

A new method to control error rates in automated species identification with deep learning algorithms

Sébastien Villon^{a,b}, David Mouillot^{a,e}, Marc Chaumont^{b,c}, Gérard Subsol^b,
Thomas Claverie^{a,d}, Sébastien Villéger^a

a) MARBEC, Univ of Montpellier, CNRS, IRD, Ifremer, Montpellier, France

b) Research-Team ICAR, LIRMM, Univ of Montpellier, CNRS, Montpellier, France

c) University of Nîmes, Nîmes, France

d) CUFR Mayotte, Dembeni, France

e) Australian Research Council Centre of Excellence for Coral Reef Studies, James Cook University, Townsville, QLD 4811 Australia.

Contribution

S. Villon wrote the main manuscript text, prepared the figures, and carried the analysis.

Sébastien Villon, David Mouillot, Sébastien Villéger, Thomas Claverie, Marc Chaumont, Gérard Subsol designed the project and the experiments.

All authors reviewed the manuscript.

Abstract

Processing data from surveys using photos or videos remains a major bottleneck in ecology. Deep Learning Algorithms (DLAs) have been increasingly used to automatically identify organisms on images. However, despite recent advances, it remains difficult to control the error rate of such methods.

Here, we proposed a new framework to control the error rate of DLAs. More precisely, for each species, a confidence threshold was automatically computed using a training dataset independent from the one used to train the DLAs. These species-specific thresholds were then used to post-process the outputs of the DLAs, assigning classification scores to each class for a given image including a new class called “unsure”. We applied this framework to a

study case identifying 20 fish species from 13,232 underwater images on coral reefs.

The overall rate of species misclassification decreased from 22% with the raw DLAs to 2.98% after post-processing using the thresholds defined to minimize the risk of misclassification.

This new framework has the potential to unclog the bottleneck of information extraction from massive digital data while ensuring a high level of accuracy in biodiversity assessment.

Introduction

In the context of accelerating human impacts on ecosystems (Diaz et al. 2019), the capacity to monitor biodiversity at large scale and high frequency is an urgent although challenging goal (Schmeller et al. 2015). This urgency resonates with the ambition of international initiatives like the Group on Earth Observations Biodiversity Observation Network (GEO BON) and the call for monitoring Essential Biodiversity Variables (EBVs) (Pereira et al. 2013, Kissling et al. 2017).

Remote sensors are rapidly transforming biodiversity monitoring in its widest sense from individuals (Kröschel et al. 2017) to species and communities of species (Steenweg et al. 2017). In the last decade, satellites (Wulder and Coops 2014, Schulte and Pettorelli 2018), drones (Koh and Wich 2012, Hodgson et al. 2018), camera traps (Steenweg et al. 2017), or underwater cameras (Mallet and Pelletier 2014, Aguzzi et al. 2015) have been extensively deployed to record pictures or videos of aquatic and terrestrial organisms. For instance, satellite data can be used to track whale shark movements (Robinson et al. 2016) or detect whales (Cubaynes et al. 2018) while photos from airborne or underwater vehicles can deliver accurate density estimations of vulnerable organisms like mammals or sharks (Hodgson et al. 2017, Kellengerer et al. 2018).

Such massive records are also used by citizen science programs with for example public tools like inaturalist.org which share pictures and associated metadata, or fishpix (<http://fishpix.kahaku.go.jp>) which offers the possibility to upload individual fish images that are then identified by experts at the species level.

However, processing photos or videos to identify organisms is a highly demanding task, especially in underwater environments, where some particular contexts add many difficulties (e.g., visual noise due to particles and small objects, complex 3D environment, color changing according to depth, etc.). For instance, identifying all fish individuals on videos may take up to 3

hours of expert analysis per hour of video (Francour et al. 1999). Under the avalanche of new videos and images to analyse, alternatives to fish identification by humans and trained experts must be found.

Recently, an effort to use machine learning methods (Chuang et al 2016, Marini et al. 2018) and deep learning algorithms (DLAs) for ecological analysis have been made, thanks especially to computer-vision challenges on public databases of annotated photos or videos (e.g. for fish, Fish4Knowledge database (<http://groups.inf.ed.ac.uk/f4k/>) and Seaclef challenge (<https://www.imageclef.org/lifeclef/2017/sea>)).

The last generation of DLAs offer much promise for passing the bottleneck of image or video analysis through automated species identification (Li et al. 2015, Joly et al. 2017, Wäldchen and Mäder 2018, Villon et al. 2018). DLAs, and particularly convolutional neural networks (CNNs), simultaneously combine the automatic definition of image descriptors and the optimization of a classifier based on these descriptors (Lecun et al. 2015). Even though DLAs usually have a high accuracy rate, they do not provide information on the confidence of the outputs. Hence, it remains difficult to identify and control potential misclassifications which limits their application.

Misclassification of images has two types of consequences for biodiversity monitoring. On one hand, if all individuals of a given species occurring in a given community are erroneously labelled as another species also occurring in the community, this species will be incorrectly listed as absent (false absence). The risk of missing present species because of misclassification is the highest for rare species, i.e. those with the lowest abundance in terms of the number of individuals per unit area. Yet missing these rare species can be critical for ecosystem health assessment since some play important and unique roles like large parrotfishes on coral reefs (Mouillot et al. 2013) while others are invasive like the lionfish in Eastern Mediterranean Sea (Azzuro et Bariche 2017). In addition, since most species in a community are represented by a few individuals (Gaston 1994), such misclassifications could significantly lead to the underestimation of species richness. The other error associated with misclassification is when an individual of a given species is mistaken for another species not present in the community (false presence). Such misclassifications could lead to an overestimation of the abundance or geographical range of a species as well as it could artificially increase species richness, unless a species is consistently mistaken for another. Since biodiversity monitoring should be as accurate as possible, automated identification of individuals on images should provide high correct classification rates (close to 100%) even if a subset of images has not been classified by

the algorithm with sufficient confidence and must be identified by humans *a posteriori*.

Chow (1957) was the first to introduce the concept of risk for a classification algorithm. For instance, a clustering algorithm classifying an object placed in the center of a given cluster would present a low risk of misclassification, while classifying an object placed on the edge of a cluster would be highly risky. Chow (1957) proposed a classification framework, which contains $n+1$ channels as outputs, n channels for the n classes considered and an additional channel called the "rejection" channel. When the risk of misclassification is too important, the algorithm rejects the classification.

Applied to machine learning, a first method consists in learning a rejection function during the training, in parallel to the classification learning (Cortes et al., 2016, Geifman et al. 2017, Corbière et al. 2019). Another method, called a meta-algorithm, uses two algorithms, one being a classifier, and the other one analyzing the classifier outputs, to distinguish predictions with a high risk of misclassification from those with a low risk (De Stefano et al. 2000). A recent comparative study suggests that meta-algorithm-based methods are the most efficient (Kocak et al., 2017).

An extension of meta-algorithms to control the risk of misclassification is to calibrate models obtained through Machine Learning and Deep Learning algorithms. Machine Learning methods usually produce well-calibrated models for binary tasks (Niculescu-Mizil et al. 2005). The calibration consists of a matching between the score predicted by the machine-learning model and the real probability of true positives. While Deep Learning models produce more accurate classifications than other Machine learning models, these models are not well calibrated, and thus need a re-calibration to be used for real-world decisions (Guo et al. 2017). Several propositions have been made to improve the calibration of Machine Learning models through the post-processing of outputs. The Platt scaling (Platt 1999), the Histogram binning (Zadrozny 2001), the Isotonic Regression (Zadrozny 2002) and the Bayesian Binning into Quantiles (Naeini 2015) are mapping the model outputs to real accuracy probabilities. More recently, Temperature Scaling, an extension of the Platt Scaling, was used to calibrate Deep Learning models using a single parameter for all classes (Guo et al. 2017). This parameter is used, instead of the traditional *softmax* function, to convert the vector output from the neural network into a real probability.

However, such calibration methods are based on a discretization of the Deep model outputs into bins. Many bins are not useful as they only contain a few outputs with low values, whereas many high values fall in the same bin and

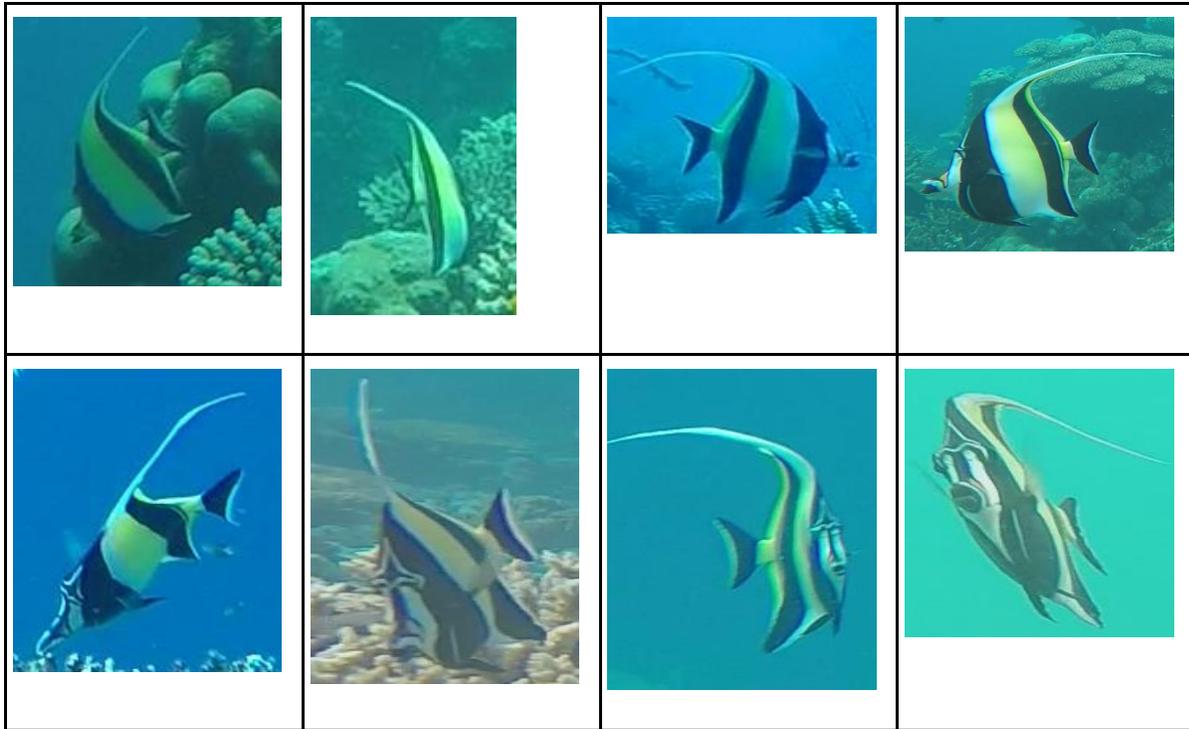
are thus not discriminated. Moreover, the choice of the number of bins is left to the user, and therefore is not optimized to the Deep model nor to a specific application (Nixon et al. 2019).

