Suivi et identification temps-fréquence bioacoustique par transfert deep learning sur YOLO : gestion des chorus

DELOUSTAL Nicolas, CHAVIN Stéphane, GLOTIN Hervé

Technical Report DYNI LIS UTLN CNRS 20230119

Remerciements

Ce rapport de recherche bénéficie de l'environnement et certains des algorithmes développés dans le cadre de la Chaire nationale en IA, Bioacoustique ADSIL, co-financée par l'ANR, l'AID et la DGA, PI Glotin, 2021-2025, ANR-20-CHIA-0014-01.

Sommaire

Introduction

La détection d'objets sur des images de spectrogramme est l'une des utilisations principales de la détection et la classification d'animaux dans un paysage acoustique en traitement du signal. Le spectrogramme est une représentation à deux dimensions (temps-fréquence) mais, à trois paramètres (temps-fréquence-intensité) du signal acoustique.

Au fil des années, différentes approches ont été proposées pour résoudre la problématique de la détection d'objets. C'est alors que les réseaux de neurones convolutifs (CNN) ont émergé comme l'une des méthodes les plus performantes. Les CNN sont des systèmes d'intelligence artificielle (IA) qui utilisent un apprentissage profond (deep learning), sous la forme et le fonctionnement des neurones du cerveau humain, pour effectuer des tâches discriminantes et/ou descriptives (Lecun *et al*., 1998; Nielsen, 2015). Une des premières réalisations fonctionnelles d'un CNN est celle de LeCun *et al*, en 1998 également connus sous le nom de LeNet-5 utilisé dans la reconnaissance d'image (Lecun *et al*., 1998).

Cependant, avec la disponibilité croissante des jeux de données annotées tels que PASCAL VOC (Everingham *et al*., 2010) et ImageNet (Russakovsky *et al,*. 2015*)*, ainsi que l'amélioration des architectures de réseau de neurones, la détection d'objets à l'aide de CNN est devenue plus précise et plus rapide. Ils peuvent dorénavant être utilisés pour de la détection et classification d'espèces animales tel que des oiseaux (Nielsen, 2015; Sevilla et Glotin 2017; Deloustal et Glotin 2022).

You Only Look Once (YOLO), un algorithme de détection d'objets, créé par Redmon & Farhadi en 2018 a été utilisé dans sa version améliorée, YOLOv5 (Jocher *et al,*. 2020) pour la classification des sons d'animaux. Il a été choisi notamment pour sa polyvalence, de plus il permet de traiter un grand nombre d'images en peu de temps tout en conservant une bonne précision. YOLOv5 (Jocher *et al,*. 2020) est un algorithme de détection/classification de formes qui utilise une architecture en réseau de neurones convolutifs profonds (CNN).

Ce premier rapport explique la méthodologie d'utilisation d'une nouvelle méthode de détection d'animaux d'intérêt ayant été utilisée sur des enregistrements sonores annotés ou non. La méthode a été affinée empiriquement, puis décrite. Premièrement sur la création du jeu de données, en passant par leurs structures. Puis, avec le modèle utilisé dont les hypers paramètres expliquent l'obtention de performances optimales à ce jour. Enfin les métriques et scores de performances utilisés ont été mis en évidence et discutés.

Les exemples ici traités sont génériques et s'appliquent autant aux vocalises et clics sous-marins. Ils peuvent être étendus au terrestre.

Méthode

Description des données

Les données utilisées pour effectuer ce rapport ont été celles de la campagne Québec plus précisément décrite dans le rapport de Chavin *et al.,* (2023). Il s'agit de données résultantes d'enregistrements répartis sur toute la ville du Québec. Un total de 11 366 annotations ont été réalisées par des ornithologues.
Seules les annotations d'espèces possédant un bon rapport signal sur bruit (SNR) et une assez bonne quantité d'annotations (tableau 1) ont été conservées. Au total, cela correspond à 21 espèces d'oiseaux différentes avec un nombre d'annotations allant de 58 à 493 par espèce.

Tableau. 1. Liste des espèces et répartition des annotations des jeux de données d'entraînement et de validation (Chavin et al., 2023).

Code	Nom français	Nom latin	Entraînement	Validation	Total
alfl	Moucherolle des aulnes	*Empidonax alnorum*	164	77	**241**
amre	Paruline flamboyante	*Setophaga ruticilla*	69	28	**97**
blja	Geai bleu	*Cyanocitta cristata*	102	17	**119**
btbw	Paruline bleue	*Setophaga caerulescens*	50	20	**70**
btnw	Paruline à gorge noire	*Setophaga virens*	72	42	**114**
coye	Paruline masquée	*Geothlypis trichas*	237	106	**343**
gcki	Roitelet à couronne dorée	*Regulus satrapa*	78	33	**111**
heth	Grive solitaire	*Catharus guttatus*	180	80	**260**
lisp	Bruant de Lincoln	*Melospiza lincolnii*	40	18	**58**
mawa	Paruline à tête cendrée	*Setophaga magnolia*	73	31	**104**
nawa	Paruline à joues grises	*Oreothlypis ruficapilla*	166	70	**236**
oven	Paruline couronnée	*Seiurus aurocapilla*	158	59	**217**
rcki	Roitelet à couronne rubis	*Regulus calendula*	60	29	**89**
revi	Viréo aux yeux rouges	*Vireo olivaceus*	116	38	**154**
rwbl	Carouge à épaulettes	*Agelaius phoeniceus*	118	55	**173**
sosp	Bruant chanteur	*Melospiza melodia*	44	14	**58**
swsp	Bruant des marais	*Melospiza georgiana*	211	95	**306**
swth	Grive à dos olive	*Catharus ustulatus*	75	44	**119**
veer	Grive fauve	*Catharus fuscescens*	69	18	**87**
wtsp	Bruant à gorge blanche	*Zonotrichia albicollis*	352	141	**493**
yrwa	Paruline à croupion jaune	*Setophaga coronata*	61	33	**94**
		Nombre d'annotation :	**2495**	**1048**	**3543**

