Ecological Informatics 57 (2020) 101088

Contents lists available at ScienceDirect

Ecological Informatics

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

Fish detection and species classification in underwater environments using deep learning with temporal information

Ahsan Jalal[a], Ahmad Salman[a,*], Ajmal Mian[b], Mark Shortis[c], Faisal Shafait[a]

[a] School of Electrical Engineering and Computer Sciences, National University of Sciences and Technology, Islamabad 44000, Pakistan
[b] School of Computer Science and Software Engineering, University of Western Australia, 35 Stirling Hwy, Crawley 6009, WA, Australia
[c] School of Science, RMIT University, GPO Box 2476, Melbourne 3001, VIC, Australia

ARTICLE INFO

Keywords:
Automatic fish detection
Fish species classification
Fish sampling
Biomass estimation
Underwater video imagery
Gaussian mixture models
Optical flow
Deep learning

ABSTRACT

It is important for marine scientists and conservationists to frequently estimate the relative abundance of fish species in their habitats and monitor changes in their populations. As opposed to laborious manual sampling, various automatic computer-based fish sampling solutions in underwater videos have been presented. However, an optimal solution for automatic fish detection and species classification does not exist. This is mainly because of the challenges present in underwater videos due to environmental variations in luminosity, fish camouflage, dynamic backgrounds, water murkiness, low resolution, shape deformations of swimming fish, and subtle variations between some fish species. To overcome these challenges, we propose a hybrid solution to combine optical flow and Gaussian mixture models with YOLO deep neural network, an unified approach to detect and classify fish in unconstrained underwater videos. YOLO based object detection system are originally employed to capture only the static and clearly visible fish instances. We eliminate this limitation of YOLO to enable it to detect freely moving fish, camouflaged in the background, using temporal information acquired via Gaussian mixture models and optical flow. We evaluated the proposed system on two underwater video datasets i.e., the LifeCLEF 2015 benchmark from the Fish4Knowledge repository and a dataset collected by The University of Western Australia (UWA). We achieve fish detection F-scores of 95.47% and 91.2%, while fish species classification accuracies of 91.64% and 79.8% on both datasets respectively. To our knowledge, these are the best reported results on these datasets, which show the effectiveness of our proposed approach.

1. Introduction

It is inevitable to perform regular sampling of fish populations to monitor the trends in relative abundance, composition, size, and fish biomass in oceans and fresh water bodies (Jennings and Kaiser, 1998). Marine biologists and conservationists are very keen in using non-destructive and automatic ways for fish sampling to cut down labour costs and delays in achieving the outcomes as a result of manual sampling (McLaren et al., 2015). Several approaches using non-destructive sampling and automatic fish detection and species classification in underwater videos have been used (Harvey and Shortis, 1995; Shortis and Abdo, 2016). However, challenges posed by variations in terms of poor light conditions, water murkiness, occlusions and jitters in imagery, similarity in shape and texture among different fish species, moving aquatic plants and background confusion hamper the use of such techniques in real-life scenarios due to below-par accuracy.

Previously, several approaches have been used for automatic fish recognition using different image and video processing algorithms. Rova et al. and Spampinato et al. proposed image processing and pattern recognition based techniques for unconstrained underwater fish classification by capturing the scale texture pattern and fish outline features (Rova et al., 2007; Spampinato et al., 2010) . Huang et al. presented an efficient approach to minimize the effect of environmental variability by employing decision trees with Support Vector Machine (SVM) trained on fish images (Huang et al., 2015). They improved the fish species classification accuracy to 74.8% on dataset of 24,000 images covering 15 different fish species as compared to standard Principal Component Analysis (PCA) features trained with SVM proposed by (Duan and Keerthi, 2005). Similarly, a technique which utilises Efficient Match Kernels (EKM) and Kernel Descriptors (KDES) as fish features and trained a multi-class SVM classifier to perform underwater fish identification achieving 84.4% classification accuracy on dataset of 50,000 fish images of 10 different species (Palazzo and Murabito, 2014). Hsiao et al. proposed an approach using temporal

* Corresponding author.
E-mail address: ahmad.salman@seecs.edu.pk (A. Salman).

https://doi.org/10.1016/j.ecoinf.2020.101088
Received 23 August 2019; Received in revised form 16 March 2020; Accepted 26 March 2020
Available online 03 April 2020

Table 1
Summary of LCF-15 and UWA fish datasets.

Data	No. of videos:	Video resolution	FPS	Annotated Video frames	No. of training samples	No. of test samples
LCF-15	93	640 × 480 320 × 240	24	14,765	29,965	12,813
UWA	4418	1920 × 1080	24	4278	3091	1327

Fig. 1. Various frames extracted from LCF-15 dataset videos showing background variability.

features of fish for their detection in videos (Hsiao et al., 2014). They used adaptive background subtraction using Gaussian Mixture Models (GMM) to model background pixels in the video frames. It is assumed that the video frames used for training GMM consist of pure background without any fish instance. GMM detects motion in video frames (most probably by fish) when a certain region of the frame fails to match the background model distribution. They achieved promising results on the fish detection task with average success rate of 83.99% on a dataset comprising several underwater videos recorded near the Southern Taiwan region. Palazzo et al. also proposed a similar approach on covariance modelling of background and foreground (fish instances) in the video frames using colour and scale texture features (Palazzo and Murabito, 2014). They were able to achieve an average detection accuracy of 78.01% using a dataset of four underwater videos with high contrast and illumination variations, strong background object movements, dynamic textures and highly populated background instances. To date, GMM is considered as state-of-the-art approach for fish detection in underwater videos (Spampinato et al., 2014). Moreover, optical flow, another motion based approach, has also been used recently for underwater fish detection and tracking in (Shin, 2016) where they proposed realization of robot fish motion-tracking control using the optical flow as an object detecting algorithm in the controlled aquarium environment. We will compare the performance of various GMM-based and other popular detection and classification techniques with ours in a later section.

Recently, Deep Neural Networks (DNNs) have been used for fish detection and species classification tasks. Salman et al. provided a comparative analysis between different conventional machine learning techniques and a deep network called Convolution Neural Network (CNN) on Fish4Knowledge LifeCLEF 2014 and LifeCLEF 2015 fish images (Salman et al., 2016). Similarly, a fish detection and classification using YOLO (You Only Look Once), a variant of CNN was proposed in (Sung et al., 2017). They trained the YOLO network on 829 images and achieved fish species classification accuracy of 93% on 100 images. In another work, YOLO was applied to detect fish instances in three different datasets and achieved mean average precision of 53.92% (Xu and Matzner, 2018), while YOLO with parallel correlation filter appeared in (Liu et al., 2018) to detect and track underwater fish instances. They showed simulation results on two underwater fish datasets and proposed the effectiveness of their fish tracker. (Jäger et al., 2016) also used LifeCLEF 2015 for fish species classification task using multi-class SVM on features extracted from deep CNN based on AlexNet architecture and achieved F-score of 73.5% on underwater videos. Mokhov and Serguei built an application for fish classification task on

Table 2
The LCF-15 fish dataset. Each row contains information for a single species.

