

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

En informatique

École doctorale I2S

Unité de recherche LIRMM & MISCA

**Amélioration des modèles de détection d'objets rares au
sein d'images sonar par l'utilisation d'apprentissage actif
et de modèles génératifs**

**Présentée par Hugo RUIZ
le 17 décembre 2025**

**Sous la direction de Marc CHAUMONT
et Jérôme PASQUET**

Devant le jury composé de

Géraldine MORIN, Professeure des Universités, IRIT, INP-ENSEEIH

Présidente

Sébastien LEFÈVRE, Professeur des Universités, IRISA, Université de Bretagne Sud

Rapporteur

Marc CHAUMONT, Maître de Conférences, IRISA et LIRMM, Université de Bretagne Sud (auparavant à l'Université de Nîmes)

Directeur de thèse

Jérôme PASQUET, Maître de Conférences, TETIS, Université Paul-Valéry

Co-encadrant de thèse



**UNIVERSITÉ
DE MONTPELLIER**

Remerciements

Je souhaite exprimer ma profonde gratitude à toutes les personnes qui ont, de près ou de loin, contribué à la réalisation de cette thèse.

Tout d’abord, je tiens à remercier mon directeur de thèse, Marc CHAUMONT, ainsi que mon co-encadrant, Jérôme PASQUET, pour leur confiance, leurs conseils et leur accompagnement tout au long de ces années. Leur expertise scientifique, leur disponibilité et leur bienveillance ont été très importants dans l’aboutissement de ce travail. Sans eux, je n’aurais pas pu mener à bien ce projet de recherche, et je leur suis profondément reconnaissant pour tout ce qu’ils m’ont apporté.

Je remercie également les rapporteurs, Géraldine MAURIN et Sébastien LEFÈVRE, pour l’attention portée à ce manuscrit, pour leurs commentaires constructifs et pour l’honneur qu’ils me font en acceptant d’évaluer ces travaux.

J’adresse aussi ma gratitude aux co-financeurs de ce projet, l’Agence de l’Innovation de Défense (AID) pour la Direction Générale de l’Armement (DGA) et Thales, dont le soutien financier a rendu possible la réalisation de cette thèse.

Mes remerciements vont également à mes collègues du LIRMM, avec qui j’ai eu le plaisir d’échanger, de collaborer et parfois de partager des discussions qui allaient bien au-delà de la science.

Cela a commencé avant la thèse avec Frédéric et sa joie de vivre inégalable. Gérard a aussi fait partie de ce projet et a apporté son professionnalisme, sa justesse, sa rigueur et sa petite touche d’humour, je le remercie pour cela. Emmanuel, pour son expertise dans les calculateurs et ses précieux conseils qui m’ont évité de nombreuses heures de débogage. Noura et Quentin, pour les nombreuses sorties de décompression, les discussions passionnantes et les moments de convivialité partagés.

Mehdi, tu resteras le premier collègue avec qui j’ai passé des heures à échanger professionnellement et amicalement, et je te remercie pour ta disponibilité, tes conseils et ta bonne humeur. Anass, mon cobu, avec qui j’ai partagé de bons moments de rigolades et la passion pour la F1. Thibault, pour les nombreuses pauses cafés à parler de tout et de rien, mais en étant extrêmement précis. Tristan, même si c’était rare de se croiser avec ton entreprise, ça me faisait plaisir de pouvoir manger et qu’on puisse parler, je te souhaite bien du courage pour la suite de ta thèse. Cyril, toujours à 200% dans ses projets, mais aussi dans les échanges et les

délires qu'on ait pu avoir. Robin, compagnon de thèse, d'ATER et de pauses café, autant sur les moments de travail que de détente, un plaisir d'échanger quelques bonbons avec toi. Marc, ça y est, on a enfin réussi à finir nos thèses, merci pour ta bonne humeur, le soutien mutuel et l'entraide qu'on a pu s'apporter pendant ces derniers mois qui n'ont pas été simples du tout, félicitations ! Maxime, tes nombreuses venues au LIRMM après la fin de ta thèse pour garder le lien, discuter de YOLO mais aussi de tout le monde informatique, toujours un plaisir d'avoir pu t'avoir dans la place vacante de mon bureau.

