Méthodes d'intelligence artificielle pour la surveillance d'une colonie de phoques gris

Culture Sciences de l'Ingénieur

La Revue 3E.I

Édité le 17/03/2025

école normale supérieure paris-saclay

Thibault NAPOLÉON[1] – Ayoub KARINE[2] – Hamza MASMOUDI[1]

[1] *ISEN Ouest, LabISEN, Vision-AD, 20 rue Cuirassé Bretagne, 29200 Brest*
[2] *Université Paris Cité, LIPADE, F-75006 Paris, France*

Cette ressource fait partie du N°115 de La Revue 3EI du deuxième trimestre 2025.

Dans le cadre de la conservation de la biodiversité marine, les technologies de surveillance et d'analyse jouent un rôle crucial. En effet, la gestion efficace des habitats naturels nécessite une compréhension approfondie des écosystèmes et de leur dynamique. Cela est particulièrement vrai dans des zones sensibles comme l'îlot de Morgol, situé dans la réserve naturelle nationale d'Iroise, qui abrite une population importante de phoques gris. Ces dernières années, les méthodes traditionnelles de suivi ont été remises en question, notamment à cause de leur impact potentiellement perturbateur sur la faune.

Cet article se concentre sur le développement d'une méthode de surveillance qui minimise les interférences avec les espèces surveillées tout en fournissant des données précises et exploitables. En particulier, nous proposons ici une approche basée sur la vision par ordinateur et les méthodes modernes d'intelligence artificielle pour : 1. Mesurer la fréquence des dérangements liés au passage de bateau ou au débarquement d'humains. 2. Estimer la densité de phoque gris sur l'îlot. La méthodologie employée s'appuie d'une part sur l'utilisation de *YOLO v8* pour la détection des perturbateurs et sur l'approche *IOCFormer* ré-entrainée pour la phase de comptage d'autre part. Comparés à la méthode précédemment utilisée, les résultats obtenus sont plus riches et montre une nette amélioration de l'estimation de la colonie pour des coûts de calcul acceptable.

1 - Introduction

La vision par ordinateur présente aujourd'hui un fort intérêt pour l'observation des espaces naturels au travers de tâches allant de la détection d'objets à la segmentation sémantique en passant par le comptage d'individus [1]. Cet attrait pour ce champ de recherche est poussé par le coût faible des capteurs de vision ainsi que par la croissance des approches d'intelligence artificielle traitant les données issues des capteurs optiques. En particulier, les réseaux de neurones profonds ont permis un développement rapide de nouvelles méthodes grâce à l'essor de deux éléments complémentaires. D'une part, les grandes bases de données d'images, telles qu'*ImageNet* [2], nécessaires à l'apprentissage des modèles d'intelligence artificielle. D'autre part, les puissances de calcul disponible au travers des processeurs graphiques (c.-à-d. GPU) qui se sont avérés particulièrement efficaces pour exécuter les approches neuronales. Ainsi, il est désormais possible d'employer ce type d'approche d'intelligence artificielle pour des projets concrets alliant robustesse des traitements et rapidité des calculs dans des applications diverses telles que la reconnaissance de visages, le déploiement de voitures autonomes ou encore la préservation de l'environnement.

1.1 - Présentation du projet

L'approche présentée ici s'inclut dans le projet de remplacement de l'observatoire qui permettait avant 2024 d'effectuer le comptage des individus d'une colonie de phoques gris après sa destruction par les intempéries. Ce dispositif, installé sur l'îlot de Morgol dans l'archipel de Molène au large des côtes Finistériennes, a pour rôle de faciliter le suivi de la faune sauvage qui s'y trouve avec comme objectif de minimiser les dérangements. L'observatoire mis en place est équipé d'une caméra, de panneaux solaires offrant l'autonomie énergétique, ainsi qu'un lien radio permettant la retransmission des images à terre, voir figure 1. La caméra disponible sur l'observatoire possède deux capteurs, dans les domaines visible et infrarouge, mais seule la première modalité est utilisée ici. Les images captées par l'observatoire sont traitées à terre à l'aide d'une carte de traitement *Jetson Nano Orin* embarquant les algorithmes mis en œuvre dans le projet. En particulier, deux réseaux de neurones ont été déployés pour le suivi de la colonie dans le temps :

1. Un réseau dédié à la détection de bateaux naviguant aux abords de l'îlot ou de personnes y débarquant.
2. Un réseau dédié au comptage s'appuyant sur l'estimation de la densité de phoques gris sur l'îlot.

