ICES Journal of Marine Science

ICES Journal of Marine Science (2019), doi:10.1093/icesjms/fsz025

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system

Ahmad Salman[1]*, Shoaib Ahmad Siddiqui[2], Faisal Shafait[1], Ajmal Mian[3], Mark R. Shortis[4], Khawar Khurshid[1], Adrian Ulges[5], and Ulrich Schwanecke[5]

[1]*School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), Sector H-12, Islamabad 44000, Pakistan*
[2]*German Research Center for Artificial Intelligence (DFKI), Trippstadter Strasse 122, Kaiserslautern D-67663, Germany*
[3]*School of Computer Science and Software Engineering, The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia*
[4]*School of Science, RMIT University, GPO Box 2476, Melbourne, VIC 3001, Australia*
[5]*Faculty of Design—Computer Science—Media (DCSM), RheinMain University of Applied Sciences, Unter den Eichen 5, Wiesbaden D-65195, Germany*

*Corresponding author: *tel:* +92 0 51 90852559; *fax:* +92 0 51 8317363; *e-mail:* ahmad.salman@seecs.edu.pk.

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

Received 13 November 2018; revised 23 January 2019; accepted 24 January 2019.

It is interesting to develop effective fish sampling techniques using underwater videos and image processing to automatically estimate and consequently monitor the fish biomass and assemblage in water bodies. Such approaches should be robust against substantial variations in scenes due to poor luminosity, orientation of fish, seabed structures, movement of aquatic plants in the background and image diversity in the shape and texture among fish of different species. Keeping this challenge in mind, we propose a unified approach to detect freely moving fish in unconstrained underwater environments using a Region-Based Convolutional Neural Network, a state-of-the-art machine learning technique used to solve generic object detection and localization problems. To train the neural network, we employ a novel approach to utilize motion information of fish in videos via background subtraction and optical flow, and subsequently combine the outcomes with the raw image to generate fish-dependent candidate regions. We use two benchmark datasets extracted from a large Fish4Knowledge underwater video repository, Complex Scenes dataset and the LifeCLEF 2015 fish dataset to validate the effectiveness of our hybrid approach. We achieve a detection accuracy (F-Score) of 87.44% and 80.02% respectively on these datasets, which advocate the utilization of our approach for fish detection task.

Keywords: deep learning, fish assemblage, fish detection, fisheries management, neural networks, stock assessment, underwater video

Introduction

Monitoring the effect of preventive and recovery measures requires the estimation of fish biomass, and abundances by sampling their populations in water bodies like lakes, rivers and oceans on a regular basis (Jennings and Kaiser, 1998). This requires observation of the interaction of different fish species with changing environmental conditions. This is an essential process, especially in those regions of the world where certain species of fish are either threatened or at the risk of extinction due to habitat loss and modification, industrial pollution, deforestation, climate change, and commercial overfishing (Tanzer *et al.*, 2015). There is a well-established and increasing interest in using non-destructive fish sampling techniques by marine biologists and conservationists (McLaren *et al.*, 2015). Underwater video-based fish detection approaches have been used to achieve non-destructive and repeated sampling for many years (Harvey and

© International Council for the Exploration of the Sea 2019. All rights reserved.
For permissions, please email: journals.permissions@oup.com

Shortis, 1995; Shortis *et al.*, 2009). Manual processing of underwater videos is labour intensive, time consuming, expensive and prone to fatigue errors. In contrast, automatic processing of the underwater videos for fish species classification and biomass measurement is an attractive alternative. However, high variability in underwater environments due to changes in lighting conditions, clarity of water, and background confusion due to vibrant seabed structure pose great challenges towards automatic detection of fish. These factors result in a compromise on accuracy, which supports the continuing practice of less cost effective and cumbersome manual sampling and tagging of fish.

In general, automatic fish sampling involves the following three major tasks: (i) Fish detection, which discriminates fish from non-fish objects in underwater videos. Non-fish objects include coral reefs, aquatic plants, sea grass beds, sessile invertebrates such as sponges, gorgonians, ascidians, and general background. (ii) Fish species classification, which identifies the species of each detected fish from the predetermined pool of different species (Siddiqui *et al.*, 2017). (iii) Fish biomass measurement, using length to biomass regression methods (Froese, 2006). This article addresses the first task and the interested reader is referred to the literature for details of the following two steps in the overall process.

Various approaches have been followed for fish detection and consequently their assemblage estimation using image and video processing algorithms. Broadly speaking, these approaches can be divided into two categories based on the medium available for sampling, namely constrained and unconstrained sampling. In the former case, early attempts were made that involved detection of fish using information of their shape and colour (Strachan and Kell, 1995) or 3D modelling of fish to acquire features like height, width, or thickness (Storbeck and Daan, 2001). Harvey and Shortis (1995) presented an approach to acquire underwater images of fish under controlled conditions. This was achieved by making fish swim through a chamber with controlled illumination. Unconstrained underwater fish detection and classification does not assume any specific environmental conditions and, therefore, faces difficulty in achieving the required accuracy due to high variations in the aforementioned conditions. To address this problem, Spampinato *et al.* (2008) presented an image processing based method for fish detection and counting by capturing the texture pattern of fish in the natural underwater environment. They were able to achieve an average accuracy of about 84% on five underwater videos. In the past, several attempts have been made to solve the same problem in underwater videos using machine learning. Principal component analysis (Turk and Pentland, 1991), linear discriminant analysis (Mika *et al.*, 1999), and sparse representation-based detection (Hsiao *et al.*, 2014) presented some ways to capture fish-dependent features through mathematical modelling, which assumed independence of modelled fish with surrounding environments in videos. In other words, information like fish colour, texture, and shape was extracted with the prior assumption that foreground fish instance was easily distinguishable from the background. In reality, it is challenging to differentiate fish within underwater video/images due to camouflage with the background, poor visibility, and loss of contrast as a result of light attenuation through the water medium, low light, and water turbidity. In pursuit of suppressing the effects of environmental variability, Kernel Descriptors in Kernel density estimation (KDE) approach with colour information for background pixel modelling in images were used by Sheikh and Shah (2005). In contrast, texture-dependent features computed via local binary patterns for background modelling was proposed in Yao and Odobez (2007).

Background modelling is a popular technique to segment moving foreground objects from the background in video sequences. An approach using motion-based fish detection in videos was presented by Hsiao *et al.* (2014). This method implements background subtraction by modelling background pixels in the video frames using Gaussian mixture models (GMMs). Although training the GMM, it is assumed that subsequent frames of video lack fish instances. Motion is detected in the video frames (apparently from fish) when a certain region of the frame does not fit into the trained background model. This approach produces fish detection results with an average success rate of 83.99% on several underwater videos collected near southern Taiwan. A similar scheme was proposed on covariance modelling of background and foreground (fish instances) in the video frames using colour and texture features of fish (Palazzo and Murabito, 2014). Using a dataset of four underwater videos with a high variation in luminosity, strong background movements, dynamic textures, and rich background, they were able to achieve an average detection accuracy of 78.01%. Presently, GMM- and KDE-based fish detection approaches are considered state-of-the-art (Spampinato *et al.*, 2014). We will compare the performance of various state-of-the-art techniques with our proposed approach in a later section.

All of the above-mentioned machine learning and feature extraction approaches fall into the category of shallow learning architectures (Bengio, 2009). These techniques are unable to accurately model the complexity of fish-dependent features in the presence of highly variable and diverse environments, and therefore these video or image-based fish detection techniques exhibit low performance in real-world scenarios (Siddiqui *et al.*, 2017). In the last decade, deep learning has been at the centre of attention for many researchers developing detection and classification algorithms in computer vision. Marked by their ability to extract and model highly nonlinear data, deep architectures have been utilized in numerous tasks related to computer vision, including facial recognition, speech processing, generic object detection, and classification in video and still images producing state-of-the-art results (Lin *et al.*, 2015; Ren *et al.*, 2017). In realizing deep architectures, multilayer deep neural networks are among the most successful schemes capable of extracting task-dependent features in the presence of variability in the images. Most commonly used variants of deep neural networks include deep convolutional neural networks (CNNs) which are parametric neural network models capable of extracting task-specific features and are widely used in computer vision problems like object recognition in images and facial recognition (LeCun *et al.*, 2015).