Traitement des données

Le jeu de données était constitué d'enregistrements originaux ayant été scindés toutes les 5 secondes, avec une fréquence d'échantillonnage normalisée dans un premier temps à 22050 Hz. Chaque morceau de 5 secondes a été annoté en temps/fréquence puis soumis à l'algorithme YOLO (Jocher *et al,*. 2020) pour l'entraînement. Le jeu de données utilisé était des images de spectrogramme annotées par des "bounding boxes" comme représenté en

figure 1, ce sont des rectangles délimitant les zones choisies, cela a eu pour intérêt de délimiter les formes correspondant aux bruits d'animaux.

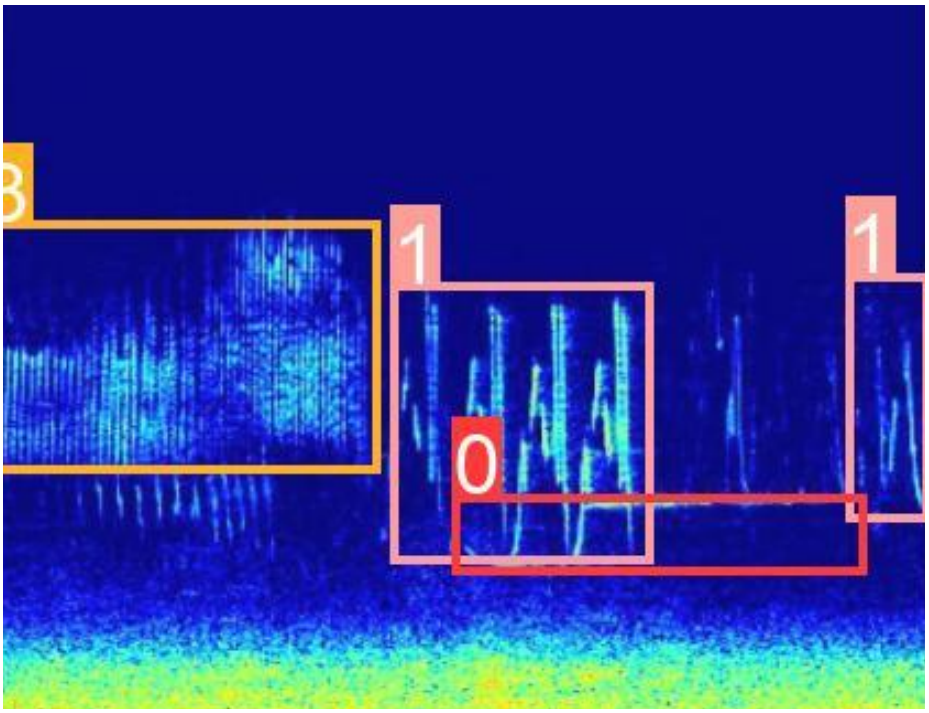

Figure. 1. Spectrogramme d'un enregistrement préalablement découpé en 5 secondes avec des annotations. Les nombres et couleurs correspondent aux espèces cible.

Le spectrogramme (figure 1) de chaque enregistrement a été obtenu en utilisant la transformée de Fourier à court terme (STFT) (Sejdić *et al*., 2009), qui a permis de représenter le signal en temps-fréquence en calculant des transformées de Fourier discrètes (TFD) sur de courtes fenêtres qui se chevauchent. Le résultat sous la forme d'un nombre complexe a été ajouté à une matrice, qui enregistrait l'amplitude et la phase pour chaque point dans le temps et la fréquence (Shentov *et al*., 1995). Sa définition pour un signal S de N échantillons était :

$$S(K) = \sum_{n=0}^{N-1} S(n) e^{-2i\pi k \frac{n}{N}} \quad pour \quad 0 \leq k < N.$$

Le fenêtrage utilisé est de 1024 en type Hann, définit tel que (Blackman *et al*. 1958).

$$w(n) = 0.5 - 0.5 \cos cos\left(\frac{2\pi n}{M-1}\right) \quad pour \quad 0 \leq n < M - 1.$$

Avec M : Nombre de pixels dans la fenêtre de sortie.