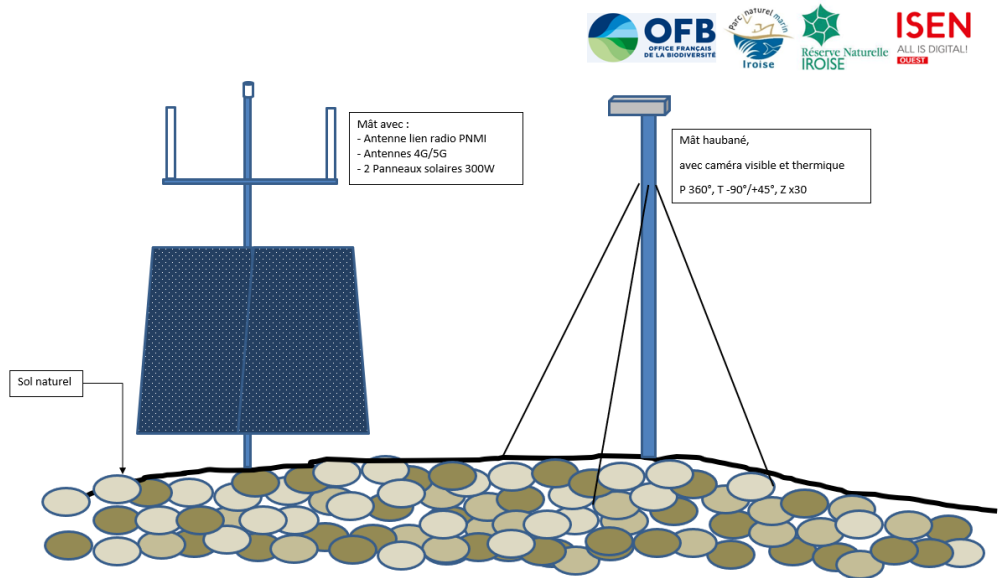

Figure 1 : Schéma de l'observatoire qui sera installé sur l'îlot de Morgol. On y trouve une caméra bispectrale, deux panneaux solaires ainsi qu'une antenne pour le lien radio.

1.2 - Travaux existants

Une première version de l'observatoire sur l'îlot de Morgol avait été réalisée en 2019 pour répondre à la problématique du suivi de la colonie de phoque. La méthode proposée par Karine et al [3] s'appuie sur une approche de classification des vidéos de phoques en utilisant l'apprentissage par transfert sur un réseau de neurones convolutifs (CNN). Grâce au transfert de connaissance apporté par cette technique, qui consiste à réutiliser les hyperparamètres d'un modèle pré-entraîné sur un large ensemble de données, il est possible de limiter la quantité d'annotations nécessaire à la maturation (c.-à-d. apprentissage) du réseau de neurone employé. La mise en œuvre de méthode proposée repose sur deux phases distinctes présentes dans la majorité des approches d'apprentissage supervisé : 1. Une phase « hors ligne » permettant l'entraînement du modèle à partir de vidéos annotées. 2. Une phase « en ligne » utilisant le modèle entraîné pour classer automatiquement les images issues des nouvelles vidéos pour en extraire les phoques et ainsi estimer l'évolution de leur nombre au cours du temps.

Phase « hors ligne »

Cette phase est appelée hors ligne, car elle est réalisée une seule fois pour apprendre au réseau de neurones profond à reconnaitre les phoques sur une image. Dans celle-ci, des imagettes de taille 70 × 100 pixels sont utilisés pour entraîner le modèle de classification. En particulier, une annotation de 353 imagettes de la classe « phoque » ont été utilisée conjointement à 357 imagettes de la classe « non-phoque » comme vérité terrain pour la phase d'apprentissage, voir figure 2. En effet, étant donné le cadre applicatif de la méthode, aucun jeu de données n'était disponible pour entraîner le modèle. Le réseau utilisé est un CNN basé (a) sur l'architecture VGG-16 [4] pré-entraîné sur la base de données *ImageNet*. Ce réseau contient 13 couches convolutives suivies de couches de *pooling* et de couches complètement connectées (c.-à-d. des couches denses). Une couche de 256 neurones ayant une fonction d'activation de type *ReLU* et avec un dropout fixé à 0.5 a été ajoutée pour éviter le surapprentissage. Enfin, la dernière couche de sortie est une classification binaire avec fonction d'activation de type *Softmax* pour permettre une prédiction probabiliste. L'apprentissage par transfert du réseau a été réalisé avec un optimiseur Adam, une taille de lot de 8 sur 50 époques. Pour cette phase, la base d'annotation a été séparée en trois parties, entrainement (60 %), validation (20 %) et test (20 %). La base de validation permet de limiter le risque du surapprentissage tandis que la base de test permet d'évaluer la pertinence du modèle sur des données annotées, mais inconnues du système d'apprentissage. Les résultats obtenus montrent une précision globale de 87.24 % sur la base de test. Les erreurs de classification sont en partie dues à la ressemblance entre les phoques et les galets présents dans les images.