Fish name	Profile image	Sample image from dataset	Occurance in training split	Occurance in test split
Abudefduf vaigiensis			371	159
Acanthurus nigrofuscus			2054	880
Amphiprion clarkii			2693	1154
Chaetodon lununatus			3911	1675
Chaetodon speculum			114	48
Chaetodon trifascialis			1419	608
Chromis chrysura			2721	1166
Dascyllus aruanus			2649	1134
Dascyllus reticulatus			7841	3360
Hemigymnus malapterus			253	108
Myripristis kuntee			2355	1009
Neoglyphidodon nigroris			150	64
Pempheris vanicolensis			734	314
Plectrogly-Phidodon dickii			2338	1001
Zebrasoma scopas			312	133

LifeCLEF 2015 dataset based on pattern recognition pipeline implemented in an open-source modular A* Recognition Framework (MARF) (Mokhov, 2015). They were able to achieve 74.3% F-score on species recognition task. On the other hand Zhuang et al. proposed automatic fish identification and species recognition on LifeCLEF 2015 dataset using advanced deep learning models with pre and post-processing to enhance accuracy of the system (Zhuang et al., 2017). They chose Single-Shot Multi-box Detector (SSD) algorithm (Liu et al., 2016) to differentiate the regions between foreground fish and background. They used ResNet-10 (Hariharan and Girshick, 2017), yet another CNN variant, as a classifier for fish species identification and achieved 83.8% F-score. LifeCLEF 2015 data was again employed in fish detection and classification system which was based on GoogleNet CNN architecture. They achieved 84.8% F-score on species classification task (Choi, 2015).

The motivation of our work is to combine motion detectors in highly complex environments and deep CNN like YOLO to detect fast moving fish in dynamic backgrounds. Temporal features extracted through

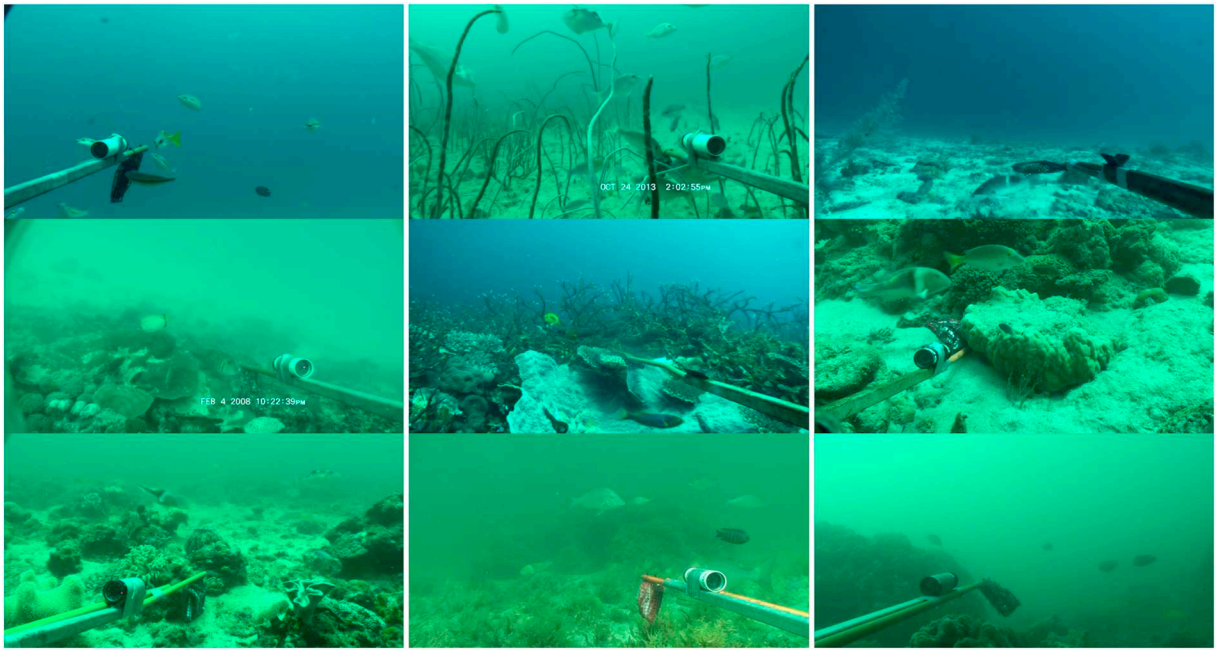

Fig. 2. Various frames extracted from UWA dataset videos showing challenges in terms of poor visibility.

GMM and optical flow are first classified, which then supplement YOLO when it fails to detect fish where they are camouflaged in the background, lack clearly defined shape due to murkiness of water and also moving in front of extremely contrastive background. On the other hand, YOLO assists in suppressing the false alarms of GMM/optical flow motion detectors by accurately detecting static or slowly moving fish. Thus, the gap between all above mentioned approaches and our proposed is filled with a hybrid scheme which brings forward the advantages of both techniques in achieving favourable detection and classification results for the task in hand.

2. Materials and procedures

2.1. Dataset

We have used two datasets to justify the effectiveness of our proposed technique. The first dataset is taken from LifeCLEF 2015 (hereinafter called LCF-15) fish task[1] which consists of 93 annotated videos comprising instances of 15 different species. This dataset is derived from a larger repository of underwater videos called Fish4Knowledge (Fisher et al., 2016). With over 700,000 unconstrained underwater videos captured with stationary cameras, Fish4Knowledge is collected over a period of five years to monitor the marine ecosystem of Taiwan coral reefs. This region is home to the largest fish biodiversity environments in the world with over 3000 different fish species. The second dataset is collected by our research group in the University of Western Australia (UWA) comprising 4418 videos with low visibility of fish instances in deep waters. Once again using stationary cameras, videos are collected from several baited remote underwater video sampling programs that occurred between Cape Naturaliste and the Houtman Abrolhos islands in the temperate and subtropical coastal waters of Western Australia during 2011 to 2013. Details regarding the study areas, camera system and video analysis is given in (Siddiqui et al., 2017).

Table 1 summarizes the technical information regarding both datasets which are used in our work. LCF-15 dataset is labelled by marine experts and used to benchmark video based fish detection and species classification methods. In total, there are 14,765 annotated video frames containing multiple fish instances. From these frames, we extracted 42,778 fish patches and out of those, 29,965 are kept for training and 12,813 are reserved for testing. Moreover, there are 22,444 fish patches of 15 different fish species also given in the dataset separately which can be used for training classification systems if required. Fig. 1 shows some examples of LCF-15 datasets exhibiting variation in surrounding environment, fish pattern, shape, size and video quality while Table 2 enlists species and their train-test splits given in the dataset.

Fig. 2 depicts some examples from the UWA dataset. There are 4278 annotations provided in this dataset. Due to severe luminosity challenges, we find it interesting to test our proposed approach on the specific environment of this dataset. Therefore, similar to LCF-15 dataset, we use the UWA dataset for fish detection and their species classification according to the given annotations. The list of fish species and their train-test split is provided in Table 3. Fig. 3 exhibits the intra and inter-species variations in some image samples, which together with poor luminosity makes this dataset challenging to get high performance in detection and classification tasks.