Dans un second temps, un algorithme a été créé afin de convertir les annotations temporelles en ratio de pixels en utilisant les équations (1) et (2) pouvant se résumer sur le tableau 2 correspondant à l'image en figure 2.

$$x_{(pixels)} = \frac{x(s) \times largeur\ en\ pixels}{5\ (durée\ en\ s) \times\ largeur\ en\ pixels} \Leftrightarrow \frac{x(s)}{5}. \qquad (1)$$

$$y_{(pixels)} = 1 - \frac{y(Hz) \times hauteur\ en\ pixels}{11025\ (hauteur\ en\ Hz) \times hauteur\ en\ pixels} \Leftrightarrow 1 - \frac{y(Hz)}{11025}. \qquad (2)$$

Tableau. 2 . Coordonnées en pixels de l'annotation sur l'image de spectrogramme. L'id représente l'identifiant de la classe annotée.

id	x	y	w	h
24	0.10223420000	0.92536109006	0.13659599999	0.14927781986

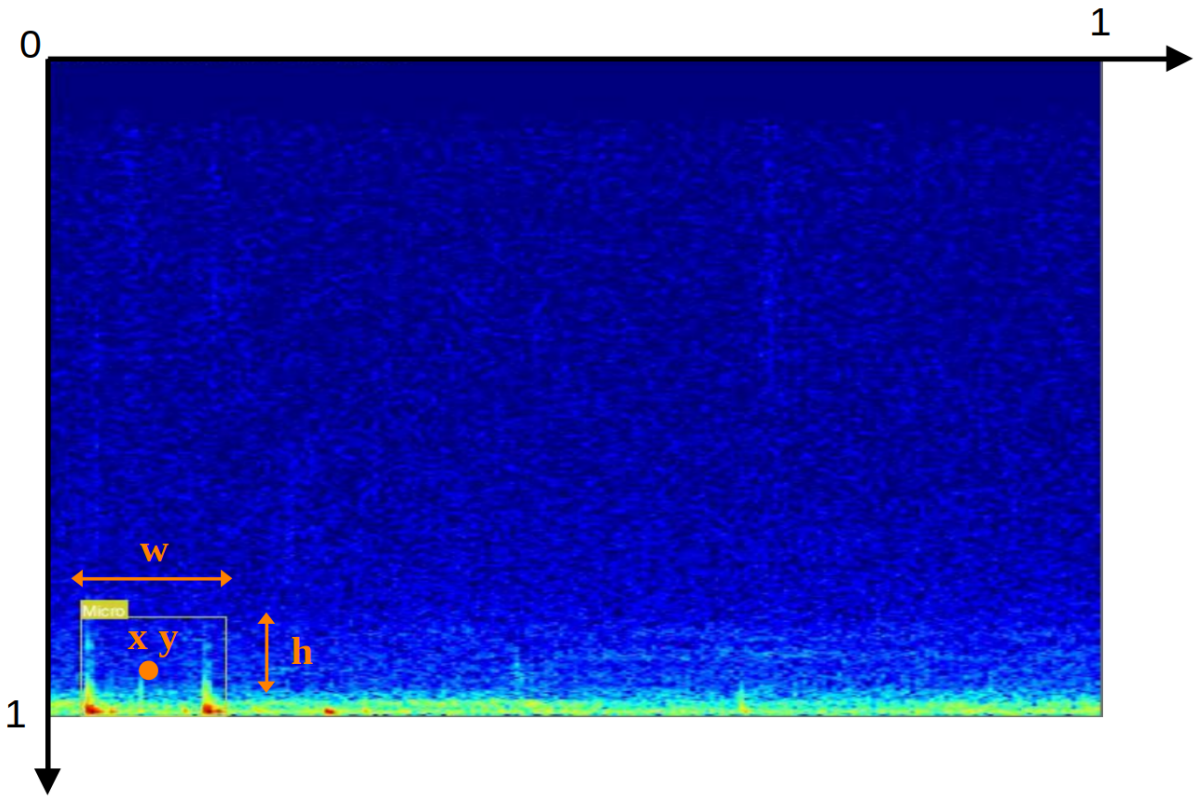

Figure. 2. Exemple d'un spectrogramme d'un enregistrement annoté par une "bounding box" . Les axes représentent le ratio d'image en pixels.

La couleur du signal sur le spectrogramme à été normalisée pour chaque enregistrement, en sélectionnant l'intensité maximum du signal en tant que maximum du spectrogramme. Cette normalisation a pu entraîner une perte d'information sur les différences d'intensité enregistrée. Celle-ci varie principalement en fonction de la distance de l'émetteur et du microphone, aucune mesure de distance ne pourrait alors être réalisée, ce qui n'était pas la cible de cette étude.

Augmentation des données

Au vue de la relativement faible quantité d'images annotées, une large augmentation des données a été utilisée, résumé dans le tableau 2. Le but est de modifier légèrement l'image de spectrogramme afin de créer plusieurs versions d'une unique représentation. Tout cela sans perdre l'information du signal sonore émis par l'animal d'intérêt.

Tableau. 2. Hyperparamètres utilisées.

lr0: 0.01 # initial learning rate (SGD=1E-2, Adam=1E-3)
lrf: 0.1 # final OneCycleLR learning rate (lr0 * lrf)
momentum: 0.937 # SGD momentum/Adam beta1
weight_decay: 0.0005 # optimizer weight decay
warmup_epochs: 3.0 # warmup epochs
warmup_momentum: 0.8 # warmup initial momentum
warmup_bias_lr: 0.1 # warmup initial bias lr
box: 0.05 # box loss gain
cls: 0.3 # cls loss gain
cls_pw: 1.0 # cls BCELoss positive_weight
obj: 0.7 # obj loss gain (scale with pixels)
obj_pw: 1.0 # obj BCELoss positive_weight
iou_t: 0.20 # IoU training threshold
anchor_t: 4.0 # anchor-multiple threshold
hsv_h: 0.01 # image HSV-Hue augmentation (fraction)
hsv_s: 0.1 # image HSV-Saturation augmentation (fraction)
hsv_v: 0.1 # image HSV-Value augmentation (fraction)
translate: 0.1 # image translation (+/- fraction)
scale: 0.1 # image scale (+/- gain)
mosaic: 1.0 # image mosaic (probability)
mixup: 0.1 # image mixup (probability)
copy_paste: 0.1 # segment copy-paste (probability)