La meilleure pour la fin, Pauline, soutien infailible, depuis le premier jour au LIRMM jusqu'à la fin de la thèse, même à l'autre bout de la France, tu as réussi à être là pour moi et je ne te remercierai jamais assez pour ça, tu es une collègue et une amie en or. Merci pour tout ce que tu m'as apporté pendant ces années, je te souhaite le meilleur pour la suite, tu le mérites amplement !

Il m'est difficile de citer toutes les personnes que j'ai pu croiser durant ces années de thèse, que ce soit au LIRMM ou ailleurs, je vous remercie.

Enfin, je souhaite remercier mes amis, qui m'ont permis de garder un équilibre nécessaire en dehors de la thèse, que ce soit par des discussions, des rires ou des moments partagés. Il m'est impossible de citer tout le monde un par un, ne m'en voulez pas d'avoir fait des groupes mais je pense à toutes les personnes qui se sentent incluses.

Les amis du tennis et du padel, de Vendargues ou de Teyran, tellement d'apéros, d'entraînements, de matchs, de compétitions et de moments de convivialité partagés, merci à vous tous pour ces moments de détente et de plaisir. En revanche, je ne vous remercie pas pour les nombreuses défaites au palet, même si au fond j'en ressors que plus fort mentalement.

Mes copains de bière, de soirées, de festivals, de voyage et de rigolades, vous m'avez permis de décompresser quand j'en avais besoin. Autant ceux qui sont proches géographiquement, que ceux qui sont éloignés mais avec qui un lien est resté présent malgré la distance. Nous avons partagé tellement de moments importants et inoubliables, et je vous remercie pour tous ces moments de joie.

Il y a quand même une personne qui ne rentre dans aucune catégorie, mon meilleur ami, mon complice de toujours, mon frère d'une autre mère, Arnaud, avec qui j'ai partagé tant de choses depuis notre enfance, et qui a toujours été là pour moi, dans les bons comme dans les mauvais moments. Merci d'être dans ma vie.

Cette thèse n'aurait pas été possible sans le soutien de ma famille, qui a toujours été là pour moi, à n'importe quel moment.

Mes grands-parents, Marcel, Kéké et Jacky, malgré le décalage de génération, vous avez toujours veillé à mon bien-être et je vous en suis reconnaissant. À mon grand-père Cristobal, je ne t'oublie pas, jusqu'à tes derniers moments tu m'as soutenu et encouragé, cette thèse est aussi la tienne, de maçon à docteur Ruiz.

Aux cousines et cousins, tantes et oncles, malgré la distance qui nous sépare, vous avez toujours répondu présents quand il le fallait, merci à vous.

Clément, mon frère, pour m'avoir remis sur pied, littéralement, un bon nombre

de fois quand mon corps n'arrivait plus à suivre et les nombreuses parties de jeux vidéo, de tennis ou de padel que nous avons pu partager, merci pour tout.

À mes parents, pour leur amour inconditionnel, leur soutien sans faille et leur confiance en moi. Vous avez toujours cru en mes capacités et m'avez encouragé à poursuivre encore plus loin, même quand c'était difficile. Je vous suis profondément reconnaissant pour tout ce que vous avez fait pour moi.

À toutes celles et ceux qui ont contribué à ce parcours, par leur soutien, leurs encouragements ou simplement leur présence : un sincère merci.

Table des matières

Liste des Figures	xi
Liste des Tables	xiii
Glossaire	xiii
Introduction	1
Contexte de la thèse	2
Problématiques	3
Organisation du manuscrit	3
 I Contexte de la thèse	 5
1 L'imagerie sonar	7
1.1 Onde acoustique	10
1.2 Propagation d'une onde	12
1.2.1 Concepts fondamentaux des ondes acoustiques	13
1.2.2 Mécanismes de propagation de l'onde	15
1.2.3 Interactions avec des objets	16
1.3 Les différents sonars	19
1.3.1 Sondeur latéral	20
1.3.2 Sondeur multifaisceaux	20
1.3.3 Sonar à ouverture synthétique (SAS)	22
 2 Présentation des bases de données sonar	 25
2.1 Données simulées (base SIMSON)	26
2.2 Données réelles	29
2.3 Construction des ensembles d'apprentissage	30
2.3.1 Modifications des données	30
2.3.2 Structuration des bases	35
 3 Détection d'objets	 39
3.1 Introduction	40
3.2 Principe des détecteurs et RPN (<i>Region Proposal Network</i>)	40
3.2.1 Détection d'objets	40

