





Article

Automatic Sorting of Dwarf Minke Whale Underwater Images [†]

Dmitry A. Konovalov ^{1,2,*} , Natalie Swinhoe ^{1,†} , Dina B. Efremova ^{3,‡}, R. Alastair Birtles ¹ , Martha Kusetic ¹, Suzanne Hillcoat ¹, Matthew I. Curnock ⁴, Genevieve Williams ¹ and Marcus Sheaves ^{1,2} 

¹ College of Science and Engineering, James Cook University, Townsville, QLD 4181, Australia

² Marine Data Technology Hub, James Cook University, Townsville, QLD 4811, Australia

³ Funbox Inc., 119017 Moscow, Russia; dina.efremova85@gmail.com

⁴ CSIRO Land and Water, James Cook University, Townsville, QLD 4811, Australia;

* Correspondence: dmitry.konovalov@jcu.edu.au

† This paper is an extended version of our paper published in This paper is an extended version of our paper published in 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019.

‡ These authors contributed equally to this work.

Received: 25 February 2020; Accepted: 5 April 2020; Published: 9 April 2020



Abstract: A predictable aggregation of dwarf minke whales (*Balaenoptera acutorostrata* subspecies) occurs annually in the Australian waters of the northern Great Barrier Reef in June–July, which has been the subject of a long-term photo-identification study. Researchers from the Minke Whale Project (MWP) at James Cook University collect large volumes of underwater digital imagery each season (e.g., 1.8TB in 2018), much of which is contributed by citizen scientists. Manual processing and analysis of this quantity of data had become infeasible, and Convolutional Neural Networks (CNNs) offered a potential solution. Our study sought to design and train a CNN that could detect whales from video footage in complex near-surface underwater surroundings and differentiate the whales from people, boats and recreational gear. We modified known classification CNNs to localise whales in video frames and digital still images. The required high classification accuracy was achieved by discovering an effective negative-labelling training technique. This resulted in a less than 1% false-positive classification rate and below 0.1% false-negative rate. The final operation-version CNN-pipeline processed all videos (with the interval of 10 frames) in approximately four days (running on two GPUs) delivering 1.95 million sorted images.

Keywords: computer vision; dwarf minke whales; convolutional neural networks; underwater object classification; image classification; deep learning

1. Introduction

A predictable aggregation of dwarf minke whales [1] (*Balaenoptera acutorostrata* subsp.) occurs in the Australian waters of the northern Great Barrier Reef (GBR) in June and July each year. The dwarf minke whale is the second smallest baleen whale, born at approximately 2 m in length and growing to a maximum measured length of 7.8 m [2]. While in the northern GBR, these whales regularly interact with people and boats [3]. To date, there is no population estimate for this undescribed subspecies, and the GBR aggregation represents a unique opportunity to study and improve our understanding of this poorly-known whale. Since the mid-1990s, a tourism industry has established around this aggregation, providing swim-interactions for dive tourists [4,5], as well as “platforms-of-opportunity” for researchers to collect various data, including underwater images of individual whales [3–8].

The whales' natural colour patterns are complex (especially around their head and shoulder areas), individually variable, and likely remain stable through a whale's life, enabling photo-identification (photo-ID) of individual animals [7–10], see a typical example in Figure 1.



Figure 1. An example of individual dwarf minke whale distinct colour patterns. Image was enhanced by the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm [11].

The identification of individual whales from digital imagery underpins research on population characteristics, biology and behaviour [6]. Once whales are identified, the location, time, and additional data from that sighting (such as behaviour and size measurements) are used to understand the whale's movement in time and space. Unique whale identifications provide insight into whale re-sightings and overall whale numbers, valuable data in understanding population dynamics and the sustainability of the associated tourism industry. This information is critical for the management and monitoring of the species and tourism activities [4,5,8]. Each year an ever-growing volume of images and videos of dwarf minke whales is collected by the Minke Whale Project (MWP) research team and by tourists on permitted swim-with-whale vessels aboard GBR diving expeditions operating out of Cairns and Port Douglas, Queensland, Australia. The volume of imagery collected is growing rapidly (order of 50,000 images in each of the last three years) with advancements in underwater digital camera technology and accessibility. This volume of data combined with the labour-intensive nature of dwarf minke whale photo-ID analyses has created a bottleneck in processing data, whereby it is not time-effective for researchers to process and analyse the overwhelming seasonal stream of dwarf minke whale imagery. As a result, there is a large historical database of partially-analysed imagery. In 2019, ten volunteer researchers contributed over 2000 h toward photo-ID, processing approximately 31,000 photos and videos. However, in 2019 alone, the research team collected over 62,000 images.

Video monitoring of aquatic and terrestrial animals is an important tool for ecologists and biologists [12,13]. As digital storage capacity continues to decrease in cost, video monitoring methods are shifting towards longer intervals of uninterrupted recordings. Similarly, in our study underwater digital cameras were often left recording continuously to ensure that all minke whale encounters were captured with maximum information to facilitate the photo-ID of the whales. Therefore, the limited researcher and volunteer human resources are inefficiently used by searching through large amounts of imagery with no whales. Often 90–99% of video content is not research relevant but must be watched for quality assurance and research rigor. At present, there is no efficient way of discarding the negative content. Therefore, the full videos are retained which dramatically increases researchers' effort, disk storage requirements and file transfer times.

The focus of this study was to develop an image classifier, which could automatically detect a dwarf minke whale in an image or video frame with a high degree of accuracy.

1.1. Related Work

To introduce the terminology, given an image, a binary image classifier assigns the image to a *positive* or *negative* class label [14]. In our case, an image (or an individual video frame) of the positive class contained one or more dwarf minke whales. An image was labelled as negative if it did not contain a whale. Two definitions of *whale-detection* were considered. First, a biologist could detect and/or infer a whale in an image. Second, a whale was visible with enough details making the image suitable for photo-identification of the whale. For example, Figure 2a technically contains a whale and is labelled as positive under the first definition of whale-detection but the image is not useful for the photo-ID processing, hence it was labelled as negative under the second definition.

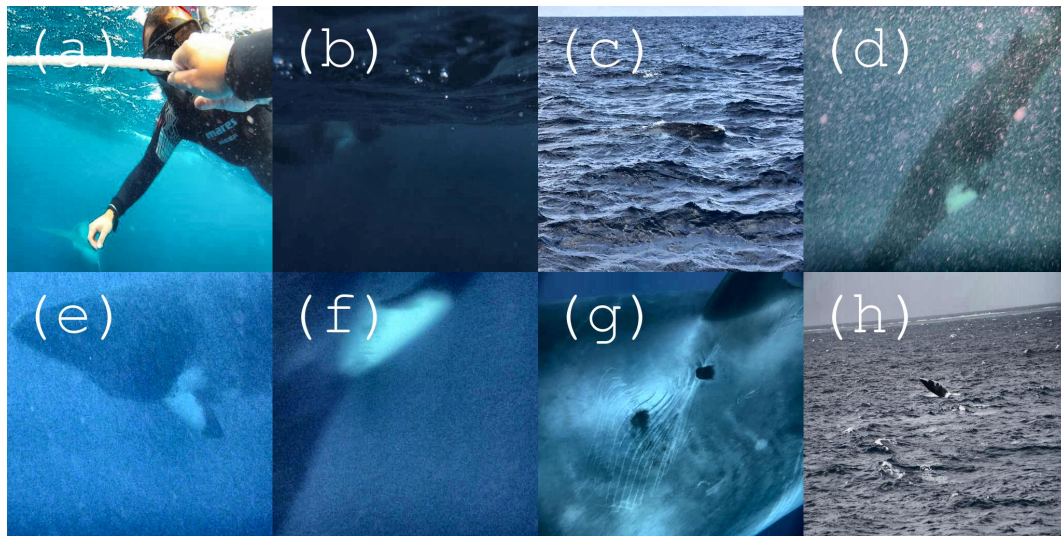


Figure 2. Sub-figures: (a–h) are examples of false-negative classification errors, which were the positive test images incorrectly predicted as negatives (missing whale) by Minke Whale Detector (MWD)-v1 after the *first* train-predict cycle using the first strictly technical definition of whale-detection.

