underwater vision based on the deep CNN method. *Mathematical Problems in Engineering*, http://dx.doi.org/10.1155/2020/3937580.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *Computer Vision and Pattern Recognition (CVPR)*.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Hossain, E., Alam, S. M. S., Ali, A. A., & Amin, M. A. (2016). Fish activity tracking and species identification in underwater video. In *2016 5th International conference on informatics, electronics and vision* (pp. 62–66). http://dx.doi.org/10.1109/ICIEV.2016.7760189.

Hu, Y., Mian, A. S., & Owens, R. (2012). Face recognition using sparse approximated nearest points between image sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*, 1992–2004. http://dx.doi.org/10.1109/TPAMI.2011.283.

Huang, P. X., Boom, B. J., & Fisher, R. B. (2014). GMM improves the reject option in hierarchical classification for fish recognition. In *IEEE winter conference on applications of computer vision* (pp. 371–376). IEEE, http://dx.doi.org/10.1109/WACV.2014.6836076, URL https://ieeexplore.ieee.org/document/6836076.

Huang, Z., XinggangWang, J., Liu, W., & Wang, J. (2018). Weakly-supervised semantic segmentation network with deep seeded region growing. In *IEEE* (pp. 7014–7023).

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*.

Iqbal, M. A., Wang, Z., Ali, Z. A., & Riaz, S. (2021). Automatic fish species classification using deep convolutional neural networks. *Wireless Personal Communications*, http://dx.doi.org/10.1007/s11277-019-06634-1.

Islam, M. J., Edge, C., Xiao, Y., Luo, P., Mehtaz, M., Morse, C., Enan, S. S., & Sattar, J. (2020). Semantic segmentation of underwater imagery: Dataset and benchmark. URL http://arxiv.org/abs/2004.01241.

Islam, M. A., Howlader, M. R., Habiba, U., Faisal, R. H., & Rahman, M. M. (2019). Indigenous fish classification of Bangladesh using hybrid features with SVM classifier. In *5th International conference on computer, communication, chemical, materials and electronic engineering*. http://dx.doi.org/10.1109/IC4ME247184.2019.9036679.

Jahanbakht, M., Xiang, W., Hanzo, L., & Azghadi, M. R. (2021). Internet of underwater things and big marine data analytics - A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *23*(2), 904–956. http://dx.doi.org/10.1109/COMST.2021.3053118, URL https://ieeexplore.ieee.org/document/9328873/.

Jalal, A., Salman, A., Mian, A., Shortis, M., & Shafait, F. (2020). Fish detection and species classification in underwater environments using deep learning with temporal information. *Ecological Informatics*, *57*, Article 101088. http://dx.doi.org/10.1016/j.ecoinf.2020.101088.

Jing, L., Chen, Y., & Tian, Y. (2020). Coarse-to-fine semantic segmentation from image-level labels. *IEEE Transactions on Image Processing*, http://dx.doi.org/10.1109/TIP.2019.2926748.

Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.-P., Planque, R., Rauber, A., Fisher, R., & Müller, H. (2014). Lifeclef 2014: Multimedia life species identification challenges. In E. Kanoulas, M. Lupu, P. Clough, M. Sanderson, M. Hall, A. Hanbury, & E. Toms (Eds.), *Lecture notes in computer science*: *vol. 8685*, *Information access evaluation. multilinguality, multimodality, and interaction* (pp. 229–249). Cham: Springer International Publishing.

Juliani, C., & Juliani, E. (2021). Deep learning of terrain morphology and pattern discovery via network-based representational similarity analysis for deep-sea mineral exploration. *Ore Geology Reviews*, http://dx.doi.org/10.1016/j.oregeorev.2020.103936.

Kang, D., Ma, Z., & Chan, A. B. (2018). Beyond counting: comparisons of density maps for crowd analysis tasks-counting, detection, and tracking. *IEEE Transactions on Circuits and Systems for Video Technology*.

Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, *53*(8), 5455–5516. http://dx.doi.org/10.1007/s10462-020-09825-6.

Khazukov, K., Shepelev, V., Karpeta, T., Shabiev, S., Slobodin, I., Charbadze, I., & Alferova, I. (2020). Real-time monitoring of traffic parameters. *Journal of Big Data*, *7*(1), http://dx.doi.org/10.1186/s40537-020-00358-x.

Khoreva, A., Benenson, R., Hosang, J. H., Hein, M., & Schiele, B. (2017). Simple does it: Weakly supervised instance and semantic segmentation. *Conference Vision and Pattern Recognition*, 876–885.

Kim, S., Park, B., Song, B. S., & Yang, S. (2016). Deep belief network based statistical feature learning for fingerprint liveness detection. *Pattern Recognition Letters*, *77*, 58–65. http://dx.doi.org/10.1016/j.patrec.2016.03.015, URL https://linkinghub.elsevier.com/retrieve/pii/S0167865516300198.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Knausgård, K. M., Wiklund, A., Sørdalen, T. K., Halvorsen, K. T., Kleiven, A. R., Jiao, L., & Goodwin, M. (2021). Temperate fish detection and classification: a deep learning based approach. *Applied Intelligence*, http://dx.doi.org/10.1007/s10489-020-02154-9.

Konovalov, D. A., Saleh, A., Bradley, M., Sankupellay, M., Marini, S., & Sheaves, M. (2019). Underwater fish detection with weak multi-domain supervision. In *2019 International joint conference on neural networks, vol. 2019-July* (pp. 1–8). IEEE, http://dx.doi.org/10.1109/IJCNN.2019.8851907, URL https://ieeexplore.ieee.org/document/8851907/.

Konovalov, D. A., Saleh, A., Domingos, J. A., White, R. D., & Jerry, D. R. (2018). Estimating mass of harvested Asian seabass lates calcarifer from images. *World Journal of Engineering and Technology*, *6*(03), 15. http://dx.doi.org/10.4236/wjet.2018.63b003.

Konovalov, D. A., Saleh, A., Efremova, D. B., Domingos, J. A., & Jerry, D. R. (2019). Automatic weight estimation of harvested fish from images. In *Digital image computing: techniques and applications* (pp. 1–7).

Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. In *Informatica (Ljubljana)*. http://dx.doi.org/10.31449/inf.v31i3.148.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems 25* (pp. 1097–1105). Curran Associates, Inc..

Kukačka, J., Golkov, V., & Cremers, D. (2017). Regularization for deep learning: A taxonomy. arXiv preprint arXiv:1710.10686.

Kushawaha, R. K., Kumar, S., Banerjee, B., & Velmurugan, R. (2021). Distilling spikes: Knowledge distillation in spiking neural networks. In *IEEE*. http://dx.doi.org/10.1109/icpr48806.2021.9412147.

Labao, A. B., & Naval, P. C. (2017). *Lecture Notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, *Weakly-labelled semantic segmentation of fish objects in underwater videos using a deep residual network*. http://dx.doi.org/10.1007/978-3-319-54430-4{_}25.

Labao, A. B., & Naval, P. C. (2019a). Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild. *Ecological Informatics*, http://dx.doi.org/10.1016/j.ecoinf.2019.05.004.

Labao, A. B., & Naval, P. C. (2019b). Simultaneous localization and segmentation of fish objects using multi-task CNN and dense CRF. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*. http://dx.doi.org/10.1007/978-3-030-14799-0{_}52.

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*, http://dx.doi.org/10.1109/ACCESS.2019.2911709.

Laradji, I., Rodriguez, P., Manas, O., Lensink, K., Law, M., Kurzman, L., Parker, W., Vazquez, D., & Nowrouzezahrai, D. (2021). A weakly supervised consistency-based learning method for COVID-19 segmentation in CT images. In *WACV*.