3.2.2	Approches initiales basées sur la recherche de régions	41
3.2.3	Détecteurs à deux étapes et Region Proposal Networks (RPN)	42
3.3	You Only Look Once (YOLO)	47
3.3.1	Historique et évolutions de You Only Look Once (YOLO)	47
3.3.2	YOLOv5	49
3.4	Détecteur DETR basé sur les ViT	57
3.4.1	Mécanisme d'auto-attention	58
3.4.2	Architecture DEctor with TRansformer (DETR)	58
3.4.3	Fonction de perte et optimisation	59
II	Apprentissage actif dans le cadre de la détection d'objets	61
4	Apprentissage actif	63
4.1	Introduction	64
4.2	Stratégies d'apprentissage actif	65
4.2.1	Incertitude	65
4.2.2	Diversité	66
4.2.3	Hybride	67
4.2.4	Autre	67
4.3	Méthodes pour la détection d'objets	68
4.3.1	Incertitude	69
4.3.2	Diversité	75
4.3.3	Hybride	78
5	Apprentissage actif dans le cas de la détection : une approche de sélection par clustering	81
5.1	Méthodologie	82
5.2	Définition du critère de sélection automatique	85
5.2.1	Aléatoire	85
5.2.2	Confiance des prédictions	86
5.2.3	Clustering grâce aux cartes de caractéristiques	89
5.2.4	Choix de la méthode à utiliser	92
5.3	Détails des expérimentations	93
5.3.1	Protocole détaillé	93
5.3.2	Résultats	97
III	Génération contrôlée d'images	105
6	Réseaux génératifs	107
6.1	Adaptation de domaine	108
6.1.1	Définition et enjeux	108
6.1.2	Décalage de distribution et particularités des images sonar	108
6.1.3	Stratégies générales d'adaptation	109

6.1.4	Apprentissage non-supervisé de la translation inter-domaines, sans appariement	111
6.1.5	Vers l'usage de modèles génératifs	112
6.2	Principe	112
6.3	CycleGAN	115
6.3.1	Architecture	115
6.3.2	Apprentissage	115
6.3.3	Conclusion	118
6.4	StarGAN	118
6.4.1	Architecture	119
6.4.2	Apprentissage	121
6.4.3	Objectif global	122
7	Adaptation des GAN aux contraintes imposées	125
7.1	Modifications apportées	126
7.1.1	CycleGAN	126
7.1.2	StarGAN	129
7.2	Résultats	132
7.2.1	Cycle GAN	132
7.2.2	StarGANv2	135
	Conclusion & Perspectives	141
	Conclusion	141
	Perspectives	141
	Liste des publications	147
	Bibliographie	148
	Annexes	173
	Notions fondamentales sur les réseaux de neurones	175
	NN	175
	Réseaux neuronaux artificiels (RNA)	175
	Différentes architectures de réseaux neuronaux	176
	CNN	180
	Convolution	182
	Fonction d'activation	187
	Pooling	189
	Fonction de pertes	190
	Rétropropagation	192
	Métriques	192
	Terminologie de l'évaluation des prédictions	193
	Matrice de confusion	193
	Exactitude, précision, rappel, F1 score, spécificité	194
	Receiver Operating Characteristic (ROC)	194

Courbe précision-rappel (PR)	195
Intersection over Union (IoU)	196
Average Precision (AP) / Mean Average Precision (mAP)	199

Table des figures

1	Différences image optique/sonar	1
2	Cas d'usage des systèmes sonar	2
1.1	Applications militaires sonar	9
1.2	Fonctionnement général d'un système sonar	10
1.3	Principe sonar à balayage latéral	11
1.4	Différences basses/hautes fréquences	12
1.5	Profil de célérité	14
1.6	Influence absorption acoustique	16
1.7	Phénomène de réflexion	17
1.8	Réflexion et réfraction	17
1.9	Trajet de la réfraction	18
1.10	Images sondeur latéral	20
1.11	Acquisition sonar multifaisceau	21
1.12	Cartographie 3D sonar	22
1.13	Comparaison SAS/sonar latéral	23
1.14	Réduction bruit image sonar	23
2.1	Types d'objets sur fonds variés	27
2.2	Exemples images base réelle	30
2.3	Données simulées non séparées	31
2.4	Passage échelle décibels	32
2.5	Retrait colonne d'eau	33
2.6	Gradient et valeurs moyennes	34
2.7	Détection/suppression colonne d'eau	35
2.8	Exemples images bases préparées	36
3.1	Différences tâches traitement d'images	41
3.2	Étapes du R-CNN	43
3.3	Pipeline Fast R-CNN	44
3.4	Pipeline Faster R-CNN	45
3.5	Region Proposal Network (RPN)	45
3.6	Filtrage propositions région	46
3.7	Chronologie versions YOLO	47
3.8	Avancées majeures YOLO	48
3.9	Performances versions YOLO	49
3.10	Structure YOLOv5	50