In this paper, we present a simple, yet efficient method that accounts for uncertainty in the classifier outputs. Unlike calibration methods, our approach is not changing algorithm outputs. Instead, we simply assess the behaviour of the model thanks to a validation dataset. We can then set-up a fine tuned threshold per class, allowing us to take into account that the model confidence can be highly variable between “easy” classes and “difficult” classes. Then, through the addition of a new class “unsure”, corresponding to predictions with scores lower than the predicted class threshold, we can control the coverage (total amount of images automatically identified) and misclassification rates. We applied this framework to classify 20 species of coral reef fishes in underwater images and assessed its efficiency for 3 real-case scenarios.

Material and methods

We decided to build our own dataset instead of using existing datasets (e.g. Fish4Knowledge: <http://groups.inf.ed.ac.uk/f4k/>), to be in phase with quality of videos currently used by marine ecologists. We used 3 independent fish images datasets from the Mayotte Island (Western Indian Ocean) to train and test our CNN model and our post processing method. For the 3 datasets, we used fish images extracted from 175 underwater high-definition videos which lasted between 5 and 21 minutes for a total of 83 hours. The videos were recorded in 1920x1080 pixels with GoPro Hero 3+ black and Hero 4+ black. The videos were recorded between 2 and 30 meters deep, with a broad range of luminosity, transparency, and benthic environment conditions on fringing and barrier reefs.

We extracted 5 frames per second from these videos. Then, we cropped images to include only one fish individual with its associated habitat in the background. Thus, images of the same species differed in terms of size (number of pixels), colors, body orientation, and background (e.g. other fish, reef, blue background) (Fig. 1).



*Fig. 1: Diversity of individual images and their environment for the same fish species (Moorish idol, *Zanclus cornutus*).*

We used 130 videos for the training dataset, from which we extracted a total 69,169 images of 20 different fish species (Supp. Fig. 1). We extracted between 1,134 and 7,345 images per species.

In order to improve our model, we used data augmentation (Perez et al. 2017). Each “natural” image yielded 4 more images: 2 with increased contrast (120% and 140%) and 2 with decreased contrast (80% and 60%) (Supp. fig 2). We then horizontally flipped all images to obtain our final training dataset (T_0) composed of 691,690 images (Supp. Tab. 1).

We then used two independent datasets made of different videos recorded on different days and on different sites than videos used to build the training dataset. The first dataset (T_1) contained 6,320 images from 20 videos with at least 41 images per species, and the second (T_2) contained 13,232 images from 25 videos with at least 55 images per species (Supp. Tab. 1). We then used dataset T_1 to tune the thresholds and T_2 as the test dataset. This method ensures that our results are not biased by similar acquisition conditions between the training, tuning and testing dataset and hence that algorithm performance was evaluated using a realistic full cross-validation procedure.

Building the convolutional neural network

Convolutional neural networks (CNNs) belong to the class of DLAs. For the case of species identification, the training phase is supervised, which means that the classes to identify are pre-defined by human experts while the parameters of the classifier are automatically optimized in order to accurately classify a “training” database (Lecun et al. 2015). CNNs are composed of neurons, which are organized in layers. Each neuron of a layer computes an operation on the input data and transfers the extracted information to the neurons of the next layer. The specificity of CNNs is to build a descriptor for the input image data and the classifier at the same time, ensuring they are both optimized for each other (Goodfellow et al. 2016). The neurons extracting the characteristics from the input data in order to build the descriptors are called convolutional neurons, as they apply convolutions, i.e. they modify the value of one pixel according to a linear weighted combination of the values of the neighbor pixels. In our case, each image used to train the CNN is coded as 3 matrices with numeric values describing the color component (R, G, B) of the pixel. The optimization of the parameters of the CNN is achieved during the training through a process called back-propagation. Back-propagation consists of automatically changing parameters of the CNN through the comparison between its output and the correct class of the training element to eventually improve the final classifications rate. Here we used a 100-layer CNN based on the TensorFlow (Abadi et al. 2016) implementation of ResNet (He et al. 2016). The ResNet architecture achieved the best results on ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2015, considered as the most challenging image classification competition. It is still one of the best classification algorithms, while being very easy to use and implement.

All fish images extracted from the videos to build our datasets were resized to 64x64 pixels before being processed by the CNN. Our training procedure lasted 600,000 iterations; each iteration processed a batch of 16 images, which means that the 691,690 images of the training dataset were analyzed 14 times each by the network on average. We then stopped the training to prevent from overfitting (Sarle et al., 1996), as an over fit model is too restrictive and only able to classify images that were used during the training.

Assigning a confidence score to the CNN outputs

The last layer of our architecture, as in most CNNs, is a “softmax” layer (He et al. 2016). When input data passing through the network reaches this layer, a function is applied to convert the image descriptors into a list of n scores S_i , with $i = \{1, \dots, n\}$, and n the number of learned classes (here the 20 different fish species), with the sum of all scores equal to 1. A high score means a “higher

chance” for a given image to belong to the predicted class. However, a CNN often outputs a class with a very high score (more than 0.9) even in case of misclassification. To prevent misclassifications, the classifier should thus be able to add a risk or a confidence criterion to its outputs.

Assessing the risk of misclassification by the CNN

For a given input image, a CNN returns a predicted class, in our case a fish species. As seen in the previous section, the CNN outputs a decision based on the score, without any information on the risk of making an error (i.e. a misclassification). Following De Stefano et al. (2000), we thus propose to apply a post-processing step on the CNN outputs in order to accept or reject its classification decision. The hypothesis is that the higher the similarity between an unknown image and the images used for the training, the stronger the activation in the CNN during the classification process (i.e. the higher the score is), and thus, the more robust the classification is.

For this method, the learning protocol is thus made of two consecutive steps performed on 2 independent training datasets.

In the first phase, a classification model is built by training a CNN on a given database $T0$ (Fig. 2 (a))

Then, the second phase consists of tuning a risk threshold τ_i specific to each class (i.e. each species in our case), noted i , with $i \in \{1, \dots, n\}$, using a second and independent database noted $T1$ (Fig. 2 (b)).