Laradji, I., Saleh, A., Rodriguez, P., Nowrouzezahrai, D., Azghadi, M. R., & Vazquez, D. (2020). Affinity LCFCN: Learning to segment fish with weak supervision. *Scientific Reports*, URL http://arxiv.org/abs/2011.03149.

Laradji, I. H., Saleh, A., Rodriguez, P., Nowrouzezahrai, D., Azghadi, M. R., & Vazquez, D. (2021). Weakly supervised underwater fish segmentation using affinity lcfcn. *Scientific Reports*, *11*(1), 17379. http://dx.doi.org/10.1038/s41598-021-96610-2, https://www.nature.com/articles/s41598-021-96610-2 http://www.ncbi.nlm.nih.gov/pubmed/34462458 http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC8405733.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. http://dx.doi.org/10.1038/nature14539, URL http://www.nature.com/articles/nature14539.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., & Shi, W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings - 30th IEEE conference on computer vision and pattern recognition, vol. 2017-Janua* (pp. 105–114). http://dx.doi.org/10.1109/CVPR.2017.19.

Lee, K. H., He, X., Zhang, L., & Yang, L. (2018). CleanNet: Transfer learning for scalable image classifier training with label noise. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*. http://dx.doi.org/10.1109/CVPR.2018.00571.

Lennox, R. J., Aarestrup, K., Cooke, S. J., Cowley, P. D., Deng, Z. D., Fisk, A. T., Harcourt, R. G., Heupel, M., Hinch, S. G., Holland, K. N., Hussey, N. E., Iverson, S. J., Kessel, S. T., Kocik, J. F., Lucas, M. C., Flemming, J. M., Nguyen, V. M., Stokesbury, M. J., Vagle, S., ... Young, N. (2017). Envisioning the future of aquatic animal tracking: Technology, science, and application. *BioScience*, http://dx.doi.org/10.1093/biosci/bix098.

Li, D., & Du, L. (2021). Recent advances of deep learning algorithms for aquacultural machine vision systems with emphasis on fish. *Artificial Intelligence Review*, 1–40. http://dx.doi.org/10.1007/s10462-021-10102-3, https://link.springer.com/article/10.1007/s10462-021-10102-3 https://link.springer.com/10.1007/s10462-021-10102-3.

Li, L.-J., Li, K., Li, F. F., Deng, J., Dong, W., Socher, R., & Fei-Fei, L. (2009). ImageNet: a large-scale hierarchical image database shrimp project view project hybrid intrusion detction systems view project ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*.

Li, Z., Li, W., Li, F., & Yuan, M. (2021). *A Review of computer vision technologies for fish tracking*. IEEE, http://dx.doi.org/10.48550/arxiv.2110.02551, URL http://arxiv.org/abs/2110.02551.

Li, X., Shang, M., Qin, H., & Chen, L. (2015). Fast accurate fish detection and recognition of underwater images with fast R-CNN. In *OCEANS 2015 - MTS/IEEE Washington* (pp. 1–5). http://dx.doi.org/10.23919/OCEANS.2015.7404464.

Li, D., Wang, G., Du, L., Zheng, Y., & Wang, Z. (2022). Recent advances in intelligent recognition methods for fish stress behavior. *Aquacultural Engineering*, *96*, Article 102222. http://dx.doi.org/10.1016/J.AQUAENG.2021.102222.

Lin, D., Dai, J., Jia, J., He, K., & Sun, J. (2016). Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *IEEE* (pp. 3159–3167).

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal loss for dense object detection. In *2017 IEEE international conference on computer vision, vol. 2017-October* (pp. 2999–3007). IEEE, http://dx.doi.org/10.1109/ICCV.2017.324, URL http://ieeexplore.ieee.org/document/8237586/.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, *Microsoft COCO: common objects in context*. http://dx.doi.org/10.1007/978-3-319-10602-1{_}48.