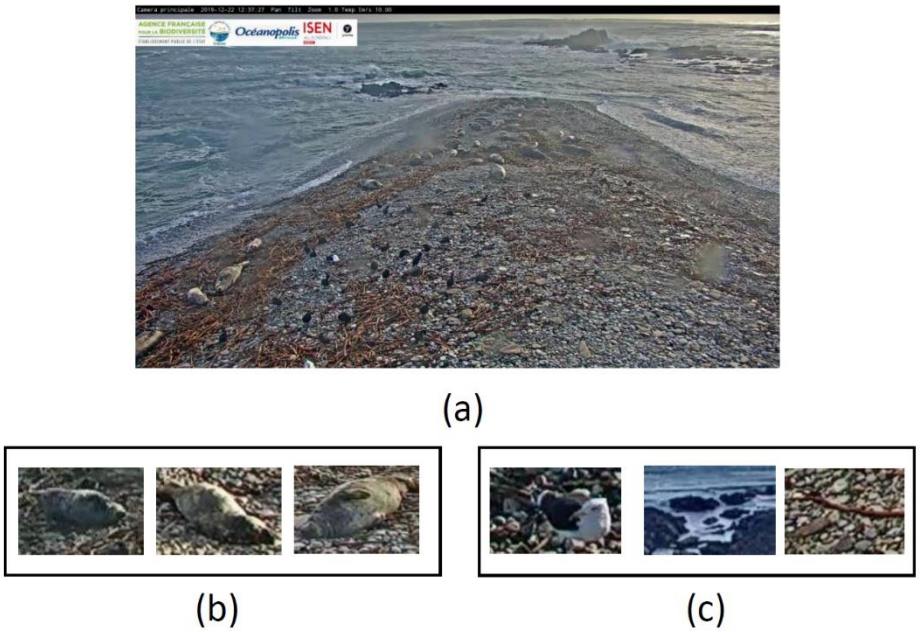

Figure 2 : (a) Exemple d'image issue de la caméra installée sur l'îlot de Morgol. (b) Exemple de vignettes de la classe « phoque ». (c) Exemple de vignettes de la classe « non-phoque ».

Phase « en ligne »

Une fois l'entraînement du modèle réalisé, le système peut alors être utilisé pour estimer le nombre de phoques sur l'îlot. Pour cette phase, une stratégie de fenêtre glissante, avec un décalage de 100 pixels, est utilisée pour classifier chaque vignette de l'image entre les classes « phoque » et « non-phoque ». Ici, la taille des vignettes est identique à celle de la phase d'apprentissage, à savoir 70 × 100 pixels. Dans cette approche, pour toutes les images captées sur l'îlot et retransmises sur le continent, le système d'intelligence artificielle va parcourir l'image pour y détecter les vignettes présentant des phoques. Conjointement à cette détection, des informations temporelles sont extraites de la vidéo pour horodater les présences/absences de phoque. En particulier, les données temporelles sont composées de la date et de l'heure d'acquisition de la vidéo ainsi que de l'instant

précis de la détection en heure, minute et seconde. Ainsi, à la suite de la phase de classification, une agrégation des résultats permet d'estimer le nombre phoque sur chaque image et de suivre ainsi l'évolution de la colonie au cours du temps.

Ce dispositif est désormais inopérant après 5 années en conditions difficiles, incluant la tempête Ciarán qui a eu lieu à la fin de l'année 2023. De ce fait, nous proposons une nouvelle implémentation de l'approche de suivi de la colonie de phoque au travers de nouvelles méthodes qui suivent l'évolution de l'état de l'art dans le domaine.

2 - Approche proposée

Dans le nouveau système, deux approches disjointes, mais complémentaires, ont été mise en place. La première est basée sur *YOLO v8* pour détecter les bateaux et les humains qui sont des perturbateurs pour l'écosystème marin tandis que la seconde s'appuie sur l'approche *IOCFormer* ré-entrainée pour estimer la densité de phoque gris.

2.1 - Mesure des perturbations : *YOLO v8*

Afin d'étudier précisément le comportement de la colonie de phoque gris, l'évaluation de l'impact des évènements extérieurs est nécessaire. Pour cela, un outil d'intelligence artificielle permettant la détection d'évènements tels que le passage d'un bateau, l'arrivée d'un kayak ou le débarquement de personnes sur l'île est mise en place. Ainsi, associé aux informations temporelles, il est possible d'évaluer le degré de perturbation de la colonie et le temps de retour à l'état normal du reposoir. L'outil développé est basé sur un réseau de neurones convolutifs « léger » permettant le traitement des vidéos en temps réel. En effet, l'une des contraintes du projet est la quantité de données à analyser, à savoir, un flux vidéo Full HD permanent. Le réseau de neurones profonds utilisé est la version 8 de la famille *YOLO* [5] (You Only Look Once) développé par Ultralytics qui améliore la précision et la vitesse par rapport aux versions précédentes grâce à des optimisations architecturales et de nouvelles techniques d'apprentissage. Ce modèle modulaire et polyvalent permet de réaliser plusieurs tâches en vision par ordinateur, dont la détection d'objets.

L'architecture de *YOLO v8* est composée de trois parties principales : *backbone*, *neck* et *head*. Le réseau de base (c.-à-d. Le *backbone*) permet d'extraire les caractéristiques visuelles de l'image telles que les contours, les textures ainsi que les motifs colorimétriques. Pour cela, il s'appuie sur des optimisations de l'architecture *CSPDarknet* permettant une meilleure efficacité. La partie *neck* de l'architecture apporte une capacité multi-échelle à la détection en fusionnant les informations issues de différentes échelles de l'image. Il est basé sur une optimisation de la méthode *PANet*. Enfin, la partie *head* du réseau génère les prédictions sous forme de boîtes englobantes associées, pour chacune, aux classes prédites avec leurs scores de confiances.