Evaluation du modèle

L'entraînement a été lancé sur les GPU du Laboratoire LIS en utilisant en entrée les spectrogrammes annotés. Les images de résolution 640 pixels, traitées sur 32 lots en simultanés (batch) sur une durée de 50 à 150 époques, se stoppant en moyenne lorsque les métriques se stabilisent. L'architecture du modèle utilisé a été la "small" dans le but de posséder un compromis entre des meilleurs scores de performance possibles pour une vitesse de calcul la plus rapide.

En sortie d'entraînement, des courbes roc, des scores de mAP, de F1, des matrices de confusions, ont été calculés pour chaque classe préalablement définis dans les annotations.
La ROC a été représentée sous la forme d'une courbe donnant le taux de vrais positifs ou sensibilité (l'annotation manuelle de l'espèce correspond à la prédiction automatique du modèle) en fonction du taux de faux positifs ou l'antispécificité (1 moins la spécificité) (l'annotation ne correspond pas à la prédiction) (Egan, 1975; Swets *et al.*, 2000; Fawcett, 2006). Les taux sont :

$$Taux\ de\ vrais\ positifs\ (True\ Positive\ Rate) = \frac{nombre\ de\ classifications\ réussites}{nombre\ d'échantillons\ fournis\ de\ l'espèce\ étudiée},$$

$$Taux\ de\ faux\ positifs\ (False\ Positive\ Rate) = \frac{nombre\ de\ mauvaises\ classifications}{nombre\ d'échantillons\ ne\ correspondant\ pas\ à\ l'espèce\ cible}.$$

La Précision Moyenne ou mAP a été utilisée pour calculer la précision du classifieur à partir des scores de prédiction. Pour obtenir la mAP, les valeurs de précision et de rappel ont été exploités, définis par :

$$Précision = \frac{Vrai\ Positif}{Vrai\ Positif + Faux\ Positif},$$

$$Rappel = \frac{Vrai\ Positif}{Vrai\ Positif + Faux\ Négatif}.$$

Par la suite, la moyenne pondérée des précisions de chaque label (AP) a été calculée indépendamment d'après la formule :

$$\mathrm{AP} = \sum_n (R_n - R_{n-1}) P_n$$

Où Pn et Rn correspondaient à la précision et au rappel au n^{ième} seuil.

La mAP obtenue était la moyenne de la moyenne pondérée des précisions de chaque label (espèce). La mAP était donc une valeur qui permettait de tenir compte de l'exactitude (précision) par rapport au nombre de prédictions de labels (rappel) définit par :

$$mAP = \frac{\sum_{q=1}^{Q} AP_{(q)}}{Q},$$

où Q est le nombre de label (espèces).

Enfin, le score F1 a été utilisé avec les valeurs de précisions et de rappels, il calcule la moyenne harmonique, il a l'avantage d'être robuste en présence de données déséquilibrées, définis par :

$$F1 = 2 \times \frac{Précision \times Rappel}{Précision + Rappel}.$$

Cela a permis de tracer des courbes de performance du modèle afin de pouvoir à plus long terme comparer plusieurs jeux de données/modèles.

Résultats

Prédictions sur les données de terrains

Les résultats sont présentés en figure 3, sur la figure 3A, les prédictions semblent correspondre aux bonnes espèces (tableau 1). De plus, il y a une très bonne reconnaissance dans les chorus. En effet, lorsque plusieurs chants se superposent, les espèces sont bien prédites séparément.

En revanche, les spectrogrammes présentés en figure 3B montrent qu'il peut avoir une mauvaise détection des espèces. En effet, sur la figure 3B du haut 4 espèces sont prédites alors qu'il n'y aurait qu'à priori 1 seul espèce qui chante et l'harmonique résultant de son chant. Autre problème pour la figure 3B du bas, aucune espèce n'est détectée, ce problème pourrait provenir d'un trop gros chorus entrainant une non discrimination des différents chants, ou encore d'une nouvelle espèce pas encore annotée à ce jour.