Lindeberg, T. (2012). Scale Invariant Feature Transform. *Scholarpedia*, *7*(5), 10491. http://dx.doi.org/10.4249/scholarpedia.10491, URL http://www.scholarpedia.org/article/Scale_Invariant_Feature_Transform.

Liu, L., Lu, H., Cao, Z., & Xiao, Y. (2018). Counting fish in sonar images. In *Proceedings - international conference on image processing*. http://dx.doi.org/10.1109/ICIP.2018.8451154.

Liu, Z., Miao, Z., Zhan, X., Wang, J., Gong, B., & Yu, S. X. (2019). Large-scale long-tailed recognition in an open world. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, vol. 2019-June* (pp. 2532–2541). IEEE, http://dx.doi.org/10.1109/CVPR.2019.00264, URL https://ieeexplore.ieee.org/document/8953407/.

Lopez, S., Jinks, E., Buelow, C. A., Brown, C. J., Wang, D., Kusy, B., Ditria, E., & Connolly, R. M. (2021). Automatic detection of fish and tracking of movement for ecology. *Ecology and Evolution*, *11*(12), 8254–8263. http://dx.doi.org/10.1002/ece3.7656, URL https://onlinelibrary.wiley.com/doi/10.1002/ece3.7656.

Lumauag, R., & Nava, M. (2019). Fish tracking and counting using image processing. In *2018 IEEE 10th international conference on humanoid, nanotechnology, information technology, communication and control, environment and management*. http://dx.doi.org/10.1109/HNICEM.2018.8666369.

Mader, A. O., Lorenz, C., Bergtholdt, M., von Berg, J., Schramm, H., Modersitzki, J., & Meyer, C. (2018). Detection and localization of spatially correlated point landmarks in medical images using an automatically learned conditional random field. *Computer Vision and Image Understanding*, http://dx.doi.org/10.1016/j.cviu.2018.09.009.

Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., Kendrick, G., & Fisher, R. B. (2016). Coral classification with hybrid feature representations. In *Proceedings - international conference on image processing*. http://dx.doi.org/10.1109/ICIP.2016.7532411.

Mancusi, M., Zonca, N., Rodolà, E., & Zuffi, S. (2022). Fish sounds: towards the evaluation of marine acoustic biodiversity through data-driven audio source separation. http://dx.doi.org/10.48550/arxiv.2201.05013, URL https://arxiv.org/abs/2201.05013v2.

Mandal, R., Connolly, R. M., Schlacher, T. A., & Stantic, B. (2018). Assessing fish abundance from underwater video using deep neural networks. In *Proceedings of the international joint conference on neural networks*. http://dx.doi.org/10.1109/IJCNN.2018.8489482.

Mathur, M., Vasudev, D., Sahoo, S., Jain, D., & Goel, N. (2020). Crosspooled FishNet: transfer learning based fish species classification model. *Multimedia Tools and Applications*, http://dx.doi.org/10.1007/s11042-020-09371-x.

McCann, E., Li, L., Pangle, K., Johnson, N., & Eickholt, J. (2018). An underwater observation dataset for fish classification and fishery assessment. *Scientific Data*, *5*(1), 1–8. http://dx.doi.org/10.1038/sdata.2018.190, https://www.nature.com/articles/sdata2018190 https://www.nature.com/articles/sdata2018190/.

Meng, L., Hirayama, T., & Oyanagi, S. (2018). Underwater-drone with panoramic camera for automatic fish recognition based on deep learning. *IEEE Access*, http://dx.doi.org/10.1109/ACCESS.2018.2820326.

Molchanov, P., Tyree, S., Karras, T., Aila, T., & Kautz, J. (2016). Pruning convolutional neural networks for resource efficient transfer learning, CoRR, /abs/1611.0. arXiv:1611.0.

Moniruzzaman, M., Islam, S. M. S., Bennamoun, M., & Lavery, P. (2017). *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*: *vol. 10617 LNCS*, *Deep learning on underwater marine object detection: A survey* (pp. 150–160). Springer Verlag, http://dx.doi.org/10.1007/978-3-319-70353-4{_}13, URL http://link.springer.com/10.1007/978-3-319-70353-4_13.