3.11	CSPNet	51
3.12	Module C3	51
3.13	Bloc SPPF	52
3.14	Réseau PANet	53
3.15	Processus DETR	57
3.16	Architecture DETR	59
4.1	Fonctionnement apprentissage actif	64
4.2	Score moindre confiance	65
4.3	Architectures réseaux	73
4.4	Regroupement par clustering	76
4.5	Traitement clusters déséquilibrés	79
5.1	Répartition dossiers entraînement	82
5.2	Illustrations des différentes étapes du protocole d'apprentissage actif	84
5.3	Images avec des confiances faibles	86
5.4	Évolution du nombre d'images extraites	87
5.5	Images de la méthode confiance moyenne	88
5.6	Distributions des images extraites	88
5.7	Rides de sable et cartes de caractéristiques	90
5.8	Clusters basés cartes de caractéristiques	90
5.9	Visualisation t-SNE des centroïdes	91
5.10	Images avec objet extraites	92
5.11	Création nouvelles bases apprentissage actif	95
5.12	Apprentissages à effectuer	96
5.13	Processus évaluation	97
5.14	Courbes ROC méthode confiance moyenne	99
5.15	Courbes ROC méthode aléatoire	101
5.16	Évolution TP/FN/FP méthode confiance moyenne	103
5.17	Évolution TP/FN/FP méthode aléatoire	103
6.1	Adaptation de domaine	110
6.2	Adaptation de domaine par translation	110
6.3	Cycle d'apprentissage GAN	113
6.4	Composants CycleGAN	115
6.5	Reconstruction CycleGAN	117
6.6	Vue d'ensemble StarGANv2	119
6.7	Composition du StarGANv2	120
7.1	Schéma CycleGAN avec masque	126
7.2	Exemple masque CycleGAN	127
7.3	Transformations encourageantes CycleGAN	128
7.4	Transformations problématiques CycleGAN	129
7.5	Bloc utilisant le masque	130
7.6	Images CycleGAN à 400 étapes	133
7.7	Ombres mal reconstruites CycleGAN	134
7.8	Artefact sévère par le CycleGAN	135

7.9	Résultats StarGANv2 sim2real	136
7.10	Disparition objet StarGANv2	137
7.11	Échec transfert style	137
7.12	Style incohérent StarGANv2	138
7.13	Positionnement partiel mais style incorrect	138
14	Illustration d'un réseau de neurones dit "classique"	176
15	Schéma exemple d'un Multi-Layer Perceptron (MLP)	177
16	Illustration de la compression de données. (Source Faust)	178
17	Schéma d'un Auto-Encoder (AE)	179
18	Utilisation Auto-Encoder	179
19	Noyau de convolution de dimension 3×3	183
20	Illustration d'une convolution discrète.	183
21	Exemple de calcul d'une convolution discrète avec $p = 1$ et $s = 2$	184
22	Convolution par canal	185
23	Filtre multi-noyaux	186
24	Schéma du processus global d'un neurone artificiel.	186
25	Fonction d'activation ReLU	187
26	Représentations graphiques des fonctions d'activation majeures.	188
27	Dérivées fonctions d'activation	189
28	Max/Average Pooling	190
29	Schéma exemple d'une matrice de confusion	193
30	Courbe ROC	195
31	Exemple d'une courbe Précision-Rappel (PR)	196
32	Ensembles pour IoU	197
33	Illustration de l'intersection avec les ensembles A et B	197
34	Illustration de l'union avec les ensembles A et B	198
35	Illustration de l'Intersection over Union (IoU) avec les aires qui sont utilisées en vert	198
36	Exemples de résultats lorsque le seuil de l'IoU est fixé à 0.5	198
37	Courbe Précision-Rappel et AP	199