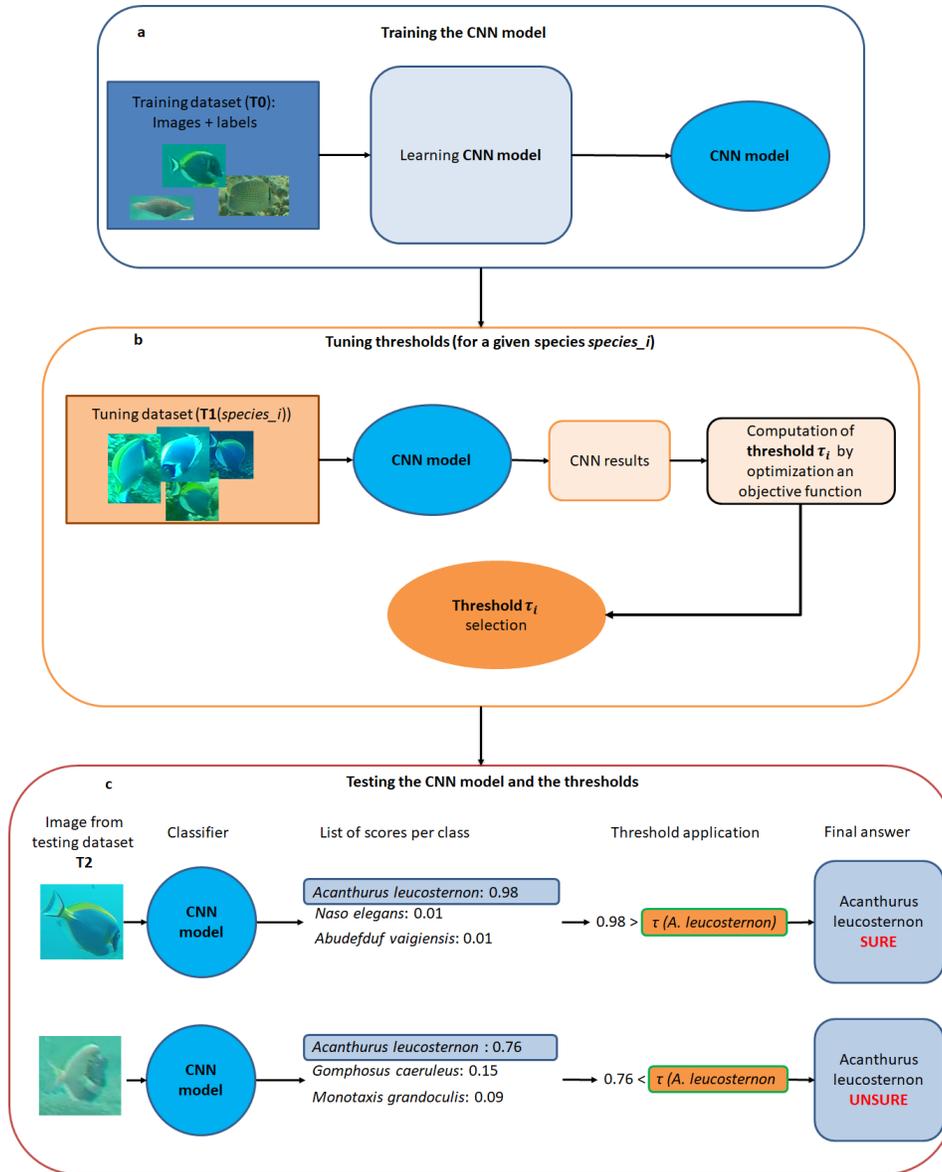


Fig. 2: Overview of the 3 parts of our framework: 2 consecutive steps for the learning phase, followed by the applicative testing step.

(a) We trained a CNN model with a training dataset (T0) composed of images and a label for each image, in our case, the species corresponding to each fish individual.

(b) Then, for each species i , we processed an independent dataset T1, with our model. For each image, we obtained the species j attributed by the CNN to the image and a classification score S_j . We have the ground truth and the result of the classification (correct/incorrect), so we can define a threshold according to the user goal. This goal is a trade-off between the accuracy of the result and the proportion of images fully processed.

(c) We then used this threshold to post-process outputs of the CNN model. More precisely, for a given image, the classifier of the CNN returns a score for each class (here for each fish species). The most likely class $C(X)$ for this image is the one with the highest score $S(X)$. We then compared this highest score $S(X)$ with the computed confidence threshold for this species ($\tau_{C(X)}$) obtained in the second phase. If the score was lower than the computed threshold that is $S(x) > \tau_{C(X)}$, then the input image was classified as "unsure". Otherwise, we kept the CNN classification.

In terms of classification, it means we transform the 2 classification options (correct, wrong) in 3 options (Fig. 3) by applying equations (9, 10).

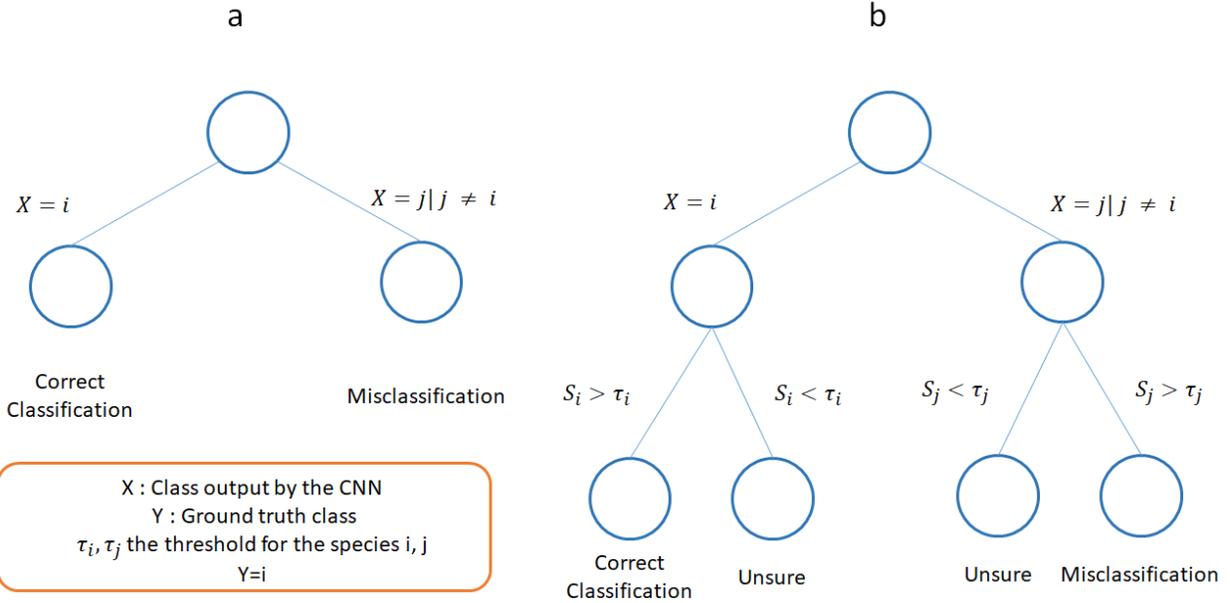


Fig.3: Impact of the post-processing framework on classification of images for a given species and a given threshold.

Usually, the classification of an image of class i can either be correct, if the model classifies it as i , or wrong, if the classifier classifies it as j with $j \neq i$ (a). We propose a post processing to set a confidence threshold for each class to obtain 3 types of results, correct, misclassified, and unsure (b). The goal is then to transform as many misclassifications as possible as "Unsure", while preventing to transform too many correct classifications "Unsure".

Computing the confidence thresholds

After the phase 1 (model training phase), for an image X of the threshold tuning dataset processed by the classifier, we obtain an output $\{C(X), S(X)\}$, where $C(X)$ is the class (i.e. species, belonging to the trained set of species) with the highest classification score $S(X)$. For this image, we know the ground truth Y in $\{1, \dots, n\}$ belonging to the same set of species classes.

So with $C(X)$ being the output class, Y the ground truth class, and $\#(.)$ the enumeration function, the standard definition for Correctly Classified images (or true positives) rate of a class i is:

$$CC_i = \frac{\#(C(X) = i \text{ AND } Y = i)}{\#Y = i} \quad (0.1)$$

We can also write the standard definition of Misclassified images rate (or false negatives) of a class i as:

$$MC_i = \frac{\#(C(X) \neq i \text{ AND } Y = i)}{\#Y = i} \quad (0.2)$$

Then, we can extend the Correct Classification rate (CC) and Misclassification (MC) rate of a species i by introducing the thresholds τ_i and by adding the Unsure Classification (UC) rate:

$$CC_i(\tau_i) = \frac{\#((C(X) = i) \text{ AND } (S(X) > \tau_i)) \text{ AND } (Y = i))}{\#(Y = i)} \quad (1)$$

$$MC_i(\tau_i) = \frac{\#((C(X) \neq i) \text{ AND } (S(X) > \tau_i)) \text{ AND } (Y = i))}{\#(Y = i)} \quad (2)$$

$$UC_i(\tau_i) = \frac{\#((C(X) = i) \text{ OR } (C(X) \neq i)) \text{ AND } (S(X) < \tau_i))}{\#(Y = i)} \quad (3)$$

For each species we have:

$$CC_i(\tau) + MC_i(\tau) + UC_i(\tau) = 1 \quad (4)$$

We can also note that the standard coverage definition (COV, the rate of images for which a classification is given) of a species i can be extend with the introduction of thresholds as threshold τ as:

$$COV_i(\tau) = CC_i(\tau) + MC_i(\tau) \quad (5)$$

The question is now to select "optimal" thresholds $\{\tau_i\}_{i=1}^{i=n}$ based on the dataset T1. This is not straightforward as is it up to user specific objective, such as minimizing MC, maximizing CC, minimizing UC... In the following, we analyze three different goals corresponding to some standard protocols in marine ecology:

The first goal $G1$ consists of keeping the best correct classification rate while reducing the misclassification error rate. For this, we used two steps. First, we identified the threshold(s) τ which maximizes $CC_i(\tau)$. Since several thresholds could reach this maximum, we get a set of threshold(s) Se_{g1} . Then, we selected the threshold with the lower $MC_i(\tau)$. This can be mathematically written as:

$$Se_{g1} = \arg \max_{\tau} CC_i(\tau) \quad (6.1)$$

$$\tau_i = \arg \min_{\tau' \text{ in } Se_{g1}} MC_i(\tau') \quad (6.2)$$

The second goal $G2$ consists in constraining the misclassification error rate to an upper bound of 5% while maximizing the correct classification rate. Reaching this goal requires to first find Se_{g2} the set of threshold(s) such as $MC_i(\tau) < 5\%$. If there is none, we considered Se_{g2} as the set of threshold(s), which minimize MC_i . Then we defined the optimal threshold τ_i by choosing the one in Se_{g2} which maximizes CC_i :

$$Se_{g2} = \tau / MC_i(\tau) < 5\% \quad (7.1)$$

$$\text{if } Se_{g1} = \emptyset \text{ then } Se_{g2} = \arg \min_{\tau} MC_i(\tau) \quad (7.2)$$

$$\tau_i = \arg \max_{\tau' \text{ in } Se_{g2}} CC_i(\tau') \quad (7.3)$$

The third goal $G3$ consists of keeping the lowest misclassification rate while raising the correct classification error rate (implying a lower coverage). First, we defined Se_{g3} as the set of threshold(s) τ that minimizes $MC_i(\tau)$. If there were several thresholds with the same minimal value, we chose τ_i as the one which maximizes CC_i :

$$Se_{g3} = \arg \min_{\tau} MC_i(\tau) \quad (8.1)$$

$$\tau_i = \arg \max_{\tau' \text{ in } Se_{g3}} CC_i(\tau') \quad (8.2)$$

For a given image X in the test dataset, the classification and post-process is sequential as follows (Fig. 2 (c)):

First, the image is given to the CNN, which outputs a list of scores, including $S(X)$ the highest score obtained by a class.