Deep learning is being used lately to solve fish-related tasks (Moniruzzaman *et al.*, 2017). An important work using CNN was proposed by Sung *et al.*, (2017) to detect fish in underwater imagery with 65.2% average accuracy on a dataset containing 93 images having fish instances. The system was trained on raw fish images to capture colour and texture information for localizing and detecting fish instances in the images. In a similar work, deep region-based CNN (R-CNN) were used for the abundance estimation of fish from 4909 underwater images recorded in the coast of Southeast Queensland, Australia. In this work, an accuracy of 82.4% was reported using the R-CNN system tuned for

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

locating and detecting fish instances in an image with a unified network framework.

Despite the high accuracy achieved by the deep learning based fish species classification, the task of vision-based automatic fish detection in unconstrained underwater videos is still under extensive investigation as most of the previous attempts reported results on relatively small datasets with a limited variety in the surrounding environment. Therefore, it is important to judge the robustness and effectiveness of any system in a large dataset with a high degree of environmental variation.

In this article, we address fish detection in highly diverse and unconstrained underwater environments and propose a hybrid system based on explicit motion-based feature extraction followed by a final detection phase using deep CNNs. In the first step, we use background subtraction by modelling still pixels of the video frames using GMMs. These models represent pixels related to a range of coral reefs, seabed features, and aquatic plants. Foreground objects are segmented from the background based on the motion in the scene that does not match the background model. To enhance the quality of the extracted features in each video frame, we concatenate the GMM candidate output blobs with the moving candidates generated by optical flow, a well-established approach used for motion detection in videos (Brox *et al.*, 2004). However, due to poor image quality, noise and background confusion, the detection remains far from perfect. To address this problem, we tune the parameters of GMM and optical flow systems to generate high recall by trying various values of the number of Gaussian distributions, initial variance, blob size and sensitivity in case of GMM, as well as pyramid size, number of pyramid layers, and window size in case of optical flow. The details of these parameters are given in Zivkovic and Heijden (2006) for GMM and in Beauchemin and Barron (1995) for optical flow. Specifically, in this step, all entities that exhibit even a slight movement are detected as fish. In the second step, we discriminate all the candidate regions in the video frames as fish and non-fish entities using a CNN architecture arranged in a hierarchical fashion to fine tune the detection system. Our CNN is trained using a supervised training style in which the GMM and optical flow blobs acts as the input while ground truth blobs (given in the training data) acts as the desired output. We worked on two different datasets; the Fish4Knowledge Complex Scenes Dataset, where the aim is fish detection with videos arranged into seven different categories based on the variation in the underwater environment; and the LifeCLEF 2015 (LCF-15) dataset, which is also designed for the detection of freely swimming fish in video sequences. These datasets contain marine scenes and species; unfortunately, there is no public domain benchmark datasets available containing underwater recordings in fresh water bodies.

The contribution of this work is to overcome the main challenge faced by the conventional motion detection and image classification approaches using deep learning. These deep learning modules are trained to select the relevant information from the data and minimize confusion which contributes to false alarms or missed detections. This approach improves the detection and classification accuracies especially in the data marked by high environmental variability like unconstrained underwater videos of fish. Our novelty lies in the proposed hybrid setup to mine the relevant motion information content by pooling the information generated by GMM and optical flow and refining the outcome by deep CNNs. Our approach is capable of detecting fish in the video in its stationary or moving state with region-based feature localization. This equips our detection system with motion-influenced temporal information that is not available otherwise, in order to enhance detection performance in cases where fish is occluded or camouflaged in the background.

Material and methods

Dataset

We use two benchmark datasets in our study, both of which are specially designed to provide a resource for testing algorithms for detection and classification of fish in images and video sequences and have been used for benchmarking a number of approaches. The first dataset is used for the fish detection task and is a collection of 17 videos under different environmental conditions (http://f4k.dieei.unict.it/datasets/bkg_modeling/). The second dataset is taken from the LCF-15 fish task (http://www.imageclef.org/lifeclef/2015/fish). This dataset contains 93 underwater videos comprising 15 different fish species. Both datasets are derived from a very large fish database called Fish4Knowledge (Fisher *et al.*, 2016). With over 700 000 underwater videos in unconstrained conditions, the Fish4Knowledge dataset has been collected over a period of 5 years to monitor the marine ecosystem of coral reefs around Taiwan. This region is home to one of the largest fish biodiversity environments in the world with more than 3000 fish species.

The first dataset, dubbed FCS (Fish4Knowledge with Complex Scenes) hereinafter, comprises seven sets of selected videos recorded in typical underwater conditions addressing complex variability in the scenes. Thereby, the environmental variations provide a major challenge for fish identification and are categorized as follows:

(1) ***Blurred***, comprising three low contrast, blurred videos.

(2) ***Complex background***, composed of three videos with rich seabed structures that provide a high degree of background confusion.

(3) ***Crowded***, in which three videos with a high density of moving fish in each video frame imposes specific challenges for fish detection techniques, especially when it comes to high recall and precision in the presence of occluding objects.

(4) ***Dynamic background***, in which two videos are provided with rich textures of coral reef background and moving plants.

(5) ***Luminosity variation*** composed of two videos involving sudden luminosity changes due to surface wave action. This phenomenon can induce false positives in detection due to moving light beams.

(6) ***Camouflage foreground***, two videos are chosen, addressing the challenge of detecting fish camouflaged in the presence of textured and colourful background.

(7) ***Hybrid***, in which two videos are selected to show a combination of all the above-mentioned conditions of variability.

Table 1 summarizes the technical information regarding both datasets used in this article. For the FCS dataset, complexity is specifically depicted for all seven environmental conditions. The LCF-15 dataset is used to detect fish instances in the video i.e. to count all the fish in the video regardless of their species. Of the 93 videos given in LCF-15, 20 are used for training the computer

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

Table 1. Information about LCF-15 and FCS fish datasets.

Dataset	No. of videos	Format	Resolution	Frames/sec	No. of labelled fish instances
LCF-15	93	FLV	640 × 480, 320 × 240	24	42 493
FCS	17	FLV	640 × 480, 320 × 240	24.5	1 328

Figure 1. Sample images to illustrate the high variation in underwater environment. The first two rows depict seven categories of the FCS dataset from left to right top to bottom being *Blurred, Complex background, Dynamic background, Crowded, Luminosity variation, Camouflage foreground,* and *Hybrid*. The last row shows an excerpt from different videos of the LCF-15 dataset.

vision or machine learning modules, while the remaining 73 videos are set aside for testing/validating the developed algorithms. In total, there are 9000 annotated fish instances available in the LCF-15 training set, and 13 493 annotated instances for the test videos. All these videos are manually annotated by experts. Apart from videos, there are 20 000 labelled still images in LCF-15, where each image comprises of a single fish. These images can also be used to supplement the training set if required. Thus, in total there are 42 493 labelled fish instances in videos and still images in the LCF-15 dataset. The FCS dataset is also designed and used for the fish detection task. Therefore, ground truth is available for all moving fish, frame by frame in each video. There are a total of 1328 fish annotations available for the FCS dataset. Figure 1 shows some video frames extracted from FCS and LCF-15 datasets exhibiting the variation in the surrounding environment, fish patterns, shape, size, and image quality.

Proposed algorithm

To perform fish detection, we propose a hybrid system based on the initial motion-based feature extraction from videos using GMM and optical flow candidates. These feature images are combined with raw greyscale images and fed to the CNN system to mark final detected fish. Therefore, our proposed hybrid fish detection system is made up of three components i.e. GMM, optical flow and a CNN.

