THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITE DE MONTPELLIER

En Écologie Fonctionnelle

École doctorale : GAIA

Unité de recherche : MARBEC (MARine Biodiversity, Exploitation and Conservation)

Estimation automatisée sur vidéos de la biodiversité et de l'abondance des poissons coralliens

Présentée par Sébastien Villon

Le 25 novembre 2019

Sous la direction de Marc Chaumont et David Mouillot

Devant le jury composé de

José Antonio García-Charton, Prof., Universidad de Murcia	Rapporteur
Sébastien Lefèvre, Prof., Université Bretagne-Sud, IRISA	Rapporteur
Daniel Barthélémy, DR INRA, AMAP	Examinateur
Ronan Fablet, Prof., IMT Atlantique	Examinateur
Diane Lingrand, MCF, Polytech'Nice-Sophia, I3S	Examinatrice
Marc Chaumont, MCF, Université de Nîmes, LIRMM	Co-Directeur
David Mouillot, Prof., Université de Montpellier, MARBEC	Co-Directeur
Thomas Claverie, MCF, Centre Universitaire de Mayotte, MARBEC	Invité
Gérard Subsol, CR CNRS, LIRMM	Invité
Sébastien Villéger, CR CNRS, MARBEC	Invité

UNIVERSITÉ DE MONTPELLIER

Résumé

Les récifs coralliens soutiennent une forte biodiversité en poissons (environ 7000 espèces) qui est la source de plusieurs services écosystémiques comme l'apport en protéines via la pêche, la régulation des flux de matière mais aussi le support d'activités récréatives comme la plongée. Cependant, ces poissons subissent des pressions croissantes comme la surexploitation par la pêche et la destruction du corail par réchauffement climatique. Dans ce contexte, un des enjeux majeurs de l'écologie marine est d'estimer précisément la biodiversité, l'abondance et la biomasse de ces poissons récifaux et ce, avec une fréquence temporelle permettant de détecter les modifications liées aux changements environnementaux, aux pressions anthropiques et aux stratégies de gestion (e.g. réserves marines). Jusqu'à récemment, le recensement des poissons récifaux s'effectuait principalement en plongée au cours desquelles l'observateur identifiait toutes les espèces visibles et estimait leurs abondances (nombre d'individus). Ce protocole induit des limites comme la durée et la profondeur des plongées ainsi que des erreurs ou des biais liés à l'expérience du plongeur qui ne sont pas quantifiables ou corrigibles a posteriori. Face à ces limitations, les récents développements technologiques dans la prise de vidéos sous-marines en haute définition à moindre coût offrent des protocoles beaucoup moins contraignants. Cependant, il n'existe à l'heure actuelle aucun moyen rapide et fiable d'analyser ces quantités de données ce qui empêche l'essor de ces suivis vidéos à grande échelle. Au cours de cette thèse, nous avons mis en place des algorithmes d'identification et de localisation automatiques de poissons dans des vidéos sous-marines. L'ensemble du processus fut abordé, depuis les campagnes terrain permettant de récolter les vidéos à l'annotation de ces données afin de les rendre exploitables par des algorithmes d'apprentissage profond (ou *Deep Learning*), à la conception des modèles, au test de ces modèles et au traitement des sorties des différents modèles. Nous avons ainsi récolté plus de 380.000 images appartenant à plus de 300 espèces de poissons récifaux. Nous avons développé des méthodes d'identification précises (94% de bonnes classifications) pour 20 espèces parmi les plus présentes sur les récifs coralliens autour de Mayotte, ainsi que des méthodes de post-traitement permettant de détecter et de supprimer les erreurs commises par le modèle (diminuant ainsi le taux d'erreur jusqu'à 2%). Nous avons aussi développé un algorithme de détection permettant de localiser plus de 84% des individus présents à l'image sur une vidéo.

Abstract

Coral reefs are home of a great fish biodiversity (approximately 7000 species). This biodiversity is the source of many vital ecosystem services such as protein intakes for local populations, nutrients cycle or regulation of algae abundancy. However, increasing human pressure through over-fishing and global warming is destroying both fish populations and their habitats. In this context, monitoring the coral reef fish biodiversity, abundancy and biomass with precision is one of the major issues for marine ecology. To face the increasing pressure and fast globals changes, such monitoring has to be done at a large sclae, temporally and spatially. Up to date, most of fish underwater census is achieved through diving, during which the diver identify fish species and count them. Such manual census induces many constraints (depth and duration of the dive) and biais due to the diver experience. These biais (mistaking fish species or over/under estimating fish populations) are not quantifiable nor correctable. Today, thanks to the improvement of high resolution, low-cost, underwater cameras, new protocoles are developed to use video census. However, there is not yet a way to automaticaly process these underwater videos. Therefore, the analysis of the videos remains a bottleneck between the data gathering through video census and the analysis of fish communities. During this thesis, we developped automated methods for detection and identification of fish in underwater videos with Deep Learning based algorithm. We work on all aspects of the pipeline, from video acquisition, data annotation, to the models and post-processings conception, and models testing.

Today, we have gather more than 380,000 images of 300 coral reef species. We developped an identification model who successfully identified 20 of the most common species on Mayotte coral reefs with 94% rate of success, and post-processing methods allowing us to decrease the error rate down to 2%. We also developped a detection method allowing us to detect up to 84% of fish individuals in underwater videos.

Remerciements

Je tiens en premier lieu à remercier mon Jury de thèse, mes rapporteurs José Antonio García-Charton et Sébastien Lefèvre, ainsi que mes examinateurs Daniel Barthélémy, Ronan Fablet et Diane Lingrand pour avoir pris le temps d'examiner mes travaux de thèse et ce manuscrit.

Je suis particulièrement reconnaissant envers mes directeurs, Marc Chaumont qui m'a donné ma chance lors du stage de Master 2 qui a conduit à cette thèse, et David Mouillot qui aura été mon parrain de l'écologie marine, et qui a été d'un soutien et d'une motivation incroyable pendant la troisième année de thèse ! Je remercie aussi mes co-encadrants, Sébastien Villéger qui a tenté de refaire toute mon éducation scientifique, Thomas Claverie qui m'a fait passer mon baptême du feu (dans l'eau) et Gérard Subsol qui m'a suivi et conseillé depuis 2015. Je tiens aussi à noter que cette thèse à l'interface entre les deux domaines, informatique et écologie marine, était un exercice très particulier et *challenging* pour tous les participants, et je suis particulièrement content des efforts des membres de chaque domaine pour mener à bien ce projet.

Je remercie aussi tous mes compagnons doctorants qui m'ont supporté en plus de supporter leur propre thèse.
Les informaticiens : Lionel avec qui j'ai partagé mon bureau, 8 ans d'études , 3 ans de co-pilotage, et probablement tous les sujet de discussions possible ; Marion (la fit-partner), Mehdi, Quentin, qui auront fait partie avec moi cette génération sportive de l'équipe ICAR, Seb et Ahmad qui auront pris le contre-pied de cette génération, Jérôme dans les traces duquel il n'a pas été facile d'aller, Pauline, Florentin, (Dr) Alex (et les histoires d'épinards)... Les écologues : Eva qui a été mon modèle à suivre en tant que doctorant en écologie, mes compagnons de bureau, Marie-Charlotte (et son requin), Fabrice, Kien, Théo, Julia (beaucoup de personnes pour un bureau pour 4), (Dr) Fabien...

Je remercie aussi tous les membres permanents de MARBEC, du LIRMM et du CUFR de Mayotte qui m'ont accompagné au long de ma thèse et avec qui j'ai pu échanger sur des sujets plus ou moins scientifiques, mais toujours éducatifs (!), Laure et Nicolas (mes sauveurs sur R), Remy, Fabien et Jimmy qui auront supporté une partie du développement

lié à ma thèse ; Clément qui a fait un travail incroyable pendant 3 ans grâce auquel toutes les bases de données de la thèse ont pu être construites, mon partenaire d'aventures Angela ; Raphael, Nacim (la relève) et Camille qui ont guidé et géré l'effort d'annotation de toutes nos données pendant 3 ans, et plus généralement toutes les personnes qui ont participé à cet effort d'annotation (plus de 30 personnes au total) : Quentin, Paul, Gaël, Matthias, Lucie, Rùben, Erwan, Alexandre, Hugo, Laura, Sheherazade...

Je tiens aussi à remercier l'ensemble des personnels administratifs de l'Université de Montepllier, du LIRMM, de MARBEC et du CUFR de Mayotte qui m'ont aidé pour toutes mes démarches au cours de cette thèse, ainsi que le LabEX CEMEB pour avoir financé ces travaux de thèse.

Je remercie finalement mes parents, sans l'éducation et le soutien inconditionnel desquels je n'aurai pas pu réaliser ces travaux, ni accéder à cette thèse.

À Kaylyn pour m'avoir suivi (littéralement) pendant 4 ans d'aventures.

Sommaire

Chapitre 1

Introduction

1.1 Importance des récifs coralliens et de leur communautés de poissons

Les communautés de poissons coralliens se développent au sein de récifs composés d'assemblages de coraux, d'algues, et d'éponges. Les coraux sont des animaux marins invertébrés vivant en colonie, qui, grâce à leur fonction de bio-constructeurs, forment une matrice calcaire tridimensionnelle complexe.

FIGURE 1.1 – Exemples d'écosystèmes coralliens. Les matrices 3D complexes créées par les assemblages de coraux fournissent un habitat et une protection à de nombreuses espèces.

La majorité des coraux ont besoin de lumière et vivent dans des eaux peu profondes (< 60 mètres) et tropicales (entre 17° et 29°C). Parmi les plus grands systèmes coralliens au monde on retrouve la grande barrière de corail en Australie qui couvre une superficie de 344 400 km^2, soit environ la moitié de la superficie de la France, ou la barrière de corail à Belize (Caraïbes). Au travers des territoires et départements d'outre-mer, la France

possède une grande surface de récif coralliens, quasiment 10% de la surface mondiale, avec notamment des structures remarquables telles que le lagon de Nouvelle-Calédonie (2ème plus grand lagon mondial) ou la double barrière récifale de Mayotte.

Malgré la faible surface des océans qu'ils occupent (0,1%), les récifs coralliens abritent entre 550 000 et 1 330 000 espèces multicellulaires [Fisher et al., 2015] dont environ 6 000 espèces de poissons [Mouillot et al., 2014], soit ¼ de la totalité des espèces de poissons marins [Allsopp et al., 2008]. Cette richesse est en partie due à la structure complexe et tridimensionnelle des coraux (illustrée en Fig. 1.1), qui offre des niches écologiques (protection, nourriture, habitat) pour de nombreuses espèces, en particulier les individus de petite taille (juvéniles ou espèces cryptobenthiques [Brandl et al., 2018]).

En plus de son intérêt écologique, les récifs coralliens constituent un écosystème essentiel à l'homme via les services qu'ils procurent. En effet, les récifs coralliens offrent de nombreuses ressources aux millions de gens qui en dépendent directement [Rogers et al., 2018]. Des dizaines de millions d'individus (Environ 39% de la population mondiale (repartie dans 100 pays) habite à moins de 100 kilomètres des côtes [Cesar et al., 2003]) dépendent des écosystèmes côtiers et plus particulièrement des poissons récifaux pour vivre en tant que source de nourriture ou de revenus via le tourisme, la pêche ou les activités de loisir [Hughes et al., 2003] [Moberg and Folke, 1999] [Salvat, 1992] (Fig. 1.2)... De plus, les barrières de corail protègent les infrastructures humaines côtières, fournissant une protection naturelle en diminuant l'intensité les courants et les vagues [Harris et al., 2018] (par exemple dans l'océan Indien, où les houles cycloniques et australes sont présentes et particulièrement fortes).

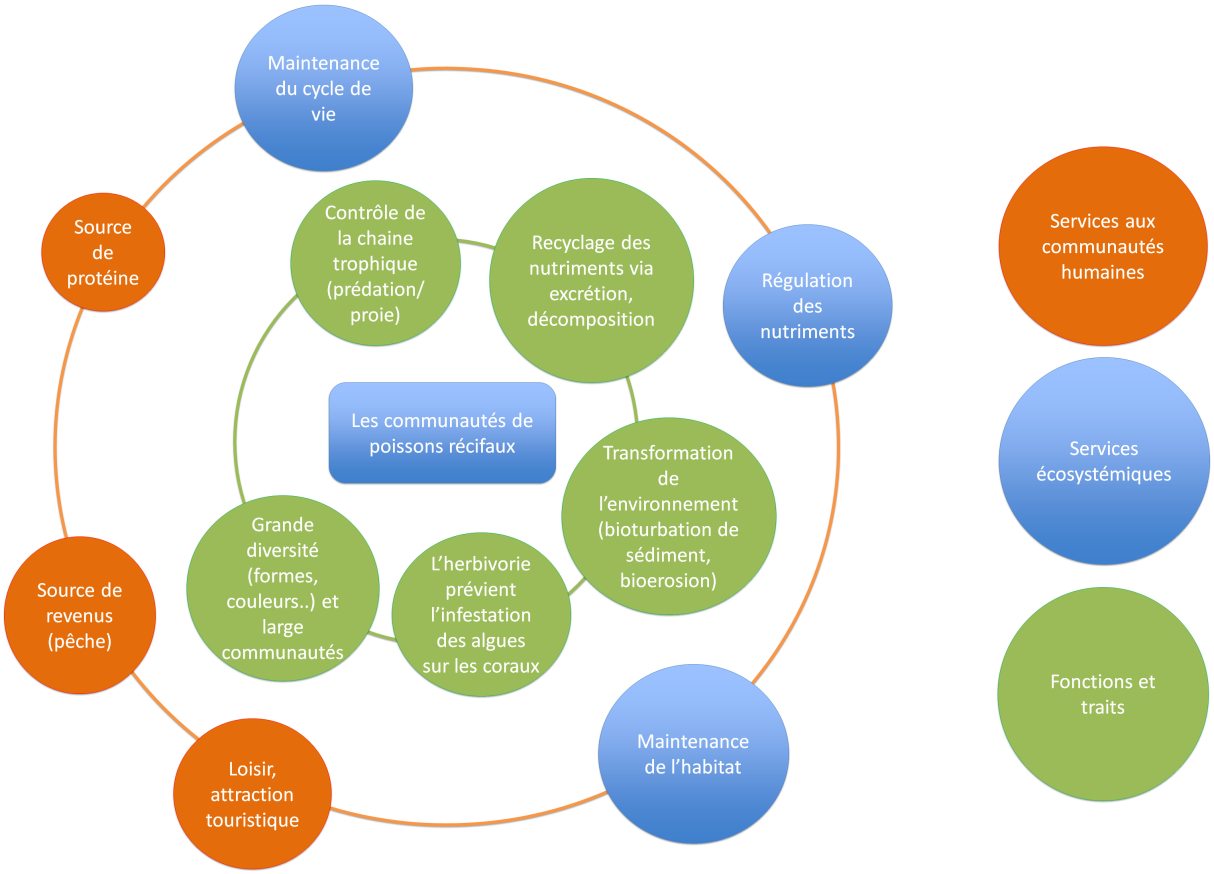


FIGURE 1.2 – Exemples de services des communautés de poissons coralliens fournis aux écosystèmes récifaux et à l'humain. Le cercle interne représente les fonctions remplies par ces communautés, et le cercle externe représente les services rendus par ces fonctions.

1.2 Vulnérabilité des poissons coralliens aux changements globaux

Alors que les coraux sont nécessaires au développement, à la richesse et à la biomasse des communautés de poisson coralliens, cet habitat subit de nombreuses perturbations dues aux changements globaux [Gordon et al., 2018] et on estime qu'environ 75% des récifs coralliens mondiaux sont menacés [Burke et al., 2011]. Ces changements globaux concernent en particulier les vagues de chaleur (e.g. El Niño) qui affectent durablement les récifs coralliens [Graham et al., 2011] [Hughes et al., 2017] [Leggat et al., 2019]. En 1998, 16% des récifs mondiaux ont été détruits par des épisodes de blanchissement dus à une forte augmentation de la température de l'eau de surface [Dimitrov, 2002]. Ainsi uujourd'hui, 83% des récifs coralliens de l'Indo-Pacifique ont été exposés à un blanchissement depuis 2014 [Darling et al., 2019]. Ces épisodes pourraient devenir annuels avant 2050 [Cesar et al., 2003], dépassant de loin la capacité de résilience des récifs coralliens. Lors de leur synthèse, [Belwood et al., 2004] présentent une analyse de l'état de l'art du sujet, et mettent en avant la diminution de la couverture corallienne en corrélation avec l'augmentation du nombre de

coraux touchés par des épisodes de blanchissement et les invasions d'Acanthaster pourpres (*Acanthaster planci*), une espèce d'étoile de mer se nourrissant quasi exclusivement de corail [Bellwood et al., 2004] (Fig.1.3).

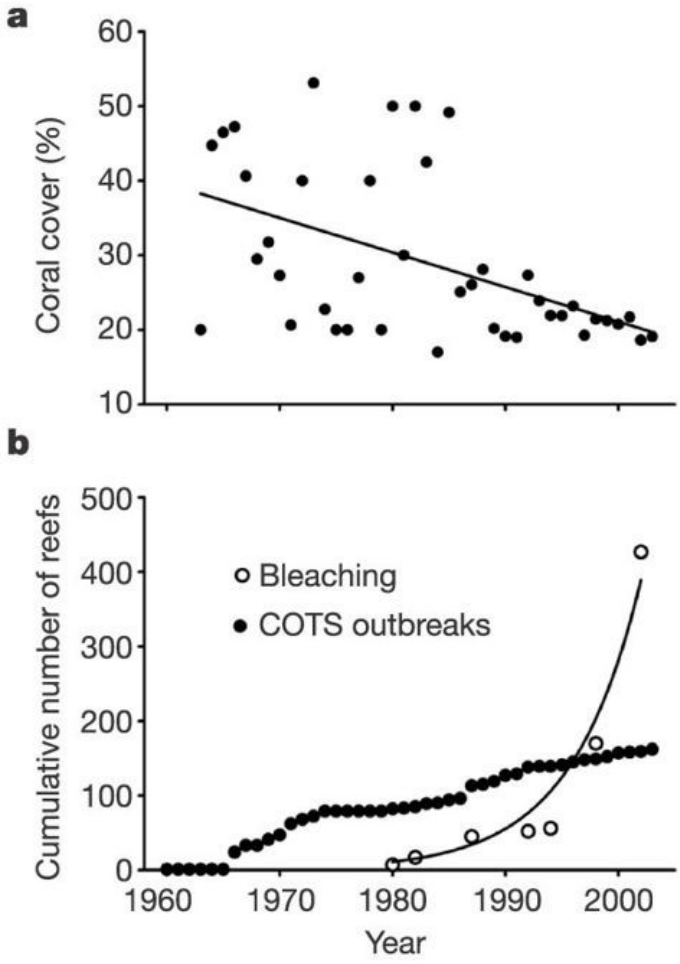


FIGURE 1.3 – Corrélation entre la diminution de la couverture corallienne et l'augmentation des perturbations les affectant (blanchissement, invasions d'Acanthaster pourpres (*Crow-of-Thorns Strafish*, COTS)). [Bellwood et al., 2004]

Les causes de ces dégradations des récifs sont multiples. Localement, le tourisme non contrôlé [Grigg, 1991], la pollution [Johannes, 1975], et les maladies [Peters, 2015] affectent les structures coralliennes, mais aussi les communautés de poissons, entraînant une diminution de la biomasse et la disparition des fonctions écologiques remplies par certaines espèces [Bellwood et al., 2011] [Cinner et al., 2018] [Edgar et al., 2018] (Fig. 1.4).

Au delà de la destruction des habitats des poissons récifaux, la sur-pêche, les méthodes de pêche destructives et le braconnage [Robinson et al., 2017] [Roberts, 1995] affectent directement les communautés de poissons.

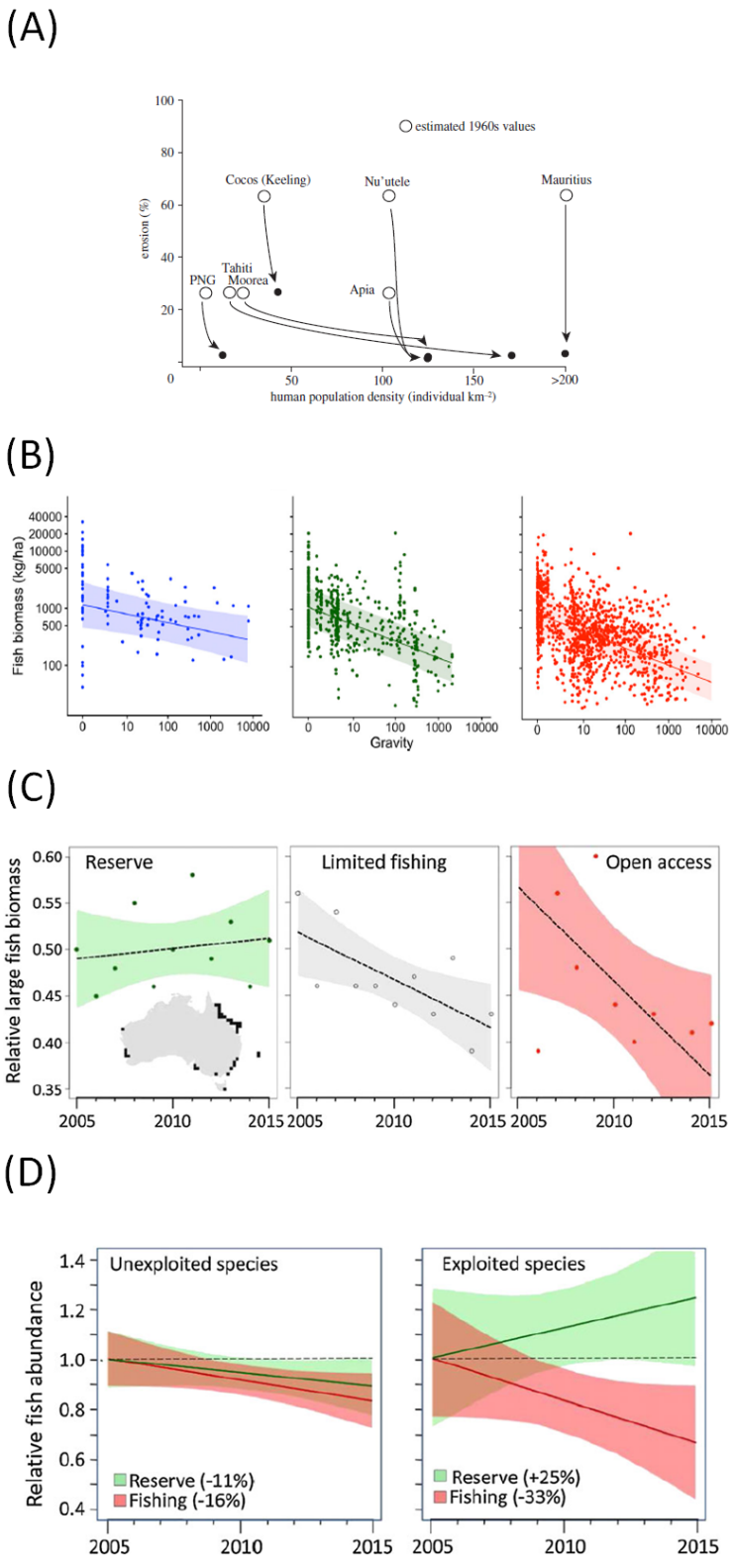


FIGURE 1.4 – Impact de l'homme sur les communautés de poissons.
De nombreuse études démontrent la relation entre l'augmentation de la population humaine (représentée par la "gravité" sur (B), la gravité prenant en compte la taille de la population humaine et sa distance au récif) et la diminution de certaines fonctions écologiques (comme la bio-érosion, effectuée par les poissons perroquets (A) [Bellwood et al., 2011]) ou de la biomasse ((B) [Cinner et al., 2018], en zone protégée à gauche, en zone de pêche restreinte au centre, en zone de pêche libre à droite)). D'autres études démontrent l'impact des pressions de pêche au cours du temps, en particulier sur les gros individus (C) [Edgar et al., 2018], montrent les tendances sur les individus >20 cm, et en particulier sur les espèces exploitées (D) (la tendance général étant montrée en rouge).

L'ensemble de ces activités humaines entraînent une chute importante du nombre d'espèces de coraux et de poissons coralliens, amenant à la disparition de certaines fonctions clefs nécessaires à la résilience et à la survie des récifs [D'agata et al., 2014]. Récemment, les perturbations subies par les écosystèmes marins et en particulier les écosystèmes récifaux s'accélèrent [Butchart et al., 2010] [Teichert et al., 2017] [Hoegh-Guldberg et al., 2017] [Weijerman et al., 2018] [Hughes et al., 2018] [Sully et al., 2019] et appellent à un effort de surveillance sans précèdent [Veitch et al., 2012] [Hughes et al., 2017]. L'utilisation de protocoles d'observations des écosystèmes récifaux ponctuels (spatialement et temporellement) ne permet plus de suivre le rythme des perturbations qu'ils subissent. Il est aujourd'hui important d'adapter nos méthodes de suivi et d'analyses des écosystèmes pour étudier leurs dynamiques à haute fréquence temporelle et spatiale.

Ces communautés doivent être étudiées comme un ensemble d'espèces interagissant entre elles et avec leur environnement y compris l'habitat et les pressions humaines.

Finalement, la structure des communautés de poissons est aussi largement utilisée comme indicateur de l'état des écosystèmes coralliens dans leur globalité, que ce soit la biomasse [McClanahan, 2018] ou des espèces particulières comme les herbivores [Goatley et al., 2016].

1.3 Méthode de recensement des communautés de poissons

L'étude de la structure d'une communauté nécessite de recenser l'ensemble des individus qui en font partie. Il existe plusieurs méthodes de recensement utilisées pour analyser la dynamique de ces communautés de poissons : les méthodes destructives, les méthodes visuelles réalisées en plongée, et les méthodes visuelles assistées par caméras (le tableau 1.1 récapitule l'ensemble des méthodes décrites dans la suite du chapitre et leurs limites). Le recensement d'une communauté de poisson se divise en deux phases : la première consiste en l'acquisition de la donnée brutes, et la seconde est dédiée au comptage et à l'identification des poissons présents. Certains recensements effectuent les deux phases simultanément (comme les méthodes visuelles réalisées en plongée), et d'autres séparément (par exemple les méthodes destructives et les méthodes assistées par caméras).

Les méthodes destructives

Les premiers recensements des communautés de poissons furent réalisés via des méthodes destructives (ou "méthodes par extraction"). La pêche, en particulier par dragage ou chalutage, fait partie des méthodes les plus destructives [Engel and Kvitek, 1998], à la fois

TABLE 1.1 – Tableau récapitulatif des différentes méthodes de recensement des communautés sous marines, basé sur [Mallet and Pelletier, 2014].

Méthode	**Moyen principal** d'**acquisition**	**Destructif (Oui/Non)**	**Donnée brutes sauvegardées (Oui/Non)**	**Limite(s) principale(s) et biais**
UVC mobile (*Line transect*) **UVC fixe** (*Point count*)	Visuel (Humain)	Non	Non	Présence du plongeur. Limites humaines : -temps d'observation, profondeur, etc... Nombre de personnels qualifiés limité.
Extraction (Dragage, Chalutage) **Extraction** (Chimique)	Visuel (Humain, une fois l'extraction terminée)	Oui	Oui	Destruction des communautés et des habitats. Pas d'observations comportementale. Impossible à grande échelle.
RUV	Caméra vidéo	Non	Oui	Champs de vision restreint et immobile.
BRUV	Caméra vidéo	Non	Oui	Modification des comportements. Modification de la composition.
DOV	Caméra vidéo	Non	Oui	Présence du plongeur. Limite humaine : -profondeur de la plongée. -nombre de plongées par jour.
ROV	Caméra vidéo	Non	Oui	Affecte le comportement des espèces sous-marines. Limites liées à la maniabilité des ROVs
TOWV	Caméra vidéo	Non	Oui	Complexe en milieu récifale. Mobilité restreinte.

pour l'ensemble de la chaîne trophique [Jennings et al., 2001] [Board et al., 2002] [Poiner et al., 1998], mais aussi pour les habitats et pour le fond marin [Watson et al., 2006]. De plus, elle reste difficilement applicable sur les écosystèmes coralliens.

Les effets négatifs de telles méthodes sur la biodiversité marine à court et à long terme ainsi que sur le fond marin ont été discutés de nombreuses fois, que ce soit à des fins scientifiques [Jouffre et al., 2009] [Trenkel and Cotter, 2009] ou commerciales [Jones, 1992] [Kaiser et al., 1996] [Eigaard et al., 2015] [Kaiser et al., 2002] [Thrush and Dayton, 2002].

Une seconde méthode destructive est l'extraction par produit chimique. Cette pratique consiste à diffuser dans l'eau un produit létal ou anesthésiant pour ensuite collecter la communauté de poissons. Les deux molécules les plus communément utilisées sont la roténone et l'eugénol. La roténone est une molécule toxique et létale pour les espèces à sang froid qui est produite par certaines plantes tropicales (*Derris*, *Lonchocarpus* [Lecointe, 1936]). L'utilisation de la roténone est moins destructrice pour les habitats que les approches par pêche [Robertson and Smith-Vaniz, 2008], tout en permettant d'obtenir de meilleurs résultats en terme de biomasse et d'espèces observées par rapport au recensement visuel [Ackerman and Bellwood, 2000] [Dibble, 1991]. On peut aussi noter l'utilisation de produits anesthésiants (eugénol, méthanesulfonate de tricaïne, et 2-phenoxyethanol [Priborsky and Velisek, 2018]) non létaux selon la concentration utilisée [Fernandes et al., 2017].

Bien que les méthodes de recensements destructifs puissent procurer de bons résultats en terme d'estimation de la biodiversité d'un écosystème, en particulier pour les espèces cryptiques [1], elles présentent des faiblesses : elles ne permettent pas d'effectuer des analyses de comportements ; les méthodes par extractions chimiques se concentrent principalement sur les espèces de poissons (la ou les analyses visuels permettent aussi de traiter l'environnement) ; et finalement, leurs effets négatifs à court, moyen, et long termes sur les écosystèmes marins ne leur permettent pas d'être utilisées à grande échelle. Il est aussi impossible d'utiliser ces protocoles pour réaliser le suivi d'une région ou d'une communauté dans le temps puisque celle ci étant détruite par le protocole.

Les méthodes visuelles en plongée

Pour pallier aux inconvénients et à l'éthique des méthodes destructives, le protocole le plus largement utilisé depuis plus de 70 ans [Brock, 1954] est le recensement visuel des communautées de poissons (*Underwater Visual Census*, UVC), effectué par un ou plusieurs plongeurs. Aujourd'hui, les deux procédures les plus communes pour pratiquer un UVC sont le transect linéaire (Fig. 1.5) (*line transect*) mobile, et le comptage statique (*static point count*, SPC). Lors d'un transect linéaire, plusieurs plongeurs se déplacent le long d'une ligne. Il observent et notent le nombre d'individus présent, leur espèce, leur taille, et

1. Cachées.

leur écentuellement leur comportement [Hill et al., 2005] à une certaine distance maximum de la ligne (entre 1 et 5 mètres, selon le protocole employé et la visibilité). Ce transect est souvent associé à une autre méthode appelée *belt transect*, qui consiste à se déplacer le long d'un transect linéaire, mais de définir sur celui-ci des quadrats précis, afin d'étudier le benthos [2]. Seule la zone du quadrat est analysée [Caldwell et al., 2016].

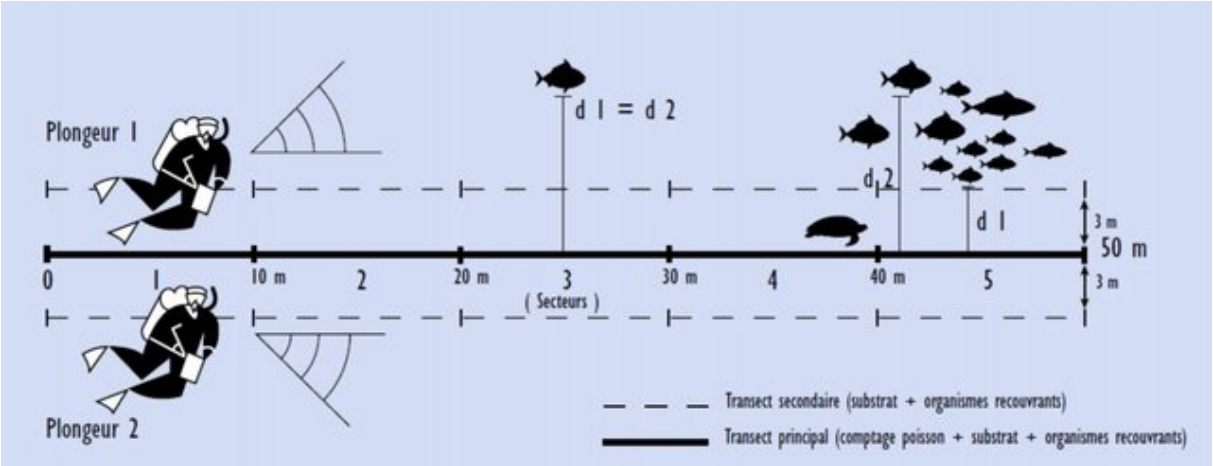


FIGURE 1.5 – Illustration d'un transect linéaire [Elise and Kulbicki, 2015]. 2 plongeurs se déplacent le long d'un transect de 50 mètres de long, et analyse la communauté de poissons, 3 mètres de chaque côté de la ligne de transect (pour une surface de $300m^2$). Selon la visibilité, l'étude peut être étendue à 5 mètres de chaque coté ($500\ m^2$).

Lors d'un comptage par comptage statique, un plongeur stationnaire va analyser la communauté de poissons dans un certain rayon autour de lui. Par rapport à la méthode transect, le SPC permet d'obtenir une meilleure approximation de la densité de poissons [Colvocoresses and Acosta, 2007], tout en demandant moins de temps au plongeur [Facon et al., 2016]. Cette méthode permet aussi d'observer les évolutions (changement d'espèces présentes) dans la communauté présente et d'évaluer les comportements (prédation). De plus, cette méthode permet à des plongeurs moins expérimentés d'obtenir de bons résultats en terme d'analyse des communautés, alors que les transects linaires demandent des experts confirmés [Facon et al., 2016].

Ces différentes approches de recensements visuels comportent cependant des faiblesses. Les études sont limitées par la physiologie du plongeur. En effet, les plongées sont limitées en durée, en profondeur, et en nombre par jour (2 maximum, pour une durée totale de 2 heures). De plus, chaque comptage sur transect peut être biaisé de manière non quantifiable (nombre d'individus observés, tailles) en fonction de l'expertise du plongeur (état de fatigue, entraînement) et de la visibilité, ce qui pose des problèmes de reproductibilité. Il est aussi

2. Organismes vivant sur le fond marin.

reconnu que le recensement visuel sous-estime la présence des espèces cryptiques (comme par exemple les *Tripterygiidae* ou *Gobiesocidae*) [Willis, 2001], nocturnes [Brock, 1982] ou méfiantes [Kulbicki, 1998].

Enfin, le plus gros inconvénient vient de l'introduction du plongeur dans l'écosystème marin, ce qui provoque une réaction (*diver effect*) d'attraction ou de fuite chez les poissons présents [Dickens et al., 2011] induisant, une fois de plus, un biais difficilement quantifiable.

Méthodes assistées par vidéo

Les approches de recensement par vidéo permettent de répondre à une demande et une nécessité d'intensifier les efforts d'échantillonnage sur les récifs. Elles sont développées depuis 1952 [Barnes, 1952] [Barnes, 1955]. Les méthodes les plus classiques sont les vidéos distantes sous-marines, ou *remote underwater video*, RUV. Il s'agit de systèmes de caméras classiquement posés en plongée ou depuis la surface sur un substrat sous-marin. On retrouve aussi des systèmes de caméras embarquées sur des *Dispositifs de concentration de poisson* (DCP) [Fonteneau et al., 2000] qui sont des systèmes flottants concentrant des communautés de poissons pélagiques [3] [Doray et al., 2007] [Merten et al., 2018]. Francour et al. [Francour et al., 1999] montrent que les RUVs sous estiment le nombre d'espèces présentes dans une communauté par rapport aux UVCs, cependant ils estiment que ce biais est principalement dû à l'aspect statique des systèmes vidéos par rapport aux UVC réalisés en transect. Les systèmes vidéos fixes présentent toutefois d'autres avantages. Tout d'abord, les données brutes d'observation sont enregistrées et peuvent être archivées. Cela permet donc :

1. D'effectuer les mêmes analyses par différentes personnes (reproductibilité/robustesse).
2. D'utiliser les données stockées pour faire des analyses non prévues lors des enregistrements (e.g. des vidéos réalisées dans un but de comptage de poissons peuvent être utilisées ultérieurement pour observer des comportements).
3. D'accumuler un important volume de données ponctuellement et de reporter l'effort de traitement humain, de le déplacer géographiquement ou de le décaler temporellement.

L'utilisation de caméras permet aussi de supprimer les effets de la présence du plongeur. Cela permet de dépasser les contraintes humaines (profondeur et temps sous l'eau), ce qui réduit les coûts et optimise les campagnes d'observation (en terme de coût/temps), pour collecter plus de données sur de plus grandes surfaces à des échelles temporelles plus longues. Cette accélération permet de répondre aux nouveaux enjeux de suivi des communautés de poissons à grande échelle et à haute fréquence.

3. Vivant en haute mer.

Les systèmes RUVs sont le plus souvent équipés d'une seule caméra fixe, mais peuvent aussi être équipés d'un mécanisme permettant à la caméra d'effectuer des rotations à 360 degrés (comme le système STAVIRO [Pelletier et al., 2012]) afin d'observer l'ensemble de l'environnement (à la manière d'un SPC), ou d'un système de stéréo-vision [Nedevschi et al., 2004] [Lazaros et al., 2008] [Mustafah et al., 2012], c'est-à-dire un ensemble de 2 caméras calibrées qui permettent par triangulation de récupérer des informations sur distance et la taille des organismes [Shortis and Abdo, 2016] [Harasti et al., 2017].

Les RUVs peuvent être équipés d'un appât. On parle alors de caméras appâtées, ou *baited remote underwater videos*, BRUVs. Cet appât est dans la majorité des cas composé de poissons gras (sardine, maquereau, etc.), et permet d'attirer des individus piscivores dans le champ de vision de la caméra. L'utilisation de BRUVs est privilégiée pour l'observation de prédateurs [Brooks et al., 2011] [Willis and Babcock, 2000] [Andradi-Brown et al., 2016] et en particulier de requins [Acuña-Marrero et al., 2018] [Kilfoil et al., 2017] [Juhel et al., 2018]. Les BRUVs ne biaisent pas le nombre d'individus non prédateurs recensés par rapport aux autres méthodes [Harvey et al., 2007]. La nature même des BRUVs, qui agrègent les individus autour de la caméra, induit quelques biais ou imprécisions notamment la surface ou le volume d'échantillonnage donc la zone géographique potentielle où l'appât est efficace [Dorman et al., 2012] et la modification de la composition de la communauté observée [Ghazilou et al., 2016] [Wraith et al., 2013]. Les études comparant les méthodes BRUVs et UVCs ont montré que les UVCs permettent d'observer une plus grande diversité d'espèces alors que les BRUVs permettent d'attirer certaines espèces non observées en UVCs, ce qui indique la complémentarité des deux approches [Lowry et al., 2012] [Colton and Swearer, 2010].

Ces systèmes de caméras peuvent aussi être mobiles, par exemple les plongeurs équipés de caméras (*diver operated video*, DOV), les systèmes tractés (*towed video*, TOWV) et les systèmes sous-marins pilotés à distance (*remote operated vehicle*, ROV). Les DOVs sont aussi soumis au *diver effect*, mais l'observation se faisant sur ordinateur plutôt qu'*in situ*, il est possible d'effectuer de plus longues distances et de minimiser le temps sous l'eau par rapport aux UVCs. De plus, grâce à la maniabilité du système par rapport aux RUVs et BRUVs, certains auteurs notent de meilleurs recensements en terme de nombre d'espèces observées [Goetze et al., 2015], en particulier pour les poissons dissimulés dans la matrice 3D (e.g. résidant dans les crevasses [Watson et al., 2005]). Les TOWVs [Assis et al., 2007] sont des systèmes de caméras tractés depuis un bateau, effectuant des transects allant de 30 mètres à 20 kilomètres [Mallet and Pelletier, 2014]. Ces systèmes sont principalement utilisés pour étudier les communautés et les habitats benthiques [Holmes et al., 2008] [Rooper and Zimmermann, 2007] [Foveau et al., 2017] [Underwood et al., 2018]. Les ROVs, utilisés à des fins scientifiques depuis 1996 [Sward et al., 2019] on pour but d'associer les avantages

des systèmes de caméras (e.g. dépasser les limites humaines en terme de temps d'analyse sous l'eau et de profondeur observée, tout en supprimant l'effet plongeur) et la maniabilité des observations par UVC/DOV. Il existe de nombreux types de véhicules sous-marins depuis les plus petits et maniables (3-20 kg), dédiés aux tâches d'observation en surface (e.g. $<$ 300 mètres) jusqu'aux plus grosses infrastructures (>5000kg), permettant d'accéder à des profondeurs supérieures à 2500 mètres [Huvenne et al., 2018]. Les principales limites de ces systèmes sont des problèmes de robotique liés au contrôle sous l'eau (courants, obstacles), à la gestion du câble permettant l'alimentation et/ou le transfert d'information vers la surface, et à la maniabilité en général. Finalement, certains auteurs notent que les DOVs induisent aussi une modification du comportement de certaines espèces par leur présence [Lorance and Trenkel, 2006] [Stoner et al., 2008] [Makwela et al., 2016]. Tous comme les RUVs, toutes ces approches vidéos (BRUV, DOV, TOWV et ROV) peuvent bénéficier de l'utilisation de caméras stéréos [Watson et al., 2005].

De nombreuses initiatives créées autour de la surveillance des récifs coralliens et de leurs écosystèmes utilisent des systèmes vidéos afin de collecter d'importants volumes de données. Les projets menés par *XL Catlin Seaview Survey* depuis 2012 ont permis d'accumuler des images sur 150km de récifs le long de la grande barrière de corail [González-Rivero et al., 2014], et ils sont aujourd'hui suivis de campagnes dans les Caraïbes, les Bermudes, ou encore en Asie du sud-est. L'initiative *Healthy Reefs for Healthy People*[4] effectue des opérations de surveillance et d'analyse sur les récifs coralliens mésoaméricains depuis 2003. *GlobalFinPrint*[5], au travers de nombreuses campagnes d'acquisition vidéos autour du globe, analyse la présence et le comportement des espèces de requins et de raies autour des récifs. D'autres organismes, tel que *Coral Reef Alliance*[6] ou *International Coral Reef Initiative*[7] sont centrés autour de la communication internationale afin de sensibiliser les communautés locales à l'entretien et à la compréhension des récifs. Ces projets et de nombreuses campagnes scientifiques, ont amené à une popularisation du recensement visuel assisté par caméra.

Cette popularisation de l'utilisation de caméras et d'appareils photos a impacté la surveillance de la biodiversité terrestre [Steenweg et al., 2017] et sous-marine [Mallet and Pelletier, 2014] [Bicknell et al., 2016] [Mohamed et al., 2018] [Kilfoil et al., 2017]. Ces appareils sont aussi utilisés pour la diffusion en temps réel (comme par exemple sur la côte est des États-Unis par le *National Oceanic and Atmospheric Administration*, ou sur le *Barkley canyon* par *Ocean Networks Canada* [Matabos et al., 2014]).

4. http ://www.healthyreefs.org/cms/
5. https ://globalfinprint.org/
6. https ://coral.org/
7. https ://www.icriforum.org/

Avec de tels outils, les données s'accumulent rapidement, et l'analyse de ces vidéos devient le principal goulot d'étranglement entre l'acquisition de données et l'extraction d'informations [Spampinato et al., 2008]. En effet, l'ensemble des méthodes de recensement présentées dans cette section ont en commun de nécessiter des experts humains *in fine*, que ce soit après extraction dans le cadre des méthodes destructives, au cours de la plongée dans le cadre des recensements sous-marins, ou sur un écran dans le cadre du recensement assisté par caméras. L'étape suivante pour accélérer l'étude des écosystèmes récifaux serait donc d'associer l'utilisation de caméras pour le recensement à des méthodes informatiques de traitement d'images pour effectuer la tâche d'analyse des enregistrements vidéos automatiquement.

Il est donc nécessaire de produire de nouvelles outils d'annotation automatiques de vidéos, pour pouvoir utiliser l'ensemble des enregistrements accumulées et ainsi rapidement analyser les communautés de poisson à grande échelle spatiale et temporelle. Dans le cadre de notre thèse, nous avons traité des images couleur issues des vidéos sous-marines. Les but des outils d'annotation développés dans la thèse est de localiser (grâce à des boîtes englobantes ou à un détourage) et nommer/classer des poissons dans des images sous-marines. Nous nous sommes donc intéressé aux algorithmes d'apprentissage automatique (*Machine Learning*, ML) et d'apprentissage profond (*Deep Learning*, DL). Ces méthodes sont très répandue afin d'effectuer les deux tâches principales nécessaires pour l'analyse automatique d'images : détecter les objets dans les images et vidéos, et/ou labeliser et classer des objets (dans notre cas, identifier des individus à un certain niveau taxonomique, généralement l'espèce).

1.4 Apprentissage automatique et réseaux de neurones

L'apprentissage automatique (ou *Machine Learning*, ML) est le nom donné à l'ensemble des méthodes informatiques permettant à une machine d'apprendre à effectuer une tâche [Nasrabadi, 2007]. Cette tâche peut être une action de classification de sons [Maglogiannis et al., 2009] ou d'images [Millán et al., 2018], un déplacement physique [Pratihar et al., 1999], une action de communication [Simeone, 2018]... L'ensemble de ces méthode ont pour but de créer un modèle, c'est a dire un ensemble de paramètre permettant d'effectuer une tâche.

Lors des tâches de classification d'images, on utilise généralement 3 types de bases de données pour entraîner et évaluer les approches ML :

- La base d'entraînement (ou d'apprentissage), permet d'entraîner le modèle, c'est à dire permet à l'algorithme de modifier les paramètres du modèle pour l'adapter à

une tâche.

- La base de validation permet d'évaluer le modèle durant l'entraînement, et ainsi de pouvoir contrôler son efficacité et éventuellement intervenir (arrêter l'entraînement, modifier des paramètres manuellement).

- La base de test permet d'évaluer le modèle une fois l'apprentissage terminé. Ce test doit se rapprocher de conditions réelles d'utilisation, car il est la dernière étape avant l'application du modèle à des tâches réelles.

Pour la classification d'images, l'apprentissage est généralement supervisé, c'est à dire que l'on connaît à l'avance les différentes classes auxquelles appartiennent les objets que l'on va identifier (ici des espèces de poissons). L'apprentissage basé sur des caractéristiques va se dérouler en 2 temps. Dans un premier temps, chaque objet utilisé lors de l'apprentissage (des images dans le cas de la classification d'images), va être transformé en une paire de variables. Chaque paire est constitué d'un vecteur caractérisant l'objet (*features vector*) et d'un label qui caractérise la classe. Ces caractéristiques peuvent être choisies manuellement par l'humain (dans le cas de l'étude de poissons, différents traits peuvent être utilisés, comme la taille de l'œil, la distance entre les nageoires, ou encore leur texture [Spampinato et al., 2010]...) ou calculées automatiquement. Un certains nombre de vecteurs de description ont été définis, par exemple les descripteurs SIFT (*Scale-invariant feature transform*) [Matai et al., 2012] [Shiau et al., 2012], permettant de décrire une image grâce à des points d'intérêts, *shape context* (SC) [Rova et al., 2007], basé sur la description du contour des objets ou encore "*Histogram of Gradient*" (HOG) [Zhu et al., 2006], permettant lui aussi de décrire avec précision les contours et les formes caractéristiques d'un objet).

L'objectif de l'algorithme d'apprentissage est d'apprendre à associer un label aux vecteurs qui lui, sont associés [Nasrabadi, 2007], et à discriminer les objets d'une classe par rapport à ceux des autres classes. Une fois l'apprentissage terminé, le modèle obtenu peut être utilisé pour prédire la classe d'un objet non labelisée. Pour cela, le vecteur caractéristique de cet objet est calculé, puis le modèle fournit une décision (le plus souvent un score) permettant d'associer une classe à ce vecteur.

Un certain nombre d'approches par apprentissage automatique ont été utilisés lors des tâches de classification d'images de poisson, ainsi que de la localisation d'individus dans des images, parmi lesquels on peut citer les séparateurs à vastes marges (SVM), les arbres de décisions et forêts aléatoires, les méthodes par plus proches voisins, et les réseaux de neurones [Kotsiantis et al., 2007].

Les séparateurs à vaste marges (*support vecteur machine*, SVM) [Joachims, 1998] [Blanc et al., 2014] cherchent à séparer les différentes classes en maximisant la marge, c'est à dire la distance entre la frontière d'une classe et les plus proches individus des autres classes.

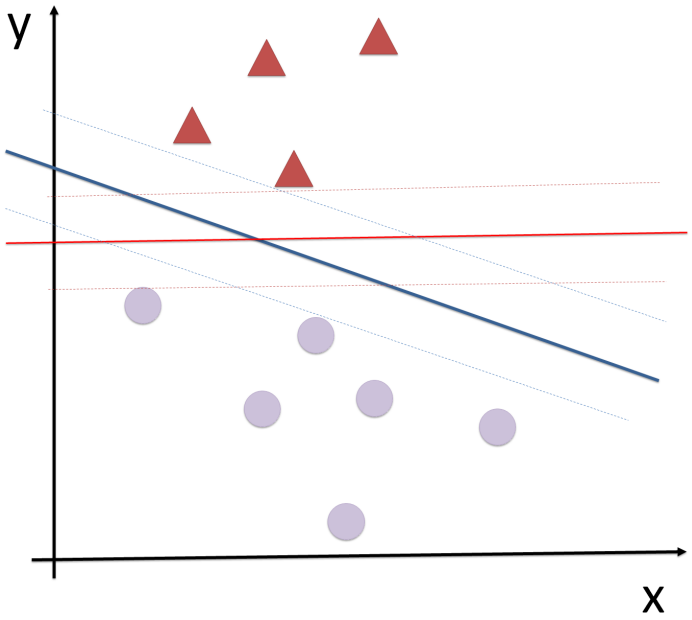


FIGURE 1.6 – Algorithme SVM.
Exemple de séparation d'un espace 2D grâce à un algorithme de type SVM. Les axes x et y représentent l'espace de classification (de vecteurs caractéristiques à 2 valeurs). Le SVM crée ensuite des séparations (lignes pleines rouges/bleu) afin de maximiser les marges (lignes interrompues), c'est à dire la distance entre les deux classes.

Cette méthode est très utilisé, et particulièrement performante pour les grands vecteurs.

Les arbres de décision [Safavian and Landgrebe, 1991] sont constitués d'un ensemble de nœud. Ces noeuds sont distribués sur plusieurs couches. Chaque nœud d'une couche C est relié aux nœuds de la couche suivante C+1 par des liaisons, appelées "branches". Chaque noeud effectue un test sur le vecteur d'entrée. Chaque branche correspond à un résultat du test effectué par le noeud. Finalement, chaque "feuille", ou noeud de la dernière couche, correspond à une sortie attendue par l'utilisateur.