Ce réseau pré-entraîné sur la base de données *COCO* [6] permet de détecter et de reconnaître un ensemble de concept, dont les bateaux et les humains. Appliqué sur chaque image de la vidéo, il permet de détecter les évènements extérieurs pouvant perturber la colonie. Afin de rendre plus robuste le système dans les conditions difficiles, une étape de post-traitements a été mise en place. Étant donné qu'une perturbation de type « humain » ou « bateau » dure dans le temps, un lissage des détections est ajouté au système en suivant deux stratégies : 1. Une fusion des détections similaires proches dans le temps (perte de détection). 2. Une suppression des détections isolées (fausse détection). La première stratégie permet d'agréger deux détections similaires qui apparaissent à moins de 10 secondes d'intervalle tandis que la seconde élimine ensuite les détections de moins de 5 secondes.

2.2 - Comptage s'appuyant sur l'estimation de la densité : *IOCFormer*

Le modèle utilisé dans la version précédente [3] présente des lacunes significatives lorsqu'il s'agit de résoudre les ambiguïtés créées par des phoques qui sont partiellement occultés par d'autres. Cette limitation découle de la nature des données d'entraînement et de la structure même du réseau de neurones, qui ne permet pas de détecter plus d'un phoque dans une fenêtre de taille 70 x 100 pixels. Cette incapacité peut avoir des conséquences importantes pour l'estimation de la taille de la colonie, dégradant de ce fait les données fournies au parc marin pour lequel la qualité du comptage est importante pour évaluer correctement l'état de la population de phoque dans le but de conserver leur habitat. Pour pallier ces imprécisions, un changement de paradigme a été opéré dans la nouvelle version du projet. En effet, l'algorithme proposé ne repose plus sur une classification par fenêtre glissante, mais sur un comptage par estimation de densité.

Cette technique vise à convertir le problème de comptage en une tâche de prédiction de densité. Au lieu de compter directement les objets, cette méthode génère une carte de densité où chaque pixel représente la densité probable d'objets à cet emplacement. Cette technique est particulièrement utile dans des scènes où les objets sont groupés ou se chevauchent, car elle permet de modéliser la densité d'objets même dans les zones où ils ne sont pas clairement séparés. En particulier, le modèle *IOCFormer* [7] a été utilisé, car il présente une approche performante pour traiter le problème du comptage des objets dits indiscernables. Après une extraction de caractéristiques au moyen d'un encodeur (*ResNet-50* [8]), ce modèle s'appuie sur deux branches complémentaires, à savoir : 1. Une branche de densité qui estime la position des objets (c.-à-d. des phoques dans notre cas) de manière floue. 2.Une branche de régression qui prédit la position exacte des objets précédemment détectés.

Précisément, la branche de densité utilise un ensemble de convolutions pour produire une carte de densité approximative de la position des objets à partir des caractéristiques disponibles en sortie de l'encodeur. Cette carte est supervisée par une fonction de perte pour aligner la densité estimée avec le nombre réel d'objets dans l'image. La seconde branche, quant à elle, prend en entrée les caractéristiques extraites par l'encodeur ainsi que la carte de densité produite par la première branche pour affiner les prédictions, notamment en améliorant la détection des objets qui se chevauchent ou se fondent dans l'environnement. Ceci est mis en œuvre à l'aide d'un *transformer*, nommé *DETE*, supervisé par une fonction de perte qui compare les prédictions aux annotations. L'innovation clé de l'approche est la mise au point du *DETE* (c.-à-d. *Density-Enhanced Transformer Encoder*) qui joue un rôle central dans l'amélioration des performances, voir figure 3.

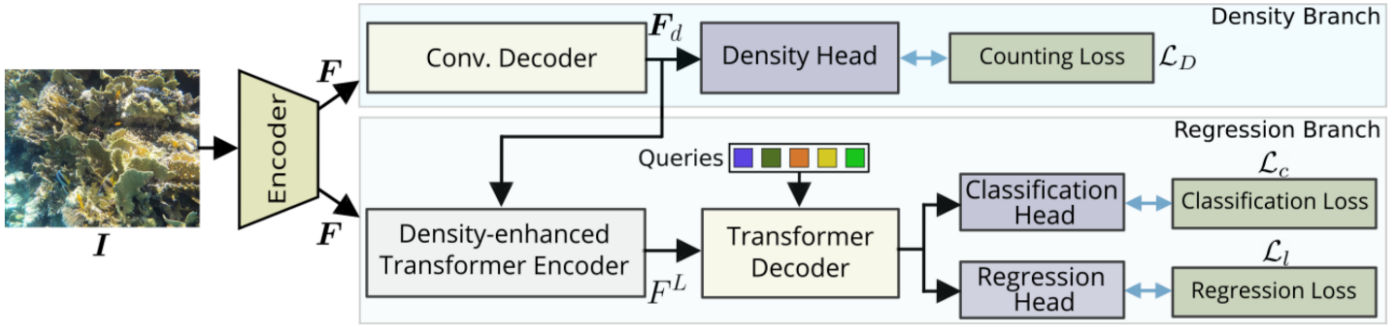

Figure 3 : Architecture de la méthode IOCFormer avec, à gauche, l'entrée est l'extraction des caractéristiques par l'encodeur ResNet-50, en haut à droite la branche de densité et en bas à droite la branche de régression (image issue de l'article original [7]).