Muñoz, L., Aspillaga, E., Palmer, M., Saraiva, J. L., & Arechavala-Lopez, P. (2020). Acoustic telemetry: A tool to monitor fish swimming behavior in sea-cage aquaculture. *Frontiers in Marine Science*, *7*, 645. http://dx.doi.org/10.3389/FMARS.2020.00645/BIBTEX.

Naseer, A., Baro, E. N., Khan, S. D., & Gordillo, Y. V. (2020). Automatic detection of nephrops norvegicus burrows in underwater images using deep learning. In *2020 Global conference on wireless and optical technologies*. http://dx.doi.org/10.1109/GCWOT49901.2020.9391590.

Nilssen, I., Moller, T., & Nattkemper, T. W. (2017). Active learning for the classification of species in underwater images from a fixed observatory. In *Proceedings - 2017 IEEE international conference on computer vision workshops, vol. 2018-Janua* (pp. 2891–2897). http://dx.doi.org/10.1109/ICCVW.2017.341.

Niu, B., Li, G., Peng, F., Wu, J., Zhang, L., & Li, Z. (2018). Survey of fish behavior analysis by computer vision. *Journal of Aquaculture Research and Development*, *09*(05), http://dx.doi.org/10.4172/2155-9546.1000534.

Ogunlana, S. O., Olabode, O., & Oluwadare, S. A. A. (2015). Fish classification using support vector machine. *African Journal of Computing & ICT*, *8*(2), 75–82.

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2015). Is object localization for free? - weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*. http://dx.doi.org/10.1109/CVPR.2015.7298668.

Pathak, D., Krähenbühl, P., Darrell, T., Krahenbuhl, P., & Darrell, T. (2015). Constrained convolutional neural networks for weakly supervised segmentation. In *2015 IEEE international conference on computer vision* (pp. 1796–1804).

Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of deep learning for object detection. In *Procedia computer science*. http://dx.doi.org/10.1016/j.procs.2018.05.144.

Potdar, A. M., Narayan, D. G., Kengond, S., & Mulla, M. M. (2020). Performance evaluation of docker container and virtual machine. In *Procedia computer science. vol. 171* (pp. 1419–1428). http://dx.doi.org/10.1016/j.procs.2020.04.152.

Qi, X., Liu, Z., Shi, J., Zhao, H., & Jia, J. 0000, Augmented feedback in semantic segmentation under image level supervision, European conference on computer vision, 90–105, Springer.

Qian, Z.-M., Wang, S. H., Cheng, X. E., & Chen, Y. Q. (2016). An effective and robust method for tracking multiple fish in video image based on fish head detection. *BMC Bioinformatics*, *17*(1), 251. http://dx.doi.org/10.1186/s12859-016-1138-y, URL http://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-016-1138-y.

Qiu, C., Zhang, S., Wang, C., Yu, Z., Zheng, H., & Zheng, B. (2018). Improving transfer learning and squeeze- and-excitation networks for small-scale fine-grained fish image classification. *IEEE Access*, *6*, 78503–78512. http://dx.doi.org/10.1109/ACCESS.2018.2885055.

Rajchl, M., Lee, M. C. H., Oktay, O., Kamnitsas, K., Passerat-Palmbach, J., Bai, W., Damodaram, M., Rutherford, M. A., Hajnal, J. V., & Kainz, B. (2016). Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE Transactions on Medical Imaging*, *36*(2), 683.

Rassadin, A. G., & Savchenko, A. V. (2017). Compressing deep convolutional neural networks in visual emotion recognition. In *CEUR workshop proceedings*. http://dx.doi.org/10.18287/1613-0073-2017-1901-207-213.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6), 1137–1149. http://dx.doi.org/10.1109/TPAMI.2016.2577031, URL http://ieeexplore.ieee.org/document/7485869/.