Les forêts aléatoires d'arbres de décisions, ou *random forests* [Breiman, 2001] sont des ensemble constitués de plusieurs arbres de décision. Au lieu d'apprendre toutes les caractéristiques discriminantes fournis par l'utilisateur, chaque arbre apprend sur un sous-échantillon aléatoire de ces caractéristiques. Ainsi, chaque arbre se spécialise sur une tpache particulière, rendant l'ensemble plus efficace. Une fois l'apprentissage terminé, chaque arbre effectue une classification. Cette classification est ensuite considérée comme un vote, et la décision prise par l'algorithme est celle du vote majoritaire.

Les algorithmes basés sur la méthode du plus proche voisin ou *K-nearest neighbors*,

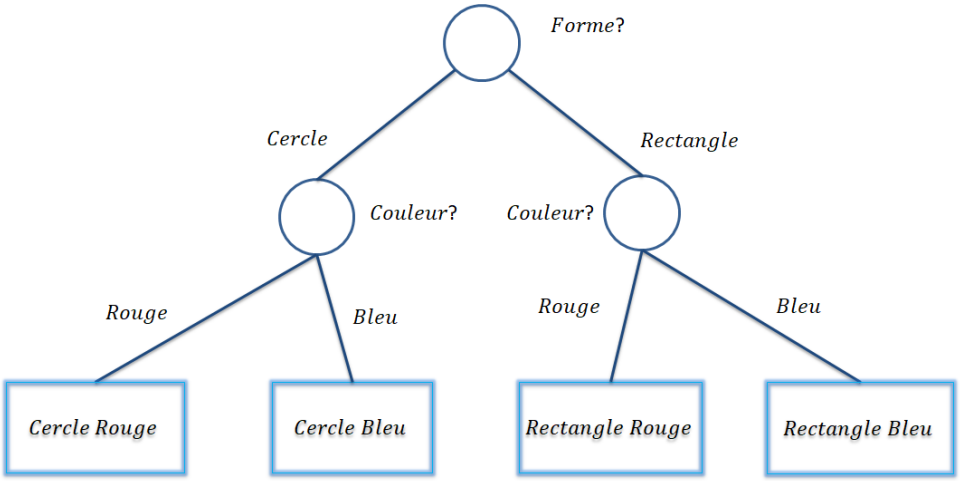


FIGURE 1.7 – Algorithme d'arbre de décision.
Chaque noeud vérifie la valeur d'un attribut de l'objet. Les feuilles correspondent à la classification faites par l'arbre grâce à la discrimination ainsi effectuée.

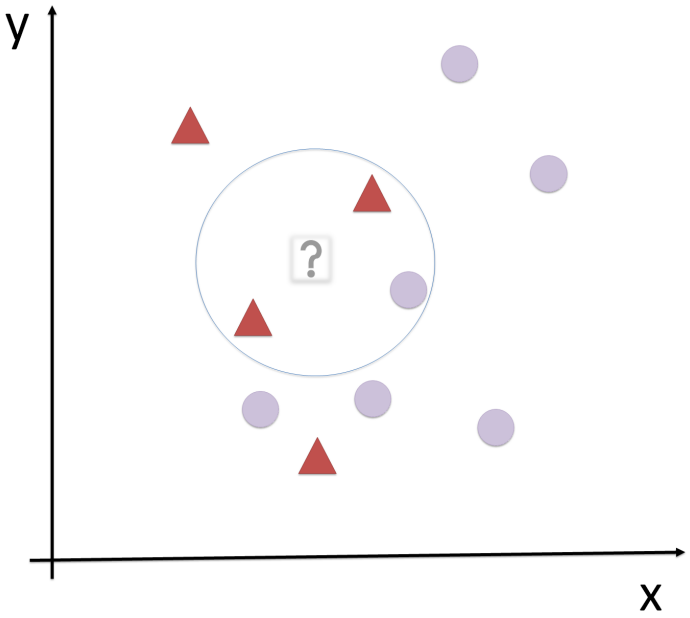


FIGURE 1.8 – Algorithme kNN.
Exemple d'attribution de classe à un nouvel élément noté **?** grâce à un algorithme kNN. Ici k=3. On considère donc les 3 plus proches voisins du nouvel élément, puis on lui attribue la classe la plus probable (triangle).

KNN [Keller et al., 1985] [Zhang and Zhou, 2005] sont aussi largement utilisés en classification. Pour chaque objet à classifier, son vecteur est comparé à celui des objets de la base d'entraînement (Fig. 1.8). On recherche dans l'espace dimensionnel un certain nombre (k) dont la distance entre les vecteurs est la plus petit, c'est-à-dire ses "voisins" grâce à une

distance (euclidienne par exemple). Dans le cas d'une classification, on attribue ensuite à chaque classe possible pour l'objet un score selon le nombre de ses voisins appartenant à ces classes. On peut aussi pondérer ce score selon la distance entre l'objet à classer et ses voisins.

Les réseaux neuronaux, ou *neural networks*, NN [Lettvin et al., 1959], [Hecht-Nielsen, 1992], inspirés directement du fonctionnement du cerveau humain et des neurones biologiques, sont aussi largement utilisés pour des tâches de classifications d'images [Wan, 1990], [Zhang, 2000] [Egmont-Petersen et al., 2002] et d'identification de poissons [Hernández-Serna and Jiménez-Segura, 2014]. Un NN est constitué de neurones. Chaque neurone effectue un calcul simple et retourne une valeur réelle en sortie. Ces neurones sont organisés en couches. Dans les réseaux classiques dits *feed forward*, chaque neurone d'une couche I est relié à l'ensemble des neurones de la couche suivante I+1 (Fig. 1.9), et les informations sont transformées de l'entrée vers la sortie du réseau. En dehors des couches de sortie et d'entrée on appelle les couches d'un réseau de neurones des couches cachées, ou couches entièrement connectées(*Fully connected layers*, FCs).

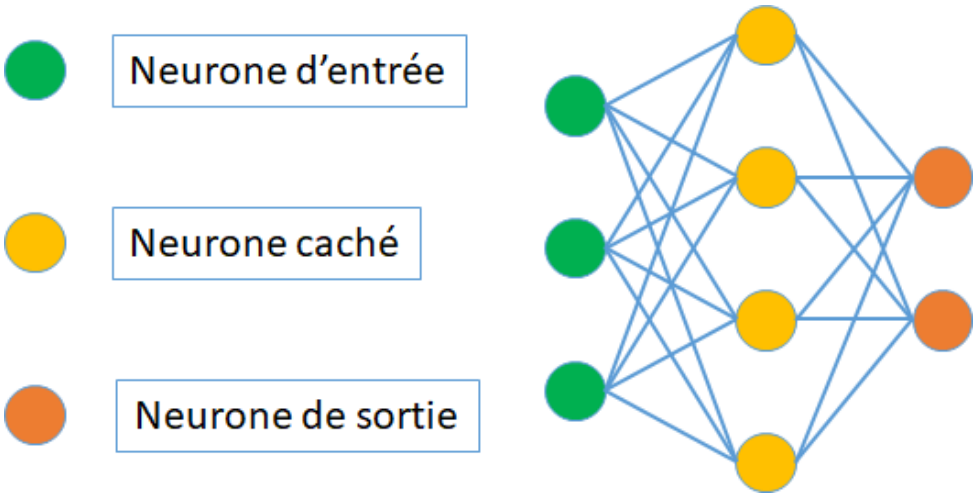


FIGURE 1.9 – Exemple de réseau de neurones simple, avec les neurones d'entrée recevant le signal d'entrée, les neurones cachées, et les neurones de sorties du réseau.

Ce signal est transmis par les connexions, une connexion étant une liaison entre 2 neurones. Pour transformer les informations d'entrée, un neurone va effectuer plusieurs opérations. Tout d'abord, le neurone va appliquer une fonction d'entrée $\alpha^{(n)}$. Pour ce faire, il effectue une pondération sur chaque connexion, afin d'amplifier ou de diminuer un signal particulier provenant d'un neurone de la couche précédente. On peut définir cette fonction d'entrée $\alpha^{(n)}$ pour un neurone n :

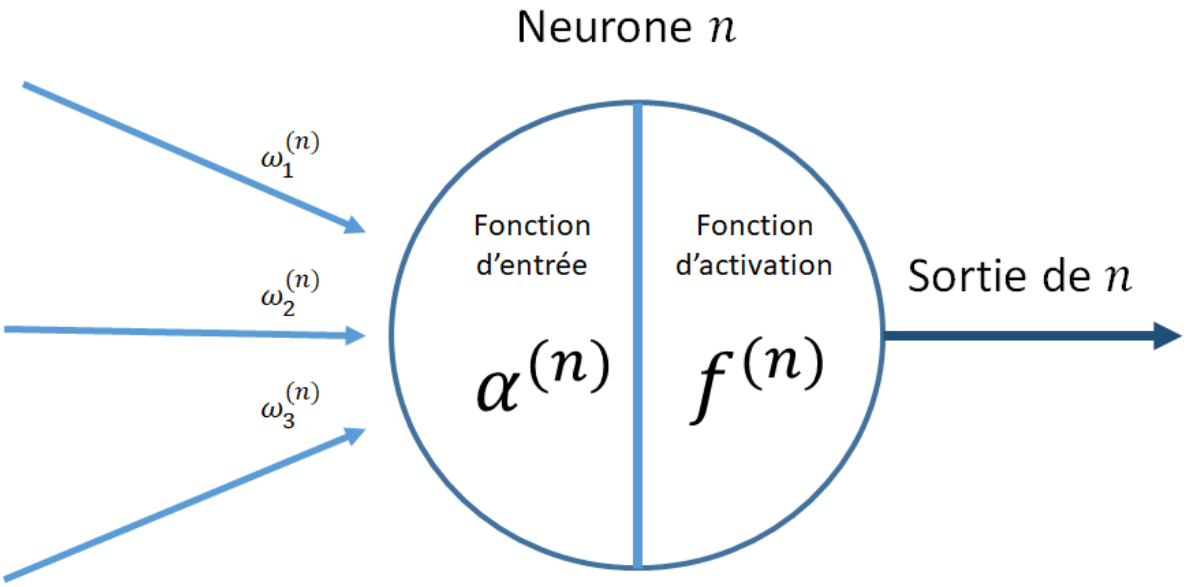


FIGURE 1.10 – Illustration d'un neurone.
On voit ici un neurone n, avec $w_i^{(n)}$ le poids affecté à une connexion, $\alpha^{(n)}$ la fonction d'entrée et $f^{(n)}$ la fonction d'activation, transformant les entrées avant de les transmettre à la couche suivante.

$$x_i^{(n)} \in \mathbb{R}$$
$$w_i^{(n)} \in \mathbb{R}$$
$$\mathbf{x}^{(n)} \in \mathbb{R}^c$$

$$\alpha^{(n)}(\mathbf{x}^{(n)}) = \sum_{i=1}^{c} w_i^{(n)} x_i^{(n)}$$

Avec $\mathbf{x}^{(n)}$, les valeurs provenant des neurones qui sont connectées au neurone n, c le nombre de connexions entre le neurone n et les neurones de la couche précédente, $x_i^{(n)}$ une valeur réelle transmise depuis n neurones d'une couche précédente grâce à une connexion, et $w_i^{(n)}$ le poids affecté à une connexion i du neurone n.

Le neurone applique ensuite une fonction d'activation $f^{(n)}$ qui transforme les valeurs réelles avant de les transférer vers les couches suivantes. Ces fonctions d'activations sont définies par l'utilisateur. Elles peuvent être simples (fonction identité) ou plus complexes (sigmoïde, exponentielle paramétrique...). On peut ensuite définir la sortie du neurone $\sigma^{(n)}$ ainsi :

$$\sigma^{(n)}(\mathbf{x}^{(n)}) = f^{(n)}(\alpha^{(n)}(\mathbf{x}^{(n)}))$$

Les paramètres de chaque neurones (les poids ω) seront modifiés pendant l'appren-

tissage (ou entraînement) afin d'améliorer les résultats de la classification. Cette étape d'apprentissage est obtenue par rétro-propagation, ou *back-propagation*. Durant l'entraînement, pour chacun des objets présents dans la base d'entraînement, le réseau reçoit en entrée un vecteur et un label k. Ensuite, la dernière couche du réseau va interpréter le vecteur caractéristique créé par le réseau, c'est à dire qu'elle va transformer ce vecteur caractéristique (obtenue grâce à l'activation du réseau) en scores de classification. Pour ce faire, la plupart des réseaux utilisent une fonction *softmax*, qui transforme un vecteur en vecteur de taille m, avec m égal au nombre de classes apprises par le réseau. La somme des valeurs de ce nouveau vecteur est égale à 1. Pour un objet, la sortie du réseau est donc une liste contenant l'ensemble des classes possibles (i.e. l'ensemble des classes présentes dans la base d'entraînement), et pour chaque classe un score. On compare ensuite le score idéal (100% de chance d'appartenir à la classe k), et les scores obtenus. Cette comparaison permet de calculer une erreur, qui sera ensuite retro-propagée dans le réseau. Les poids ω sont ensuite modifiés grâce à une optimisation de paramètres de type descente de gradient afin de rapprocher le score obtenu du score optimum. Aujourd'hui, la plupart des algorithmes de type "réseaux de neurones" utilisés pour des tâches de classification d'images reposent sur des méthodes d'apprentissage profond, ou *Deep Learning* (DL).

1.5 Apprentissage profond ou Deep Learning

Tout comme les réseaux neuronaux classiques, les réseaux neuronaux profonds (*Deep Neural Networks*, DNN) reposent sur une architecture composée de neurones et de connexions. Les DNNs les plus utilisés pour les tâches de classification d'images sont les réseaux convolutionnels (*convolutional neural networks*, CNNs) [Rawat and Wang, 2017]. Utilisés pour la classification d'images depuis 1990 [LeCun et al., 1990], les CNNs ont connu un regain de popularité en 2012 grâce aux possibilités offertes par un matériel plus puissant (e.g. calcul déporté sur cartes graphiques) et des algorithmes plus avancés [Krizhevsky et al., 2012]. Les CNNs tendent à être plus efficaces que les méthodes en deux temps (*Machine Learning*) vues précédemment [Joly et al., 2016], et à être de plus en plus utilisés pour les travaux de classification du vivant, notamment des poissons [Salman et al., 2016] [Marburg and Bigham, 2016] [Qin et al., 2016] [Joly et al., 2017]. En plus des couches effectuant la classification, les CNNs possèdent aussi des couches permettant d'extraire les informations des images, et de transformer la donnée d'entrée (image) en carte de caractéristiques (*feature map*, FM), l'équivalent des vecteurs caractéristiques utilisés en ML.

Cette tâche d'extraction de caractéristiques, qui s'effectue en amont de la classification lors des autres approches par ML, est donc effectuée en même temps que la classification, et profite ainsi du mécanisme d'apprentissage grâce à la rétro-propagation. Comme toutes les couches du réseau, les couches de convolution sont composées de neurones. La première

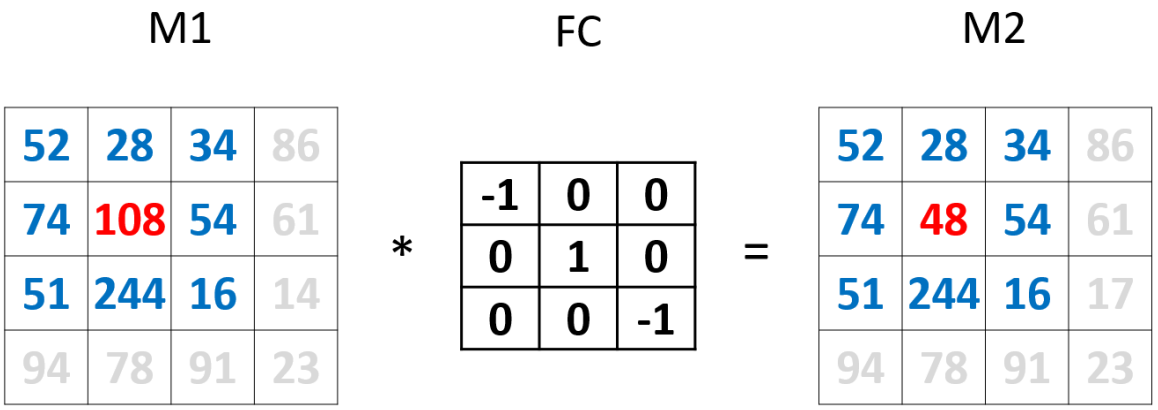


FIGURE 1.11 – Exemple d'application d'un filtre de convolution.
Le filtre de convolution *FC* modifie la valeur du pixel central (en rouge) de la matrice *M1*. Pour ce faire, elle pondère les pixels voisins avec les valeurs qui leur correspondent dans le filtre de convolution, puis les somme pour obtenir une combinaison linéaire. On obtient ainsi la nouvelle valeur du pixel modifiée
$Vm = (\mathbf{52*(-1)})+(28*0)+(34*0)+(74*0)+(\mathbf{108*1})+(54*0)+(51*0)+(244*0+(\mathbf{16*(-1)})$
dans la matrice *M2*. Le filtre est ensuite déplacé pour traiter le pixel suivant, jusqu'à ce que toutes les valeurs de pixels de l'image soient modifiées.

opération de ces couches est d'effectuer une convolution sur l'image d'entrée. Une convolution est l'application d'un noyau de convolution, aussi appelé matrice de convolution, à une image. L'application d'un filtre de convolution est une opération modifiant la valeur d'un pixel selon une combinaison linéaire de la valeur des pixels voisins. Ainsi, l'ensemble des poids de la combinaison linéaire définit le filtre de convolution (Fig. 1.11).

Elles peuvent être de tailles différentes (3×3 pixels, 5×5 pixels, 7×7pixels...) selon l'information à extraire [Szegedy et al., 2015]. Plus la taille de la matrice de convolution est importante, plus grand sera le voisinage considéré pour redéfinir la valeur d'un pixel. Les convolutions sont couramment utilisées en manipulation d'images. On peut par exemple les utiliser pour extraire des contours, diminuer ou accentuer le flou d'une image, etc (Fig. 1.12).

Au sein des CNNs, les valeurs des ces noyaux de convolution sont initialisées aléatoirement, et sont modifiées au cours de l'entraînement. Lors d'un traitement d'images classique, des filtres convolutifs peuvent être appliqués pour augmenter le contraste d'une image, pour le diminuer, ou encore pour accentuer les contours des objets. Dans le cas des CNNs, les convolutions doivent être capables d'intensifier les composantes permettant de discriminer les individus d'une classe par rapport aux individus des autres classes.

Dans les architectures CNN, chaque couche de convolution est suivie d'une couche de regroupement (*pooling layer*, PL). Ces PLs permettent d'effectuer 2 tâches : le filtrage des valeurs de l'image, ainsi qu'un sous-échantillonage des valeurs qui composent l'image. Ainsi, elles permettent de diminuer la dimension des cartes de caractéristiques de l'image.

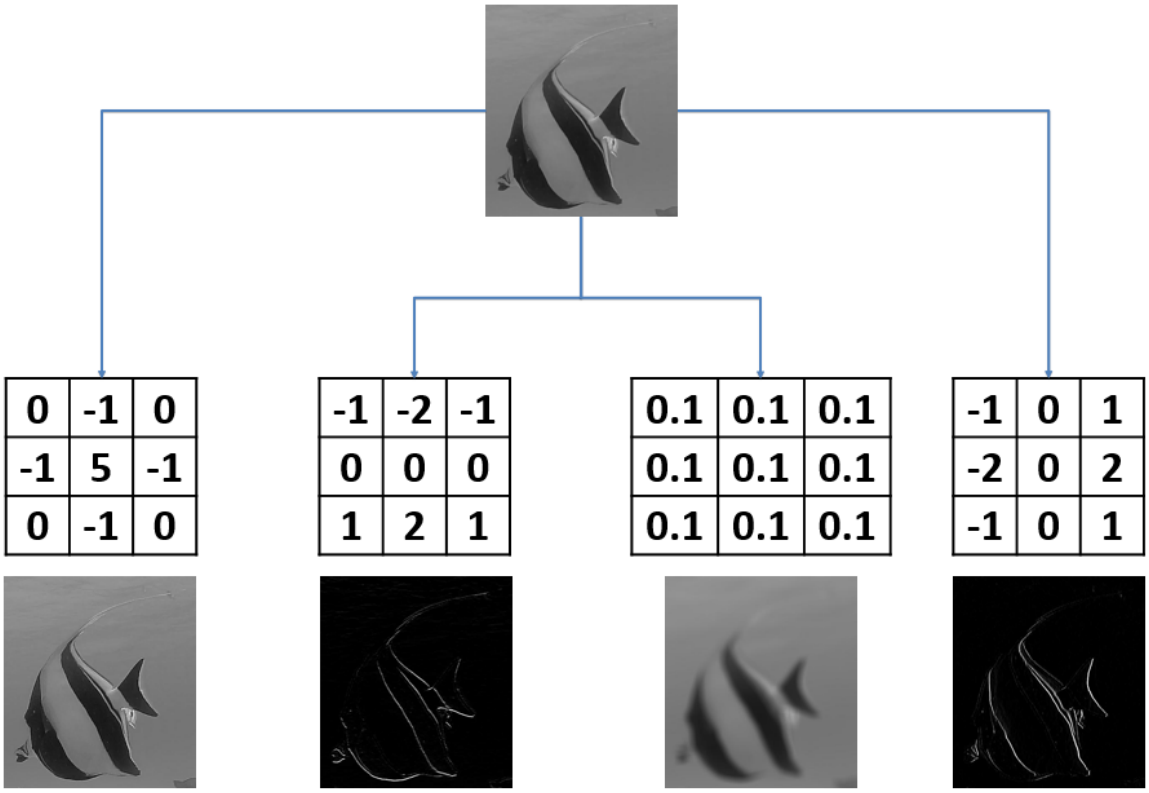


FIGURE 1.12 – 4 exemples d'application d'un filtre de convolution à une image.

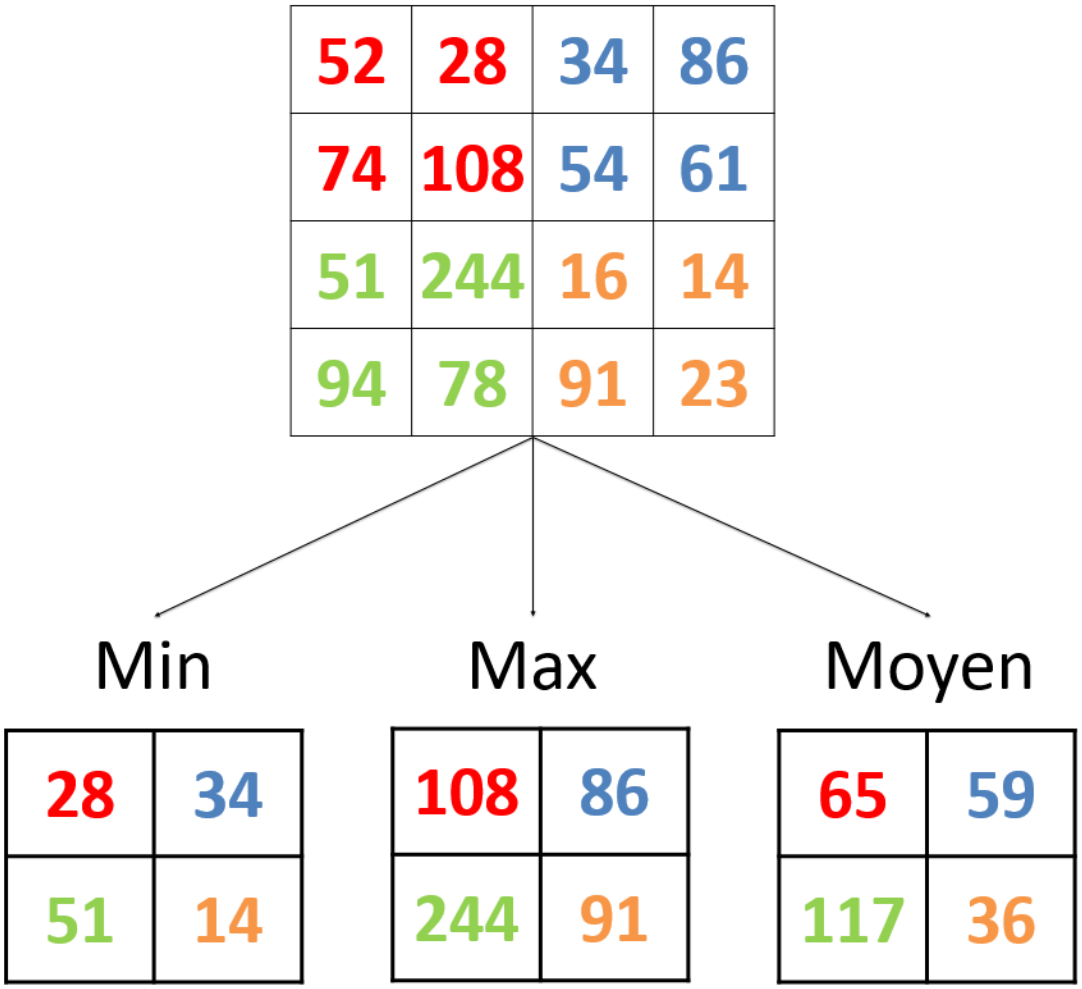


FIGURE 1.13 – Exemple d'application d'un *pooling* de taille 2*2, avec un déplacement de taille 2. Les trois cas (max, min, et moyen) sont représentés.

Les PLs utilisent 3 attributs :

- La taille du filtre, qui correspond aux nombres de valeurs utilisées pour le regroupement ainsi qu'à la forme du filtre (classiquement 2*2)
- Le type de regroupement (Fig. 1.13), qui serviront au filtrage des valeurs de l'image
- Le déplacement (*stride*), qui correspond à la distance en pixels entre une zone traitée par le filtre à la suivante, et qui serra utilisé pour le sous échantillonage

Le regroupement moyen (*average pooling*) était principalement utilisé lors des premières implémentations de CNNs [LeCun et al., 1998], mais tend à être remplacé par le regroupement max (*max pooling*) [Xu et al., 2015] [Boureau et al., 2010], qui permettent de rendre les résultats du *pooling* invariants aux distorsions et aux translations [LeCun et al., 2015]. Une fois ces couches de convolution et de regroupement passées, les cartes de caractéristiques (*feature maps*, FMs) sont transférées aux couches cachées, c'est-à-dire la sous

partie du réseau dédiée à la classification. Ces couches vont avoir le même comportement que celui décrit dans la définition du réseau de neurones *NN*. Comme vue précédemment, la dernière couche cachée va transférer la FM à une couche transformant le vecteur en score de classification. La couche la plus utilisée dans l'état de l'art applique une fonction *softmax* à la FM obtenue [LeCun et al., 2015] [Szegedy et al., 2015] [Xu et al., 2015]. Aujourd'hui, l'intérêt des écologues pour les méthodes de traitement automatique poussent à un effort de développement pour explorer les différentes possibilités offertes par les algorithmes CNN. Les méthodes d'identification et de localisation de poissons dans des vidéos sous-marines continuent donc de se développer [Christensen et al., 2018] [Wang et al., 2018], sous la forme d'utilisation d'architectures existantes [Redmon and Farhadi, 2017] [Liu et al., 2016b] ou de création d'architectures originales [Labao and Naval Jr, 2019]. L'étude des poissons coralliens va donc offrir un contexte (cadre, complexité, enjeux) très stimulant pour développer une nouvelle génération de CNNs et de statistiques de sorties.

1.6 Application de l'apprentissage profond à la localisation d'objets dans des images

Là ou les méthodes décrites précédemment effectuent une classification, c'est-à-dire attribuent une classe unique à une image, d'autres algorithmes, eux aussi basés sur les CNNs, permettent de détecter plusieurs objets d'intérêt dans une image (phase de détection/localisation), puis d'attribuer à chacun de ces objets une classe (phase d'identification). Les CNNs dédiés à la localisation et à l'identification sont divisés en deux catégories : les algorithmes qui effectuent la localisation et l'identification conjointement (*one-step algorithms* ou *one-shot*), et ceux qui l'effectuent en deux temps successifs (*two-step algorithms*).

Les architectures *one-step* (YOLO, SSD, MobileNet, RetinaNet) proposent des temps de calcul largement inférieurs aux architectures *two-step* (R-CNN, Faster R-CNN...) [Redmon et al., 2016] [Shafiee et al., 2017] [Liu et al., 2016b] mais présentent cependant de moins bon résultats. Lors de leur étude comparative, [Huang et al., 2017] montrent un temps de calcul inférieur d'un facteur 3.5 pour l'architecture SSD par rapport à l'architecture Faster R-CNN et une utilisation de mémoire inférieure d'un facteur 10, mais une précision moyenne (*mean Average Precision*, mAP), une métrique classique d'évaluation des performances d'un réseau de localisation, inférieure d'un facteur 1.5 (étude réalisée sur le jeu de donnée COCO, un *benchmark* utilisé pour les tâches de localisation et d'identification [Lin et al., 2014]). Lors d'une étude préliminaire, nous avons pu corroborer ces résultats (Fig. 1.14), dans laquelle nous avons constaté qu'un modèle créé à partir d'une architecture Faster R-CNN obtenait de meilleurs résultats que ceux créé grâce aux architectures SSD et RetinaNet avec la même entraînement. Notre étude ne portant pas sur des cas où la

puissance de calcul et le temps de traitement étaient limités (pas de traitement en temps réel), nous nous sommes exclusivement intéressés aux architectures en 2 temps qui seront détaillées ci dessous.

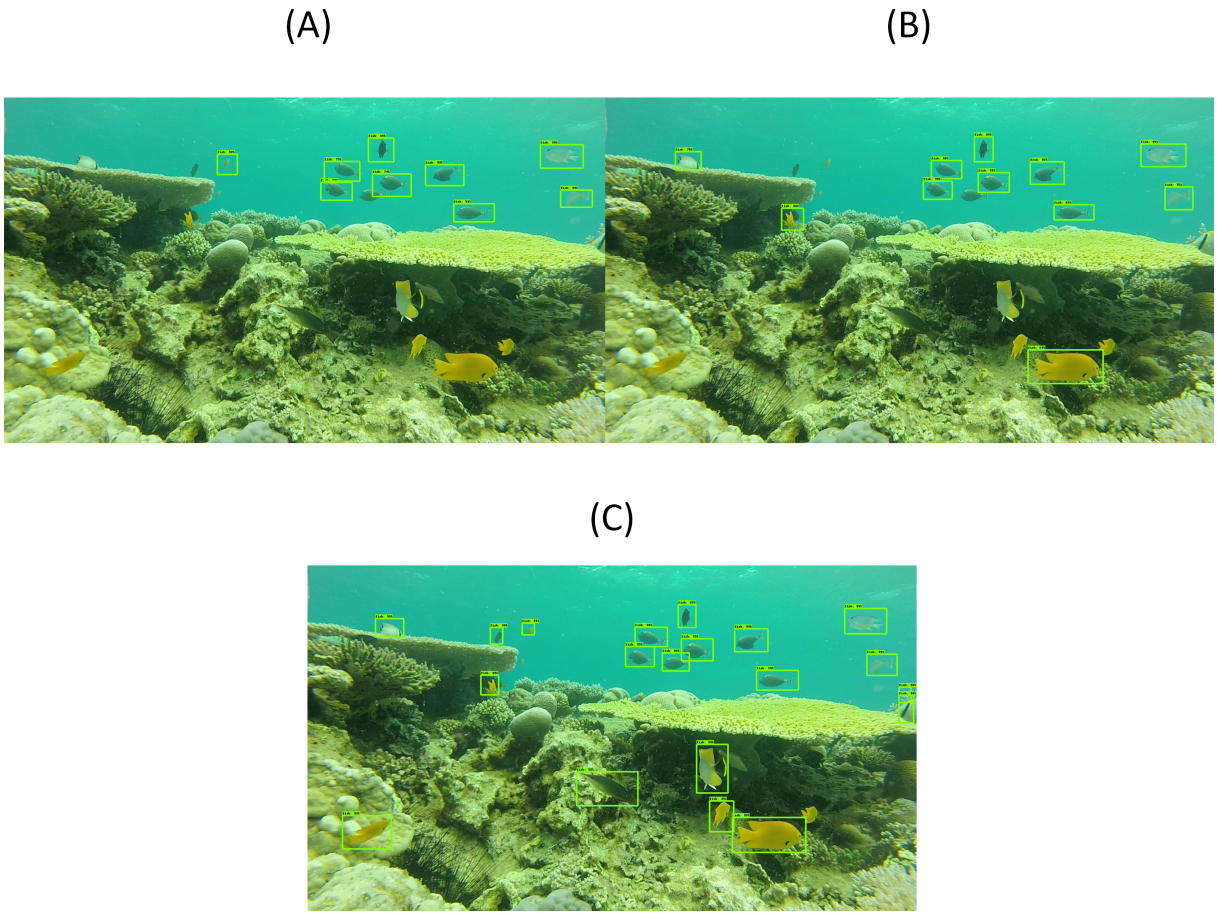


FIGURE 1.14 – Comparaison des résultats de 3 architecture de détection : (A) SSD [Liu et al., 2016b], (B) RetinaNet [Lin et al., 2017] ; (C) Faster R-CNN [Ren et al., 2015]. Les cadres verts correspondent aux boites englobantes prédites par les 3 réseaux.
Chaque réseau a reçu le même entraînement, effectué sur la même base. Nous avons donc donc conservé l'architecture Faster R-CNN, plus efficace, pour nos travaux.

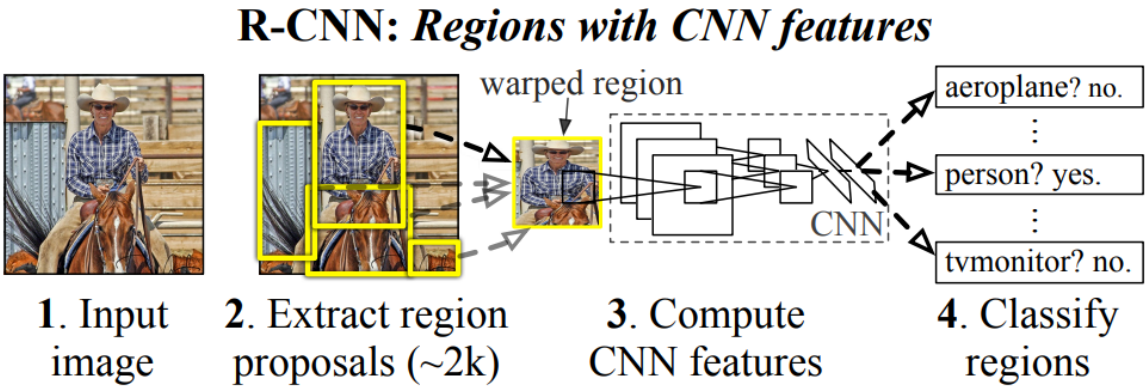


FIGURE 1.15 – Architecture du R-CNN.
En premier lieu l'algorithme de *selective search* définit les régions cohérentes dans l'image, à toutes les échelles de taille possibles. Ensuite, le CNN transforme et classe ces régions [Girshick et al., 2014].

Les architectures effectuant les deux tâches séparément (*Region based CNN*, R-CNN) [Girshick et al., 2014] présentent une première phase qui consiste à chercher un certain nombre de zones d'intérêt (*Region Proposal*), à toutes les tailles possibles, initialement grâce à une méthode de recherche sélective (*selective search*) [Uijlings et al., 2013]. Ce nombre de propositions de zones d'intérêt est définie *a priori* par l'utilisateur (e.g. 2000 régions pour l'architecture original du R-CNN) (Fig. 1.15).

L'algorithme de *selective search* commence par initialiser de petites régions homogènes dans l'image [Felzenszwalb and Huttenlocher, 2004]. Ensuite, un algorithme va permettre de fusionner ces régions. Pour cela un indice de similarité entre les régions voisines est calculé. Cet indice compare la similarité entre les couleurs (dans plusieurs domaines de représentation de couleurs, e.g. RGB, HVS, niveau de gris, rgI...), les textures, les tailles et la superposition des régions (i.e deux régions avec un fort taux de recouvrement auront plus de chance de fusionner que des régions distantes). A chaque fusion, les indices sont recalculés avec les nouvelles régions obtenues, jusqu'à ce que l'ensemble des régions soient fusionnées. Cette méthode permet ainsi de rechercher des objets de toutes les dimensions. Une fois les régions acquises, une seconde étape va se dérouler de la même manière qu'un CNN classique, et chaque région proposée va être identifiée en une classe (labelisée) . Finalement, l'algorithme renvoie une classe,ainsi que 4 valeurs correspondant aux dimensions de la région. Par la suite, des améliorations ont été apportées à cette méthode, dont l'inconvénient principal est le temps de calcul nécessaire à chaque itération.

[Girshick, 2015] propose donc de calculer le vecteur caractéristique de l'image entière, plutôt que de le calculer indépendemment sur chaque région (Fig. 1.16), ce qui permet un gain de temps considérable par rapport à la méthode précédente qui devait recalculer le vecteur caractéristique pour chaque objet d'intérêt.

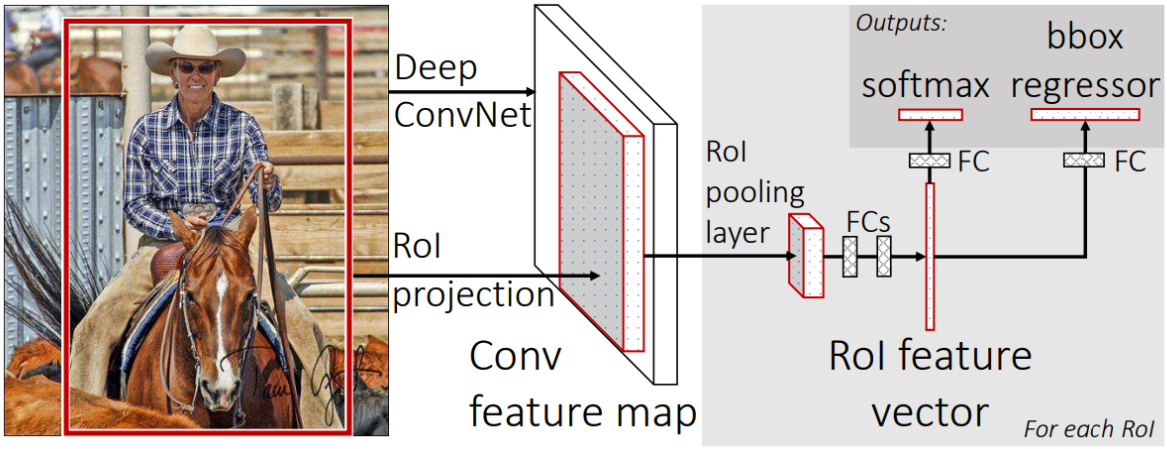


FIGURE 1.16 – Architecture du Fast R-CNN.
Le Fast R-CNN (FRCNN) calcule une seul carte de caractéristiques pour toute l'image (par rapport au RCNN calculant une carte par région), diminuant énormément le temps de calcul total pour analyser une image [Girshick, 2015].

Plus récemment, [Ren et al., 2015] propose de supprimer l'utilisation du *selective search*, et de la remplacer par un réseau de proposition de région (*Region Proposal Networks*, RPNs). Les phases d'apprentissage alternent alors entre la modification des paramètres du RPN, et la modification des paramètres d'identification. Le RPN est utilisé de la même manière que l'algorithme *selective search*, prenant en entrée une image et renvoyant en sortie une liste de boites englobantes, chacune associée à un score (*object score*). Le RPN étant composé de couches cachées, il profite alors de la rétro-propagation et de la fonction d'apprentissage, pouvant alors s'adapter spécifiquement aux objets de la base d'entraînement (Fig. 1.17). Les algorithmes basés sur les RPNs sont à ce jour les plus efficaces dans les tâches de détection d'objets, leur principale faiblesse étant le temps de calcul d'apprentissage qui reste important.

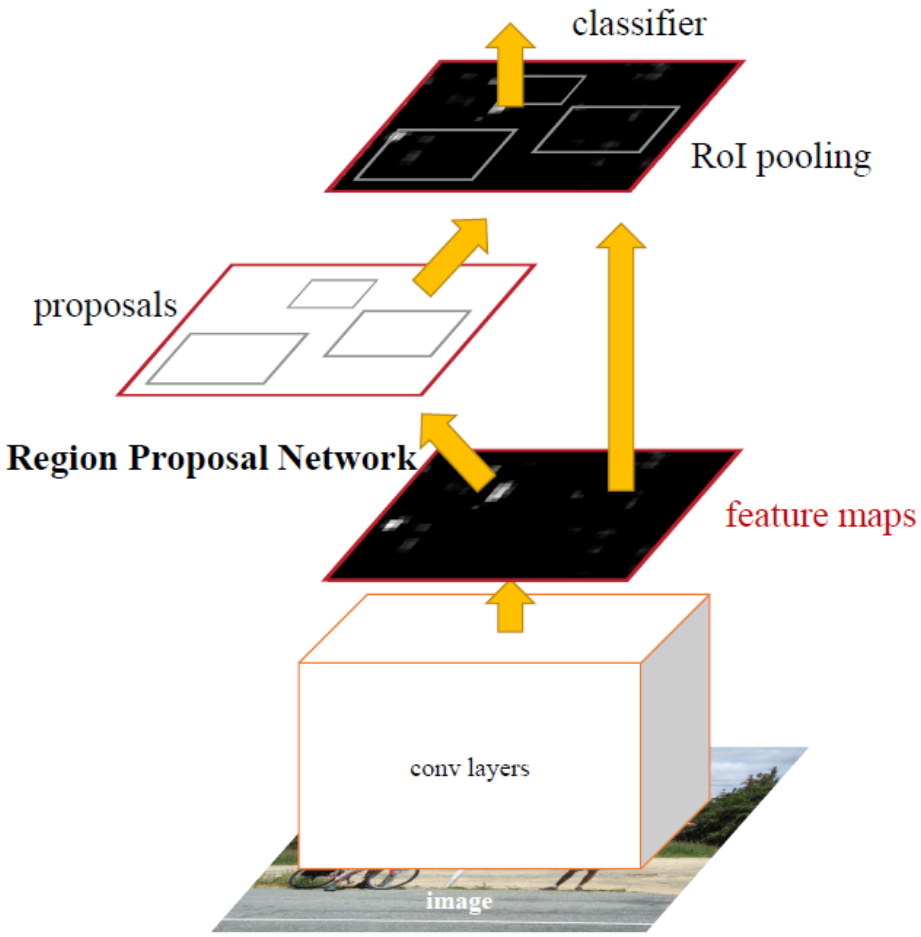


FIGURE 1.17 – Description de l'architecture du Faster R-CNN.
Le Faster R-CNN supprime l'algorithme de *selective search* et le remplace par un réseau *fully connected* choisissant les régions pertinentes, et qui s'adapte aux objets présents dans la base d'apprentissage [Ren et al., 2015]

.

1.7 Métriques d'évaluations

Lorsqu'une prédiction est faite par un algorithme de classification (un label est attribué par l'algorithme à une image ou à une région d'image non labelisée), deux résultats sont possibles, la prédiction peut être correcte ou incorrecte.

À l'échelle d'une classe, cela implique 4 résultats possibles (pour simplifier, nous parlerons au début simplement de la tâche classification d'une image contenant uniquement 1 objet appartenant à 1 classe (Fig. 1.18)) :

- Une image appartenant à la classe A est correctement labelisée par l'algorithme. On parle alors de vrai positif.
- Une image appartenant à la classe A est incorrectement labelisée en tant que classe B. On parle alors de faux négatif (pour la classe A).
- Une image appartenant à la classe B est incorrectement labelisée en tant que classe A. On parle alors de faux positif (pour la classe A).
- Une image n'appartenant pas à la classe A n'est pas labelisée par l'algorithme en tant que classe A. On parle alors de vrai négatif.

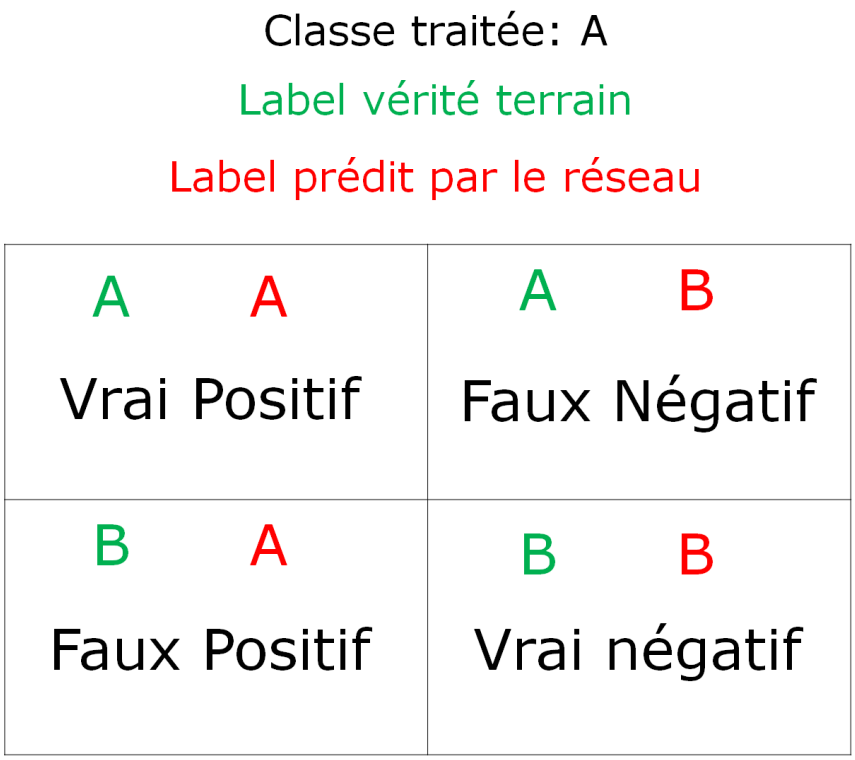


FIGURE 1.18 – Résumé des résultats de classification possible.

.

À partir de ces 4 types de résultats, deux types de métriques sont calculées pour chaque classe : le rappel, qui correspond au ratio entre le nombre d'individus par classe trouvés par l'algorithme par rapport au nombre d'individus par classe de la vérité terrain :

$$Rappel(A) = \frac{VP_A}{VP_A + FN_A}$$

avec VP_A le nombre de vrais positifs de la classe A et FN_A le nombre de faux négatifs de la classe A.

La précision correspond au pourcentage de labelisations correctes d'individus en tant que classe A soit :

$$Precision(A) = \frac{VP_A}{VP_A + FP_A}$$

avec VP_A le nombre de vrais positifs de la classe A et FP_A le nombre de faux positifs de la classe A.

Une troisième métrique, appelée F-mesure et combinant les deux premières, est aussi souvent utilisée ; elle est définie ainsi :

$$Fmesure(A) = 2.\frac{Rappel(A).Precision(A)}{Rappel(A) + Precision(A)}$$

Lorsque l'algorithme effectue la tâche de localisation en plus de la tâche d'identification, il est alors nécessaire d'évaluer la correspondance entre les boites englobantes définies par les humains et celles définies par l'algorithme. On utilise classiquement la métrique d'*intersection sur union* (*Intersection over Union*, IoU), qui calcule le ratio de recouvrement de la boite englobante définie par le modèle algorithmique et de la vérité terrain (Fig. 1.19).

La plupart des compétitions informatiques traitant des algorithmes de localisation estime qu'une boite englobante est correctement ajustée si l'$IoU > 0{,}5$ [Everingham et al., 2010] [Park et al., 2017].

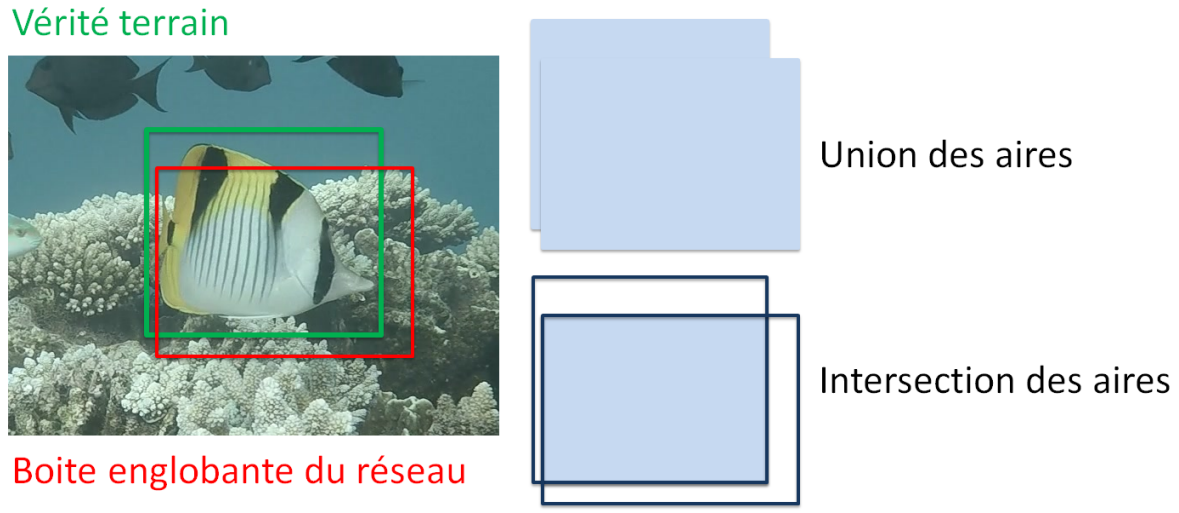


FIGURE 1.19 – Définition de la métrique de précision de la localisation.
On considère généralement qu'un IoU (*Intersection over Union*, ou intersection des aires sur union des aires) de plus de 0,5 indique que la prédiction est bien ajustée à la vérité terrain.

1.8 Problématiques de la thèse

Si le Deep Learning est de plus en plus utilisé dans l'analyse automatique d'images terrestres et aériennes en écologie [McGregor et al., 2015] [Alarcón-Nieto et al., 2018] [Christie et al., 2016], très peu de travaux ont été réalisés en milieu sous-marin [Moniruzzaman et al., 2017]. Le milieu marin ajoute de nombreuses contraintes à l'analyse d'images automatisée, comme la déformation de l'image sous l'eau ou la disparition d'une gamme de couleur (disparition du spectre rouge sous 5 mètres). Le milieu corallien rajoute des challenges aux tâches de localisation et d'identification de poissons en raison de la très grande diversité de cet habitat complexe (coraux, éponges, algues), du nombre important d'individus et d'espèces présents à l'écran simultanément, de la petite taille de nombreux individus (les espèces crypto-benthiques <10cm), et de l'occultation partielle de certains individus à l'image par l'habitat corallien ou d'autres poissons (Fig. 1.20).

De plus, un bon nombre de travaux traitant de la détection et de l'identification d'espèces récifales ont été réalisés sur les bases de données de Fish4Knowledge [8], composées de vidéos à la résolution de 320x240 et 480x320 pixels qui ne correspondent plus aux standards actuels (une GoPro réalise désormais des vidéos à la résolution de 1920x1080 pixels). Ces changement permettent d'augmenter le nombre de pixels disponibles à l'image, augmentant la complexité algorithmique du traitement, mais augmentant du même coup le nombre d'informations disponible pour entraîner les algorithmes profonds.

8. http ://homepages.inf.ed.ac.uk/rbf/Fish4Knowledge/

FIGURE 1.20 – Exemples d'environnements complexes dans les écosystèmes coralliens.

L'entraînement d'un algorithme de type CNN permettant de localiser puis d'identifier les poissons coralliens sur des vidéos aux standards actuels se heurte encore à de nombreuses limitations.