3 - Expérimentations et résultats

Avant une utilisation en conditions réelles, les deux outils proposés doivent être amenés à maturité, quand c'est nécessaire, et validés sur un jeu de données préétabli. Pour cela, nous présentons dans cette partie, les bases de données qui ont permis l'évaluation quantitative des modèles, les détails liés à l'apprentissage de l'architecture *IOCFormer* ainsi que les résultats obtenus.

3.1 - Validation des mesures de perturbations

Pour la détection des perturbations humaines, le réseau de neurones *YOLO v8 m* (c.-à-d. la version de taille moyenne) a été utilisé tel qu'il est fourni, pré-entrainé sur la base de données d'images *COCO*. Cette version offre un bon compromis entre performance et rapidité en adéquation avec la puissance de calcul disponible à terre pour analyser les images. Afin de détecter les perturbations, seules les détections relatives aux classes « humain » et « bateau » sont conservées. Aussi, toutes les détections dont la confiance est inférieure à 0.75 (c.-à-d. 75 % de confiance) sont éliminées pour éviter les fausses alarmes. Enfin, les deux stratégies présentées dans la partie 2.1 sont appliquées pour consolider les détections. La première permet de fusionner les détections similaires proches dans le temps pour corriger les pertes de détection. La seconde limite les fausses alarmes en supprimant les détections isolées.

Base de données

Afin de valider les performances de l'approche mise en œuvre, une base de données a été constituée. Elle comprend 200 images réparties équitablement en deux jeux de données : perturbation par un humain et/ou un bateau (possiblement plusieurs), pas de perturbation. Pour chacune des images de la première catégorie, les boites minimales englobant les perturbations ont été manuellement annotées et utilisées comme vérité terrain, voir figure 4. Ces images sont issues de la précédente campagne d'acquisition qui embarquait une caméra similaire avec une résolution de 1920 × 1080 pixels filmant à 10 images par seconde.

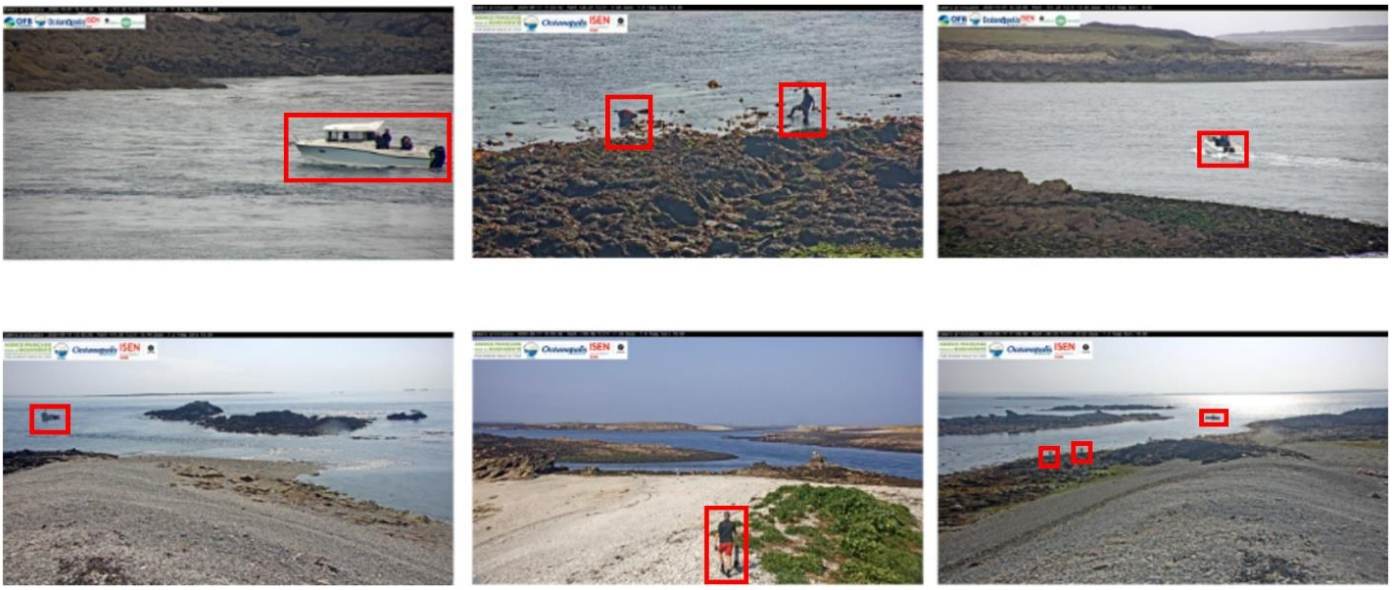

Figure 4 : Exemple de 6 images de la base de données appartenant à la catégorie « perturbation ». En rouge, les boites minimales englobantes des objets (humain ou bateau) utilisées comme vérité terrain.