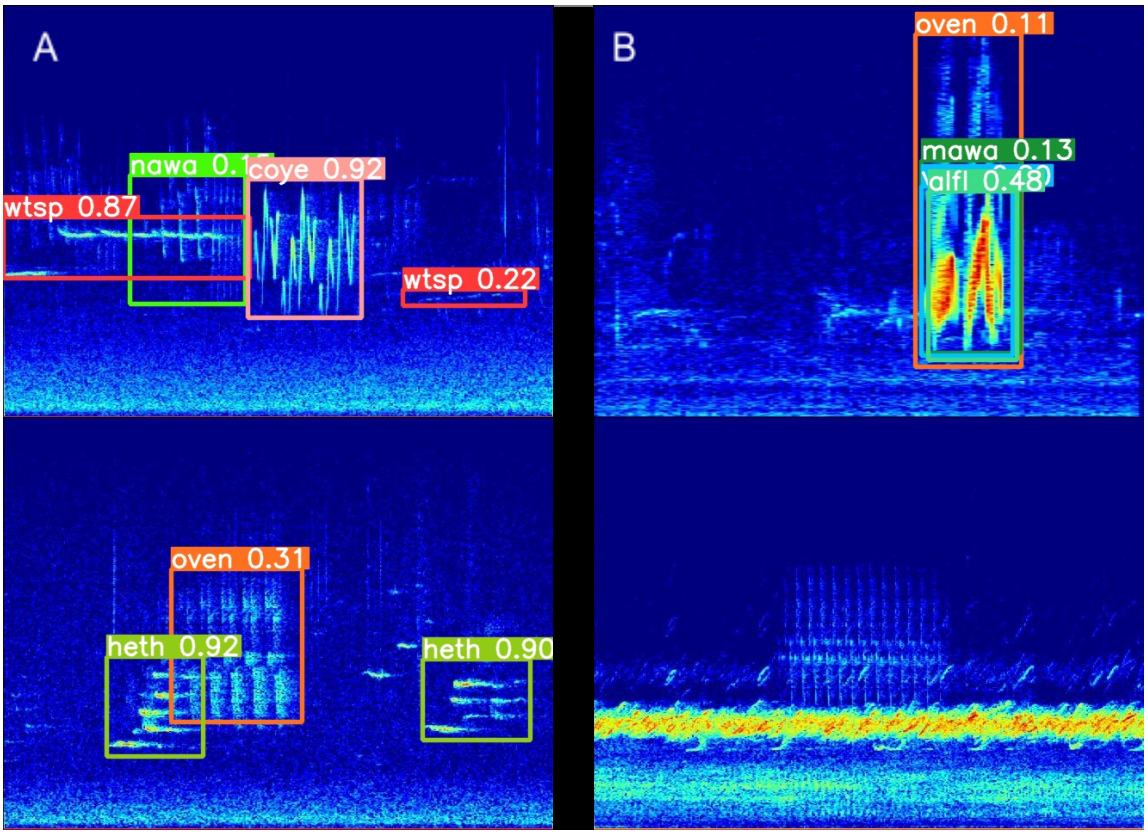

Figure. 3. Spectrogrammes des prédictions d'un enregistrement de Québec non annoté. Sur la gauche (A), les prédictions avec leurs probabilités qui fonctionnent correctement. Sur la droite (B), les prédictions qui ne semblent pas fonctionner.

Matrice de confusion

Les performances du détecteur/classifieur YOLO peuvent être exprimées par une matrice de confusion (figure 4 et figure 1 de l'annexe pour les explications). Sur des données de terrain, un entraînement peut donner des matrices qui ressemblent à celles présentées en figure 4. Plus la case correspondante aux TP est proche de 1, plus le modèle se rapproche des performances optimales, dans ce cas les prédictions possèdent en globalité des scores correctes.

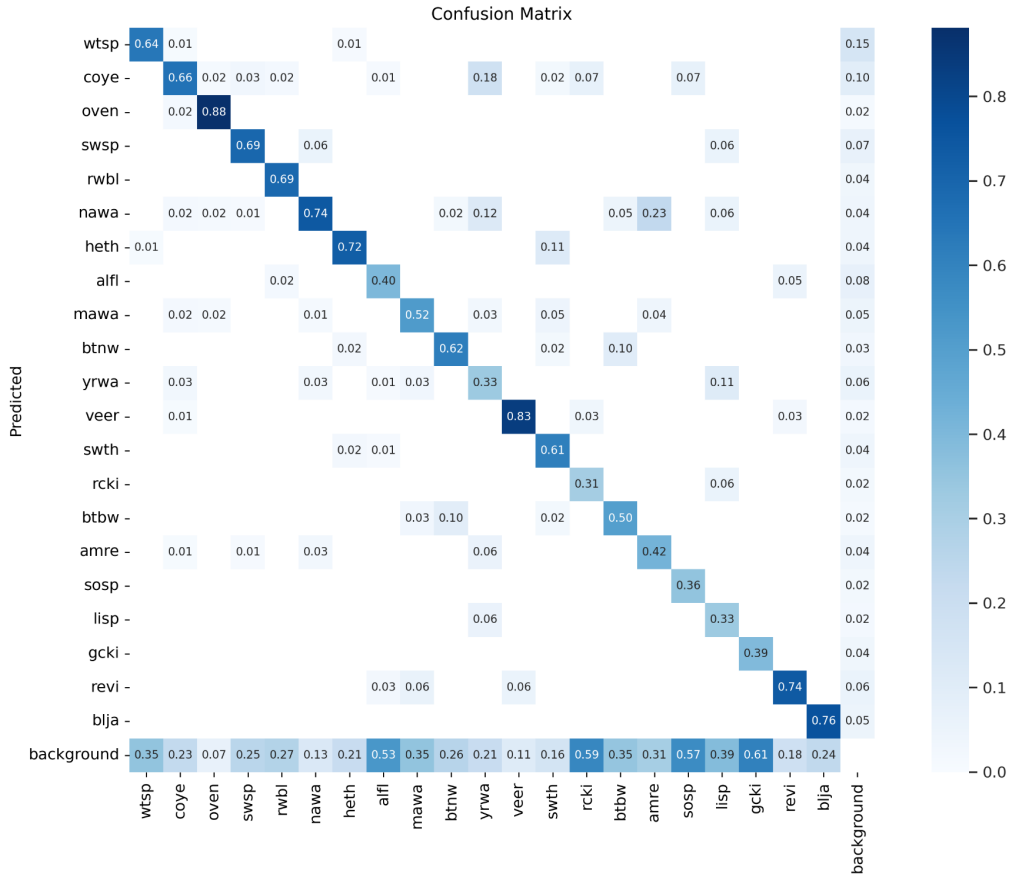

Figure. 4. Matrice de confusion de la validation de modèle. Chaque classe correspond à une espèce d'oiseaux du Québec (Chavin et al., 2023).