Second, for the class $C(X)$ (i.e the class with the highest classification score), the post-processing step estimates the risk of classifying the image as belonging to the class $C(X)$. If $(X) < \tau_j$, the prediction is changed to "Unsure", otherwise, it is confirmed as the class j (Fig. 2 c).

The misclassification rate for a species $Y = i$ after post-processing thus equals:

$$MC'_i = \frac{\#((C(X) \neq Y) \text{ AND } (S(X) > \tau_j)) \text{ AND } (Y = i))}{\#(Y = i)} \quad (9)$$

and the unsure classification rate equals:

$$UC'_i = \frac{\#((C(X) = j) \text{ AND } (S(X) < \tau_j)) \text{ AND } (Y = i))}{\#(Y = i)} \quad (10)$$

First, to assess the effectiveness of our framework, we processed all the images contained in $T2$ through the DL algorithm, without post processing (threshold tuning+ threshold application).

Second, we assessed whether a unique threshold for all the classes was sufficient to separate correct classifications from misclassifications for all species. For this test, we computed the distribution of correct classifications and misclassifications over scores for each species. During this study, we multiplied the softmax scores, which ranged from 0 to 1, by 100, for an easier reading.

Then, to study the impact of the post-processing method in an hypothetical ideal condition, we selected the thresholds based on the dataset $T2$ and we applied them to the same dataset $T2$. For this experiment and the following, we also measured both the Correct Classification rate and the Accuracy, defined for a species i as

$$Accuracy_i = \frac{\#((C(X) = i) \text{ AND } (S(X) > \tau_i)) \text{ AND } (Y = i))}{\#((C(X) = i) \text{ AND } (S(X) > \tau_i))}$$

The accuracy varies from 0 to 1, and increases when the number of false positives decreases and the number of true positives increases. Meanwhile, the CC rate varies from 0 to 100, and increases when the number of false negatives decreases and the number of true positives increases.

Finally, to ensure that the post-processing method was relevant for any real-life application, i.e. that thresholds are defined and tested on independent datasets, we used the dataset $T1$ for the threshold-setting phase and the dataset $T2$ for the testing phase. To assess the robustness of our method, we repeated the same experiment while switching the roles of $T1$ and $T2$. Note that we limited our experiments to the use of $T1$ and $T2$, but that it could be interesting in further work to assess the robustness of this method with datasets composed of less data.

Results

Results of the CNN model classification

The mean rate of correct classification of fish images in $T2$ by the raw CNN was of 78.0%, with rates of correct classifications per species ranging from 54.4% to 99.1% (sd= 15.16) (Tab. 1). These results are the baseline for our following experiments.

Table 1: Output of the deep learning classifier without post-processing. Percentages of correct classifications are shown for the 20 fish species. Each line shows the species name, the correct classification rate of images of this

species present in the dataset T2, the softmax score above which we have 95% of the correct classification (noted sq0.05), and the percentage of Misclassified images with score equal or above sq0.05.

Species	Test dataset T2 (% of correct classifications)	Softmax score for the 0.05 quantile of Correct Classification (sq0.05)	% of Misclassification for sq0.05
<i>Chaetodon trifasciatus</i>	87.80	99.91	20
<i>Chaetodon trifascialis</i>	90.00	99.98	11.11
<i>Naso brevirostris</i>	54.14	99.92	29.91
<i>Chaetodon guttatissimus</i>	85.50	99.82	10.77
<i>Thalassoma hardwicke</i>	90.90	99.92	0
<i>Pomacentrus sulfureus</i>	90.14	99.88	0
<i>Oxymonacanthus longirostris</i>	96.43	99.98	0
<i>Monotaxis grandoculis</i>	57.10	98.78	34.1
<i>Zebrasoma scopas</i>	63.04	96.78	19.92
<i>Abudefduf vaigiensis</i>	99.07	99.99	0
<i>Amblyglyphidodon indicus</i>	58.78	92.85	22.04
<i>Acanthurus lineatus</i>	59.72	99.98	16.38

<i>Chromis ternatensis</i>	59.61	86.74	26.98
<i>Chromis opercularis</i>	61.29	99.00	16.67
<i>Gomphosus caeruleus</i>	75.72	99.84	33.33
<i>Acanthurus leucosternon</i>	86.15	99.94	16.65
<i>Halichoeres hortulanus</i>	82.93	99.96	16.33
<i>Naso elegans</i>	93.24	99.78	6.46
<i>Chaetodon auriga</i>	87.05	99.98	10.77
<i>Zanclus cornutus</i>	81.36	99.68	9.1
Mean	78.00	98.64	17.49
Standard Deviation	15.16	3.27	10.84

Images obtained softmax scores between 41 and 100 with 80% of images classified with a score between 60 and 100 (Fig. 4 a). The distribution of correct classifications and misclassifications among scores was highly variable among species (Fig. 4 b, c, Tab.1).

We plotted the results for all species (a), and for 2 species, the Brown unicornfish (*Naso brevirostris*) (b) and the Maldives damselfish (*Amblyglyphidodon indicus*) (c). We also plotted the 5% bottom line for each type of classification. We used violin plots for the visualisation. Violin plot are histograms with inverted axis allowing a graphical visualisation of a distribution, with the number of individuals on the Y axis and their value on X axis. The borders of the shapes show the number of individuals while the dots show the local density" (Hintze and Neslon 1998).

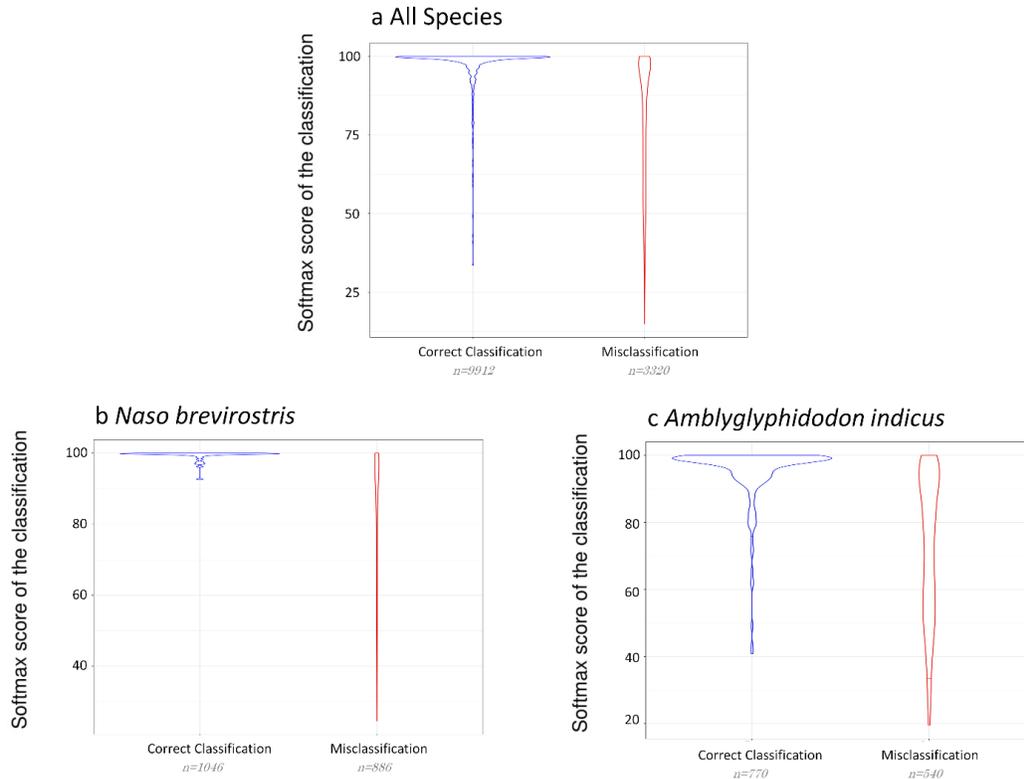


Fig 4: Distribution of correct classifications and misclassifications of fish images with respect to the score from the CNN model.

Benchmark of the threshold fine-tuning method

For each species i , we computed CC_i , MC_i , UC_i values while varying the threshold. We computed and applied the thresholds on $T2$, according to equations 6, 7, 8, 9 and 10. As the score varied from 0 to 99.9, the misclassification rate decreased to 0.9% (Fig. 5). This decrease was mainly compensated by the increasing rate of unsure classifications between 0 and 99.9 of classification scores.

Indeed, the rate of correct classifications experienced little variation along this distribution of threshold scores, remaining between 74-78% for threshold scores between 0 and 99.8 and decreasing to 61% for threshold scores >99.8 . However, correct, wrong, and unsure classification rates were highly variable among species (Supp. Tab. 2).