Rojas, R., & Rojas, R. (1996). The backpropagation algorithm. In *Neural Networks*. http://dx.doi.org/10.1007/978-3-642-61068-4{_}7.

Rova, A., Mori, G., & Dill, L. M. (2007). One fish, two fish, butterfish, trumpeter: Recognizing fish in underwater video. In *Proceedings of IAPR conference on machine vision applications* (pp. 404–407).

Saberioon, M. M., & Cisar, P. (2016). Automated multiple fish tracking in three-dimension using a structured light sensor. *Computers and Electronics in Agriculture*, http://dx.doi.org/10.1016/j.compag.2015.12.014.

Saleh, A., Jones, D., Jerry, D., & Azghadi, M. R. (2022). A lightweight transformer-based model for fish landmark detection. http://dx.doi.org/10.48550/arxiv.2209.05777, URL https://arxiv.org/abs/2209.05777v1.

Saleh, A., Laradji, I. H., Konovalov, D. A., Bradley, M., Vazquez, D., & Sheaves, M. (2020a). A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Scientific Reports*, *10*(1), 14671. http://dx.doi.org/10.1038/s41598-020-71639-x, http://www.ncbi.nlm.nih.gov/pubmed/32887922, http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC7473859, https://www.nature.com/articles/s41598-020-71639-x.

Saleh, A., Laradji, I. H., Konovalov, D. A., Bradley, M., Vazquez, D., & Sheaves, M. (2020b). A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Scientific Reports*, *10*(1), 14671. http://dx.doi.org/10.1038/s41598-020-71639-x, URL https://www.nature.com/articles/s41598-020-71639-x.

Saleh, A., Sheaves, M., Jerry, D., & Azghadi, M. R. (2022). Unsupervised fish trajectory tracking and segmentation. http://dx.doi.org/10.48550/arxiv.2208.10662, URL https://arxiv.org/abs/2208.10662v1.

Saleh, A., Sheaves, M., & Rahimi Azghadi, M. (2022). Computer vision and deep learning for fish classification in underwater habitats: A survey. *Fish and Fisheries*, *23*(4), 977–999. http://dx.doi.org/10.1111/faf.12666, URL https://onlinelibrary.wiley.com/doi/10.1111/faf.12666.

Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M. R., Khurshid, K., Ulges, A., & Schwanecke, U. (2019). Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES Journal of Marine Science*.

Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M. R., Khurshid, K., Ulges, A., & Schwanecke, U. (2020). Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES Journal of Marine Science*, http://dx.doi.org/10.1093/icesjms/fsz025.

Sarigül, M., & Avci, M. (2017). Comparison of different deep structures for fish classification. *International Journal of Computer Theory and Engineering*, http://dx.doi.org/10.7763/ijcte.2017.v9.1167.

Schneider, S., & Zhuang, A. (2020). Counting fish and dolphins in sonar images using deep learning. arXiv preprint arXiv:2007.12808.

Shelhamer, E., Long, J., & Darrell, T. (2017). Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(4), 640–651. http://dx.doi.org/10.1109/TPAMI.2016.2572683, URL http://www.ncbi.nlm.nih.gov/pubmed/27244717.

Shimada, T., Bao, H., Sato, I., & Sugiyama, M. (2021). Classification from pairwise similarities/dissimilarities and unlabeled data via empirical risk minimization. *Neural Computation*, *33*(5), 1234–1268. http://dx.doi.org/10.1162/neco{_}a{_}01373, URL https://direct.mit.edu/neco/article/33/5/1234/97483/Classification-From-Pairwise-Similarities.

Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., & Harvey, E. S. (2018). Automatic fish species classification in underwater videos: Exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, http://dx.doi.org/10.1093/icesjms/fsx109.

Sun, S., Cao, Z., Zhu, H., & Zhao, J. (2019). A survey of optimization methods from a machine learning perspective. *IEEE Transactions on Cybernetics*, 1–14. http://dx.doi.org/10.1109/tcyb.2019.2950779.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *2015 IEEE conference on computer vision and pattern recognition* (pp. 1–9).