Tout d'abord l'entraînement des algorithmes CNNs est particulièrement exigeant en nombre d'images par classe. Il est donc nécessaire d'annoter une importante quantité de vidéos sous-marines afin d'accumuler un grand nombre d'images de chaque espèce (1000+) dans de nombreuses conditions. Afin d'améliorer les résultats des modèles, il faut aussi éviter le déséquilibre entre les classes (i.e. autant d'individus de chaque classe, qu'elles soient communes ou rares).

Il faut aussi être capable d'étudier la robustesse ou fiabilité des modèles créés à partir de cette base de données pour des cas concrets d'identification et localisation de poissons récifaux dans leur milieu naturel et de comparer les méthodes de *Maching Learning* existantes et leurs potentiels en écologie marine. Pour ce faire, il est nécessaire de comparer les performances de modèles profonds face aux autres approches de *machine learning*, mais aussi par rapport à des experts humains. De plus, la détection et le contrôle des erreurs produites par les modèles profond (rendant les résultats inutilisables pour obtenir des mesures écologiques) pourrait aussi rendre le *Deep Learning* applicable à des cas concrets.

Nous devons ensuite mettre en place des protocoles permettant d'automatiser totalement ou partiellement le traitement de vidéos sous-marines.

L'identification des individus, associée à des méthodes de détection robuste, pourrait rendre possible le suivi de l'évolution d'une communauté de poissons récifaux au cours du temps une fois que l'effort d'échantillonage sera suffisamment important pour un grand nombre d'espèces.

1.9 Objectifs et structure de la thèse

Les objectifs de nos travaux sont donc :

- D'annoter et d'extraire des images d'individus de nombreuses espèces depuis des vidéos sous-marines, afin de créer une base de données correspondant aux standards actuels de qualité, et contenant suffisamment d'information pour permettre d'entraîner des algorithmes de *Deep Learning*. Les missions de terrain, la construction de toutes nos bases de données, ainsi que leurs spécificités et leurs intérêts sont détaillées dans le chapitre 2.

- De comparer les résultats obtenus par des modèles issus de réseaux profonds par rapport à l'état de l'art. Ainsi, nous allons comparer l'approche d'identification des espèces de poissons par *Deep Learning* à d'autres méthodes de Machine Learning dans le chapitre 3.

- D'étudier et d'améliorer la création des bases de données utilisées pour l'entraînement et le test de réseaux profonds. En particulier, nous avons étudié l'impact de différents types d'enrichissement de données sur les performances d'un modèle de classification entrainé par *Deep Learning* pour identifier des espèces de poissons récifaux dans le chapitre 4. Nous y comparons aussi les performances de notre meilleurs réseau avec un panel d'experts humains entraînés à l'identification de poissons.

- D'étudier et de proposer des solutions afin d'améliorer les résultats proposés par les algorithme d'identification basés sur des méthodes d'apprentissage profond. Dans le but d'utiliser des algorithmes pour automatiser l'analyse de vidéos sous marines, il est important de développer des modèles performants, mais aussi d'être capable de détecter les erreurs commises par ces modèles. Ainsi, nous proposons dans le chapitre 5 une méthode pour contrôler les erreurs de classification de notre modèle, selon plusieurs scénarios de cas concrets nécessitant l'identification automatisée de poissons récifaux.

- D'étudier la transférabilité et la robustesse d'un modèle de détection de poissons en milieu récifal. Nous proposons dans le chapitre 6 d'étudier le comportement d'un modèle entraîné dans l'océan Indien pour une tâche de localisation de poissons dans d'autres contextes (Méditerranée, Caraïbes, Pacifique...).

1.10 Implémentation des calculs

Toutes les architectures profondes développées pendant la thèse ont été implémentées sur des infrastructures logicielles dédiés, en particulier DIGITS [9] pour nos premiers travaux (chapitre 3), puis TensorFlow [10] [Abadi et al., 2016] à partir du chapitre 4. L'apprentissage de ces algorithmes, en particulier les plus récents (ResNET, Faster R-CNN) a été réalisé sur des configurations d'ordinateur conçues spécifiquement pour le Deep Learning (128 Go de mémoire vive, carte graphique Tesla Quadro P6000...). Ces configurations permettent un temps d'apprentissage acceptable mais pouvant monter à 1 semaine pour réaliser un apprentissage traitant 120 fois un ensemble de 250.000 images de la base d'entraînement, tout en restant bien moins chères qu'un super-calculateur dédié ($\sim$10.000 euros la station contre $\sim$120.000 euros pour le calculateur GPU DGX-1, et $\sim$340.000 euros pour le calculateur DGX-2, proposés par NVIDIA). Cependant, une fois l'apprentissage effectué, n'importe quelle configuration récente standard est capable d'analyser (classer) rapidement une image par le réseau entrainé, rendant ces modèles CNN utilisables sans matériel spécifique.

9. https ://developer.nvidia.com/digits
10. https ://www.tensorflow.org/

Chapitre 2

Construction de bases de données pour entraîner et tester des algorithmes de *Deep Learning*. Acquisition de données, campagnes de terrain, et annotation des bases de données.

L'acquisition d'images pour la construction de bases de données a représenté une partie non négligeable de cette thèse. L'ensemble de nos travaux reposent sur des modèles créés par apprentissage, ce qui implique que la fiabilité et la robustesse du modèle reposent à la fois sur les caractéristiques du modèle (architecture, type d'apprentissage, etc) et sur les données d'entraînement. De plus, les bases de données de test doivent elles aussi refléter des cas d'applications réalistes aussi complets que possibles.

2.1 Cas d'étude principal : Mayotte

Durant la thèse, la majorité de nos expériences (Section 2.2) ont été réalisés à Mayotte, un département d'outre-mer Français situé au nord du canal du Mozambique, au sein de l'archipel des Comores (Voir Fig. 2.1).

La surface du lagon (1500km^2), sa surface récifale, sa double barrière de corail (externe et interne), ainsi que sa richesse spécifique (760 espèces de poissons dont 80% inféodées aux coraux [Wickel et al., 2014]), en font un cas d'étude particulièrement intéressant. En effet, les études de suivi de l'évolution des récifs peuvent être effectuées à travers un large panel

FIGURE 2.1 – Localisation de Mayotte.
La majorité de nos travaux (chapitre 3,4,5) ont été menés grâce à des vidéos réalisées à Mayotte, au Nord de Madagascar et à l'Est du canal du Mozambique (12°50' 35" sud, 4°08' 18" est)

de communautés de poissons, mais aussi d'habitats et de conditions (moment de la journée, météo). On peut ainsi vérifier la robustesse des méthodes employées à l'ensemble des variations décrites précédemment. Mayotte présente aussi une grande diversité au niveau de la faune présentes sur ses récifs avec respectivement 360 genres et 118 familles de poissons, ainsi que la présence de nombreuses espèces emblématiques (une vingtaine de mammifères marins, 5 espèces de tortues marines). De plus, l'île bénéficie d'une certaine attention scientifique, en terme de campagnes de suivis de sa biodiversité et de ses écosystèmes : *IFRECOR* (Initiative française pour les récifs coralliens) depuis 2000, *ORC* (Observatoire des récifs coralliens) depuis 1998, après un épisode de blanchiment traumatisant pour le récif, *Reef Check* depuis 2002. Mayotte profite aussi de suivis spécifiques de certaines zones d'intérêts particulières tel que le suivi de l'aire marine protégée (AMP) de la "passe en S", qui étudie l'intérêt de la protection de la zone sur sa communauté de poisson depuis 1995, ou le suivi de l'évolution des récifs coralliens de la réserve naturelle de l'îlot Mbouzi depuis 2010. La création du Parc marin, englobant l'ensemble du lagon de Mayotte, est une initiative représentative de l'envie de contrôler, comprendre et suivre l'impact des pressions humaines et naturelles exercées sur la biodiversité et les écosystèmes récifaux à grande échelle. Tous ces efforts représentent cependant des suivis ponctuels sur des zones restreintes, qui pourraient être remplacés par un suivi généralisé sur l'ensemble du lagon. Lors de notre étude, nous nous sommes intéressé aux espèces et aux familles les plus présentes sur les récifs de Mayotte (*Pomacentridae*, *Acanthuridae*, *Chaetodontidae*,*Lethrinidae*, *Balistidae*, *Monacanthidae*, *Serranidae*, *Labridae*).

(A)

(B)

FIGURE 2.2 – (A) Images de la base de données Fish4knowledge [Kavasidis et al., 2014], et (B) images provenant de notre base de données.

2.2 Aller plus loin que les bases de données existantes

Au départ de nos travaux, la seule base de données publique utilisée comme benchmark pour la localisation et l'identification de poissons était mise à disposition par *fish4knowledge*[1]. Les 112 téraoctets de données de cette base proviennent de captures vidéos de 4 sites de Taiwan enregistrées entre 2010 et 2015. *fish4knowledge* a en particulier mis a disposition un jeu de donné pour l'identification d'espèces de poissons, contenant 27370 annotations manuelles appartenant à 23 espèces de poissons, issues de ces vidéos [Boom et al., 2012a] [Boom et al., 2012b]. Cependant, pour des raisons de volume de données et de technologies, ces vidéos étaient limitées à des tailles de 320*240 pixels et 640*480 pixels, et hautement compressées, comme illustré en Fig. 2.2. Nous avons donc décidé de constituer notre propre base de données pour plusieurs raisons :

- Tout d'abord, pour travailler sur des vidéos HD, ce qui nous permet d'avoir d'avantage d'informations à traiter lors de nos analyses d'images (un nombre de pixels plus important (Fig. 2.3), et une meilleure qualité de représentation des poissons à l'image.
- Ensuite, pour maîtriser l'ensemble du pipeline d'analyse, depuis la capture vidéo jusqu'aux analyses statistiques. Ainsi, nous pouvons modifier chaque partie du processus afin d'en améliorer le résultat final.

Pour l'acquisition des vidéos, nous nous sommes orienté vers des caméras compactes de type "action caméra". Ces caméras sont largement utilisées pour réaliser des enregistrements

1. http ://groups.inf.ed.ac.uk/f4k/index.html

(A)

(B)

FIGURE 2.3 – Zoom sur l'impact de l'augmentation de la résolution sur l'aspect des individus.
L'augmentation de la résolution des vidéos nous permet d'exploiter d'avantages de détails présents sur les individus (texture, motifs, couleurs). (A) présente une région correspondant à 1/10ème d'une image extraite d'une vidéo du jeu de donnée Fish4Knowledge, (B) une région de même proportion issue d'une de nos vidéos.

sous-marins [Boada et al., 2015] [Letessier et al., 2015] [Pergent et al., 2017] [Sirjacobs et al., 2017], terrestres [McGregor et al., 2015] [Alarcón-Nieto et al., 2018]) ou aériens [Koh and Wich, 2012] [Christie et al., 2016]. Elles permettent d'enregistrer en Full HD (1920×1080 pixels) tout en étant étanches jusqu'à 40m de profondeur grâce à des caissons et offrant une bonne autonomie (1h30). Bien qu'il existe d'autres types de caméras sous-marines permettant d'enregistrer avec une meilleurs qualité [Ho, 2007] l'utilisation d'action caméra permet de nombreux déploiements à moindre coût.

Nous avons utilisé deux types de systèmes d'enregistrement pour récolter nos données. Les vidéos étant réalisés avec des appareils posés sur un substrat (coraux, fonds marins...), les 2 systèmes sont composés d'une ou de plusieurs caméras et d'un support.

- Le premier système est composé d'une ou de plusieurs caméras (GoPro hero 4 et GoPro hero 5) et d'un monopode ou d'un trépied lesté (Fig. 2.4 (A)). C'est un système de caméra fixe, RUV.

- Le deuxième système est composé d'une ou plusieurs caméras, d'un support, et d'un appât [Colton and Swearer, 2010] [Wraith et al., 2013] (Fig. 2.4 (B)).C'est un système de caméra fixe appâtée, BRUV.

Ces systèmes ont été utilisés pour acquérir des vidéos sur les sites de Mayotte, Madagascar, Moorea, Martinique, Méditerranée, et Mer Rouge. Ils ont été déposés entre 1m et 40m de profondeur, en apnée, en plongée en scaphandre autonome, ou depuis un bateau dans le cadre des systèmes appâtés.

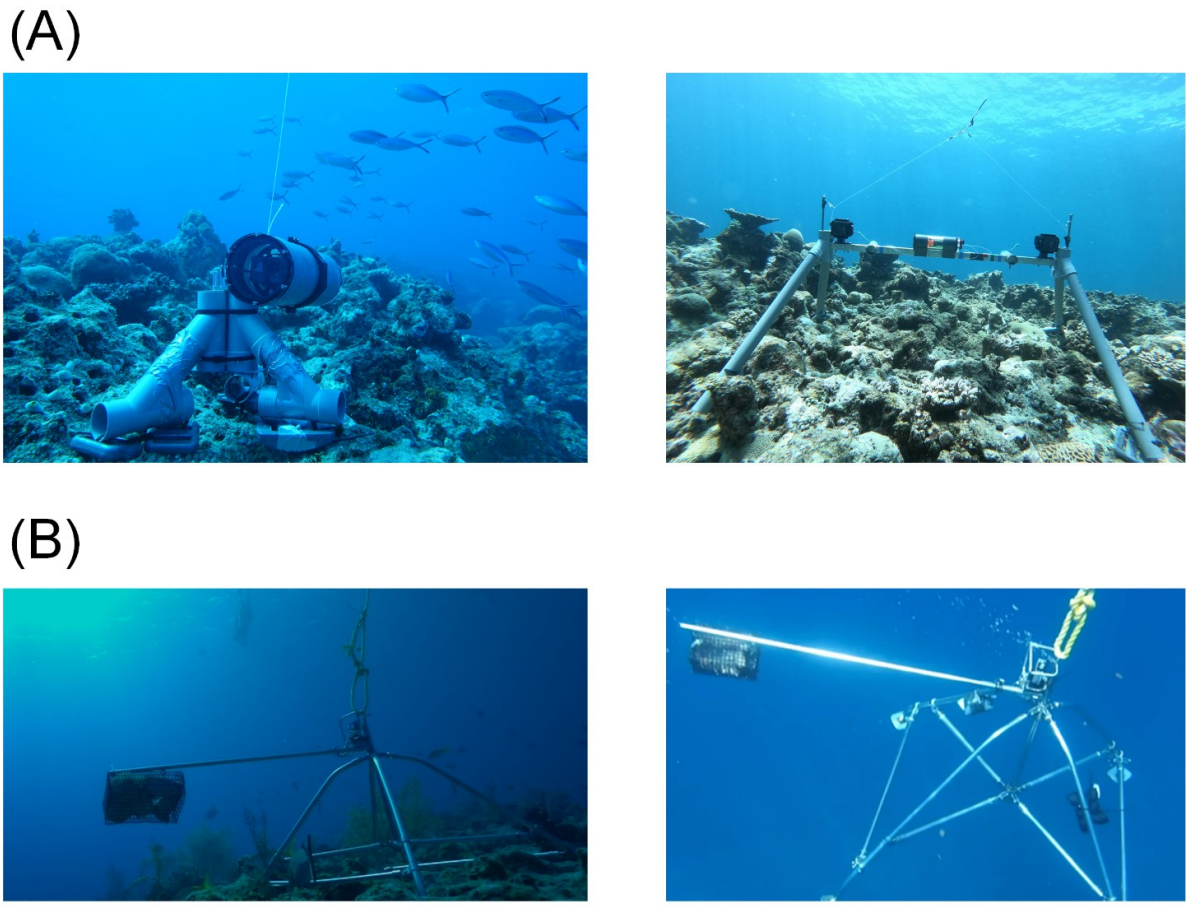


FIGURE 2.4 – Caméras et supports sans appâts (A) et avec appâts (B) tel qu'utilisés pour enregistrer nos données vidéos.

2.3 Campagnes et acquisitions de données vidéos

Les données utilisées pendant la thèse proviennent de plusieurs campagnes et missions décrites ci-dessous.

Mayotte, 2014-2016

Nous avons d'abord utilisé des vidéos enregistrées de manière opportuniste entre 2014 et 2016 par Sébastien Villéger et Thomas Claverie, sur différents sites de Mayotte. Ce jeu contient 102 enregistrements vidéos réalisés en RUV statique (pour une durée totale de 20h) toutes réalisées entre 1 et 30 mètre de profondeur, sans appâts. Ce sont les premières données que j'ai eu à disposition et qui ont été utilisées pour nos premiers travaux (voir chapitre 3).

Mayotte 2017

Deux campagnes ont été réalisées à Mayotte en 2017. La première a été réalisée du 9 au 12 Octobre 2017 par Thomas Claverie et moi. Nous avons posé des caméras sur récif frangeant de la zone nord de la Pointe de Bouéni (voir Fig. 2.5). Nous avons effectué une pose de caméra RUV entre 1 et 3 mètres de profondeur sur 7 points séparés de 200 mètres. Nous avons alterné les vidéos « courtes » d'une heure trente et les vidéos « longues » de 4 heures. Pour chaque point GPS, nous avons tourné 3 vidéos courtes ou 1 vidéo « longue » par jour, pour un total de 12 vidéos courtes et 3 vidéos longues par jour. Les vidéos longues permettent d'évaluer les changements de la communauté de poissons pendant une longue période sans perturbations extérieures dues au plongeur, et les vidéos courtes permettent de changer l'angle ou la direction de la caméra, et d'enregistrer ainsi plusieurs séquences différentes pour un même point GPS. Au total, nous avons récolté 41 vidéos le long du récif frangeant de Bouéni, pour une durée totale de 87 heures.

La seconde campagne a été réalisé du 13 au 18 Octobre 2017. Elle a été mené par Conrad Speed, Mark Chinkin, Phillip John Mcdowall et Thomas Claverie, dans le cadre de l'initiative Global FinPrint [2]. Cette initiative, lancé en 2015, a pour but d'étudier la présence, le déplacement et le comportement des espèces de requins et de raies. Cette campagne nous a permis de récolter 125 heures de vidéo sur le récif extérieur de Mayotte, étalés entre 124 points GPS (Fig.2.5), entre 10 et 40 mètres de profondeur. Les systèmes étaient cette fois ci équipés d'appâts (BRUVs simples caméras et stéréo), permettant d'attirer différentes espèces par rapport à celles présentes dans les jeux de données précédents. En effet, la campagne visant principalement à recenser les espèces de requins et de raies, les protocoles d'acquisition se font grâce à des caméras appâtées afin d'attirer et d'agréger le maximum d'individus [Brooks et al., 2011] [Goetze and Fullwood, 2013].

2. https ://globalfinprint.org/

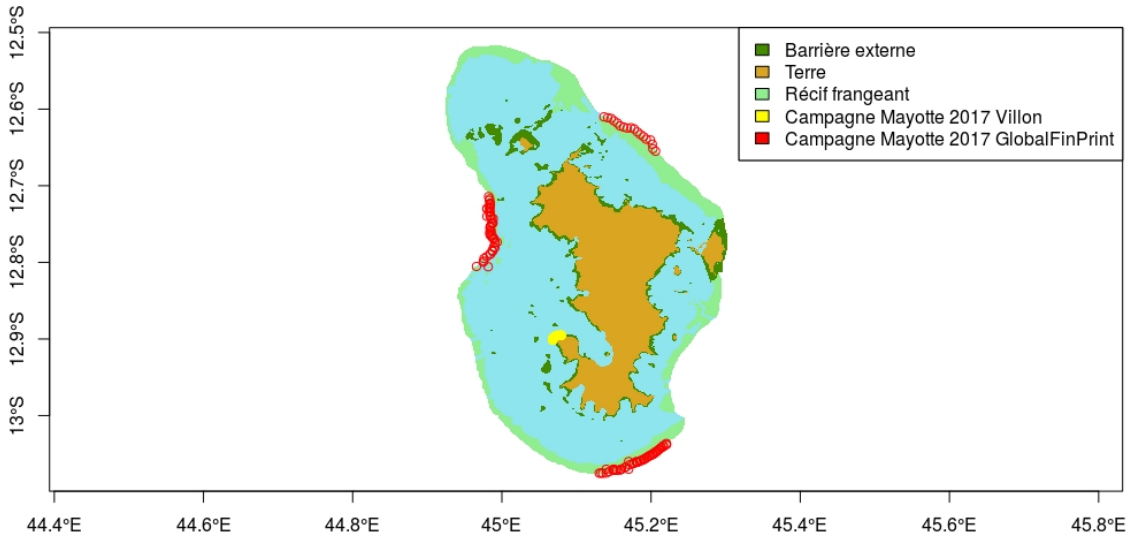


FIGURE 2.5 – Visualisation de Mayotte et de ses récifs, et localisation des 2 campagnes réalisées en 2017

Océan Pacifique, Océan Ouest Indien, Mer des Caraïbes, Mer Rouge et mer Méditerrannée

Afin de tester nos méthodes dans différents contextes, nous avons aussi récolté des données dans d'autres régions de l'océan Ouest Indien (Europa), mais aussi dans l'océan Pacifique (Moorea), dans la mer des caraïbes (Martinique), en mer Méditerrané (Crête, Israël) et en Mer Rouge (Israël). Ces données nous ont permis de créer des jeux de tests extrêmement différent des jeux d'entraînement, en terme d'environnement, d'espèces/familles présentes, de turbidité, de conditions climatiques...

2.4 Création d'un outil pour constituer nos bases de données d'images

L'annotation manuelle des poissons dans les vidéos collectées grâce aux différentes campagnes en mer est le principal goulot d'étranglement en terme de qualité et de quantité de travail à fournir pour l'entraînement et la validation des algorithmes de Deep Learning, en particulier des réseaux de neurones convolutifs (CNNs). Le but de nos CNNs est soit : 1) de localiser et d'identifier des poissons dans des images sous marine, ou 2) d'identifier des poissons une fois qu'ils ont été sélectionnés à l'aide de boites englobantes (*bounding boxes*) par un humain. Dans les deux cas, l'algorithme a besoin d'une base d'entraînement, c'est à dire de nombreuses images d'individus appartenant aux différentes classes qui vont être reconnues par le modèle. Ces classes sont généralement les espèces, mais peuvent être

moins précises (famille, genre) ou plus précise (sexe, âge), si le dimorphisme le permet. Ces images doivent être sélectionnées et annotées manuellement par des experts grâce à un logiciel d'annotation.

Logiciels d'annotation d'images sous-marines existants

Les logiciels d'annotation sous-marine (Marine image annotation software, MIAS) permettent d'obtenir deux types principaux d'annotation : Le premier est l'annotation grâce à des marqueurs (points) dans la vidéo. Ces marqueurs fournissent des informations sémantiques (nombre d'individus, d'espèces, tailles, etc), qui seront ensuite utilisés lors d'analyses plus poussées (analyses des communautés, comptage d'individus...). ClickPoint [Gerum et al., 2017] et VIDLIB [Marcon et al., 2015] par exemples sont conçus pour afficher et annoter des séries temporelles, et EventMeasure [3], ou VidSync [4], permettent à leurs utilisateurs d'annoter des vidéos stéréos. Le deuxième type d'annotation permet de construire des bases de données d'images utilisées pour entraîner les algorithmes de Deep Learning. Pour ce faire, l'annotation consiste à entourer les objets d'intérêts (Object Of Interest, OOI) grâce à une boite englobante rectangulaire ou polygonale. Les objets d'intérêts seront ensuite extraits de l'image principale, et puis stockés. Nous appellerons ces images spécifiques contenant 1 et 1 seul objet d'interet "vignettes". Certains outils grand public existent (Dataturks par exemple [5]), mais leur architecture globale ne correspond pas aux attentes demandées pour un travail scientifique en terme de métadonnées, de traçabilité des données, ou de structuration des données. De nombreux outils ont été développés spécifiquement pour annoter des images ou des vidéos sous-marines depuis les années 2000 [Gomes-Pereira et al., 2016].

Pourquoi développer notre propre application d'annotation

Parmi ces outils, très peu sont gratuits et collaboratifs. De plus, nous avions aussi besoin de faire évoluer l'application selon nos besoins, soit en adaptant le code d'une application existante s'il est disponible, soit en développant notre propre application. Aucune application existante ne correspondant à ces trois attentes (gratuit, open source, et collaboratif), nous avons donc décidé de créer notre propre outil. J'ai travaillé pendant la thèse en étroite collaboration avec Clément Desgenetez que j'ai tout d'abord co-encadré pendant son stage de deuxième année d'IUT, puis ensuite pendant 2 ans, pour créer et optimiser cette application, à la fois pour faciliter l'import/export de données, mais surtout pour optimiser la qualité et le rendement des annotations, c'est à dire rendre cette tâche aussi confortable que possible pour les experts, tout en facilitant la traçabilité des données. Finalement, nous avons opté pour le développement d'une application Web afin

3. https ://www.seagis.com.au
4. http ://www.vidsync.org/HomePage
5. https ://dataturks.com/

d'être indépendant de tout système d'exploitation, de centraliser les données, et de pouvoir travailler de manière collaborative et simultanée sur les mêmes données.

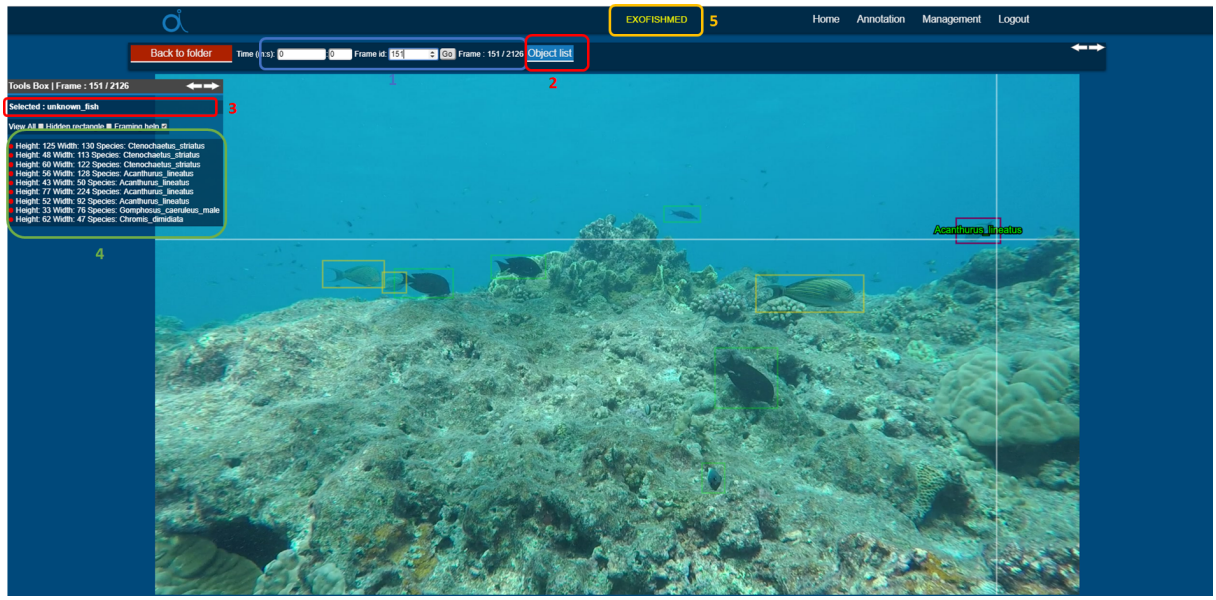


FIGURE 2.6 – Page d'annotation des images extraites d'une vidéo.
1) Permet de se déplacer à travers les images de la vidéos par plusieurs moyens, en entrant directement le numéro de l'images ou le temps de la vidéo, ou en se déplaçant à l'aide des flèches gauche/droite du clavier.
2) Présente la liste des classes d'intérêt du projet, et permet à l'utilisateur d'en choisir un pour les prochaines annotations.
3) Montre l'objet d'intérêt courant.
4) Présente la liste des boites englobantes présentes dans l'image courante : taille, et nom de l'objet.
5) Nom du projet auquel appartient la vidéo
Enfin, la partie principale de l'application permet de visualiser l'image courante et de dessiner les boites englobante grâce à un curseur (en blanc sur l'image). Il est aussi possible de regarder les différentes classes des objets présents en survolant les boites englobantes (en violet sur l'image), et de la comparer aux autres objets de la même classe présents sur l'image (boîtes englobantes vertes).

Fonctionnement de l'application d'annotation

L'application permet à un utilisateur (appelé chef de projet, CDP) de créer un projet (qui peut correspondre à une campagne de terrain, un projet scientifique, ou plus simplement une vidéo), pour lequel il définira une liste de classes. Ces classes seront celles les objets d'intérêt du projet (e.g. des espèces, genres, familles de poissons). Cette liste peut être définie spécifiquement pour le projet, ou bien importé depuis un autre projet existant. Une structure arborescente est proposée afin d'organiser les classes étudiées suivant la classification taxonomique.

Dans le cas de l'annotation d'espèces de poissons, nous proposons une vérification de nom des espèces via le site WORMS (WOrld Register of Marine Species [6]). L'ajout de classes à la liste d'un projet est entièrement à la discrétion de l'utilisateur et peut donc aller plus loin que l'espèce (sexe, comportement, etc). Une fois cette liste définie, le CDP pourra charger des vidéos, puis les découper en suite d'images à un certain nombre d'image par secondes, selon son besoin. Lors de l'import d'une vidéo dans l'application, nous imposons au chef de projet de rentrer un certain nombre de méta-données (Type d'appareil photo/caméra, heure, coordonnées GPS, profondeur de la vidéo, etcpermettant la traçabilité des données. Après avoir découpé une vidéo en suite d'image, le CDP, ainsi que des utilisateurs invités, peuvent ensuite annoter ces images, c'est-à-dire localiser les objets d'intérêt dans les images grâce à une boite englobante, ainsi que les identifier grâce aux classes définies précédemment. De nombreuses options sont présentes pour faciliter le travail des annotateurs (Fig. 2.6)

2.5 Construction des bases de données pour nos travaux de recherche

Grâce aux dizaines de vidéos à disposition, à cet outil et au travail de nombreux annotateurs (+40 annotateurs volontaires ont participé à l'effort d'annotation, pendant des stages ou via l'association du Groupe Naturaliste de l'Université de Montpellier), nous avons constitué plusieurs bases de données d'images pour répondre aux différentes problématiques de la thèse. Un tableau récapitulatif (Tab. 2.4) résume les bases de données principales construites pendant les travaux de thèse à la fin du chapitre.

Base de données pour comparer le *Machine Learning* et le *Deep Learning* sur une tâche d'identification

La première base de données a été conçue pour des algorithmes de détection et de localisation (chapitre 3) de poissons dans des vidéos sous-marines. Nous avons extrait 7256 images appartenant aux 9 espèces les plus communes des vidéos tournées à Mayotte entre 2014 et 2016 (Fig.2.8). Deux des neuf espèces ont été considérées comme une classe unique, car elles ont été jugé extrêmement difficile à différencier sur les vidéos (*Chromis viridis*/*Chromis atripectoralis*). La taille des images varie entre 20x40 pixels à 150x200 pixels. Nous avons ensuite effectué des rotations et des flips horizontaux sur ces images pour obtenir une base de données de 30 000 images. Nous avons ensuite ajouté 3 classes à ce jeu de données. La première classe est constitué d'images de fond marin. Cette classe est constituée de 116.820 images de fond aléatoire (extraites de n'importe quelle partie de l'image) et de 91.247 image de fond «spécifique», c'est-à-dire des images de fond extraites

6. http ://www.marinespecies.org/

des parties bordant les images de poisson (Fig.2.7). C'est cette classe qui permettra à

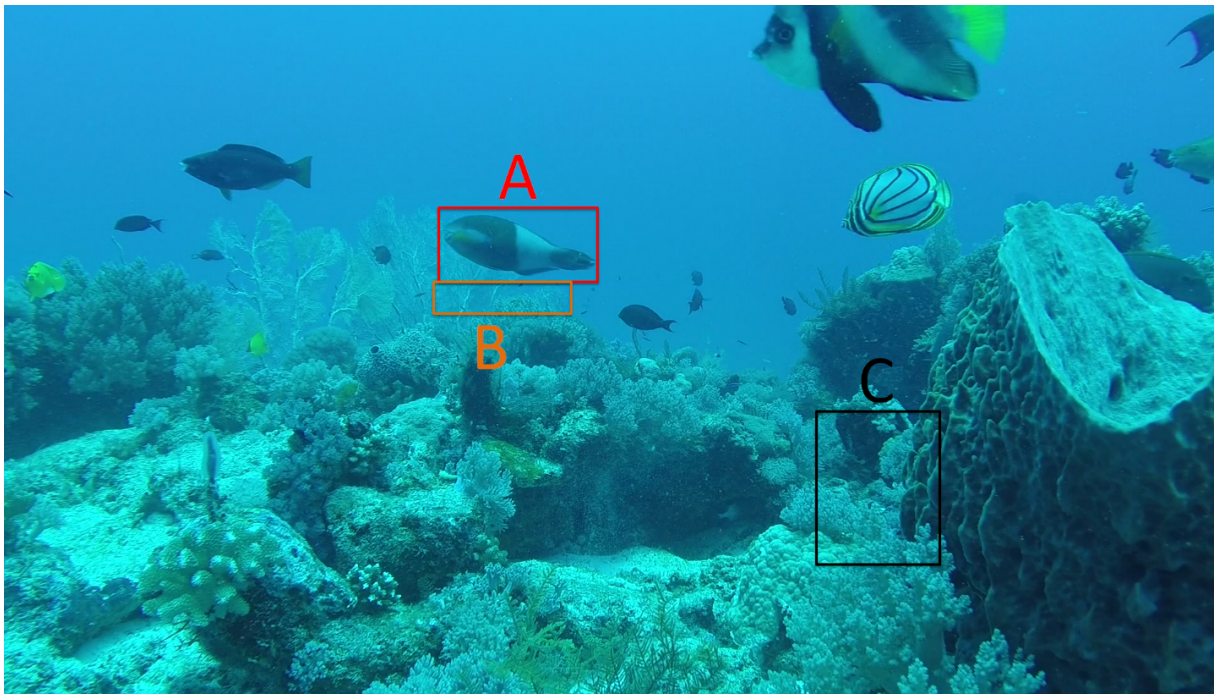


FIGURE 2.7 – Construction des classes "espèces", "fonds spécifiques" et "fonds aléatoires". Après avoir transformé les vidéos en suite d'images, nous avons sélectionné des vignettes de poissons (A), des vignettes de fonds "spécifiques" (B), c'est-à-dire proche des individus appartenant aux classes étudiées, et des vignettes de fonds "aléatoires" (C), c'est-à-dire découpées n'importe où dans l'image. Ces nouvelles vignettes seront ensuite extraites, et utilisées pour entraîner des algorithmes d'apprentissage automatique.

l'algorithme de détection de différencier les objets d'intérêts (poissons) du fond marin. La deuxième classe est constituée de parties d'individus, afin de s'assurer que l'algorithme apprenne effectivement à localiser et identifier des individus entiers, sans se focaliser sur des parties spécifiques (nageoires, taches, etc). La troisième classe est constituée de divers poissons qui n'étaient pas suffisamment représentés pour être appris en tant que classe spécifique. Cette classe permet d'effectuer la localisation sur tous les individus, même ceux qui ne peuvent pas être identifiés à l'espèce. Au final, la base de données permettant l'entraînement de nos premiers modèles d'identification et de détection est constituée de 286 800 images (Tab.2.5).

Nous avons ensuite utilisé 4 vidéos présentant différentes caractéristiques en terme de profondeur, d'environnement, et de communautés, que nous avons découpé en une série d'images, et sélectionné aléatoirement 100 images pour chaque vidéo. Nous avons ensuite annoté l'ensemble des poissons présents de taille $>$ 50x50 pixels. Ces images et annotations composeront notre base de test, pour évaluer nos modèles créés par apprentissage lors des différentes expériences.

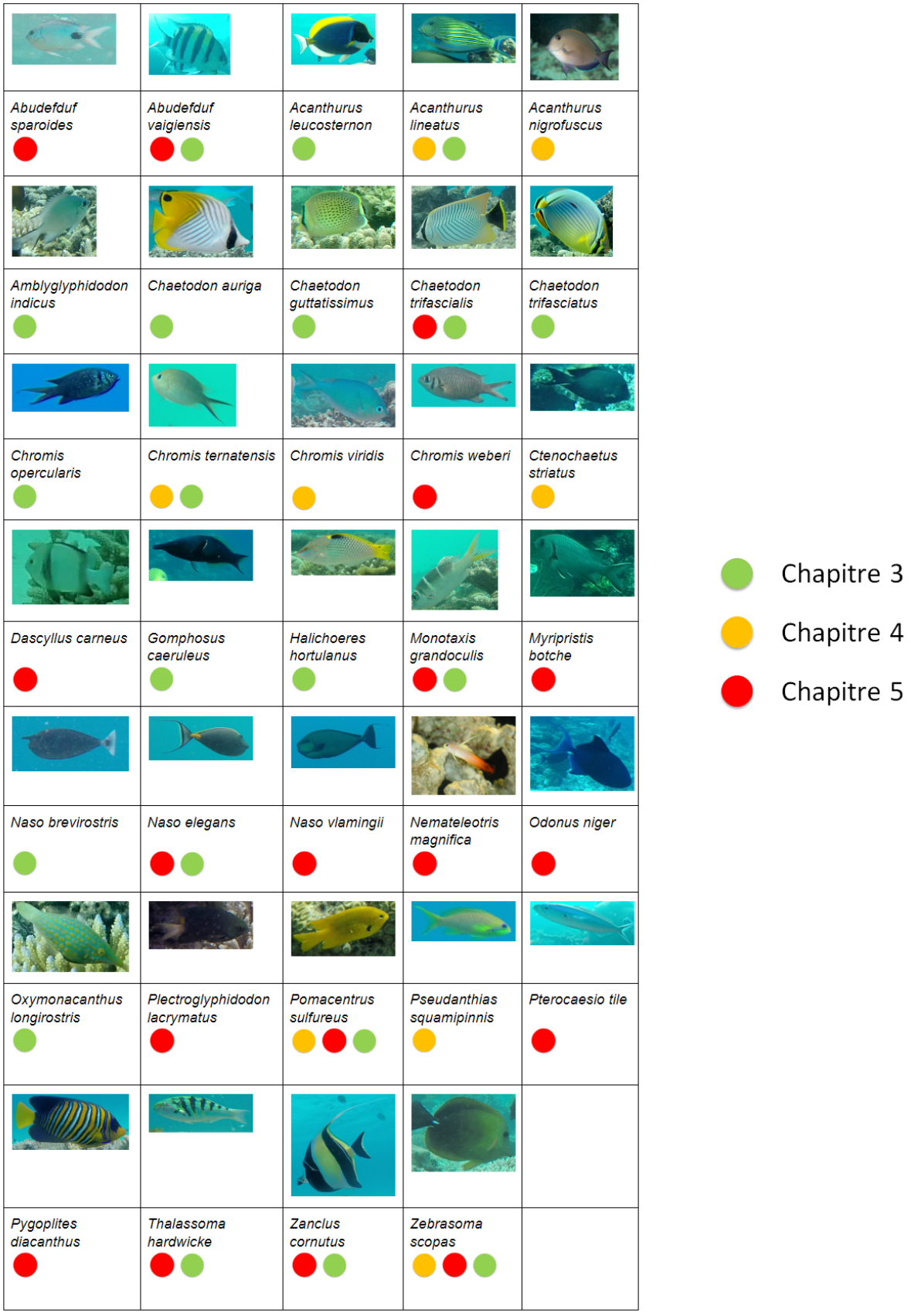


FIGURE 2.8 – Liste des espèces étudiées dans les chapitres 3, 4 et 5.

TABLE 2.1 – Tableau récapitulatif des données d'entraînement du modèle pour le chapitre 3.

Classe	Images d'entraînement
Acanthurus lineatus	2465
Acanthurus nigrofuscus	3923
Chromis ternatensis	4755
Chromis viridis/Chromis atripectoralis	2619
Pomacentrus sulfureus	3830
Pseudanthias squamipinnis	5900
Zebrasoma scopas	2400
Ctenochatus striatus	4000
Fonds Marins Aléatoires/Spécifiques	116.820/91.247
Parties de poissons	55.848
Poissons Inconnus	970

Amélioration de la détection et comparaison à l'humain

Pour le second article (Chapitre 4), nous avons utilisé les même 116 vidéos que pour le jeu de données précèdent. En revanche, nous avons défini des règles plus précise pour la délimitation des boites englobantes permettant l'extraction des images spécifiques demandées pour l'entraînement de nos modèles :

1) Un individu est annoté si et seulement si moins de 10% de sa surface est couverte par d'autre individu, coraux, etc.

2) Un individu est annoté si et seulement si l'individu est identifiable à l'espèce sur l'image courante, ainsi, nous n'avons pas d'individu vues de face ou de dos, car la majorité des caractéristiques permettant l'identification sur les images n'est pas visible.

3) Un individu est annoté si et seulement si sa taille est supérieure à 3000 pixels carrés.

Nous avons ainsi collecté 44.625 images de 20 espèces (Fig.2.8), appartenant aux 12 familles les plus présentes sur le récif. Grâce à ce jeu de données (T0), nous avons créé 4 bases pour entraîner nos modèles Fig. 2.9.

La première base *D1* contient les images originales, ainsi que les flips horizontaux. Cette transformation permet d'équilibrer les individus orientés vers la gauche et ceux orienté vers la droite, afin d'éviter un biais éventuel. Nous avons supprimé les rotations afin de conserver uniquement des images "naturelles". En effet, les rotations pouvaient entraîner des images impossible (individus à l'envers par exemple). La seconde base *D2* contient les images de *D1+* une classe "partie de poisson" comme vue précédemment. La troisième base *D3* contient les mêmes images que *D2*, plus une classe "environnement". Contrairement au premier article, le second est uniquement consacré à l'identification, sans tâche de localisation. La classe environnement permet ici d'ajouter de la diversité lors de l'entraînement des modèles pour les renforcer. La quatrième base *D4* contient les

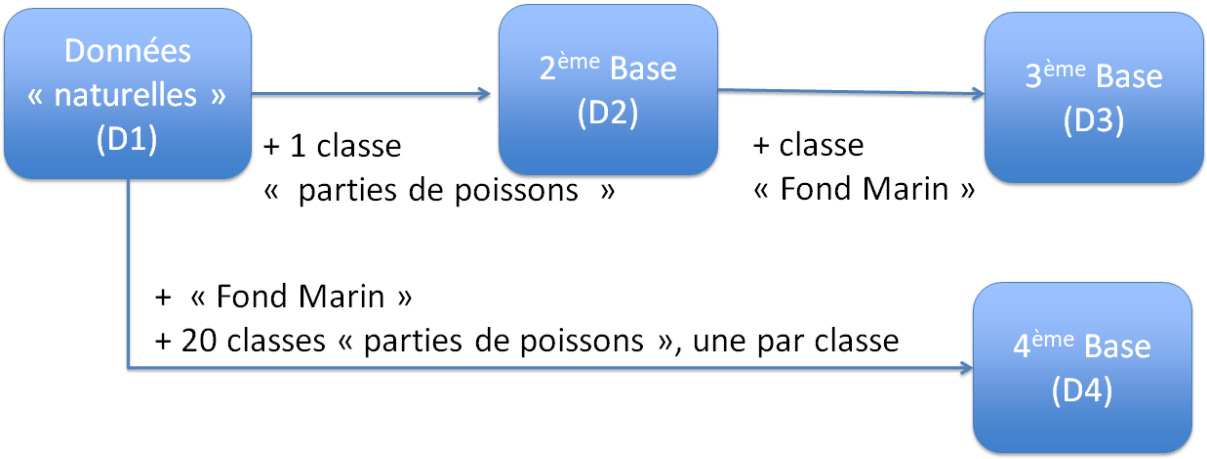


FIGURE 2.9 – Construction des bases de données pour le chapitre 4.

images de la base *D3* moins la classe "partie de poissons". À la place, nous avons ajouté une classe "partie d'une espèce de poisson" pour chaque espèce apprise. Nous avons créé ces classes afin de pouvoir identifier des individus occultés en grande partie.

Nous avons ensuite utilisé 6 vidéos indépendantes (sites et jours différents de ceux de l'entraînement). Nous avons utilisé les mêmes règles de sélection que celles vues précédemment pour collecter les vignettes de test, en ajoutant les images d'individus partiellement cachés, ainsi que ceux vu de dos ou de face Nous avons extrait 4.405 images appartenant à 18 des 20 espèces utilisées pour l'entraînement. Nous avons ensuite choisi sous échantillon de 1197 individus appartenant à 9 espèces pour constituer un troisième jeu de test, qui sera utilisé lors de nos comparaison entre nos algorithmes et un panel humain (Fig. 2.2). Nous avons utilisé ce sous échantillon pour des raisons pratiques, afin de réduire les données à faire traiter par des humains, tout en conservant la moitié des espèces afin de rester statistiquement représentatif.

Amélioration de l'identification et contrôle du taux d'erreur d'un CNN

Pour les besoins de notre 3ème article (Chapitre 5), lui aussi consacré à de l'identification, nous avons eu besoin de créer 3 jeux de données. Pour le premier jeu de données *T1*, nous avons utilisé 69.169 images extraites de 130 vidéos tournées entre 2016 et 2017. Ces images serviront de base d'entraînement pour nos algorithmes. Pour chacune de ces images, nous avons appliqué 4 modifications : 2 augmentations de contraste (120% et 140%) et 2 diminutions de contraste (80% et 60%). Nous obtenons alors 5 images. Nous appliquons alors un filtre de translation horizontal afin d'obtenir 5 images de plus et d'équilibrer les individus orienté vers la droite et vers la gauche. Ainsi, nous obtenons une base d'entraînement contenant un total de 691.690 images.

Pour les 2 jeux de données suivant, nous avons utilisés les vidéos réalisées lors de notre

TABLE 2.2 – Tableau récapitulatif des données d'entraînement du modèle finale (D4) pour le chapitre 4.

Espèces	(D4) Images d'entraînement	Test des performances du modèle	Comparaison des performances : modèle vs humain
Abudefduf sparoides	2482	103	88
Abudefduf vaigiensis	11328	59	47
Chaetodon trifascialis	2912	208	146
Chromis weberi	7152	269	
Dascyllus carneus	4552	269	
Monotaxis grandoculis	3300	72	
Myripristis botche	2478	20	
Naso elegans	2528	189	165
Naso vlamingii	2528	358	
Nemateleotris magnifica	2378	246	
Odonus niger	5972	176	
Plectroglyphidodon lacrtymatus	1304	150	
Pomacentrus sulfureus	10352	1567	443
Pterocaesio tile	6176	215	
Pygoplytes diacanthus	2212	39	35
Thalassoma hardwicke	3158	111	73
Zanclus cornutus	3772	64	53
Zebrasoma scopas	3670	184	144
Environnement	862174		

campagne à Mayotte (Chapitre 2.3). Ces vidéos ont été séparées selon les jours et les sites d'enregistrement afin de s'assurer qu'ils soient indépendants. Comme pour les jeux de données précédents, nous avons ensuite découpé ces vidéos à 5 images par secondes, avant de faire les annotations, puis d'extraire les vignettes de 20 espèces. Nous avons utilisé 6320 images appartenant aux 20 espèces extraites de 20 vidéos pour le 2ème jeu de données *T2* et 13.232 images extraites de 25 vidéos pour le dernier jeu de données *T3* (Tab. 2.3).

TABLE 2.3 – Tableau récapitulatif des données d'entraînement (T1), de selection du seuil de confiance (T2) et de test (T3) du modèle pour le chapitre 5.

Espèces	*T1*	*T2*	*T3*
Abudefduf vaigiensis	51.240	376	216
Acanthurus leucosternon	35.590	235	491
Acanthurus lineatus	10.080	114	864
Amblyglyphidodon indicus	11.880	636	1.310
Chaetodon auriga	21.340	737	502
Chaetodon guttatissimus	11.820	221	68
Chaetodon trifascialis	52.340	41	630
Chaetodon trifasciatus	44.210	71	82
Chromis opercularis	15.250	81	93
Chromis ternatensis	36.400	300	156
Gomphosus caeruleus	31.310	57	173
Halichoeres hortulanus	31.820	40	287
Naso brevirostris	11.340	539	1.932
Naso elegans	73.450	1.436	3.896
Monotaxis grandoculis	38.930	797	1.422
Oxymonacanthus longirostris	25.530	54	55
Pomacentrus sulfureus	54.090	270	142
Thalassoma hardwicke	49.510	181	275
Zanclus cornutus	38.760	86	59
Zebrasoma scopas	49.700	48	579

Détection de poissons dans des vidéos sous-marines.

Les besoins pour la base d'entraînement de notre 4ème article (Chapitre 6), était différents de ceux des autres articles. En effet, pour entraîner notre modèle de localisation de poisson, nous avons besoin d'utiliser des images complètes (issues du découpage de la vidéo en frames), dans lesquelles tous les individus visibles et identifiables sont annotés (Fig. 2.10).

Les vidéos utilisées pour construire la base de données d'entraînement provennaient toutes du canal du Mozambique. Plus particulièrement, nous avons utilisé 20 vidéos réalisées à Mayotte, et 8 vidéos réalisées à Europa (22°22'03.7"S 40°21'13.6"E). Nous

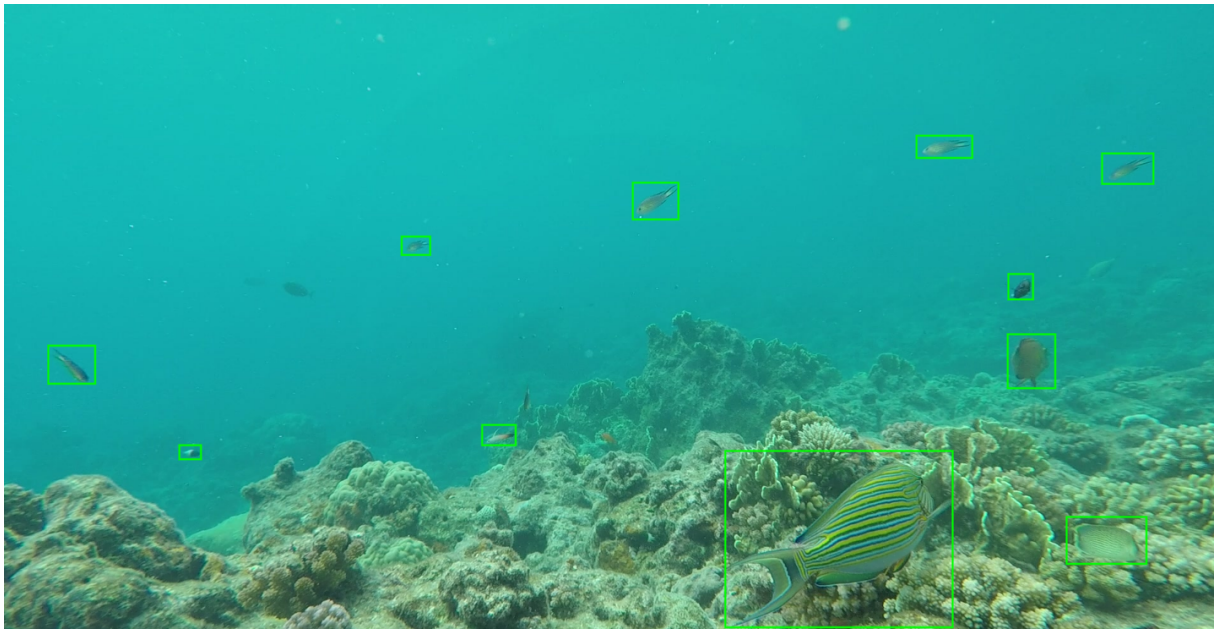


FIGURE 2.10 – Exemple d'annotation exhaustif d'une frame.
Tous les individus identifiables (au genre/espèces) sont annotés.

avons ensuite découpé ces vidéos en séries de frames à un ratio de 2 frames par seconde, puis extrait des séries de 11 frames consécutives, annotées exhaustivement, au taxon le plus précis possible (famille/genre/espèce). L'information du taxon n'est pas utilisé par l'algorithme d'identification, (qui localisera une classe unique "poisson"), mais a été utilisé afin de contrôler les espèces présentes ou absentes lors de l'apprentissage, ainsi que leur représentation (nombre d'individu). Chaque frame contient entre 0 et 32 annotations.