Gaussian mixture modelling

In machine learning, GMM is an unsupervised generative modelling technique to learn first and second order statistical estimates of input data features (Stauffer and Grimson, 1999; Zivkovic and Heijden, 2006). This approach and its variants are frequently used in computer vision and speech processing tasks. GMM represents a probability density function $P(x_t)$ at time t of data x as a weighted sum of multiple individual normal distributions $\eta(x_i)$ for pixel i. Thereby, each density is characterized by the mean and covariance of the represented data. Using a combination of individual Gaussian densities, any probability distribution can be estimated with arbitrary precision (Reynolds and Rose, 1995). In our case, each pixel value with a fixed location in the video frame acts as a feature. Multiple such values from successive frames are combined to form a feature vector. As elaborated in Figure 2, we end up with a total number of feature vectors that

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

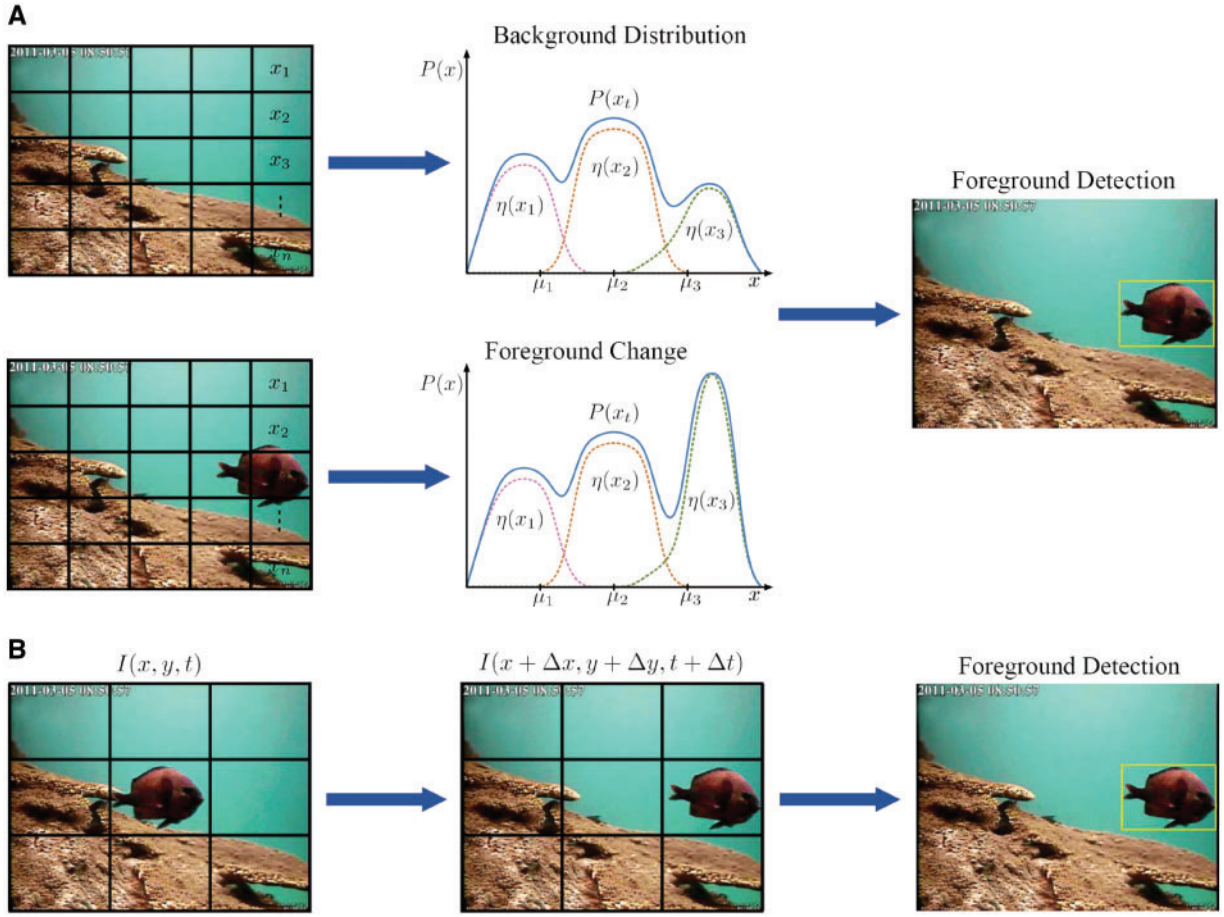

Figure 2. (A) Illustration of background subtraction and foreground segmentation using GMM which detects any change in the foreground by matching it with the background model. (B) Motion detection in an optical flow setup to estimate the direction of moving objects in two dimensions (x, y) for consecutive frames in time *t* of a video sequence.

equals the total number of pixels in a video frame. The GMM requires a certain amount of training data to effectively estimate the mean and covariance of an object class. For fish detection in videos, there are two classes i.e. background and foreground. Ideally, the background in underwater videos should cover everything in the frame but moving fish. For example, seabed structure, coral reef, aquatic plants, and wave action causing variation in light intensity are categorized as background. Freely moving fish, on the other hand, constitute as foreground. The GMM is used to learn the background features in a statistical model using mean and covariance values of pixel data and separate them from the foreground pixel distribution. In other words, any random and sudden change in the pixel value of a frame causes a mismatch with the background model of that pixel and hence, a motion is assumed to be detected. The statistical pattern of foreground (fish in our case) movement is usually different from the pattern of fixed objects like seabed structures, coral reefs and also objects with confined movement like to and fro motion of plants and refracted light rays from the surface. The outputs of the GMM are the candidate blobs marked by bounding boxes localizing the moving objects in a frame (see Figure 2).

The video frames that are used to train the GMM should not include any fish instance but only the background. However, it is very difficult to capture such videos in a natural environment as fish can appear in any number of frames. When a GMM is trained on videos that do not have pure background but also some fish, the fish will also be modelled as background resulting in misdetections in the upcoming test frames.

Optical flow

To compensate for this shortcoming of GMM, we additionally extracted optical flow features which are purely generated by motion occurring in the underwater videos (see Figure 2). Optical flow is a 2D motion vector in the video footage caused by the 3D motion of the displayed objects (Warren and Strelow, 1985). There are various methods to estimate optical flow. We opted for a simple yet effective method where motion is detected between two successive video images taken at times t and $t + \Delta t$ at every position using Taylor series approximation with partial derivatives with respect to spatial and temporal coordinates (Beauchemin and Barron, 1995). A region of interest (ROI) in a video frame at time t and coordinates x, y can be represented in terms of intensity as $I(x, y, t)$. After any motion in the next frame, the intensity becomes $I(x + \Delta x, y + \Delta y, t + \Delta t)$ where the notation Δ represents the change in coordinates and time. Based

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

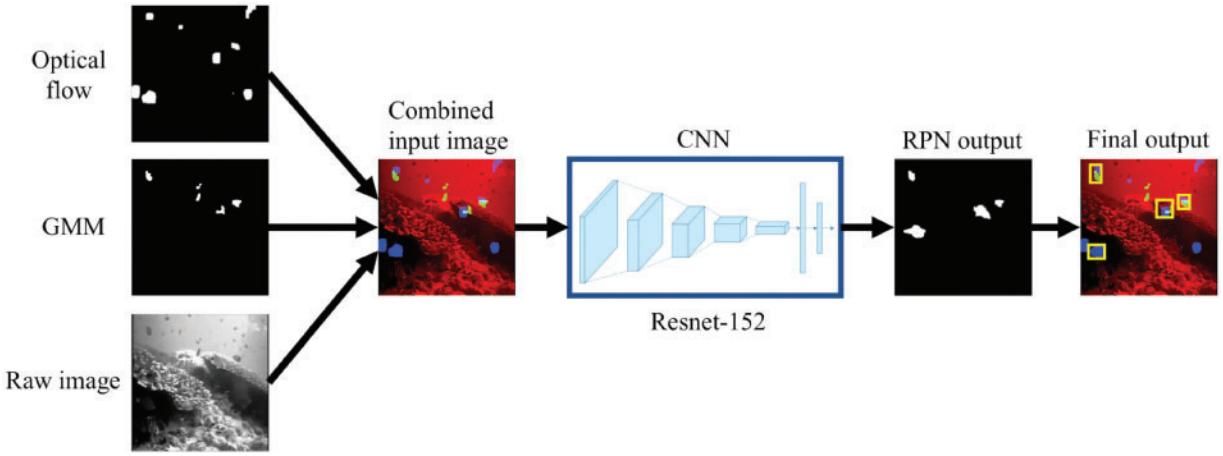

Figure 3. The proposed hybrid system, where ResNet-152 CNN is trained on images that are created by combining the motion-influenced outputs of GMM and optical flow algorithms with raw greyscale video images. This is analogous to three-channel RGB image.

on the motion constraint, optical flow can be determined as described in, for example, Beauchemin and Barron (1995).