Liste des tableaux

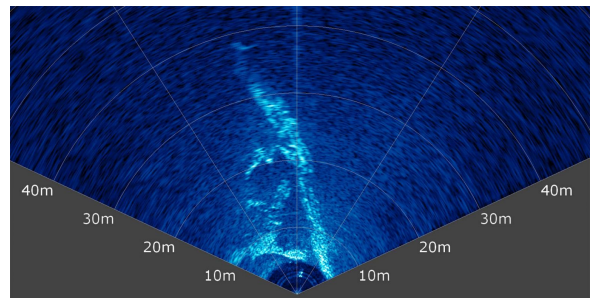
2.1	Paramétrage sonar Klein 5000	28
2.2	Paramètres croisés base	29
2.3	Caractéristiques bases données	36
5.1	Évolution ratio images objets	91
5.2	Comparaison images ajoutées clustering/moyenne confiances	98
5.3	Comparaison AUC 0-1 clustering/moyenne confiances	98
5.4	Comparaison AUC 0.5-1 clustering/moyenne confiances	98
5.5	Comparaison AUC 0.5-1 clustering/aléatoire	100
5.6	Comparaison images ajoutées clustering/aléatoire	100
5.7	Résultats TP/FN/FP clustering/moyenne des confiances	102
5.8	Résultats TP/FN/FP clustering/aléatoire	103
7.1	Résultats FID StarGANv2	139
7.2	Résultats LPIPS pour StarGANv2	139

Introduction

L'exploration des profondeurs océaniques est depuis longtemps un domaine clé des avancées scientifiques et technologiques, où l'imagerie sonar joue un rôle essentiel dans l'étude des environnements sous-marins. Les techniques ayant recours à la propagation d'ondes sonores se sont montrées plus efficaces que celles utilisant des ondes électromagnétiques. Les ondes sonores permettent notamment de parcourir plus efficacement de longues distances dans le milieu marin. Ainsi, les systèmes sonar sont plus adaptés que les méthodes d'imagerie optique. En effet, les facteurs comme la faible visibilité, l'atténuation de la lumière et la turbidité des eaux rendent les techniques d'imagerie traditionnelles inefficaces. Les différences entre ces types d'images sont illustrées dans la figure 1.



(a) Image RVB issue d'un appareil photo.



(b) Image issue d'un sonar. [Robotics, 2024]

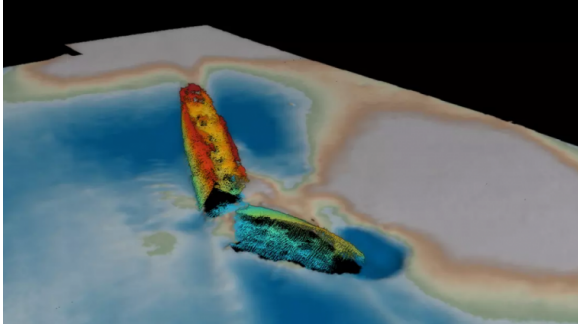
FIGURE 1 – Différences entre une image optique et une image sonar. (1a Une image obtenue grâce à un téléphone portable ; 1b Une image capturée avec un système sonar)

Les ondes acoustiques sont naturellement utilisées par des espèces comme les dauphins et les baleines pour communiquer sur de longues distances, parfois sur plusieurs kilomètres, dans le milieu marin. De son côté, l'Homme a initialement exploité les ondes acoustiques pour des applications militaires, notamment la détection de sous-marins. L'acoustique sous-marine s'est progressivement étendue à des domaines civils tels que la biologie marine, l'océanographie et la géophysique. Elle permet de détecter, localiser et classifier des cibles, de caractériser les environnements marins et de faciliter les communications sous l'eau.

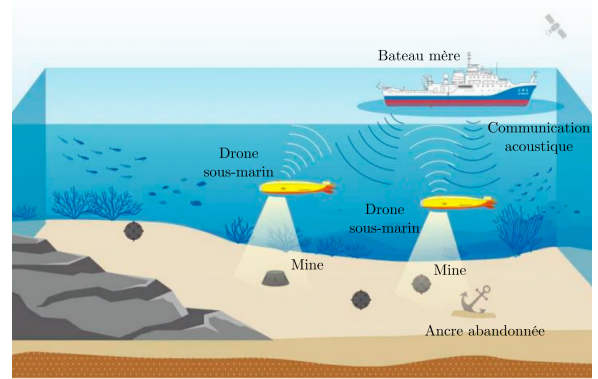
Les images issues de la réflexion acoustique fournissent des informations essentielles sur les objets, les reliefs et les environnements sous-marins. Voici quelques exemples d'utilisation de système sonar, utilisées pour l'exploration sous-marine : localisation et cartographie des

épaves, des sites archéologiques immergés et des formations géologiques, surveillance des habitats aquatiques, analyse de la répartition des espèces, étude des changements environnementaux, détection des mines, navigation des sous-marins et surveillance des menaces sous-marines.