2.2. Proposed algorithm

In order to perform fish detection followed by classification on both LCF-15 and UWA datasets, we propose a hybrid approach which is based on motion-based features acquired from GMM and optical flow while spatial features from YOLO.

GMM is an unsupervised machine learning algorithm to learn first and second order statistics of input data (Stauffer and Grimson, 1999; Zivkovic and Van Der Heijden, 2006). In our case it is utilised to model the background using training data so that the foreground can be segmented. To achieve this, each pixel value in the fixed location of the background acts as a feature and multiple such pixels for several frames are combined to form a feature vector. Therefore, we end up with a total number of feature vectors that equals the total number of pixels in a video frame. The scheme is explained in (Salman et al., 2019). GMM requires a certain amount of data for training purpose to effectively estimate the mean and covariance of the background which comprises non-fish objects including coral reefs, kelp, sea grass beds and other

[1] http://perceive.dieei.unict.it/index–dataset.php?name = Fish$_S$pecies

Table 3
The UWA fish dataset. Each row contains information for a single species.

Fish name	Profile image	Sample image from dataset	Occurrence in training split	Occurrence in test split
Abudefduf bengalensis			136	58
Carangoides fulvoguttatus			148	64
Choerodon cyanodus			154	66
Choerodon rubescens			140	60
Coris auricularis			127	55
Lethrinus atkinsoni			146	62
Lethrinus nebulosus			146	62
Lethrinus sp			137	59
Lutjanus carponotatus			139	59
Pagrus auratus			140	60
Pentapodus emeryii			129	55
Pentapodus porosus			144	62
Plectropomus leopardus			134	58
Scarus ghobban			132	56
Scombridae spp			125	53
Thalassoma lunare			139	59
Other	 7		875	379

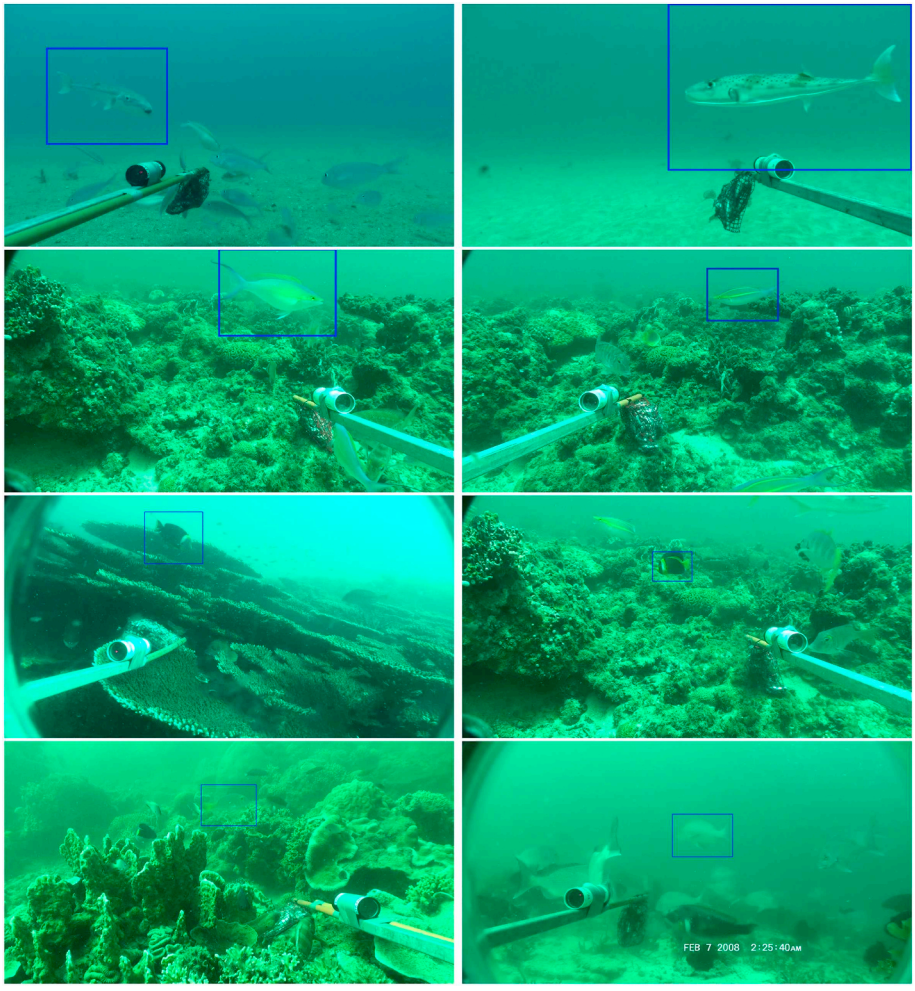

Fig. 3. Intra-species dissimilarity in UWA dataset where images in each row belong to same fish species called *Lagocephalus secleratus* (first row) and *Pentapodus emeryii* (second row). Similarly, Inter-species similarity comprising *Chaetodontoplus personifer*, *Chaetodontoplus duboulayi*, *Lethrinus atkinsoni* and *Lethrinus nebulosus* (third and forth row) where fish of different species appear similar.

Fig. 4. Combination of GMM and Optical flow output after ResNet-50 classification is merged with the output of YOLO in preferential manner to get the final output.

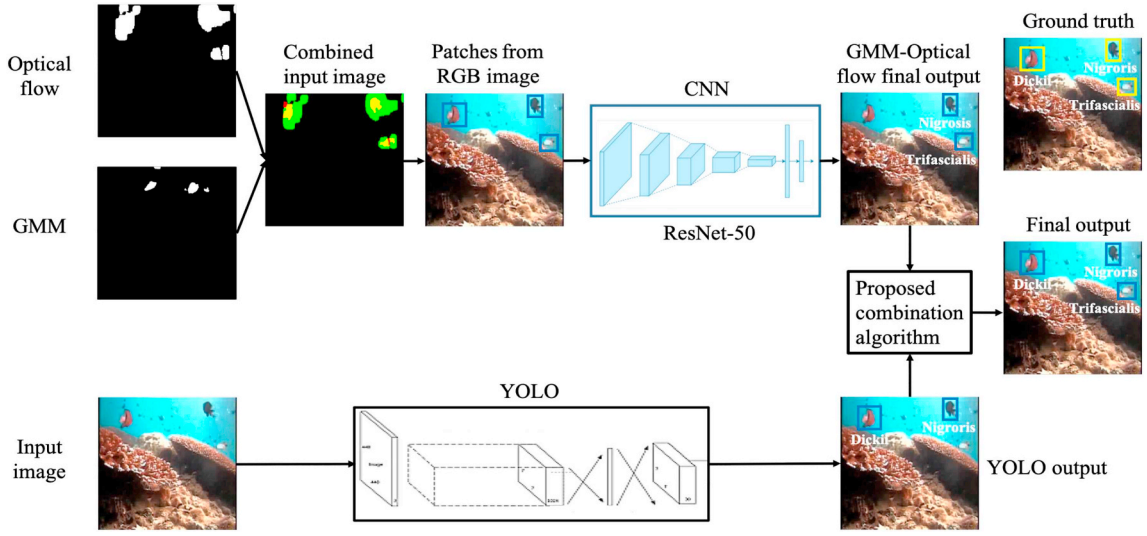

Fig. 5. Block diagram of our proposed algorithm. The outputs from boosted GMM and optical flow are combined for blob analysis and corresponding blobs from RGB frame are extracted and passed to ResNet-50 classifier for fish classification. The RGB frame is also passed to YOLO classifier for fish localization and classification. Results from YOLO output and ResNet-50 output are then merged in a YOLO preference manner for overlapped regions to get the final output.

aquatic plants, sessile invertebrates such as sponges, gorgonians and ascidians, and the physical structure of the sea floor. The statistical pattern representing fish movement (foreground in our case) is usually different from the pattern of fixed objects like coral reefs and seabed structures and also objects showing limited movement like to and fro motion of plants and dynamic light beams resulting from disturbance in the water surface.