Nous avons utilisés la base de données ainsi obtenue pour créer 4 jeux d'entraînement, le but étant d'étudier le compromis entre le temps d'annotation et la variabilité des vidéos au sein des vidéos utilisées pour créer la base d'entraînement (nombre de vidéos et d'arrières plans, d'environnements différents), et les résultats des modèles obtenus à partir de ces bases. La première base (B1) contient 3603 annotations appartenant à 40 espèces (entre 1 et 882 annotation par classes). La seconde base (B2) contient 6146 annotations appartenant à 57 espèces (entre 1 et 1719 annotation par classes), la troisième (B3) 11230 annotations appartenant à 97 classes (entre 1 et 2404 annotations par classes), et la quatrième (B4) 17708 annotations appartenant à 125 classes (entre 1 et 10675 annotation par classes). La base d'entraînement est très inégale en terme de nombre d'annotations par classes, la base B4 contenant par exemple 10675 annotations de *Chromis dimidiata* et 2404 annotations de *Pomacentrus sulfureus*, pour seulement 3 annotations de *Chaetodon lunulatus* et 1 annotation de *Kyphosus vaigiensis*. Nous avons ensuite construit 5 jeux de test, grâce, à des vidéos indépendantes (e.g. réalisés dans d'autres lieux et à une autre date (Fig.

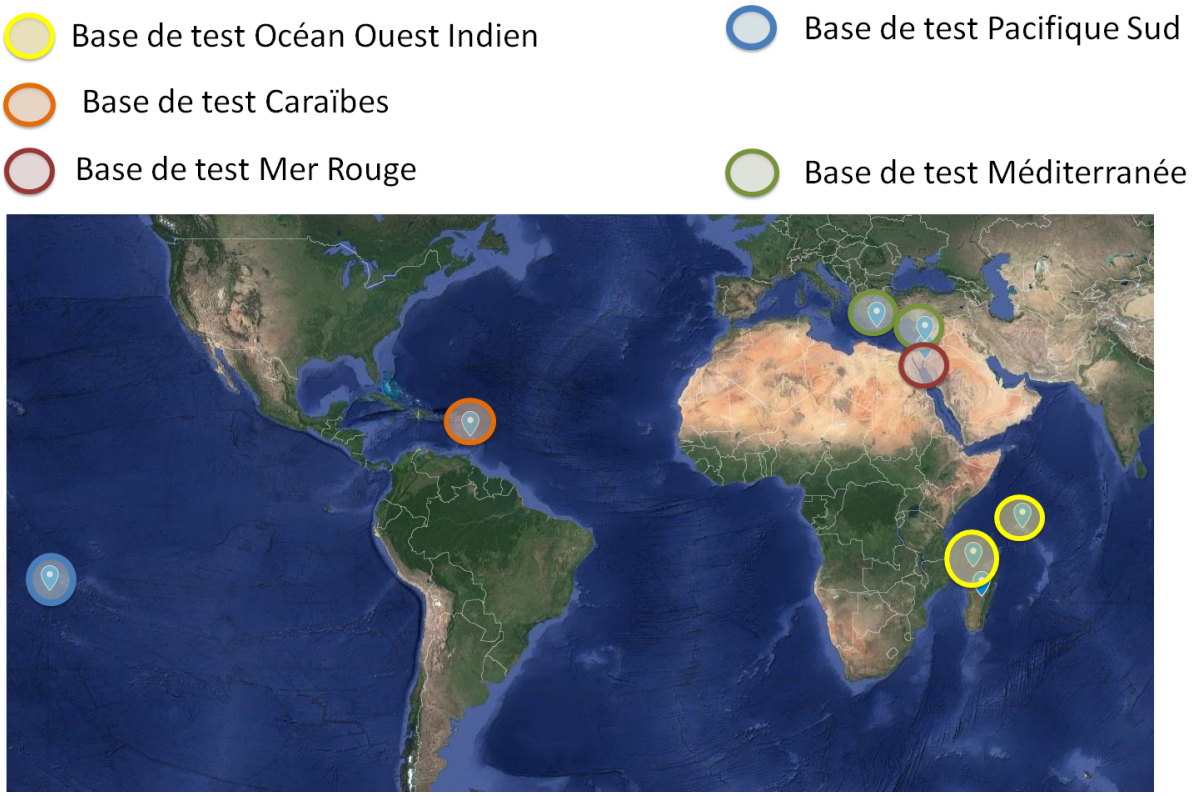


FIGURE 2.11 – Localisation des enregistrements vidéos pour les bases de tests de notre modèle.

2.11). Les 5 jeux de test, du plus ressemblant à l'entraînement en terme de conditions et d'espèces présente à l'écran au plus éloigné sont :

- Un jeu de données "Océan Ouest Indien", construit à partir de RUVs réalisées à Mayotte en Septembre 2016 et dans les Seychelles en Septembre 2017, contenant 5431 annotations de 66 classes (entre 1 et 1794 annotations par classes).
- Un jeu de données "Mer Rouge" réalisé par RUVs à Eilat (Israël) en Mai 2018, contenant 2930 annotations de 28 classes (entre 1 et 963 annotations par classes), dont 19 classes non présentes à l'apprentissage.
- Un jeu de données "Océan Pacifique Sud" réalisé par RUVs à Moorea en Octobre 2017, contenant 2342 annotations de 36 classes (entre 2 et 500 annotations par classes), dont 17 classes non présentes à l'apprentissage.
- Un jeu de données "Mer des Caraïbes" réalisé par RUVs en Martinique Décembre 2017, contenant 1866 annotations de 21 classes (entre 2 et 1035 annotations par classes), dont 20 classes non présentes à l'apprentissage.
- Un jeu de données "Mer Méditerrannée" réalisé par RUVs en à Haifa (Israël) en Mai 2019 et en Crète en Juin 2019, contenant 714 annotations de 13 classes (entre 1 et

TABLE 2.4 – Tableau récapitulatif des principales bases de données créées au cours de la thèse.
Les classes fond/contexte/"parties de poissons" ne sont pas présentées dans ce tableau, seules les classes correspondant à des espèces, ou à des paires d'espèces (e.g. la classe *Chromis viridis*/*Chromis atripectoralis*) dans la base d'entraînement du chapitre 3).

Base de donnée	Type d'annotation	Nombre de classes traitées	Nombre total d'individus annotés
Chapitre 3	Vignettes	8	Entrainement 5856 Test 3600
Chapitre 4	Vignettes	18	Entrainement 46325 Test 4405
Chapitre 5	Vignettes	20	Entrainement 69169 Test 13232
Chapitre 6	Frames annotées exhaustivement	1 (détection uniquement)	Entrainement 17708 Test 13283

122 annotations par classes), dont 13 classes non présentes à l'apprentissage.

L'ensemble de ces bases de données nous a permis d'étudier la robustesse de modèles entraînés dans des conditions spécifiques (Ouest Indien), pour identifier des espèces inconnues dans des conditions différentes. Chaque jeu de données contient au minimum 3 vidéos (3 contextes/fonds différent).

Chapitre 3

Le Deep Learning est il plus efficace que le Machine Learning pour des tâches d'identification d'espèces de poisson ?

Bien que le l'apprentissage profond ai montré un fort potentiel pour la classification d'images [Krizhevsky et al., 2012] [He et al., 2015] [Wang et al., 2017], il était nécessaire de vérifier que ce type d'approche fonctionne correctement pour de l'identification d'espèces de poissons dans des vidéos sous marines naturelles, par rapport aux approches par machine learning [Spampinato et al., 2010] [Villegas et al., 2015] [Shafait et al., 2016].

Nous avons donc comparé deux approches : une méthode classique de l'état de l'art en *machine learning* et un algorithme profond. La première approche consiste à 1) transformer chaque vignette de la base d'entraînement en un vecteur caractéristique grâce à une méthode d'extraction de HOG (*Histogram of Gradient*, telle que décrite en section 1.4) en cascade [Zhu et al., 2006] 2) entraîner avec ces vecteurs un classifieur SVM (Séparateur à Vastes Marges). La deuxième méthode consiste à utiliser la base de vignettes pour entraîner un classifieur profond, implémenté sur le framework DIGITS [1]. Une fois l'entraînement des deux modèles terminés, la phase de test s'effectue en 2 étapes identiques pour les 2 approches.

Premièrement, grâce à une fenêtre glissante multi-résolutions, chaque image de test est découpée en sous-régions rectangulaires de taille variable, allant incrémentalement de la taille totale de l'image à 1/18 de chaque dimension (largeur et longueur), avec un recouvrement égal à un tiers de la taille des sous-régions.

Une fois les images à traiter découpées en sous-régions, chaque algorithme effectue la même chaîne de traitement. Tout d'abord, il classifie l'ensemble des sous-régions en

1. https ://developer.nvidia.com/digits

leur attribuant une classe. Deuxièmement, un fois que chaque sous-région est classée, un post-traitement est effectué afin de fusionner les régions se superposant fortement ensemble et affectées à la même classe. Ce post-traitement permet d'éviter de multiples identifications du même objet.

Finalement, les deux approches sont comparées en terme de rappel (nombre d'individus à détecter effectivement détectés) et de précision (ratio de boîtes englobantes attribuées à la bonne classe). Le remplacement de la méthode *Machine Learning* par une méthode *Deep Learning* pour la tâche d'identification nous a permis d'obtenir une Fmesure entre 1.5 fois supérieure et 3 fois supérieure selon la vidéo traitée.

Nous avons aussi observé la diminution (puis la disparition) de méthodes de ML pour les tâches d'identification de poissons au profit d'emploi de plus en plus fréquent de DL. [Salman et al., 2016] [Qin et al., 2016] [Salman et al., 2019].

Coral reef fish detection and recognition in underwater videos by supervised machine learning : Comparison between Deep Learning and HOG+SVM methods

Sébastien Villon[1], Marc Chaumont[1,2], Gérard Subsol[2], Sébastien Villéger[3], Thomas Claverie[3], and David Mouillot[3]

[1] LIRMM, University of Montpellier/CNRS, France
[2] University of Nîmes, France
[3] MARBEC, IRD/Ifremer/University of Montpellier/CNRS, France

Abstract. In this paper, we present two supervised machine learning methods to automatically detect and recognize coral reef fishes in underwater HD videos. The first method relies on a traditional two-step approach: extraction of HOG features and use of a SVM classifier. The second method is based on Deep Learning. We compare the results of the two methods on real data and discuss their strengths and weaknesses.

1 Introduction

Quantifying human impact on fish biodiversity in order to propose solutions to preserve submarine ecosystems is an important line of research for marine ecology. This quantification requires in situ sampling of the fish community. Measurements based on extraction-fishing give only limited data, and could lead to misinterpretation [1]. Moreover, the use of fishing, even for survey purposes, impacts the studied biodiversity.

Another standard method consists in two divers who note visual observations of fishes under water. This kind of survey is expensive in both time and money, and results are greatly impacted by divers' experience and fish behavior. Moreover, data acquisition remains limited by the human physical capacities [1].

A more recent method consists in acquiring underwater images or videos [3], with either a moving or a fixed camera. An expert will then be asked to detect, count and recognize fishes on a screen offline. At the moment, this task is performed entirely manually, and the amount of data is often too large to be completely analyzed on screen. Moreover, the latest technical improvements of HD camera allow recording fish communities for a long time at a very low cost. Significant examples of a huge amount of underwater HD images/videos, that have been collected for assessing fish biodiversity, are the 115 terabytes

of the European project Fish4Knowledge [3], or the XL Catlin Seaview Survey Initiative[4].

The research community in image processing has been asked to propose algorithms in order to assist, and recently even automatize the detection/identification of fishes in images or videos. Recently, a challenge called Coral Reef Species Recognition has been proposed in the evaluation campaign SeaClef [5], which is based indeed on Fish4Knowledge data. Unfortunately, in this task, the video quality remains quite limited (640×480 pixels) whereas current acquisitions are in High Definition or even in 4K (see Figure 1). This offers much more details for image processing but increases the processing time.

Fig. 1. Left, a 640×480 highly compressed frame extracted from the SeaClef database. Right, a 1280×720 HD frame from the MARBEC laboratory database.

Among the issues and difficulties of detecting and recognizing fish in underwater videos, there are color variations due to the depth, lighting variations from one frame to another, the sediments and dirt which degrades the videos quality, or the seaweed which makes the background changing and moving [4]... The classification itself encounters other issues such as the variation of shape or color within the same species and moreover the variation of size and orientation due to the fish position. We chose not to avoid these issues,and to take into account all these problems as we work on videos acquired in natural conditions instead of controlled acquisitions [5]. We chose for this study to focus on the processing of each frame and not on the video.

Many methods to detect and recognize fishes in underwater videos were proposed these last years [3]. In general, the first step consists in selecting features based on the shape, color, context, specifics landmarks or texture [6]. Some algorithms use specific feature vectors computed at some landmarks. Other use more complex features such as SIFT [8–10] or *shape context* (SC) [5]. But in [11], the authors conclude that the Histogram Of Oriented Gradients feature leads to better results than both SIFT and SC.

In the 2015 SeaClef contest [23], the best results have shown that Deep Learning can achieve a better classification for fish detection than SVM or other

[4] http://catlinseaviewsurvey.com/
[5] http://www.imageclef.org/lifeclef/2016/sea

classical methods. This may be due to the fact that in Deep Learning, features are automatically built by the classifier itself, in an optimal way. The winner of the SeaClef contest used several Deep classifiers and fused the results to obtain the definitive scores. Unfortunately, we will not be able to compare our approach with his as the databases are different (we have a higher definition and mobile cameras).

In this paper, we propose also to explore the performances of fish detection and classification by Deep Learning. In particular, we assess the results with respect to a more classical method based on a combination of HOG feature extraction and SVM classification.. For this, we will use High Definition videos acquired for an actual marine ecology study. In section 2, we briefly present Deep-learning and SVM+HOG methods. In section 3, we detail the implementation in particular the multi-resolution approach and data preprocessing. In section 4 we present the results and compare both methods. In section 5, we present some future work.

2 Presentation of the Methods

2.1 Histogram of Oriented Gradients + Support Vector Machine

The Histogram of Oriented Gradients [12] characterizes an object in an image based on its contours by using the distribution of the orientations of local gradients. As shown in [13], HOG features may lead to better results even in a complex classification task as ours, where a fish can be hidden in coral reefs or occluded by another fish.

The Support Vector Machine (SVM) [7] is a supervised method to classify feature vectors. SVM method represents each vector in a high dimensional space, mapped so that the samples of the different classes are separated by a clear gap that is as wide as possible. Support vector machines have been used in a lot of applications and have shown good overall results [14–17].

2.2 Deep Learning

Since the 2012 ImageNet competition, and new computational power accessible through latest GPU, Neural Network came back as a strong possibility for classification tasks [18]. Moreover, by integrating convolutional layers, Deep Neural Networks (DNN) are able to both create features vectors and classify them.

Neural network is a mathematical model which tries to mimic human brains [19]. Like SVM, neural networks may classify feature vectors after a training phase. A neural network is composed of interconnected nodes called neurons and each neuron of each layer receives a signal from the neurons of the previous layer. This signal is modified according to an activation function and transferred to the neurons of the next layer.

We can define for the neuron n, the first operation $\alpha^{(n)}$ as:

$$\alpha^{(n)}(\mathbf{x}^{(n)}) = \sum_{i=1}^{c} w_i^{(n)} x_i^{(n)} \tag{1}$$

where $\mathbf{x}$ is the input vector, a given neuron, c the number of connections of this neuron, $w_i^{(n)}$ the weight of rank i of a neuron n, and $x_i^{(n)}$ the input of rank i of a neuron n.

We can then define the output of a neuron n as $\sigma^{(n)}$ with $f^{(n)}$ the activation function:

$$\sigma^{(n)}(\mathbf{x}^{(n)}) = f^{(n)}(\alpha^{(n)}(\mathbf{x}^{(n)})) \tag{2}$$

Each layer of a neural network except the first one which receive the feature vector and the last one are called hidden layers. During the learning process, the network will have its parameters modified in order to optimize its classification rate of learning.

We will use the back-propagation. Given a feature vector representing an object from class $k \in \{1..K\}$ as network input, we compare the expected value (100% of probability to belong to class k) to the results obtained by the network, and we compute the error of the output layer. Then, the error is back-propagated from the layer j to the neurons of the layer $j-1$. Finally, the weight of each neuron is updated according to a gradient-based descent in order to get a computed value closer to the expected value [20].

To make a network able to build its own feature, we move from a simple network to a Convolutional Neural Network (CNN). One or more convolutional layers are connected between the input layer and the hidden layers. Each convolutional layer transforms the signal sent from the previous layer using convolutional kernels, an activation function breaking the linearity and a pooling phase which reduces the image and strengthens the learning by selecting significant pixels (the highest value from a region for instance). The last convolutional layer eventually concatenates all the information in one feature vector and sends it to another layer or to a classifier.

For the training phase, a CNN is given a database consisting of couples $(I^i, l^i)_{i=1}^{i=N}$ with I^i, the image $i \in \{1, ..., N\}$ and l^i its label. Basically, in our application, the label $l^i \in \{1, ..., L\}$ is the fish species.

3 Practical implementation

3.1 Data preprocessing

The choice of learning data is a crucial point. We worked with biology experts of the MARBEC laboratory to label many videos. We cropped some frames of the

videos and created a training database composed of 13000 fish thumbnails. The thumbnail size varies from a minimum of 20×40 pixels to a maximum of 150×200 pixels. Each thumbnail contains only one labeled-fish as shown on Figure 2.

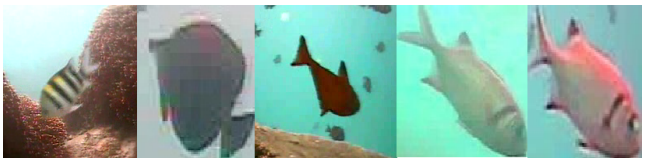


Fig. 2. Some training thumbnails from the MARBEC database

We decided to keep only the species with more than 450 thumbnails. We also widen the database by applying rotations and symmetries in order to capture all the possible position of the fishes. Table 1 lists the retained species.

Species	**Thumbnails**	**Rotations and symmetries**
Acanthurus lineatus	493	2465
Acanthurus nigrofuscus	1455	3923
Chromis ternatensis	951	4755
Chromis viridis/Chromis atripectoralis	523	2619
Pomacentrus sulfureus	766	3830
Pseudanthias squamipinnis	1180	5900
Zebrasoma scopas	488	2400
Ctenochatus striatus	1400	4000

Table 1. Fish species in the learning database

Due to the highly textured natural background, we also added a class for the background. This class is constituted with *random* thumbnails of the background which were randomly selected in frames and *specific* background thumbnails which were taken around the fish thumbnails.
To be able to do multi-resolution classification, all the background thumbnails were taken with random dimensions, from 40×60 pixels to 400×500 pixels.
Finally, in order to improve the localization accuracy, we decided to create another class called *part of fish*, to ensure that the network does not focus on a distinctive part of a fish as a stripe, a fin, the head, but processes the fish as a whole. We also created a class *fish* which contains some unknown fishes to make the method able to recognize any fish even though it is not in the learning database. However, this class must contain less samples in order to be sure that a fish will most likely be labeled by its specific class rather than the generic class *fish*.Finally, we added 3 classes to our initial thumbnail database as listed in Table 2.

3.2 Detection/Recognition Pipeline

The HOG+SVM and the Deep Learning methods process the video frames through the same pipeline (see Figure 3). We chose for this study to process

Class Label	Samples
Random/specific background	116,820/91,247
Part of Fish	55,848
Fish	970

Table 2. Classes added to the species database

each frame independently without introducing any object tracking algorithm. First, we pass a multi-resolution sliding window through the frame. For each position of the window, the method gives a probability score for each class. We also compute a motion probability score based on the comparison of the current and the previous frame. We then compare the probability scores given by the classifier to some predefined thresholds. If the scores are over the thresholds, we output a bounding box corresponding to the window position. At the end, for each position, we will fuse all the bounding boxes found at different resolutions.

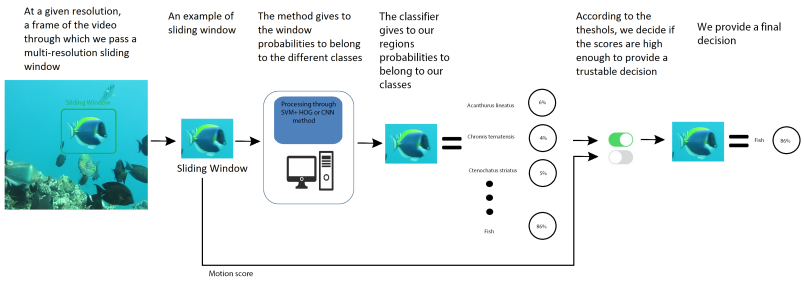


Fig. 3. Detection/Recognition pipeline

Multi-resolution sliding window In order to deal with multi-resolution detection, the size of the sliding window varies from 1/18 of the frame at least, and 1/1 at most. This allows to recognizes fishes with a minimum of 60 pixel length and 40 pixel width, and a maximum equal to the full size of the frame. The sliding window is displaced by a stride equals to a third of the window width.

HOG + SVM We divide each thumbnail in 10 zones (one zone is the complete thumbnail, and the 9 others are the thumbnail divided in 9 even parts). For each zone, we compute a HOG feature over 8 direction axes, and we concatenate all these HOG features in a unique feature vector. For each fish species, we fed a SVM with all the corresponding thumbnail features as a class, and all the other thumbnails features (other species and background) in order to obtain a specific classifier.

The SVM we used non-linear SVR (Support Vector Machine for regression) implemented using the library libsvm[6] with a Gaussian radial basis function kernel. We obtained a clean separation for the training database (over 85% of backgrounds thumbnails have a regression score lesser than 0.5, and 85% of fishes thumbnails have a regression score greater than 0.5) We built as many classifiers as there are classes, and each classifier discriminates one class against all the others. In the end, if none of the classifiers class the window as a fish, it will be classified as "background".

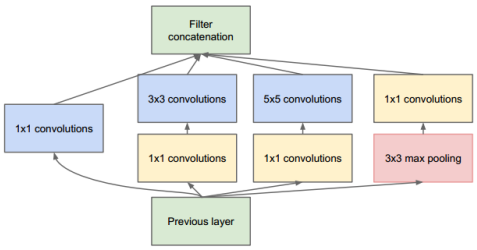


Fig. 4. An inception module, as presented in SZEGEDY et al. [22]

Deep Learning The architecture of our network follows the GoogLeNet's with 27 layers, 9 inception layers, and a soft-max classifier. Once we have a list of cropped thumbnails and their labels, we send them to our network. We use inception layers (fig 4) based on GoogLeNet architecture [22]. The inceptions here allows us to reduce the dimension of the picture to one pixel, and therefore not to be dependent of the dimensional impact. We adapted some parameters such as the size of the strides and the first convolutions adapted to the size of our thumbnails, which allowed us to achieve better results than a classic architecture (e.g [18]).

3.3 Post-processing and Bounding Box Fusion

For each sliding window, we define a motion score by computing the average absolute difference with the window at the same position in the previous frame. Based on the hypothesis that most of the fishes are moving, we use this score for the final detection decision.

After processing all the resolutions of a frame, we obtain a list of bounding boxes, and for each bounding box a probability of belonging to a class. Yet the classification remains ambiguous if there is more than one bounding box corresponding to the same potential fish (see Figure 5).

[6] https://www.csie.ntu.edu.tw/ cjlin/libsvm/

To suppress the redundant bounding boxes, we first keep the boxes whose probabilities are above a given probability threshold T (T =98%) in the case of fig. 5. So if the motion score is greater than our motion threshold and the probabilities to belong to a class of fish is greater than 98%, we keep the box. Then, we fuse the remaining boxes following based on the following properties: if two bounding boxes are labeled with the same species, and their overlap ratio is greater than 30%[7], we suppress the bounding box with the lower probability.

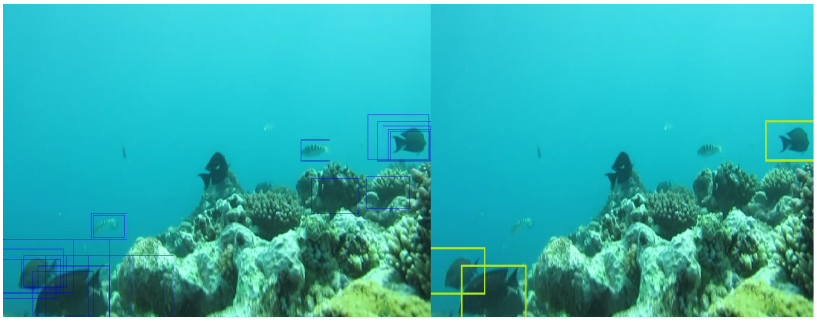


Fig. 5. Results of the detection before (left) and after (right) post-processing. Note that two fishes in the center are not displayed because they do not belong to one of the species which have been learned.

4 Results, Comparison and Discussion

We used 4 test videos (which are different from the training videos) to experiment our complete process. The 4 videos were taken on coral reefs, and the diver was holding the camera which then slightly moves. The video acquisition were not deep, and therefore we had a lot of colors on both the fishes and the background but the light is moving with the waves, bringing many distortions. The videos are very different in terms of fish species, background texture, colors, fish density, etc. Biology experts from MARBEC selected 400 frames all over the videos and defined ground-truth bounding boxes of all the visible fishes in the frame.

To determine if a detected bounding box is correct, we compute its overlap ratio with the ground truth bounding box. If this value is over a threshold λ, then the detection is considered as true positive, otherwise it is labeled as false positive. On the opposite, if a ground truth bounding box has no over the threshold overlap with any detected bounding box, it counts as a true negative. We chose to put the value 0.5 to λ.

Results (recall, precision and F-Measure) of the entire detection/recognition process with Deep-Learning method is given in Table 3 with T=98%.

[7] The overlap ratio is defined as $OA = IS/US$ with IS the intersection surface and US the union surface.

We also show on Figure 6 the relation between recall and precision with respect to the threshold T. The differences come mostly from the texture of the background, but also from the species, as some fishes are easier to detect (bright colors, stripes...).

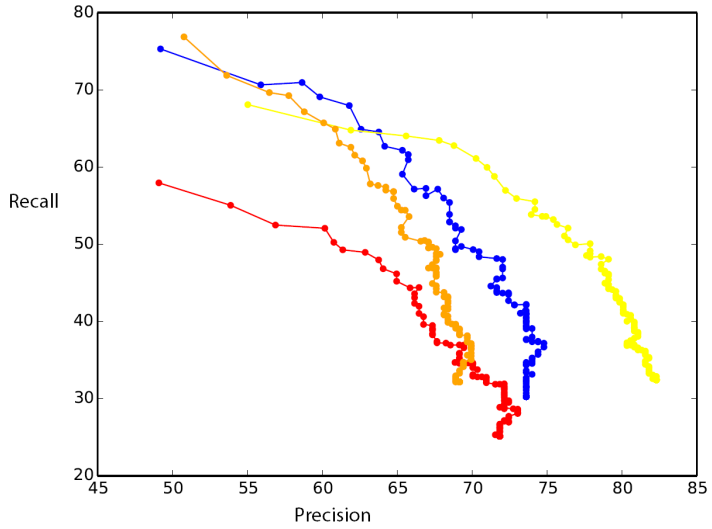


Fig. 6. ROC Curves of the Deep Learning method on the four test videos, with the threshold T as parameter.

Video	Precision	Recall	F-measure
1655	0.58	0.69	0.62
1654	0.68	0.63	0.65
1547	0.77	0.64	0.70
1546	0.60	0.52	0.55

Table 3. Results with the Deep Learning method (T = 98%)

We can now compare in Table 4 the results of the two methods, for the same threshold. It seems that the discrimination of the HOG+SVM is less efficient than the CNN's. Indeed, the F-measure of the HOG+SVM is always below 49% whereas it is always above 55% for the CNN.

As we can observe in Figure 7, the Deep Learning method approach efficiently recognizes fishes on different resolutions even when there is a strongly textured background and is able to distinguishes fishes which are close.

On the other hand, parts of the coral can be misclassified. In Figure 8, we were able to detect the three fishes we were supposed to, but we also detected a

Video	F-measure from HOG+SVM	F-measure Deep Learning
1655	0.28	0.62
1654	0.24	0.65
1547	0.49	0.64
1546	0.14	0.55

Table 4. F-measure of the two methods with the same parameter

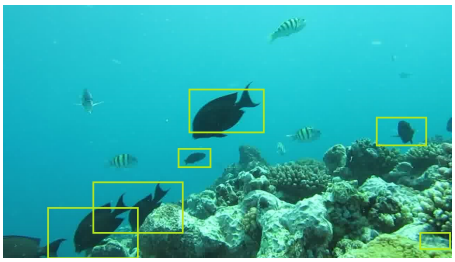


Fig. 7. The Deep Learning method succeeds in detecting fishes partially occluded by coral (bottom left)

part of the coral reef which presents features we can also find on fishes such as an enlighten top and a darker bottom, an oval shape, etc.

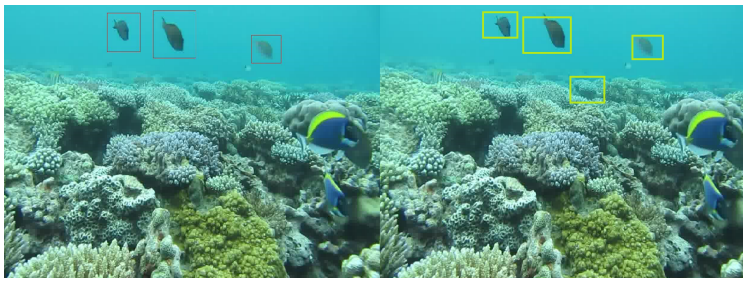


Fig. 8. A rock detected as a fish. On the left, the ground truth, on the right, the results of our processing

5 Future Work

In this paper, we have presented two methods to detect and recognize fishes in underwater videos.

When we apply the Deep Learning method directly on test thumbnails, which consists in recognizing if a thumbnail belong to a class, we reach a F-score of 98%. We believe that the results can not really be improved as long as we keep the same network architecture. According to this, we focused our work on the

post and pre-processing. The reduction of performance on a frame, in most case, comes from fishes which overlap or occlude and from confusion with the background. We tried to improve the method by adding three more classes and also through the use of a well chosen overlap decision.
At the moment, the Deep Learning method gives quite good results. A possible way to treat errors is to integrate the temporal aspect in a more advanced way by implementing a fish tracking algorithm.

Acknowledgement

This work has been carried out thanks to the support of the LabEx NUMEV project (n° ANR-10-LABX-20) funded by the "Investissements d'Avenir" French Government program, managed by the French National Research Agency (ANR). We thank very much Jérôme Pasquet and Lionel Pibre for scientific discussions.

References

1. MALLET D, PELLETIER D. Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952-–2012). Fisheries Research, 2014, vol. 154, p. 44-62.
2. BOOM, Bastiaan J., HUANG, Phoenix X., BEYAN, Cigdem, et al. Long-term underwater camera surveillance for monitoring and analysis of fish populations. VAIB12, 2012.
3. FISHER, Robert B., CHEN-BURGER, Yun-Heh, GIORDANO, Daniela, et al. Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data. Springer, 2015.
4. ALSMADI, Mutasem Khalil Sari, OMAR, Khairuddin Bin, NOAH, Shahrul Azman, et al. Fish recognition based on the combination between robust feature selection, image segmentation and geometrical parameter techniques using Artificial Neural Network and Decision Tree. Journal of Computer Science Volume 6, Issue 10 Pages 1088-1094
5. ROVA, Andrew, MORI, Greg, et DILL, Lawrence M. One Fish, Two Fish, Butterfish, Trumpeter: Recognizing Fish in Underwater Video. In : Machine Vision Applications. 2007. p. 404-407.
6. SPAMPINATO, Concetto, GIORDANO, Daniela, DI SALVO, Roberto, et al. Automatic fish classification for underwater species behavior understanding. In : Proceedings of the first ACM International Workshop on Analysis and Retrieval of tracked events and motion in imagery streams, 2010. p. 45-50.
7. HEARST, Marti A. , DUMAIS, Susan T., OSMAN, Edgar, et al. Support vector machines. Intelligent Systems and their Applications, IEEE, 1998, vol. 13, no 4, p. 18-28. MLA
8. MATAI, J., KASTNER, R., CUTTER JR, G. R., et al. Automated techniques for detection and recognition of fishes using computer vision algorithms. In : NOAA Technical Memorandum NMFS-F/SPO-121, Report of the National Marine Fisheries Service Automated Image Processing Workshop, Williams K., Rooper C., Harms J., Eds., Seattle, Washington (September 4–7 2010). 2010.

9. SHIAU, Yi-Haur, LIN, Sun-In, CHEN, Yi-Hsuan, et al. Fish observation, detection, recognition and verification in the real world. In : Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV), 2012. p. 1.
10. BLANC, Katy, LINGRAND, Diane, et PRECIOSO, Frédéric. Fish species recognition from video using SVM classifier. In : Proceedings of the 3rd ACM International Workshop on Multimedia Analysis for Ecological Data. ACM, 2014. p. 1-6.
11. ZHU, Qiang, YEH, Mei-Chen, CHENG, Kwang-Ting, et al. Fast human detection using a cascade of histograms of oriented gradients. In : Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. IEEE, 2006. p. 1491-1498.
12. DALAL, Navneet et TRIGGS, Bill. Histograms of oriented gradients for human detection. In : Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005. p. 886-893.
13. PASQUET, jérôme, CHAUMONT, Marc, et SUBSOL, Gérard. Comparaison de la segmentation pixel et segmentation objet pour la détection d'objets multiples et variables dans des images. In : CORESA: COmpression et REprésentation des Signaux Audiovisuels. 2014. Reims. (in French)
14. DAS, Sukhendu, MIRNALINEE, T. T., et VARGHESE, Kuruvilla. Use of salient features for the design of a multistage framework to extract roads from high-resolution multispectral satellite images. Geoscience and Remote Sensing, IEEE Transactions on, 2011, vol. 49, no 10, p. 3906-3931.
15. SUN, Xian, WANG, Hongqi, et FU, Kun. Automatic detection of geospatial objects using taxonomic semantics. Geoscience and Remote Sensing Letters, IEEE, 2010, vol. 7, no 1, p. 23-27.
16. ZHANG, Wanceng, SUN, Xian, FU, Kun, et al. Object detection in high-resolution remote sensing images using rotation invariant parts based model. Geoscience and Remote Sensing Letters, IEEE, 2014, vol. 11, no 1, p. 74-78.
17. ZHANG, Wanceng, SUN, Xian, WANG, Hongqi, et al. A generic discriminative part-based model for geospatial object detection in optical remote sensing images. ISPRS Journal of Photogrammetry and Remote Sensing, 2015, vol. 99, p. 30-44.
18. KRIZHEVSKY, Alex, SUTSKEVER, Ilya, et HINTON, Geoffrey E. ImageNet classification with deep convolutional neural networks. In : Advances in neural information processing systems. 2012. p. 1097-1105.
19. ATKINSON, Peter M. et TATNALL, A. R. L. Introduction neural networks in remote sensing. International Journal of remote sensing, 1997, vol. 18, no 4, p. 699-709.
20. SCHMIDHUBER, Jürgen. Deep learning in neural networks: An overview. Neural Networks, 2015, vol. 61, p. 85-117.
21. LECUN, Yann, BOTTOU, Léon, BENGIO, Yoshua, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, vol. 86, no 11, p. 2278-2324.
22. SZEGEDY, Christian, LIU, Wei, JIA, Yangqing, et al. Going deeper with convolutions. In : Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. p. 1-9.
23. JOLY, Alexis, GOËAU, Hervé, GLOTIN, Hervé, et al. LifeCLEF 2015: multimedia life species identification challenges. In : Experimental IR Meets Multilinguality, Multimodality, and Interaction. Springer International Publishing, 2015. p. 462-483.

Chapitre 4

Améliorer les résultats de classification obtenues grâce à un modèle Deep Learning : Focalisation sur la construction des bases de données d'entrainement, et comparaison de l'algorithme et de l'humain.

Suite à nos premiers travaux, nous avons décidé de nous concentrer sur la tâche d'identification/classification des individus.

Nous avons entrainé un modèle d'identification sur 20 espèces particulièrement communes dans les vidéos réalisées à Mayotte. En plus des 20 classes correspondant aux 20 espèces, l'apprentissage du modèle a été renforcé grâce à 21 autres classes. Les 20 premières classes ajoutées correspondent à des classes "parties d'individus" (1 classe "partie d'individus" par espèce), et contiennent, pour chaque individu, 4 parties (2 obtenues par divison horizontale, 2 parties obtenues par division verticale). La 21$^{\text{ème}}$ classe ajoutée est la classe "fond marin" qui nous permet à la fois de renforcer l'entraînement de l'algorithme en ajoutant de la diversité à la base d'entrainement, mais qui nous permettrait aussi d'utiliser ce réseau dans une tâche de détection. Cette base a ensuite été utilisée pour entrainer une architecture profonde GoogLeNet [Szegedy et al., 2015] légèrement modifiée (dimension des images traitées, nombre de pixels de déplacement lors des *poolings* et des convolutions), implémentée sur Tensorflow [1] [Abadi et al., 2016], un environnement logiciel plus manipu-

1. https ://www.tensorflow.org/

lable que DIGITS, bien que moins adapté pour des utilisateurs non avancés. Nous avons ensuite comparé les résultats obtenus avec ce nouveau modèle par rapport au modèle utilisant uniquement les classes "espèces" pour son apprentissage. Finalement, sur un sous-échantillon de 9 espèces choisies aléatoirement, nous avons comparé les performances du modèle avec celles d'un panel de 14 personnes formées à l'identification d'espèces. Notre modèle final de classification a obtenu des scores moyens de bonne classification de 94,1% sur 18 espèces, et de 95,7% sur 9 espèces, pour un temps de classification par vignette de 0,06 seconde. Le panel humain présente un score moyen de 89,3%, pour un temps de classification par vignette de 5 secondes en moyenne. Cette étude a aussi permis de mettre en évidence les cas pour lesquels l'algorithme était plus robuste (souvent des images de très petites, donc de faible résolution, contenant des individus très pixelisés). Nous avons aussi mis en valeur les cas pour lesquels les annotateurs humains entraînés étaient plus efficace (en particulier pour les poissons en position de face ou de dos), correspondant aux cas les plus rarement présent dans la base d'apprentissage.

A Deep Learning method for accurate and fast identification of coral reef fishes in underwater images

Sébastien Villon[a,b], David Mouillot[a,g] , Marc Chaumont[b,c], Emily S. Darling[d,e], Gérard Subsol[b], Thomas Claverie[a,f], Sébastien Villéger[a]

villon@lirmm.fr

a MARBEC, University of Montpellier,CNRS, IRD, Ifremer, Montpellier, France
b LIRMM, University of Montpellier/CNRS, France
c University of Nîmes, Nîmes, France
d Department of Ecology and Evolutionary Biology, University of Toronto, Toronto, Canada
e Marine Program, Wildlife Conservation Society, Bronx, United States
f CUFR Mayotte, France
g Australian Research Council Centre of Excellence for Coral Reef Studies, James Cook University, Townsville, QLD 4811 Australia.

Abstract

Identifying and counting fish individuals on photos and videos is a crucial task to cost-effectively monitor marine biodiversity, yet it remains difficult and time-consuming. In this paper, we present a method to assist the identification of fish species on underwater images, and we compare our model performances to human ability in terms of speed and accuracy. We first tested the performance of a convolutional neural network (CNN) trained with different photographic databases while accounting for different post-processing decision rules to identify 20 fish species. Finally, we compared the performance of species identification of our best CNN model with that of humans on a test database of 1197 fish images representing nine species. The best CNN was the one trained with 900 000 images including (*i*) whole fish bodies, (*ii*) partial fish bodies and (*iii*) the environment (e.g. reef bottom or water). The rate of correct identification was 95.7%, greater than the rate of correct identification by humans (89.3%). The CNN was also able to identify fish

individuals partially hidden behind corals or behind other fish and was more effective than humans to identify fish on smallest or blurry images while humans were better to identify fish individuals in unusual positions (e.g. twisted body). On average, each identification by our best CNN using a common hardware took 0.06 seconds. Deep Learning methods can thus perform efficient fish identification on underwater images and offer promises to build-up new video-based protocols for monitoring fish biodiversity cheaply and effectively.

Keywords: marine fishes, convolutional neural network, underwater pictures, machine learning, automated identification

Introduction

Coral reefs host a massive and unique biodiversity with, for instance, more than 6,000 fish species (Mouillot et al., 2014) and provide key services to millions of people worldwide (Rogers et al., 2017). Yet, coral reefs are increasingly impacted by global warming, pollution and overfishing (Graham et al., 2011; Robinson et al., 2017; Scott and Dixson, 2016; Hughes et al., 2017; Cinner et al. 2018). The monitoring of fish biodiversity through space and time on coral reefs (Halpem et al., 2008; Jackson et al., 2001) is thus a critical challenge in marine ecology in order to better understand the dynamics of these ecosystems, predict fisheries productivity for dependent human communities, and improve conservation and management strategies to ensure their sustainability (Krueck et al., 2017; Pandolfi et al., 2003).

Most surveys of coral reef fishes are based on underwater visual censuses (UVC) carried out by scuba divers (Brock, 1954; Cinner et al., 2016, 2018; Thresher and Gunn, 1986). While non-destructive, this protocol requires the identification and enumeration of hundreds of individuals belonging to hundreds of species so it can only be performed by highly trained scientific divers while being time consuming. In addition, the accuracy of such visual-based assessments is highly dependent on conditions (depth, dive duration) and divers experience while the presence of diver biases the detection of some furtive species (Chapman and Atkinson, 1986; Harvey et al., 2004; Sale and Sharp, 1983; Watson and Harvey, 2007; Willis, 2001).

Over the last decade, underwater cameras have been increasingly used to record fish individuals on fixed videos, along belt transects (Cappo, 2003; Langlois et al., 2010; Mallet and Pelletier, 2014), or around baits to attract predators (Harvey et al., 2007; Watson et al., 2005; Willis and Babcock, 2000). Video-based surveys provide estimations of fish abundance and species diversity similar to UVC-based surveys (Pelletier et al., 2011). Video-based methods can be used to overcome the limitations of human-based surveys (depth, time underwater). They also provide a permanent record that could later be re-analyzed. However, assessing fish biodiversity and abundance from videos requires annotation by highly trained specialists and is a demanding, time-consuming and expensive task with up to several hours required to identify fish individuals per hour of video (Francour et al. 1999). There is thus an urgent need to develop new tools for automatic identification of fish individuals on photos and videos to provide accurate, efficient, repeatable and cost-effective monitoring of reef ecosystems.

Automatic and accurate identification of organisms on photos is crucial to move toward automatic

video processing. In addition, automatic identification of species on photos is especially relevant for citizen science. For instance, the application pl@ntNet (https://plantnet.org/) automatized the identification of 13,000 species of plants. For fishes, some public tools like inaturalist.org or fishpix (http://fishpix.kahaku.go.jp) offer the possibility to upload images that will be manually identified by experts. These valuable initiatives would benefit from the support of automatic identification algorithms to save time of experts.

The performance of recent methods dedicated to the automatic identification of objects on images has drastically increased over the last decade (Siddiqui et al, 2017; Lowe, 1999). However, some of these methods have been tested only on images recorded in standardized conditions, in terms of light and/or fish position (e.g. only lateral views) (Levi, 2008; Alsmadi et al, 2010). Identification of fish individuals on 'real-life' underwater images is more challenging because (*i*) color and brightness are highly variable between images and even within a given image, (*ii*) the environment is textured and has a complex 3-dimentional architecture, (*iii*) fish can be recorded in various positions and are often hidden behind other fish or corals, and (*iv*) the acquisition camera and its internal parameters can be variable.

Recently, an accurate automation of detection and identification of fish individuals has been obtained (Shortis et al., 2016) using machine-learning methods such as support vectors machines (Blanc et al. 2014), nearest neighbor classifiers (Levi, 2008), discriminant analysis classifiers (Spampinato et al., 2010) or Deep Learning (Li et al., 2015). The latest competitions (Joly et al., 2106) and comparisons (Villon et al., 2016) show that Deep Learning based methods, which are a type of neural network combining simultaneously automatic image descriptor and descriptor classification, tend to achieve the highest performance, particularly convolutional neural network (CNN) that add deep layers to classical neural networks (Lecun et al., 2015).

However, the accuracy of CNN methods is highly dependent on the extent and the quality of data used during the training phase, i.e. the set of images annotated by experts for all classes to identify. The effects of the extent of the training database (i.e. the number of images per class) and associated post-processing decision rules on the performance of the whole identification process remain untested. Since real-life videos of coral reef fishes and thus images extracted from those videos are highly diverse in terms of surrounding conditions (environment, light, contrast) and fish positions, the performance of identification methods must be carefully tested using an independent dataset to assess its robustness over changing conditions.

Furthermore, the performance of models should be compared to the performance of humans to determine whether machine-based assessment of fish biodiversity provides an advantage over traditional human processing of images (Matabos et al., 2017). Here we tested the performance of 4 models, built with the same CNN architecture, for automatic identification of fish species on coral reefs. Specifically, we assessed the effect of several training image datasets and several decision rules, with a particular focus to identify fish partially hidden behind the coral habitat. We then compared the performances of the best CNN models to those of humans.

Methods

Image acquisition for training and testing CNN models

We used GoPro Hero3+ black and GoPro Hero4+ black cameras to record videos at 30 fps over 50 reef sites around the Mayotte island (Mozambique Channel, Western Indian Ocean) including fringing and barrier reefs, and at depth from 1 to 25m. Videos were recorded from April to November 2015. Recording conditions varied between sites and days, especially in term of light and environment (i.e. proportion of hard and soft corals, sand and water visible). All videos were recorded with a resolution of 1280x720 (HD) and 1920x1080 pixels (full HD) with default settings for color temperature and exposure (i.e. no use of protune or automatic color balance adjustment). For all recordings, the cameras remained stationary and no artificial light or filter were used. We recorded 116 videos representing a total of 25 hours.

For all videos, 5 frames per second were extracted leading to a database of 450,000 frames. Fish individuals were delineated and identified by undergraduate, master degree students and PhD students in marine biology trained for fish identification on videos with the support of identification keys and under the supervision of experts (Froese and Pauly, 2000; Taquet and Diringer, 2007). Each annotation consisted in drawing a rectangle bounding box around a single fish individual, including only its very close context as illustrated on Fig.1.a, and associating a label (i.e. species name) to this individual. We call those specific images “thumbnails”.

The criteria for the annotation were:

1) Annotate a fish only if there is no more than 10% of its surface covered by another object (fish, coral, or substrate).

2) Annotate a fish only if it can be identified at the species level in the frame (i.e. independently

from previous or next frames where the same fish could have a better position for identification).
3) Annotate a fish only if its apparent size is larger than 3,000 squared pixels, i.e. ignoring fish individuals too far from the camera.
4) Annotate images from different habitats and depths to represent a broad range of light conditions and environment, and target at least 1,200 thumbnails per species.

We did not consider thumbnails of individuals in positions where they are hard to identify (such as fish seen from front) since they would bring more noise than relevant information for the algortihm as the discriminating parts of the fish are hidden (specific color pattern, marks, etc). We did not process the image with background subtraction for 2 reasons:
1) We did assume that in our case the context helps to identify fish species, as some species tend to be associated with some particular environment such as Amphiprion in sea anemone, Chromis viridis on Acroporas, Caesionidae in plain water etc...
2) We wanted our process to be used on full images. In such context, separating fish individuals from their background would be either manual or not reliable.

This annotation procedure yielded a training dataset (T0) with 44,625 annotated fish thumbnails belonging to 20 species (Table 1). The 20 species present in the training dataset represent the most common species appearing in the videos and belong to 12 families among the most diverse and abundant on coral reefs worldwide (e.g. Pomacentridae, Acanthuridae, Chaetodontidae, Labridae). Models were then tested using a set of images independent from the ones used for the training phase to ensure a cross validation procedure and that model performance reflects real-life study case. More specifically, the test dataset was built using 6 videos recorded in contexts different from those of videos used for training (i.e. sites or days not included in the training database). Annotations of these videos were made like the training dataset except that it included fish individuals partially hidden by other fish or by corals as well as fish individuals viewed from front or back (their identity being checked using when necessary previous or next frames). As our goal is to identify fish species on images and photos, the test without any filter allows to assess to which extent our algorithm is performing to help users to take a picture good enough for fish identification.

We obtained a test dataset of 4,405 annotated fish thumbnails belonging to 18 out of the 20 species present in the training dataset (Table S3). We then randomly selected a subset of 1,197 fish thumbnails belonging to 9 species to compare the performance of humans vs. obtained models (Table S3).

Deep-learning algorithm

We used a convolutional neural network (CNN) architecture to build a fish identification model (Schmidhuber, 2015). CNNs are a class of deep learning algorithms used to analyze data and particularly to classify objects from images (Krizhevsky et al., 2012).

CNNs are made of layers of interconnected neurons and each neuron includes a 'convolutional kernel' that computes a set of mathematical operations (defined by 'weights') on the matrices of values describing the image (i.e. values for each color channel for each pixel).
Convolutional features are combinations of pixel values that encode information about target classes. Low level features can detect edges or color patterns, while, high level features might differentiate different fish shapes.

This process yielded 'feature maps', i.e. a vector describing image characteristics (shapes, colors, statistical information of the image).

The main difference between CNNs and other classifiers is that CNNs build the "feature extractors" (convolutions in the case of CNN) and the classifier conjointly.
Then the last layer of the network classifies those feature maps with a soft-max method and gives as output scores corresponding to the "probability" that each image belongs to each of the learned classes (Lecun et al., 2015). More precisely, the training phase of the network consists in iteratively modifying the weights of the convolutional kernels (hence features maps) to optimize the classification score of all classes.

We used a GoogLeNet architecture as it was the winner of the 2015 competition imageNet (Szegedy et al., 2015), an identification challenge on 1,000 different classes. This CNN is composed of 22 layers. It uses inception modules. Inception modules allow the network to use convolutions of different sizes (1*1, 3*3 and 5*5 pixels) and to weight each of these convolutions. This network could thus account more or less strongly for the context of each pixel, which increases the range of possibilities to improve its performance during the training.

A link to a depository with architecture details is given at the end of references. We stopped the network training after 70 epochs (i.e. a complete scope of the dataset where each image is used only once), to prevent overfitting. We used a learning rate of 10^{-5}, an exponential learning decay with a Gamma of 0.95, a dropout of 50% and an Adam Solver type as learning parameters. Those are

classic hyper-parameters for a fast convergence of the network without over-fitting (Srivastava, 2014). The weight initialization is also classic with a random Gaussian initialization. The training lasted 8 days on our configuration; we trained and ran our code on a computer with 64GB of RAM, an i7 3.50GHz CPU and a Titan X GPU card for 900,000 images.

We used at least 2200 thumbnails per fish species class, and batches of 16 images to train our network. We ran this architecture on Caffe (Jia et al, 2014). To focus on the impact of the training data, we used the same CNN architecture for our training and test procedures.

Building the training datasets

Using the raw training dataset of 20 fish species (Table S1) we built 4 different datasets to assess the influence of the dataset building on classification results (Table S2).