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). Rethinking the inception architecture for computer vision, CoRR, abs/1512.0. arXiv:1512.0.

Tarling, P., Cantor, M., Clapés, A., & Escalera, S. (2021). *Deep learning with self-supervision and uncertainty regularization to count fish in underwater images*: *Technical report*, Other, URL http://www.echoview.com.

Villon, S., Chaumont, M., Subsol, G., Villéger, S., Claverie, T., & Mouillot, D. (2016). Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between deep learning and HOG+SVM methods. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, http://dx.doi.org/10.1007/978-3-319-48680-2{_}15.

Villon, S., Iovan, C., Mangeas, M., Claverie, T., Mouillot, D., Villéger, S., & Vigliola, L. (2021). Automatic underwater fish species classification with limited data using few-shot learning. *Ecological Informatics*, *63*, Article 101320. http://dx.doi.org/10.1016/j.ecoinf.2021.101320, URL https://linkinghub.elsevier.com/retrieve/pii/S1574954121001114.

Villon, S., Iovan, C., Mangeas, M., & Vigliola, L. (2022). Confronting deep-learning and biodiversity challenges for automatic video-monitoring of marine ecosystems. *Sensors*, *22*(2), 497. http://dx.doi.org/10.3390/s22020497, URL https://www.mdpi.com/1424-8220/22/2/497.

Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., & Villéger, S. (2018). A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, http://dx.doi.org/10.1016/j.ecoinf.2018.09.007.

Wang, P., Chen, Q., He, X., & Cheng, J. (2020). *Towards accurate post-training network quantization via bit-split and stitching* (pp. 9847–9856). PMLR, URL http://proceedings.mlr.press/v119/wang20c.html.

Wang, G., Hwang, J. N., Williams, K., Wallace, F., & Rose, C. S. (2017). Shrinking encoding with two-level codebook learning for fine-grained fish recognition. In *Proceedings - 2nd workshop on computer vision for analysis of underwater imagery, CVAUI 2016 - in conjunction with international conference on pattern recognition* (pp. 31–36). http://dx.doi.org/10.1109/CVAUI.2016.18.

Wang, S., & Kanwar, P. (2021). BFloat16: The secret to high performance on cloud TPUs $vert$ google cloud blog. In *Google Cloud Blog*. URL https://cloud.google.com/blog/products/ai-machine-learning/bfloat16-the-secret-to-high-performance-on-cloud-tpus.

Wang, D., Vinson, R., Holmes, M., & Seibel, G. (2018). Convolutional neural network guided blue crab knuckle detection for autonomous crab meat picking machine. *Optimization and Engineering*, http://dx.doi.org/10.1117/1.oe.57.4.043103.

Wang, Y., Yao, Q., Kwok, J. T., & Ni, L. M. (2021). Generalizing from a few examples. *ACM Computing Surveys*, *53*(3), 1–34. http://dx.doi.org/10.1145/3386252, URL https://dl.acm.org/doi/10.1145/3386252.

Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., & Huang, T. S. (2018). Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *IEEE* (pp. 7268–7277).

Wörz, S., & Rohr, K. (2006). Localization of anatomical point landmarks in 3D medical images by fitting 3D parametric intensity models. *Medical Image Analysis*, http://dx.doi.org/10.1016/j.media.2005.02.003.

Wu, H., & Prasad, S. (2018). Semi-supervised deep learning using pseudo labels for hyperspectral image classification. *IEEE Transactions on Image Processing*, *27*(3), 1259–1270. http://dx.doi.org/10.1109/TIP.2017.2772836, URL http://ieeexplore.ieee.org/document/8105856/.

Xu, W., & Matzner, S. (2018). Underwater fish detection using deep learning for water power applications. In *2018 International conference on computational science and computational intelligence* (pp. 313–318). IEEE, http://dx.doi.org/10.1109/CSCI46756.2018.00067, URL https://ieeexplore.ieee.org/document/8947884/.