La figure 2 montre l'exemple d'une cartographie d'une épave de bateau faite grâce à un sonar multifaisceaux, ainsi que l'utilisation de sonar latéral pour la détection de mines dans le contexte militaire.



(a) Exemple d'une cartographie d'une épave de bateau [International, 2024].



(b) Illustration de l'utilisation de systèmes sonar pour la détection de mines [Tang et al., 2023].

FIGURE 2 – Deux illustrations de cas d'usage de systèmes sonar dans divers domaines.

Dans le cadre de cette thèse, nous étudions l'imagerie sonar et plus exactement la détection automatique de mines.

Contexte de la thèse

Au fil des années, diverses techniques ont été développées pour surmonter les défis liés à l'analyse automatique des images sonar. Les premières approches s'appuyaient sur l'extraction manuelle de caractéristiques et des méthodes classiques d'apprentissage automatique, telles que les *Support Vector Machine (SVM)* [Cortes et Vapnik, 1995] et les arbres de décision [Belson, 1959; Breiman et al., 2017]. Cependant, ces méthodes présentent des limites, notamment leur dépendance à des caractéristiques conçues manuellement, souvent peu adaptées à des ensembles de données variés.

Les avancées dans le domaine de l'intelligence artificielle ont aidé au développement du domaine de l'imagerie sonar, notamment grâce à l'apprentissage profond. Les réseaux neuronaux profonds [LeCun et al., 1989; Rumelhart et al., 1986], *Deep Neural Network (DNN)*, capables d'apprendre et d'extraire automatiquement des caractéristiques à partir de données brutes, constituent une approche intéressante pour traiter la complexité de ce type de données.

L'application des méthodes d'apprentissage profond à l'imagerie sonar se heurte à plusieurs défis spécifiques. Contrairement aux images optiques, les images sonar diffèrent de par leur mode de génération et de par la nature des informations qu'elles transmettent. Elles sont souvent caractérisées par une faible résolution, un bruit important et des distorsions environnementales, comme les réverbérations et les interférences acoustiques. Ces particularités limitent l'efficacité des techniques traditionnelles de détection d'objets et compliquent l'adaptation des modèles d'apprentissage profond conçus à l'origine pour des données optiques.

Problématiques

Dans ce contexte, la détection d’objets dans les images sonar nécessite une précision élevée. Améliorer la fiabilité et l’exactitude des systèmes de détection reste une priorité. Les techniques d’apprentissage profond appliquées au traitement des données sonar permettent de relever ces défis tout en augmentant les performances des systèmes. Cependant, les méthodes d’apprentissage supervisé nécessitent des bases de données annotées pour obtenir de bons résultats. La rareté des données disponibles nous place dans une situation où il est nécessaire d’utiliser l’apprentissage supervisé avec de petits ensembles de données.

Pour tirer le meilleur parti des données disponibles, l’apprentissage actif constitue une approche pertinente. En ciblant les échantillons les plus informatifs à annoter, il réduit la dépendance à de grands ensembles de données labellisés. Ce constat est d’autant plus pertinent en imagerie sonar, où le processus d’annotation se révèle particulièrement complexe et coûteux.

Organisation du manuscrit

Ce manuscrit de thèse est structuré en trois grandes parties, précédées d’une introduction générale et suivies d’une conclusion.

La partie I présente le contexte de la thèse. Elle comprend un état de l’art sur l’imagerie sonar, une description des jeux de données exploités, ainsi qu’un chapitre sur les détecteurs d’objets, couvrant différentes familles de modèles (RPN [Ren *et al.*, 2016], YOLO [Redmon *et al.*, 2015], DETR [Carion *et al.*, 2020]).

La partie II est dédiée à l’apprentissage actif, avec un état de l’art ciblé sur ses applications en détection d’objets. Elle regroupe les travaux visant à sélectionner efficacement les données à annoter dans un contexte de données limitées.