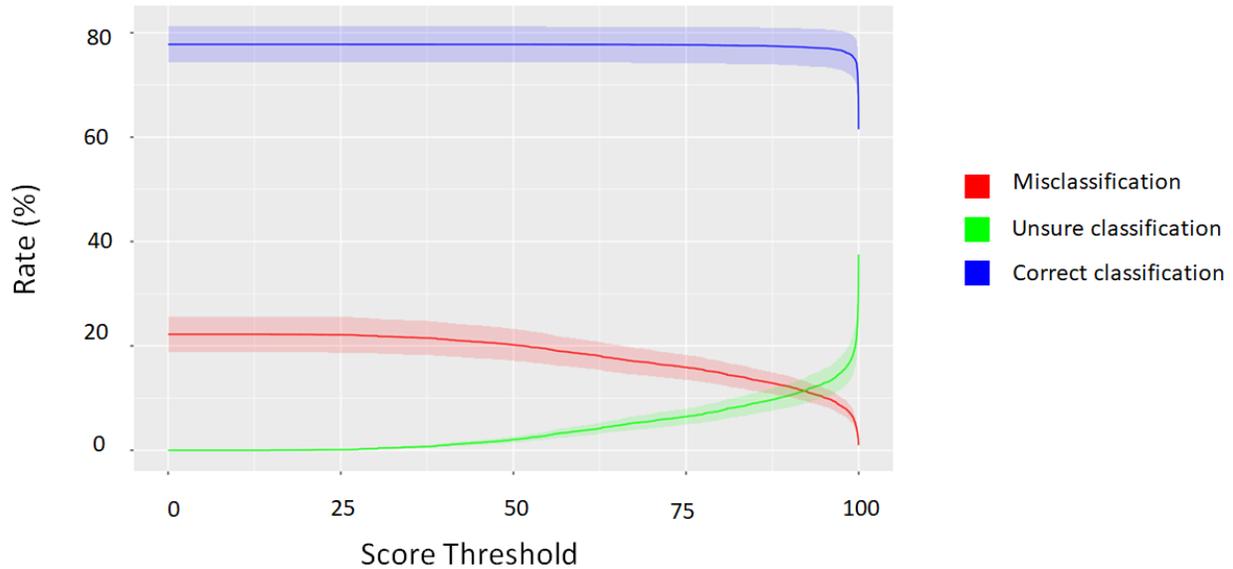


Fig. 5: Average distribution of correct, wrong, and unsure classifications for all species along a gradient of confidence threshold score.

For the first goal G1, we defined the thresholds (one per species) to minimize the misclassification with $CC_i = \max CC_i$. We obtained a mean rate of 78% (standard deviation= 15.15%) of correct classifications, 10.81% (s.d= 8.15%) of unsure classifications, and 11.19% (s.d= 9.58%) of misclassifications (Fig. 6. a).

For the second goal G2, we maximized the correct classifications while constraining the misclassification error rate to an upper bound of 5% (if possible). We obtained a rate of 75.47% (s.d= 17.83%) of correct classifications, 17.88% (s.d= 14.22%) of unsure classifications, and 6.66% (s.d= 6.44%) of misclassifications.

For the third goal G3, we maximized the number of correct classifications with $MC_i = \min MC_i$. We obtained a rate of 68.21% (s.d= 22.41%) of correct classifications, 29.71% (s.d= 22.14%) of unsure classifications, and 2.07% (s.d= 3.20%) of misclassifications, on average. Compared to the first goal, we decreased the rate of correct classifications by 8.9% and the rate of misclassifications by 2.6% (Supp. Tab. 4).

The accuracy of the goals G1, G2, and G3 were, on average, higher than the raw accuracy (0.53) with respectively 0.72, 0.89 and 0.94. (Tab. 2).

The thresholds showed higher variations among species for G1, with values ranging from 33.46 to 99.97, than for G3 for which values ranged from 99.86 to 99.98 among the 20 species (Supp. Tab. 2 and 3).

Table 2: Accuracy of the models without post-processing, and with post processing according to our goals, with thresholds tuned and applied on T2. Each line shows the result for a species, with: the species name, the accuracy of the model without post processing, and the accuracy of the model with post processing according to the 3 goals defined earlier.

Species	Raw Accuracy	G1 Accuracy	G2 Accuracy	G3 Accuracy
<i>Abudefduf vaigiensis</i>	0.51	0.65	0.9	0.97
<i>Acanthurus leucosternon</i>	0.61	0.69	0.87	0.96
<i>Acanthurus lineatus</i>	0.87	0.91	0.97	0.97
<i>Amblyglyphidodon indicus</i>	0.08	0.74	0.94	0.98
<i>Chaetodon auriga</i>	0.95	0.99	1	1
<i>Chaetodon guttatissimus</i>	0.16	0.84	0.95	0.98
<i>Chaetodon trifascialis</i>	0.97	0.87	0.95	0.96
<i>Chaetodon trifasciatus</i>	0.56	0.62	0.79	0.97
<i>Chromis opercularis</i>	0.68	0.8	0.96	1
<i>Chromis ternatensis</i>	0.01	0.44	0.79	0.9
<i>Gomphosus caeruleus</i>	0.24	0.31	0.54	0.72
<i>Halichoeres hortulanus</i>	0.51	0.59	0.8	0.93
<i>Monotaxis grandoculis</i>	0.77	0.81	0.96	0.99
<i>Naso brevirostris</i>	0.02	0.9	0.96	1
<i>Naso elegans</i>	0.89	0.92	0.97	0.97
<i>Oxymonacanthus longirostris</i>	0.36	0.46	0.89	0.85
<i>Pomacentrus sulfureus</i>	0.52	0.7	0.91	0.95
<i>Thalassoma hardwicke</i>	0.78	0.85	0.93	0.95
<i>Zanclus cornutus</i>	0.55	0.68	0.87	1

<i>Zebrasoma scopas</i>	0.61	0.7	0.81	0.81
Mean	0.53	0.72	0.89	0.94
Standard Deviation	0.30	0.18	0.10	0.07

Application of the method

For a real cross-validation experiment, thresholds were set using *T2* and then applied on *T1*. The correct, wrong and unsure classification rates were very close (difference < 2.6%) to those obtained with the benchmark situation (Supp. Tab. 5).

The proposed post-processing was able to decrease the misclassification rate by at least 10.05%, for all goals, and 19.02% at most compared to the raw output of the Deep Learning model (Fig. 6. b). The accuracy followed the same tendency, with an average accuracy for G1, G2 and G3 respectively equal to 0.74, 0.81 and 0.92 (Tab. 3).

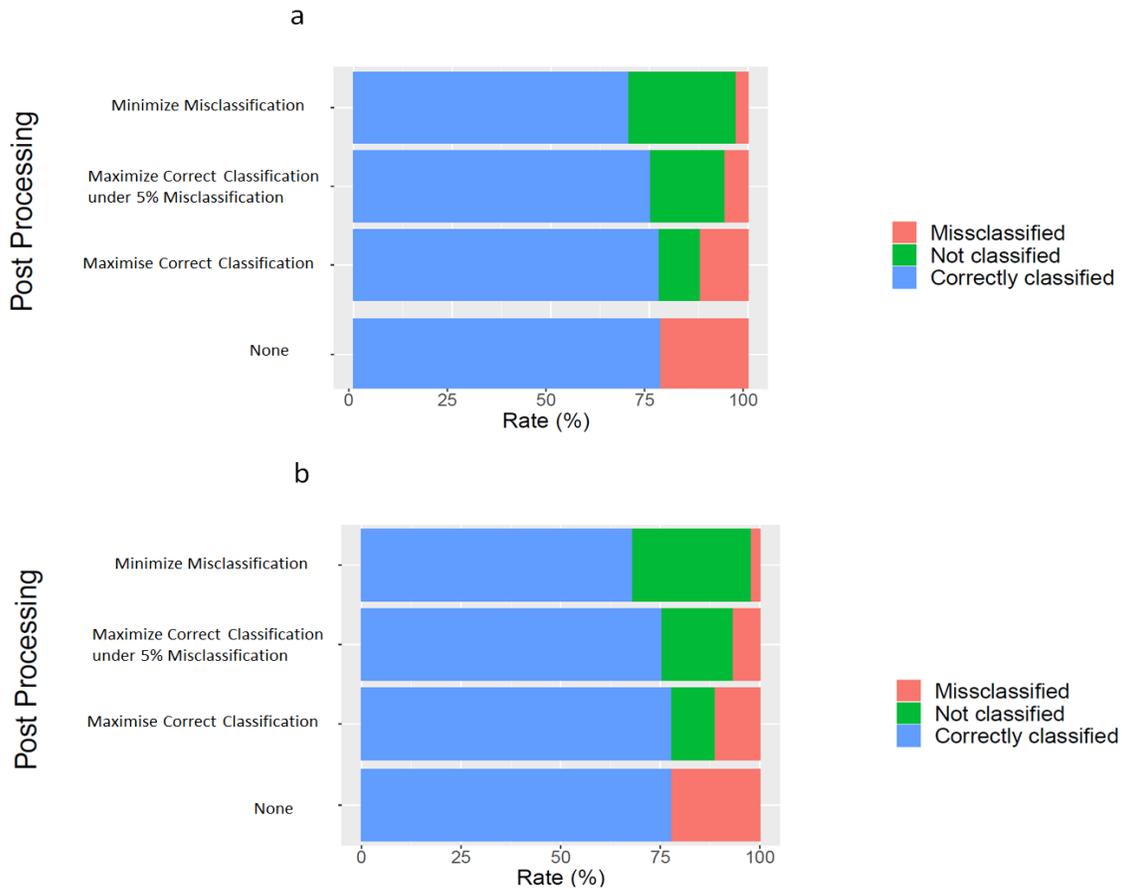


Fig. 6: Benchmark scenario and cross-validation classification rates.

We compare results obtained by tuning the thresholds on $T2$ and using $T2$ as a testing set (a) and real-life scenario obtained by tuning the thresholds on $T1$ and using $T2$ as a testing set (b).