One disadvantage of relying only on GMM to acquire motion-inspired features is that it requires pure background images without any fish instances for training a background model. However, LCF-15 dataset cannot provide this luxury as all recorded videos are random excerpts from Fish4Knowledge repository and with visual inspection we couldn't find adequate number of frames lacking fish to train ideal GMM models. This issue is less observed in UWA dataset though. Therefore, as a result of an imperfect background modelling, GMM sometimes causes miss detection of fish, marking them as background objects. To address the weakness of GMM, we exploit optical flow as an additional way to capture fish motion-related features in underwater videos. As explained in (Burton and Radford, 1978; Warren and Strelow, 2013), optical flow detects motion pattern generated by moving objects in consecutive video frames and therefore captures even a slight movement minimizing miss detections. However, it is also prone to false motion detection apart from fish since the background has dynamics in terms of moving aquatic plants and specially light variations due to disturbance at the water surface.

The blobs, representing the foreground fish candidates, extracted by GMM and optical flow by working on input data video frames are merged together to form a hypothetical RGB image. The green and red channels of the image are filled with blobs from GMM and optical flow respectively, while the third blue channel is left black. Therefore, the resultant image contains the regions of interest (RoIs) for foreground fish candidates. These RoIs are mapped back to original RGB video frame to extract the coloured patches which are further subjected to a classifier to recognise either fish species or non-fish entity. We have used ResNet-50 CNN (He et al., 2016a) as classifier which is trained on fish and non-fish (for negative examples in training) patches from the training set. The non-fish patches include random extracts of background from video frames. Here we have utilised (29,965 + 22,444) fish patches for training ResNet-50 (see Section 2.1). The parameters of GMM and optical flow for fish detection are chosen as such to detect even minute movement in the subsequent video frames. This produces high recall rates. In the next step, the precision of the system is increased by fine-tuning and refining regions in the frames to classify moving fish using ResNet-50. This constitutes the first branch of our proposed architecture.

In the second branch, we have incorporated YOLOv3 version of YOLO deep CNN (Redmon and Farhadi, 2018) which is based on Darknet-53 neural network architecture. Our solution lies in employing deep CNN architectures which are highly non-linear parametric models capable of extracting and learning complex yet abstract features when trained with sufficient amount of data (Chatfield et al., 2014; LeCun et al., 2004). Environmental factors in underwater scenes such as changes in light intensity, variability in size, shape and orientation of fish, poor quality of image and noise are the factors that introduce non-linearity in the data (Bengio et al., 2009). Since all of these challenges are encountered in the videos of the datasets in hand, conventional machine learning and computer vision algorithms fail to yield favourable results. Typically, a CNN is composed of different types of layers with each layer performing a different operation. For example, a convolution layer performs mathematical operation of convolution, which is used to find spatial correlation between input image and trainable network weights. After convolution layers, data is passed through the nonlinear activation layer to induce nonlinearity in data distribution. Pooling layers follow the activation layers to reduce the dimensionality of data and refine output features of convolution layers. Finally, fully-connected layers are responsible for predicting class label. The architectural details of YOLO deep CNN are given in (Redmon and Farhadi, 2018).

The two branches comprising GMM-optical flow-ResNet and YOLO are combined in a preferential manner to get our proposed final output. As per our observation with rigorous cross-validation, we prefer species classification results of YOLO over ResNet-50 where fish candidate blobs from both algorithms overlap each other. The disjoint blobs from both algorithms are taken as it is in the final output. This idea is depicted in Fig. 4. The entire system is utilised for fish detection and classification as depicted in Fig. 5 and elaborated in Algorithm 1.

Data: Underwater fish imagery, extracted foregrounds from GMM and optical flow
Result: Fish localization and classification
i = 0;
TotalCount = 0;
while *i < TotalFrameCount* **do**
  Read RGB frame, GMM and optical flow corresponding outputs;
  Apply YOLO fish localization and classification on RGB frame;
  Blob analysis on merged GMM and optical flow outputs;
  Extract corresponding RGB patches using blobs coordinates;
  if *ContourArea > 600 & ContourArea < 12000* **then**
    ResNet-50 fish classification model on RGB patches;
  end
  if *Overlapping instances between GMM-optical flow and YOLO outputs* **then**
    Accept YOLO classification result and discard GMM-optical flow;
  end
  else
    Merge outputs from GMM-optical flow and YOLO ;
  end
  i=i+1;
end
TruePositives = 0;
FalsePositives = 0;
for *All labeled frames* **do**
  GroundTruth = Load(current frame annotations);
  result=Load(corresponding frame GMM-optical flow-YOLO merged output);
  FalsePositives+ = Abs(len(GroundTruth)-len(result));
  TotalCount+ = len(GroundTruth);
  for *All GroundTruth instances* **do**
    match = 0;
    if *overlap is true* **then**
      match+ = 1;
      if *Label is matched* **then**
        TruePositives+ = 1;
      end
      else
        FalsePositives+ = 1;
      end
    end
    if *match = 0 (no overlap with any final output)* **then**
      FalsePositives+ = 1;
    end
  end
end
FalseNegatives = TotalCount - TruePositives;
Presision = float(TruePositives)/(TruePositives + FalsePositives);
Recall = float(TruePositives)/(TruePositives + FalseNegatives);
F-Score = float(2 × Precission × Recall)/(Precission + Recall);

Algorithm 1: Proposed algorithm

Table 4
Runtime parameters for the training of GMM, optical flow, ResNet-50 and YOLO models for LCF-15 and UWA datasets.

Parameter	Value
GMM (LCF-15/UWA)	
IntialFrames for training	100/200
MinBackground Ratio	0.7/0.7
Initial variance	0.013/0.013
No. of Gaussians	20/20
Blob threshold	200/10,000
Optical flow	
Pyramid size	0.95
Pyramid layer	10
Window size	15
Iterations per pyramid	3
ResNet-50	
Batch size	64
Subdivisions	16
Learning rate	0.01
Total iterations	150,000
Hue	0.1
Saturation	0.75
Exposure	0.75
Aspect parameter	0.75
YOLO	
Frame resizing	640 × 640
Batch norm decay	0.0005
Saturation	0.1
Exposure	1.5
Hue	1.5
Initial learning rate	0.0001
Learning rate scale factor	@5, 00 0.1@80,00 0.1@90,000

Table 5
F-score (in percentage) for fish detection task on LCF-15 and UWA datasets. Highest scores are highlighted in bold. GMM_R, OF_R, GMM_R-Y and OF_R-Y represent GMM_ResNet-50, Optical flow_ResNet-50, GMM_ResNet-50-YOLO and Optical Flow_ResNet-50-YOLO respectively.