La partie III porte sur la génération d’images sonar à l’aide de réseaux adversaires génératifs, *Generative Adversarial Networks (GAN)*. Elle comprend un état de l’art sur les GAN appliqués à la génération d’images (StarGAN [Choi *et al.*, 2018, 2020], CycleGAN [Zhu *et al.*, 2017]), ainsi que les contributions développées au cours de cette thèse, incluant les adaptations architecturales et les résultats expérimentaux.

La conclusion revient sur les principales contributions et ouvre sur plusieurs perspectives de recherche.

Première partie

Contexte de la thèse

Chapitre 1

L'imagerie sonar

Sommaire

1.1	Onde acoustique	10
1.2	Propagation d'une onde	12
1.2.1	Concepts fondamentaux des ondes acoustiques	13
1.2.2	Mécanismes de propagation de l'onde	15
1.2.2.1	Notation logarithmique en sonar	15
1.2.2.2	Pertes dans la propagation	15
1.2.3	Interactions avec des objets	16
1.2.3.1	Réflexion	16
1.2.3.2	Réfraction des ondes	17
1.2.3.3	Rétrodiffusion des ondes d'une cible	18
1.2.3.4	Perte de réflexion du fond marin	19
1.3	Les différents sonars	19
1.3.1	Sondeur latéral	20
1.3.2	Sondeur multifaisceaux	20
1.3.3	Sonar à ouverture synthétique (SAS)	22
1.3.3.1	Principe	22
1.3.3.2	Traitement des données	23

Ce chapitre présente les fondements physiques et technologiques de l'imagerie sonar, principal domaine d'application de cette thèse. L'acoustique sous-marine y est introduite à travers l'étude de la propagation des ondes et de leurs interactions avec le milieu marin, éléments déterminants pour la qualité et la portée des mesures. Les principaux types de sonars sont ensuite décrits, avec leurs principes de fonctionnement et leurs applications.

L'ensemble fournit les bases nécessaires à la compréhension des enjeux liés à l'acquisition d'images sonar et aux problématiques traitées dans les chapitres suivants, notamment la détection automatique d'objets dans les images sonar.

Le développement de méthodes de transmission et de communication à longue distance, telles que le téléphone, la radio et la télévision, représente l'un des accomplissements technologiques majeurs de l'époque moderne. La plupart de ces systèmes de télécommunication exploitent la propagation des "ondes électromagnétiques" [NASA, 2010], qui sont de nature similaire à la lumière. Théorisées par James Clerk Maxwell dans les années 1860 puis démontrées expérimentalement par Heinrich Hertz en 1888, les ondes électromagnétiques ont trouvé dès le début du XX^e siècle des applications concrètes dans les télécommunications, grâce aux travaux pionniers de chercheurs tels qu'Alexandre Popov, Édouard Branly et Nikola Tesla. Depuis, elles se sont imposées comme un moyen de transmission d'informations particulièrement efficace, mais également comme un outil d'exploration et de surveillance de l'environnement, notamment à travers leur utilisation dans les systèmes de "*RAdio Detection And Ranging (RADAR)*" [Rao *et al.*, 2020 ; Yan *et al.*, 2021 ; Zhang *et al.*, 2022].

Il s'avère que les ondes électromagnétiques sont inefficaces dans l'eau. En effet, l'eau, et en particulier l'eau salée, a un caractère dissipatif lié à sa forte conductivité, ce qui atténue fortement ces ondes les rendant inutilisables. Le seul moyen de transmission dans la mer est l'utilisation des "ondes acoustiques" [NASA, 1988]. La propagation des ondes sera détaillée dans la section 1.2, mais de manière générale, les ondes acoustiques correspondent à des vibrations mécaniques qui engendrent des variations de pression au sein du milieu de propagation. L'eau de mer étant un milieu très favorable, les ondes acoustiques ont une meilleure vitesse de propagation et peuvent atteindre des profondeurs plus importantes, et surtout, elles subissent moins d'atténuation augmentant la distance de propagation de celles-ci. Les signaux utiles peuvent être facilement perturbés par des échos parasites (végétation, fonds particuliers, débris...), masqués ou occultés par des éléments de l'environnement, ou encore noyés dans les signaux ambiants (bateaux, poissons...).