For sub-figure: From top to bottom, rates of correct classifications, misclassifications, and unsure classifications for each post-processing: 1) Goal 1: Minimizing misclassification with $CC_i = \max CC_i$, 2) Goal 2: maximizing correct classifications under the constraint of having less than 5% of misclassifications, 3) Goal 3: maximizing correct classification with $MC_i = \min MC_i$, 4) No post-Processing.

Table 3: Accuracy of the model without post-processing, and with post processing according to our goals, on the cross-validation, with thresholds tuned on $T1$ and applied on $T2$. Each line shows the result for a species, with: the species name, the accuracy of the model without post processing, and the accuracy of the model with post processing according to the 3 goals defined earlier.

Species	Raw Accuracy	G1 Accuracy	G2 Accuracy	G3 Accuracy
<i>Abudefduf vaigiensis</i>	0.51	0.61	0.92	0.97
<i>Acanthurus leucosternon</i>	0.61	0.7	0.92	0.94
<i>Acanthurus lineatus</i>	0.87	0.91	0.95	0.97
<i>Amblyglyphidodon indicus</i>	0.08	0.72	0.97	0.97
<i>Chaetodon auriga</i>	0.95	0.99	0.95	1
<i>Chaetodon guttatissimus</i>	0.16	0.88	0.72	0.96
<i>Chaetodon trifascialis</i>	0.97	0.9	0.96	0.98
<i>Chaetodon trifasciatus</i>	0.56	0.62	0.43	0.85
<i>Chromis opercularis</i>	0.68	0.83	0.03	1
<i>Chromis ternatensis</i>	0.01	0.47	0.97	0.87
<i>Gomphosus caeruleus</i>	0.24	0.31	0.89	0.75
<i>Halichoeres hortulanus</i>	0.51	0.57	1	0.9
<i>Monotaxis grandoculis</i>	0.77	0.82	0.99	0.98
<i>Naso brevirostris</i>	0.02	0.92	0.89	1

<i>Naso elegans</i>	0.89	0.91	0.99	0.97
<i>Oxymonacanthus longirostris</i>	0.36	0.46	0.72	0.8
<i>Pomacentrus sulfureus</i>	0.52	0.92	0.71	0.91
<i>Thalassoma hardwicke</i>	0.78	0.94	0.77	0.94
<i>Zanclus cornutus</i>	0.55	0.64	0.4	0.98
<i>Zebrasoma scopas</i>	0.61	0.66	0.99	0.8
Average	0.53	0.74	0.81	0.93
Standard Deviation	0.30	0.19	0.25	0.07

Finally, we also performed the same experiment while switching $T1$ and $T2$ roles (Supp. Tab. 6, 7, 8). For each goal, the unsure classification rate was higher after the switch (+3.8% for G1, +4.4% for G2, and +8.9% for G3), implying lower scores were obtained in both correct classification (-3.5%, -5%, -7.3%) and misclassification, with the exception of the 2nd goal (-0.2%, +0.6%, -1.6%).

Discussion

Biodiversity monitoring is experiencing a revolution with the emergence of new sensors (light, noise, image, environmental DNA) that generate massive datasets and require powerful and accurate treatment tools. Indeed, species misclassifications must be controlled and limited to avoid false negatives or absences i.e., missing species that are actually present and false positives or presences i.e., detecting species that are actually absent.

In this paper, we demonstrated that the risk of misclassification by CNN algorithms can be measured and controlled in a post-processing step to provide more accurate identification of species on pictures. Such post-processing can be applied with any classifier as long as the output is a vector of scores. Reducing the misclassification rate is at the detriment of the correct classification rate and increases “unsure” classifications, which implies a low coverage and a greater human effort needed to identify unclassified individuals. Hence, there is a trade-off between a more secure (less misclassifications) or a more automatic (more classifications) method so species thresholds can be set according to the goal or priority of the study or the availability and time of experts. Here we define three main goals which represent archetypal study cases. The first goal, maximizing the correct classification rate but not limiting misclassifications, can be applied when

avoiding false negatives is more important than detecting false positives. This can be the case for monitoring invasive species, since the priority is to detect the first occurrence of such invasive individuals with potential deleterious consequences on native biodiversity and ecosystem functioning (Catford 2018) particularly on islands (Spatz 2017, Leclerc 2018). For instance, the Indo-Pacific predator lionfish (*Pterois volitans* and *P. miles*) has invaded most reefs of the Western Atlantic and depleted many native prey populations, and are starting to spread in the Eastern Mediterranean Sea (Azzuro et Bariche 2017).. To better anticipate the impact of such species, ecosystem managers need to be aware of the first occurrence on reefs and can thus accept having “false alarms”. The same constraint applies for detection of particular or emblematic individuals, like Whale Sharks, through photo-identification (McKinney 2017) where the primary goal is to avoid missing an occurrence. In both ecological cases, experts will eventually validate the few false positive identifications of targeted organisms by the algorithm to discard them.

The second goal, maximizing the correct classification rate while limiting misclassifications at 5% maximum per species, can be applied when avoiding false negatives and false positives are both important. This is the trade-off scenario that requires the least human effort and that can process massive datasets with few errors. It can be recommended to analyze long videos (>2 hours) for monitoring biodiversity metrics that are weakly influenced by undetected species (rare or classified as “unsure”), like the assessment of taxonomic or functional diversity (Mouillot et al. 2013), and that can feed initiatives like the Group on Earth Observations Biodiversity Observation Network (GEO BON) and provide robust estimates of Essential Biodiversity Variables (EBVs) (Pereira et al. 2013, Kissling et al. 2017).

The third goal, minimizing the misclassification rate, can be applied when detecting false positives is more problematic than avoiding false negatives, which creates many “unsure” classifications. This can be the case when priority is to accurately analyze a relatively small dataset with the support of many experts who can help to identify species on potentially a high number of “unsure” images. For instance, assessing abundance of all species within a given area to explain ecosystem functioning (e.g. Maire et al. 2018) or to monitor changes in species relative abundances (e.g. Newbold 2018) requires a minimum number of misclassifications.

Whatever the goal, our framework is highly flexible and can be adapted by tuning the species thresholds regulating the trade-off between classification robustness and coverage in an attempt to monitor biodiversity through big datasets where species are unidentified. To unclog the bottleneck of information extraction about organism forms, behaviors and sounds from

massive digital data, machine learning algorithms, and particularly the last generation of deep learning algorithms, offer immense promises. Here we propose to help the users to control their error rates in ecology. This is a valuable addition to the ecologist's toolkit towards a routine and robust analysis of big data and real-time biodiversity monitoring from remote sensors. With this control of error rate in the hands of users, Deep Learning Algorithms can be used for real applications, with acceptable and controlled error rates, lower than any state of the art fully automatic process, while fixing the effort by human experts to correct algorithm mistakes.

References

- [1] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., & Kudlur, M. (2016, November). Tensorflow: a system for large-scale machine learning. In OSDI (Vol. 16, pp. 265-283).
- [2] Aguzzi, J., Doya, C., Tecchio, S., De Leo, F. C., Azzurro, E., Costa, C., ... & Favali, P. (2015). Coastal observatories for monitoring of fish behaviour and their responses to environmental changes. *Reviews in fish biology and fisheries*, 25(3), 463-483.
- [3] Azzurro, E., & Bariche, M. (2017). Local knowledge and awareness on the incipient lionfish invasion in the eastern Mediterranean Sea. *Marine and Freshwater Research*, 68(10), 1950-1954.
- [4] Catford, J. A., Bode, M., & Tilman, D. (2018). Introduced species that overcome life history tradeoffs can cause native extinctions. *Nature communications*, 9(1), 2131.
- [5] Chuang, M. C., Hwang, J. N., & Williams, K. (2016). A feature learning and object recognition framework for underwater fish images. *IEEE Transactions on Image Processing*, 25(4), 1862-1872.
- [6] Corbière, C., Thome, N., Bar-Hen, A., Cord, M., Pérez, P. (2019). Addressing Failure Prediction by Learning Model Confidence. arXiv e-prints (arXiv:1910.04851).
- [7] Cortes, C., DeSalvo, G., & Mohri, M. (2016). Boosting with abstention. In *Advances in Neural Information Processing Systems* (pp. 1660-1668).
- [8] De Stefano, C., Sansone, C., & Vento, M. (2000). To reject or not to reject: that is the question-an answer in case of neural classifiers. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(1), 84-94.

[9]Díaz, S., Settele, J., Brondízio, E. S., Ngo, H. T., Agard, J., Arneeth, A., . & Garibaldi, L. A. (2019). Pervasive human-driven decline of life on Earth points to the need for transformative change. *Science*, 366(6471).

[10] Fernandez, M., Fernández, N., García, E.A., Guralnick, R.P., Isaac, N.J.B., Kelling, S., Los, W., McRae, L., Mihoub, J.-B., Obst, M., Wee, B. & Hardisty, A.R. (2018). Building essential biodiversity variables (EBV s) of species distribution and abundance at a global scale. *Biological reviews*, 93(1), 600-625.

[11] Francour, P., Liret, C. & Harvey, E. (1999). Comparison of fish abundance estimates made by remote underwater video and visual census. *Naturalista sicil*, 23, 155–168.

Froese, R., & Pauly, D. (Eds.). (2000). *FishBase 2000: Concepts Designs and Data Sources* (Vol. 1594). WorldFish

[12] Gaston, K. J. (1994). What is rarity?. In *Rarity* (pp. 1-21). Springer, Dordrecht

[13] Geifman, Y., & El-Yaniv, R. (2017). Selective classification for deep neural networks. In *Advances in neural information processing systems* (pp. 4878-4887).