Optical flow depends on the analysis of the consecutive frames to estimate the difference in the intensity vector of a pixel at a particular location. However, such an analysis is also prone to false motion detection apart from fish when applied to a dynamic background with moving aquatic plants and abrupt luminosity variation due to disturbance at the water surface. The parameters of the GMM and optical flow algorithm are chosen such that even the smallest movements are detected. In other words, the sensitivity of the algorithms is maximized, producing a high rate of false alarm in addition to detecting fish instances leading to high recall rates. In the next step, the precision of the system is further increased by fine-tuning and refining regions in the frames to localize moving fish. This requires a robust detector to categorize fish motion in complex and variable environments. We propose the use of a R-CNNs (hereinafter referred to as R-CNN) trained on images, created by combining candidate regions generated by the GMM and optical flow together with the original greyscale images in a supervised learning setup.

Region-based convolutional neural network

A deep CNN is a nonlinear parametric neural network model capable of extracting and learning complex yet abstract features of the input data. Variations in the lighting condition, size, shape, and orientation of the fish, poor image quality and significant noise are the factors that introduce nonlinearity into the data (Bengio, 2009). Since all of these challenges are encountered in the videos recorded in an unconstrained underwater scenario, it is difficult for conventional machine learning algorithms to model data features in the presence of such nonlinearity. However deep neural architectures, especially CNNs, learn to extract invariant and unique features of the objects of interest in data when properly trained with a sufficient amount of labelled data (LeCun *et al.*, 2004; Simonyan and Zisserman, 2014). The deep architecture exemplified by the R-CNN employed in our study is a hierarchical parametric model composed of two modules. The first module is a generic deep CNN trained for generic object recognition on a very large dataset called ImageNet (Deng *et al.*, 2009). Smaller than the first module CNN, the second module is another CNN, which acts as the object detector and called region proposal network (RPN) (Ren *et al.*, 2017). It selects candidate regions in the feature space of the input image in which a motion is likely to have occurred.

The entire system is used for detecting moving objects as depicted in Figure 3. The first module utilizes the concept of transfer learning (Siddiqui *et al.*, 2017). It learns characteristic feature representation of the object of interest in the input image in order to recognize and classify the objects in the imagery. In transfer learning, a CNN pretrained on totally different, yet relevant dataset, is utilized as a generic feature extractor for the dataset of interest. In our case, the CNN was trained on the vast ImageNet dataset that contains 1.2 million images of a very large and diverse number of objects. This dataset is not related or designed for fish species recognition or fish detection in underwater videos. However, it provides a high degree of variability to detect generic objects with different backgrounds in input images based on their texture, size and shape features. Once the network is trained, it can be applied to a different dataset, in our case on underwater video imagery of fish, as a feature extractor. Transfer learning is suitable for the applications where a large amount of training data is not available to train the deep CNNs (Siddiqui *et al.*, 2017). This is exactly the problem in the current underwater datasets. Training on such relatively small datasets (see Table 1) overfits a deep CNN to generate better performance on training dataset and fails on previously unseen test datasets. In other words, the training dataset is so small that the CNN is able to memorize it and produce good results only on the training dataset. We utilize a deep CNN known as ResNet-152 as the pretrained model (He *et al.*, 2016). The parameters of this network are further refined by including examples of our fish dataset video imagery in training. This network is composed of an input layer, various hidden layers and an output layer to process an input image to obtain its output feature representation (LeCun *et al.*, 2004). Starting from an input layer that represents the pixels of an image, the hidden layers are interconnected by a set of weights that are tuneable as a result of the training procedure. Thereby, each hidden layer represents a higher-level form of feature representation. There are several types of hidden layers used in our network, e.g. a convolution layer that performs the mathematical operation of convolution between image pixels (values of the

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

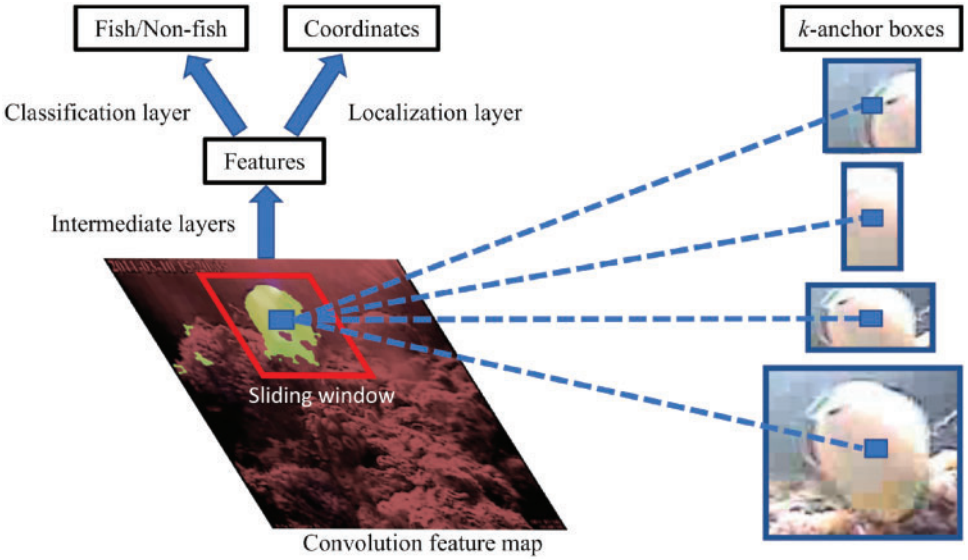

Figure 4. Illustration of the functionality of a RPN to detect and localize fish. The proposal with the best fit to the fish instance is selected out of *k* choices.

input layer) or feature values (values of the hidden layers) with the weight vectors. Convolution is generally used in image processing for noise reduction or detecting features such as edges or other patterns (LeCun *et al.*, 1989). In a CNN, convolution is followed by a nonlinear activation layer to induce nonlinearity in the feature vectors. There are several types of nonlinear functions, e.g. ReLUs (rectified linear units), Sigmoid and Hyperbolic Tangent (LeCun *et al.*, 2004; Simonyan and Zisserman, 2014; He *et al.*, 2016). The choice of the nonlinear function depends on the data distribution and nonlinearity of the input data. Due to the saturating regions of the Tangent Hyperbolic and Sigmoid function, the ReLU function is the defacto-standard in the latest state-of-the-art models. Max pooling and average pooling layers sift out the most prominent values from the output of nonlinearity inducing layers based on maxima or an averaging operations to reduce the dimension of feature vectors and retain useful information while discarding the redundancy. The final layer is the output layer which usually is a classification layer with output nodes equal to the number of desired classes for a given dataset. Each output node produces a score or probability for the associated class. The predicted label is then matched with the ground truth label to calculate accuracy.

ResNet-152 is a modular deep CNN with various hidden layers. The architecture is designed to process images of size 224 $\times$ 224 given the fact that this resolution is enough to extract useful features within reasonable computational time. Thus, after applying five pooling layers, the feature map size shrinks to 7 $\times$ 7 which can be processed by fully-connected layer of 1000 label prediction nodes, since ResNet-152 was designed to train on a subset of ImageNet dataset with 1000 classes.

The complete architecture details of ResNet-152 can be found in He *et al.* (2016). The arrangement of the above-mentioned layers in this architecture is experimentally determined to yield greater success on visual features from the large-scale ImageNet dataset. Using this network as a pretrained model on our FCS and LCF15 fish datasets, an informative visual representation of fish objects and their motion can be extracted. After applying the pretrained ResNet-152 network on the input which is a concatenation of the raw greyscale video frames and the motion candidates generated by GMM and optical flow, we get the output features. This three-input combination is alternative to the standard three-channel RGB image. The output features extracted by applying ResNet-152 are fed into RPN to generate candidate regions where fish might be present. This is achieved by sliding a small window of size 3 $\times$ 33 $\times$ 3 on each of the feature maps to produce k proposals, called anchor boxes, of different aspect ratio and scale. We use three different scales (128×128, 256×256, 512×512) each with 3 different aspect rations (2:1, 1:1, and 1:2) to make $k = 9$ proposals. The aim of using different proposals is to capture fish of different sizes that may appear in an image. These proposals are then classified with a binary classification layer of the RPN to detect the ROI. Another sibling layer of RPN outputs coordinate encodings for each classified proposal. This operation is depicted in Figure 4. The ROIs proposed by the RPN are pooled using an ROI pooling layer and passed onto the final classification head which refines and classifies the proposed ROI into the actual number of classes present at hand, namely fish and non-fish. The complete network is trained in an end-to-end fashion using the features generated by ResNet-152 model as the input and the corresponding ground truths provided by the dataset. While training, we employ an error backpropagation algorithm (Hinton *et al.*, 2006).