Currently, many successful methods for image classification and object detection in images are based on Convolutional Neural Networks (CNN) [15]. Due to their extraordinary success, a variety of image classification CNN architectures were developed [16–19] and are routinely used in practice. For example, the Xception [18] CNN was used to detect invasive cane toads in surveillance videos [20]; and ResNet-50 [17] and Inception-v3 [16] were used to classify weeds in images [21]. However, the object detection and image classification tasks are not identical. A CNN-based image classification method outputs a probability of an image containing a required class of objects without specifying the locations of the objects. Explanation of the decision process of the modern highly complex CNNs is presently a separate research-grade task [22–24].

In contrast to the classification CNNs (C-CNNs), a CNN-based object detection method outputs locations of the requested objects, where the locations are typically reported as bounding boxes, for example, Reference [25]. While the resulting localisation bounding boxes do not explain *how* the CNN arrived at their locations, they nevertheless provide instant visual feedback to a user, who could easily verify if the CNN is working correctly or not. For example, Parker et al. [26] applied R-CNN [27], Fast R-CNN [28], and Faster R-CNN [25] methods to detect seals and dolphins in underwater videos. The Faster R-CNN with 2000 proposals (i.e., potential bounding-boxes) was the most accurate in correctly detecting and localising the considered animals. This result was consistent with the Faster R-CNN being the second iterative improvement of the original R-CNN method [27].

The object-detection CNNs (OD-CNNs) are clearly a better choice than classification CNNs (C-CNNs) in terms of *explainability* [24]. However, the OD-CNNs have two major disadvantages compared to C-CNNs. First, OD-CNNs are typically much slower than C-CNNs, for example, Faster-RCNN required

1.5 s per image [26]. This is a speed/accuracy trade-off [29], where accurate localisation of objects requires additional processing time. The second disadvantage is that OD-CNNs require thousands of human-annotated bounding boxes (for each category, i.e., class of objects) as training data, which are very time-consuming to prepare. To illustrate the availability gaps in image-level annotations (required for C-CNNs) and the bounding-boxes (for OD-CNNs), note that 36 million image-level labels are currently available in Open-Images-V5 [30,31] for almost 20,000 categories while 16 million boxes are available for only 600 categories. While the OD-CNN processing speed was improving [32], the second OD-CNN's disadvantage remained unavoidable and hence was the deciding factor against using OD-CNNs in this project, where the OD-CNN training bounding boxes for the underwater images of dwarf minke whales were deemed too expensive and/or time-consuming to prepare.

Somewhat like object-detection CNNs, semantic segmentation CNNs [33] could also be used to localise dwarf minke whales [10]. For example, in Reference [10], 100 segmentation masks were prepared manually. However, such a small number of masks could only teach FCN-8s CNN [33] to localise whales within simple monotonic surroundings such as the example in Figure 1. Once applied to the unfiltered raw videos (see frame examples in Figure 3), the FCN-8s CNN (from Reference [10]) yielded unacceptably large number of false-positives. Therefore, a much larger number of segmentation masks would be required, which are more difficult to prepare than the bounding-boxes. For example, only 2.8 million segmentation masks are available for an even smaller number of 350 categories in Open-Images-V5 [30,31] compared to 16 million boxes for 600 categories.

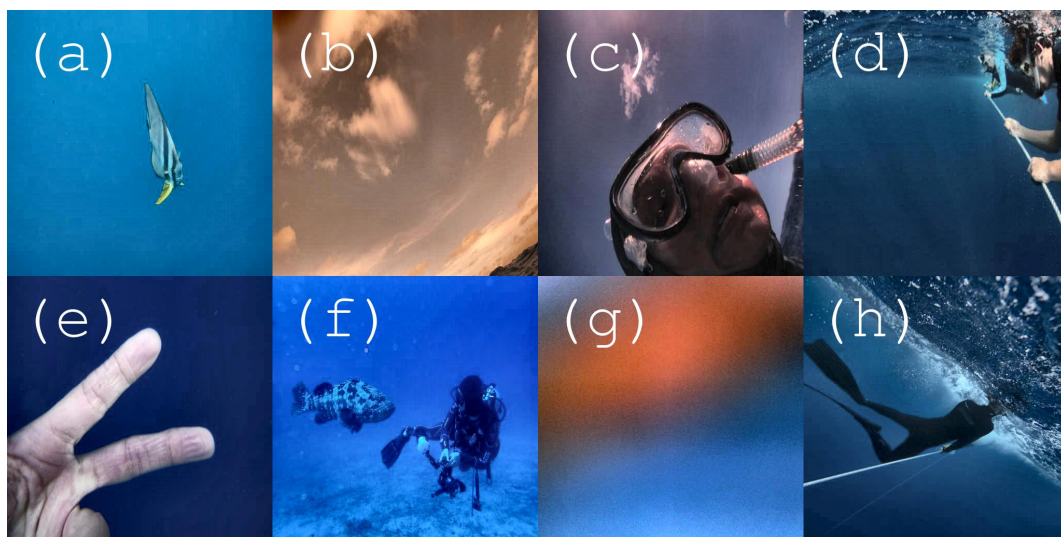


Figure 3. Sub-figures: (a–h) are examples of false-positive classification errors, which were the negative test images incorrectly predicted as positives (contain a whale) by MWD-v1 after the *first* train-predict cycle.

1.2. Method Overview

A typical architecture of modern classification CNNs contains a *feature-encoder* section [16–18], which compresses image's spatial dimensions by a factor of 32 while extracting a large number of image features, for example, 2048. An input $512 \times 512 \times 3$ RGB image is converted to $16 \times 16 \times 2048$, where the three RGB colour channels are converted to the 2048 features. If the reduced-by-32 spatial dimensions are retained, a classification 1×1 convolutional layer could then be used to convert the 2048 features to a localisation *probability-map* (also known as *heatmaps*), where a higher value would indicate higher predicted probability of the object located at that output spatial pixel. This approach was successfully tested for detecting cane toads in night-time surveillance videos [20] and multiple fish species in underwater videos in very complex tropical habitats [34].

In this study, the exact localisation bounding-boxes of animals were not particularly important, as long as the biologists could easily verify that a CNN detected (i.e., true-positives) or did not detect

(i.e., true-negatives) the required animals correctly. The classification CNN 16×16 probability-maps were re-scaled back to the original input shape (512×512) and overlapped with the corresponding images to produce the localisation versions (nicknamed the *why*-images), see an example in Figure 4a. This technique fulfilled an auxiliary social goal of this study to accelerate the adoption and acceptance of the Deep Learning CCNs and to overcome the black-box stigma of the CCNs. In fact, the *why*-images were used nearly exclusively for testing and verification in the later stages of this study.