[14] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1). Cambridge: MIT press.

[15] Green, S. J., Akins, J. L., Maljković, A., & Côté, I. M. (2012). Invasive lionfish drive Atlantic coral reef fish declines. *PloS one*, 7(3), e32596.

[16] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017, August). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (pp. 1321-1330). JMLR. org.

[17] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[18] Hintze, J. L., & Nelson, R. D. (1998). Violin plots: a box plot-density trace synergism. *The American Statistician*, 52(2), 181-184.

[19] Hodgson, J. C., Mott, R., Baylis, S. M., Pham, T. T., Wotherspoon, S., Kilpatrick, A. D., ... & Koh, L. P. (2018). Drones count wildlife more accurately and precisely than humans. *Methods in Ecology and Evolution*, 9(5), 1160-1167.

- [20] Hodgson, A., Peel, D., & Kelly, N. (2017). Unmanned aerial vehicles for surveying marine fauna: assessing detection probability. *Ecological Applications*, 27(4), 1253-1267.
- [21] Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W. P., ... & Müller, H. (2017, September). Lifeclef 2017 lab overview: multimedia species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages* (pp. 255-274). Springer, Cham.
- [22] Kellenberger, B., Marcos, D., & Tuia, D. (2018). Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning. *Remote sensing of environment*, 216, 139-153.
- [23] Kocak, M. A., Ramirez, D., Erkip, E., & Shasha, D. E. (2017). SafePredict: A Meta-Algorithm for Machine Learning That Uses Refusals to Guarantee Correctness. arXiv preprint arXiv:1708.06425.
- [24] Koh, L. P., & Wich, S. A. (2012). Dawn of drone ecology: low-cost autonomous aerial vehicles for conservation. *Tropical Conservation Science*, 5(2), 121-132.
- [25] Kröschel, M., Reineking, B., Werwie, F., Wildi, F., & Storch, I. (2017). Remote monitoring of vigilance behavior in large herbivores using acceleration data. *Animal Biotelemetry*, 5(1), 10.
- [26] Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems* (pp. 6402-6413).
- [27] Leclerc, C., Courchamp, F., & Bellard, C. (2018). Insular threat associations within taxa worldwide. *Scientific reports*, 8(1), 6393.
- [28] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436.
- [29] Levi, D. M. (2008). Crowding—An essential bottleneck for object recognition: A mini-review. *Vision research*, 48(5), 635-654.
- [30] Li, X., Shang, M., Qin, H., & Chen, L. (2015, October). Fast accurate fish detection and recognition of underwater images with fast r-cnn. In *OCEANS'15 MTS/IEEE Washington* (pp. 1-5). IEEE.
- [31] Lyons, K. G., & Schwartz, M. W. (2001). Rare species loss alters ecosystem function—invasion resistance. *Ecology letters*, 4(4), 358-365.

- [32] Mallet, D., & Pelletier, D. (2014). Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952–2012). *Fisheries Research*, 154, 44-62.
- [33] Maire, E., Villéger, S., Graham, N. A., Hoey, A. S., Cinner, J., Ferse, S. C., ... & Sandin, S. A. (2018). Community-wide scan identifies fish species associated with coral reef services across the Indo-Pacific. *Proceedings of the Royal Society B: Biological Sciences*, 285(1883), 20181167.
- [34] Marini, S., Fanelli, E., Sbragaglia, V., Azzurro, E., Fernandez, J. D. R., & Aguzzi, J. (2018). Tracking fish abundance by underwater image recognition. *Scientific reports*, 8(1), 1-12.
- [35] McKinney, J. A., Hoffmayer, E. R., Holmberg, J., Graham, R. T., Driggers III, W. B., de la Parra-Venegas, R., ... & Dove, A. D. (2017). Long-term assessment of whale shark population demography and connectivity using photo-identification in the Western Atlantic Ocean. *PloS one*, 12(8), e0180495.
- [36] Mouillot, D., Bellwood, D. R., Baraloto, C., Chave, J., Galzin, R., Harmelin-Vivien, M., ... & Paine, C. T. (2013). Rare species support vulnerable functions in high-diversity ecosystems. *PLoS biology*, 11(5), e1001569.
- [37] Naeini, M. P., Cooper, G., & Hauskrecht, M. (2015, February). Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.
- [38] Newbold, T., Hudson, L. N., Contu, S., Hill, S. L., Beck, J., Liu, Y., ... & Purvis, A. (2018). Widespread winners and narrow-ranged losers: Land use homogenizes biodiversity in local assemblages worldwide. *PLoS biology*, 16(12), e2006841.
- [39] Niculescu-Mizil, A., & Caruana, R. (2005, August). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning* (pp. 625-632). ACM.
- [40] Nixon, J. Dusenberry, M., Zhang, L. Jerfel, G. Tran, D. (2019). Measuring Calibration in Deep Learning. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (pp. 38-41).
- [41] O'Connell, A. F., Nichols, J. D., & Karanth, K. U. (Eds.). (2010). *Camera traps in animal ecology: methods and analyses*. Springer Science & Business Media.
- [42] Pereira, H. M., Ferrier, S., Walters, M., Geller, G. N., Jongman, R. H. G., Scholes, R. JBruford, M.W., Brummitt, N., Butchart, S.H.M., Cardoso, A.C.,

Coops, N.C., Dulloo, E., Faith, D.P., Freyhof, J., Gregory, R.D., Heip, C., Höft, R., Hurtt, G., Jetz, W., Karp, D.S., McGeoch, M.A., Obura, D., Onoda, Y., Pettoirelli, N., Reyers, B., Sayre, R., Scharlemann, J.P.W., Stuart, S.N., Turak, E., Walpole, M. & Wegmann, M. (2013). Essential biodiversity variables. *Science*, 339(6117), 277-278.

[43] Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:1712.04621.

[44] Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3), 61-74.

[45] Robinson, D. P., Bach, S. S., Abdulrahman, A. A., & Al-Jaidah, M. (2016). Satellite tracking of whale sharks from Al Shaheen. *QScience Proceedings*, 52.

[46] Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., & Harvey, E. (2016). Fish species classification in unconstrained underwater environments based on deep learning. *Limnology and Oceanography: Methods*, 14(9), 570-585.

[47] Sarle, W. S. (1996). Stopped training and other remedies for overfitting. *Computing science and statistics*, 352-360.

[48] Schulte to Bühne, H., & Pettoirelli, N. (2018). Better together: Integrating and fusing multispectral and radar satellite imagery to inform biodiversity monitoring, ecological research and conservation science. *Methods in Ecology and Evolution*, 9(4), 849-865.

[49] Schmeller, D. S., Julliard, R., Bellingham, P. J., Böhm, M., Brummitt, N., Chiarucci, A. A., Couvet, D., Elmendorf, S., Forsyth, D.M., Moreno, J.G., Gregory, R.D., Magnusson, W.E., Martin, L.J., McGeoch, M.A., Mihoub, J.B., Pereira, H.M., Proença, V., van Swaay, C.A.M., Yahara, T. & Belnap, J. (2015). Towards a global terrestrial species monitoring program. *Journal for Nature Conservation*, 25, 51-57.

[50] Spatz, D. R., Zilliacus, K. M., Holmes, N. D., Butchart, S. H., Genovesi, P., Ceballos, G., ... & Croll, D. A. (2017). Globally threatened vertebrates on islands with invasive species. *Science advances*, 3(10), e1603080.

[51] Steenweg, R., Hebblewhite, M., Kays, R., Ahumada, J., Fisher, J. T., Burton, C., ... & Brodie, J. (2017). Scaling-up camera traps: monitoring the planet's biodiversity with networks of remote sensors. *Frontiers in Ecology and the Environment*, 15(1), 26-34.

- [52] Varshney, K. R. (2011, June). A risk bound for ensemble classification with a reject option. In *Statistical Signal Processing Workshop (SSP), 2011 IEEE* (pp. 769-772). IEEE.
- [53] Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., & Villéger, S. (2018). A Deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48, 238-244.
- [54] Wäldchen, J., & Mäder, P. (2018). Plant species identification using computer vision techniques: A systematic literature review. *Archives of Computational Methods in Engineering*, 25(2), 507-543.
- [55] Wulder, M. A., & Coops, N. C. (2014). Make Earth observations open access: freely available satellite imagery will improve science and environmental-monitoring products. *Nature*, 513(7516), 30-32.
- [56] Zadrozny, B., & Elkan, C. (2001, June). Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In *Icml (Vol. 1, pp. 609-616)*.
- [57] Zadrozny, B., & Elkan, C. (2002, July). Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 694-699)*. ACM.

Acknowledgements

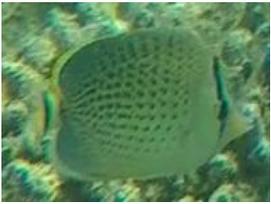
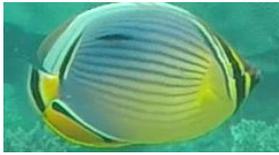
We thank Emily S. Darling and Matthew J. McLean for taking on their time to comment our work, and help us to improve the manuscript, the GNUM for helping us to annotate the data and Clément Desgenetez who supported the annotation work.

This work benefited from the Montpellier Bioinformatics Biodiversity platform supported by the LabEx CeMEB, an ANR "Investissements d'avenir" program (ANR-10-LABX-04-01).

The CEMEB Laboratory of Excellency of Montpellier funded this study through a PhD grant to S Villon. NVidia supported this study by providing GPU device through the GPU Grant Program.

Supplementary

Supp. Fig. 1: The 20 reef fish species considered in the study.