As mentioned earlier, the parameters of the GMM-based motion detection algorithm are chosen such that it detects even a very small motion by either fish or non-fish objects producing high false alarm or recall rates. The R-CNN architecture, which is a combination of the ResNet-152 based feature extraction and RPN followed by a final classification layer for localizing moving objects, refines the output of the GMM and optical flow motion candidates. Therefore, the information of motion coming from GMM and optical flow is fed into R-CNN to finally detect and localize objects. Apart from motion candidates generated by GMM and optical flow, the use of greyscale raw images in combination with motion candidates as input to the ResNet-152 CNN helps in

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

Table 2. Performance analysis of individual components of our proposed hybrid framework in comparison to their joint accuracy.

Dataset	GMM	Optical flow	R-CNN	Our hybrid system
FCS				
Blurred	77.80	45.94	85.62	**86.76**
Complex background	75.94	49.77	52.74	**89.54**
Crowded	74.41	67.48	53.23	**84.27**
Dynamic background	64.30	44.62	62.06	**90.36**
Luminosity change	59.07	58.67	70.17	**81.44**
Camouflage foreground object	70.03	67.00	66.25	**89.97**
Hybrid videos	75.50	59.44	64.90	**91.50**
Average	71.01	56.13	64.99	**87.44**
LCF-15	76.21	52.73	77.30	**80.02**

F-scores (in percentage) for three different methods i.e. GMM (Stauffer and Grimson, 1999), Optical flow (Warren and Strelow, 1985), and R-CNN (Ren *et al.*, 2017) on FCS and datasets for seven categories of video complexity. Highest scores are highlighted in bold

preserving the textural information of fish appearing in the video frame, which increases the capability of the network to induce separability between fish and non-fish objects. The reason of using greyscale image instead of RGB to fine-tune the R-CNN is the observation that colour information in the employed datasets are not distinct enough to enhance the accuracy of detection as the background is also vibrant in colours. Moreover, doing so increases the computational overhead.

In this work, we utilized computer systems equipped with Intel Core-i7 processors and Nvidia Titan X graphical processing units (GPUs). The proposed system is trained and tested using TensorFlow deep learning library (https://www.tensorflow.org) with *Tf-faster-rcnn version* while GMM and optical flow source codes were taken from publicly available authors' repositories (https://github.com/andrewssobral/bgslibrary).

Fish detection system utility

Our software system is available for deployment and ready to be used by marine scientists for automatic fish detection in any dataset. As described in *Region-based convolutional neural network* Section, the deep network, which is the backbone of our algorithm, is pretrained on a large and generic object image repository called ImageNet and acts as a generic feature extractor. However, for using a pretrained network in such a transfer learning approach, the system must be fine-tuned to the actual datasets in hand; therefore, a complete end-to-end re-training on a new dataset is not required. In our case, we utilized FCS and LCF-15 datasets by updating the weights of the top fully connected layers of ResNet-152 of R-CNN, while keeping the lower layers intact. Furthermore, the GMM and optical flow algorithms can be used as is since they only require the available dataset to generate output. The source code for our proposed hybrid system is available for download from the following repository: https://github.com/ahsan856jalal/Fish-Abundance. Scientists can use this code off-the-shelf for fish/object detection in any dataset, video recordings or even still imagery.

Results

The underwater video background is modelled by GMM using training data from the initial few frames of the video while the remainder of the video is treated as the test dataset. Since each video in our datasets has a different background, we need to keep the first N frames of each video for background modelling. We take $N = 50$ in our experiments as this value was chosen on a trial basis to get optimum GMM performance on our datasets. Smaller values of N produces an inferior performance, while increasing beyond this value does not bring any improvement and increases GMM training time. Optical flow does not require any training data but simply uses adjacent frames to calculate a motion representation. The R-CNN, on the other hand, requires more data to tune the weight parameters for refined motion detection. The raw video images and the motion candidates generated by the GMM and optical flow are fed to the R-CNN for training. One video from each of the seven categories of FCS dataset is set aside for training the GMM and R-CNN. On top of that, GMM also requires the first 50 frames of each video to make a background model and to generate a blob of moving objects in the test frames. The LCF-15 dataset, on the other hand, is already segmented into training and test sets, 20 videos out of a total of 93 are used in training and the remainder is used for testing. Once again, GMM models for all 93 videos are created using the N initial frames. Table 2 lists the performance measure for the fish detection task as an F-measure (Palazzo and Murabito, 2014) for our proposed hybrid system and its independent constituents of GMM, optical flow and standalone R-CNN, which are trained on raw RGB images from videos.

The F-score is calculated as,

$$F = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

where

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

and

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

These scores are computed based on overlap between the areas of bounding boxes related to ground truths and detected fish. An average detection accuracy of 87.44% was achieved by the proposed hybrid system for the FCS dataset for all seven categories of environmental variation. In comparison, the GMM alone yielded an average accuracy of 71.01% exceeding the optical flow and standalone R-CNN with significant margins. We also performed similar experimentation on LCF-15 test dataset of 73 underwater videos. There, our proposed hybrid system outperforms all the other systems, yielding an accuracy of 80.02% as compared with 76.21, 52.73, and 77.30% by the GMM, optical flow and standalone R-CNN, respectively.

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 did not improve the overall performance significantly. Similarly, for our proposed hybrid system and also for the standalone R-CNN trained on the raw RGB images, various state-of-the-art pretrained CNNs were tried that include Inception-V4

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

(Szegedy *et al.*, 2016) and DenseNet (Huang *et al.*, 2017). All of these networks are pretrained on the ImageNet dataset with the same experimental settings. Moreover, different numbers of convolution layers for the RPN network were also evaluated and the choice of sliding window size of $2 \times 22 \times 2$, 4×4, and 5×5 was tested, with the performance maximized at $3 \times 33 \times 3$. The performance started to deteriorate slightly beyond the $3 \times 33 \times 3$ window size probably due to more overlap between intrinsic size of fish covering the frame of videos in our datasets. The results generated by Inception-V4 and ResNet-152 were comparable without any significant difference but the latter utilizes less processing power in training and testing compared with the former. Our implementation of optical flow on the other hand is a non-trainable processing approach for motion detection and therefore, does not have any trainable parameters. It is worth mentioning here that the GMM chosen for our proposed hybrid system differs with the one listed in Table 2 as its parameters were tuned to produce higher recall rates at the cost of decreased precision to cover maximum possible pixel motion in the video by both fish and non-fish objects. The CNN and RPN subsystems then learn to select the relevant motion candidate through refining the results generated by the GMM and optical flow. Figure 5 shows the performance outcome on a sample video for GMM, optical flow, R-CNN, and the proposed hybrid system for both the FCS and the LCF-15 datasets. It is evident that the optical flow algorithm generates more false alarms and is sensitive to even very slight motion, which can be attributed to disturbances in the scene or luminosity changes. On the other hand, the GMM and stand-alone R-CNN, which is only trained on raw RGB images, also exhibits false alarms and/or missed detection. However, they both yield better scores as compared with the optical flow due to effective background modelling and end-to-end supervised training; capabilities which optical flow lacks and are necessary to reduce the irrelevant motion created by non-fish entities. Our proposed hybrid system, on the other hand is successful in achieving the best performance (see Table 2).

To validate the effectiveness of our system, in Table 3 we have drawn a comparison with various published benchmark approaches which are frequently used for motion-based object detection in either still or video imagery. The comparison is made on the FCS dataset for which we can directly tabulate published scores by these techniques with the same experimental settings as ours. It is evident that our proposed hybrid system outperforms all others in most environmental conditions and the overall average F-scores. In another set of experimentation not reported here, we changed the train-test split in the FCS and LCF-15 datasets to calculate the detection scores but observed no significant change. This demonstrates a good generalization capability of our system.