The first training dataset T1 contained raw fish thumbnails (T0) and their respective mirror images. More precisely, we doubled the number of thumbnails per fish individual by flipping each thumbnail with respect to the vertical axis. Such a procedure homogenizes the proportion of left-oriented and right-oriented individuals in the database and we hypothesize it could improve the average identification rate since fish individuals are seen in all positions.

The second training dataset T2 contained fish thumbnails from T1 plus “part of fish” thumbnails. Thumbnails of this class were obtained by splitting each thumbnail of T0 into 4 parts: upper part, lower part, right part, and left part as shown on Fig.1. b. We hypothesized that this class can prevent from misidentification of partially hidden individuals. For instance, if a black and white fish is partially hidden so that only its dark part is visible it would likely be confounded with a full dark fish.

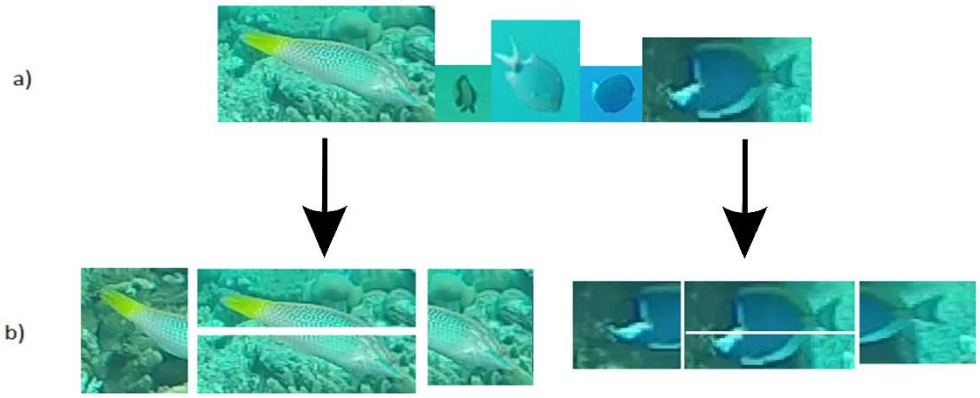
The third training dataset T3 contained fish thumbnails from T2 plus thumbnails of a single class “Environment”. Environment thumbnails were extracted at random in portion of frames where no fish was detected. We hypothesized that such a procedure can help distinguishing between fish species given the high diversity of environments present around them, i.e. allowing CNN models to find more efficiently features discriminating fishes whatever the background around them.

The fourth training dataset T4 contained thumbnails from T3 minus the “part of fish”, which is replaced by 20 classes “part of species” obtained by splitting thumbnails from each species. The difference between T3 and T4 was that T3 contained only one global class “part of fish” whereas T4

contained as many “part of species” classes as there were “fish” species.

Figure 1: Thumbnails samples.

a) Examples of thumbnails of whole fish individuals from the training database and b) examples of thumbnails extracted from whole fish picture to build “part of fish” and “part of species” classes.



Testing the performance of models

We first compared the performance of the 4 models trained using each of the 4 training datasets. In addition, we tested the performance of models after correcting their raw outputs using two *a posteriori* decision rules. First, since the networks trained with T2, T3 or T4 are likely to recognize environment samples with a high confidence score (over 99%) they could thus classify some fish as an environment class (i.e. false positive). We therefore defined a decision rule (r1): when the first proposition of the network was ‘environment’ with a confidence lower than 99% we provide, as final output, the fish class with the highest probability.

Similarly, as “part of species” classes present in T4 were just a methodological choice to improve model performance (and hence were absent from the test database), we defined a second decision rule (r2): when the result given by the network is “part of species X”, we provide, as final output, “species X”.

We then compared the performance of the best model with the performance of humans, in terms of accuracy and time needed to identify fish thumbnails. This experiment aimed to compare the results obtained by humans to those obtained by the CNN using a fair method. This means that during the comparison procedure both CNN and humans were shown thumbnails without any contextual information (there was no general view of the scene), and the thumbnails were never seen before the test procedure. The procedure could even be slightly in favor of humans because they knew that there were only 9 species to classify, whereas the CNN worked from the 21 species learned and misclassification could occur with a higher probability.

Our goal was to allow humans to identify species as fast as possible in this particular context. For this purpose, we developed an online survey tool operating in Chrome web browser which allowed users to easily and quickly identify a fish on a picture displayed at the center of the window by either writing the name of the species (with auto-completion) or to select it from a list. A “help” sheet showing a reference picture of the fish species to identify was available in the same window (Fig. S1). Once a user selected a species, time to perform the identification was saved and a new randomly chosen fish picture was displayed.

This comparison was performed on 1197 randomly chosen thumbnails of only 9 species present in the test thumbnail dataset (Table S3) to ease the test for humans. The test lasted 20 minutes with the help of 10 undergraduate students, 2 Master Degree and 2 PhD student in biology from the University of Montpellier who were previously trained to identify these fish species. Such a short test duration for humans reduces tiredness that could decrease identification accuracy and rapidity. We then compared the answers to the ground truth (i.e. identification made by experts in fish taxonomy) and computed the time needed to perform each identification. We finally compared correct identification rate and time per fish individual between humans and the best CNN model.

Results

Influence of the training database and of post-processing on model performance

The 4 CNN models obtained with 4 different datasets (T1, T2, T3, T4) had similar mean identification success rate, close to 87% (Table 1). However, there were marked differences in correct identification rate between models for several species. For instance, *Dascyllus carneus* was correctly identified in only 4% of the cases by model trained with only whole fish thumbnails (T1) while it was correctly identified in more than 90% of cases by the three other models. Conversely, *Pomacentus sulfureus* was more often correctly identified by the models trained with T1 than by models trained with environment thumbnails (T3 and T4).

Table 1: Raw success rate (%) of the 4 CNN models trained with different thumbnails datasets for identifying 18 fish species. See details about training databases in Table S2.

Species	Only whole fish (T1)	Whole fish and part of fish (T2)	Whole fish, environment and part of fish (T3)	Whole fish, environment and part of species (T4)
Abudefduf sparoides	80.8	94.9	85.8	82.8
Abudefduf vaigiensis	94.5	89.0	89.0	80.0
Chaetodon trifascialis	94.7	90.4	91.0	85.1
Chromis weberi	98.8	96.6	92.9	98.8
Dascyllus carneus	4.0	91.5	92.3	91.5
Monotaxis grandoculis	90.0	68.0	77.7	79.1
Myripristis botche	100	80.0	75.0	95.0
Naso elegans	96.2	92.4	89.7	95.1
Naso vlamingii	92.6	95.3	89.1	95.8
Nemateleotris magnifica	100	98.2	99.5	99.1
Odonus niger	79.5	91.4	92.6	81.8
Plectroglyphidodon lacrymatus	100	100	74.2	94.0
Pomacentrus sulfureus	97.8	67.6	82.5	73.8
Pterocaesio tile	100	100	100	99.5
Pygoplytes diacanthus	84.2	91.5	84.2	86.8
Thalassoma hardwicke	83.9	82.7	88.0	87.3
Zanclus cornutus	93.3	84.3	86.4	89.0
Zebrasoma scopas	89.0	88.8	88.8	92.7
Mean identification success rate	87.6	87.9	87.7	86.9

Post-processing raw outputs of the model T4 following decision rule r1 (i.e. environment not considered as a correct result), improved correct identification rate from 86.9 to 90.2% (Table 2). Adding decision rule r2 (i.e. identification of a part of a species considered as a correct answer) increased this success rate to 94.1% (Table 2). Hence, post-processing raw outputs of the model trained with the most complete dataset provided the best identification rate. Among the 18 species, success rate ranged from 85.2 to 100%, with only 3 species being correctly identified in less than 90% of cases and 9 species being correctly identified in more than 95% of cases, including 3 with a correct identification rate >99%.

Confusions between 2 fish species were lower than 4% (Table 3). Confusion between a fish and the

environment was common when no post-processing was applied with for instance up to 20.9% of *Pomacentrus sulfureus* individuals misidentified as environment (Tables S4, S5). However, applying decision rule r1 decreased this error rate to less than 4% (Table 3).

Table 2:

Success rate (%) of 3 CNN models for identifying 18 fish species. First column presents accuracy based on raw output of a deep-learning model trained with thumbnails of whole fish, part of species and environment (as last column of Table 2). Second column presents accuracy after applying a decision rule 'r1' keeping most likely fish class if 'environment' was the most likely class. Third column presents results after applying decision rule 'r1' plus decision rule 'r2': "part of species X" is equivalent to "species X". Numbers are percentages of correct fish identification.

Species	Raw output	Decision Rule r1	Decision Rules r1 and r2
Abudefduf sparoides	82	88	91.9
Abudefduf vaigiensis	80	89	98
Chaetodon trifascialis	85.1	87.8	91.5
Chromis weberi	98.8	98.8	99.2
Dascyllus carneus	91.5	91.5	91.5
Monotaxis grandoculis	79.1	83.3	86.1
Myripristis botche	95	95	95
Naso elegans	95.1	96.7	97.8
Naso vlamingii	95.8	96	96
Nemateleotris magnifica	99.1	100	100
Odonus niger	81.8	81.8	85.2
Plectroglyphidodon lacrymatus	94	94	96
Pomacentrus sulfureus	73.7	78.1	87.9
Pterocaesio tile	99.5	100	100

Pygoplytes diacanthus	86.8	89.4	92.1
Thalassoma hardwicke	87.3	89.6	94.2
Zanclus cornutus	89	95.3	98.4
Zebrasoma scopas	92.7	92.7	92.7
Average success rate	86.9	90.2	94.1

Table 3. Performance and confusion rates of CNN model for 9 fish species.

The CNN was trained with dataset T4 (see Table 1), including thumbnails of whole fish, part of species and environment. Raw CNN outputs were post-processed with following decision rules:

'r1': If the highest probability is lower than 99% and is for class "environment" then the fish class with the second highest probability is kept.

'r2': Outputs "part of species X" are considered as equivalent to "species X" (i.e. the scores of *A. sparoides* and *part of A. sparoides* were merged).

Columns indicate the species to classify, and rows indicate the results (most probable species) given by the model (i.e. percentages on the diagonal indicate success rate). Only values over 1% are shown. Full names of species are in Table 1

Species	*A.sparoides*	*A. vaigiensis*	*C. Trifascialis*	*N. elegans*	*P. sulfureus*	*P. diacanthus*	*T. hardwicke*	*Z. cornutus*	*Z. scopas*
A.sparoides	91.9						1.3		
A. vaigiensis	1.1	98.2							
C. Trifascialis			91.5				1.0		
C. Weberi	2.2						1.1	1.5	
D. caruleus									3.9
N. elegans				97.8					
P. sulfureus	1.0	1.8	1.0		87.9	2.5			
P. diacanthus					3.8	92.1			
P. lacrymatus						2.6			
T. Hardwicke	2.0		1.5				94.2		
Z. cornutus	1.0							98.5	
Z. scopas									92.7
Environment					3.6	2.6			1.0

Performance of CNN models vs. humans

On average, each human identified 270 fish thumbnails during the 20-minute test. Mean rate of correct classification for humans was of 89.3% with a standard deviation of 6% (Table 4). Rate of correct classification achieved by the best model on the same thumbnails was of 95.7% with a standard deviation of 3.3%. Correct classification rate by the best model ranged from 88.2% (*Abudefduf sparoides*) to 98.2% (*Abudefduf vaigiensis*). For only one species (*Zanclus cornutus*), the best model had a lower performance than humans but both were higher than 97%. The mean time needed to identify a fish by humans was 5 seconds, with the fastest answer given in 2 seconds and the longest in 9 seconds. On average, each classification by our final model took 0.06 seconds with hardware detailed above.

When tested against humans using a challenge with only 9 potential species, the network was more effective on smaller or blurrier thumbnails, while humans were better to recognize unusual positions (Fig. 2). There were only 2% of fish individuals which were neither identified by humans nor by the network (Fig. 2).

However, experts with more than 10 years of experience in the field may have outperformed the CNN model in terms of correct identification particularly for hidden or unusually positioned fish.

Table 4. Accuracy (success rate in %) of fish identification by humans and by the best CNN model for 9 species.

The model was trained using thumbnails of whole fish, part of fish species and environment (T?). Raw outputs were post-processed applying two decision rules: (r1) keeping most likely fish class if "environment" was the most likely class, and (r2) considering "part of species X" equivalent to "species X".

Species	Number of thumbnails tested	Deep-learning model	Humans
Abudefduf sparoides	88	93.4	87.7
Abudefduf vaigiensis	47	97.3	84.7
Chaetodon trifascialis	149	95.1	89.4
Naso elegans	165	98.4	94.8
Pomacentrus sulfureus	443	97.9	93.2
Pygoplites diacanthus	35	90.4	77.4
Thalassoma hardwicke	73	96	91
Zanclus cornutus	53	97.1	97.8
Zebrasoma scopas	144	96.2	88.3
Average success rate	1197	95.7	89.3

Discussion

Assessing the performance of the same CNN trained with four different datasets demonstrates that correct identification rates were all close to 87% . Thus, a training dataset made of more than 1300 thumbnails of each species could yield a success rate similar to the ones obtained in image identification challenges in more controlled conditions (Siddiqui et al., 2017). Beyond their number, thumbnails of each species used to train the network were extracted from different videos and different sites to include as many orientations of fish as possible and to embrace a strong environmental variability in terms of light, colors and depth. However, our best CNN model may perform more poorly with a broader range of species across other locations and environments. Our 18 species belong to 12 different families so are likely to differ in shape or color. With much more

congeneric species these differences would make the identification much more challenging.

Despite a similar mean success rate, the performance of the four models differed markedly for some species. Ten out of the 18 species were more often correctly identified when CNN models were trained using thumbnails of part of fish or environment, and eight other species were better identified by the model trained with only whole fish picture. Additionally, some species were often misidentified as environment (Table S5), even if the probability of this class was lower than 99%.Such confusion could be explained by the fact that some small species are always close to corals and of similar colors, e.g. the yellow benthic fish *Pomacentrus sulfureus.* Similarly, for the small *Dascyllus carneus* case, which is often misclassified with almost all fish species when background was not included in the training dataset, the addition of environment thumbnails certainly helps the network to focus on features unique to the fish body rather than to its surrounding.

We demonstrate that the best results were obtained after applying two *a posteriori* decision rules on raw outputs from the neural network trained with the most complete set of thumbnails. This model reached a success rate of 94.1% for the 18 species tested, with only 3 species being correctly identified in less than 90% of cases. Therefore, training a neural network with thumbnails from surrounding environment and thumbnails of part of each fish species is important to reach a high correct identification rate in real-life cases. The class "Environment" adds versatility to the training and hence helps the network to select features that are robust to the context around fish. Including classes "part of species" allows the network to classify correctly individuals partially hidden by other fish or corals. Such situations were common in the test dataset as illustrated by the fact that up to 9% of individuals of *Abudefduf vaigiensis* were classified as "part of *A. vaigensis*" rather than "whole *A. vaigensis*".

The success rate of the best model is similar to that of the model of Siddiqui et al. (2016) which reached a success rate of 94.3% on 16 species. This latter model was trained on a much smaller training dataset of 1309 thumbnails than our model (> 900 000 thumbnails). However, Siddiqui's model was designed to identify fish on videos recorded in partially controlled conditions (i.e. fish swimming close to a baited camera) while in our case we tested the ability of the model to identify fish partially hidden by corals as well as shot in all positions and orientations. The few misidentifications by our best model mostly occurred when only the face or back of fish was visible. Such an issue could be easily circumvented in practice when analyzing videos because it is likely that each fish will be seen from the side on at least one frame (out of the 25 frames recorded

per second by most cameras).

Figure 2: Samples of thumbnails recognized by the CNN model and not recognized by humans (a), samples of thumbnails recognized by humans and not recognized by the CNN model (b) and sample of thumbnails misidentified by both humans and the CNN model (c).

Identification methods such as the ones presented here pave the way towards new ecological applications. First, such methods can work continuously and their performance is constant through time and hence reproducible, contrary to human experts who work discontinuously and are likely to perform differently through time. Given the high rate of correct identifications, the best model could be used to pre-process a massive number of thumbnails: up to 1 million thumbnails per day. Furthermore, additional post processing procedures could be used. For example, under a certain threshold (e.g. 98% certainty), human experts could be asked to check the thumbnails identified by CNN models. Such a two-step workflow would ensure a very high identification rate while saving time of experts in fish taxonomy who will not have to identify “obvious” fish that can be accuratly identified by models. In addition, identification methods could also be used as a tool to initiate

citizen science programs, for example where divers upload images of fish and obtain the most likely taxonomic identification from a CNN model. Therefore, the continued development of these identification tools could potentially offer benefits for both professional scientists collecting massive raw data from the field, and for citizens to improve their awareness and knowledge about biodiversity (e.g. Bradley et al., 2017)

The method tested here is one step towards the identification of hundreds or thousands of fish species that occur on coral reefs (Kulbicki et al., 2013). Since the performance of CNNs is known to increase with the number of classes (i.e. the 1000 classes of ImageNet) (Krizhevsky et al., 2012), there is no theoretical limit to such upscaling, the main challenge being to increase the size of the training dataset and the computer power. However, the identification of rare species will remain challenge given the difficulty to collect enough thumbnails of such species in different conditions to train the model. Future work is also needed to broaden the range of conditions where the model is efficient for most of species. In this paper, we considered only fixed videos recorded between 1m and 25m for both our training and testing datasets. It would relevant to include deeper videos as well as videos recorded with other protocols (e.g. baited remote underwater videos, transects).

Ultimately, the goal of automatic identification is not only to classify fish into species, but also to localize and count them, and estimate their size (body length) on videos. The detection task in underwater videos remains challenging as the context is particularly complex. Towards this aim, including “environment” and “part of species” classes in the training of models will enhance the accurate detection of fish inidividuals partially hidden behind corals or other fish, for instance using a sliding windows approach over a video frame. We could also associate a classifier with a detector (Weinstein et al., 2015, Price Tack et al, 2016). Such algorithms focus on the detection of objects of interest (such as fish individuals) in images. Ultimately, deep-learning based methods could help marine ecologists to develop new video-based protocols for a massive monitoring of increasingly imperiled reef fish biodiversity, in the same way as next-generation sequencing of DNA has revolutionized several research domains including biodiversity monitoring (Deiner et al., 2017).

Acknowledgement

We want to thank the CEMEB Label of Excellency of Montpellier for funding this work.
We want to thank the reviewers of our work for their insightful remarks which help us for this work.

References

Alsmadi, M. K., Omar, K. B., Noah, S. A., & Almarashdeh, I. (2010). Fish recognition based on robust features extraction from size and shape measurements using neural network. Journal of Computer Science, 6(10), 1088.

Blanc, K., Lingrand, D., & Precioso, F. (2014, November). Fish species recognition from video using SVM classifier. In Proceedings of the 3rd ACM International Workshop on Multimedia Analysis for Ecological Data (pp. 1-6). ACM.

Bradley M. Norman, Jason A. Holmberg, Zaven Arzoumanian, Samantha D. Reynolds, Rory P. Wilson, Dani Rob, Simon J. Pierce, Adrian C. Gleiss, Rafael de la Parra, Beatriz Galvan, Deni Ramirez-Macias, David Robinson, Steve Fox, Rachel Graham, David Rowat, Matthew Potenski, Marie Levine, Jennifer A. Mckinney, Eric Hoffmayer, Alistair D. M. Dove, Robert Hueter, Alessandro Ponzo, Gonzalo Araujo, Elson Aca, David David, Richard Rees, Alan Duncan, Christoph A. Rohner, Clare E. M. Prebble, Alex Hearn, David Acuna, Michael L. Berumen, Abraham Vázquez, Jonathan Green, Steffen S. Bach, Jennifer V. Schmidt, Stephen J. Beatty, David L. Morgan; Undersea Constellations: The Global Biology of an Endangered Marine Megavertebrate Further Informed through Citizen Science, 2017/11/29, BioScience, bix127.

Brock, V. E. (1954). A preliminary report on a method of estimating reef fish populations. *The Journal of Wildlife Management*, *18*(3), 297-308.

Cappo, M., Harvey, E., Malcolm, H., & Speare, P. (2003). Potential of video techniques to monitor diversity, abundance and size of fish in studies of marine protected areas. Aquatic Protected Areas-what works best and how do we know, 455-464.

Chapman, C.J. & Atkinson, R.J.A. (1986). Fish behaviour in relation to divers. Prog Underw Sci, 11, l–14.

Cinner, J. E., Huchery, C., MacNeil, M. A., Graham, N. A., McClanahan, T. R., Maina, J., ... & Allison, E. H. (2016). Bright spots among the world's coral reefs. Nature, 535(7612), 416.

Cinner, J. E., Maire, E., Huchery, C., MacNeil, M. A., Graham, N. A., Mora, C., ... & D'Agata, S. (2018). Gravity of human impacts mediates coral reef conservation gains. *Proceedings of the National Academy of Sciences*, 201708001.

Deiner, K., Bik, H. M., Mächler, E., Seymour, M., Lacoursière-Roussel, A., Altermatt, F., ... & Pfrender, M. E. (2017). Environmental DNA metabarcoding: Transforming how we survey animal and plant communities. Molecular ecology,26(21), 5872-5895.

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

Graham, N. A., Chabanet, P., Evans, R. D., Jennings, S., Letourneur, Y., Aaron MacNeil, M., ... & Wilson, S. K. (2011). Extinction vulnerability of coral reef fishes. Ecology Letters, 14(4), 341-348.

Halpern, B. S., Walbridge, S., Selkoe, K. A., Kappel, C. V., Micheli, F., D'agrosa, C., ... & Fujita, R. (2008). A global map of human impact on marine ecosystems. Science, 319(5865), 948-952.

Harvey, E., Fletcher, D., Shortis, M. R., & Kendrick, G. A. (2004). A comparison of underwater visual distance estimates made by scuba divers and a stereo-video system: implications for underwater visual census of reef fish abundance. Marine and Freshwater Research, 55(6), 573-580.

Harvey, E.S., Cappo, M., Butler, J., Hall, N. & Kendrick, G. (2007). Bait attraction affects the performance of remote underwater video stations in assessment of demersal fish community structure. Mar. Ecol. Prog. Ser., 350, 245–254

Hughes, T. P., Barnes, M. L., Bellwood, D. R., Cinner, J. E., Cumming, G. S., Jackson, J. B., ... & Palumbi, S. R. (2017). Coral reefs in the Anthropocene. *Nature*, *546*(7656), 82.

Jackson, J. B., Kirby, M. X., Berger, W. H., Bjorndal, K. A., Botsford, L. W., Bourque, B. J., ... & Hughes, T. P. (2001). Historical overfishing and the recent collapse of coastal ecosystems. Science, 293(5530), 629-637.

Javed, O., & Shah, M. (2002, May). Tracking and object classification for automated surveillance. In European Conference on Computer Vision (pp. 343-357). Springer, Berlin, Heidelberg.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... & Darrell, T. (2014, November). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia* (pp. 675-678). ACM.

Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W. P., ... & Müller, H. (2016, September). LifeCLEF 2016: multimedia life species identification challenges. In International Conference of the Cross-Language Evaluation Forum for European Languages (pp. 286-310). Springer International Publishing.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. InAdvances in neural information processing systems (pp. 1097-1105).

Krueck, N. C., Ahmadia, G. N., Possingham, H. P., Riginos, C., Treml, E. A., & Mumby, P. J. (2017). Marine reserve targets to sustain and rebuild unregulated fisheries. PLoS biology, 15(1), e2000537.

Kulbicki, M., Parravicini, V., Bellwood, D. R., Arias-Gonzàlez, E., Chabanet, P., Floeter, S. R., ... & Mouillot, D. (2013). Global biogeography of reef fishes: a hierarchical quantitative delineation of regions.PLoS One,8(12), e81847.

Langlois, T.J., Harvey, E.S., Fitzpatrick, B., Meeuwig, J., Shedrawi, G. & Watson, D. (2010). Cost-efficient sampling of fish assemblages: comparison of baited video sta=tions and diver video transects. Aquat. Biol., 9, 155–168.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature,521(7553), 436.

Levi, D. M. (2008). Crowding—An essential bottleneck for object recognition: A mini-review. Vision research, 48(5), 635-654.

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.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on (Vol. 2, pp. 1150-1157). Ieee.

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.

Matabos, M., Hoeberechts, M., Doya, C., Aguzzi, J., Nephin, J., Reimchen, T. E., ... & Fernandez-Arcaya, U. (2017). Expert, Crowd, Students or Algorithm: who holds the key to deep-sea imagery 'big data'processing?.Methods in Ecology and Evolution.

Mouillot, D., Villéger, S., Parravicini, V., Kulbicki, M., Arias-González, J. E., Bender, M., ... & Bellwood, D. R. (2014). Functional over-redundancy and high functional vulnerability in global fish faunas on tropical reefs. Proceedings of the National Academy of Sciences, 111(38), 13757-13762.

Pandolfi, J. M., Bradbury, R. H., Sala, E., Hughes, T. P., Bjorndal, K. A., Cooke, R. G., & Warner, R. R. (2003). Global trajectories of the long-term decline of coral reef ecosystems. Science, 301(5635), 955-958.

Pelletier, D., Leleu, K., Mou-Tham, G., Guillemot, N. & Chabanet, P. (1/2011). Comparison of visual census and high definition video transects for monitoring coral reef fish assemblages. Fish. Res., 107, 84–93.

Price Tack, J. L. et al. 2016. AnimalFinder: A semi-automated system for animal detection in time-lapse camera trap images. - Ecol. Inform. 36: 145–151.

Robinson, J. P., Williams, I. D., Edwards, A. M., McPherson, J., Yeager, L., Vigliola, L., & Baum, J. K. (2017). Fishing degrades size structure of coral reef fish communities. Global change biology, 23(3), 1009-1022.

Rogers, A., Blanchard, J. L., & Mumby, P. J. (2017). Fisheries productivity under progressive coral reef degradation. Journal of Applied Ecology.

Sale, P. F., & Sharp, B. J. (1983). Correction for bias in visual transect censuses of coral reef fishes. Coral reefs, 2(1), 37-42.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview.Neural networks,61, 85-117

Scott, A., & Dixson, D. L. (2016, May). Reef fishes can recognize bleached habitat during settlement: sea anemone bleaching alters anemonefish host selection. In Proc. R. Soc. B (Vol. 283, No. 1831, p. 20152694). The Royal Society

Shortis, M. R., Ravanbakhsh, M., Shafait, F., & Mian, A. (2016). Progress in the automated identification, measurement, and counting of fish in underwater image sequences. Marine Technology Society Journal, 50(1), 4-16.

Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., & Harvey, E. S. (2017). Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. ICES Journal of Marine Science, fsx109.

Spampinato, C., Giordano, D., Di Salvo, R., Chen-Burger, Y. H. J., Fisher, R. B., & Nadarajan, G. (2010, October). Automatic fish classification for underwater species behavior understanding. In Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams (pp. 45-50). ACM.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929-1958.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D. & Rabinovich, A. (2015). Going deeper with convolutions. InProceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

Taquet, M., & Diringer, A. (2007). *Poissons de l'océan Indien et de la mer Rouge*. Editions Quae.

Thresher, R. E., & Gunn, J. S. (1986). Comparative analysis of visual census techniques for highly mobile, reef-associated piscivores (Carangidae). *Environmental Biology of Fishes*, *17*(2), 93-116.

Villon, S., Chaumont, M., Subsol, G., Villéger, S., Claverie, T., & Mouillot, D. (2016, October). Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between Deep Learning and HOG+ SVM methods. In International Conference on Advanced Concepts for Intelligent Vision Systems (pp. 160-171). Springer International Publishing.

Watson, D.L., Harvey, E.S., Anderson, M.J. & Kendrick, G.A. (12/2005). A comparison of temperate reef fish assemblages recorded by three underwater stereo-video techniques. Mar. Biol., 148, 415–425

Watson, D.L. & Harvey, E.S. (2007). Behaviour of temperate and sub-tropical reef fishes towards a stationary SCUBA diver. Mar. Freshw. Behav. Physiol., 40, 85–103

Weinstein, B. G. 2015. MotionMeerkat: Integrating motion video detection and ecological

monitoring (S Dray, Ed.). - Methods Ecol. Evol. 6: 357–362.

Willis, T. J., & Babcock, R. C. (2000). A baited underwater video system for the determination of relative density of carnivorous reef fish. Marine and Freshwater research, 51(8), 755-763.

Willis, T. J. (2001). Visual census methods underestimate density and diversity of cryptic reef fishes. Journal of Fish Biology, 59(5), 1408-1411.

https://github.com/NVIDIA/DIGITS/blob/master/digits/standard-networks/caffe/googlenet.prototxt 2018

SUPPLEMENTARY TABLES

Table S1. Raw fish thumbnails training dataset

Classes	Number of thumbnails
Abudefduf sparoides	1241
Abudefduf vaigiensis	5674
Chaetodon trifascialis	1456
Chromis weberi	3576
Dascyllus carneus	2276
Lutjanus kasmira	1652
Monotaxis grandoculis	1239
Myripristis botche	1264
Naso elegans	2068
Mulloidichtys vanicolensis	1264
Naso vlamingii	1789
Nemateleotris magnifica	1189
Odonus niger	2986
Plectroglyphidodon lacrymatus	652
Pomacentrus sulfureus	5176
Preocaesio tile	3088
Pygoplytes diacanthus	1106
Thalassoma hardwicke	1579
Zanclus cornutus	1886
Zebrasoma scopas	1835

Table S2. The four thumbnails datasets used to train the four models, with for each the number of thumbnails per class in the training datasets, with class “environment” gathering thumbnails of water and substrate (sand, corals) while “Part of fish” gathers all thumbnails of half of a fish individual and the "part of species" classes contain thumbnails of half of individuals for each species.

Species	Only whole fish (T1)	Whole fish + “Part of fish” (T2)	Whole fish + Environment + “Part of fish” (T3)	Whole fish + environment + “Part of species” (T4)
Abudefduf sparoides	2482	2482	2482	2482
Abudefduf vaigiensis	11328	11328	11328	11328
Chaetodon trifascialis	2912	2912	2912	2912
Chromis weberi	7152	7152	7152	7152
Dascyllus carneus	4552	4552	4552	4552
Lutjanus kasmira	3300	3300	3300	3300
Monotaxis grandoculis	2478	2478	2478	2478
Mulloidichtys vanicolensis	2528	2528	2528	2528
Myripristis botche	2528	2528	2528	2528
Naso elegans	4138	4138	4138	4138
Naso vlamingii	3578	3578	3578	3578
Nemateleotris magnifica	2378	2378	2378	2378
Odonus niger	5972	5972	5972	5972
Plectroglyphidodon lacrymatus	1304	1304	1304	1304
Pomacentrus sulfureus	10352	10352	10352	10352
Preocaesio tile	6176	6176	6176	6176
Pygoplytes diacanthus	2212	2212	2212	2212
Thalassoma hardwicke	3158	3158	3158	3158
Zanclus cornutus	3772	3772	3772	3772
Zebrasoma scopas	3670	3670	3670	3670
Part of *Abudefduf sparoides*				4964
Part of *Abudefduf vaigiensis*				22656
Part of *Chaetodon trifascialis*				5824
Part of *Naso elegans*				14304
Part of *Pomacentrus sulfureus*				20704
Part of *Lutjanus kasmira*				6600
Part of *Pygoplites diacanthus*				4424
Part of *Thalassoma hardwicke*				6316
Part of *Zanclus cornutus*				7544
Part of *Zebrasoma scopas*				7340
Part of				14034

Chromis weberi			
Part of			4956
Monotaxis grandoculis			
Part of			1304
Plectroglyphidodon lacrymatus			
Part of			9097
Dascyllus carneus			
Part of			5056
Myripristis botche			
Part of			7156
Naso vlamingii			
Part of			4744
Nemateleotris magnifica			
Part of			11944
Odonus niger			
Part of			12352
Pterocaesio tile			
Part of			7528
Mulloidichtys vanicolensis			
Part of Fish	521555	521555	
Environment		862174	862174

Table S3: Number of thumbnails of each fish species present in test datasets used in this study

Class	Dataset for testing models performance	Dataset for testing model performance vs human performance
Abudefduf sparoides	103	88
Abudefduf vaigiensis	59	47
Chaetodon trifascialis	208	146
Chromis weberi	269	
Dascyllus carneus	269	
Monotaxis grandoculis	72	
Myripristis botche	20	
Naso elegans	189	165
Naso vlamingii	358	
Nemateleotris magnifica	246	
Odonus niger	176	
Plectroglyphidodon lacrymatus	150	
Pomacentrus sulfureus	1567	443
Pterocaesio tile	215	
Pygoplytes diacanthus	39	35
Thalassoma hardwicke	111	73
Zanclus cornutus	64	53
Zebrasoma scopas	184	144
Total	4405	1197

Table S4. Performance of CNN model trained with T4 thumbnails set to identify nine fish species with no post processing; species are identified in columns and rows refer to whole fish and parts of fish present in the training dataset.

Part of species X means that some individual were recognized as part of a fish species.

Only percentages of over 1% are shown.

Species	*A.sparoides*	*A.vaigiensis*	*C.trifascialis*	*N.elegans*	*P.sulfureus*	*P.diacanthus*	*T.hardwicke*	*Z.cornutus*	*Z.scopas*
A. sparoides	82.8								
A .vaigiensis	1.1	80.0							
C. trifascialis			85.1						
C. weberi							1.1		
N. elegans				95.1					3.9
P. sulfureus					73.8	2.6			
P. diacanthus							86.8		
T. hardwicke								87.3	
Z. cornutus									89.0
Z. scopas									92.7
Part of *A. sparoides*	6.0								
Part of *A. vaigiensis*	1.0	9.1							
Part of *C. trifascialis*			2.6						
Part of *N. elegans*				1.6					
Part of *P. sulfureus*			1.1		4.3				
Part of *P. diacanthus*									
Part of *T. hardwicke*							2.6		
Part of *Z. cornutus*								6.2	
Part of *Z. scopas*									
Environment	8.0	10.9	9.5	2.2	20.9	7.9	9.2	4.6	2.8

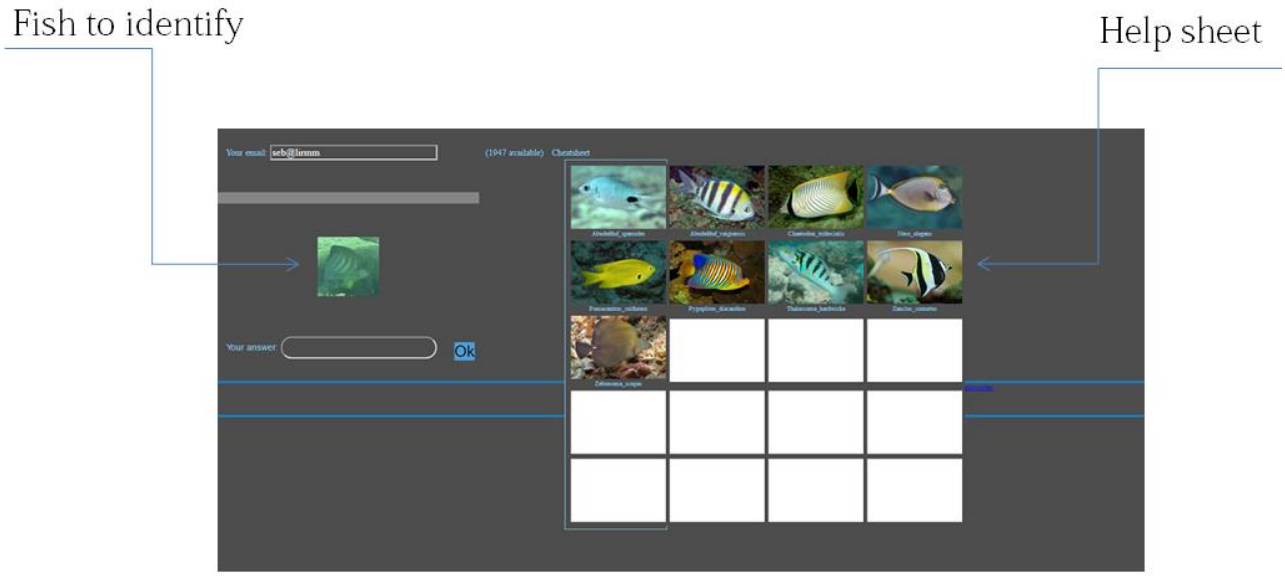
Table S5. Performance of our final CNN model to identify 9 fish species

Raw model output was post-processed with following decision rule: outputs " part of species X" and "species X" are considered the same (i.e., the results of *A. sparoides* and *part of A. sparoides* are added together*); s*pecies are in columns with rows indicating the percentage of good identification for each species and only values over 1% are shown.

Species	*A. sparoides*	*A. vaigiensis*	*C. trifascialis*	*N.elegans*	*P.sulfureus*	*P.diacanthus*	*T.hardwicke*	*Z.cornutus*	*Z.scopas*
A. sparoides	89.0								
A. vaigiensis	2.1	89.1							
C. trifascialis			97.7						
C. weberi							1.1		
D. caruleus									3.9
N. elegans				95.7					
P. sulfureus			3.1		78.1	2.6			
P. diacanthus						86.8			
T. hardwicke						89.4			
Z.cornutus								95.2	
Z. scopas									92.7
Environment	8.0	10.9	9.5	2.1	20.9	7.9	9.2	4.6	2.8

SUPPLEMENTARY FIGURES

Supplementary Figure 1. Screenshot of the online application used for testing human performance in identifying fish on thumbnails. Picture of fish to identify is displayed on the left part. Name for species should be typed in the bottom text bar (with auto-completion). The help box with examples of the 9 species to identify is visible on the right.



Chapitre 5

Contrôle et prévention des erreurs d'un CNN d'identification d'espèces de poisson.

Après avoir observé l'efficacité des algorithmes de Deep Learning pour effectuer des tâches d'identification à l'espèce que ce soit par rapport aux autres méthodes d'analyses automatiques ou par rapport aux humains, nous avons décidé d'améliorer l'efficacité de ces algorithmes, en renforçant la maîtrise de leur taux d'erreur. Pour créer notre modèle, nous avons tout d'abord défini plusieurs types d'enrichissement de données (*data augmentation*) pertinents à notre sujet. Par exemple, un enrichissement par rotation à 180° des vignettes (créant des images avec des poissons sur le dos) ne présente pas d'intérêt, puisqu'il décrit une situation qui ne sera jamais vu en pratique plus tard, apportant plus de bruit que d'information. En revanche, un enrichissement obtenu grâce au changement de résolution des vignettes représente un cas possible (un poisson éloigné sera de moins bonne résolution qu'un poisson proche de la caméra). Nous avons donc sélectionné 20 enrichissement basés sur des transformations appliqués à l'image (flips horizontaux), la réduction du nombre de pixels (changement de résolution) ou la modification de la valeur de ses pixels. La modification de ces valeurs peut changer la luminosité de l'image, la force du contraste, l'équilibre entre les couleurs, mais aussi le renforcer ou la diminuer des contours.

Ensuite, à partir d'un jeu d'entrainement contenant 25.000 vignettes de poissons de 20 espèces, nous avons entrainé 20 architectures profondes ResNet (Residual Network [He et al., 2016]) pour obtenir 20 modèles. Chacun de ces modèles a été entraîné à partir d'une base de données contenant les vignettes "naturelles", ainsi que les vignettes obtenues grâce à un et un seul enrichissement. Finalement, nous avons testé ces modèles sur une base de test contenant entre 16 et 80 individus de chaque espèces, afin d'observer les enrichissement les plus efficaces pour augmenter les résultats de l'identification.

Nous avons ensuite sélectionné les 4 enrichissements les plus efficaces, à savoir, 2

diminutions de contraste, et 2 augmentations de contraste (nous avons obtenu environ 10% d'amélioration de classification sur notre base de test grâce aux réseaux entraînés avec ces enrichissements par rapport au modèle créé sans enrichissement de données). Ces enrichissements, associés aux flips horizontaux, nous ont permis d'augmenter la taille de la base d'entrainement d'un facteur 10. Cette base a été utilisée pour créer un nouveau modèle grâce à une architecture ResNet. Nous proposons ensuite une méthode de post-traitement sur les sorties de ce modèle, permettant de contrôler et de maîtriser le taux d'erreur, au moyen de l'ajustement d'un seuil de confiance par espèce obtenu grâce à un apprentissage. Cette méthode de post-traitement a permis de diminuer le taux moyen de mauvaises classifications des espèces de 22% à 2.2%.

A calibration 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]

Abstract

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

Here, we proposed a new framework to control the error rate of DLAs. More specifically, for each species, a confidence threshold was automatically computed using a training dataset independent from the one used to train the DLAs. These thresholds were then applied to post-process the outputs of the DLAs, assigning images with classification scores under the species threshold to a class called "unsure". We apply this framework to a study case identifying 20 coral reef fish species from 13,232 underwater images.

The overall rate of species misclassification decreased from 22% with the raw DLAs to 2.2% after post processing using 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 impacts of human activities on ecosystems, the capacity to monitor biodiversity at large scales and high frequency is becoming an urgent need but also a major challenge (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, image or video sensors have been extensively deployed on land and sea using satellite imagery (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) to record living 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). For fish, classic imagery analysis is also used by citizen science; some public tools like inaturalist.org or fishpix

(http://fishpix.kahaku.go.jp) offer the possibility to upload individual fish images that are manually identified by experts at the species level.

However, processing photos or videos manually is a highly demanding task, especially in underwater environments, where the specific 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 takes 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.

The last generation of Deep Learning Algorithms (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 of 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.

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 of all its individuals is the highest for rare species given their low abundances, while these rare species are the most endangered and often play important ecosystem roles (Mouillot et al. 2013). 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 increase species richness and could also overestimate the home or geographical range of species.

Since biodiversity monitoring should be as accurate as possible, automated identification of individuals on images should provide 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 notion 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 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 framework can reject the classification.

Applied to machine learning, two main architectures can currently measure the risk associated to outputs. The first one can learn a rejection function during the training, in parallel to the classification learning (Cortes et al.,

2016, Geifman et al. 2017). The second one, 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 based (De Stefano et al. 2000). A recent comparative study suggests that meta-algorithm-based methods are the most efficient (Kocak et al., 2017).

Another way 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 consist of the matching between the score predicted by the machine-learning model and the real probability of true positive. While Deep Learning models produces more accurate models 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 issues through 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 Scalling, 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.

In this paper, we present a general method that accounts for uncertainty in the classifier outputs. Unlike calibration methods, our approach doesn’t

change the algorithm's outputs. Instead, we simply assess the behaviour of the model thanks to a validation dataset. We can then define a fine tuned threshold per class, allowing us to take into account that the Deep model confidence can be highly variable between "easy" classes and "difficult" classes.

Through the addition of a new class "unsure" (being the class of all predictions with scores lower than the predicted class threshold), we allowed the control of the coverage (total amount of images automatically identified) and misclassification rate. We applied this framework to classify 20 species of coral reef fish in underwater images, and assess its efficiency through 3 real life scenarios.

Material and methods

Data

We used 3 independent fish datasets from 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 differ 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 images of individuals and their environment for one species (Moorish idol/*Zanclus cornutus*).

We used 130 videos for the training dataset, from which we extracted 69.169 images of 20 fish species (Fig. sup 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 (*T0*) composed of 691.690 images (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 (*T1*) contained 6,320 images from 20 videos with at least 41 images per species, and the second (*T2*) contained 13,232 images from 25 videos with at least 55 images per species (Supp. Tab. 1). We later used dataset *T1* to tune the thresholds and *T2* 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 defined by human experts and 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, which ensures that 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 output

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 $i = \{1, .., n\}$, 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 the image to belong to the predicted class, while a lower score implies a "lower chance". 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 output.

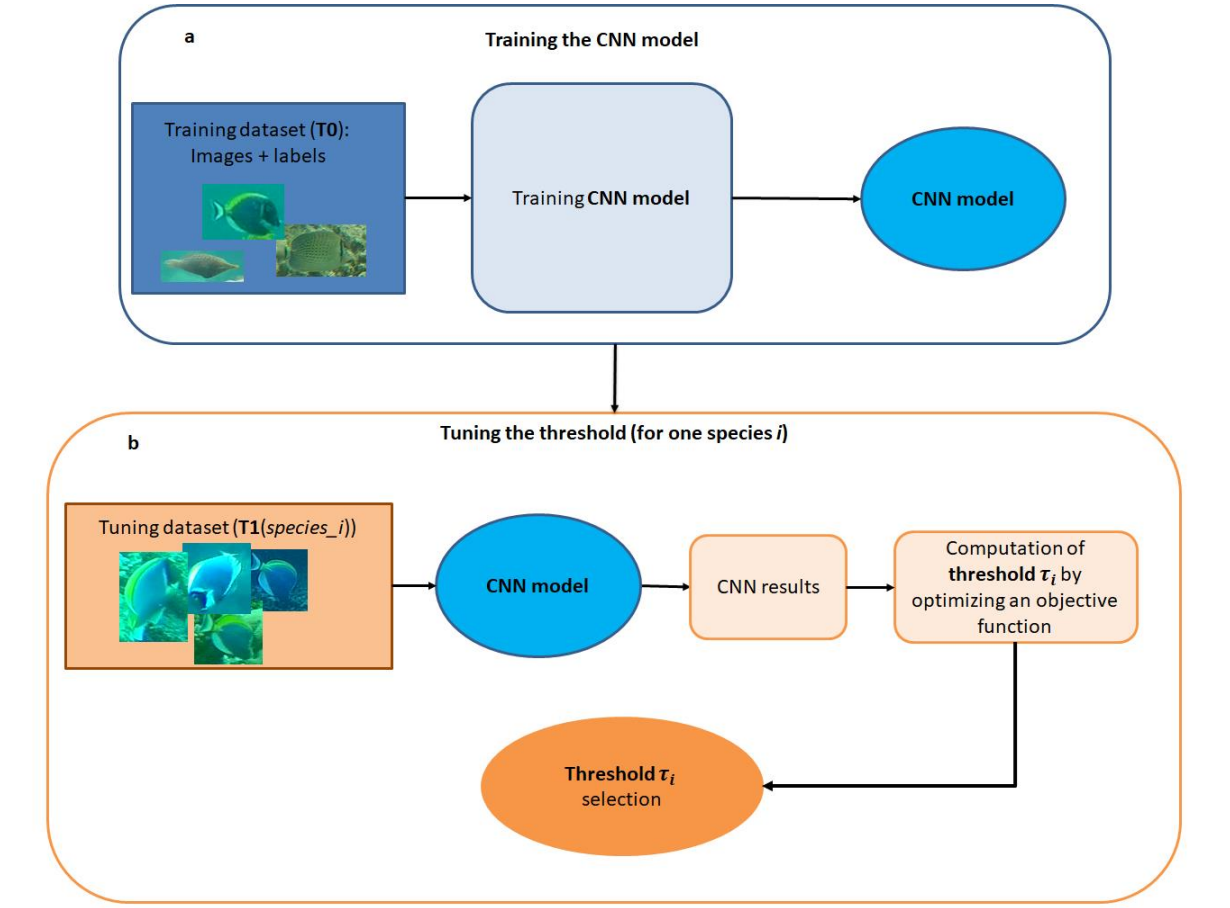
Assessing the risk of misclassification by the CNN

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 confident the classification is.

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

- First, a classification model is built by training a CNN on a given database *T0* (Fig. 2 (A))
- Then, the 2nd phase consists of tuning a risk threshold τ_i specific to each species, noted *i,* with $i \in \{1,\ldots,n\}$, using a second and independent database noted *T1* (Fig. 2 (B)).


a
Training the CNN model
Training dataset (T0): Images + labels
Training CNN model
CNN model
b
Tuning the threshold (for one species i)
Tuning dataset (T1(species_i))
CNN model
CNN results
Computation of threshold τ_i by optimizing an objective function
Threshold τ_i selection

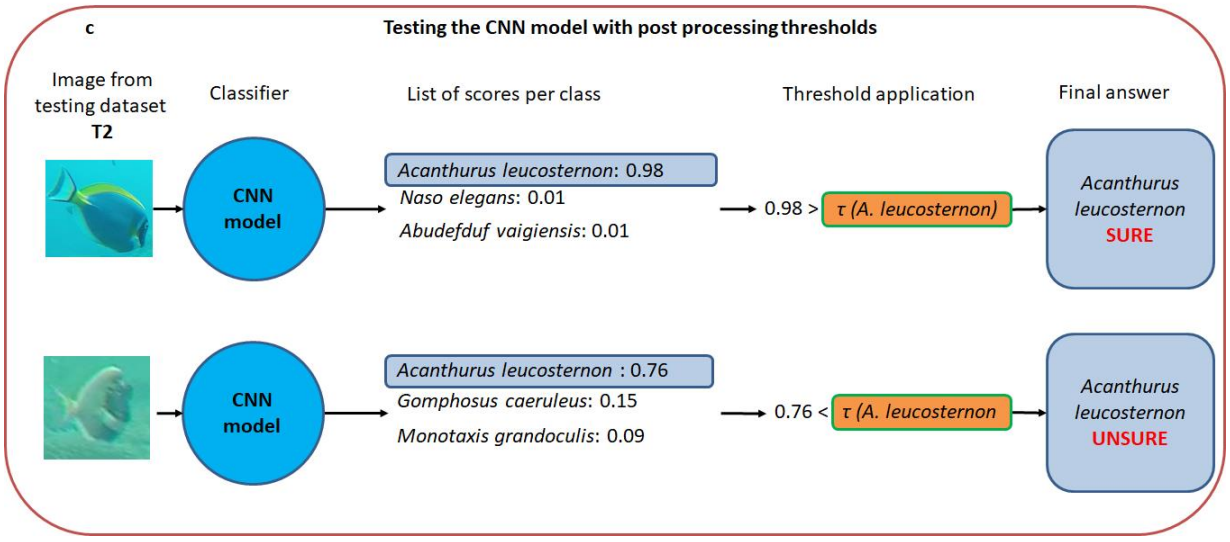

c
Testing the CNN model with post processing thresholds
Image from testing dataset T2
Classifier
List of scores per class
Threshold application
Final answer
CNN model
Acanthurus leucosternon: 0.98
Naso elegans: 0.01
Abudefduf vaigiensis: 0.01
0.98 > τ (A. leucosternon)
Acanthurus leucosternon SURE
CNN model
Acanthurus leucosternon : 0.76
Gomphosus caeruleus: 0.15
Monotaxis grandoculis: 0.09
0.76 < τ (A. leucosternon
Acanthurus leucosternon UNSURE

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 users' goal. This goal is a trade-off between the accuracy of results 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 fish species). The most likely class *i_max* for the image is the one with the highest score S_i. We then compared this highest score $S_{i_max}$with the computed confidence threshold for this species (τ_{i_max}) obtained in phase (B). If the score was lower than the computed threshold, then the input image was classified as "unsure". Otherwise, we kept the CNN classification.

Computing user-based thresholds

We modeled the classification results of a CNN, when fed with a sequence of images, as a random variable *X* which correspond to the fish species belonging to the set $\{1,..,n\}$, with *n* being the number of different species *i*. Similarly, we modeled the ground truth of a sequence of images as a random variable *Y* which corresponds to the same set of species $\{1,..,n\}$ as *X*.

Y and *X* are sequentially linked by the CNN classification process, i.e. when an image is processed by the CNN, a result is given as *X* and the ground truth is given by *Y*. Additionally, we modeled the score given by the CNN for each species by a vector of *n* real positive variables noted S_i with $i \in \{1,..,n\}$.