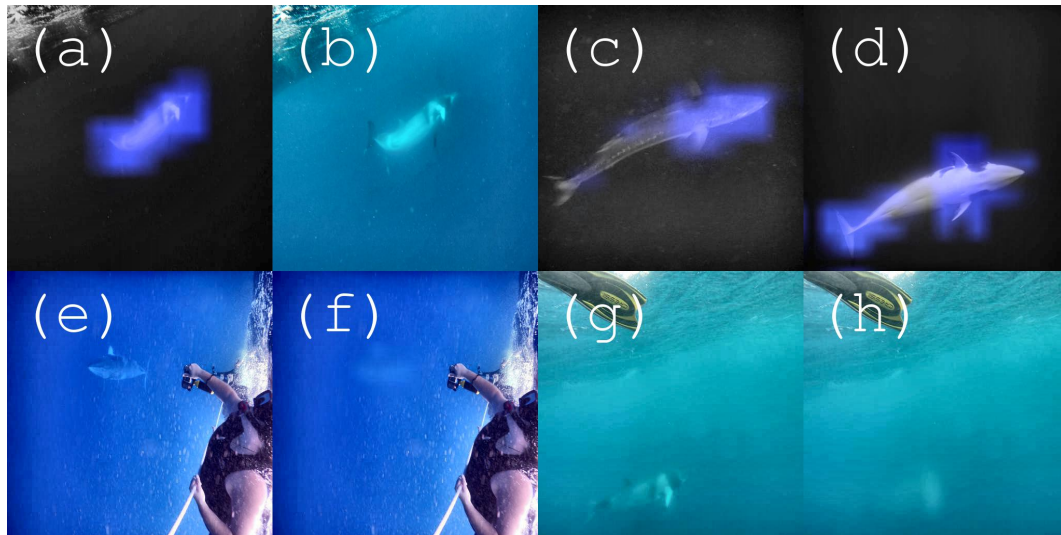


Figure 4. Sub-figures: (a,c,d) are examples of localisation by MWD; (b) is the corresponding original for (a); (f,h) are examples of the negative-labelling technique, where the corresponding original images are (e) and (g), respectively.

Since our approach is based on classification CCNs, it requires many thousands of positively and negatively labelled training images to become sufficiently accurate. Fortunately, in some circumstances and especially in surveillance videos [20,34], positive and negative video clips could be easily selected yielding many thousands of training frames with minimal human involvement.

In this study, we extended this approach to videos where cameras were constantly moving. Therefore, the negative video frames without a whale were often visually very different from the positive frames containing the whales. As such, the difference in positive and negative images could be very large and still assist effective training, as was the case of moving debris and underwater vegetation in the underwater fish detection [34]. In the surveillance videos [20,34], the “intuition” was that moving/changing background items repeat themselves in some fashion in both negative and positive images. Hence, a CNN could learn to identify the objects only present in the positive frames, for example, fish in Reference [34]. However, the same intuition worked equally well in this study, where a vast number of items, textures and/or patterns often appeared only in positive examples and therefore were incorrectly learned as a “whale” by the CNN. This problem was solved here by inventing a negative labelling technique, where in relatively small number of cases (less than 100) the whales were manually edited out by blurring or excessive distortion.

The total number of video frames was estimated to be in the order of 19 million images. Therefore, an iterative CNN-refinement approach was adopted, where the training pool of negative and positive images were extended by manually correcting false-positive and false-negative errors. However, as the CNN become more accurate with each iteration, the main theoretical challenge of this study was revealed—how to achieve low rates of false-negatives and false-positives without retesting all available imagery (1.8 TB), where each testing round took many hours of manual inspection by a researcher while viewing only a tiny fraction of the CNN outputs.

By adding greater numbers of negative and positive images, the CNN reduced false-positives and increased false-negatives or vice versa. By trying to reduce both false-positives and false-negatives

at the same time, we discovered the main novel contribution of this paper, the negative labelling technique. Whenever the CNN made a false-negative error (i.e., did not detect an existing whale), that image was added to the pool of positive training images. Additionally, the same image was manually edited by blurring or distorting the whale beyond human recognition and added to the pool of negative images.

In summary, this study made the following contributions:

- Negative-labelling technique was proposed and verified to be effective in assisting *approximate* object localisation via classification CNNs.
- Simple and very effective architectural modifications (Table 1) to modern *off-the-shelf* classification CNNs were verified to be valuable to the end-users by simultaneously yielding approximate object localization and image classification. The combined localization heatmaps and the original images could assist in *explaining* CNN's results to the users (marine biologists in this study) and hence to accelerate overall acceptance of the deep CNN technologies.
- The generality of the proposed localising classification architecture (Table 1) was verified by using ResNet-50 and VGG-13bn CNNs with minimal modifications (PyTorch versions) confirming the initial results of Reference [34] obtained via Keras/Tensorflow-based Xception CNN [18].
- We developed a very accurate (below 0.1% false-negatives and below 1% false-positives) pipeline for processing large volumes (1.8TB) of digital imagery of dwarf minke whales.
- The following CNN training techniques were demonstrated to work in a complementary manner for this study's domain of near-surface underwater imagery: linear learning rate annealing, uniform class undersampling, layer-specific learning rate reduction, trainable conversion of greyscale images for ImageNet-pretrained CNNs, weak cross-domain negative supervision (VOC [35] was used).

Table 1. MWD Convolutional Neural Network (CNN) architecture yielding simultaneous approximate whale localization and image classification.

Input Dimensions		Layer Description	Output Dimensions	
Spatial	Channels		Spatial	Channels
512 × 512	1	Trainable conversion to 3-channels, $conv(1 \times 1, 1 \rightarrow 3)$ *	512 × 512	3
512 × 512	3	An ImageNet-trained CNN without its classification top (ResNet-50 was used)	16 × 16	2048 **
16 × 16	2048	Trainable object localization heatmap, $conv(1 \times 1, 2048 \rightarrow 1) + sigmoid$ ***	16 × 16	1
16 × 16	1	Image classification output via <i>maxpool</i>	1 × 1	1

* $conv(k \times k, n \rightarrow m)$ is a $(k \times k)$ -kernel convolution layer converting n channels to m channels. ** The number of CNN output channels (or *features*) was 2048 or 512 when using ResNet-50 or VGG-13bn, respectively.

*** *sigmoid* is the Sigmoid activation function.

2. Results

2.1. First Train-Predict Cycle

The first train-predict cycle adopted the VGG-13bn based Minke Whale Detector (MWD) and the first definition of whale-detection, which did not consider if detected whales were suitable for photo-ID or not. The collection of 1320 individually labelled minke images from Reference [10] was used as the training positives. The collection was denoted as the MWPID-2014 dataset after **Minke Whale Photo-ID** since it contained 76 manually identified individual animals from the 2014 observation season. The VOC-2012 [35] collection of 17,000 images was used as negatives. The VOC-2012 images contained 20 different object classes within four categories: Vehicles (e.g., bus), Household (e.g., chair), Animals

(e.g., cat), and People. For training, 90% of MWPID-2014 and randomly selected 1000 VOC-2012 were used, where the 1000 VOC-2012 images were randomly re-drawn for each epoch of training.

To monitor (formally *validate*, or *cross-validate*) the model prediction (*generalization*) performance, the remaining 10% of MWPID-2014 images were used as the validation subset together with 100 randomly selected (again, different for each validation epoch) VOC-2012 images. By observing the loss and accuracy values on the validation subset, it was found that only 10–20 training epochs were needed to achieve 98–100% validation accuracy when training the first version of MWD (MWD-v1). After training, the MWD-v1 was applied to a subset of images from the 2018 observation season collection, the MWS-2018i dataset of 11,704 images. MWS-2018i was sorted into predicted negatives (no detected whale) and positives (at least one whale). All labelled by MWD-v1 images (resized to 512×512 for file management convenience) were visually inspected and all detected prediction errors were recorded, see the first cycle row in Table 2.