			
<i>Abudefduf vaigiensis</i>	<i>Acanthurus leucosternon</i>	<i>Acanthurus lineatus</i>	<i>Amblyglyphidodon indicus</i>
			
<i>Chaetodon auriga</i>	<i>Chaetodon guttatissimus</i>	<i>Chaetodon trifascialis</i>	<i>Chaetodon trifasciatus</i>
			
<i>Chromis opercularis</i>	<i>Chromis ternatensis</i>	<i>Gomphosus caeruleus</i>	<i>Halichoeres hortulanus</i>

			
<i>Monotaxis grandoculis</i>	<i>Naso brevirostris</i>	<i>Naso elegans</i>	<i>Oxymonacanthus longirostris</i>
			
<i>Pomacentrus sulfureus</i>	<i>Thalassoma hardwicke</i>	<i>Zanclus cornutus</i>	<i>Zebrasoma scopas</i>

Supp. Fig 2: Example of training dataset augmentation

Each original image is transformed 9 times using flips and different contrast enhancements

	
Original	Original flipped
	
Less contrast (80%)	Less contrast on flipped image (80%)

	
<p>Less contrast (60%)</p>	<p>Less contrast on flipped image (60%)</p>
	
<p>More contrast (120%)</p>	<p>More contrast on flipped image (120%)</p>
	
<p>More contrast (140%)</p>	<p>More contrast on flipped image (140%)</p>

Supp. Tab. 1: Number of images per species in our 3 datasets (after data augmentation).

Family	Species	Training dataset <i>T0</i>	First dataset <i>T1</i>	Second dataset <i>T2</i>
Acanthuridae	<i>Acanthurus leucosternon</i>	32,590	235	491
Acanthuridae	<i>Acanthurus lineatus</i>	10,080	114	864
Acanthuridae	<i>Naso brevirostris</i>	11,340	539	1932
Acanthuridae	<i>Naso elegans</i>	73,450	1,436	3,896
Acanthuridae	<i>Zebrasoma scopas</i>	49,700	48	579
Chaetodontidae	<i>Chaetodon auriga</i>	21,340	737	502
Chaetodontidae	<i>Chaetodon guttatissimus</i>	11,820	221	68
Chaetodontidae	<i>Chaetodon trifascialis</i>	52,340	41	630
Chaetodontidae	<i>Chaetodon trifasciatus</i>	44,210	71	82
Labridae	<i>Gomphosus caeruleus</i>	31,310	57	173
Labridae	<i>Halichoeres hortulanus</i>	31,920	40	287
Labridae	<i>Thalassoma hardwicke</i>	49,510	181	275
Lethrinidae	<i>Monotaxis grandoculis</i>	38,930	797	1,422
Monacanthidae	<i>Oxymonacanthus longirostris</i>	25,530	54	55

Pomacentridae	<i>Abudefduf vaigiensis</i>	51,240	376	216
Pomacentridae	<i>Amblyglyphidodon indicus</i>	11,880	636	1,310
Pomacentridae	<i>Chromis opercularis</i>	15,250	81	93
Pomacentridae	<i>Chromis ternatensis</i>	36,400	300	156
Pomacentridae	<i>Pomacentrus sulfureus</i>	54,090	270	142
Zanclidae	<i>Zanclus cornutus</i>	38,760	86	59
TOTAL		691,690	6,320	13,232

Supp. Tab. 2: Values of misclassification scores without post processing, and after processing with the threshold selected by optimizing the correct classification rate (threshold tuned and tested on the same dataset).

Species	Without processing	post Goal1	Misclassification rate	Unsure rate
<i>Chaetodon trifasciatus</i>	12.19	94.23	6.10	6.10
<i>Chaetodon trifascialis</i>	10	99.83	6.35	3.65
<i>Naso brevirostris</i>	45.86	33.47	34.36	11.49
<i>Chaetodon guttatissimus</i>	14.59	99.73	14.49	0
<i>Thalassoma hardwicke</i>	9.09	96.15	1.45	8
<i>Pomacentrus sulfureus</i>	9.85	99.66	2.82	7.04
<i>Oxymonacanthus longirostris</i>	3.57	99.97	3.57	0
<i>Monotaxis grandoculis</i>	42.89	40.86	27.78	15.12
<i>Zebrasoma scopas</i>	36.96	66.78	19.17	19.51
<i>Abudefduf vaigiensis</i>	0.92	99.71	0.46	0.46
<i>Amblyglyphidodon indicus</i>	41.22	40.86	18.55	23.36
<i>Acanthurus lineatus</i>	40.28	98.74	23.15	17.13
<i>Chromis ternatensis</i>	40.38	33.47	12.18	31.41
<i>Chromis opercularis</i>	38.71	97.52	19.35	21.50
<i>Gomphosus caeruleus</i>	24.28	99.21	16.18	8.09
<i>Acanthurus leucosternon</i>	13.85	96.15	7.94	5.90
<i>Halichoeres hortulanus</i>	17.07	98.86	9.75	8.36
<i>Naso elegans</i>	6.8	33.47	3.90	2.93
<i>Chaetodon auriga</i>	12.95	99.8	6.37	6.57
<i>Zanclus cornutus</i>	18.64	99.71	5.08	13.56

Supp. Tab. 3: Values of misclassification scores without post processing, and after processing with the threshold selected by optimizing the Misclassification rate (threshold tuned and tested on the same dataset).

Species	Without processing	post Goal 3	Misclassification rate	Unsure rate
<i>Chaetodon trifasciatus</i>	12.19	94.22	0	12.19
<i>Chaetodon trifascialis</i>	10	94.63	3.65	6.35
<i>Naso brevirostris</i>	45.86	99.98	12.73	38.72
<i>Chaetodon guttatissimus</i>	14.59	99.84	11.59	7.25
<i>Thalassoma hardwicke</i>	9.09	99.39	0	10.55
<i>Pomacentrus sulfureus</i>	9.85	99.98	0.70	23.24
<i>Oxymonacanthus longirostris</i>	3.57	99.98	0	3.57
<i>Monotaxis grandoculis</i>	42.89	99.98	3.66	62.59
<i>Zebrasoma scopas</i>	36.96	99.9	1.90	51.64
<i>Abudefduf vaigiensis</i>	0.92	99.98	0.46	0.93
<i>Amblyglyphidodon indicus</i>	41.22	99.98	1.22	66.34
<i>Acanthurus lineatus</i>	40.28	99.94	9.26	32.18
<i>Chromis ternatensis</i>	40.38	99.98	0	75
<i>Chromis opercularis</i>	38.71	99.65	1.08	43.01
<i>Gomphosus caeruleus</i>	24.28	99.87	4.05	24.28
<i>Acanthurus leucosternon</i>	13.85	99.97	2.44	16.50
<i>Halichoeres hortulanus</i>	17.07	99.79	3.13	14.98
<i>Naso elegans</i>	6.8	99.98	0.38	16.30
<i>Chaetodon auriga</i>	12.95	99.97	3.39	13.15
<i>Zanclus cornutus</i>	18.64	99.95	0	25.42

Supp. Tab. 4: Rates of unsure, correct, and misclassifications for each goal, with a threshold learned and applied on the same dataset.

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	10.8	17.88	29.71
Misclassifications	11.19	6.66	2.07
Correct classifications	78	75.47	68.22

Supp. Tab. 5: For each case, the first number shows the result shown obtained with thresholds tuned in real cross validation, and the second number corresponds to the difference between benchmark conditions and real cross validation.

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	10.51 (-0.3)	18.80(+0.92)	27.21(-2.5)
Misclassifications	11.95(+0.77)	5.77(-0.89)	2.98(+0.91)
Correct classifications	77.53(-0.46)	75.43(-0.03)	69.81(+1.59)

Supp. Tab. 6: Difference between 1) results obtained with the classifier without post processing and 2) results obtained with post processing with a threshold learned on an independent dataset (cross-validation). For each case, the number shown corresponds to the results obtained with cross-validation threshold minus the results obtained without post processing.

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	10.51	18.80	27.21
Misclassifications	-10.04	-16.23	-19.01
Correct classifications	-0.46	-2.57	-8.19

Supp. Tab. 7: Classification results of our model without post processing.

Species	Dataset 1 (<i>T1</i>)
<i>Chaetodon trifasciatus</i>	0.96
<i>Chaetodon trifascialis</i>	0.71
<i>Naso brevirostris</i>	0.45
<i>Chaetodon guttatissimus</i>	0.49
<i>Thalassoma hardwicke</i>	0.84
<i>Pomacentrus sulfureus</i>	0.90
<i>Oxymonacanthus longirostris</i>	0.87
<i>Monotaxis grandoculis</i>	0.61
<i>Zebrasoma scopas</i>	0.69
<i>Abudefduf vaigiensis</i>	0.88
<i>Amblyglyphidodon indicus</i>	0.62
<i>Acanthurus lineatus</i>	0.83

<i>Chromis ternatensis</i>	0.78
<i>Chromis opercularis</i>	0.68
<i>Gomphosus caeruleus</i>	0.72
<i>Acanthurus leucosternon</i>	0.84
<i>Halichoeres hortulanus</i>	0.92
<i>Naso elegans</i>	0.90
<i>Chaetodon auriga</i>	0.71
<i>Zanclus cornutus</i>	0.91
Average	76.33

Supp. Tab. 8: Rates of unsure, correct, and misclassifications for each goal. The table shows the results obtained when we tuned the thresholds on T2 and applied them on T1 (cross validation).

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	14.29	23.22	36.14
Misclassifications	11.72	6.36	1.42
Correct classifications	73.99	70.43	62.44