Résultats

Sur la base de données de test, les résultats obtenus montrent une performance de 97 % avec la répartition suivante entre les deux jeux de données : 94 % de bonnes détections pour la catégorie « perturbation », 100 % de non-détections pour la catégorie « pas de perturbation ». En particulier,

1 erreur a été commise sur la non-détection d'un bateau et 5 sur la non-détection d'humains. Ces erreurs sont dues à une difficulté importante à différencier les pêcheurs à pied de certains rochers lorsque les individus sont très éloignés de la caméra, voir figure 4. Concernant les bateaux, la détection est plus aisée, même s'ils restent parfois difficiles à détecter lorsqu'ils sont au loin, parfois masqués à intervalles réguliers par les vagues au large de l'îlot. Pour permettre une exploitation future par les conservateurs du parc marin, les résultats obtenus sont exportés dans un fichier CSV afin d'archiver les perturbations. Un exemple d'extraction est visible dans le tableau 1. On y retrouve la date et l'heure de la perturbation, son type et sa position dans l'image. Ainsi, il est aisé de croiser les résultats des perturbations et de comptage pour estimer l'impact humain sur la colonie. À titre d'exemple, la figure 4 montre que lorsqu'il y a une perturbation, les phoques quittent généralement l'îlot.

Date et heures	Types d'objets	Position
21/09/2020 13:32:15	Bateau	[10, 12, 512, 312]
21/09/2020 14:52:10	Bateau	[150, 565, 220, 622]
21/09/2020 18:02:10	Humain	[155, 154, 223, 312]
21/09/2020 18:02:10	Humain	[256, 189, 353, 359]
22/09/2020 10:17:15	Bateau	[45, 652, 125, 732]
22/09/2020 12:22:48	Bateau	[456, 785, 589, 936]
22/09/2020 15:02:51	Bateau	[221, 375, 452, 427]

Tableau 1 : Exemple d'export au format CSV des perturbations détectées.

3.2 - Validation du comptage

Le comptage des phoques gris a été réalisé avec l'approche *IOCFormer*, initialement utilisée et évaluée sur une base de données de poissons. Étant donné les différences entre les deux contextes expérimentaux, poissons et phoques, une nouvelle base de données annotée a dû être mise en place pour permettre un réapprentissage du réseau de neurones.

Base de données

La base de données établie a été collectée à partir de 5 vidéos, chacune d'une durée de 60 minutes, capturant divers scénarios. Cette extraction vise à capturer des scènes variées afin d'assurer que le modèle puisse généraliser à différentes conditions de prise de vue ou de météo par exemple. Un échantillon de 1000 images a été sélectionné pour servir de base à l'entraînement, à la validation et au test du modèle. Afin d'établir une vérité terrain nécessaire à mesurer quantitativement la robustesse de la méthode, chacune des 1000 images a été annotée à l'aide des outils *LabelImg* [9] et *RectLabel*. En particulier, les phoques présents dans les images ont été annotés par un point situé au centre de l'individu, comme le requiert la méthode *IOCFormer*. Cette phase a permis d'obtenir un total de 6084 annotations. Afin d'entraîner le modèle, de le valider et finalement de le tester, la base de données a été divisée en 3 jeux de données répartis comme suit : 800 images pour l'entraînement, 100 images pour la validation et 100 images pour le test.

Apprentissage du modèle

L'implémentation de la méthode *IOCFormer* s'appuie sur le code officiel [11] développé avec *PyTorch* pour des GPU NVIDIA. Les spécificités de l'architecture sont les mêmes que pour la méthode originale, à savoir 4 blocs *transformer* pour la partie *DETE* avec 700 *queries*. Les caractéristiques sont extraites avec un encodeur ResNet-50 pré-entraîné sur *ImageNet*. Pour les augmentations de données, nous utilisons un redimensionnement aléatoire et un retournement horizontal. Les images sont recadrées aléatoirement pour obtenir des images d'entrées de taille 256 x 256 pixels. L'entraînement est réalisé sur 2 GPU NVIDIA Quadro RTX 8000 avec des lots de 4 images sur une

durée de 1500 *epochs* avec l'optimiseur Adam [12]. Lors de l'inférence, les images sont divisées en vignettes de taille identique à celles utilisées pendant l'entraînement. Enfin, un seuil de 0.35 est utilisé pour filtrer les prédictions [13].

Les métriques utilisées pour vérifier la qualité de l'apprentissage sont le MAE (*Mean Absolute Error*) et le MSE (*Mean Squared Error*). L'Erreur Absolue Moyenne (MAE) est la moyenne des différences absolues entre les valeurs prédites et la vérité terrain. Elle est également connue sous le nom de norme L1 ou distance de Manhattan :

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

Où N est le nombre d'observation, y_i la valeur prédite pour la i^e observation et $\hat{y}_i$ la vérité terrain associée à cette même observation.

Le MSE (Erreur Quadratique Moyenne), quant à lui, est la moyenne des carrés des différences entre les valeurs prédites et la vérité terrain. Elle est également connue sous le nom de norme L2 ou distance euclidienne :

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

La figure 5, montre l'évolution des métriques au cours de l'apprentissage. On remarque des résultats cohérents pour les deux métriques observées avec des erreurs qui se stabilisent à partir de 1200 *epochs*. La sélection du meilleur modèle a été réalisée en étudiant les performances du MSE sur le jeu de données de validation à chaque *epoch*.