Table 2. Results of the train-predict cycles.

Cycle	Training Images (Count)	Test Images (Count)	FN * (FN/Count, %)	FP * (FP/Count, %)
1	MWPID-2014 [10] (1300) + VOC-2012 [35] (17,000)	MWS-2018i (11,704)	908 (7.8%)	395 (3.4%)
2	+ MWS-2018i (11,704)	MWS-2018-s100 (8373)	1973 (23.6%)	61 (0.73%)
3	+ MWS-2018-s100 **	MWS-2018-s10a ($\approx 40,000$)	377 (0.9%)	not recorded
Final	+ MWS-2018-s10a ** (16,471) ***	MWS-2018-s10b ($\approx 243,000$)	< 100 (<0.04%)	< 1000 (<0.4%)

* FN and FP denote false-negatives and false-positives, respectively. ** Images not suitable for photo-ID were removed from positives, and new negatives were added via the negative-labelling procedure. *** Manually curated final dataset of 16,471 images (excluding the 17,000 VOC-2012 [35] negatives).

2.2. Second Train-Predict Cycle

The MWS-2018i collection of 11,704 images was sorted by the first version of MWD (MWD-v1) and manually verified and, if needed, corrected. Examples of the first-cycle false-positives are displayed in Figure 3, which illustrate the diversity of the imagery. Conceptually and technically, the false-positives were easy to deal with by adding them to the pool of negative training images for the next train-predict cycle. However, the false-negatives presented a pivotal point of the study, see Figure 2. For example, Figure 2a technically has a whale in it (as per the first definition of whale-detection), which is the unidentifiable whale next to the person's hand. Similarly, the above-water images in Figure 2c,h contain whales but the images could not contribute to the photo-identification.

Exploring whether the first (strictly technical) definition of whale-detection is viable, the manually verified MWS-2018i images were added to the pool of training images and the second version of MWD (MWD-v2) was trained. Note false-positive examples in Figure 5 and false-negative examples in Figure 6. When MWD-v2 (still based on VGG-13bn) was applied to every 100th frame (step of 100 frames) from a subset of videos (denoted MWS-2018-s100 and contained 8,373 images) the rate of false-negatives increased to 23.6% (see the 2nd cycle row in Table 2). Investigation of such dramatic deterioration of the false-negative rate revealed the following interesting finds, which are detailed in the next three subsections.

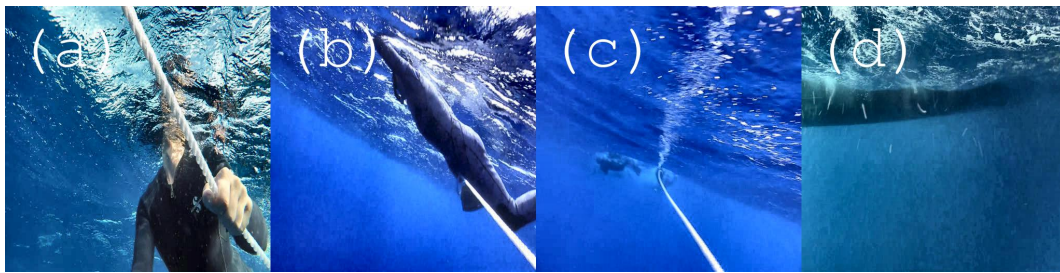


Figure 5. Sub-figures: (a–d) are examples of false-positive classification errors by MWD-v2 after the *second* train-predict project cycle. Note, (d) is possibly a labelling error.



Figure 6. Sub-figures: (a–d) are examples of false-negative classification errors by MWD-v2 after the *second* train-predict project cycle using the first definition of whale-detection.

2.3. Switching from VGG-13bn to ResNet-50

Similar results (first and second cycle rows in Table 2) were observed by using the ResNet-50 based MWD (Table 1) and 512×512 training image shapes, instead of the VGG-13bn base and 256×256 images. In general, the ResNet-50 version was slightly more accurate for images including larger, blurry whales, for example, Figure 6d, while the VGG-13bn version was marginally better for smaller whale images, for example, Figure 6c. Since large whale images were more likely to be useful for photo-ID, only the ResNet-50 versions were considered for the remaining of this study. Furthermore, the ResNet-50 based MWD was faster to train and it had faster processing speed with 512×512 images than the VGG-13bn based MWD on 256×256 images.

2.4. Negative Labelling

Investigating the large false-negative rate (second row of Table 2) revealed that training with different compositions of the negative and positive pools of images made the numbers of false-negatives and false-positives highly unstable and oscillating. The main reason for the instability was traced to the first strictly technical definition of whale-detection. For example, Figure 6a contained the tail sections next to the person's knee. When such images were added to the pool of positive images, any other objects in the images were effectively labelled as a "whale". To overcome this problematic effect of the per-image labelling, it was discovered that both false-negative and false-positive rates could be consistently improved by using *negative-labelling*. For any false-negative erroneous classification, the negative-labelling technique consisted of creating a corresponding negative image by editing the whale out of the image, see sample pairs in Figure 4e,f, as well as (g) and (h). Then the original positive image was added to the pool of positive images and the corresponding manually edited negative image was added to the negative pool.

2.5. Uniform Class Sampling

Another source of training instability was due to potentially highly imbalanced numbers of positive and negative images. Often, very large numbers of only positive or only negative video frames were added to the training pool of images, which did not necessarily improve the validation accuracy. Such training experiments were counter-intuitive, when one would expect to obtain more accurate classifier

from additional training images even if they were all new negatives or all new positives. The limitation of carefully balancing the negative and positive counts was not practical as well as theoretically unsatisfactory. We solved this challenge by applying the uniform class *undersampling* [36] method. Hence, one training *epoch* was fixed at 1000 samples, which on average contained approximately 500 negative and 500 positive images. One validation epoch was defined as per standard machine learning convention to include each available validation image exactly once, see Figure 7.

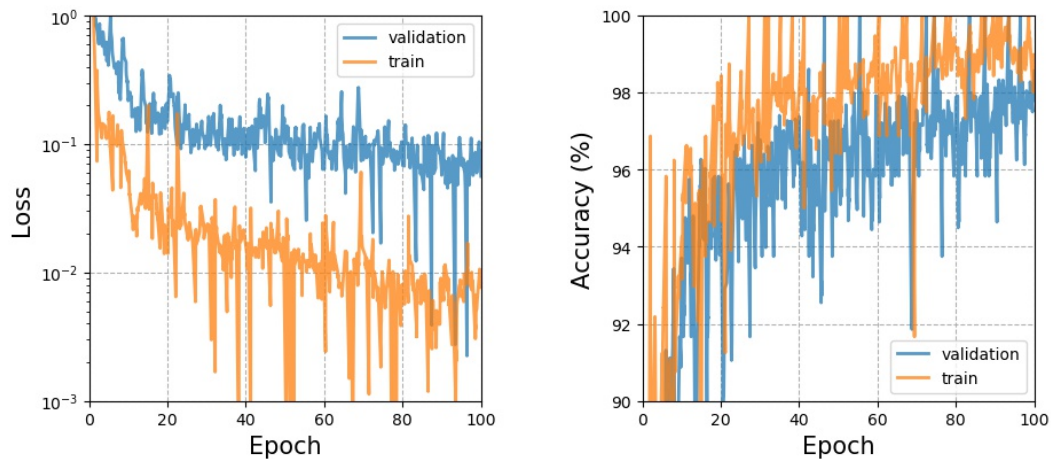


Figure 7. MWD-v4 training history of binary cross-entropy loss and accuracy calculated on the training and validation subsets.