Score F1

Par la suite, les performances de la méthode utilisée sont représentées par des courbes, notamment la courbe du score F1 (figure 5), ce qui est interprétable est la variation du score F1 en fonction d'un seuil de confiance. Il y a donc une décroissance du score F1 avec l'augmentation du seuil de classification au-delà de 10%.

Ce qui faut retenir est que cette courbe classe relativement bien les positif/négatif pour une grande gamme de confiance. En effet, lorsque le seuil est trop faible (<10%), les positifs sont bien prédits mais il y a des erreurs sur les négatifs, le F1 augmente en même temps que la confiance.

Pour un seuil optimal, le F1 reste relativement constant (60% environ) pour une valeur de 10 à 80% de confiance, il y a donc peu d'erreurs sur les positifs et les négatifs.

Pour un seuil trop fort (> 80%), les négatifs sont bien prédits mais les positifs ne le sont pas, plus le seuil de confiance se rapproche des 100% plus la F1 décroît.

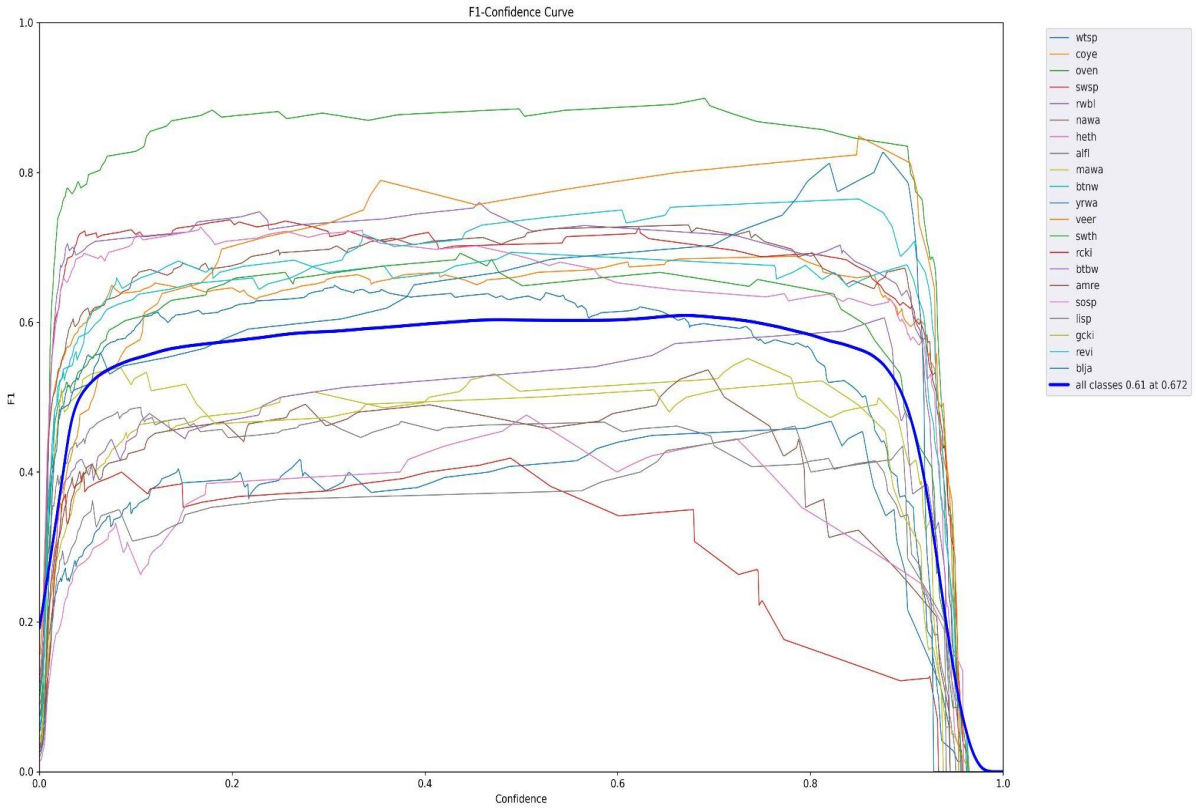

Figure. 5. Courbe F1 en fonction du seuil de confiance. Chaque couleur correspond à une classe, la courbe bleu foncé correspond à la moyenne de toutes les classes.

Fonction d'efficacité du récepteur (courbe ROC)

La courbe ROC (de l'anglais receiver operating characteristic) AUC (area Under the Curve) (figure 6) peut aussi être tracée pour estimer les performances du modèle sur chaque classe.
L'AUC est l'aire sous la courbe ROC, plus l'aire est importante, plus la courbe s'éloigne de la ligne du classificateur aléatoire et se rapproche du classificateur idéal (Bradley, 1997; Hanley et McNeil, 1982).
Plus les courbes de ROC sont aux coordonnées (0:1) du graphique et les AUC proches de 1, moins il n'y a de faux positifs (FP). A contrario, lorsque la courbe ROC est inférieure ou égale à la droite diagonale de coordonnées (0:0),(1:1), cela correspond à une classification aléatoire (AUC de 0.5 ou moins).
Dans ce cas, les AUC sont toutes relativement proches de 85% ce qui se traduit par un faible taux de faux positifs en fonction des vrais négatifs, pouvant être mis en relation avec la matrice de confusion (figure 3 et 1 de l'annexe).