Xue, Y., Ray, N., Hugh, J., & Bigras, G. (2016). *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, *Cell counting by regression using convolutional neural network*. http://dx.doi.org/10.1007/978-3-319-46604-0{_}20.

Yang, L., Liu, Y., Yu, H., Fang, X., Song, L., Li, D., & Chen, Y. (2021). Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: A review. *Archives of Computational Methods in Engineering*, *28*(4), 2785–2816. http://dx.doi.org/10.1007/s11831-020-09486-2, URL https://link.springer.com/10.1007/s11831-020-09486-2.

Yang, X., Zhang, S., Liu, J., Gao, Q., Dong, S., & Zhou, C. (2021). Deep learning for smart fish farming: applications, opportunities and challenges. *Reviews in Aquaculture*, *13*(1), 66–90. http://dx.doi.org/10.1111/raq.12464, URL https://onlinelibrary.wiley.com/doi/10.1111/raq.12464.

Zhang, W., Wu, C., & Bao, Z. (2022). DPANet: Dual pooling-aggregated attention network for fish segmentation. *IET Computer Vision*, *16*(1), 67–82. http://dx.doi.org/10.1049/cvi2.12065, URL https://onlinelibrary.wiley.com/doi/10.1049/cvi2.12065.

Zhang, S., Wu, G., Costeira, J. P., & Moura, J. M. F. (2017). Understanding traffic density from large-scale web camera data. In *2017 IEEE Conference on computer vision and pattern recognition* (pp. 5898–5907). http://dx.doi.org/10.1109/CVPR.2017.454.

Zhao, K. L., Jin, X. L., & Wang, Y. Z. (2021). Survey on few-shot learning. *Ruan Jian Xue Bao/Journal of Software*, *32*(2), http://dx.doi.org/10.13328/j.cnki.jos.006138.

Zhao, X., Yan, S., & Gao, Q. (2019). An algorithm for tracking multiple fish based on biological water quality monitoring. *IEEE Access*, http://dx.doi.org/10.1109/ACCESS.2019.2895072.

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., & Zhao, R. (2021). Application of machine learning in intelligent fish aquaculture: A review. *Aquaculture*, *540*, Article 736724. http://dx.doi.org/10.1016/j.aquaculture.2021.736724, URL https://linkinghub.elsevier.com/retrieve/pii/S0044848621003860.

Zhou, Z. H. (2018). A brief introduction to weakly supervised learning. *National Science Review*, http://dx.doi.org/10.1093/nsr/nwx106.

Zhou, Y., Yu, H., Wu, J., Cui, Z., & Zhang, F. (2019). Fish behavior analysis based on computer vision: A survey. *Communications in Computer and Information Science*, *1059*, 130–141. http://dx.doi.org/10.1007/978-981-15-0121-0{_}10/FIGURES/4, URL https://link.springer.com/chapter/10.1007/978-981-15-0121-0_10.

Zhou, X., Yu, C., Yuan, S., Yuan, X., Yu, H., & Luo, C. (2022). Learning visual representation of underwater acoustic imagery using transformer-based style transfer method. http://dx.doi.org/10.48550/arxiv.2211.05396, URL https://arxiv.org/abs/2211.05396v1.

Zhuang, P., Xing, L., Liu, Y., Guo, S., & Qiao, Y. (2017). Marine animal detection and recognition with advanced deep learning models. In *CEUR workshop proceedings*.

Zivkovic, Z., & van der Heijden, F. (2006). Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, *27*, 773–780. http://dx.doi.org/10.1016/j.patrec.2005.11.005.

Zurowietz, M., & Nattkemper, T. W. (2020). Unsupervised knowledge transfer for object detection in marine environmental monitoring and exploration. *IEEE Access*, *8*, 143558–143568. http://dx.doi.org/10.1109/ACCESS.2020.3014441.