2.6. Switching to the Photo-ID Definition of Whale-Detection

The negative-labelling technique stabilized the process of consistent improvement of the MWD. However, the number of required manual edits (to create the matching negatives) became impractically large within the scope of the first definition of whale-detection. That is, there were too many cases similar in fashion to Figure 6a,c. Furthermore, if an image (or video frame) was retained but it was not suitable for the photo-ID, the image would have to be manually discarded by researchers at a later stage. Therefore, it was concluded that the first definition of whale-detection was not suitable for this study and the second definition of the positives was adopted for the final stages of this work, which was “*image containing a whale suitable for the photo-ID labelling*”. Thus, starting from the third cycle, all images not suitable for photo-ID were removed from the training pool of positives. For example, Figures 2a,c,h and 6a,c were removed.

2.7. Third Train-Predict Cycle with ResNet-50-Based MWD

For the third train-predict cycle, MWD-v3 (with the ResNet-50 backbone CNN) was tested on an unseen (by MWD-v3) collection of video frames, where every 10th frame (step of 10) was taken from a previously unused subset of videos and denoted MWS-2018-s10a containing approximately 40,000 images. At this point, the most critical performance question was assessed: “*How many whales were completely missed due to the false-negatives?*”, where the missed whales occurred only in the false-negatives but not in true-positives. There were 377 false-negative images spread across 83 videos. Each of the false-negatives were assessed to see if the whale was identifiable (i.e., recognizable colour patterns, distinct scar(s), unique whales). In the 377 false negatives (the third cycle row in Table 2), there were 12 identifiable whale images that had recognizable coloration patterns or a distinct scar(s). The 12 whales were cross-checked against the true-positives from the same videos and verified that all 12 whales were detected correctly in the corresponding true-positives. Hence, all identifiable whales were detected.

2.8. Final Train-Predict Cycle

Due to project timing and team members' availability, the final version of the MWD could only be comprehensively acceptance tested once, which required approximately one full-time-person-week of manual checking. If failed to achieve the required accuracy, especially for the 0.1% false-negative rate target, the proposed automatic sorting of the 2018 imagery would be possibly postponed for many months or until after the 2020 field season. Keeping in mind that a number of non-standard techniques (e.g., weak cross-domain supervision with VOC [34], uniform class sampling and negative labelling) were used to progress in this study, the final training dataset was carefully curated in an attempt to mitigate identified domain challenges. Specifically, many thousands of repetitive video frames were removed arriving at the final set of 16,471 highly diverse images containing $P = 13,173$ positives and $N = 3298$ negatives. For the final MWD-v4 training (see Figure 7), the final dataset was randomly split 90% for training and 10% for validation preserving the percentages of the positive and negative classes, that is, stratified on the class labels. Then the trained MWD-v4 (Table 1) was tested on MWS-2018-s10b containing approximately 243,000 images, where every 10th frame (step of 10) was taken from a previously unused subset of videos. The planned target performance was well exceeded by the achieved 0.04% false-negative and 0.4% false-positive rates, see the final cycle row in Table 2.

2.9. Localisation Heatmaps

The localisation heatmaps (produced by MWD-v4) were exclusively used to check the sorted (whale/no-whale) predictions, see examples in Figure 4a,d. Note the interesting false-positive in Figure 4c, where MWD detected a shark. This could be viewed as a serendipitous property of the MWD (and CNNs in general), which would detect any visual features similar to those in dwarf minke whales (training positives). Furthermore, the advantage of using the localisation heatmap is particular pronounced in that "miss-classification", Figure 4c, which highlighted the head and fins of the shark but not its body. Similar, the middle section of the whale in Figure 4d is not highlighted indicating that MWD did not see sufficient number of such cases during training. This visual conformation (of *why* any given classification occurred) could be easily used to design or adjust training set of images. For example, the ignored (not highlighted) ventral area of the whale in Figure 4d allows for sex identification of individuals but it is currently not detected on its own.

2.10. Negative-Labeling Viability

In total, less than 100 manual negative-labelling edits were eventually required for the final training of the MWD (ResNet-50 based), see examples in Figure 4f,h, which confirmed the viability of the image-level labelling approach adopted in this study. Whale blurring (editing out) could be performed in many commonly used image manipulation programs and importantly it could be done by the researchers using MWD if required in the future. For example, we used the freely available GNU Image Manipulation Program (GIMP), where a range of blurring filters could be applied to a selected image area. In contrast, the bounding-boxes labelling approach requires specialised software and user training. Furthermore, the bounding boxes could introduce additional training uncertainty and instability when a non-whale background is selected, for example, for the bounding-box labelling of Figure 2d.

2.11. Final Sorting of the 2018 Season Imagery

While examining the sorted predictions in the third and fourth train-predict cycles, it was noted that the step of 10 frames (every 10th frame and 0.1–0.4 seconds interval) was sufficiently frequent (within each video) to yield a comprehensive set of images to identify each detected whale. Running on two NVIDIA 1080Ti GPUs, it took approximately four days to process all available 1.8TB imagery from the 2018 field season. The resulting image-sorted folders contained approximately 1.95 million images from which 805,000 (41%) were positives and 1,149,500 (59%) negatives. We are planning to open-source the pre-trained dwarf minke whale detector and the filtering pipeline to the public in due course.

3. Discussion

Underwater videos are a valuable tool for capturing non-symmetrical identification colorations and markings of dwarf minke whales in situ, and a full analysis of video imagery is a core goal of the MWP research team. Utilizing a low-error MWD to sort through high volumes of videos greatly reduces the time and effort needed by researchers to manually sort through non-whale imagery. This frees researchers to focus on more substantial biological analyses. Hundreds of hours of footage from multiple field seasons can be sorted by the MWD in a matter of days providing positive images of dwarf minke images for identification more efficiently. This will allow more time to be spent on individual whale identification and matching re-sightings via a catalogue. CNN classification has also shown the capacity for further biological analysis, with the ventral side identification of dwarf minke whales, as in Figure 4d. Ventral (currently indirect) recognition allows for sex identification of individuals and indicates the capability to better understand dwarf minke whale population statistics in the Great Barrier Reef.

The primary beneficiaries of this study are, ultimately, the whales themselves, with their effective management and conservation dependent on an improved understanding of population dynamics and exposure to a range of anthropogenic pressures. In the short and medium term, the immediate beneficiaries are the MWP research team, and managers of the Marine Park who seek to improve monitoring of migratory species in the Great Barrier Reef. The MWP research team benefits in reclaimed research time, data outputs and future efficiency with this CNN method for photo-ID processing time. Reef managers (and the tourism operators and tourists who interact with the whales) benefits from the improved efficiency of research and monitoring in which they participate, via more timely results and an improved understanding of the resource [37].

4. Materials and Methods

4.1. Minke Whale Photographic Database

Our 2018 Minke Whale Field Season generated a collection of digital still images and videos with total size of 1.8 TB. It contained approximately 50,000 still images and 5000 videos. In this study, the per-image classification was adopted and every video was converted to images (one video frame converted to one image). This resulted in more than 19 million video frames.