Dataset	GMM_R	OF_R	YOLO	GMM_R-Y	OF_R-Y	Our proposed
LCF-15	71.03	55.65	90.67	92.18	91.47	**95.47**
UWA	32.49	58.8	85.75	86.35	89.6	**91.2**

Table 6
F-score (in percentage) for fish species classification on LCF-15 and UWA datasets. Highest scores are highlighted in bold. GMM_R, OF_R, GMM_R-Y and OF_R-Y represent GMM_ResNet-50, Optical flow_ResNet-50, GMM_ResNet-50-YOLO and Optical Flow_ResNet-50-YOLO respectively.

Dataset	GMM_R	OF_R	YOLO	GMM_R-Y	OF_R-Y	Our proposed
LCF-15	39.05	28.76	83.73	90.21	90.02	**91.64**
UWA	8.9	10.24	72.17	77.17	78.34	**79.8**

Algorithm 1. Proposed algorithm.

2.2.1. Utility

We have packaged our proposed algorithm in a form of a software solution that is available for deployment and ready to use by marine scientists for automatic fish detection and classification. Although we have trained and evaluated our system on LCF-15 and UWA dataset, the software can be used for any dataset for which a step-by-step guide is available along with the source code.[2]

In this work we have used a computer system with a Intel®Cor™e-i7 processor, 32 GB random access memory (RAM) and NVIDIA GeForce GTX 1080 Ti graphical processing unit (GPU). For YOLO, we have utilised Tensor Flow deep learning libraries[3] while GMM and optical flow source codes are taken from publicly available authors' repository.[4]

3. Results

Selection of parameters for GMM, optical flow, ResNet-50 and YOLO are critical to get optimum performance. Parameters for all these algorithms especially GMM are chosen using ten-fold cross-validation on training dataset after experimenting with their different ranges. The parameters of optical flow, ResNet-50 and YOLO are generic with little or no change from their default values which work on most of the image-related datasets.

To model video backgrounds using GMM, a few trainable parameters are required. Optical flow, on the other hand, does not require any training data but simply uses adjacent frames to calculate a motion representation. Similarly, ResNet-50 classifier is trained on the training split of fish instances from LCF-15 as well as UWA dataset to segment out and classify fish patches from motion-based detections from GMM and optical flow. YOLO-based detection and classification system is trained directly on annotated frames from the training splits of LCF-15 and UWA datasets to perform detection as well as classification on the test splits using RGB frames. Table 4 enlists parameter settings for GMM, optical flow, ResNet-50 and YOLO respectively.

The comparative analysis is reported for GMM, optical flow, standalone YOLO trained on raw images from videos and our proposed algorithm. Moreover, GMM-YOLO and optical flow-YOLO combination is also tried to advocated the effectiveness of combining all three as constituents of our proposed system. Here again, It is worth mentioning that GMM and optical flow are not classification algorithms as they just predict the moving region or RoIs. Therefore, for classification of fish, ResNet-50 is used on top of GMM and optical flow. On the other hand YOLO in itself is an end-to-end detector and classifier.

The performance metric used in our experiments is F-score (Derczynski, 2016). An average detection F-score of 95.47% was achieved on LCF-15 dataset while 91.2% on UWA dataset respectively using our proposed system, surpassing its individual components i.e., GMM, optical flow and YOLO. Similarly, for fish species classification task, an average F-score of 91.64% and 79.8% are achieved by our proposed system on LCF-15 and UWA datasets respectively. Fish detection and species classification scores are tabulated in Tables 5 and 6 respectively. A visual illustration of fish detection comparison is shown in Fig. 6.

To validate the effectiveness of our system, in Table 7, we have drawn a comparison with various published approaches which have been used for motion-based fish detection and classification for video imagery such as (Choi, 2015; Jäger et al., 2016; Mokhov, 2015). The comparison is made on the LCF-15 dataset for which we can directly tabulate the published scores. For fair comparison, we have used similar training and test protocols in experiments for our proposed algorithm. It is evident that our proposed system outperforms all others in the overall average F-scores. Fig. 7 shows the species-wise accuracies (in terms of count of correct classifications) yielded by our proposed system and other approaches. No direct comparison can be made on UWA dataset for fish species classification as there is no comparative work available in this regard.

[2] https://github.com/ahsan856jalal/Fish-detection-and-classification-using-HOGY.git

(*footnote continued*)
[3] https://www.tensorflow.org
[4] https://github.com/andrewssobral/bgslibrary