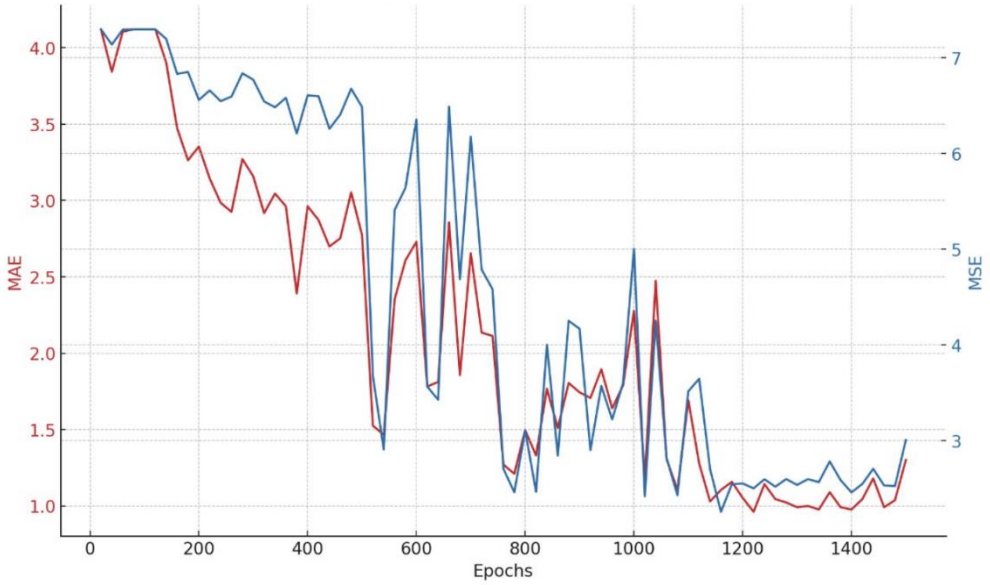

Figure 5 : Visualisation de l'évolution du MAE et du MSE au cours de l'apprentissage. On remarque une stabilisation des métriques après 1200 epochs environ.

Résultats quantitatifs

Les résultats obtenus sur la base de données de test montrent de bons résultats qualitatifs, même sur des images présentant un grand nombre de phoques, voir figure 6. La performance globale du système sur l'ensemble de test est de 91.2 %, en progression par rapport à la méthode précédente qui avait une performance de 87.24 %. On note cependant que les résultats ne sont pas

comparables, car les bases de données utilisées pour l'évaluation sont différentes. Cependant, l'approche proposée montre des perspectives intéressantes pour le comptage des phoques en situation difficile. Finalement, les résultats du comptage sont exportés dans un fichier CSV regroupant les informations visibles dans le tableau 2. On y retrouve l'horodatage du comptage ainsi que le nombre d'individus présent sur l'image.

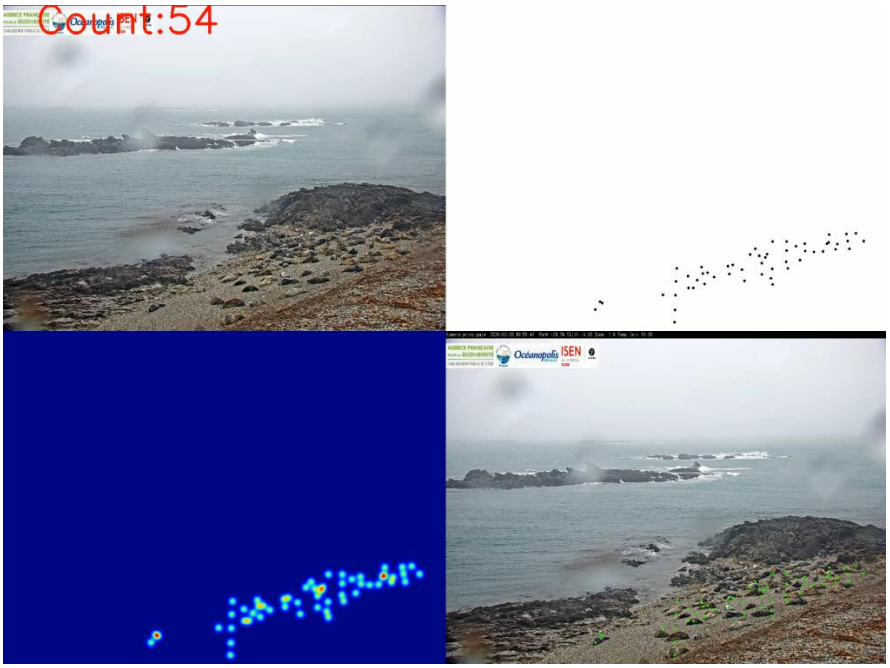

Figure 6 : Résultats de la méthode de comptage proposée avec, l'image originale avec le nombre de phoques (en haut à gauche), l'estimation de la densité (en bas à gauche), l'extraction des individus à partir de la densité (en haut à droite) et la prédiction finale (en bas à droite).