4.2. Requirements and Constraints

The following requirements for the **Minke Whale Detector (MWD)** were identified in consultation with the project stakeholders:

1. *Image-level*: MWD should work on a per-image level since many thousands of still digital images are collected in each observation season. The videos were converted to individual frames for processing.
2. *Below 0.1% false-negative-rate per image and 0% false-negatives per seasonal imagery*: The key goal was to detect every individual whale at least once within all available imagery for a given season. That is, if a whale was missed in a frame, there were many other frames or images (in the same or different video clip) where MWD would detect the same whale for the photo-ID purposes with high certainty. Due to the nature of the encounters and whale behaviour, our testing confirmed that in the small number of whales that were missed, every individual whale was detected in other identifiable imagery from the corresponding season. The accuracy we aimed for and indeed achieved was that all possible whale IDs were found, and any identifiable whales that were “missed” were found in another part of the video/imagery. This means that there was a 0% error rate, after missed whales were cross checked. This very high accuracy performance requirement was the main practical challenge of this study.
3. *Below 1% false-positive rate*: MWD should have sufficiently low false-positive rate, where the classification error of less than 1% was deemed acceptable. Thus, at least 99% of negative (missing

whale) images or video frames would be correctly filtered out. Note the chosen trade-off in favour of the lowest possible false-negative rate (not missing a whale), rather than a balanced number of false-negatives and false-positives.

4. *Practical training times and processing speed:* MWD could be trained and then process the current yearly volume of imagery in the matter of days, where 300 GB, 1.8 TB and 1.9 TB were collected in 2017, 2018 and 2019, respectively.

4.3. Dwarf Minke Whale Detector

Each video was converted into a sequence of individual images. The available imagery also contained approximately 50,000 digital photographs. The adopted per-image classification approach worked identically regardless if a given image was originally a video frame or a digital photograph.

The goal of this study was to develop and deploy an exceptionally accurate underwater whale classification CNN, where the per-image false-positive and false-negative rates were required to be below 1% and 0.1%, respectively. Therefore, any experimentation with custom hand-crafted CNN architectures were considered an unnecessary and unproductive risk, which could only be justified if the existing *off-the-shelf* CNNs could not deliver the required accuracy and processing speed.

One of the simplest modern *deep* CNN architectures is the VGG CNN [38], which is built essentially from only two types of neural network layers: (3×3) convolutional and (2×2) maximum pooling layers. In Reference [10], its sixteen-layer version (VGG16) was used together with the FCN-8s [33] segmentation CNN to verify that an individual dwarf minke whale could be recognized in the collection of 1320 images containing 75 other whales. However, the requirements for the fast training and processing times ruled out using VGG via a segmentation CNN (for example, FCN-8s [10,33]), which require much longer training times (compared to a classification CNN). Therefore, and as per detailed justification in this paper's introduction, only the readily-available classification CNNs were considered.

The fastest training times are typically achieved by the *knowledge-transfer* (also known as *transfer learning*) [39] when training a classification CNN, which was already trained on the ImageNet [40] collection of images. Such ImageNet-trained (or simply *pretrained*) CNNs are commonly available for download and when used could shorten the training times from days to hours, which was an important consideration for this project. Following Reference [34], VGG [38] and ResNet-50 [17] CNNs were converted to the *approximate* object-localisation CNNs by replacing the ImageNet-trained last layers with a single 1×1 convolution layer followed by a *sigmoid* activation, see the architecture summary in Table 1. Additionally, and only while training, a global maximum pooling layer was added to train the CNNs with zero target values (for negative images) and the target values of one (for the positive images). The standard binary cross-entropy was used as the loss function. When used for prediction/testing, the CNNs produced heat-maps, see examples in Figure 4. The VGG [38] and ResNet-50 [17] CNNs (rather than the Xception CNN [18]) were selected here to confirm that the utilized approach [34] was not specific to only Xception CNN (used in Reference [34]) and was generic in nature.

The RAdam [41] version of the Adam [42] algorithm was used as the training optimizer since Adam is known to be more forgiving to non-optimal learning rates when compared to the Stochastic Gradient Descent [43,44]. Furthermore, RAdam did not require the learning-rate warm-up stage [41]. After experimenting with step-function drops [17,19,45], exponential-decay, and cosine-decay [46,47] for the learning rate annealing schedules, the simplest possible linear-decay schedule was utilized throughout this study, where the Adam learning rate was reduced linearly from its initial value ($l = 1 \times 10^{-3}$) to the fine-tuning level ($l = 1 \times 10^{-5}$) over 100 epochs. The main identified advantage of the linear annealing was its robustness to non-optimal learning rates, where identical starting ($l_{\max} = 1 \times 10^{-3}$) and finishing ($l_{\min} = 1 \times 10^{-5}$) learning rates were used for all experiments in this study. The cosine-annealing [46,47] was tuned (by cross-validation) to perform marginally better; however, it required multiple training sessions with different initial learning rates and therefore was deemed not practical for this study. The linear and cosine schedules need the following four interlinked hyper-parameters: number of epochs, batch size, starting and finishing learning rates.

Therefore, a cyclic learning rate (CLR) schedule [48] was discarded due to its introducing yet additional hyper-parameter (the number of steps per cycle), which also required fine-tuning. Furthermore, Reference [48] reported only a relatively small (order of 1%) accuracy improvement due to the CLR, where the best reported accuracy values were 94.9%. The regularization weight decay was fixed at 1×10^{-5} and not optimized by cross-validation. Effective batch size was fixed at 32 images per batch with the actual batch size of eight images and four gradient accumulation steps. Every training session was done with 100 epochs.

In order to utilize the benefits of the *knowledge-transfer* [39] to their full potential, it is normally required to *freeze* (exclude from training) the ImageNet-trained layers and to train only the project specific last classification layer. The training is completed by unfreezing (including in training) the pretrained-layers and by training the CNN with a lower (fine-tuning) learning rate. Following Reference [49], a more convenient approach (originally popularised by FastAI [50]) was adopted by reducing the learning rates (by ten times) of the ImageNet-trained ResNet-50 or VGG-13bn layers thus avoiding the freezing/unfreezing training complication and significantly accelerating the training convergence process [49].

4.4. Training Pipeline

The VGG [38] deep learning CNN architecture is a common classification baseline and it was considered first. Furthermore, in addition to the VGG structural simplicity, various *pretrained* VGG versions were freely available in such popular machine learning platforms as Keras/TensorFlow, FastAI [50], and PyTorch [51]. In particular, the pretrained 11-, 13-, 16-, and 19-layer VGG versions were available in PyTorch with or without additional batch-normalization [52] layers. Historically, the batch-normalization (BN) technique [52] was discovered after the publication of VGG [38], and therefore BN does not normally appear in the *classic* versions of VGG. The BN [52] layers are currently standard components of most modern CNNs, therefore, only the BN-containing versions of VGG were considered in this study. Specifically, the VGG-13bn CNN (once converted to this study's localisation configuration, see Table 1) was selected after confirming that it performed nearly identically to the 16-layer version (VGG-16bn) using the CIFAR-10 and CIFAR-100 benchmarks [53].