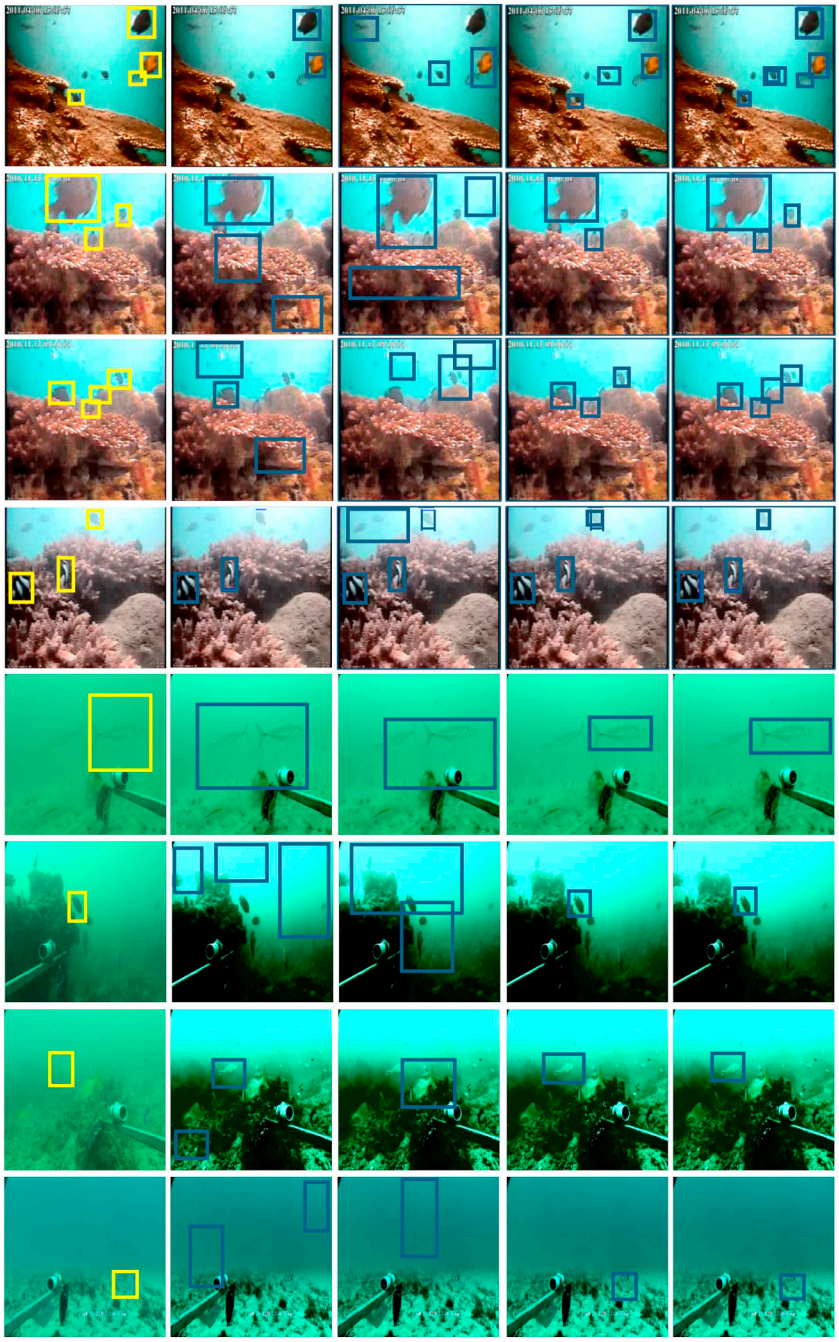

Fig. 6. Comparative performance for fish detection task on LCF-15 dataset (first four rows) and UWA dataset (last four rows). From left to right, bounding boxes representing ground truth, detections by GMM, optical flow, YOLO and our proposed system respectively.

Table 7
F-scores (in percentage) for different methods on LCF-15 datasets for fish detection followed by classification. The results of our proposed system are copied from Table 6 for easy comparison in this table.

Dataset	Choi. [GoogleNet]	Jäger et al. [AlexNet + SVM]	Mokhov [SSD+ ResNet-10]	Our proposed
LCF-15	83.8	75.5	74.3	**91.6**

4. Discussion

There are several hundred fish species in the oceans and freshwaters around the world which are at the verge of extinction due to over-fishing, climate change or environmental pollution. This unavoidable situation makes it inevitable to devise and deploy efficient fish sampling methods for rapid estimation of fish biomass and abundance in water bodies. This can alert marine scientists and conservationists for taking appropriate measures for monitoring fish population and their habitat.

The most important outcome of our work is an automatic fish detection and species classification methodology that yields high accuracy

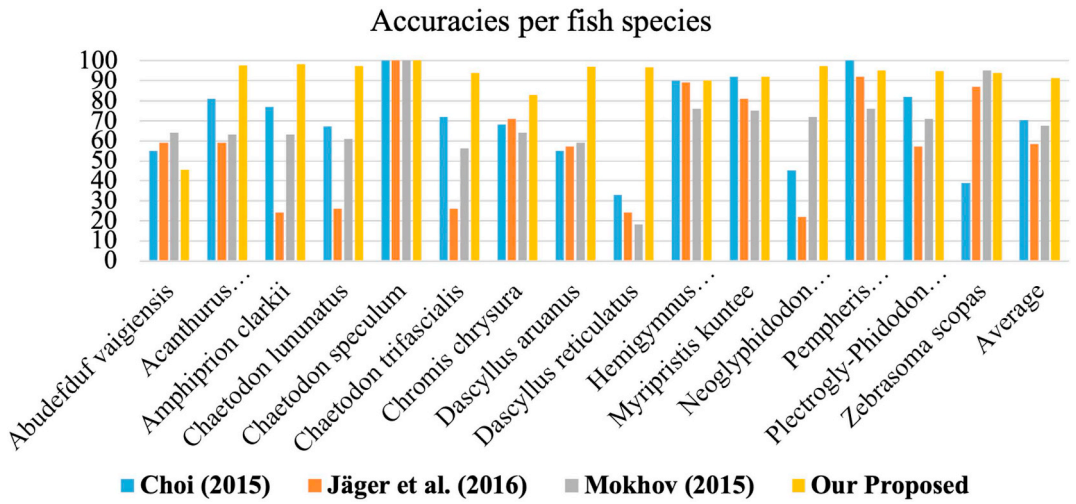

Fig. 7. Species-wise classification accuracies (count of correct classifications) by various techniques on LCF-15 dataset.