Date et heures	Nombre
22/12/2019 11:00	58
22/12/2019 11:15	55
22/12/2019 11:30	59
22/12/2019 11:45	62
22/12/2019 12:00	58
22/12/2019 12:15	46
22/12/2019 12:30	48

Tableau 2 : Exemple d'export au format CSV du nombre de phoques détectés.

4 - Conclusion

Les résultats obtenus avec les deux approches proposées montrent que l'étude des perturbations ainsi que le comptage sont possibles en utilisant la vision par ordinateur. D'une part, l'architecture *YOLO v8* a permis d'identifier avec robustesse les perturbations humaines qui peuvent survenir autour de l'îlot. D'autre part, l'approche *IOCFormer* a montré sa pertinence pour l'estimation de la densité de la colonie dans des conditions parfois difficiles. Cependant, même si les résultats obtenus répondent bien aux attentes du projet, certains points d'améliorations restent possibles. Le premier élément qui n'a pas été pris en compte dans le projet, malgré sa disponibilité, est l'information issue du capteur infrarouge de la caméra. En effet, cette nouvelle modalité pourrait permettre de robustifier l'estimation de la densité de phoques gris en réalisant une fusion des informations issues des deux types d'images. Cette fusion serait alors réalisée, soit tardivement en fusionnant les estimations de densité, soit précocement lors de l'apprentissage du réseau de neurones qui prendrait alors en entrée les deux modalités d'image plutôt que seulement l'image

dans le domaine visible. D'autre part, il faut noter que derrière les réussites liées aux grandes tâches de vision telles que la reconnaissance de visage, le développement des voitures autonomes ou la génération d'images synthétiques se cache des besoins importants en bases de données annotées, nécessaires aux phases d'entrainement, ainsi qu'en puissance de calcul. Afin de pallier ces contraintes, qui pourraient avoir un impact significatif si le système de décision était amené à devenir autonome sur l'îlot, une approche de frugalité pourrait être envisagée. En particulier, une approche de réduction de la taille, et donc du coût calculatoire, des réseaux de neurones pourrait être envisagée en s'appuyant sur les techniques de distillation de connaissances par exemple [14]. Finalement, la réussite de ce projet montre que les techniques de vision par ordinateur récentes peuvent permettre de répondre, avec une certaine simplicité, à des projets tels que la préservation de la biodiversité, en limitant les opérations humaines qui peuvent perturber les écosystèmes au travers de leurs observations sur le terrain. Ainsi, les données qui seront recueillies en 2025 et 2026 pourront permettre de valider la méthodologie mise en œuvre pour mieux comprendre les impacts humains sur la colonie de phoques gris installée sur l'îlot de Morgol dans l'archipel de Molène au large des côtes Finistérienne.

Remerciement :

Nous remercions Jean-Yves MULOT, Philippe FORJONEL et Léo-Paul PELLETIER du LabISEN pour leur aide tout au long du projet ainsi que l'Office Français de la Biodiversité et le Parc Naturel Marin d'Iroise pour le financement du projet ainsi que pour leur soutien.

Références :

[1] Zizhu FAN, Hong ZHANG, Zheng ZHANG et al. A survey of crowd counting and density estimation based on convolutional neural network. Neurocomputing. 2022. vol. 472, p. 224-251.

[2] Jia DENG, Wei DONG, Richard SOCHER et al. Imagenet: A large-scale hierarchical image database. Conference on Computer Vision and Pattern Recognition (CVPR). 2009. p. 248-255.

[3] Ayoub KARINE, Thibault NAPOLÉON, Jean-Yves MULOT et al. Video seals recognition using transfer learning of convolutional neural network. International Conference on Image Processing Theory, Tools and Applications (IPTA). 2020. p. 1-4.

[4] Karen SIMONYAN and Andrew ZISSERMAN. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[5] Glenn JOCHER, Ayush CHAURASIA and Jing QIU. Ultralytics YOLOv8. https://github.com/ultralytics/ultralytics. 2023.

[6] Tsung-Yi LIN, Michael MAIRE, Serge BELONGIE et al. Microsoft coco: Common objects in context. European Conference on Computer Vision (ECCV). 2014. p. 740-755.

[7] Guolei SUN, Zhaochong AN, Yun LIU et al. Indiscernible object counting in underwater scenes. Computer Vision and Pattern Recognition (CVPR). 2023. p. 13791-13801.

[8] Kaiming HE, Xiangyu ZHANG, Shaoqing REN et al. Deep residual learning for image recognition. Computer Vision and Pattern Recognition (CVPR). 2016. p. 770-778.

[9] https://github.com/HumanSignal/labelImg

[10] https://rectlabel.com/

[11] https://github.com/GuoleiSun/Indiscernible-Object-Counting/tree/main

[12] Diederik P. KINGMA and Jimmy BA. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[13] Dingkang LIANG, Wei XU and Xiang BAI. An end-to-end transformer model for crowd localization. European Conference on Computer Vision (ECCV). 2022. p. 38-54.

[14] Ayoub KARINE, Thibault NAPOLÉON and Maher JRIDI. Channel-spatial knowledge distillation for efficient semantic segmentation. Pattern Recognition Letters, 2024, vol. 180, p. 48-54.