Discussion

In this study, we have proposed a R-CNN to detect fish using enhanced features sensitive to natural fish motion in underwater videos in addition to features also representing distinguishable shape and textural information specific to fish in a supervised training hierarchy. The motivation behind using such a deep neural network is to model complex and highly nonlinear attributes in underwater imagery of fish. These attributes are not modelled effectively by conventional machine learning algorithms and image processing techniques (Hinton and Salakhutdinov, 2006; Larochelle *et al.*, 2009). This hybrid approach has resulted in a detection accuracy at reasonable level for use of this technique in fish detection from recorded videos.

The most important gain of this research is high detection accuracy of freely swimming fish. With our proposed hybrid system that incorporates motion sensitive features, taken as input to the R-CNN, we are able to achieve 87.44% detection accuracy on the FCS dataset. This performance exceeds the best reported results on this dataset by a significant margin. The second best average accuracy of 81.80% for all seven categories of variability has been produced using KDE to model background and foreground objects by capturing texture information in very low contrast regions of the video frames (Spampinato *et al.*, 2014). An interesting observation can be drawn from Table 3 for video classes *Dynamic background*, *Camouflage foreground object* and *Hybrid videos* that the performance gap between our proposed hybrid system and rest of the techniques is significantly wide. *Dynamic background* videos exhibit disturbance in water surface and movement of aquatic plants which causes confusion with motion of fish. Therefore, KDE, ML-BKG, and TKDE algorithms, which are based on estimating foreground data distribution by modelling background data, fails in separating motion of fish and non-fish objects. EIGEN and VIBE algorithms also produced poor performance due to similar reasons. Here, our proposed hybrid system utilizes the fish-dependent features captured through the R-CNN component using greyscale images in accurate detection of fish. On the other hand, fish in *Camouflage foreground object* videos are extremely hard to segregate from the background. Therefore, all the algorithms once again fail to yield better results due to inability in creating difference between foreground and background models. Here, our approach makes use of the motion information from GMM and optical flow to maximize its fish detection potential as shape, texture and colour of fish in this case resemble the background and are difficult to detect by the R-CNN component. Similarly, *Hybrid videos* combine all the challenges of other six classes and our proposed hybrid system is more effective than all other approaches. To further endorse the effectiveness of our approach, we employed a larger dataset by including LCF-15 with 93 videos. Our solution acquired an average accuracy of 80.02%. Table 2 lists the comparative performance of our proposed hybrid system with three other techniques, namely GMM, optical flow and R-CNN, which are the components of our overall system. The GMM outperforms optical flow and standalone R-CNN, trained on raw images, with a significant margin, for the FCS dataset. On the LCF-15 dataset, the GMM produces better results than optical flow and is comparable with the R-CNN. This signifies effective learning of the background model by the GMM on every new video sequence. The model covers all background variations exhibited by non-fish objects for a static underwater camera configuration, which assists in detecting even subtle movements through non-uniform change in pixel intensities that does not match with the distribution of background pixels.

We observe that the training of the GMM background model balances the rate of false alarm and misdetection, which produces a better F-score. The GMM and its variants are considered to give excellent performance in general for motion-based object detection tasks (Yong, 2013; Spampinato *et al.*, 2014). Optical flow, on the other hand, lagged behind all other methods in terms of performance on both datasets. The reason behind this behaviour can be attributed to the non-trainable structure of this algorithm, as the system cannot adapt to the dynamic environment in the videos. There is no learning involved to discriminate background

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

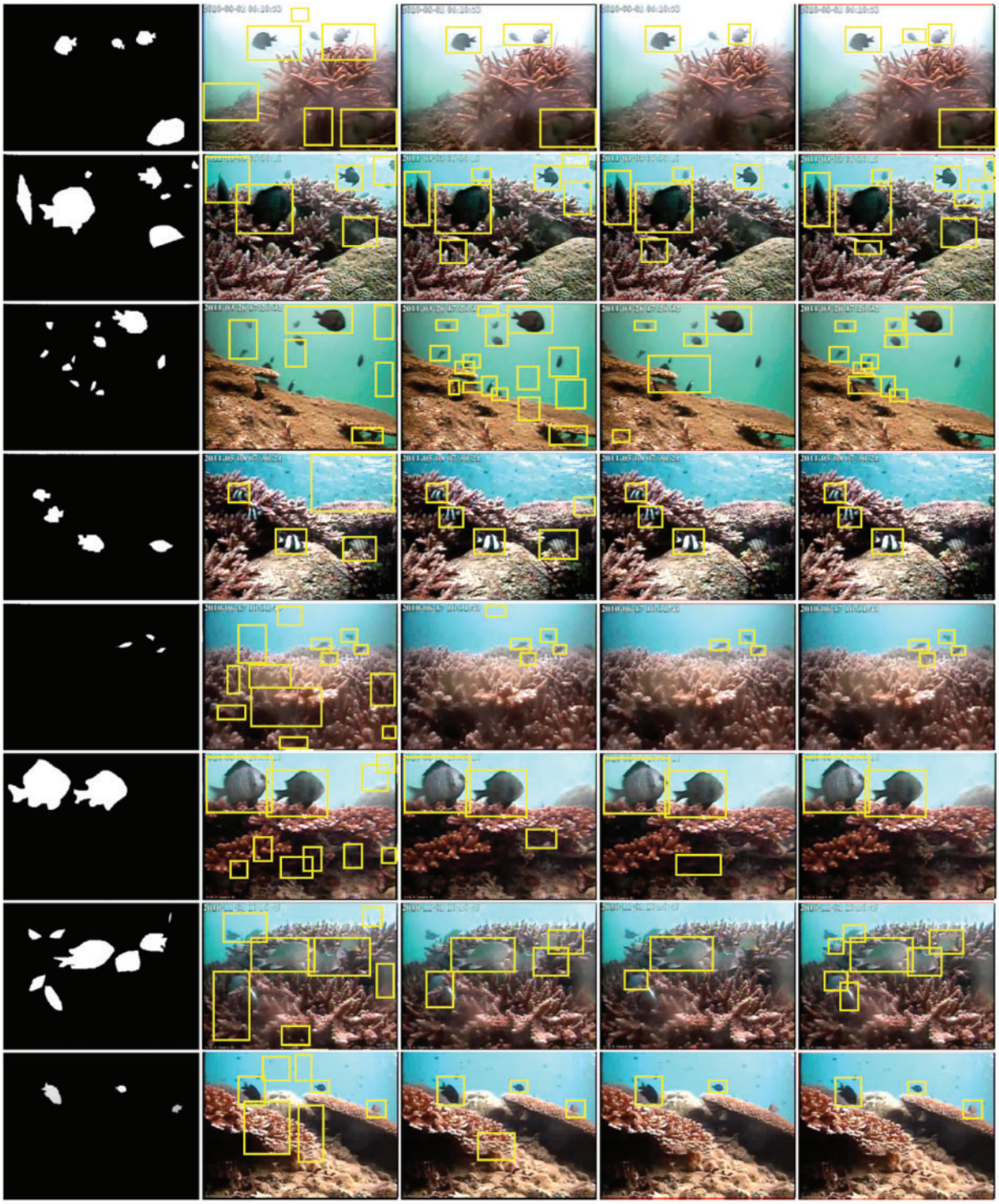

Figure 5. Example of fish detection outcomes by various algorithms. Left to right, ground truth, optical flow, GMM, stand-alone R-CNN, and proposed hybrid system on all seven categories of FCS dataset category (the first seven rows) and one video of LCF-15 dataset (the last row).

and foreground modelling, like that in the GMM and neural networks. Optical flow involves a direct comparison between adjacent frames of video and any slight disturbance in the pixel intensity, either due to fish or non-fish objects generating luminosity variation, translates into a valid motion. This gives rise to numerous false alarms, which results in a very high recall but consequently a low precision that ends up in producing low F-score. Since the datasets we have chosen involve high environmental variation, especially in the FCS dataset, optical flow fails to perform well as opposed to the other algorithms. On comparative grounds, both GMM and optical flow lags our proposed hybrid system for the FCS and LCF-15 datasets. Another

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

Table 3. *F*-scores (in percentage) for different methods on FCS datasets for fish detection, as given in Spampinato *et al.* (2014).