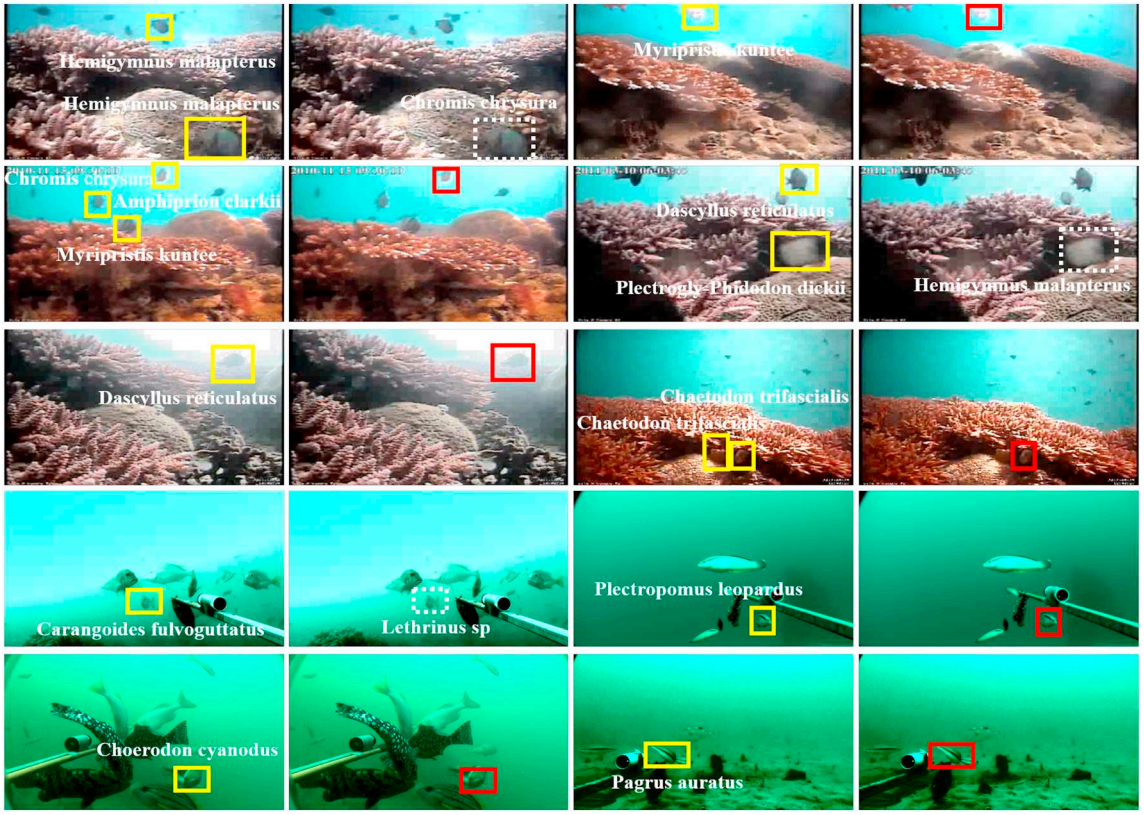

Fig. 8. Various instances from LCF-15 and UWA datasets where our proposed system was unable to detect or classify fish species correctly due to environmental variations. First and third columns represent the ground truth frames while the second and fourth columns are the outputs from our proposed model. Red solid boxes represent those instances where the system fails to detect fish at first place whereas white dotted boxes represent those instances where fish is detected but the species is misclassified. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and is relatively robust against underwater environmental variations. We have proposed a scheme to acquire motion-based features through GMM and optical flow which are classified into fish species label using ResNet-50 model, and combine ResNet-50 model's outcome with fish-dependent features extracted through YOLO to detect and classify fish in underwater videos. The motivation behind using this combinational approach is to supplement the shortcomings of one technique from strengths of the other. Motion-based feature extractors like GMM and optical flow help in detecting relatively fast moving fish instances where YOLO deep CNN often fails to localize fish in the presence of moving aquatic plants, seaweeds and most importantly when the moving fish camouflage in the background. Similarly, YOLO takes centre stage in detecting and classifying those fish that show static profile or partial movement and therefore, missed by GMM and optical flow. Furthermore, CNN models, in general, are more robust in extracting texture and shape related features of fish in the severe contrast and luminosity variations. The aforementioned attributes are not modelled effectively by conventional shallow machine learning algorithms and image processing techniques as discussed in (Hinton and Salakhutdinov, 2006; Larochelle et al., 2009).

In this work, we have presented our results in two ways. First, we have compared the fish detection performance of our proposed system and its individual components i.e., GMM, optical flow, YOLO and their combination in the form of GMM-YOLO and optical flow-YOLO on LCF-15 and UWA datasets as tabulated in Table 5. Secondly, on the same two datasets, we presented species classification scores for fish detected in the first step as given in Table 6. Our results show that our proposed system outperforms all the individual component algorithms and their combinations in both fish detection and their species classification tasks. On the other hand, performance of YOLO is superior to GMM and optical flow on average due to its capability to model complex data distribution in a supervised training paradigm that ensures output features to be rich in information related to the texture and shape of fish. This observation is consistent with the comprehensive study about deep neural networks (LeCun et al., 2015). GMM is also a trainable algorithm and it yields better results compared to optical flow on LCF-15 dataset due to the clarity of scenes where fish is more visible and also due to the ability to model backgrounds to segment out the foregrounds (fish). In contrast, optical flow is not a trainable system rather a scheme to estimate a change in pixel location in two adjacent video frames. However, it outperforms GMM in UWA dataset in both fish detection and species classification tasks. The reason for this behaviour is poor visibility of fish in UWA scenes (see Fig. 2) where they blend with the background. Therefore, fish is also modelled as background by GMM without clear distinction and this results in a large number of miss detections. The same reason also contributes towards miss classification of fish species on this dataset. We have observed that fish species including *Abudefduf vaigiensis*, *Chaetodon speculum*, *Neoglyphidodon nigroris* and *Zebrasoma scopas* remain stationary or exhibit partial movement in several videos of LCF-15 dataset. Moreover, they are relatively harder to discriminate from the background which results in poor performance of GMM and optical flow as they solely depend on moving profile of these fish species. YOLO and our proposed system, on the other hand, make use of their ability to meticulously extract texture and shape-dependent features to complement GMM and optical flow hence, show much better performance in classifying these fish species. Similarly, our proposed system makes use of motion information contributed by GMM and optical flow in addition to the texture and shape-dependent features extracted by YOLO in classifying *Acanthurus nigrofuscus*, *Amphiprion clarkii*, *Chaetodon lununatus*, *Myripristis kuntee* and *Pempheris vanicolensis* where it exceeds standalone YOLO with significant margin. In the case of UWA dataset, GMM and optical flow fail to yield acceptable scores for fish detection and their species classification tasks although, they still contribute to enhance the performance of our proposed algorithm as compared to standalone YOLO. In general, the UWA dataset remains challenging due to extremely poor luminosity conditions in general. The comparative analysis of GMM-YOLO and optical flow-YOLO combinations coupled with ResNet-50 classifier also goes in our favour, which validates the effectiveness of the incorporation of both GMM and optical flow with YOLO deep neural network in our proposed system. The benefits of our proposed framework come out with greater margin in LCF-15 dataset which comes with relatively more clearer videos as compared to UWA dataset..

From our extensive literature review, we came up with the research work from (Choi, 2015; Jäger et al., 2016; Mokhov, 2015). Their methodologies are coherent to our work in terms of approach and they have also used LCF-15 dataset for fish detection and their species classification tasks. Therefore, we have compared our proposed system with all of these state-of-the-art techniques, which rely on deep learning and utilise several deep CNN architectures including R-CNN (Ren et al., 2015), Alexnet (Iandola et al., 2016), GoogleNet (Szegedy et al., 2015) and SSD (Liu et al., 2016). The performance comparison is presented in Table 7 and Fig. 7 which show that our proposed system outperforms all the other techniques with significant margin and justify our idea of using motion information together with physical appearance of fish.

The parameters of the GMM were carefully chosen to produce best possible results by altering the variance for model fitting and the number of frames for training the model on each video. A fewer number of training frames per video results in degraded performance. However, increasing the number of training frames beyond 50 does not improve the overall performance significantly. Similarly, for our hybrid system and also for the standalone GMM and optical flow classification on raw images, various state-of-the-art convolution neural networks are tried that include ResNet-152 (He et al., 2016b), VGG-16 (Han et al., 2015), AlexNet (Iandola et al., 2016) and DenseNet (Iandola et al., 2014). The results generated by these CNNs were comparable with ours without any significant improvement however, our choice needs less processing power in training and testing compared to the others. It is worth mentioning here that the GMM chosen for our hybrid system differs with the one listed in Tables 5 and 6 as it is engineered to produce higher recall rates to cover each and every possible pixel motion in the video by both fish and non-fish objects. The Resnet-50 classifier then learn to select the relevant motion candidate through refining the results generated by the GMM and optical flow.

There are some instances where our proposed system either fails to detect fish at first place or misclassify species due to extreme variations in underwater scenes as shown in Fig. 8.

5. Conclusion

In this paper, we have developed an automatic fish detection and their species classification technique, which utilises an advanced machine learning approach called YOLO for detection and species classification of fish based on their shape and textural features. To enhance the performance we incorporate motion-inspired information generated through GMM and optical flow algorithms, which support YOLO system in capturing those freely swimming fish which have either poor visibility due to water murkiness, low resolution imagery and low light conditions or due to camouflage in the background. We have evaluated our proposed system on LCF-15 benchmark dataset and also on our own collected UWA dataset and achieved promising results beating the current state-of-the-art systems. The proposed system utilises relatively more computational power as it involves complex machine learning tool as compared to conventional computer vision and image processing approaches. However, due to the advent of faster microprocessors and GPUs, our system can achieve near real-time performance and can be used by marine scientists for automatic fish fauna sampling in a non-destructive way. In future, we plan to carry forward the same motivation of fish detection/classification in unconstrained underwater videos and propose further enhancement using fish tracking with a more optimised machine learning solutions and develop a system for more accurate estimation of an abundance of a certain species of fish.

Acknowledgments

The authors acknowledge NVIDIA Corporation, USA for providing latest GPUs under their GPU Grant Program for this work and IGNITE National Technology Fund, Pakistan, Grant No. ICTRDF/TR&D/2013/61 for providing financial support.