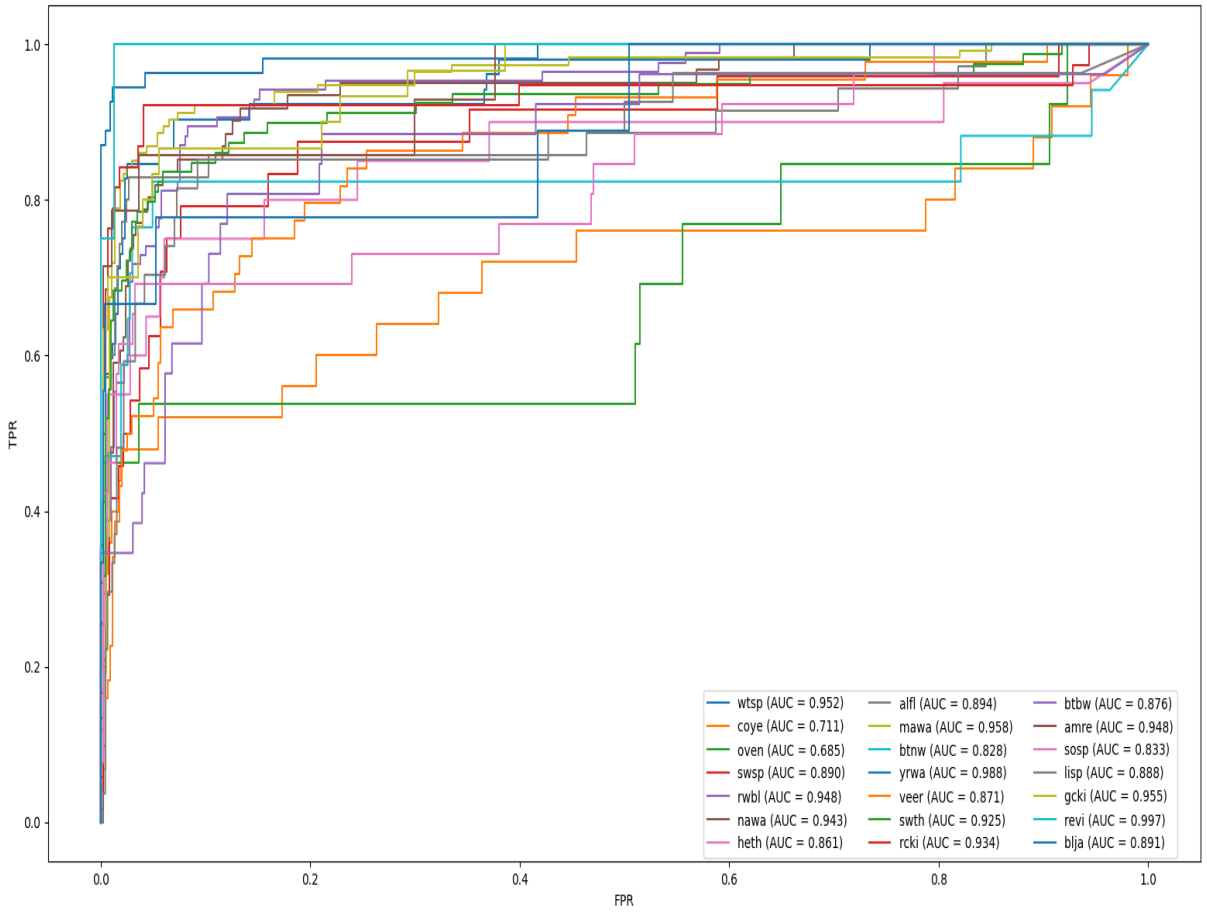

Figure. 6. Courbe de la fonction d'efficacité du récepteur (ROC). Chaque couleur correspond à une classe.

Précision Moyenne (mAP)

Enfin, la figure 7 permet de suivre l'évolution de la mAP en fonction des époques, cela peut aussi être réalisé pour d'autres métriques tel que la précision, le rappel ou encore l'AP.

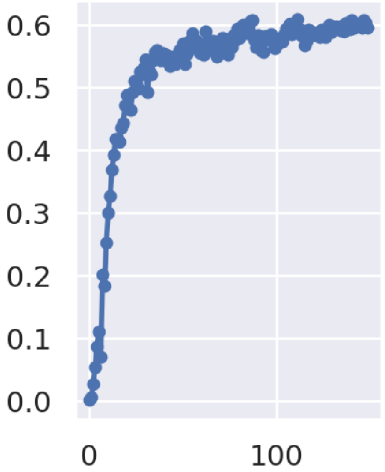

Figure. 7. Évolution de la mAP en fonction des époques. L'axe des ordonnées représente le score de la mAP. L'axe des abscisses représente le nombre d'époques.

Discussion

La méthode utilisée dans ce rapport présente des premiers résultats très intéressants, notamment dans la séparation d'espèces au sein des chorus.

La détection sur données réelles fonctionne avec l'avantage de pouvoir sélectionner les détection supérieurs à un seuil de confiance préalablement choisi en entrée.