For a given species numbered i with $i \in \{1,..,n\}$, and a chosen threshold τ, we can compute 3 variables with $\#(.)$ the enumeration function:

The Correct Classification rate of a species *i* for a given threshold τ, $CC_i(\tau)$ is given as:

$$CC_i(\tau) = (X = i|Y = i) = \frac{\#(X = i|Y = i) - \#(X = i, S_i < \tau|Y = i)}{\#(Y = i)}$$

The Misclassification rate of a species *i* for a given threshold $\tau, MC_i(\tau)$ is expressed as:

$$MC_i(\tau) = (X \neq i|Y = i) = \frac{\#(X = j|Y = i) - \#(X \neq i, S_i < \tau|Y = i)}{\#(Y = i)}$$

The Unsure Classification rate of a species *i* for a given threshold τ, $UC_i(\tau)$ is:

$$UC_i(\tau) = \frac{\#(X = i, S_i < \tau_i | Y = i) + \#(X \neq i, S_i < \tau | Y = i|)}{(Y = i)}$$

For each species we have:

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

We can also note that the coverage (the rate of images for which a classification is given) can be defined as

$$CC_i(\tau) + MC_i(\tau)$$

Then, we selected optimal thresholds $\{\tau_i\}_{i=1}^{i=n}$ in order to minimize or maximize $\{CC_i\}_{i=1}^{i=n}$ or $\{MC_i\}_{i=1}^{i=n}$ depending on the user's goal by using all the images of species *i* in the dataset *T1*. For this purpose, we defined three goals:

- The first goal *G1* consists of keeping the best correct classification rate while reducing the misclassification error rate (which implies low coverage). We used two steps. First, we found the threshold(s) τ which maximizes $CC_i(\tau)$. Notice that we can have several thresholds which reach this maximum so 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:

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

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

- 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 took as Se_{g2} the set of threshold(s) which minimize MC_i. Then we defined the optimal threshold τ_i by taking the one in Se_{g2} which maximizes CC_i:

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

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

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

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

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

$$\tau_i = \underset{\tau' \, in \, Seg3}{\arg\max} \, CC_i(\tau')$$

For a given image, the classification and post-process is sequential as follows:

- First, the image is given to the CNN, which outputs a score S_i for each class (species)
- Second, for the class $j = \underset{i}{\arg\max} \, S_i$ (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 *j*. If $S_j < \tau_j$, the prediction is changed to "Unsure", otherwise, it is confirmed as the class *j* (Fig. 2 (C)).

The misclassification rate for a species *i* during the application is noted

$$MC'_i = (X = j | \, j \neq i | Y = i) = \frac{\#(X = j | Y = i) - \#(X = j, S_j < \tau_j | Y = i)}{\#(Y = i)}$$

and the unsure classification rate is noted

$$UC'_i = \frac{\#(X = i, S_i < \tau_i | Y = i) + \#(X = j, S_j < \tau_j | Y = i | j \neq i)}{(Y = i)}$$

In terms of classification, it means we transform the 2 classification options (False, True) in 3 options (Fig.3).

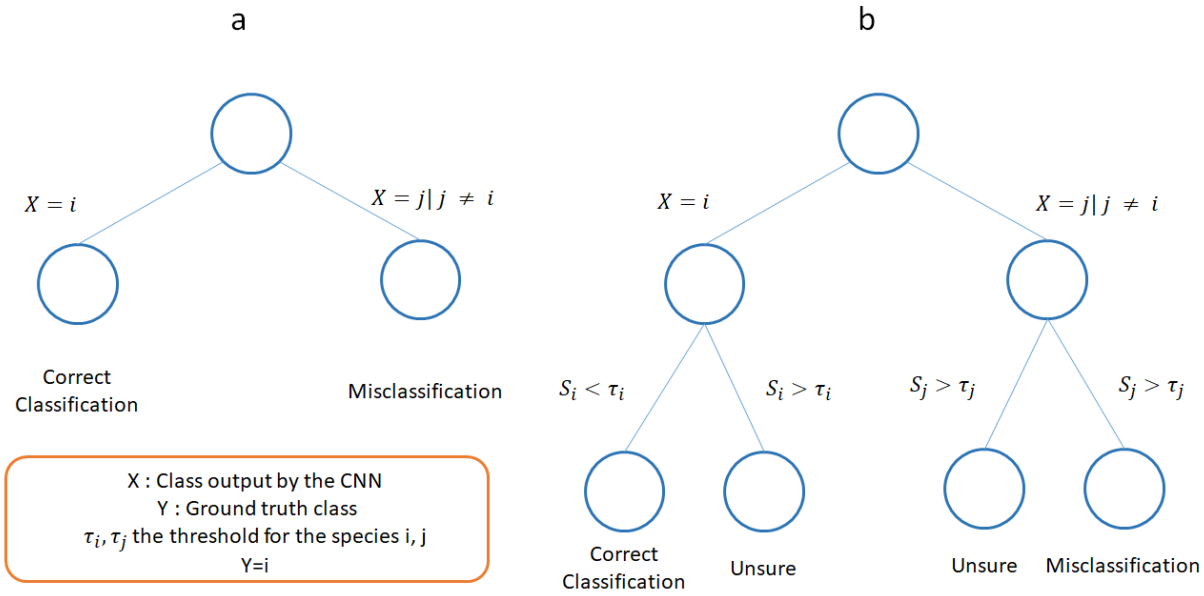


Fig.3: (A) Usually, the classification results of an image of class *i* can either be true, if the model classifies it as *i,* or wrong, if the classifier classifies it as *j* with $j \neq i$. We propose a post processing to build confidence thresholds for each class to obtain 3 types of results, true, false, and unsure, as seen on (B). The goal is then to transform as many misclassifications as possible as "Unsure", while preventing to transform as many Correct Classification as possible as "Unsure".

In our study, we identified all the images contained in *T2*, without post processing (threshold tuning+ threshold application) which also provided the accuracy of our model without cross-validation.

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 ideal conditions, we selected the thresholds based on the dataset *T2* and we applied them to the same dataset *T2*.

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 tuning 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*.

Results

Accuracy of the CNN model

The mean rate of correct classification of fish images by the raw CNN was of 77.9%, with rates of correct classifications per species ranging from 53.4% to 99.1% (sd= 15.35) (Tab. 2).

Table 2*. Output of the deep learning classifier without post processing. Percentages of correct classifications are shown for the 20 fish species.*

Species	Test dataset *T2* (% of correct classifications)
Chaetodon trifasciatus	87.80
Chaetodon trifascialis	90.15
Naso brevirostris	53.46
Chaetodon guttatissimus	84.05
Thalassoma hardwicke	90.90
Pomacentrus sulfureus	90.84
Oxymonacanthus longirostris	96.42
Monotaxis grandoculis	56.82
Zebrasoma scopas	62.69
Abudefduf vaigiensis	99.07
Amblyglyphidodon indicus	59.61
Acanthurus lineatus	59.60
Chromis ternatensis	60.89
Chromis opercularis	59.13
Gomphosus caeruleus	75.72

Acanthurus leucosternon	85.94
Halichoeres hortulanus	82.92
Naso elegans	93.09
Chaetodon auriga	87.64
Zanclus cornutus	81.35
Average	77.91

Images obtained softmax scores between 41 and 100 with 80% of images classified with a score between 98 and 100 (Fig. 4a). The score was highly variable among species (Fig. 4 b, c).

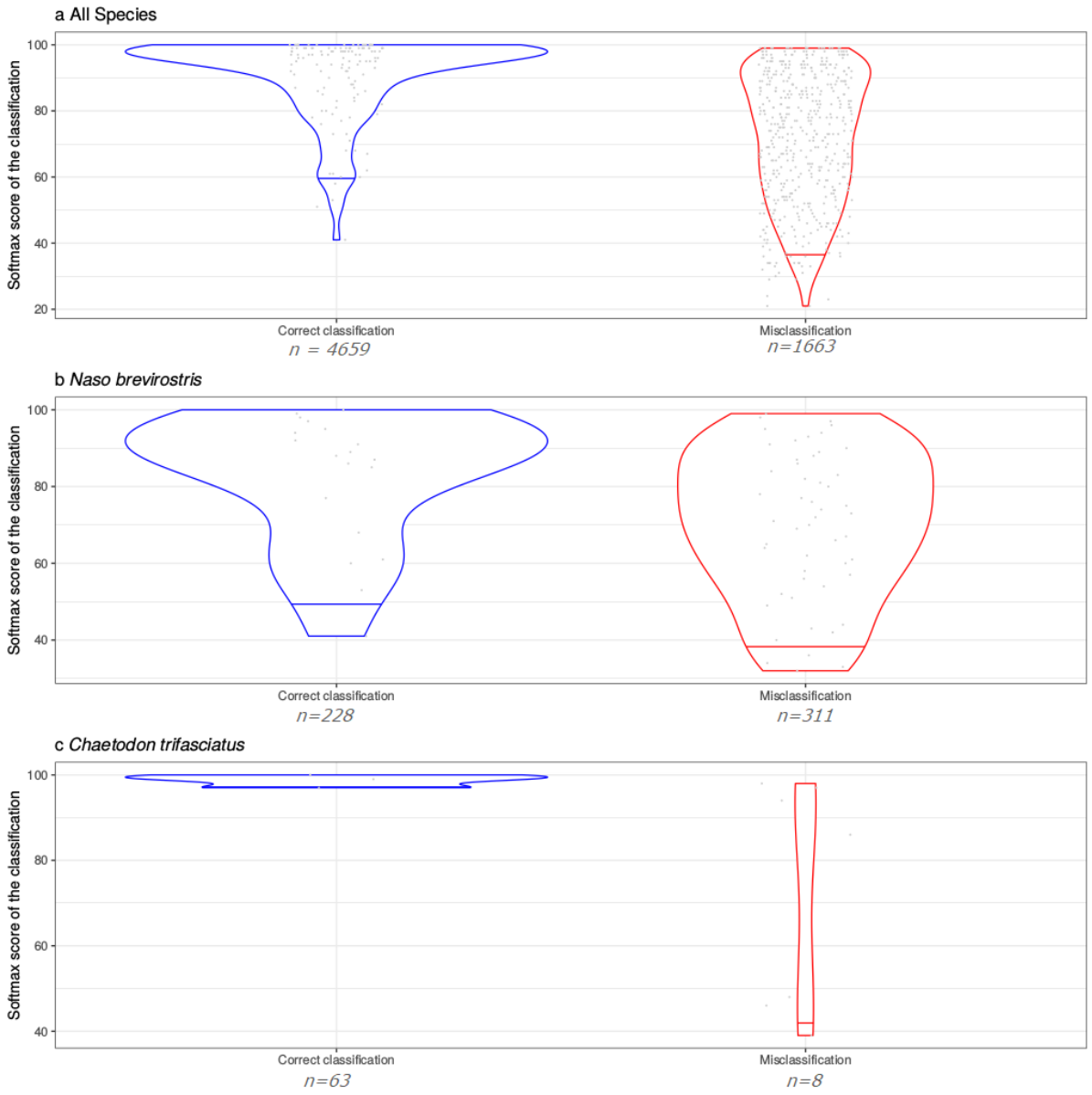


Fig 4: Distribution of correct classifications and misclassifications of fish images with respect to the score from the CNN model, for all species (A), and for 2 species, the Brown unicornfish (*Naso brevirostris*) (B) and the

Melon butterflyfish (*Chaetodon trifasciatus*) (C). We also plotted the 5% bottom line for each type of classification.

The influence of the threshold on classification results

The results here were computed with a threshold built and applied on the same dataset (ideal conditions). When the threshold score varied between 0 and 100 over all fish species, the rate of misclassifications 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.

The thresholds were species dependant with high variations for G1, with values from 41.26 to 99.95, and values from 94.63 to 99.98 for G3 (Supp. Tab. 2 and 3).

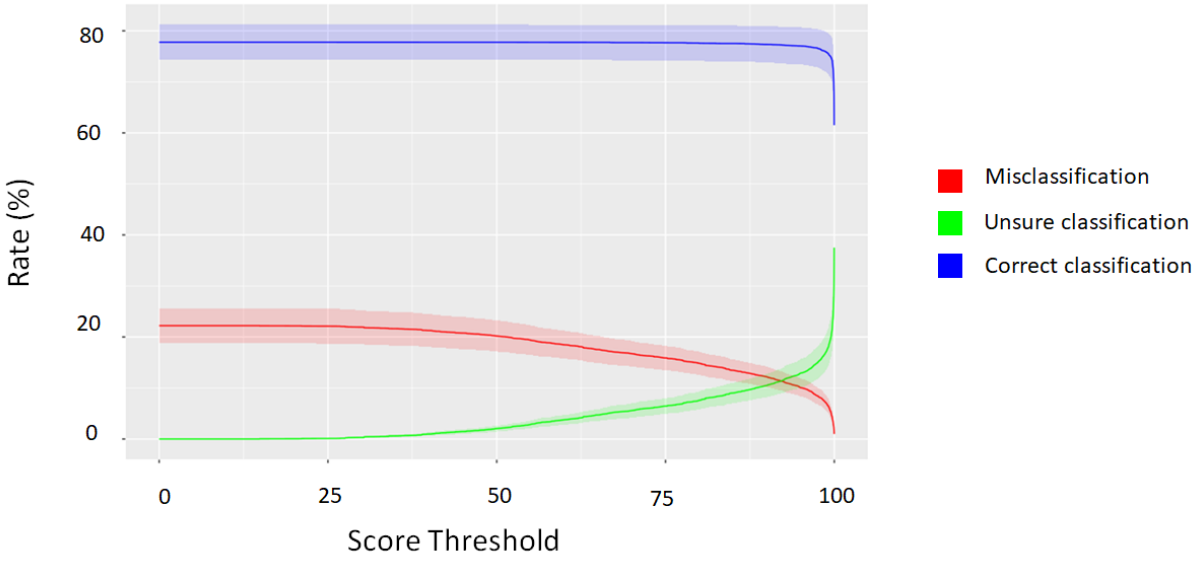


Fig. 5: Average distribution of correct, wrong, and unsure classifications for all species along gradient of confidence threshold score.

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 77.7% (standard deviation= 15.35) of correct classifications, 10.7% (s.d= 7.84) of unsure classifications, and 11.4% (s.d= 12.55) 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 75.1% (s.d= 18.33) correct classifications, 21.3% (s.d= 17.20) of unsure classifications, and 3.5% (s.d= 2.11) of

misclassifications. Compared to the first goal, we lost 2.6% of correct classifications, on average, but we decreased the rate of misclassifications by 7.9%.

For the third goal G3, we maximized the number of correct classifications with $MC_i = \min MC_i$. We obtained a rate of 66.2% (s.d= 23.65) correct classifications, 32.7% (s.d= 23.19) of unsure classifications, and 0.9% (s.d= 2.15) of misclassifications, on average. Compared to the second goal, we decreased the rate of correct classifications by 8.9% and the rate of misclassifications by 2.6% (Supp. Tab. 4).

Application of the method in real conditions

For the 3 goals, the differences in terms of correct, wrong and unsure classification rates between the optimized thresholds and the cross-validation thresholds were all under 3% (Fig sup. 3, Supp. Tab. 5).

Finally, the post processing threshold procedure decreased the misclassification rate by at least 9.9%, for all goals, and 20% at most compared to the raw output of the Deep Learning model (Fig. 6. (B)).

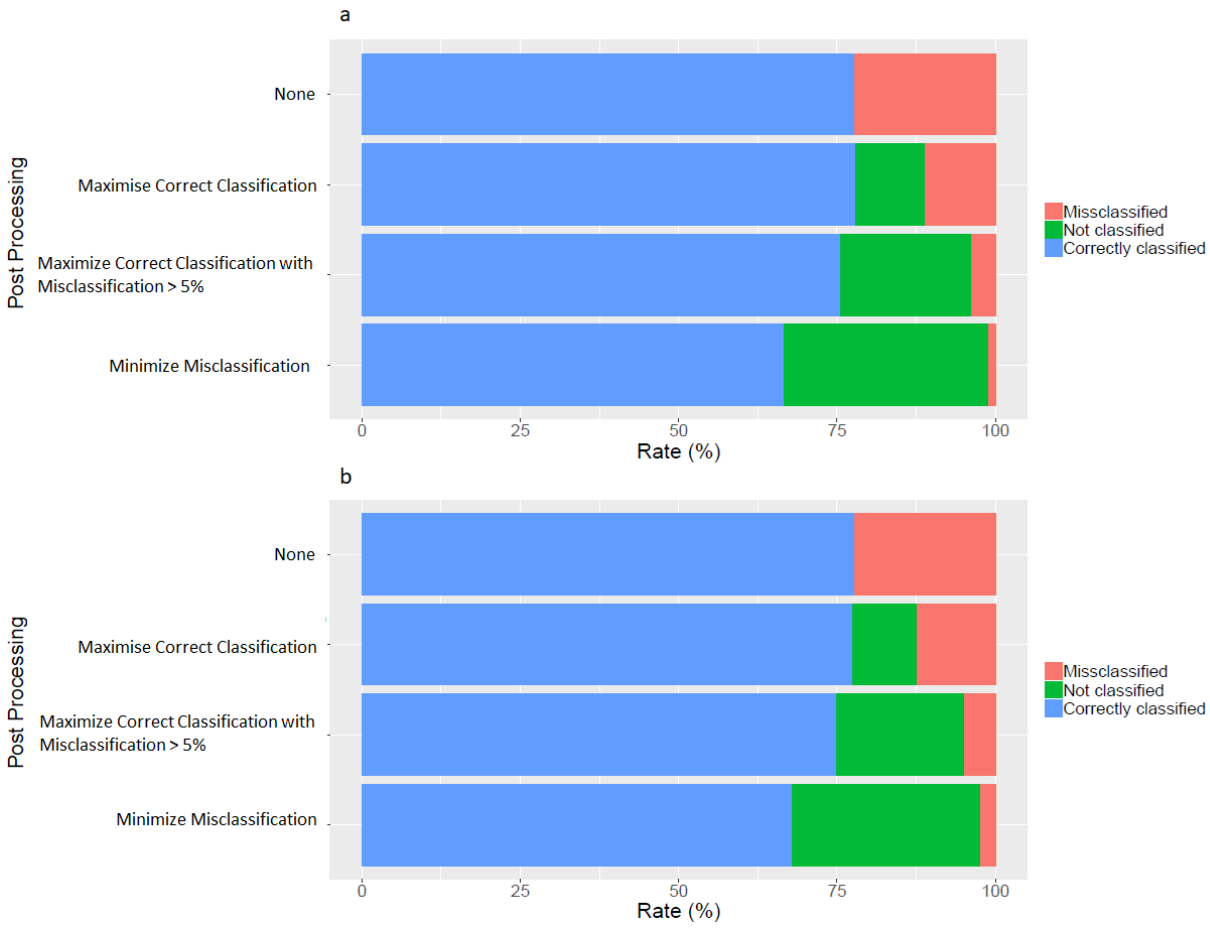


Fig. 6 Ideal scenario (rates were obtained by tuning the thresholds on *T2* and using *T2* as testing set. (A)) and real-life scenario (rates were obtained by tuning the thresholds on *T1* and using *T2* as testing set (cross-validation). (B)) classification scores.

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

than 5% of misclassifications, 4) Goal 3: maximizing correct classification with $MC_i = \min MC_i$.

Finally, we assessed the robustness or our method by doing the same experiment while switching *T1* and *T2* roles (Tab. sup. 6, 7, 8). For each goal, the unsure classification rate was higher after the switch (2.31% for G1, 7.11% for G2, and 9.57% for G3), implying lower scores were obtained in both correct classification (3.77%, 5.6%, 8.35%) and misclassification (2.2%, 1.51%, 1.23%) rates.

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 showed that the risk of misclassification by CNN algorithms can be measured and controlled in a post-processing step to provide more

accurate results. 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 correct classifications) method and the species thresholds can be set according to the goal of the study or the availability and time of experts. Here we define three main goals which can represent archetypal study cases.

The three different goals presented earlier correspond to real use-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. To better anticipate the impact of such species, the user needs to be aware of the first occurrence on reefs

and can thus accept having “false alarms” (fish wrongly identified as invasive lionfish). We can also cite the case of rare events such as the detection of specific individuals, like Whale Sharks, through photo-identification (McKinney 2017) where the primary goal is to avoid missing the true positives while the correction of false positives is very easy. In any case, experts will be required to validate the true positives since a noticeable proportion of false positives can be detected.

The second goal, maximizing the correct classification rate but 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 (>2hours) 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 a bigger problem than avoiding false negatives, which creates many “unsure” classifications. This can be the case when there is the need to accurately analyze a relatively small dataset with the support of experts who can help to identify species on “unsure” images. In this case, we assume that we have more human resources than in the two previous scenarios. For instance, assessing biodiversity 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) can require a minimum number of misclassifications and the support of experts.

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

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).

Catford, J. A., Bode, M., & Tilman, D. (2018). Introduced species that overcome life history tradeoffs can cause native extinctions. Nature communications, 9(1), 2131.

Cortes, C., DeSalvo, G., & Mohri, M. (2016). Boosting with abstention. In Advances in Neural Information Processing Systems (pp. 1660-1668).

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.

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.

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

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

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

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

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.

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.

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).

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.

Hodgson, A., Peel, D., & Kelly, N. (2017). Unmanned aerial vehicles for surveying marine fauna: assessing detection probability. Ecological Applications, 27(4), 1253-1267.

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.

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.

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.

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.

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.

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).

Leclerc, C., Courchamp, F., & Bellard, C. (2018). Insular threat associations within taxa worldwide. Scientific reports, 8(1), 6393.
LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature,521(7553), 436.
Levi, D. M. (2008). Crowding—An essential bottleneck for object recognition: A mini-review. Vision research, 48(5), 635-654.
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.
Lyons, K. G., & Schwartz, M. W. (2001). Rare species loss alters ecosystem function–invasion resistance. Ecology letters, 4(4), 358-365.
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.
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.
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.

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.

Naeini, M. P., Cooper, G., & Hauskrecht, M. (2015, February). Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

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.

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.

O'Connell, A. F., Nichols, J. D., & Karanth, K. U. (Eds.). (2010). Camera traps in animal ecology: methods and analyses. Springer Science & Business Media.

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., Pettorelli, 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.

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

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

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

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.

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

Schulte to Bühne, H., & Pettorelli, 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.

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.

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.

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.

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.

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.

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.

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.

Zadrozny, B., & Elkan, C. (2001, June). Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In *Icml* (Vol. 1, pp. 609-616).

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.

Author Information

Affiliation

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

b Research-Team ICAR, LIRMM, Univ 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.

All authors designed the project and the experiments.

All authors reviewed the manuscript.

Corresponding author

Sébastien villon

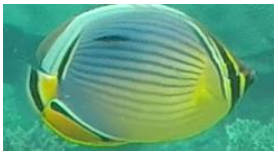
villon@lirmm.fr

Ethics declaration

The authors declare no competing interests.

Supplementary

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

			
Abudefduf vaigiensis	*Acanthurus leucosternon*	*Acanthurus lineatus*	*Amblyglyphidodon indicus*
			
Chaetodon auriga	*Chaetodon guttatissimus*	*Chaetodon trifascialis*	*Chaetodon trifasciatus*
			

Chromis opercularis	*Chromis ternatensis*	*Gomphosus caeruleus*	*Halichoeres hortulanus*
			
Monotaxis grandoculis	*Naso brevirostris*	*Naso elegans*	*Oxymonacanthus longirostris*
			
Pomacentrus sulfureus	*Thalassoma hardwicke*	*Zanclus cornutus*	*Zebrasoma scopas*

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%)
	
Less contrast (60%)	Less contrast on flipped image (60%)

	
More contrast (120%)	More contrast on flipped image (120%)
	
More contrast (140%)	More contrast on flipped image (140%)

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

Family	Species	Training dataset *T0*	First dataset *T1*	Second dataset *T2*
Acanthuridae	*Acanthurus leucosternon*	32,590	235	491
Acanthuridae	*Acanthurus lineatus*	10,080	114	864
Acanthuridae	*Naso brevirostris*	11,340	539	1932
Acanthuridae	*Naso elegans*	73,450	1,436	3,896
Acanthuridae	*Zebrasoma scopas*	49,700	48	579
Chaetodontidae	*Chaetodon auriga*	21,340	737	502
Chaetodontidae	*Chaetodon guttatissimus*	11,820	221	68
Chaetodontidae	*Chaetodon trifascialis*	52,340	41	630
Chaetodontidae	*Chaetodon trifasciatus*	44,210	71	82
Labridae	*Gomphosus caeruleus*	31,310	57	173

Labridae	*Halichoeres hortulanus*	31,920	40	287
Labridae	*Thalassoma hardwicke*	49,510	181	275
Lethrinidae	*Monotaxis grandoculis*	38,930	797	1,422
Monacanthidae	*Oxymonacanthus longirostris*	25,530	54	55
Pomacentridae	*Abudefduf vaigiensis*	51,240	376	216
Pomacentridae	*Amblyglyphidodon indicus*	11,880	636	1,310
Pomacentridae	*Chromis opercularis*	15,250	81	93
Pomacentridae	*Chromis ternatensis*	36,400	300	156
Pomacentridae	*Pomacentrus sulfureus*	54,090	270	142
Zanclidae	*Zanclus cornutus*	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.

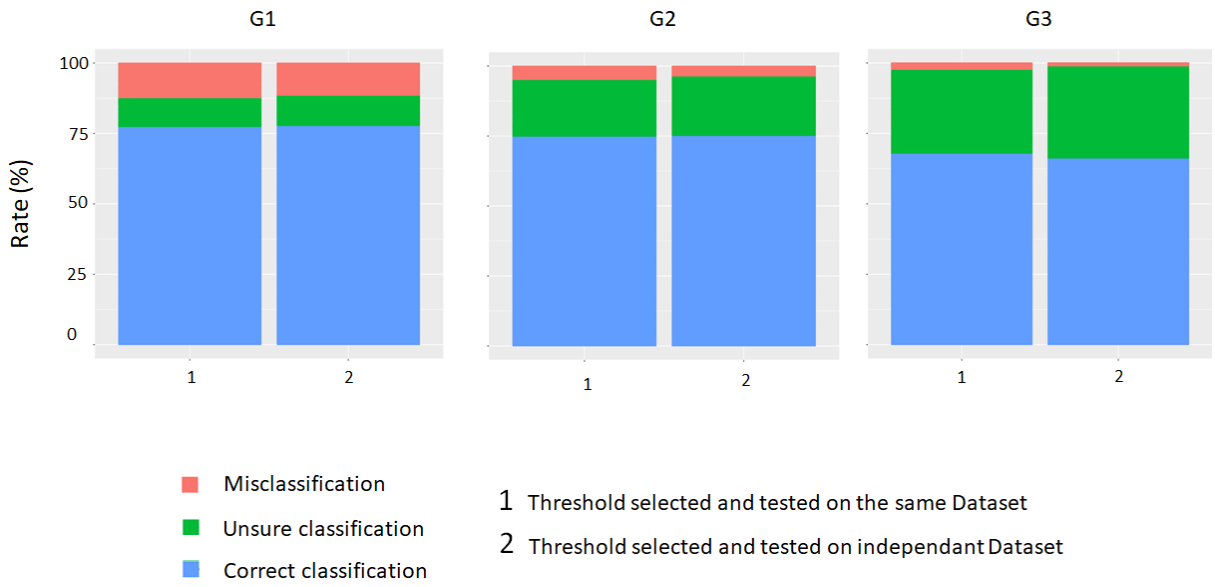
	Without post processing		Goal1	
Species	Misclassification rate	Threshold value	Misclassification rate	Unsure rate
Chaetodon trifasciatus	12.19	97.65	0	12.19
Chaetodon trifascialis	9.84	76.19	8.09	6
Naso brevirostris	46.53	41.26	45.7	10.35
Chaetodon guttatissimus	15.94	98.42	15.94	7.24
Thalassoma hardwicke	9.09	97.11	1.09	7.63
Pomacentrus sulfureus	9.85	98.91	4.92	5.63
Oxymonacanthus longirostris	3.57	99.95	0	3.57
Monotaxis grandoculis	43.17	51.31	39.87	9.07
Zebrasoma scopas	37.3	92.63	12.26	16.58
Abudefduf vaigiensis	0.925	98.93	0.46	0.46
Amblyglyphidodon indicus	40.38	60.77	27.7	0.61
Acanthurus lineatus	40.39	95.4	21.99	22.33
Chromis ternatensis	41.02	78.5	14.74	21.15
Chromis opercularis	40.86	98.65	9.677	29.03

Gomphosus caeruleus	24.27	79.64	19.07	18.49
Acanthurus leucosternon	14.05	87.62	8.96	8.75
Halichoeres hortulanus	17.07	99.79	2.43	12.19
Naso elegans	6.9	62.35	5.72	0
Chaetodon auriga	12.35	99.37	4.38	8.94
Zanclus cornutus	18.64	99.42	3.38	15.25

Supp. Tab. 3) Values of misclassification scores without post processing, and after processing with the threshold selected by optimizing the Misclassification rate.

	Without post processing	Goal 3		
Species	Misclassification rate	Threshold value	Misclassification rate	Unsure rate
Chaetodon trifasciatus	12.19	99.22	0	12.19
Chaetodon trifascialis	9.84	94.63	7.3	13.33
Naso brevirostris	46.53	99.98	8.74	44.56
Chaetodon guttatissimus	15.94	99.84	7.24	24.63
Thalassoma hardwicke	9.09	99.39	0.72	10.54
Pomacentrus sulfureus	9.85	99.98	0	26.76
Oxymonacanthus longirostris	3.57	99.98	0	3.57
Monotaxis grandoculis	43.17	99.98	0.7	71.37
Zebrasoma scopas	37.3	99.9	1.38	63.73

Abudefduf vaigiensis	0.92	99.98	0	1.38
Amblyglyphidodon indicus	40.38	99.98	0.07	75.49
Acanthurus lineatus	40.39	99.94	8.33	38.42
Chromis ternatensis	41.02	99.98	0	67.94
Chromis opercularis	40.86	99.65	3.22	55.91
Gomphosus caeruleus	24.27	99.87	2.31	40.46
Acanthurus leucosternon	14.05	99.97	1.42	22.6
Halichoeres hortulanus	17.07	99.79	2.43	24.39
Naso elegans	6.9	99.98	0.1	18.68
Chaetodon auriga	12.35	99.97	1.19	17.72
Zanclus cornutus	18.64	99.95	0	22.03



Supp. fig 3:From left to right we present the 3 goals: For each goal, the left column (1) shows the results obtained when the threshold is selected with *T1* and applied to *T2* (real cross-validation), and the right column (2) shows the results obtained when the threshold is selected with *T1* and applied on *T1* (optimal conditions).

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	10.7	21.3	32.7
Misclassifications	11.4	3.5	0.9
Correct classifications	77.7	75.1	66.2

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 ideal conditions and real cross validation.*

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	10.2 (-0.5)	20.2(-1.05)	29.7(-2.99)
Misclassifications	12.3(+0.88)	4.8(+1.28)	2.2(+1.28)
Correct classifications	77.4(-0.34)	74.9(-0.23)	67.9(+1.7)

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	10.24	20.24	29.79
Misclassifications	-9.91	-17.41	-19.97
Correct classifications	-0.34	-2.85	-9.83

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

Species	Dataset 1 (*T1*)
Chaetodon trifasciatus	88.7323944
Chaetodon trifascialis	70.7317073
Naso brevirostris	42.3005566
Chaetodon guttatissimus	48.6486486
Thalassoma hardwicke	83.9779006
Pomacentrus sulfureus	90
Oxymonacanthus longirostris	85.4545455
Monotaxis grandoculis	60.1003764
Zebrasoma scopas	68.75
Abudefduf vaigiensis	86.9680851
Amblyglyphidodon indicus	62.2641509
Acanthurus lineatus	82.4561404
Chromis ternatensis	78.6666667
Chromis opercularis	70.3703704
Gomphosus caeruleus	71.9298246
Acanthurus leucosternon	84.6808511
Halichoeres hortulanus	92.5

Naso elegans	89.5543175
Chaetodon auriga	70.8276798
Zanclus cornutus	90.6976744
Average	75.9805945

Supp. Tab. 8: *Rates of unsure, correct, and misclassifications for each goal, with a threshold learned and applied on the same dataset. The first table (a) shows the results obtained when we tuned the thresholds on T2 and applied them on T1 (cross validation), and the second table (b) shows the results obtained with a threshold build on T1 and applied to T1 (ideal condition).*

a

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	16.31	27.35	39.36
Misclassifications	10.01	3.31	1.03
Correct classifications	73.66	69.32	59.59

b

	Goal 1 (%)	Goal 2 (%)	Goal 3 (%)
Unsure classifications	10.24	20.24	29.79
Misclassifications	12.32	4.82	2.26
Correct classifications	77.43	74.92	67.94

Chapitre 6

Détection de poissons dans des vidéos sous-marines.

Le but de cette thèse étant d'obtenir une méthode permettant de localiser et d'identifier tous les individus présents dans une vidéo, nous avons distingué 2 possibilités : 1) La première est de créer un détecteur multi-classes effectuant les 2 tâches à la fois, 2) La seconde est d'utiliser un premier algorithme pour détecter les individus dans les images, puis transférer les résultats de la détection à un modèle de classification. Si la première méthode permet d'adapter la détection et la classification au cours du même apprentissage, le problème principal est d'équilibrer la base de donnée d'apprentissage. En effet, les bases d'apprentissage des algorithmes de détection sont constituées d'images sous-marines complètes, sur lesquelles tous les individus sont annotés (on parle alors d'annotation exhaustive). Hors, certaines espèces et familles présentes dans les vidéos filmées dans les récifs sont particulièrement communes (*Pomacentridae*, *Acanthuridae*), alors que d'autres sont beaucoup plus rares (*Carcharhinidae*, *Scaridae*). De plus, il est très difficile de trouver des moments ou les espèces rares sont présentes à l'écran non entourées d'espèces communes, rendant quasiment impossible l'équilibrage des classes. Ces deux conditions induisent ainsi un biais dans l'apprentissage, et un manque de robustesse pour identifier les classes rares. Nous avons donc décidé de nous orienter vers la seconde proposition, et de créer premièrement un modèle de localisation de poissons dans des vidéos sous-marines, que nous associerons ensuite à nos modèles de classification d'espèces. Ce modèle à double étape présente aussi un intérêt à très court terme, puisqu'il permettrait de pré-traiter nos vidéos, en détectant les moments de la vidéo sans intérêt (aucun poisson présent), supprimant ainsi une tâche manuelle fastidieuse. Dans ce chapitre, avons aussi étudié le compromis entre le temps d'annotation nécessaire pour créer une base de données d'entrainement et la robustesse du modèle obtenu à partir de celle-ci. Nous avons aussi observé la robustesse du réseau aux changements d'environnement (changement d'océans et d'habitats). Cette étude de la transférabilité des modèles créés par apprentissage profond nous a ainsi permis d'étudier la robustesse de notre approche face à des environnement absent de la base

d'apprentissage, principalement composés de d'assemblages denses de coraux "durs". Notre base de vidéos de test présente de nombreux autres paysages marins, composés des coraux clairsemés, ou de fonds recouverts majoritairement d'algues ou de coraux "mous", ainsi que de très nombreuses espèces non présente dans les vidéos de la base d'entraînement. Nous avons obtenu des résultats allant jusqu'à 0,84 de F-mesure moyenne avec notre modèle entraîné sur la base principale (et un nombre important d'images par vidéo analysées avec une F-mesure de 1), et une transférabilité importante, notre modèle ayant aussi été capable de localiser des poissons de Méditerranée avec une F-mesure de 0,83, bien qu'aucune classe présente en Méditerranée ne soit incluse dans la base d'apprentissage, acquise entièrement dans l'océan Ouest Indien.

Automatic detection of fish in underwater videos of different coastal marine ecosystems by Deep-Learning: influence of the ecosystem and of the manual annotation

Key words: Object detection, Underwater videos, Mismatch, Fish detection, Reef Fish, Deep Learning

Introduction

Global changes due to anthropic activities are causing major stains on coral reefs ecosystems (1) (2) (3) (4) . Hence, there is an urgent need for efficient monitoring of marine biodiversity to assess how anthropogenic disturbances affect it across space and time. As the monitoring of a whole reef ecosystem is a highly demanding task, proxies, such as reef fish communities (5) are often used. When done by humans, such monitoring consist of diving, and identifying all species of all individuals underwater.

Monitoring could aim to assess biodiversity from individuals (in particular for flagship or endangered species (6)) to communities (7) (8).

Recent technological advances in sensors have allowed the raise of large scale video-based monitoring campaigns that have collected unprecedented volumes of data (GlobalFinPrint (9) gathered >20.000 hours of videos, and Letessier et al. (10) collected more than 2000h of videos). To be analyzed, these videos require experts to watch them entirely and to annotate them manually. As accumulated data is exceeding manual human processing capacities, an effort is thus to be made to improve methods for automatic processing of videos. Deep Learning Algorithms (DLAs) have been increasingly applied for automatic species identification (11) (12) and localisation of individuals in images (13). Localisation algorithms improved greatly mainly since 2014 (14). Among

these algorithms, a trade-off is to be made between algorithms offering real time detection with potential high error rates (15) (16), and more robust but more time-consuming methods (17) (18). The later could also be used to pre-process videos data to provide semi-quantitative assessments videos (no fish, a few fish, how many individuals…). Best available method to localise fish in full hd underwater videos from the field was proposed by Labao et al. (19) , who focused on localising mostly small individuals (<50x50 pixels) on coral reefs. An effort is still to be made to pave the way for a generic and easy to use fish detection algorithm.

The main issue for the fish detection task is the variability between individuals of the >5000 fish species inhabiting reefs. Hence, fish visible on a single video display a broad range of size, body shape, color patterns and position of fins. Moreover, in all ecosystems many fish species are rare which is a challenge to build a balanced training database (i.e the images used to fit the Deep Learning model). Indeed, as Deep Learning networks extract statistic information's on images to learn how to discriminate fish and seascape, the unbalance between species could lead to a lack of robustness of the model to identify rare species

Therefore, it is a necessity to assess the robustness of Deep Learning models trained with limited data in terms of species and annotations to localise fish species absent from the training datasets. In particular, we are looking for the ability of a Deep model to be

robust to dissimilarity in species pools between seas and dissimilarity in background with reef seascape dominated by hard coral (Indo-Pacific), or soft gorgonians (Caribbean), or algae beds on temperate reefs

In this paper, we assess the trade-off between the human effort, in terms of number of videos to collect and individuals to annotate for building the training database, and the efficiency of model to localise fish in underwater videos. We then assess the robustness of the best model to localise fish species absent from the training dataset including from reefs with different seascape.

Material and Methods

Training datasets

In order to compare the impact of the training datasets, we trained the same Deep Learning architecture independently 4 times with 4 datasets to obtain 4 fish detection models.

The stationary videos (lasting between 5 minutes and 26 minutes) used to build these training datasets were all recorded in the Mozambique Channel located in the Western Indian Ocean. We recorded 20 videos from Mayotte Island in 2016 and 2017, and 8 videos from Europa Island in 2019.

Videos were recorded between 1 and 10 meter deep on 21 seascapes in Mayotte and 10 seascapes sites In Europa. Videos were all recorded in full HD (1920*1280 pixels) with 25 frames per second and a “linear” setting for field of view. All videos have different seascapes because of difference in relative cover of seafloor by corals and sediment and difference in corals diversity. In addition, fish from different videos could visually differed because of difference in light depending on time of the day, weather and water turbidity.

We extracted 2 frames per seconds from the videos to build a raw set of images.

Five trained scientists annotated all individuals on 5 to 14 sequences of 11 consecutives frames for each video. Annotation consisted in drawing a bounding box encompassing the body of each fish as well as to label its “class”, i.e. the most precise taxonomic level a

fish could be identified. Even though the species identity was not explicitly accounted for by the localisation algorithm, this information allows computing the number of individuals per species, and hence the quality of the training datasets, as well as the performance of the localisation algorithm per species. There was no annotation of the seascape, as the Deep Learning algorithm mines it during the training process.

To build our first training dataset (D1), we used 6 videos from 5 seascapes of Mayotte, containing 3603 fish individuals from 40 classes. The number of individuals per frame varied from 0 to 27, and the number of individuals present in the dataset per class varied between 1 and 882.

To build the second dataset D2, we added to this first dataset D1 660 frames of 6 others videos from 6 different seascapes. This second dataset contains 6146 individuals of 57 classes, and each class had between 1 and 1719 individuals. We repeated this step 2 times to create D3 and D4 by adding 1300 and 4114 frames from 20 and 19 videos from 10 and 10 sites, respectively. D3 contained 11230 images of 97 classes (between 1 and 2404 individuals per species) and D4 contained 17708 images of 125 classes (between 1 and 10,675 individuals per species). We observed that even if the number of annotation increased greatly between D3 and D4 (about 1/3) while the number of species increased less (about 1/4). Furthermore, due to the the rarity and abundancy of species, the training database was unbalanced (i.e. 1 annotation of *Kyphosus vaigiensis* or 3 annotations of *Chaetodon lunulatus* for 2404 annotations of *Pomacentrus sulfureus* or

10675 annotations of *Chromis dimidiata*). By counting the total of annotations and time needed to create the database, we estimated that one trained person was able to annotate up to 385 fish per hour. We used this estimation to convert each training dataset annotation number in time in Table 1.

Table 1: Summary of the training datasets, in terms of invidiual fish annotations, number of classes, and time needed to create the datasets.

Training Dataset	Annotations	Classes	Time (hour)
D1	3,603	40	9
D2	6,146	57	16
D3	11,230	97	29
D4	17,708	125	46

Testing datasets

To test the algorithm trained with videos from Mayotte and Europa islands we then build 5 additional image datasets (Table 2) coming from stationary videos recorded with the same protocol than videos used for the training. We chose to add a minimum size for individuals to be detected. This size was 1500 pixels square, which roughly correspond to 1/1300 of the image's size. We observed that smaller individuals we too far to be identified, therefore not usefull for monitoring. Testing videos were recorded on 6 locations (Figure 1): Western Indian Ocean with 12 videos recorded on a Mayotte reef different from those where the training videos were recorded, and **3** videos recorded in the Seychelles), Red Sea (4 videos from Eilat, Israel) , Mediterranean Sea (3 videos from Crete island and 2 videos from Haifa coast, Israel), South Western Pacific ocean (4 videos from Moorea island), and Caribbean Sea (4 videos from Martinique island).

The 5 testing datasets were build as following:

- The Western Indian Ocean dataset (WIO) contained 15 videos and 5431 individuals of 66 classes.
- The Red Sea dataset (RS) contained 4 videos and 2930 individuals of 28 classes, from which 19 were not included in any training dataset.
- The Southwest Pacific Ocean dataset (SPO) contained 4 videos and 2342 individuals of 36 classes, from which 17 were not included in any training dataset.

- The Caribbean Sea dataset (CS) contained 4 videos and 1866 individuals of 21 classes, from which 20 were not included in any training dataset
- The Mediterranean Sea (MS) contained 5 videos and 714 individuals of 13 classes, from which none were included in any training dataset.

Overall, testing datasets contained a total of 107 species and 139 classes, and a range of proportion of species not present in the largest training dataset D4 , from 0 in CS and MS to 2/3 in RS (Table 2).

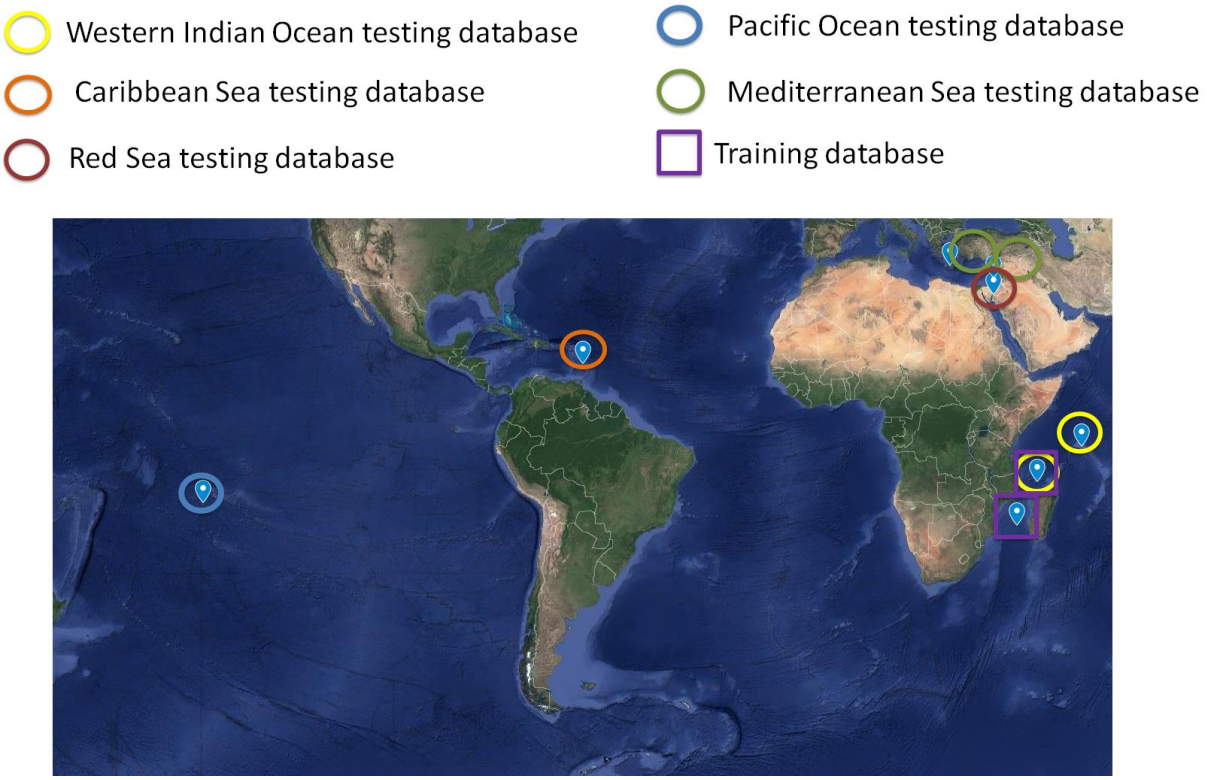


Figure 1: Locations where videos used for training and testing datasets were recorded. The Mayotte seascapes and reefs used in the testing dataset are different from those used in the training datasets.

Table 2: Summary of the testing datasets.

Testing dataset	Seascape	Number of videos	Number of fish annotated	Proportion of fish classes shared with training dataset
WIO	Dense hard corals	15	5431	24/66
RS	Scattered hard corals	4	2930	19/28
SPO	Dense hard corals	4	2342	17/36
CS	Dominated by soft corals	4	1866	20/21
MS	Dominated by algae	5	714	13/13

Like the training datasets, the testing datasets contained a few very common species (1031 annotations of *Chromis viridis* and 1964 annotations of *Ctenochaetus striatus*) and many rare species (1 *Hologymnosus annulatus* or 2 annotations of *Chrysiptera annulata*) and a median 17 individuals per class.

Algorithm

We used a state-of-the-art Deep Learning architecture, consisting of a Faster-RCNN (*faster Region-Based Convolutional Neural Network*) (20) with a NASNet (*Neural Architecture Search Network*) backbone (21). The pipeline of Faster-RCNN consists of: 1) region proposal by the network (i.e. a part of the image that is likely to be an object of interest), and 2) object identification within each proposed region. The NASnet backbone is a method where 1) the best convolutional layers (layers of operators extracting discriminant features from images) are selected with a first training dataset, 2) a new architecture is built, by stacking together copies of this layers. Therefore, the architecture of the NASnet is tuned through learning. This state-of-the-art achieved mean average precision (mAP) of 43.1% on the COCO dataset, the most used benchmark for localisation tasks in computer vision. As seen earlier, the training dataset of this algorithm consists of frames where all individuals' positions and sizes are annotated.

We used the default training parameters following guidelines (22), and each frame passed through the deep network 13 times during the training, for a total training time of 38 hours. After the training the network is able to take as input a frame, and to produce as output the same frame with predictions. These predictions are bounding boxes surrounding the detected fish individuals.

Evaluation

To lower the risk of False Positive results, we post-processed the raw output of the model by keeping only bounding boxes with classification score over 0.5. Finally, we considered a bounding box returned by the model correct if the intersection over union (IoU) with the corresponding bounding box in the ground truth (i.e. bounding box drawn by a human expert) was over 0.5 (Pascal VOC metric (23)).

Eventually, each box predicted by the model was either a :

- True Positive (TP): a fish labeled as “fish” by the model
- False Positive (FP): background labeled as “fish” by the model
- False Negative (FN): a fish labeled as “background” by the model
- True Negative (TN): background labeled as “background” by the model

We computed 3 quality metrics for each frame of the testing datasets:

Precision compute the robustness of the model to false positives and is defined as:

$$Precision = \frac{TP}{TP + FP}$$

Recall compute the robustness of the model to detect all individuals annotated in the ground truth and is defined as:

$$Recall = \frac{TP}{TP + FN}$$

The F-measure computes overall robustness of the model and is defined as:

$$Fmeasure = 2 \, . \, \frac{Precision \, . \, Recall}{Precision + Recall}$$

For each testing dataset and each model, we then averaged recall between all frames, the average precision, and the average F-measure.

Results:

The models obtained though training our Deep network with databases D1, D2, D3 and D4 are called M1, M2, M3 and M4 respectively.

Regarding the processing of the localisation of fish in the trained environment (WIO), M1 obtained a F-measure of 0.77. This F-measure went up for M2 and M3 with 0.79 and 0.81 respectively. This improvement was mainly due to the increasing recall, which went from 0.74 for M1 to 0.83 for M2 and 0.84 for M3, while the precision decreased from 0.84 to 0.81 for M2 and 0.82 for M3.

Model 4, trained with the largest training dataset (D4) reached the highest efficiency to predict fish localisation on the WIO testing dataset, with an average F-measure of 0.83. Models trained with less videos and fish annotations had F-measures lower than 0.8 (Figure 2)

Applied to the WIO testing dataset, Model 4 obtained a F-measure of 0.82 of the 4689 individuals of species present in the training dataset, and up to 0.78 of the 395 individuals belonging to the species absent from the training dataset. M4 also showed balance between recall and precision, with a recall of 0.85 (Figure 3) and a precision of 0.84 (Figure 4). M4 also reached a ratio of 34% of images processed with a F-measure of 1.

On the WIO Testing dataset, the performance of the model M4 was robust across the large range of fish size with 80% of individuals with a size smaller than 2000 square pixels correctly localised, and 85% of individuals with a size greater than 60,000 square pixels correctly localised.

We then assessed the transferability of our model (i.e. its ability to identify fish of untrained species in unseen seascapes). Model 4 reached F-measures ranging from 0.71 (Caribbean Sea) to 0.84 (Pacific Ocean), including 3 datasets analyzed with F-measures over 0.82 (Western Indian Ocean, Pacific Ocean and Mediterranean Sea). The F-measure was highly variable (Figure *2*), with confidence intervals (with α=0.05) ranging from 0.009 to 0.019 for M4.