Video class	KDE	ML-BKG	EIGEN	VIBE	TKDE	Our hybrid system
Blurred	92.56	70.26	81.71	85.13	**93.25**	86.76
Complex background	87.53	83.67	74.78	74.17	81.79	**89.54**
Crowded	82.46	79.81	73.87	**84.64**	84.19	84.27
Dynamic background	59.13	77.51	71.48	67.01	75.59	**90.36**
Luminosity change	72.06	**82.66**	70.41	70.37	72.95	81.44
Camouflage foreground object	54.14	73.51	70.20	76.30	82.23	**89.97**
Hybrid videos	85.69	72.20	80.69	79.75	82.63	**91.50**
Average	76.22	77.08	74.73	76.76	81.80	**87.44**

The results of our proposed system are copied from Table 2 for easy comparison in this table. Highest scores are highlighted in bold.

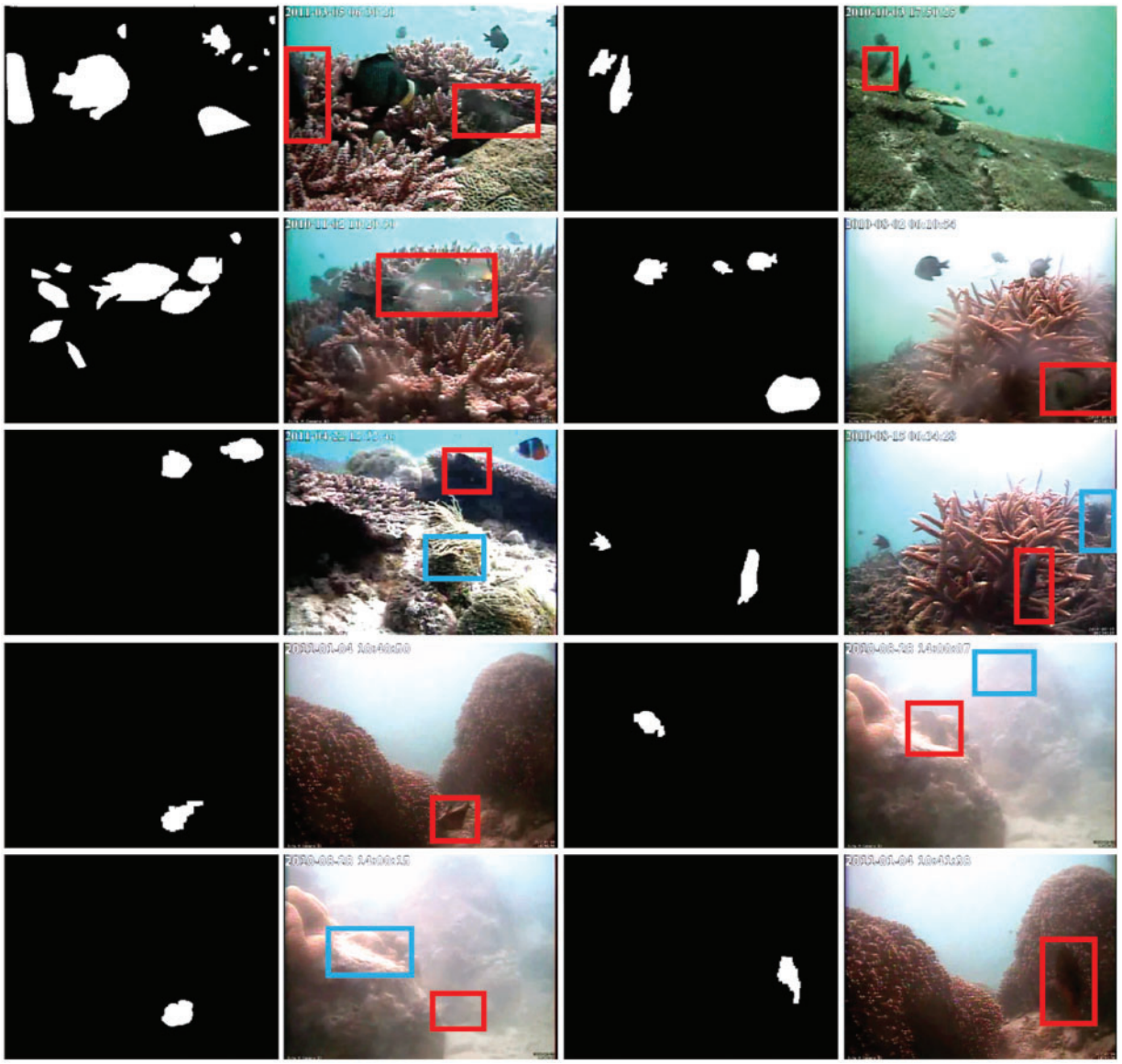

Figure 6. Examples of false detection of fish by all algorithms including our proposed hybrid system. Here, bounding boxes signify either miss detections of fish or false alarms. The black and white images are corresponding ground truths.

explanation for relatively worse performance of these approaches, as compared with the proposed system shown in Table 2, is an important observation that can be made by watching the videos in the datasets. The fish in each frame may not necessary show motion and sometimes remain dormant for multiple frames, even though for most of the time they are swimming, making the

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

scenes dynamic. The GMM sometimes confuses the fish with the stationary profile as background, especially when the appearance of fish matches the background. Therefore, lack of motion information in video frames results in failure to detect fish by the GMM and optical flow. The R-CNN on the other hand is a tailored neural network used for object localization in the images (Ren *et al.*, 2017) and learns to capture fish-dependent information from stationary images.

Underwater fish detection in unconstrained environments is a challenging task as the main aim lies in segregating fish and ignoring non-fish entities in the entire video frame. Conventional machine learning and image processing algorithms are generally designed to detect the objects of interest in the datasets where they exhibit their distinct presence in the imagery, and hence are easier to segment out (Russakovsky *et al.*, 2015). In contrast, a high degree of confusion in separating fish with vibrant, diverse and variable non-fish objects in underwater videos results in a performance compromise for a standalone R-CNN with accuracy of 64.99 and 77.30% on the FCS and LCF-15 datasets, respectively. As mentioned earlier, many videos, especially in the FCS dataset, lacks textural and shape information of fish, a necessary ingredient to yield better performance by systems like standalone R-CNN. This problem is effectively solved by our proposed hybrid system using and learning the information from motion-sensitive and textural features. Figure 6 shows some results from the FCS and LCF-15 datasets where all algorithms including our proposed system failed to detect fish. These are the extreme cases of blurriness, camouflage, water murkiness, and unrecognizable orientation, texture, and shape of fish which either results in generating false alarms or miss detections. In these situations, it is extremely difficult to capture both motion-based and shape/texture-based features.

In the future, we aim to employ a unified deep architecture capable of processing the video sequences in real-time through rigorous optimization of our algorithm and better mathematical modelling. Such a setup will be applicable for fish detection as well as their species classification at the same time and, therefore, will be more suitable for effective fish fauna sampling. Furthermore, the accuracy of the system can be improved by tracking the paths of moving fish and having prior information of their movement in several frames. This step can improve the accuracy of detection in the video frames where the proposed approach fails to recognize fish due to extreme blurriness and the camouflage of the background. We plan to incorporate this processing step using recurrent neural networks (Gordon *et al.*, 2018) with temporal processing capability in videos.

Conclusions

In this article, we have presented an automatic method that employed deep R-CNN networks to detect and localize fish instances in unconstrained underwater videos that exhibit various degrees of scene complexity. The major contribution of this work is that it utilizes a hybrid approach involving GMM and optical flow outputs to combine motion sensitive input features with raw video frames carrying textural and shape information. This mixed data is used as input to a deep R-CNN to fine-tune the categorization of fish in the presence of non-fish entities in the video frame. This assisted in achieving state-of-the-art results for the fish detection task as confirmed by the comparative study. The proposed hybrid system requires relatively more computational resources as compared with the conventional computer vision and machine learning techniques, but comes with the benefit of higher accuracy. However, with an advent of fast microprocessors and GPUs, complex mathematical operation involved in deep neural networks like CNN can be performed quickly, even making them suitable for tasks requiring near real-time processing. Therefore, combining the hybrid fish detection with other fish-related tasks like fish classification even using deep learning (Salman *et al.*, 2016) and tracking can be made possible in the pursuit of realizing fully automated systems for deployment in real world applications of fisheries. We believe that this research will help scientists related to fisheries in adopting automatic approaches for detection, classification and tracking of fish fauna in non-destructive sampling. Moreover, in the future, we aim to employ a unified deep architecture capable of processing the video sequences in real-time through rigorous optimization of our algorithm and better mathematical modelling. Such a setup will be applicable for fish detection as well as their species classification at the same time and therefore, will be more suitable for effective fish fauna sampling.