Les résultats obtenus lors de cette étude préliminaire démontrent qu'il est possible d'utiliser des méthodes de reconnaissance d'objets tels que YOLO pour détecter/classifier des chants d'animaux représentés sur spectrogramme.

Finalement, cette méthode peut commettre plus d'erreurs de localisation temps-fréquence, mais est moins susceptible de prédire des faux positifs sur l'arrière-plan par rapport aux autres méthodes de détection d'objets existants *(*Redmon *et al,.* 2016) ce qui permet d'obtenir des scores de ROC-AUC souvent très bons.

L'algorithme YOLO est optimisé pour obtenir les meilleures performances à chaque itération d'entrainements. Cependant, il optimise les erreurs de la même façon pour des petites annotations que pour des grosses *(*Redmon *et al,.* 2016). Il se pose alors le problème qu'une petite erreur dans une grande annotation est généralement bénigne, mais une petite erreur dans une petite annotation a un effet beaucoup plus important sur la reconnaissance de la forme du chant de l'animal.

Références

Blackman, R. B., & Tukey, J. W. (1958). The measurement of power spectra Dover Publications. *Inc, New York*.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern recognition, 30(7), 1145-1159.

Chavin S., Mahé P., Hermet T., Deloustal N. and Glotin H., (2023), Rapport de recherche, Analyse automatisée de la diversité acoustique, de la détection d'espèces aux indices bioacoustiques http://sabiod.lis-lab.fr/pub/LIS_QUEBEC_RR.pdf.

Deloustal N., Glotin H., (2022), Rapport de recherche, Veille bioacoustique de l'avifaune en Guadeloupe http://sabiod.lis-lab.fr/pub/LIS_OFB_GUADELOUPE_RAPPORT-1.pdf.

Egan, J. P., & Egan, J. P. (1975). Signal detection theory and ROC-analysis. Academic press.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J. and Zisserman, A. (2010). The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision, 88(2), 303-338.*

Fawcett, T. (2006). An introduction to ROC analysis. Pattern recognition letters, 27(8), 861-874.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology, 143(1), 29-36

Jocher, G., Stoken, A., Borovec, J., Chaurasia, A., & Changyu, L. (2020). ultralytics/yolov5. *Github Repository, YOLOv5*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

Nielsen, M. A. (2015). *Neural Networks and Deep Learning, Determination Press.*

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (2015). ImageNet Large Scale Visual Recognition Challenge. Int J Comput Vis, 115, 211–252.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Sejdić, E., Djurović, I., & Jiang, J. (2009). Time–frequency feature representation using energy concentration: An overview of recent advances. *Digital signal processing*, *19*(1), 153-183.

Sevilla, H Glotin, Audio bird classification with inception-v4 extended with time and time-frequency attention mechanisms, WINNER of LIFECLEF Challenge, Working Notes of CLEF 1866, ceur-ws.org/Vol-1866/paper_177.pd, 2017.

Shentov, O. V., Mitra, S. K., Heute, U., & Hossen, A. N. (1995). Subband DFT—Part I: Definition, interpretation and extensions. *Signal Processing*, *41*(3), 261-277.

Swets, J. A. (1988). Measuring the accuracy of diagnostic systems. Science, 240(4857), 1285-1293.

Annexe

Matrice de confusion

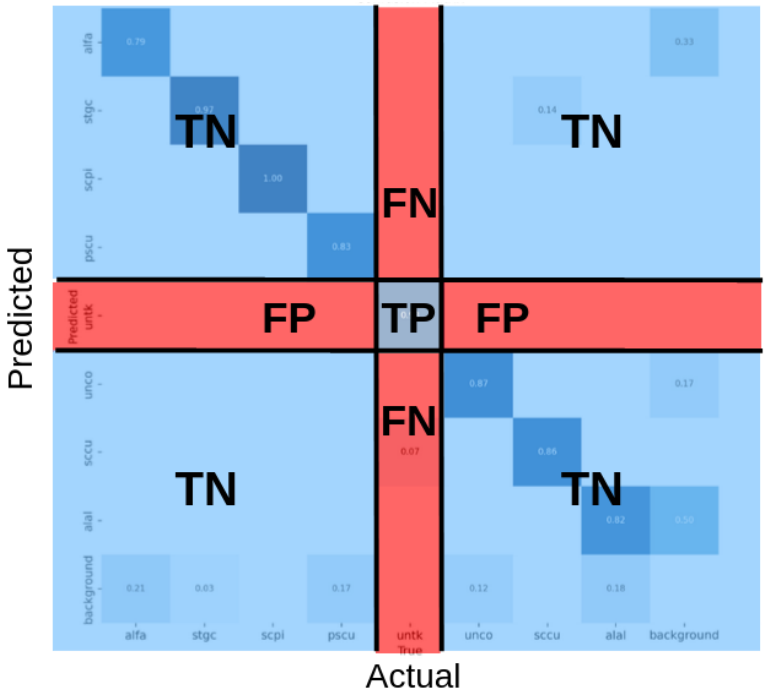

Figure. 1. Matrice de confusion des performances résultant d'un entraînement sur des sons d'orthoptères. Sur l'axe des ordonnées est représenté les classes prédites par le modèle. Sur l'axe des abscisses sont représentées les classes réelles. Où TN correspond à vrai négatif, TP : vrai positif, FN : faux négatif, FP : faux positif.