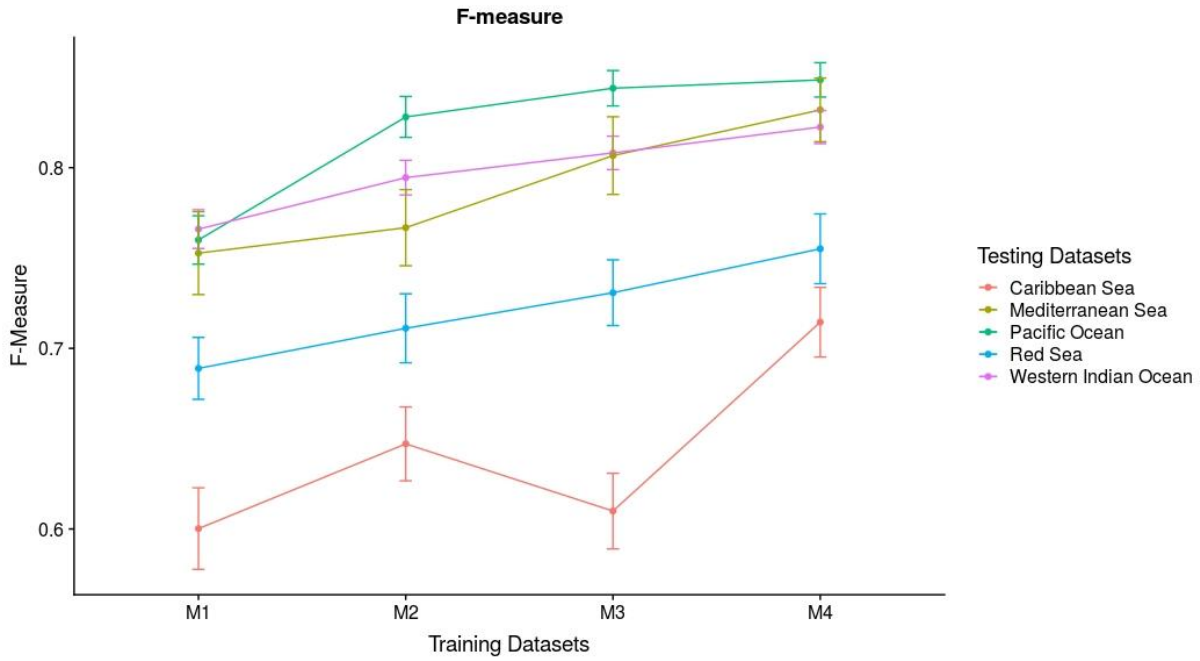


Figure 2: Average F-measure per model and testing datasets, with confidence intervals of 95%.

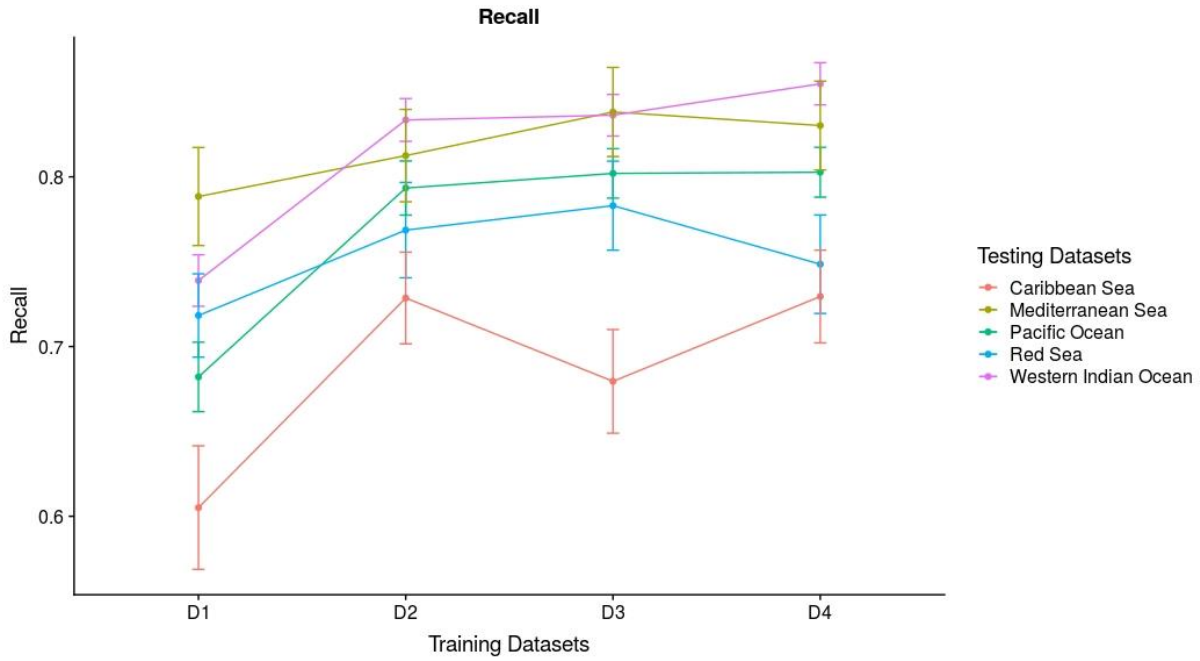


Figure 3: Average recall per model and testing datasets, with confidence intervals of 95%.

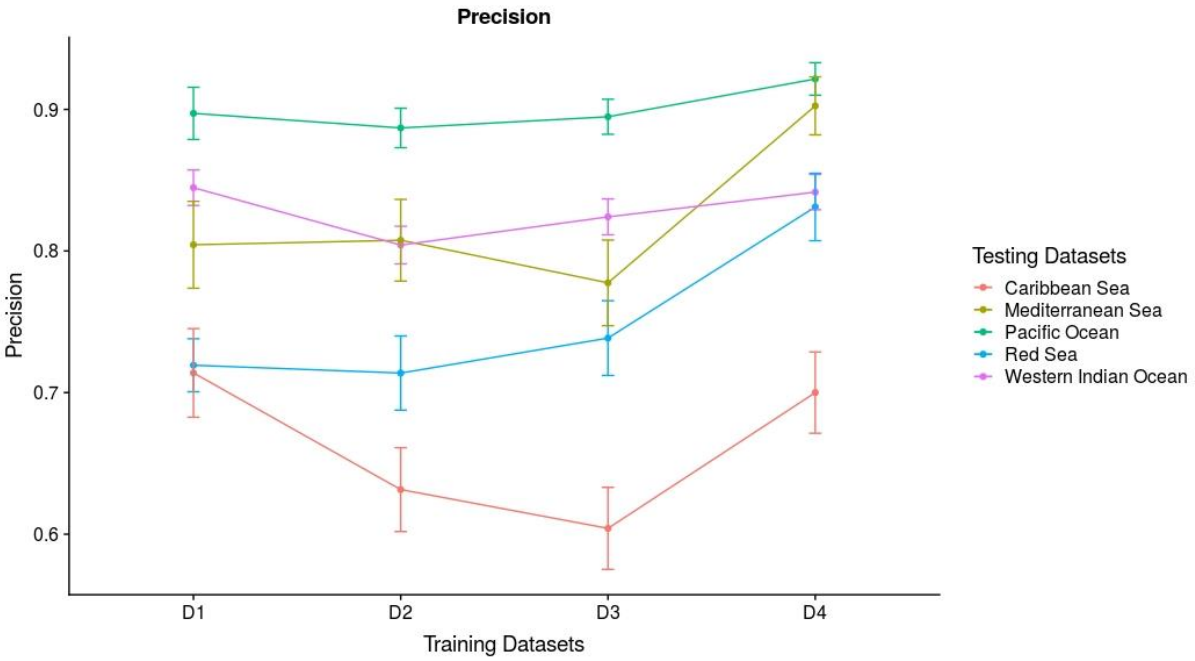


Figure 4: Average precision per model and testing datasets, with confidence intervals of 95%.

Discussion

In this paper, we showed that a Deep Learning model trained with a dataset that required less than 10h of annotation by humans was able to localise fish with a F-measure of 0.77 when the environment of the test and the environment of the training matched. We also assessed that the augmentation of the annotation number were increasing the recall (less false negatives, therefore less chances of missing individuals), while the increasing of diverse seascapes and species had more impact on the precision (less false positives, therefore less chances over estimate a fish community). It is important to note that false positives mistakes can be corrected by manual work (i.e. if a fish is detected, an expert can check a precise frame of the video to see if the prediction is correct), while false negatives can not be corrected (i.e. to prevent false negatives an expert would had to check the whole video).

The increase of species diversity (from 40 classes to 125 classes) and increase in diversity of seascape (from 6 to 28) had limited impact to process videos in a very similar environment (training on Mayotte and tested on Mayotte), but was important for the robustness and transferability of the model. This transferability implies that a “generic” fish detection algorithm can be reached with a highly diverse training.

We also showed that the F-measure curves are not converging with our 4 datasets, implying that the models can still be improved by increasing the versatility (in terms of fish communities, seascapes, algae, corals, gorgonians, but also depth, weather, turbidity…) of the training dataset.

Finally, we showed that a model trained only with video from the Western Indian Ocean was able to localize fish correctly in Mediterranean Sea, with which it shares no common classes.

We observed a considerable transferability of the model M4, trained only with videos and individuals from the Western Indian Ocean, to localise fish in the Pacific Ocean (comparable environment, 1/3 of classes in common) and in the Mediterranean Sea (different environment, no classes in common), implying the robustness of the model to detect fish individuals in seascapes absent from the training.

The SPO and RS testing dataset were closer to the training dataset in terms of species and seascapes. However, the SPO seascapes were mostly composed of dense hard corals (as in the training dataset), while the RS seascapes were composed of scattered hard corals. Some scattered hard corals were wrongly localised and identified as fish individuals, inducing a decreased precision (0.92 for SPO and 0.83 for RS)

The CS was processed with the lowest scores of F-measure, precision and recall for each model. The presence of some object present in images (such as *Gorgonacea* and sponges) induced noises, due to their lack of presence in the training datasets (Figure *5*), and produced a lot of false positives.

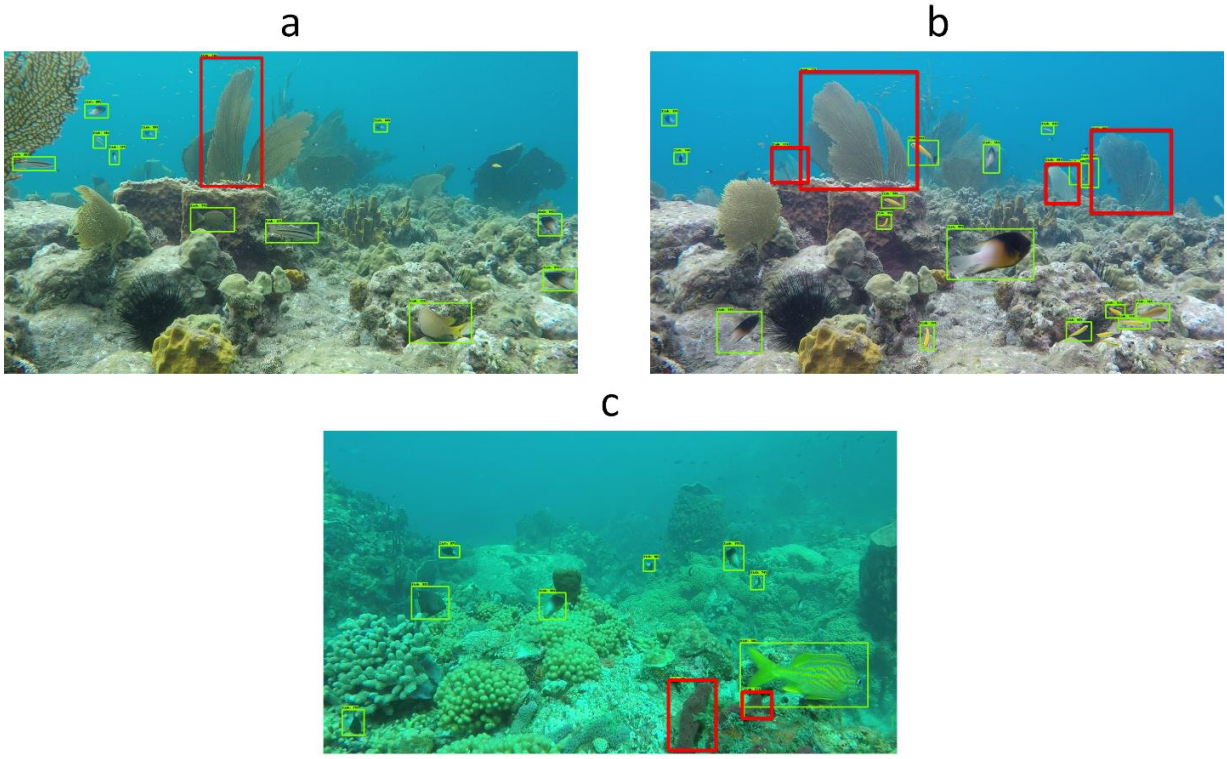


Figure 5: Gorgonians (a,b) and sponges (c) wrongly identified as fish by M4, showed with red boxes.

These false positives could be avoided by including in the training dataset frame from videos recorded in seascapes with such animals, to train the algorithm to discriminate

them from fish individuals. In general, proving the training datasets with complex seascapes can only improve the robustness of the Deep model.

The second best F-measure obtained with M4 was on the MS testing datasets, (WIO was third), while the seascapes were one of the most different from the training dataset, which was a very encouraging result.

Overall, the algorithm was able to localize fish even with important occlusion, or being partially out of the frame (Figure *6*).

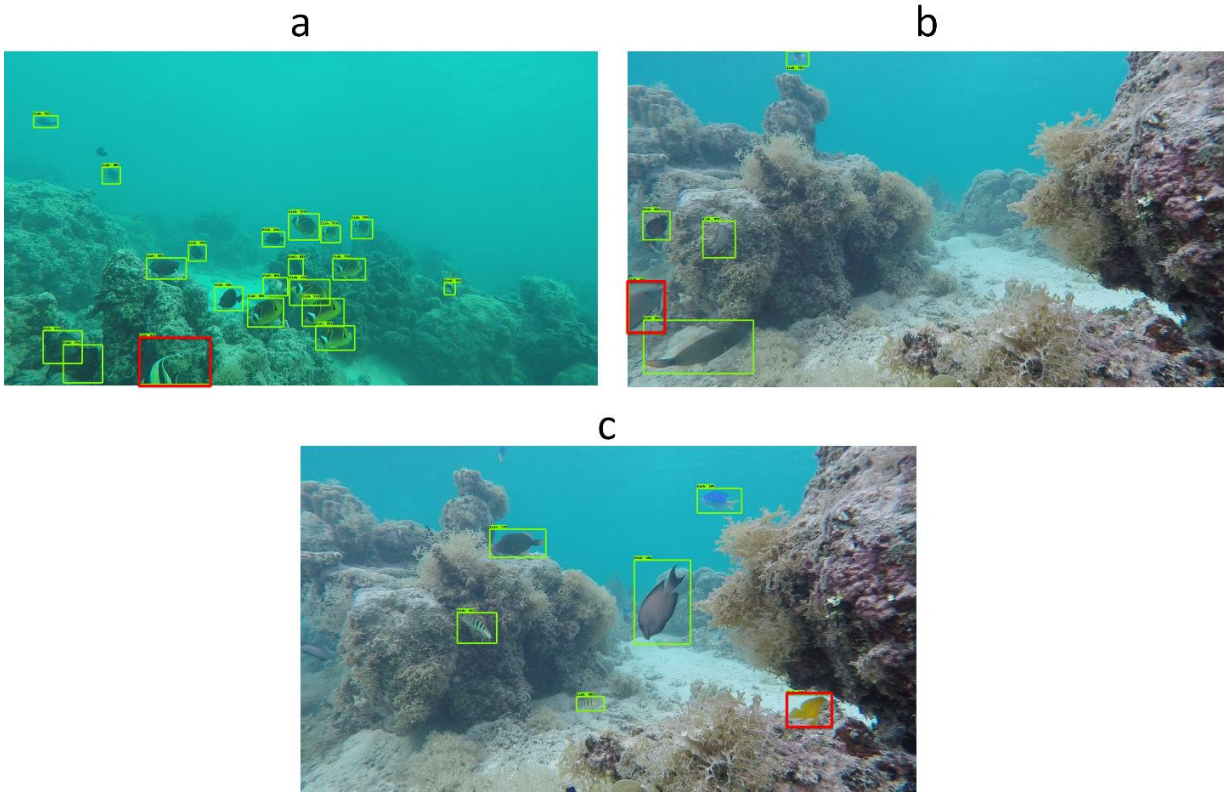


Figure 6: Model 4 was able to detect fish which body was not entirely visible (red boxes) because they were out of borders of picture (a, b) or hidden by substrate (c).

Finally, some detection were labeled as false positives due to and IoU < 0.5. This happened in particular with individuals of species with long thin fins, such *as Zanclus cornutus* or *Naso elegans* (Figure *7*). This implies that even if those bounding boxes were counted as false positives, the output of the algorithm could still be used to count fish in videos (i.e. to identify frame containing fish in a large volume of data).

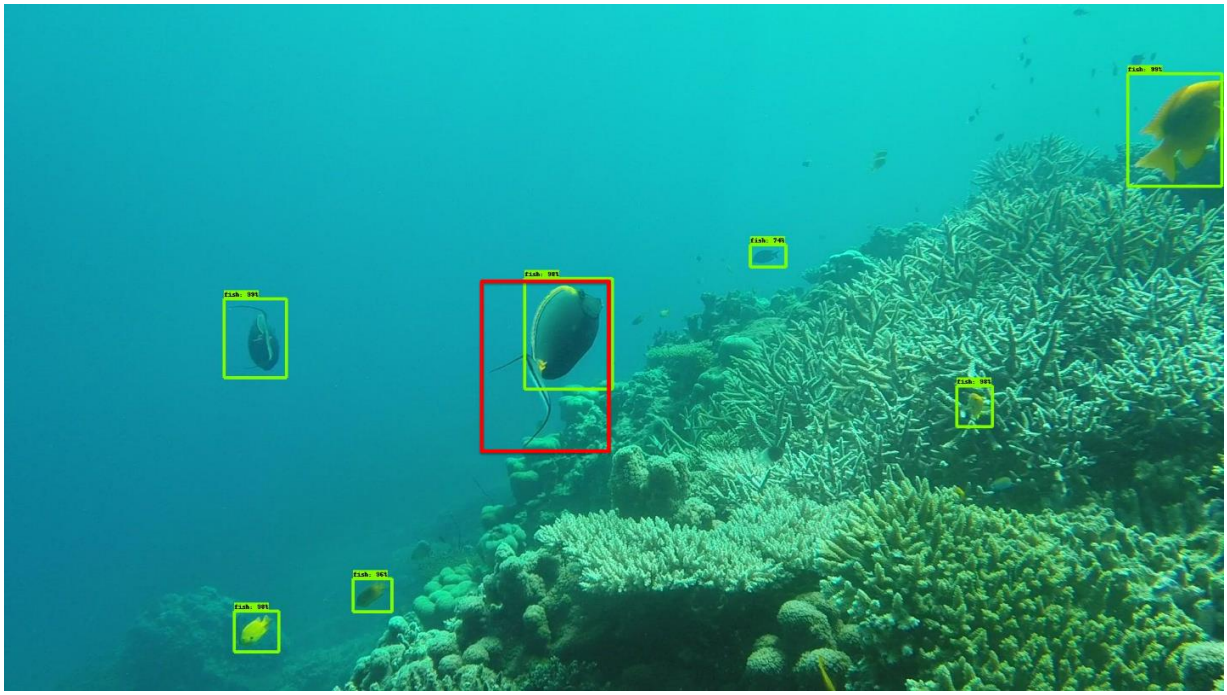


Figure 7: The detection model missed the long caudal fin of the Naso elegans in the center of the image.

The ground truth bounding (in red) of the fish and the predicted bounding box (in green) present an intersection over union (IoU) lesser than 0.5. Therefore, the prediction is labeled as a false positive, while the missed ground truth is labeled as a false negative.

This work presented encouraging results for automatic fish detection on videos. The potential use of such methods support the need of recording and annotating effort of diverse regions and seascapes on a large scale. The transferability of this method can allow users to process automatically or semi-automatically process videos. On short term, it can be used to detect part of video containing fish (e.g. baited videos attracting pelagic individuals are mostly empty from any fish, apart from specific frames). On a longer term, the association of robust detection methods and identification algorithms can be used to follow the evolution of fish communities.

References

1. *Spatial and temporal patterns of mass bleaching of corals in the Anthropocene.* Hughes, Terry P and Anderson, Kristen D and Connolly, Sean R and Heron, Scott F and Kerry, James T and Lough, Janice M and Baird, Andrew H and Baum, Julia K and Berumen, Michael L and Bridge, Tom C and others. 6371, s.l. : American Association for the Advancement of Science, 2018, Vol. 539.

2. *Natural and anthropogenic disturbance on coral reefs.* Grigg, Richard W. 1991, Coral Reef Ecosystems of the World.

3. *Habitat degradation negatively affects auditory settlement behavior of coral reef fishes.* Gordon, Timothy AC and Harding, Harry R and Wong, Kathryn E and Merchant, Nathan D and Meekan, Mark G and McCormick, Mzaark I and Radford, Andrew N and Simpson, Stephen D. 2018, Proceedings of the Notional Academy of Sciences, pp. 5193-5198.

4. *Rapid coral decay is associated with marine heatwave mortality events on reefs.* Leggat, William P and Camp, Emma F and Suggett, David J and Heron, Scott F and Fordyce, Alexander J and Gardner, Stephanie and Deakin, Lachlan and Turner, Michael and Beeching, Levi J and Kuzhiumparambil, Unnikrishnan and others. 2019, Current Biology.

5. *Fish as proxies of ecological and environmental change.* Izzo, Christopher and Doubleday, Zo{\"e} A and Grammer, Gretchen L and Gilmore, Kayla L and Alleway, Heidi K and Barnes, Thomas C and Disspain, Morgan CF and Giraldo, Ana Judith and Mazloumi, Nastaran and Gillanders, Bronwyn M. 3, Reviews in Fish Biology and Fisheries, Vol. 26, pp. 265--286.

6. *Remote monitoring of vigilance behavior in large herbivores using acceleration data.* Kröschel, Max and Reineking, Bjorn and Werwie, Felicitas and Wildi, Felix and Storch, Ilse. 1, 2017, Animal Biotelemetry, Vol. 5, p. 10.

7. *Scaling-up camera traps: Monitoring the planet's biodiversity with networks of remote sensors.* Steenweg, Robin and Hebblewhite, Mark and Kays, Roland and Ahumada, Jorge and Fisher, Jason T and Burton, Cole and Townsend, Susan E and Carbone, Chris and Rowcliffe, J Marcus and Whittington, Jesse and others. 1, 2017, Frontiers in Ecology and the Environment, Vol. 15, pp. 26--34.

8. *Satellite tracking of whale sharks from Al Shaheen.* Robinson, David P and Bach, Steffen S and Abdulrahman, Ali A and Al-Jaidah, Mohammad. 2016. The 4th International Whale Shark Conference.

9. https://globalfinprint.org. [Online]

10. *Remote reefs and seamounts are the last refuges for marine predators across the Indo-Pacific.* Letessier, Tom B and Mouillot, David and Bouchet, Phil J and Vigliola, Laurent and Fernandes, Marjorie C and Thompson, Chris and Boussarie, Germain and Turner, Jemma and Juhel, Jean-Baptiste and Maire, Eva and others. 8, s.l. : Public Library of Science San Francisco, CA USA, 2019, PLoS biology, Vol. 17, p. e3000366.

11. *Lifeclef 2017 lab overview: multimedia species identification challenges.* Joly, Alexis and Goëau, Hervé and Glotin, Hervé and Spampinato, Concetto and Bonnet, Pierre and Vellinga, Willem-Pier and Lombardo, Jean-Christophe and Planque, Robert and Palazzo, Simone and Müller, Henning. 2017. International Conference of the Cross-Language Evaluation Forum for European Languages.

12. *A Deep learning method for accurate and fast identification of coral reef fishes in underwater images.* Villon, Sébastien and Mouillot, David and Chaumont, Marc and Darling, Emily S and Subsol, Gérard and Claverie, Thomas and Villéger, Sébastien. 2018, Ecological informatics, Vol. 48, pp. 238--244.

13. *Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system.* Salman, Ahmad and Siddiqui, Shoaib Ahmad and Shafait, Faisal and Mian, Ajmal and Shortis, Mark R and Khurshid, Khawar and Ulges, Adrian and Schwanecke, Ulrich. 2019, ICES Journal of Marine Science.

14. *Rich feature hierarchies for accurate object detection and semantic segmentation.* Girshick, Ross and Donahue, Jeff and Darrell, Trevor and Malik, Jitendra. 2014. Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580--587.

15. *You only look once: Unified, real-time object detection.* Redmon, Joseph and Divvala, Santosh and Girshick, Ross and Farhadi, Ali. 2016. Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779--788.

16. *Ssd: Single shot multibox detector.* Liu, Wei and Anguelov, Dragomir and Erhan, Dumitru and Szegedy, Christian and Reed, Scott and Fu, Cheng-Yang and Berg, Alexander C. s.l. : Springer, 2016. European conference on computer vision. pp. 21--37.

17. *Fast r-cnn.* Girshick, Ross. 2015. Proceedings of the IEEE international conference on computer vision. pp. 1440--1448.

18. *Faster r-cnn: Towards real-time object detection with region proposal networks.* Ren, Shaoqing and He, Kaiming and Girshick, Ross and Sun, Jian. 2015. Advances in neural information processing systems. pp. 91--99.

19. *Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild.* Labao, Alfonso B and Naval Jr, Prospero C. s.l. : Elsevier, 2019, Ecological Informatics, Vol. 52, pp. 103--121.

20. *Faster r-cnn: Towards real-time object detection with region proposal networks.* Ren, Shaoqing and He, Kaiming and Girshick, Ross and Sun, Jian. 2015, Advances in neural information processing systems}, pp. 91--99.

21. *Learning transferable architectures for scalable image recognition.* Zoph, Barret and Vasudevan, Vijay and Shlens, Jonathon and Le, Quoc V. 2018, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8697--8710.

22. https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md. [Online]

23. *The pascal visual object classes (voc) challenge.* Everingham, Mark and Van Gool, Luc and Williams, Christopher KI and Winn, John and Zisserman, Andrew. [ed.] Springer. 2, 2010, International journal of computer vision, Vol. 88, pp. 303--338.

24. http://catlinseaviewsurvey.com/. [Online]

Chapitre 7

Conclusions et Perspectives

7.1 Rappel des principaux résultats et avancées

Les travaux développés pendant cette thèse ont prouvé l'intérêt des algorithmes d'apprentissage profond pour les tâches d'identification et de localisation de poissons dans des images sous-marines. Nous avons démontré la meilleure performance d'un modèle obtenu à partir de l'apprentissage d'un algorithme *Deep Learning* pour différencier plusieurs espèces de poissons, par rapport à d'autres méthodes d'apprentissage automatique (chapitre 3), mais aussi par rapport aux humains experts en taxonomie des poissons récifaux (chapitre 4). Nous avons aussi proposé des approches permettant d'améliorer la construction des bases de données d'entraînement, et de traiter les sorties des modèles *Deep*, afin de pouvoir éviter les erreurs de classification, et ainsi augmenter la confiance de l'utilisateur dans les prédictions de l'algorithme (chapitre 5).

Les modèles de localisation nous ont permis de mettre en évidence la robustesse et la transférabilité d'un modèle créé par apprentissage profond. Ces modèles sont capables à la fois de localiser des individus appartenant à des espèces présentes dans la base d'entraînement et dans des conditions similaires (chapitre 6), mais aussi des individus appartenant à des espèces, des genres et familles n'étant pas dans la base d'entrainement, y compris dans des contextes différents (e.g. entraînement dans un milieu recouvert de coraux "durs" et test avec des récifs dominés par les coraux "mous" ou des algues).

Néanmoins, des efforts importants restent à effectuer pour pouvoir utiliser ces méthodes dans des études à large échelle, et en particulier pour être capable d'effectuer ces analyses en temps réel, par exemple pour étudier et évaluer les récifs coralliens à haute fréquence temporelle avant et après un évènement perturbateur, ou pour suivre l'évolution d'une communauté au cours du temps suite, par exemple, à une mise en réserve.

Dans tout tentative de mise en place d'une stratégie d'identification automatique de poissons par Deep Learning, la première étape importante est l'effort d'annotation de vidéos

ou photos, afin d'augmenter le nombre d'espèces identifiables. Cet effort d'échantillonnage nécessite avant tout une planification importante (terrain, temps de vidéos, sélection des sites, etc.), ainsi qu'un nombre important d'annotateurs entrainés. Au cours de la thèse plus de 50 stagiaires (de licence 2 à Master en écologie marine) et 10 experts (Docteurs ou Doctorants en écologie marine) ont participé à l'annotation manuelle des vidéos.

En plus du travail d'annotation, ces participants doivent aussi chercher les individus appartenant à des espèces rares dans des vidéos (tâche particulièrement chronophage). Bien que les études sur le *transfert learning* c'est à dire le fait de commencer par un pré-apprentissage sur un large jeu de données générique permettant un premier apprentissage de l'algorithme, puis de faire un ré-entraînement avec un plus petit jeu de données spécialisé pour affiner le modèle, proposent des méthodes pour obtenir un modèle robuste avec un jeu d'entraînement restreint [Pan and Yang, 2009] [Torrey and Shavlik, 2010] [Ng et al., 2015], un minimum de données est nécessaire (au minimum 600 par classe), en particulier pour discriminer des espèces proches.

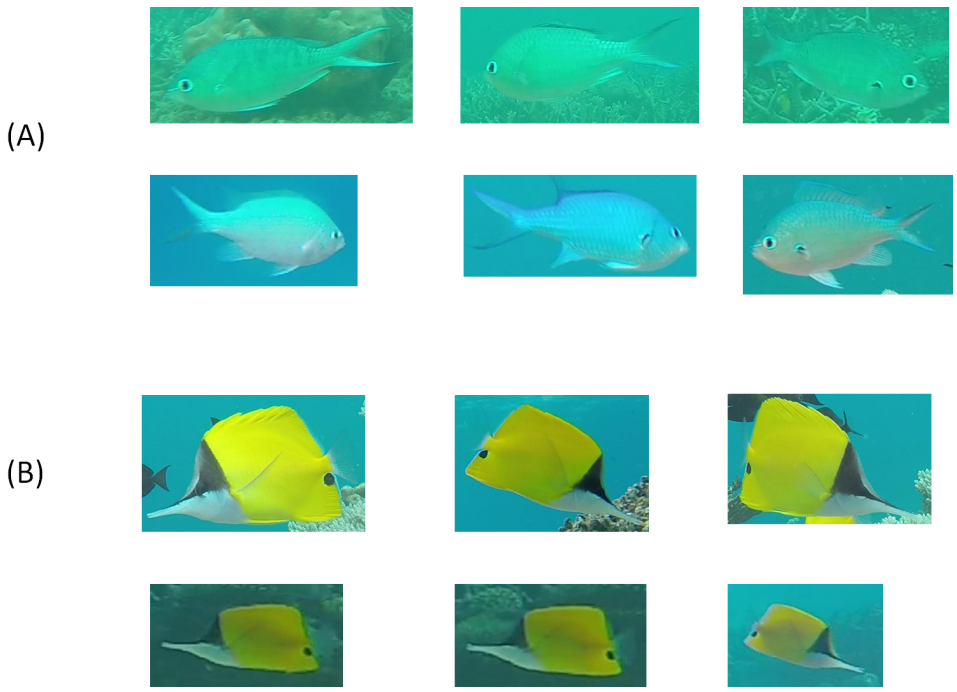


FIGURE 7.1 – Exemples d'espèces proches visuellement.
(A) : *Chromis atripectoralis* en haut et *Chromis viridis* en bas.
(B) : *Forcipiger flavissimus* en haut et *Forcipiger longirostris* en bas.
Les deux paires d'espèces paraissent très proches à l'image, et seuls quelques détails permettent de les différencier. Par exemple, dans le cas des 2 *Forcipiger*, la taille du museau, ainsi que la position exacte de la bande noire, couvrant la totalité de l'oeil chez l'un et une partie seulement chez l'autre.

Les limitations à la création de bases de données conséquentes et adaptées au deep learning pour l'identification d'espèces de poissons coralliens restent nombreux : En effet,

dans le contexte particulier de l'identification d'espèces de poissons, l'algorithme doit être capable de différencier des espèces parfois très proches visuellement (Fig. 7.1), ce qui peut demander des convolutions particulièrement fines donc un jeu d'entraînement particulièrement important composé d'images variées.

De plus, certaines espèces présentent aussi des changements de couleurs ou de motifs au cours de leur vie, notamment les poissons perroquets (*Scaridae*), et demandent donc d'avantage de données pour aboutir à un modèle robuste d'identification à l'espèce dans toute sa diversité ontogénique ou sexuelle. Finalement, de nombreuses espèces de poissons sont peu présentes lors des enregistrements vidéos, soit parce qu'elles sont peu mobiles (comme les poissons clowns) et/ou parce qu'elles sont peu abondantes sur les récifs (par exemple les carangues).

La constitution d'un jeu d'entraînement de plus en plus varié pour un grand nombre d'espèces récifales demande donc un effort exponentiel. De plus, le nombre important d'espèces à différencier vivant dans un même écosystème, ainsi que les occultations partielles d'images dues au nombre d'individus présents dans chaque image (agglomérats autour des appâts, banc de poissons...) posent des problèmes particuliers aux modèles d'identification et de localisation qui doivent apprendre à reconnaître des individus dans de nombreux et différents contextes.

Malgré ces limitations, la thèse a permis entre autre de constituer une base importante de vidéos réalisées dans de nombreux endroits (océan Ouest-Indien, océan Pacifique-Sud, Caraïbes, Méditerrané, Mer Rouge) et d'annoter plus de 200 heures de vidéos, pour obtenir une base contenant 380 000 images appartenant à 350 classes, dont 300 espèces de poissons coralliens (les autres classes correspondent à des dichotomies de sexe ou d'âge au sein d'une même espèce). Nous avons aussi mis en avant les points forts des approches de recensement par apprentissage profond, en particulier leur capacité à identifier efficacement et rapidement des individus dans des vidéos, mais aussi à localiser des individus de manière robuste dans des milieux très différents de ceux de la base d'apprentissage. Nous avons aussi pu développer de nombreux codes informatiques simples d'utilisation sous Tensorflow afin d'entraîner et d'utiliser de nouveaux réseaux de classification et de localisation pour la communauté travaillant en ichtyologie.

Bien que cette thèse ait permis de lever de nombreux verrous sur l'application de méthodes d'automatisation du traitement de vidéos sous-marines, de nombreuses améliorations restent encore à effectuer.

7.2 Association de la classification taxonomique et de l'apprentissage profond

La classification taxonomique étant le principal moyen d'identification des espèces, il est naturel de vouloir associer ces connaissances et les méthodes de classifications automatiques (et en particulier les CNNs). On pourrait donc mieux utiliser ces informations existantes soit en ajoutant des données taxonomiques lors de l'apprentissage , soit en "forçant" l'espace de classification du réseau *Deep* à correspondre à l'information taxonomique par exemple en rapprochant les espèces d'un même genre.

La principale difficulté de cette association réside dans la différence des caractéristiques utilisées par les deux approches. En effet, nous savons que les caractéristiques extraites par les algorithmes "Deep" reposent d'avantage sur des statistiques numériques (les valeurs des pixels de l'image), que sur des caractéristiques observables. Hors, la classification taxonomique repose elle sur des caractéristiques observables à l'oeil humain. L'association des deux approches repose donc sur l'intuition que les informations utilisées par la taxonomie correspondent aux informations extraites par les méthodes *Deep*.

Lors d'un travail préliminaire, nous avons proposé une architecture de multi-classifieur. Trois modèles étaient entraînés sur un même taxon à chaque niveau taxonomique (i.e. famille, genre, espèce). Lors du test, chaque image était testée par les trois modèles. Chaque modèle proposait ensuite la classe avec le plus haut score de classification. Nous avons réalisé 2 études pour tester cette stratégie :

- La première consistait en l'apprentissage de 3 modèles (Famille, Genre, Espèce) grâces à des annotations faites sur des vidéos de Mayotte, puis au test de ces modèles sur d'autres annotations provenant de l'océan Indien. Le test comprenait donc 8 classes communes avec la base d'entraînement, et 12 classes étaient inédites au test.
- La seconde était basée sur le même apprentissage, mais suivi d'un test sur des vidéos réalisées dans les Caraïbes. Le test comprenait donc uniquement des espèces inédites (13 classes), mais appartenant à 3 genres/familles représentés par d'autres espèces lors de l'apprentissage sur Mayotte (*Labridae*, *Acanthuridae*, *Pomacentridae*).

L'intérêt de la présence dans le test de certaines espèces absentes de la base d'entraînement était de vérifier la performance des modèles à prédire correctement le genre et/ou la famille d'une espèce inconnue. Pour notre premier test, nous avons constaté que les 8 classes présentes lors de l'apprentissage obtenaient de meilleurs scores de classification à l'espèce (86% en moyenne) qu'au genre (62%) ou à la famille (67%). Les espèces inconnues ont elles obtenues 39% de classification correcte au genre et 45% à la famille.

Lors du second test, nous avons obtenu en moyenne un taux de classification correct de 10% au genre et de 47% à la famille ce qui reste très faible.

Les résultats d'identification d'espèces inconnues à un taxon supérieur étaient très dépendants de la présence d'espèces semblables au sein du genre et de la famille. Ainsi, dans les familles particulièrement hétérogènes en termes de formes et de couleurs (*Labridae*), les scores d'identification correcte à la famille variaient de 12% à 85%).

Nous avons ensuite proposé un post-traitement permettant d'observer la cohérence des réponses. Si les 3 réponses étaient taxonomiquement justes (espèce $\in$ genre $\in$ famille), alors le résultat de classification à l'espèce était accepté. Si 2 réponses étaient cohérentes, alors l'identification au plus bas niveau taxonomique était acceptée, et l'utilisateur était prévenu du manque de cohérence d'une réponse. Si les 3 réponses n'étaient pas cohérentes, aucune réponse n'était donnée et l'utilisateur était prévenu. Bien que cette approche permette de limiter le nombre de faux positifs (qui sont alors "non classés"), elle ne permet pas d'augmenter le nombre de vrais positifs au niveau de l'espèce.

Aujourd'hui, d'autres études essayent de rapprocher les méthodes d'apprentissage profond et la connaissance taxonomique [Goo et al., 2016] [dos Santos and Gonçalves, 2019]. En particulier, [Goo et al., 2016] généralise le problème de la taxonomie et essayant de reproduire le comportement humain :

Lors de l'identification d'espèces, l'oeil humain se concentre en priorité sur certaines caractéristiques permettant de différencier les grandes familles (pour les espèces de poissons, les formes et la taille sont souvent discriminatoires entre familles), avant de focaliser sur de plus petits détails (motifs, couleurs) afin de différencier les espèces d'une même famille. Ce mode de fonctionnement a été repris par [Goo et al., 2016], qui propose de diviser un réseau profond en 2 phases. La première phase est une phase de généralisation, au cours de laquelle le réseau va extraire les informations partagées par plusieurs sous-catégories de la même super-classe. Cette phase de généralisation est suivie d'une phase de spécialisation, au cours de laquelle le réseau va apprendre à extraire les caractéristiques discriminant les individus appartenant à la même super-classe. Cet algorithme a ensuite été testé sur 3 *benchmarks*, en particulier "ImageNet 22K Animal", une sous partie du jeu de données ImageNet contenant 2 266 classes d'animaux (pour 1,6M images d'entraînement, et 282K images de test). Bien que les résultats ne dépassent ceux d'*AlexNet* [Krizhevsky et al., 2012] que d'1%, ces premiers résultats d'utilisation d'informations sémantiques pour la classification d'espèces animales sont prometteurs, et pourraient être utilisés pour certaines familles de poissons visuellement homogènes.

Une autre méthode pour utiliser cette information sémantique serait de forcer l'espace de représentation des cartes de caractéristiques pour qu'il soit fidèle à la classification taxonomique. En effet, grâce à une visualisation permettant de transformer l'espace multi-dimensionnel du vecteur obtenu par notre architecture profonde en 2 dimensions

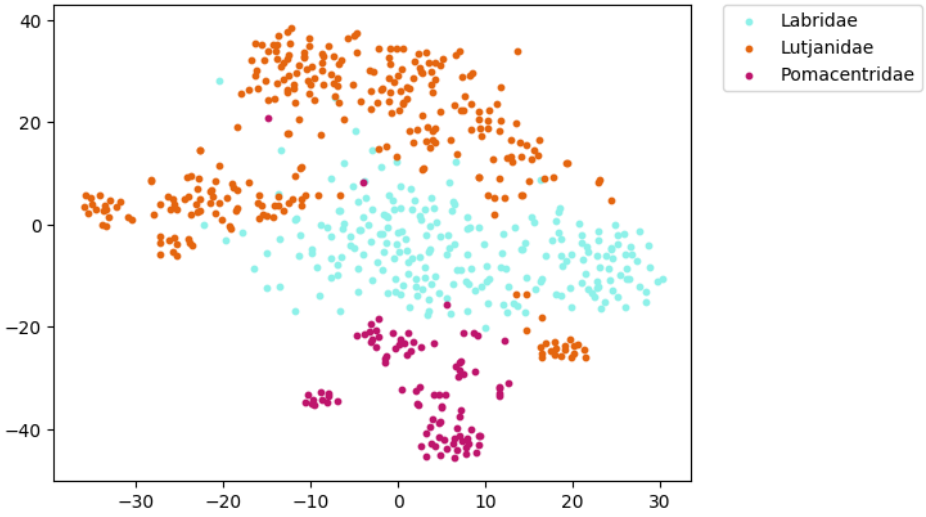


FIGURE 7.2 – Représentation de la répartition de 3 familles dans l'espace multidimensionnel du modèle obtenu par apprentissage profond.
Chaque point représente le vecteur caractéristique d'un individu après avoir été traité par le réseau profond. Un algorithme t-SNE est appliqué sur l'ensemble des vecteurs obtenus dans la dernière couche caché du modèle, afin de les transformer, pour pouvoir les représenter dans l'espace 2D. Le but de cette transformation est de représenter la distance des points dans l'espace multidimensionnel, i.e. considérant 3 points, les 2 plus proches dans l'espace multidimensionnel doivent être plus proches dans l'espace bi-dimensionel.

(algorithme *t-distributed stochastic neighbor embedding*, t-SNE [Maaten and Hinton, 2008]), nous avons constaté que des regroupements cohérents étaient visibles (Fig. 7.2). Ainsi, une approche permettant d'appuyer cette constitution de groupes (*clusters*) homogènes par famille ou genre pourrait aider à la tâche de classification hiérarchique [Law et al., 2017], mais permettrait aussi de détecter les individus qui sont sur les bords des *clusters* (possiblement des individus d'espèces non apprises, ou présentant des caractéristiques inhabituelles, tels qu'un motif unique ou une blessure).

7.3 Résoudre le problème d'équilibrage des classes

Comme nous l'avons vu dans le chapitre 6, le principal problème pour réaliser la tâche de localisation et d'identification de poissons avec un algorithme unique est le problème d'équilibrage des classes. Pour remédier à ce problème, l'architecture de RetinaNet [Lin et al., 2017] propose une modification de la *loss*. La *loss* est une fonction qui, pour un ensemble d'images traitées en une fois par un réseau profond lors de l'apprentissage (on parle alors de paquets d'images, ou *batchs*), somme les erreurs faites pour chacun des exemples (ainsi, plus la fonction de *loss* est faible, moins le réseau fait d'erreure). Le réseau cherche donc à minimiser cette fonction pendant l'entraînement. La fonction de *loss*

classiquement utilisée est appelée *loss Cross Enthtropy* (*CE loss*), que l'on peut définir ainsi, pour une classification binaire lors d'une tâche de détection :

$$p_t = \begin{cases} -p & \text{si } y = 1 \\ 1 - p & \text{sinon.} \end{cases}$$

$$CE(p, y) = CE(p_t) = -log(p_t)$$

Avec $y = -1$ (environnement) ou 1 (classe d'intérêt) la classe de la vérité terrain et $p \in [0, 1]$ le score de classification. Cependant, pour effectuer une tâche de localisation, le nombre de régions traitées par l'algorithme n'appartenant à aucune classe d'intérêt (qui peut être vue comme le "fond" de l'image, ou "l'environnement"), est beaucoup plus important que le nombre de régions contenant des objets d'intérêts. Avec ce déséquilibre et la formulation de la *loss* à minimiser, l'apprentissage de l'algorithme peut être biaisé. En effet, si l'algorithme traite 100 fois plus d'exemples "faciles" (i.e avec des scores de classification élevés, $p_t \longrightarrow 1$) correctement, mais commet des erreurs sur les exemples "difficiles" (i.e. avec des scores de classification faibles, $p_t \longrightarrow 0$), alors la *loss* sera basse, et la valeur des poids du réseau sera très peu modifié, ne permettant donc pas l'amélioration du réseau pour juger les exemples "difficiles". Les auteurs proposent donc une nouvelle formulation de la *loss*, (*Focal Loss*, FL) [Lin et al., 2017] :

$$FL(p_t) = -(1 - p_t)^{\gamma} log(p_t)$$

Avec γ le paramètre de *focus*, défini par l'utilisateur. Ainsi, avec des exemples "difficiles", la valeur de la *loss* reste très peu changée. En revanche, avec des exemples "faciles" la valeur de la *loss* sera diminuée (facteur 100 avec $p_t = 0.9$, facteur 1000 avec $p_t \approx 0.968$). Adaptée à un modèle multi-classes, l'architecture proposée par [Lin et al., 2017], avec $\gamma = 2$, améliore le score moyen de précision (mAP) sur COCO de 2.3% par rapport à l'architecture Faster R-CNN le plus récente [Shrivastava et al., 2016].

Cependant, l'architecture associant le Faster R-CNN à une architecture NAS [Zoph et al., 2018] (utilisée et expliquée dans notre chapitre 6 obtient désormais de meilleurs résultats avec 43,2% de précision moyenne sur le jeu de données COCO, contre 40,8% pour RetinaNet. Une combinaison des deux approches ainsi qu'une amélioration du paramètre de *focus*, qui pourrait correspondre à la proportion d'une classe dans le jeu d'entraînement, pourrait encore améliorer la robustesse moyenne de l'apprentissage des réseaux profonds avec des bases d'apprentissage fortement déséquilibrées entre les classes.

Il est cependant important de noter que dans le cadre d'études sur la biodiversité des poissons récifaux, l'amélioration "moyenne" des métriques obtenues par un modèle profond n'est pas nécessairement le but recherché. En effet il serait incohérent de sacrifier la

détection de quelques individus rares mais écologiquement importants (fonctions uniques, grands prédateurs...), pour améliorer la détection d'un nombre plus important d'espèces très communes mais moins critiques au niveau écologique (faible biomasse, fonctions partagées avec d'autres espèces...).

7.4 Utiliser le potentiel du *Big data*

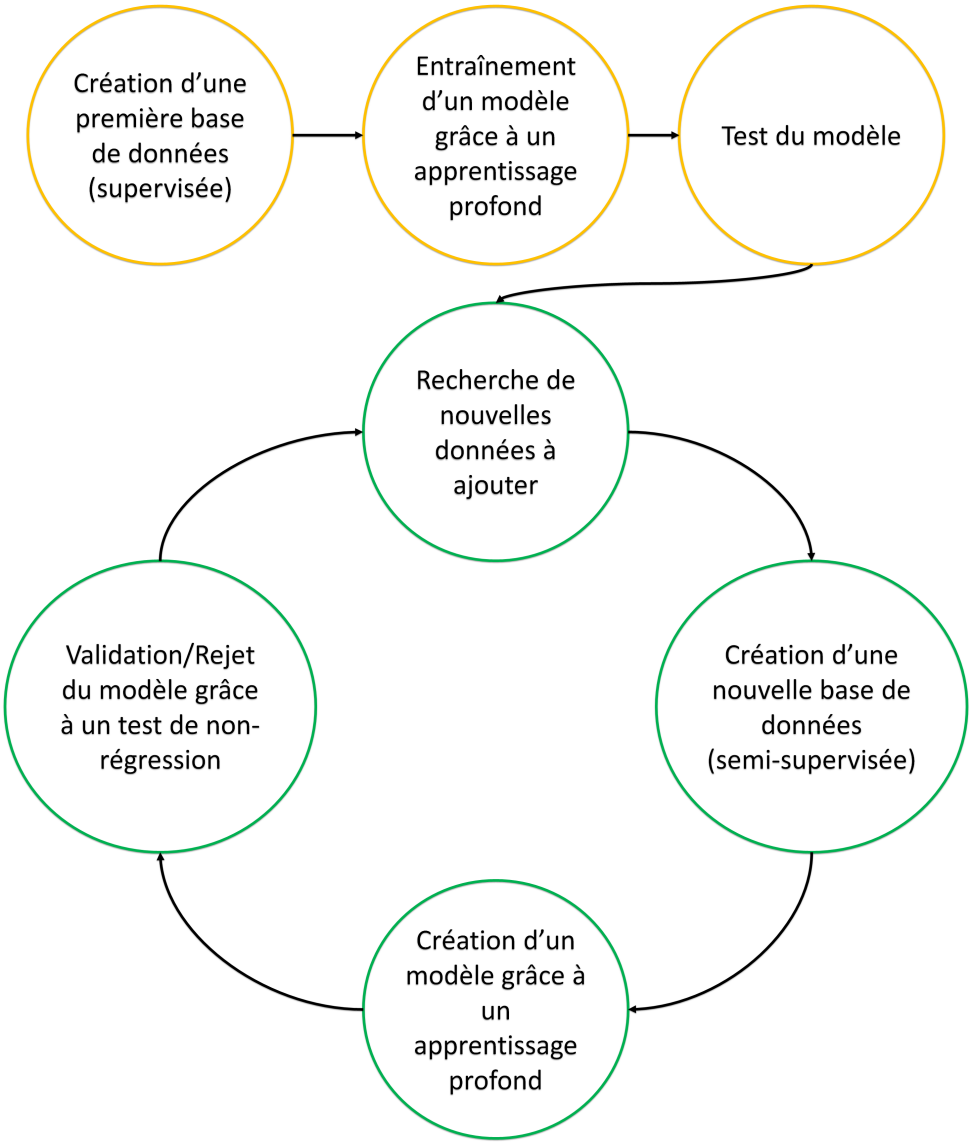


FIGURE 7.3 – Cheminement de l'apprentissage semi-supervisé.
En orange, la phase d'initialisation consiste à entraîner un réseau d'identification de manière supervisée et à le tester pour le valider. En vert, la phase itérative consiste à rechercher dans une base de données non annotée (images issues de vidéos) des vignettes pour les espèces manquant de données d'entraînement, puis de reconstruire un nouveau modèle grâce à ces données, et de vérifier que l'ajout de ces nouvelles vignettes améliore la performance globale du modèle.

À plus court terme, une autre solution est envisageable pour augmenter le nombre d'annotations pour les classes déjà présentes dans la base d'entraînement mais sous-représentées, c'est à dire peu d'images par rapport à la moyenne d'images par classe du jeu d'entraînement. Grâce à l'utilisation d'outils comme présentés dans les chapitres 5 et 6, il est possible de traiter automatiquement un jeu complet de données vidéo. Ainsi, l'algorithme

peut récupérer automatiquement des vignettes d'individus d'espèces sous-représentées avec un très haut taux de confiance, les ajouter au jeu d'entraînement, et effectuer un nouvel apprentissage (Fig. 7.3). À la fin de l'apprentissage, le nouveau modèle peut être à nouveau testé afin de réaliser un test de non-régression, en comparant les résultats obtenus par les 2 modèles sur le même jeu de données test et en vérifiant qu'ils n'ont pas été dégradés. En effectuant itérativement des étapes d'apprentissage et des tests de non-régression, il est possible de compléter le jeu d'entraînement pour les espèces les moins représentées. Parallèlement, un traitement peut aussi être effectué pour rechercher des individus d'espèces non présentes dans le jeu d'apprentissage, mais dont les familles ou les genres sont eux représentés. On peut alors entraîner un modèle d'identification de genres ou de familles qui pré-traitera de grands volumes de données vidéos en proposant uniquement aux experts certaines séquences d'intérêt à annoter manuellement.