The following training pipeline was implemented initially for the VGG-13bn CNN backbone and then re-used in identical fashion for the ResNet-50-based final version of the MWD. All still images (and video frames) were resized to the 256×256 shape (or to 512×512 when using ResNet-50) and converted to greyscale for both training and testing phases. The 256×256 and 512×512 image shapes were selected because the adopted VGG-13bn and ResNet-50 models could classify video frames at about 30–70 frames per second on Nvidia 1080Ti GPU (available for this study), which satisfied the project speed requirements. During the training phase, the following image augmentations [54] were used: the images were randomly rotated up to 360 degrees, flipped horizontally and scaled down by up to 50%. With 0.5 probability, either 3×3 -kernel blurring or the CLAHE [11] enhancement were applied. Most of the images were not square shaped originally (see, for example, Figure 1), therefore by square resizing them, significant shrinking or stretching distortions were introduced to whale shapes. However, since the whales were in a highly diverse range of spatial orientations, such distortions supplied an additional training image augmentation in a natural fashion.

The actual training history of the final MWD-v4 CNN is illustrated in Figure 7, where 16,471 manually verified and/or labelled images (see Table 2) were split 90% for training and 10% for validation. Note that the full range of the considered image augmentations (via the Albumentations library [54]) was applied while collecting the loss and accuracy performance metrics for the validation subset. Hence, the validation accuracy was still fluctuating between 96% and 100% values even during the last epochs of training.

To assess the training pipeline together with the MDW architecture (Table 1), five-fold cross-validation was performed, see Table 3. For each testing fold, the corresponding 20% of the final dataset was treated as a testing-holdout subset completely excluded from training. Five ResNet-50-based MWD (MWD-CV) instances were independently trained on the remaining 80% of the dataset (different for each fold). Since

the actual final MWD-v4 was trained on 90% of the dataset, each of the five MWD-CV models was run through the pipeline twice, where the second training pass started from the weights obtained from the first pass. Table 3 shows a comprehensive set of metrics obtained by classifying the corresponding test holdout images without augmentations except for converting to greyscale and resizing to the 512×512 shape. Nearly identical five-fold cross-validated test accuracy (97.94%) was achieved compared to the final MWD-v4's validation accuracy of approximately 98% (Figure 7). The cross-validated test values were somewhat less accurate than the actual operational false-negative and false-positive rates collected during the final acceptance testing by the project's marine biologists. This confirmed that the training dataset was sufficiently representative of the project's domain imagery.

Table 3. Five-fold averaged performance metrics of the final Minke Whale Detector training pipeline.

Metric	Description	Mean (\pm Std)	
TP, FP	Predicted * true (TP) and false (FP) positives	$TP = 2577.8(\pm 8.2)$	$FP = 11.0(\pm 4.1)$
FN, TN	Predicted false (FN) and true (TN) negatives	$FN = 56.8(\pm 8.5)$	$TN = 648.6(\pm 4.5)$
P	Actual test positives, $P = TP + FN$	$P = 2634.6$	
N	Actual test negatives, $N = FP + TN$		$N = 659.6$
Recall	TP/P	97.84% ($\pm 0.32\%$)	
Precision	$TP/(TP + FP)$	99.57% ($\pm 0.15\%$)	
Accuracy	$(TP + TN)/(P + N)$	97.94% ($\pm 0.22\%$)	
F1 score	$2/(1/recal + 1/precision)$	98.70% ($\pm 0.14\%$)	
ROC AUC	Area Under the ROC Curve [14]	99.58% ($\pm 0.12\%$)	

* Default 0.5 threshold was used to convert MWD's probability outputs to class predictions.

Author Contributions: Conceptualization, D.A.K., R.A.B., M.K. and S.H.; methodology, D.A.K. and R.A.B.; software, D.A.K. and D.B.E.; validation, D.A.K., N.S. and D.B.E.; resources, M.K.; data curation, D.A.K., R.A.B., D.B.E., N.S., S.H. and G.W.; writing—original draft preparation, D.A.K. and D.B.E.; writing—review and editing, D.A.K., N.S., D.B.E., R.A.B., M.K., S.H., M.I.C. and M.S.; supervision, M.K. and S.H.; project administration, M.K. and S.H.; funding acquisition, D.A.K., M.K. and M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors are profoundly grateful for the project assistance of Susan Sobotzick and Kent Adams, as well as for the contributions of passengers, crew and owners of the permitted swim-with-whales tourism vessels in the Great Barrier Reef who have helped to provide many of the dwarf minke whale images and videos used in this study. We are also deeply grateful to the many Minke Whale Project Volunteers who have helped to sort our minke images.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
GBR	Great Barrier Reef
GPU	Graphics Processing Unit
MWD	Minke Whale Detector
MWP	Minke Whale Project
Photo-ID	Photo Identification

References

1. Risch, D.; Norris, T.; Curnock, M.; Friedlaender, A. Common and Antarctic minke whales: Conservation status and future research directions. *Front. Mar. Sci.* **2019**, *6*, 247. [[CrossRef](#)]
2. Best, P.B. External characters of southern minke whales and the existence of a diminutive form. *Sci. Rep. Whales Res. Inst.* **1985**, *36*, 1–33.

3. Mangott, A.H.; Birtles, R.A.; Marsh, H. Attraction of dwarf minke whales *Balaenoptera acutorostrata* to vessels and swimmers in the Great Barrier Reef world heritage area—The management challenges of an inquisitive whale. *J. Ecotourism* **2011**, *10*, 64–76. [[CrossRef](#)]
4. Birtles, R.A.; Arnold, P.W.; Dunstan, A. Commercial swim programs with dwarf minke whales on the northern Great Barrier Reef, Australia: Some characteristics of the encounters with management implications. *Aust. Mammal.* **2002**, *24*, 23–38. [[CrossRef](#)]
5. Curnock, M.I.; Birtles, R.A.; Valentine, P.S. Increased use levels, effort, and spatial distribution of tourists swimming with dwarf minke whales at the Great Barrier Reef. *Tour. Mar. Environ.* **2013**, *9*, 5–17. [[CrossRef](#)]
6. Gedamke, J.; Costa, D.P.; Dunstan, A. Localization and visual verification of a complex minke whale vocalization. *J. Acoust. Soc. Am.* **2001**, *109*, 3038–3047. [[CrossRef](#)]
7. Arnold, P.W.; Birtles, R.A.; Dunstan, A.; Lukoschek, V.; Matthews, M. Colour patterns of the dwarf minke whale *Balaenoptera acutorostrata sensu lato*: Description, cladistic analysis and taxonomic implications. *Mem. Qld. Mus.* **2005**, *51*, 277–307.
8. Sobotzick, S. Dwarf Minke Whales in the Northern Great Barrier Reef And Implications for the Sustainable Management of the Swim-With Whales Industry. Ph.D. Thesis, James Cook University, Townsville, Australia, 2010. Available online: <https://bit.ly/2DORPRM> (accessed on 4 April 2020).
9. Arnold, P.; Marsh, H.; Heinsohn, G. The occurrence of two forms of minke whales in east Australian waters with description of external characters and skeleton of the diminutive form. *Sci. Rep. Whales Res. Inst.* **1987**, *38*, 1–46.
10. Kononov, D.A.; Hillcoat, S.; Williams, G.; Birtles, R.A.; Gardiner, N.; Curnock, M. Individual minke whale recognition using deep learning convolutional neural networks. *J. Geosci. Environ. Prot.* **2018**, *6*, 25–36. [[CrossRef](#)]
11. Zuiderveld, K. *Contrast Limited Adaptive Histogram Equalization*; Graphic Gems IV; Academic Press Professional: San Diego, CA, USA, 1994; pp. 474–485.
12. Verma, G.K.; Gupta, P. Wild animal detection using deep convolutional neural network. In *Proceedings of the 2nd International Conference on Computer Vision & Image Processing, Roorkee, India, 9–12 September 2017*; Chaudhuri, B.B., Kankanhalli, M.S., Raman, B., Eds.; Springer: Singapore, 2018; pp. 327–338. [27](#). [[CrossRef](#)]
13. Pelletier, D.; Leleu, K.; Mou-Tham, G.; Guillemot, N.; Chabanet, P. Comparison of visual census and high definition video transects for monitoring coral reef fish assemblages. *Fish. Res.* **2011**, *107*, 84–93. [[CrossRef](#)]
14. Fawcett, T. An introduction to ROC analysis. *Pattern Recogn. Lett.* **2006**, *27*, 861–874. [[CrossRef](#)]
15. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
16. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception architecture for computer vision. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016*; pp. 2818–2826. [[CrossRef](#)]
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016*; pp. 770–778. [[CrossRef](#)]
18. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017*; pp. 1800–1807. [[CrossRef](#)]
19. Zagoruyko, S.; Komodakis, N. Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC), York, UK, 19–22 September 2016*; Wilson, R.C., Hancock, E.R., Smith, W.A.P., Eds.; BMVA Press: Guildford, UK, 2016; pp. 87.1–87.12. [[CrossRef](#)]
20. Kononov, D.A.; Jahangard, S.; Schwarzkopf, L. In situ cane toad recognition. In *Proceedings of the 2018 Digital Image Computing: Techniques and Applications (DICTA), Canberra, Australia, 10–13 December 2018*; pp. 1–7. [[CrossRef](#)]
21. Olsen, A.; Kononov, D.A.; Philippa, B.; Ridd, P.; Wood, J.C.; Johns, J.; Banks, W.; Girgenti, B.; Kenny, O.; Whinney, J.; et al. DeepWeeds: A multiclass weed species image dataset for deep learning. *Sci. Rep.* **2019**, *9*, 2058. [[CrossRef](#)] [[PubMed](#)]
22. Zhang, Q.; Yang, Y.; Ma, H.; Wu, Y.N. Interpreting CNNs via decision trees. In *Proceedings of the CVPR IEEE, Long Beach, CA, USA, 16–20 June 2019*; pp. 6254–6263. [[CrossRef](#)]
23. Mahendran, A.; Vedaldi, A. Understanding deep image representations by inverting them. In *Proceedings of the CVPR IEEE, Boston, MA, USA, 7–12 June 2015*; pp. 5188–5196. [[CrossRef](#)]