References

Bengio, Y., et al., 2009. Learning deep architectures for ai. Found. Trends Mach. Learn. 2 (1), 1–127.

Burton, A., Radford, J., 1978. Thinking in Perspective: Critical Essays in the Study of Thought Processes. Vol. 646 Routledge.

Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A., 2014. Return of the Devil in the Details: Delving Deep Into Convolutional Nets, arXiv preprint arXiv:1405.3531.

Choi, S., 2015. Fish identification in underwater video with deep convolutional neural network: Snumedinfo at lifeclef fish task 2015. In: CLEF (Working Notes).

Derczynski, L., 2016. Complementarity, f-score, and NLP evaluation. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). European Language Resources Association (ELRA), Portorož, Slovenia, pp. 261–266. URL. https://www.aclweb.org/anthology/L16-1040.

Duan, K.-B., Keerthi, S.S., 2005. Which is the best multiclass svm method? An empirical study.

In: International Workshop on Multiple Classifier Systems. Springer, pp. 278–285.
Fisher, R., Chen-Burger, Y., Giordano, D., Hardman, L., Lin, F., 2016. Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data , Intelligent Systems Reference Library. Springer International Publishing URL. https://books.google.com.pk/books?id=j846jwEACAAJ.
Han, S., Mao, H., Dally, W.J., 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In: arXiv preprint arXiv:1510.00149.
Hariharan, B., Girshick, R.B., 2017. Low-shot visual recognition by shrinking and hallucinating features. In: ICCV, pp. 3037–3046.
Harvey, E., Shortis, M., 1995. A system for stereo-video measurement of sub-tidal organisms. Mar. Technol. Soc. J. 29 (4), 10–22.
He, K., Zhang, X., Ren, S., Sun, J., 2016a. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
He, K., Zhang, X., Ren, S., Sun, J., 2016b. Identity mappings in deep residual networks. In: European Conference on Computer Vision. Springer, pp. 630–645.
Hinton, G.E., Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507.
Hsiao, Y.-H., Chen, C.-C., Lin, S.-I., Lin, F.-P., 2014. Real-world underwater fish recognition and identification, using sparse representation. Ecol. Inform. 23, 13–21.
Huang, P.X., Boom, B.J., Fisher, R.B., 2015. Hierarchical classification with reject option for live fish recognition. Mach. Vis. Appl. 26 (1), 89–102.
Iandola, F., Moskewicz, M., Karayev, S., Girshick, R., Darrell, T., Keutzer, K., 2014. Densenet: implementing efficient convnet descriptor pyramids. In: arXiv preprint arXiv:1404.1869.
Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and < 0.5 mb model size. In: arXiv preprint arXiv:1602.07360.
Jäger, J., Rodner, E., Denzler, J., Wolff, V., Fricke-Neuderth, K., 2016. Seaclef 2016: Object proposal classification for fish detection in underwater videos. In: CLEF (Working Notes), pp. 481–489.
Jennings, S., Kaiser, M.J., 1998. The effects of fishing on marine ecosystems. In: Advances in Marine Biology. Vol. 34. Elsevier, pp. 201–352.
Larochelle, H., Bengio, Y., Louradour, J., Lamblin, P., 2009. Exploring strategies for training deep neural networks. J. Mach. Learn. Res. 10 (Jan), 1–40.
LeCun, Y., Huang, F.J., Bottou, L., 2004. Learning methods for generic object recognition with invariance to pose and lighting. In: Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. Vol. 2 IEEE pp. II–104.
LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436.
Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. Ssd: Single shot multibox detector. In: European Conference on Computer Vision. Springer, pp. 21–37.
Liu, S., Li, X., Gao, M., Cai, Y., Nian, R., Li, P., Yan, T., Lendasse, A., 2018. Embedded online fish detection and tracking system via yolov3 and parallel correlation filter. In: OCEANS 2018 MTS/IEEE Charleston. IEEE, pp. 1–6.
McLaren, B.W., Langlois, T.J., Harvey, E.S., Shortland-Jones, H., Stevens, R., 2015. A small no-take marine sanctuary provides consistent protection for small-bodied by-catch species, but not for large-bodied, high-risk species. J. Exp. Mar. Biol. Ecol. 471, 153–163.
Mokhov, S.A., 2015. A marfclef approach to lifeclef 2015 tasks. In: CLEF (Working Notes).
Palazzo, S., Murabito, F., 2014. Fish species identification in real-life underwater images. In: Proceedings of the 3rd ACM International Workshop on Multimedia Analysis for Ecological Data. ACM, pp. 13–18.
Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. In: arXiv preprint arXiv:1804.02767.
Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99.
Rova, A., Mori, G., Dill, L.M., 2007. One fish, two fish, butterfish, trumpeter: recognizing fish in underwater video. In: MVA, pp. 404–407.
Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., Harvey, E., 2016. Fish species classification in unconstrained underwater environments based on deep learning. Limnol. Oceanogr. Methods 14 (9), 570–585.
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 J. Mar. Sci. https://doi.org/10.1093/icesjms/fsz025.
Shin, K.J., 2016. Robot fish tracking control using an optical flow object-detecting algorithm. IEIE Trans. Smart Process. Comput. 5 (6), 375–382.
Shortis, M., Abdo, E.H.D., 2016. A review of underwater stereo-image measurement for marine biology and ecology applications. In: Oceanography and Marine Biology. CRC Press, pp. 269–304.
Siddiqui, S.A., Salman, A., Malik, M.I., Shafait, F., Mian, A., Shortis, M.R., Harvey, E.S., 2017. H. editor: Howard Browman, Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. ICES J. Mar. Sci. 75 (1), 374–389.
Spampinato, C., Giordano, D., Di Salvo, R., Chen-Burger, Y.-H.J., Fisher, R.B., Nadarajan, G., 2010. Automatic fish classification for underwater species behavior understanding. In: Proceedings of the First ACM International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams, ACM, pp. 45–50.
Spampinato, C., Palazzo, S., Kavasidis, I., 2014. A texton-based kernel density estimation approach for background modeling under extreme conditions. Comput. Vis. Image Underst. 122, 74–83.
Stauffer, C., Grimson, W.E.L., 1999. Adaptive background mixture models for real-time tracking. In: cvpr. IEEE, pp. 2246.
Sung, M., Yu, S.-C., Girdhar, Y., 2017. Vision based real-time fish detection using convolutional neural network. In: OCEANS 2017-Aberdeen. IEEE, pp. 1–6.
Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9.
Warren, D.H., Strelow, E.R., 2013. Electronic Spatial Sensing for the Blind: Contributions from Perception, Rehabilitation, and Computer Vision. Vol. 99 Springer Science & Business Media.
Xu, W., Matzner, S., 2018. Underwater fish detection using deep learning for water power applications. In: arXiv preprint arXiv:1811.01494.
Zhuang, P., Xing, L., Liu, Y., Guo, S., Qiao, Y., 2017. Marine Animal Detection and Recognition with Advanced Deep Learning Models, Working Notes of CLEF 2017.
Zivkovic, Z., Van Der Heijden, F., 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recogn. Lett. 27 (7), 773–780.