Une autre approche d'apprentissage faiblement supervisée est aussi possible, en suivant le même schéma que le précèdent (Fig. 7.3), mais en remplaçant la recherche automatisée de vignettes dans les vidéos par une recherche automatisée de vignettes sur Internet. Lors d'une étude préliminaire réalisée en association avec des étudiants de Master 1 informatique, nous avons implémenté un algorithme de fouille, afin d'effectuer une analyse syntaxique (*parsing*) du code source de nombreux sites web pour récupérer des images et le label qui leur est associé. Un filtrage par un premier réseau binaire permet de sélectionner uniquement les images de poissons et de les conserver, avec leur label ou annotation.

En partant d'un premier modèle entraîné avec une base supervisée, nous avons ensuite effectué une succession d'étapes d'apprentissage et de test de non régression. Nous n'avons malheureusement pas eu assez de temps pour tester l'ensemble du processus, et la précision du modèle de départ (86%) n'a jamais été dépassée. Les principales difficultés viennent notamment de la variabilité des sources d'acquisition et de la qualité des images, des traitements d'images non contrôlés (balance de couleurs, etc), des conditions de la photo (naturelles, avec flash, en milieu marin, en aquarium), et du manque d'images pour de nombreuses espèces. Les limitations dues à ces variations pourraient être surmontées avec l'accumulation d'un très important volume de données, rendant chaque variation statistiquement faible et diminuant le bruit apporté par celles-ci.

7.5 Ajout d'information pour renforcer les modèles *Deep Learning*

Une autre façon de renforcer l'apprentissage des réseaux de neurones profonds est l'ajout d'information au sein des images traitées par le réseau. Si toutes les méthodes présentées dans cette thèse utilisent les 3 canaux de couleurs RGB (donc 3 matrices de

valeurs par images), il est possible d'ajouter d'autres couches d'information.

Ajout de canaux d'information dans les images.

Les méthodes d'imagerie terrestre se penchent donc sur l'utilisation de données multi-spectrales afin d'augmenter le nombre de données présentes par image [Liu et al., 2016a]. Si certaines méthodes de détection d'espèces marines sur des vidéos acquises par drone peuvent utiliser l'espace couleur proche infra-rouge pour augmenter l'information traitable par le réseau [Gray et al., 2019], ces méthodes ne sont pas utilisables pour des vidéos sous-marines.

Les méthodes d'acquisition par stéréo-caméra en revanche, peuvent fournir des cartes de profondeurs (valeurs numériques représentant le relief d'une image). La carte de profondeur d'une image peut alors être intégrée en tant que canal supplémentaire à traiter. De la même façon, les informations issues d'écho-sondeurs peuvent être associées aux images en tant que couches supplémentaires.

Couplage des méthodes de suivie et de détection automatique.

Associer des méthodes de suivis (*tracking*) de poissons [Li et al., 2018] et des méthodes de détections/identifications par apprentissage profond pourrait aussi être efficace. Alors que le modèle créé par apprentissage profond localise tous les individus à chaque image de la vidéo indépendamment (les détections effectuées sur une image à un temps t n'ont aucune incidence sur les détections faites pour l'image à un temps $t+1$), le *tracker* permet de suivre temporellement chaque objet le long de la vidéo, profitant ainsi d'une des caractéristiques de la vidéo que les CNNs classiques n'exploitent pas. Ce suivi temporel permettrait ainsi de traiter les fautes de détection (comme un individu non détecté dans une image alors que le *traker* avait prévu sa position au sein de l'image), ou d'identification (e.g. un individu identifié en tant qu'espèce A pendant 10 images soudain identifié en tant qu'espèce B). D'un autre coté, l'initialisation du *tracker* par le réseau de neurone permettrait d'automatiser cette tâche (habituellement remplie par un humain).

7.6 Vers des applications de localisation et d'identifications automatiques en écologie

Les algorithmes d'identification et de localisation de poissons ont le potentiel de contribuer à relever de nombreux challenges importants en écologie marine :

- (1) La détection et le comptage d'une ou de plusieurs espèces invasives dans une région

- (2) La détection et le comptage d'une ou de plusieurs espèces emblématiques/commerciales dans une région
- (3) La détection d'espèces potentiellement dangereuses sur les rivages (requins)
- (4) La détection de comportements (broutage, agressivité, reproduction)
- (5) Étudier l'évolution de la communauté d'un écosystème à court terme (heure, cycle jour/nuit)
- (6) Suivre l'évolution de la communauté de poissons d'une région dans le temps à grande échelle (mois/années)
- (7) Étudier l'influence de mesures de protections appliquées à une région

Tous ces cas d'application impliquent des contraintes spécifiques. Ainsi, les cas (1) et (3) impliquent de détecter des espèces dans un contexte où elles n'ont jamais été vues ou ont rarement évolué auparavant donc l'apprentissage peut être délicate, d'où la nécessité d'intensifier nos efforts sur la transférabilité. Les cas (2) et (4) demandent un effort d'acquisition important (de nombreuses caméras, et une large couverture géographique) à haute fréquence temporelle, ainsi qu'un traitement automatique en temps réel pour fournir des informations pertinentes (savoir quand un évènement à lieu avant qu'il ne soit terminé par exemple). Les cas (5), (6) et (7) demandent eux aussi une acquisition très importante en terme de volume vidéo, mais aussi un effort d'annotation conséquent dans de nombreuses conditions, afin d'augmenter le nombre d'espèces apprises par le réseau. D'autres problèmes matériels peuvent aussi limiter l'acquisition : dépôt (*fouling*) se formant sur les objectifs des caméras sous-marines, captures vidéo nocturnes, alimentation énergétique des caméras... Cependant, les progrès récents sur les véhicules sous-marins autonomes (AUV) [Xiang et al., 2018] et les caméras sous-marines [Bosch et al., 2015], ainsi que la diminution des coûts des caméras et des caissons [Bergshoeff et al., 2017] offrent de plus en plus de solutions pour observer la biodiversité marine à large échelle et à moindre coût. Le goulot d'étranglement sera donc l'analyse automatisée, robuste et rapide des vidéos.

Si les obstacles à surmonter sont encore nombreux pour élargir ce goulot d'étranglement, les récentes avancés du traitement vidéo, ainsi que de diverses méthodes complémentaires déjà proposées pour identifier les espèces, telles que les méthodes acoustiques [Davison et al., 2015] [Bolgan et al., 2018] ou basées sur l'ADN (*environmental DNA*, eDNA) [Lacoursière-Roussel et al., 2016] [Bakker et al., 2017], permettant de détecter des espèces non présentes à l'écran (cryptobenthiques, comme les gobies), sont autant de promesses pour aller vers une analyse automatique des communautés sous-marines.

Bibliographie

[Abadi et al., 2016] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow : A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

[Ackerman and Bellwood, 2000] Ackerman, J. L. and Bellwood, D. R. (2000). Reef fish assemblages : a re-evaluation using enclosed rotenone stations. *Marine Ecology Progress Series*, 206 :227–237.

[Acuña-Marrero et al., 2018] Acuña-Marrero, D., Smith, A. N., Salinas-de León, P., Harvey, E. S., Pawley, M. D., and Anderson, M. J. (2018). Spatial patterns of distribution and relative abundance of coastal shark species in the galapagos marine reserve. *Marine Ecology Progress Series*, 593 :73–95.

[Alarcón-Nieto et al., 2018] Alarcón-Nieto, G., Graving, J. M., Klarevas-Irby, J. A., Maldonado-Chaparro, A. A., Mueller, I., and Farine, D. R. (2018). An automated barcode tracking system for behavioural studies in birds. *Methods in Ecology and Evolution*, 9(6) :1536–1547.

[Allsopp et al., 2008] Allsopp, M., Pambuccian, S. E., Johnston, P., and Santillo, D. (2008). *State of the World's Oceans*. Springer Science & Business Media.

[Andradi-Brown et al., 2016] Andradi-Brown, D. A., Macaya-Solis, C., Exton, D. A., Gress, E., Wright, G., and Rogers, A. D. (2016). Assessing caribbean shallow and mesophotic reef fish communities using baited-remote underwater video (bruv) and diver-operated video (dov) survey techniques. *PloS one*, 11(12) :e0168235.

[Assis et al., 2007] Assis, J., Narvaez, K., and Haroun, R. (2007). Underwater towed video : a useful tool to rapidly assess elasmobranch populations in large marine protected areas. *Journal of Coastal Conservation*, 11(3) :153–157.

[Bakker et al., 2017] Bakker, J., Wangensteen, O. S., Chapman, D. D., Boussarie, G., Buddo, D., Guttridge, T. L., Hertler, H., Mouillot, D., Vigliola, L., and Mariani, S. (2017). Environmental dna reveals tropical shark diversity in contrasting levels of anthropogenic impact. *Scientific reports*, 7(1) :16886.

[Barnes, 1952] Barnes, H. (1952). Underwater television and marine biology. *American Scientist*, 40(4) :679–681.

[Barnes, 1955] Barnes, H. (1955). Underwater television and research in marine biology, bottom topography and geology. *Deutsche Hydrografische Zeitschrift*, 8(6) :213–236.

[Bellwood et al., 2011] Bellwood, D. R., Hoey, A. S., and Hughes, T. P. (2011). Human activity selectively impacts the ecosystem roles of parrotfishes on coral reefs. *Proceedings of the Royal Society B : Biological Sciences*, 279(1733) :1621–1629.

[Bellwood et al., 2004] Bellwood, D. R., Hughes, T. P., Folke, C., and Nyström, M. (2004). Confronting the coral reef crisis. *Nature*, 429(6994) :827.

[Bergshoeff et al., 2017] Bergshoeff, J. A., Zargarpour, N., Legge, G., and Favaro, B. (2017). How to build a low-cost underwater camera housing for aquatic research. *Facets*, 2(1) :150–159.

[Bicknell et al., 2016] Bicknell, A. W., Godley, B. J., Sheehan, E. V., Votier, S. C., and Witt, M. J. (2016). Camera technology for monitoring marine biodiversity and human impact. *Frontiers in Ecology and the Environment*, 14(8) :424–432.

[Blanc et al., 2014] Blanc, K., Lingrand, D., and Precioso, F. (2014). Fish species recognition from video using svm classifier. In *Proceedings of the 3rd ACM International Workshop on Multimedia Analysis for Ecological Data*, pages 1–6. ACM.

[Boada et al., 2015] Boada, J., Arthur, R., Farina, S., Santana, Y., Mascaró, O., Romero, J., and Alcoverro, T. (2015). Hotspots of predation persist outside marine reserves in the historically fished mediterranean sea. *Biological Conservation*, 191 :67–74.

[Board et al., 2002] Board, O. S., Council, N. R., et al. (2002). *Effects of trawling and dredging on seafloor habitat*. National Academies Press.

[Bolgan et al., 2018] Bolgan, M., Amorim, M. C. P., Fonseca, P. J., Di Iorio, L., and Parmentier, E. (2018). Acoustic complexity of vocal fish communities : a field and controlled validation. *Scientific reports*, 8(1) :10559.

[Boom et al., 2012a] Boom, B. J., Huang, P. X., Beyan, C., Spampinato, C., Palazzo, S., He, J., Beauxis-Aussalet, E., Lin, S.-I., Chou, H.-M., Nadarajan, G., et al. (2012a). Long-term underwater camera surveillance for monitoring and analysis of fish populations. *VAIB12*.

[Boom et al., 2012b] Boom, B. J., Huang, P. X., He, J., and Fisher, R. B. (2012b). Supporting ground-truth annotation of image datasets using clustering. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 1542–1545. IEEE.

[Bosch et al., 2015] Bosch, J., Gracias, N., Ridao, P., and Ribas, D. (2015). Omnidirectional underwater camera design and calibration. *Sensors*, 15(3) :6033–6065.

[Boureau et al., 2010] Boureau, Y.-L., Ponce, J., and LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118.

[Brandl et al., 2018] Brandl, S. J., Goatley, C. H., Bellwood, D. R., and Tornabene, L. (2018). The hidden half : ecology and evolution of cryptobenthic fishes on coral reefs. *Biological Reviews*, 93(4) :1846–1873.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1) :5–32.

[Brock, 1982] Brock, R. E. (1982). A critique of the visual census method for assessing coral reef fish populations. *Bulletin of Marine Science*, 32(1) :269–276.

[Brock, 1954] Brock, V. E. (1954). A preliminary report on a method of estimating reef fish populations. *The Journal of Wildlife Management*, 18(3) :297–308.

[Brooks et al., 2011] Brooks, E. J., Sloman, K. A., Sims, D. W., and Danylchuk, A. J. (2011). Validating the use of baited remote underwater video surveys for assessing the diversity, distribution and abundance of sharks in the bahamas. *Endangered Species Research*, 13(3) :231–243.

[Butchart et al., 2010] Butchart, S. H., Walpole, M., Collen, B., Van Strien, A., Scharlemann, J. P., Almond, R. E., Baillie, J. E., Bomhard, B., Brown, C., Bruno, J., et al. (2010). Global biodiversity : indicators of recent declines. *Science*, 328(5982) :1164–1168.

[Caldwell et al., 2016] Caldwell, Z. R., Zgliczynski, B. J., Williams, G. J., and Sandin, S. A. (2016). Reef fish survey techniques : assessing the potential for standardizing methodologies. *PloS one*, 11(4) :e0153066.

[Cesar et al., 2003] Cesar, H., Burke, L., and Pet-Soede, L. (2003). The economics of worldwide coral reef degradation. Technical report, Cesar environmental economics consulting (CEEC).

[Christensen et al., 2018] Christensen, J. H., Mogensen, L. V., Galeazzi, R., and Andersen, J. C. (2018). Detection, localization and classification of fish and fish species in poor conditions using convolutional neural networks. In *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pages 1–6. IEEE.

[Christie et al., 2016] Christie, K. S., Gilbert, S. L., Brown, C. L., Hatfield, M., and Hanson, L. (2016). Unmanned aircraft systems in wildlife research : current and future applications of a transformative technology. *Frontiers in Ecology and the Environment*, 14(5) :241–251.

[Cinner et al., 2018] Cinner, J. E., Maire, E., Huchery, C., MacNeil, M. A., Graham, N. A., Mora, C., McClanahan, T. R., Barnes, M. L., Kittinger, J. N., Hicks, C. C., et al. (2018). Gravity of human impacts mediates coral reef conservation gains. *Proceedings of the National Academy of Sciences*, 115(27) :E6116–E6125.

[Colton and Swearer, 2010] Colton, M. A. and Swearer, S. E. (2010). A comparison of two survey methods : differences between underwater visual census and baited remote underwater video. *Marine Ecology Progress Series*, 400 :19–36.

[Colvocoresses and Acosta, 2007] Colvocoresses, J. and Acosta, A. (2007). A large-scale field comparison of strip transect and stationary point count methods for conducting length-based underwater visual surveys of reef fish populations. *Fisheries Research*, 85(1-2) :130–141.

[Darling et al., 2019] Darling, E. S., McClanahan, T. R., Maina, J., Gurney, G. G., Graham, N. A., Januchowski-Hartley, F., Cinner, J. E., Mora, C., Hicks, C. C., Maire, E., et al. (2019). Social–environmental drivers inform strategic management of coral reefs in the anthropocene. *Nature Ecology & Evolution*, pages 1–10.

[Davison et al., 2015] Davison, P. C., Koslow, J. A., and Kloser, R. J. (2015). Acoustic biomass estimation of mesopelagic fish : backscattering from individuals, populations, and communities. *ICES Journal of Marine Science*, 72(5) :1413–1424.

[Dibble, 1991] Dibble, E. D. (1991). A comparison of diving and rotenone methods for determining relative abundance of fish. *Transactions of the American Fisheries Society*, 120(5) :663–666.

[Dickens et al., 2011] Dickens, L. C., Goatley, C. H., Tanner, J. K., and Bellwood, D. R. (2011). Quantifying relative diver effects in underwater visual censuses. *PloS one*, 6(4) :e18965.

[Dimitrov, 2002] Dimitrov, R. S. (2002). Confronting nonregimes : science and international coral reef policy. *The Journal of Environment & Development*, 11(1) :53–78.

[Doray et al., 2007] Doray, M., Josse, E., Gervain, P., Reynal, L., and Chantrel, J. (2007). Joint use of echosounding, fishing and video techniques to assess the structure of fish aggregations around moored fish aggregating devices in martinique (lesser antilles). *Aquatic Living Resources*, 20(4) :357–366.

[Dorman et al., 2012] Dorman, S. R., Harvey, E. S., and Newman, S. J. (2012). Bait effects in sampling coral reef fish assemblages with stereo-bruvs. *PLoS One*, 7(7) :e41538.

[dos Santos and Gonçalves, 2019] dos Santos, A. A. and Gonçalves, W. N. (2019). Improving pantanal fish species recognition through taxonomic ranks in convolutional neural networks. *Ecological Informatics*, page 100977.

[D'agata et al., 2014] D'agata, S., Mouillot, D., Kulbicki, M., Andréfouët, S., Bellwood, D. R., Cinner, J. E., Cowman, P. F., Kronen, M., Pinca, S., and Vigliola, L. (2014). Human-mediated loss of phylogenetic and functional diversity in coral reef fishes. *Current Biology*, 24(5) :555–560.

[Edgar et al., 2018] Edgar, G. J., Ward, T. J., and Stuart-Smith, R. D. (2018). Rapid declines across australian fishery stocks indicate global sustainability targets will not be achieved without an expanded network of 'no-fishing'reserves. *Aquatic Conservation : Marine and Freshwater Ecosystems*, 28(6) :1337–1350.

[Egmont-Petersen et al., 2002] Egmont-Petersen, M., de Ridder, D., and Handels, H. (2002). Image processing with neural networks—a review. *Pattern recognition*, 35(10) :2279–2301.

[Eigaard et al., 2015] Eigaard, O. R., Bastardie, F., Breen, M., Dinesen, G. E., Hintzen, N. T., Laffargue, P., Mortensen, L. O., Nielsen, J. R., Nilsson, H. C., O'Neill, F. G., et al. (2015). Estimating seabed pressure from demersal trawls, seines, and dredges based on gear design and dimensions. *ICES Journal of Marine Science*, 73(suppl_1) :i27–i43.

[Elise and Kulbicki, 2015] Elise, S. and Kulbicki, M. (2015). Etat initial des paysages benthiques et des peuplements de poissons de récifs dans le périmètre d'influence du complexe industriel et minier de vale nouvelle-calédonie : rapport final : partie 1.

[Engel and Kvitek, 1998] Engel, J. and Kvitek, R. (1998). Effects of otter trawling on a benthic community in monterey bay national marine sanctuary. *Conservation Biology*, 12(6) :1204–1214.

[Everingham et al., 2010] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2) :303–338.

[Facon et al., 2016] Facon, M., Pinault, M., Obura, D., Pioch, S., Pothin, K., Bigot, L., Garnier, R., and Quod, J.-P. (2016). A comparative study of the accuracy and effectiveness of line and point intercept transect methods for coral reef monitoring in the southwestern indian ocean islands. *Ecological indicators*, 60 :1045–1055.

[Felzenszwalb and Huttenlocher, 2004] Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International journal of computer vision*, 59(2) :167–181.

[Fernandes et al., 2017] Fernandes, I., Bastos, Y., Barreto, D., Lourenço, L., and Penha, J. (2017). The efficacy of clove oil as an anaesthetic and in euthanasia procedure for small-sized tropical fishes. *Brazilian Journal of Biology*, 77(3) :444–450.

[Fisher et al., 2015] Fisher, R., O'Leary, R. A., Low-Choy, S., Mengersen, K., Knowlton, N., Brainard, R. E., and Caley, M. J. (2015). Species richness on coral reefs and the pursuit of convergent global estimates. *Current Biology*, 25(4) :500–505.

[Fonteneau et al., 2000] Fonteneau, A., Ariz, J., Gaertner, D., Nordstrom, V., and Pallares, P. (2000). Observed changes in the species composition of tuna schools in the gulf of guinea between 1981 and 1999, in relation with the fish aggregating device fishery. *Aquatic Living Resources*, 13(4) :253–257.

[Foveau et al., 2017] Foveau, A., Haquin, S., and Dauvin, J.-C. (2017). Using underwater imagery as a complementary tool for benthos sampling in an area with high-energy hydrodynamic conditions. *Journal of Marine Biology & Oceanography*, 6(02).

[Francour et al., 1999] Francour, P., Liret, C., and Harvey, E. (1999). Comparison of fish abundance estimates made by remote underwater video and visual census. *Nat Sicil*, 23 :155–168.

[Gerum et al., 2017] Gerum, R. C., Richter, S., Fabry, B., and Zitterbart, D. P. (2017). Clickpoints : an expandable toolbox for scientific image annotation and analysis. *Methods in Ecology and Evolution*, 8(6) :750–756.

[Ghazilou et al., 2016] Ghazilou, A., Shokri, M., and Gladstone, W. (2016). Animal v. plant-based bait : does the bait type affect census of fish assemblages and trophic groups by baited remote underwater video (bruv) systems ? *Journal of fish biology*, 88(5) :1731–1745.

[Girshick, 2015] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

[Girshick et al., 2014] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

[Goatley et al., 2016] Goatley, C. H., Bonaldo, R. M., Fox, R. J., and Bellwood, D. R. (2016). Sediments and herbivory as sensitive indicators of coral reef degradation. *Ecology and Society*.

[Goetze and Fullwood, 2013] Goetze, J. and Fullwood, L. (2013). Fiji's largest marine reserve benefits reef sharks. *Coral Reefs*, 32(1) :121–125.

[Goetze et al., 2015] Goetze, J., Jupiter, S., Langlois, T. J., Wilson, S., Harvey, E. S., Bond, T., and Naisilisili, W. (2015). Diver operated video most accurately detects the impacts of fishing within periodically harvested closures. *Journal of Experimental Marine Biology and Ecology*, 462 :74–82.

[Gomes-Pereira et al., 2016] Gomes-Pereira, J. N., Auger, V., Beisiegel, K., Benjamin, R., Bergmann, M., Bowden, D., Buhl-Mortensen, P., De Leo, F. C., Dionísio, G., Durden, J. M., et al. (2016). Current and future trends in marine image annotation software. *Progress in Oceanography*, 149 :106–120.

[González-Rivero et al., 2014] González-Rivero, M., Bongaerts, P., Beijbom, O., Pizarro, O., Friedman, A., Rodriguez-Ramirez, A., Upcroft, B., Laffoley, D., Kline, D., Bailhache, C., et al. (2014). The catlin seaview survey–kilometre-scale seascape assessment, and monitoring of coral reef ecosystems. *Aquatic Conservation : Marine and Freshwater Ecosystems*, 24(S2) :184–198.

[Goo et al., 2016] Goo, W., Kim, J., Kim, G., and Hwang, S. J. (2016). Taxonomy-regularized semantic deep convolutional neural networks. In *European Conference on Computer Vision*, pages 86–101. Springer.

[Gordon et al., 2018] Gordon, T. A., Harding, H. R., Wong, K. E., Merchant, N. D., Meekan, M. G., McCormick, M. I., Radford, A. N., and Simpson, S. D. (2018). Habitat degradation negatively affects auditory settlement behavior of coral reef fishes. *Proceedings of the National Academy of Sciences*, 115(20) :5193–5198.

[Graham et al., 2011] Graham, N. A., Chabanet, P., Evans, R. D., Jennings, S., Letourneur, Y., Aaron MacNeil, M., McClanahan, T. R., Öhman, M. C., Polunin, N. V., and Wilson, S. K. (2011). Extinction vulnerability of coral reef fishes. *Ecology Letters*, 14(4) :341–348.

[Gray et al., 2019] Gray, P. C., Fleishman, A. B., Klein, D. J., McKown, M. W., Bézy, V. S., Lohmann, K. J., and Johnston, D. W. (2019). A convolutional neural network for detecting sea turtles in drone imagery. *Methods in Ecology and Evolution*, 10(3) :345–355.

[Grigg, 1991] Grigg, R. W. (1991). Natural and anthropogenic disturbance on coral reefs. *Coral Reef Ecosystems of the World*.

[Harasti et al., 2017] Harasti, D., Lee, K., Laird, R., Bradford, R., and Bruce, B. (2017). Use of stereo baited remote underwater video systems to estimate the presence and size of white sharks (carcharodon carcharias). *Marine and Freshwater Research*, 68(7) :1391–1396.

[Harris et al., 2018] Harris, D. L., Rovere, A., Casella, E., Power, H., Canavesio, R., Collin, A., Pomeroy, A., Webster, J. M., and Parravicini, V. (2018). Coral reef structural

complexity provides important coastal protection from waves under rising sea levels. *Science advances*, 4(2) :eaao4350.

[Harvey et al., 2007] Harvey, E. S., Cappo, M., Butler, J. J., Hall, N., and Kendrick, G. A. (2007). Bait attraction affects the performance of remote underwater video stations in assessment of demersal fish community structure. *Marine Ecology Progress Series*, 350 :245–254.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers : Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

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

[Hecht-Nielsen, 1992] Hecht-Nielsen, R. (1992). Theory of the backpropagation neural network. In *Neural networks for perception*, pages 65–93. Elsevier.

[Hernández-Serna and Jiménez-Segura, 2014] Hernández-Serna, A. and Jiménez-Segura, L. F. (2014). Automatic identification of species with neural networks. *PeerJ*, 2 :e563.

[Hill et al., 2005] Hill, D., Fasham, M., Tucker, G., Shewry, M., and Shaw, P. (2005). *Handbook of biodiversity methods : survey, evaluation and monitoring*. Cambridge University Press.

[Ho, 2007] Ho, K. (2007). Underwater camera combination. US Patent App. 11/282,767.

[Hoegh-Guldberg et al., 2017] Hoegh-Guldberg, O., Poloczanska, E. S., Skirving, W., and Dove, S. (2017). Coral reef ecosystems under climate change and ocean acidification. *Frontiers in Marine Science*, 4 :158.

[Holmes et al., 2008] Holmes, K. W., Van Niel, K. P., Radford, B., Kendrick, G. A., and Grove, S. L. (2008). Modelling distribution of marine benthos from hydroacoustics and underwater video. *Continental Shelf Research*, 28(14) :1800–1810.

[Huang et al., 2017] Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311.

[Hughes et al., 2018] Hughes, T. P., Anderson, K. D., Connolly, S. R., Heron, S. F., Kerry, J. T., Lough, J. M., Baird, A. H., Baum, J. K., Berumen, M. L., Bridge, T. C., et al. (2018). Spatial and temporal patterns of mass bleaching of corals in the anthropocene. *Science*, 359(6371) :80–83.

[Hughes et al., 2003] Hughes, T. P., Baird, A. H., Bellwood, D. R., Card, M., Connolly, S. R., Folke, C., Grosberg, R., Hoegh-Guldberg, O., Jackson, J. B., Kleypas, J., et al. (2003). Climate change, human impacts, and the resilience of coral reefs. *science*, 301(5635) :929–933.

[Hughes et al., 2017] Hughes, T. P., Barnes, M. L., Bellwood, D. R., Cinner, J. E., Cumming, G. S., Jackson, J. B., Kleypas, J., Van De Leemput, I. A., Lough, J. M., Morrison, T. H., et al. (2017). Coral reefs in the anthropocene. *Nature*, 546(7656) :82.

[Huvenne et al., 2018] Huvenne, V. A., Robert, K., Marsh, L., Iacono, C. L., Le Bas, T., and Wynn, R. B. (2018). Rovs and auvs. In *Submarine Geomorphology*, pages 93–108. Springer.

[Jennings et al., 2001] Jennings, S., Pinnegar, J. K., Polunin, N. V., and Warr, K. J. (2001). Impacts of trawling disturbance on the trophic structure of benthic invertebrate communities. *Marine Ecology Progress Series*, 213 :127–142.

[Joachims, 1998] Joachims, T. (1998). Making large-scale svm learning practical. Technical report, Technical report, SFB 475 : Komplexitätsreduktion in Multivariaten ....

[Johannes, 1975] Johannes, R. (1975). Pollution and degradation of coral reef communities. In *Elsevier Oceanography Series*, volume 12, pages 13–51. Elsevier.

[Joly et al., 2016] Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.-P., Champ, J., Planqué, R., Palazzo, S., and Müller, H. (2016). Lifeclef 2016 : multimedia life species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 286–310. Springer.

[Joly et al., 2017] Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.-P., Lombardo, J.-C., Planque, R., Palazzo, S., and Müller, H. (2017). Lifeclef 2017 lab overview : multimedia species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 255–274. Springer.

[Jones, 1992] Jones, J. (1992). Environmental impact of trawling on the seabed : a review. *New Zealand Journal of Marine and Freshwater Research*, 26(1) :59–67.

[Jouffre et al., 2009] Jouffre, D., Borges, M. d. F., Bundy, A., Coll, M., Diallo, I., Fulton, E. A., Guitton, J., Labrosse, P., Abdellahi, K. o. M., Masumbuko, B., et al. (2009). Estimating eaf indicators from scientific trawl surveys : theoretical and practical concerns. *ICES Journal of Marine Science*, 67(4) :796–806.

[Juhel et al., 2018] Juhel, J.-B., Vigliola, L., Mouillot, D., Kulbicki, M., Letessier, T. B., Meeuwig, J. J., and Wantiez, L. (2018). Reef accessibility impairs the protection of sharks. *Journal of applied ecology*, 55(2) :673–683.

[Kaiser et al., 1996] Kaiser, M., Hill, A., Ramsay, K., Spencer, B., Brand, A., Veale, L., Prudden, K., Rees, E., Munday, B., Ball, B., et al. (1996). Benthic disturbance by fishing gear in the irish sea : a comparison of beam trawling and scallop dredging. *Aquatic Conservation : Marine and Freshwater Ecosystems*, 6(4) :269–285.

[Kaiser et al., 2002] Kaiser, M. J., Collie, J. S., Hall, S. J., Jennings, S., and Poiner, I. R. (2002). Modification of marine habitats by trawling activities : prognosis and solutions. *Fish and Fisheries*, 3(2) :114–136.

[Kavasidis et al., 2014] Kavasidis, I., Palazzo, S., Di Salvo, R., Giordano, D., and Spampinato, C. (2014). An innovative web-based collaborative platform for video annotation. *Multimedia Tools and Applications*, 70(1) :413–432.

[Keller et al., 1985] Keller, J. M., Gray, M. R., and Givens, J. A. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4) :580–585.

[Kilfoil et al., 2017] Kilfoil, J. P., Wirsing, A. J., Campbell, M. D., Kiszka, J. J., Gastrich, K. R., Heithaus, M. R., Zhang, Y., and Bond, M. E. (2017). Baited remote underwater video surveys undercount sharks at high densities : insights from full-spherical camera technologies. *Marine Ecology Progress Series*, 585 :113–121.

[Koh and Wich, 2012] Koh, L. P. and Wich, S. A. (2012). Dawn of drone ecology : low-cost autonomous aerial vehicles for conservation. *Tropical Conservation Science*, 5(2) :121–132.

[Kotsiantis et al., 2007] Kotsiantis, S. B., Zaharakis, I., and Pintelas, P. (2007). Supervised machine learning : A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160 :3–24.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

[Kulbicki, 1998] Kulbicki, M. (1998). How the acquired behaviour of commercial reef fishes may influence the results obtained from visual censuses. *Journal of Experimental Marine Biology and Ecology*, 222(1-2) :11–30.

[Labao and Naval Jr, 2019] Labao, A. B. and Naval Jr, P. C. (2019). Cascaded deep network systems with linked ensemble components for underwater fish detection in the wild. *Ecological Informatics*.

[Lacoursière-Roussel et al., 2016] Lacoursière-Roussel, A., Côté, G., Leclerc, V., and Bernatchez, L. (2016). Quantifying relative fish abundance with edna : a promising tool for fisheries management. *Journal of Applied Ecology*, 53(4) :1148–1157.

[Law et al., 2017] Law, M. T., Urtasun, R., and Zemel, R. S. (2017). Deep spectral clustering learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1985–1994. JMLR. org.

[Lazaros et al., 2008] Lazaros, N., Sirakoulis, G. C., and Gasteratos, A. (2008). Review of stereo vision algorithms : from software to hardware. *International Journal of Optomechatronics*, 2(4) :435–462.

[Lecointe, 1936] Lecointe, P. (1936). Les plantes à roténone en amazonie. *Journal d'agriculture traditionnelle et de botanique appliquée*, 16(180) :609–615.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553) :436.

[LeCun et al., 1990] LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., and Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324.

[Leggat et al., 2019] Leggat, W. P., Camp, E. F., Suggett, D. J., Heron, S. F., Fordyce, A. J., Gardner, S., Deakin, L., Turner, M., Beeching, L. J., Kuzhiumparambil, U., et al. (2019). Rapid coral decay is associated with marine heatwave mortality events on reefs. *Current Biology*.

[Letessier et al., 2015] Letessier, T. B., Juhel, J.-B., Vigliola, L., and Meeuwig, J. J. (2015). Low-cost small action cameras in stereo generates accurate underwater measurements of fish. *Journal of Experimental Marine Biology and Ecology*, 466 :120–126.

[Lettvin et al., 1959] Lettvin, J. Y., Maturana, H. R., McCulloch, W. S., and Pitts, W. H. (1959). What the frog's eye tells the frog's brain. *Proceedings of the IRE*, 47(11) :1940–1951.

[Li et al., 2018] Li, X., Wei, Z., Huang, L., Nie, J., Zhang, W., and Wang, L. (2018). Real-time underwater fish tracking based on adaptive multi-appearance model. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2710–2714. IEEE.

[Lin et al., 2017] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

[Lin et al., 2014] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco : Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

[Liu et al., 2016a] Liu, J., Zhang, S., Wang, S., and Metaxas, D. N. (2016a). Multispectral deep neural networks for pedestrian detection. *arXiv preprint arXiv :1611.02644*.

[Liu et al., 2016b] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016b). Ssd : Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.

[Lorance and Trenkel, 2006] Lorance, P. and Trenkel, V. M. (2006). Variability in natural behaviour, and observed reactions to an rov, by mid-slope fish species. *Journal of Experimental Marine Biology and Ecology*, 332(1) :106–119.

[Lowry et al., 2012] Lowry, M., Folpp, H., Gregson, M., and Suthers, I. (2012). Comparison of baited remote underwater video (bruv) and underwater visual census (uvc) for assessment of artificial reefs in estuaries. *Journal of Experimental Marine Biology and Ecology*, 416 :243–253.

[Maaten and Hinton, 2008] Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov) :2579–2605.

[Maglogiannis et al., 2009] Maglogiannis, I., Loukis, E., Zafiropoulos, E., and Stasis, A. (2009). Support vectors machine-based identification of heart valve diseases using heart sounds. *Computer methods and programs in biomedicine*, 95(1) :47–61.

[Makwela et al., 2016] Makwela, M. S., Kerwath, S. E., Götz, A., Sink, K., Samaai, T., and Wilke, C. G. (2016). Notes on a remotely operated vehicle survey to describe reef ichthyofauna and habitats-agulhas bank, south africa. *Bothalia-African Biodiversity & Conservation*, 46(1) :1–7.

[Mallet and Pelletier, 2014] Mallet, D. and 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.

[Marburg and Bigham, 2016] Marburg, A. and Bigham, K. (2016). Deep learning for benthic fauna identification. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–5. IEEE.

[Marcon et al., 2015] Marcon, Y., Ratmeyer, V., Kottmann, R., and Boetius, A. (2015). A participative tool for sharing, annotating and archiving submarine video data. In *OCEANS 2015-MTS/IEEE Washington*, pages 1–7. IEEE.

[Matabos et al., 2014] Matabos, M., Bui, A. O., Mihály, S., Aguzzi, J., Juniper, S. K., and Ajayamohan, R. (2014). High-frequency study of epibenthic megafaunal community dynamics in barkley canyon : A multi-disciplinary approach using the neptune canada network. *Journal of Marine Systems*, 130 :56–68.

[Matai et al., 2012] Matai, J., Kastner, R., Cutter, G., and Demer, D. (2012). Automated techniques for detection and recognition of fishes using computer vision algorithms. In *Report of the National Marine Fisheries Service Automated Image Processing Workshop*, pages 35–37.

[McClanahan, 2018] McClanahan, T. R. (2018). Community biomass and life history benchmarks for coral reef fisheries. *Fish and fisheries*, 19(3) :471–488.

[McGregor et al., 2015] McGregor, H., Legge, S., Jones, M. E., and Johnson, C. N. (2015). Feral cats are better killers in open habitats, revealed by animal-borne video. *PloS one*, 10(8) :e0133915.

[Merten et al., 2018] Merten, W., Rivera, R., Appeldoorn, R., Serrano, K., Collazo, O., and Jimenez, N. (2018). Use of video monitoring to quantify spatial and temporal patterns in fishing activity across sectors at moored fish aggregating devices off puerto rico. *Scientia Marina*, 82(2) :107–117.

[Millán et al., 2018] Millán, Á. R. H., Mendoza-Moreno, M., López, L. M. P., and Castro-Romero, A. (2018). Comparative study of machine learning supervised techniques for image classification using an institutional identification documents dataset. In *2018 Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI)*, pages 1–6. IEEE.

[Moberg and Folke, 1999] Moberg, F. and Folke, C. (1999). Ecological goods and services of coral reef ecosystems. *Ecological economics*, 29(2) :215–233.

[Mohamed et al., 2018] Mohamed, H., Nadaoka, K., and Nakamura, T. (2018). Assessment of machine learning algorithms for automatic benthic cover monitoring and mapping using towed underwater video camera and high-resolution satellite images. *Remote Sensing*, 10(5) :773.

[Moniruzzaman et al., 2017] Moniruzzaman, M., Islam, S. M. S., Bennamoun, M., and Lavery, P. (2017). Deep learning on underwater marine object detection : A survey. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 150–160. Springer.

[Mouillot et al., 2014] Mouillot, D., Villéger, S., Parravicini, V., Kulbicki, M., Arias-González, J. E., Bender, M., Chabanet, P., Floeter, S. R., Friedlander, A., Vigliola,

L., et al. (2014). Functional over-redundancy and high functional vulnerability in global fish faunas on tropical reefs. *Proceedings of the National Academy of Sciences*, 111(38) :13757–13762.

[Mustafah et al., 2012] Mustafah, Y. M., Noor, R., Hasbi, H., and Azma, A. W. (2012). Stereo vision images processing for real-time object distance and size measurements. In *2012 international conference on computer and communication engineering (ICCCE)*, pages 659–663. IEEE.

[Nasrabadi, 2007] Nasrabadi, N. M. (2007). Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4) :049901.

[Nedevschi et al., 2004] Nedevschi, S., Danescu, R., Frentiu, D., Marita, T., Oniga, F., Pocol, C., Schmidt, R., and Graf, T. (2004). High accuracy stereo vision system for far distance obstacle detection. In *IEEE Intelligent Vehicles Symposium, 2004*, pages 292–297. IEEE.

[Ng et al., 2015] Ng, H.-W., Nguyen, V. D., Vonikakis, V., and Winkler, S. (2015). Deep learning for emotion recognition on small datasets using transfer learning. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pages 443–449. ACM.

[Pan and Yang, 2009] Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10) :1345–1359.

[Park et al., 2017] Park, E., Liu, W., Russakovsky, O., Deng, J., Fei-Fei, L., and Berg, A. (2017). Ilsvrc-2017. *URL http ://www. image-net. org/challenges/LSVRC/2017*.

[Pelletier et al., 2012] Pelletier, D., Leleu, K., Mallet, D., Mou-Tham, G., Hervé, G., Boureau, M., and Guilpart, N. (2012). Remote high-definition rotating video enables fast spatial survey of marine underwater macrofauna and habitats. *Plos One*, 7(2) :e30536.

[Pergent et al., 2017] Pergent, G., Monnier, B., Clabaut, P., Gascon, G., Pergent-Martini, C., and Valette-Sansevin, A. (2017). Innovative method for optimizing side-scan sonar mapping : The blind band unveiled. *Estuarine, Coastal and Shelf Science*, 194 :77–83.

[Peters, 2015] Peters, E. C. (2015). Diseases of coral reef organisms. In *Coral reefs in the anthropocene*, pages 147–178. Springer.

[Poiner et al., 1998] Poiner, I., Glaister, J., Pitcher, C., Burridge, C., Wassenberg, T., Gribble, N., Hill, B., Blaber, S., Milton, D., Brewer, D., et al. (1998). Final report on effects of trawling in the far northern section of the great barrier reef : 1991-1996.

[Pratihar et al., 1999] Pratihar, D. K., Deb, K., and Ghosh, A. (1999). A genetic-fuzzy approach for mobile robot navigation among moving obstacles. *International Journal of Approximate Reasoning*, 20(2) :145–172.

[Priborsky and Velisek, 2018] Priborsky, J. and Velisek, J. (2018). A review of three commonly used fish anesthetics. *Reviews in Fisheries Science & Aquaculture*, 26(4) :417–442.

[Qin et al., 2016] Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. (2016). Deepfish : Accurate underwater live fish recognition with a deep architecture. *Neurocomputing*, 187 :49–58.

[Rawat and Wang, 2017] Rawat, W. and Wang, Z. (2017). Deep convolutional neural networks for image classification : A comprehensive review. *Neural computation*, 29(9) :2352–2449.

[Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once : Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

[Redmon and Farhadi, 2017] Redmon, J. and Farhadi, A. (2017). Yolo9000 : better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271.

[Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn : Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

[Roberts, 1995] Roberts, C. M. (1995). Effects of fishing on the ecosystem structure of coral reefs. *Conservation biology*, 9(5) :988–995.

[Robertson and Smith-Vaniz, 2008] Robertson, D. R. and Smith-Vaniz, W. F. (2008). Rotenone : an essential but demonized tool for assessing marine fish diversity. *Bioscience*, 58(2) :165–170.

[Robinson et al., 2017] Robinson, J. P., Williams, I. D., Edwards, A. M., McPherson, J., Yeager, L., Vigliola, L., Brainard, R. E., and Baum, J. K. (2017). Fishing degrades size structure of coral reef fish communities. *Global change biology*, 23(3) :1009–1022.

[Rogers et al., 2018] Rogers, A., Blanchard, J. L., and Mumby, P. J. (2018). Fisheries productivity under progressive coral reef degradation. *Journal of applied ecology*, 55(3) :1041–1049.

[Rooper and Zimmermann, 2007] Rooper, C. N. and Zimmermann, M. (2007). A bottom-up methodology for integrating underwater video and acoustic mapping for seafloor substrate classification. *Continental Shelf Research*, 27(7) :947–957.

[Rova et al., 2007] Rova, A., Mori, G., and Dill, L. M. (2007). One fish, two fish, butterfish, trumpeter : Recognizing fish in underwater video. In *MVA*, pages 404–407.

[Safavian and Landgrebe, 1991] Safavian, S. R. and Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3) :660–674.

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

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

[Salvat, 1992] Salvat, B. (1992). Coral reefs—a challenging ecosystem for human societies. *Global environmental change*, 2(1) :12–18.

[Shafait et al., 2016] Shafait, F., Mian, A., Shortis, M., Ghanem, B., Culverhouse, P. F., Edgington, D., Cline, D., Ravanbakhsh, M., Seager, J., and Harvey, E. S. (2016). Fish identification from videos captured in uncontrolled underwater environments. *ICES Journal of Marine Science*, 73(10) :2737–2746.

[Shafiee et al., 2017] Shafiee, M. J., Chywl, B., Li, F., and Wong, A. (2017). Fast yolo : a fast you only look once system for real-time embedded object detection in video. *arXiv preprint arXiv :1709.05943*.

[Shiau et al., 2012] Shiau, Y.-H., Lin, S.-I., Chen, Y.-H., Lo, S.-W., and Chen, C.-C. (2012). Fish observation, detection, recognition and verification in the real world. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer ....

[Shortis and Abdo, 2016] Shortis, M. and Abdo, E. H. D. (2016). A review of underwater stereo-image measurement for marine biology and ecology applications. In *Oceanography and marine biology*, pages 269–304. CRC Press.

[Shrivastava et al., 2016] Shrivastava, A., Sukthankar, R., Malik, J., and Gupta, A. (2016). Beyond skip connections : Top-down modulation for object detection. *arXiv preprint arXiv :1612.06851*.

[Simeone, 2018] Simeone, O. (2018). A very brief introduction to machine learning with applications to communication systems. *IEEE Transactions on Cognitive Communications and Networking*, 4(4) :648–664.

[Sirjacobs et al., 2017] Sirjacobs, D., Aguera Garcia, A., Pelaprat, C., Leduc, M., Volpon, A., Danis, B., Gobert, S., and Lejeune, P. (2017). Caractérisation des habitats et communautés benthiques en baie de calvi (corse) : évaluation du potentiel de l'imagerie rov. In *Carhambar, 2017. Cartographie des habitats marins benthiques : de l'acquisition à la restitution. Actes de colloque. Édition Ifremer-AFB. 161 p.* Édition Ifremer-AFB.

[Spampinato et al., 2008] Spampinato, C., Chen-Burger, Y.-H., Nadarajan, G., and Fisher, R. B. (2008). Detecting, tracking and counting fish in low quality unconstrained underwater videos. *VISAPP (2)*, 2008(514-519) :1.

[Spampinato et al., 2010] Spampinato, C., Giordano, D., Di Salvo, R., Chen-Burger, Y.-H. J., Fisher, R. B., and Nadarajan, G. (2010). Automatic fish classification for underwater species behavior understanding. In *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, pages 45–50. ACM.

[Steenweg et al., 2017] Steenweg, R., Hebblewhite, M., Kays, R., Ahumada, J., Fisher, J. T., Burton, C., Townsend, S. E., Carbone, C., Rowcliffe, J. M., Whittington, J., et al. (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.

[Stoner et al., 2008] Stoner, A. W., Ryer, C. H., Parker, S. J., Auster, P. J., and Wakefield, W. W. (2008). Evaluating the role of fish behavior in surveys conducted with underwater vehicles. *Canadian Journal of Fisheries and Aquatic Sciences*, 65(6) :1230–1243.

[Sully et al., 2019] Sully, S., Burkepile, D., Donovan, M., Hodgson, G., and van Woesik, R. (2019). A global analysis of coral bleaching over the past two decades. *Nature communications*, 10(1) :1264.

[Sward et al., 2019] Sward, D., Monk, J., and Barrett, N. (2019). A systematic review of remotely operated vehicle surveys for visually assessing fish assemblages. *Frontiers in Marine Science*, 6 :1–19.

[Szegedy et al., 2015] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

[Teichert et al., 2017] Teichert, N., Pasquaud, S., Borja, A., Chust, G., Uriarte, A., and Lepage, M. (2017). Living under stressful conditions : Fish life history strategies across environmental gradients in estuaries. *Estuarine, Coastal and Shelf Science*, 188 :18–26.

[Thrush and Dayton, 2002] Thrush, S. F. and Dayton, P. K. (2002). Disturbance to marine benthic habitats by trawling and dredging : implications for marine biodiversity. *Annual review of ecology and systematics*, 33(1) :449–473.

[Torrey and Shavlik, 2010] Torrey, L. and Shavlik, J. (2010). Transfer learning. In *Handbook of research on machine learning applications and trends : algorithms, methods, and techniques*, pages 242–264. IGI Global.

[Trenkel and Cotter, 2009] Trenkel, V. M. and Cotter, J. (2009). Choosing survey time series for populations as part of an ecosystem approach to fishery management. *Aquatic Living Resources*, 22(2) :121–126.

[Uijlings et al., 2013] Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, 104(2) :154–171.

[Underwood et al., 2018] Underwood, M., Sherlock, M., Marouchos, A., Forcey, K., and Cordell, J. (2018). A portable shallow-water optic fiber towed camera system for coastal benthic assessment. In *OCEANS 2018 MTS/IEEE Charleston*, pages 1–7. IEEE.

[Veitch et al., 2012] Veitch, L., Dulvy, N. K., Koldewey, H., Lieberman, S., Pauly, D., Roberts, C. M., Rogers, A. D., and Baillie, J. E. (2012). Avoiding empty ocean commitments at rio+ 20. *Science*, 336(6087) :1383–1385.

[Villegas et al., 2015] Villegas, M., Müller, H., Gilbert, A., Piras, L., Wang, J., Mikolajczyk, K., de Herrera, A. G. S., Bromuri, S., Amin, M. A., Mohammed, M. K., et al. (2015). General overview of imageclef at the clef 2015 labs. In *International conference of the cross-language evaluation forum for European languages*, pages 444–461. Springer.

[Wan, 1990] Wan, E. A. (1990). Neural network classification : A bayesian interpretation. *IEEE Transactions on Neural Networks*, 1(4) :303–305.

[Wang et al., 2017] Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., and Tang, X. (2017). Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164.

[Wang et al., 2018] Wang, M., Liu, M., Zhang, F., Lei, G., Guo, J., and Wang, L. (2018). Fast classification and detection of fish images with yolov2. In *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, pages 1–4. IEEE.

[Watson et al., 2005] Watson, D. L., Harvey, E. S., Anderson, M. J., and Kendrick, G. A. (2005). A comparison of temperate reef fish assemblages recorded by three underwater stereo-video techniques. *Marine Biology*, 148(2) :415–425.

[Watson et al., 2006] Watson, R., Revenga, C., and Kura, Y. (2006). Fishing gear associated with global marine catches : Ii. trends in trawling and dredging. *Fisheries Research*, 79(1-2) :103–111.

[Weijerman et al., 2018] Weijerman, M., Gove, J. M., Williams, I. D., Walsh, W. J., Minton, D., and Polovina, J. J. (2018). Evaluating management strategies to optimise coral reef ecosystem services. *Journal of applied ecology*, 55(4) :1823–1833.

[Wickel et al., 2014] Wickel, J., Jamon, A., Pinault, M., Durville, P., and Chabanet, P. (2014). Composition et structure des peuplements ichtyologiques marins de l'île de mayotte (sud-ouest de l'océan indien). *Cybium : Revue Internationale d'Ichtyologie*, 38(3) :179–203.

[Willis, 2001] Willis, T. J. (2001). Visual census methods underestimate density and diversity of cryptic reef fishes. *Journal of Fish Biology*, 59(5) :1408–1411.

[Willis and Babcock, 2000] Willis, T. J. and Babcock, R. C. (2000). A baited underwater video system for the determination of relative density of carnivorous reef fish. *Marine and Freshwater research*, 51(8) :755–763.

[Wraith et al., 2013] Wraith, J., Lynch, T., Minchinton, T. E., Broad, A., and Davis, A. R. (2013). Bait type affects fish assemblages and feeding guilds observed at baited remote underwater video stations. *Marine Ecology Progress Series*, 477 :189–199.

[Xiang et al., 2018] Xiang, X., Yu, C., Lapierre, L., Zhang, J., and Zhang, Q. (2018). Survey on fuzzy-logic-based guidance and control of marine surface vehicles and underwater vehicles. *International Journal of Fuzzy Systems*, 20(2) :572–586.

[Xu et al., 2015] Xu, B., Wang, N., Chen, T., and Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv :1505.00853*.

[Zhang, 2000] Zhang, G. P. (2000). Neural networks for classification : a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4) :451–462.

[Zhang and Zhou, 2005] Zhang, M.-L. and Zhou, Z.-H. (2005). A k-nearest neighbor based algorithm for multi-label classification. *GrC*, 5 :718–721.

[Zhu et al., 2006] Zhu, Q., Yeh, M.-C., Cheng, K.-T., and Avidan, S. (2006). Fast human detection using a cascade of histograms of oriented gradients. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1491–1498. IEEE.

[Zoph et al., 2018] Zoph, B., Vasudevan, V., Shlens, J., and Le, Q. V. (2018). Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710.