24. Goebel, R.; Chander, A.; Holzinger, K.; Lecue, F.; Akata, Z.; Stumpf, S.; Kieseberg, P.; Holzinger, A. Explainable AI: The new 42? In *Machine Learning and Knowledge Extraction*; Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 295–303. [\[CrossRef\]](#)
25. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal.* **2017**, *39*, 1137–1149. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Park, M.; Yang, W.; Cao, Z.; Kang, B.; Connor, D.; Lea, M.A. Marine vertebrate predator detection and recognition in underwater videos by region convolutional neural network. In *Knowledge Management and Acquisition for Intelligent Systems*; Ohara, K., Bai, Q., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 66–80. [\[CrossRef\]](#)
27. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the CVPR IEEE, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [\[CrossRef\]](#)
28. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–12 December 2015; pp. 1440–1448. [\[CrossRef\]](#)
29. Huang, J.; Rathod, V.; Sun, C.; Zhu, M.; Korattikara, A.; Fathi, A.; Fischer, I.; Wojna, Z.; Song, Y.; Guadarrama, S.; et al. Speed/accuracy trade-offs for modern convolutional object detectors. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 3296–3297. [\[CrossRef\]](#)
30. Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Mallocci, M.; Duerig, T.; et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv* **2018**, arXiv:1811.00982.
31. Krasin, I.; Duerig, T.; Alldrin, N.; Ferrari, V.; Abu-El-Haija, S.; Kuznetsova, A.; Rom, H.; Uijlings, J.; Popov, S.; Kamali, S.; et al. OpenImages: A Public Dataset for Large-Scale Multi-Label and Multi-Class Image Classification. 2017. Available online: <https://bit.ly/34lGYLn> (accessed on 4 April 2020).
32. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525. [\[CrossRef\]](#)
33. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal.* **2017**, *39*, 640–651. [\[CrossRef\]](#)
34. Kononov, D.A.; Saleh, A.; Bradley, M.; Sankupellay, M.; Marini, S.; Sheaves, M. Underwater fish detection with weak multi-domain supervision. *IEEE IJCNN* **2019**, 1–8. [\[CrossRef\]](#)
35. Everingham, M.; Eslami, S.M.A.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [\[CrossRef\]](#)
36. Buda, M.; Maki, A.; Mazurowski, M.A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw.* **2018**, *106*, 249–259. [\[CrossRef\]](#)
37. Birtles, A.; Arnold, P.; Curnock, M.; Salmon, S.; Mangott, A.; Sobtzick, S.; Valentine, P.; Caillaud, A.; Rumney, J. Code of Practice for Dwarf Minke Whale Interactions in the Great Barrier Reef World Heritage Area. 2008. Available online: <https://bit.ly/36mObKD> (accessed on 4 April 2020).
38. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2015**, arXiv:1409.1556.
39. Oquab, M.; Bottou, L.; Laptev, I.; Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In Proceedings of the CVPR IEEE, Columbus, OH, USA, 23–28 June 2014; pp. 1717–1724. [\[CrossRef\]](#)
40. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the CVPR IEEE, Long Beach, CA, USA, 16–20 June 2009; pp. 248–255. [\[CrossRef\]](#)
41. Liu, L.; Jiang, H.; He, P.; Chen, W.; Liu, X.; Gao, J.; Han, J. On the variance of the adaptive learning rate and beyond. *arXiv* **2019**, arXiv:1908.03265.
42. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
43. Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013*; Dasgupta, S., McAllester, D., Eds.; PMLR: Atlanta, GA, USA, 2013; Volume 28, pp. 1139–1147.

44. Schaul, T.; Zhang, S.; LeCun, Y. No more pesky learning rates. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; Dasgupta, S., McAllester, D., Eds.; PMLR: Atlanta, GA, USA, 2013; Volume 28, pp. 343–351.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 630–645. [\[CrossRef\]](#)
46. Cubuk, E.D.; Zoph, B.; Mané, D.; Vasudevan, V.; Le, Q.V. AutoAugment: Learning augmentation policies from data. *arXiv* **2018**, arXiv:1805.09501.
47. Loshchilov, I.; Hutter, F. SGDR: Stochastic gradient descent with restarts. *arXiv* **2016**, arXiv:1608.03983.
48. Smith, L.N. No more pesky learning rate guessing games. *arXiv* **2015**, arXiv:1506.01186.
49. Konovalov, D.A.; Saleh, A.; Efremova, D.B.; Domingos, J.A.; Jerry, D.R. Automatic weight estimation of harvested fish from images. In Proceedings of the 2019 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Perth, Australia, 2–4 December 2019, pp. 1–7. [\[CrossRef\]](#)
50. Howard, J.; Gugger, S. Fastai: A layered API for deep learning. *Information* **2020**, *11*, 108. [\[CrossRef\]](#)
51. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
52. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
53. Krizhevsky, A. Learning Multiple Layers of Features From Tiny Images. 2009. Available online: <https://bit.ly/2HfABij> (accessed on 4 April 2020).
54. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and flexible image augmentations. *Information* **2020**, *11*, 125. [\[CrossRef\]](#)



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).