Acknowledgements

The authors acknowledge support from the Australian Research Council Grant LP110201008, and German Academic Exchange Service (DAAD) Project Grant 57243488 "FIBEVID". The authors also acknowledge Nvidia Corporation, USA for their donation of graphics processing units (GPUs) under their GPU Grant Programme. Nvidia GPUs were used to carry out experiments in the work carried out in this article.

References

Bengio, Y. 2009. Learning deep architectures for AI. Foundations and Trends in Machine Learning, 2: 1–127.

Beauchemin, S. S., and Barron, J. L. 1995. The computation of optical flow. ACM Computing Surveys, 27: 433–466.

Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. 2004. High accuracy optical flow estimation based on theory for warping. *In* Computer Vision-ECCV 2004. Lecture Notes in Computer Science, 3024. Ed. by T. Pajdla and J. Matas. Springer, Berlin, Heidelberg.

Deng, J., Dong, W., Socher, R., Li, L-J., Li, K., and Fei-Fei, L. 2009. ImageNet: a large-scale hierarchical image database. IEEE CVPR-2009, Miami, FL, USA, 248–255 pp.

Fisher, R., Chen-Burger, Y-H., Giordano, D., Hardman, L., and Lin, F-P. (Eds) 2016. Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data. Intelligent Systems Reference Library, 104, Springer International Publishing. DOI: 10.1007/978-3-319-30208-9.

Froese, R. 2006. Cube law, condition factor and weight length relationships: history, meta-analysis and recommendations. Journal of Applied Ichthyology, 22: 241–253.

Gordon, D., Farhadi, A., and Fox, D. 2018. Re3: real-time recurrent regression networks for visual tracking of generic objects. IEEE Robotics and Automation Letters, 3: 788–795.

Harvey, E. S., and Shortis, M. R. 1995. A system for stereo-video measurement of sub-tidal organisms. Marine Technology Society Journal, 29: 10–22.

He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. IEEE CVPR-2016, Las Vegas, NV, USA, 770–778 pp.

Hinton, G., and Salakhutdinov, R. 2006. Reducing the dimensionality of data with neural networks. Science, 313: 504–507.

Hinton, G. E., Osindero, S., and Teh, Y-W. 2006. A fast learning algorithm for deep belief nets. Neural Computation, 18: 1527–1554.

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019

Hsiao, Y., Chen, C., Lin, S., and Lin, F. 2014. Real-world underwater fish recognition and identification using sparse representation. Ecological Informatics, 23: 13–21.

Huang, G., Liu, Z., and Weinberger, K. Q. 2017. Densely connected convolutional networks. IEEE CVPR-2017, Honolulu, HI, USA, 2261–2269 pp.

Jennings, S., and Kaiser, M. J. 1998. The effects of fishing on marine ecosystems. Advances in Marine Biology, 34: 201–352.

Larochelle, H., Bengio, Y., Louradour, J., and Lamblin, P. 2009. Exploring strategies for training deep neural networks. Journal of Machine Learning Research, 10: 1–40.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. Neural Computing, 1: 541–551.

LeCun, Y., Huang, F., and Bottou, L. 2004. Learning methods for generic object recognition with invariance to pose and lighting. IEEE CVPR-2004, Washington, DC, USA, 97–104 pp.

LeCun, Y., Bengio, Y., and Hinton, G. 2015. Deep Learning. Nature, 521: 436–444.

Lin, T. Y., RoyChowdhury, A., and Maji, S. 2015. Bilinear CNN models for fine-grained visual recognition. IEEE ICCV-2015, Santiago, Chile, 1449–1457 pp.

McLaren, B. W., Langlois, T. J., Harvey, E. S., Shortland-Jones, H., and 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. Journal of Experimental Marine Biology and Ecology, 471: 153–163.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K. R. 1999. Fisher discriminant analysis with kernels. IEEE International Workshop on Neural Networks for Signal Processing, Madison, WI, USA, 41–48 pp.

Moniruzzaman, M., Islam, S. M. S., Bennamoun, M., and Lavery, P. 2017. Deep learning on underwater marine object detection: a survey. *In* Advanced Concepts for Intelligent Vision Systems. ACIVS 2017. Ed. by J. Blanc-Talon, R. Penne, W. Philips, D. Popescu, and P. Scheunders. Lecture Notes in Computer Science, vol 10617, Springer, Cham.

Palazzo, S., and Murabito, F. 2014. Fish species identification in real-life underwater images. *In* 3rd ACM International Workshop on Multimedia Analysis for Ecological Data, Orlando, Florida, pp. 13–18.

Ren, S., He, K., Girshick, R., and Sun, J. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39: 1137–1149.

Reynolds, D. A., and Rose, R. C. 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Transactions on Acoustics, Speech and Signal Processing, 3: 72–83.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z. *et al.* 2015. ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 115: 211–252.

Salman, A, Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., and Harvey, E. 2016. Fish species classification in unconstrained underwater environments based on deep learning. Limnology and Oceanography: Methods, 14: 570–585.

Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2017. Automatic fish species classification in underwater videos: exploring pre-trained deep neural network models to compensate for limited labelled data. ICES Journal of Marine Sciences, 75: 374–389.

Sheikh, Y., and Shah, M. 2005. Bayesian modelling of dynamic scenes for object detection. IEEE Transaction on Pattern Analysis and Machine Intelligence, 27: 1778–1792.

Shortis, M., Harvey, E. S., and Abdo, D. 2009. A review of underwater stereo-image measurement for marine biology. In Oceanography and Marine Biology: An Annual Review. Ed. by R. N.Gibson, R.J.A. Atkinson, and J.D.M. Gordon. CRC Press, USA.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556.

Spampinato, C., Chen-Burger, Y., Nadarajan, G., and Fisher, R. B. 2008. Detecting, tracking and counting fish in low quality unconstrained underwater videos. International Conference on Computer Vision Theory and Applications, Funchal, Madeira, Portugal, 2: 514–519.

Spampinato, C., Palazzo, S., and Kavasidis, I. 2014. A texton-based kernel density estimation approach for background modeling under extreme conditions. International Journal of Computer Vision and Image Understanding, 122: 74–83.

Storbeck, F., and Daan, B. 2001. Fish species recognition using computer vision and a neural network. Fisheries Research, 51: 11–15.

Strachan, N. J. C., and Kell, L. 1995. A potential method for the differentiation between haddock fish stocks by computer vision using canonical discriminant analysis. ICES Journal of Marine Science, 52: 145–149.

Stauffer, C., and Grimson, W. E. L. 1999. Adaptive background mixture models for real-time tracking. IEEE CVPR-1999, Fort Collins, CO, USA, 2: 246–252.

Sung, M., Yu, S., and Girdhar, Y. (2017). Vision based real-time fish detection using convolution neural network. IEEE OCEAN-2017, Aberdeen, UK, 1–6 pp.

Szegedy, C., Ioffe, S., and Vanhoucke, V. 2016. Inception-v4, inception-resnet and the impact of residual connections on learning. AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 4278–4284 pp.

Tanzer, J., Phua, C., Lawrence, A., Gonzales, A., Roxburgh, T. and Gamblin P. (Eds) 2015. Living Blue Planet Report. Species, Habitats and Human Well-Being. WWF, Gland.

Turk, M., and Pentland, A. 1991. Eigenfaces for recognition. Journal of Cognitive Neuroscience, 3: 71–86.

Warren, D. H., and Strelow, E. R. 1985. Electronic Spatial Sensing for the Blind: Controbutions from Perception. Springer. ISBN 90-247-2689-1.

Yao, J., and Odobez, J. M. 2007. Multi-layer background subtraction based on color and texture. IEEE CVPR-2007, Minneapolis, MN, USA, 1–8 pp.

Yong, X. 2013. Improved Gaussian mixture model in video motion detection. Journal of Multimedia, 8: 527–533.

Zivkovic, Z., and Heijden, F. 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters, 27: 773–780.

Handling editor: Cigdem Beyan

Downloaded from https://academic.oup.com/icesjms/advance-article-abstract/doi/10.1093/icesjms/fsz025/5366225 by guest on 28 February 2019