Depuis longtemps il est connu que les ondes acoustiques fonctionnent bien dans le milieu marin, leur utilisation reste tout de même récente à cause des limitations technologiques. Au XXI^e siècle, le sonar est utilisé pour de nombreuses tâches dans le milieu marin, la mesure des caractéristiques du milieu (courants, composition du fond...), la transmission des signaux (données acquises, messages...) ou bien de la détection et localisation d'obstacles ou de cibles, dans la pêche ou pour une utilisation militaire.

Les sonars sont souvent associés aux sous-marins et à une application militaire, mais il y a aussi d'autres utilisations civiles. Il en existe plusieurs, par exemple, les sondeurs bathymétriques servant dans la mesure de la hauteur d'eau, les sonars latéraux permettant de faire de l'imagerie très précise du fond et les sondeurs multifaisceaux pour de la cartographie de fond marin. Tous se basent sur le même fonctionnement, l'onde acoustique, développée dans la section 1.1.

D'un point de vue militaire, le sonar s'est imposé comme un outil stratégique depuis les travaux pionniers de Paul Langevin [Langevin, 1918] durant la Première Guerre mondiale,

centrés sur la transmission de signaux et la détection acoustique active. Son importance s'est accrue pendant la Guerre froide, où l'émergence des sous-marins nucléaires a entraîné une évolution des systèmes sonar, orientée vers l'analyse de vastes zones maritimes pour répondre aux enjeux de la guerre sous-marine.

Pour une application militaire, la figure 1.1 illustre les différents types de sonars utilisés dans ce contexte.

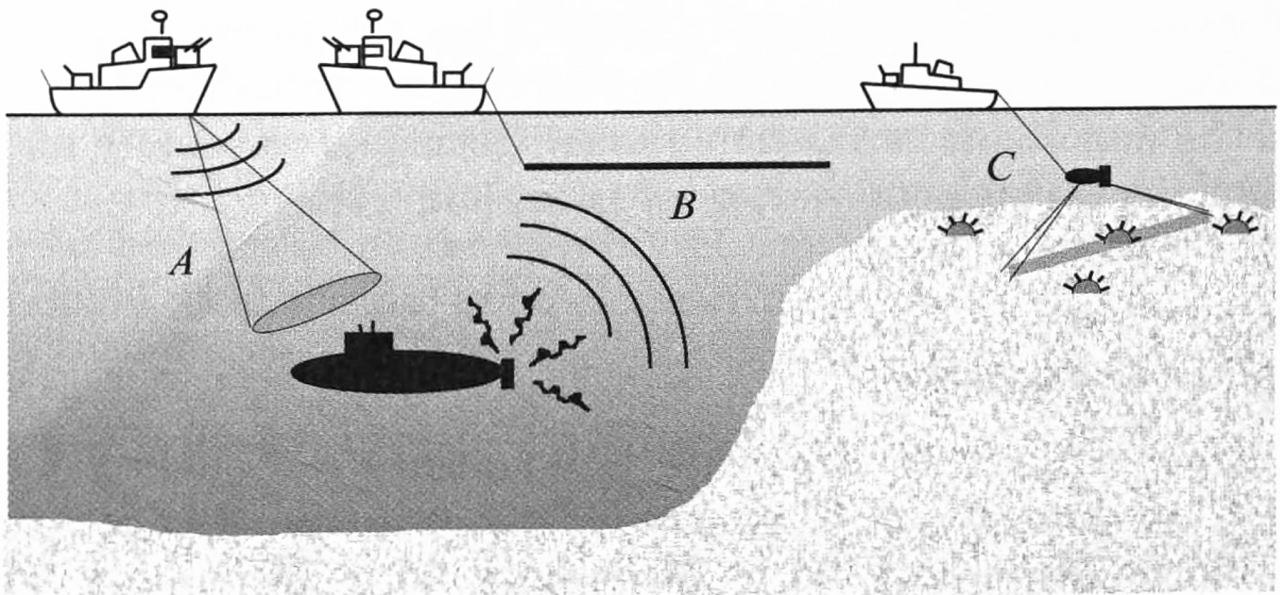


FIGURE 1.1 – Applications militaires (A : sonar actif - B : sonar passif - C : sonar actif chasse mines [Lurton, 1998])

La figure 1.1.A illustre un sonar actif cherchant à détecter un sous-marin. Ce type de sonar fonctionne en émettant des ondes acoustiques et en analysant les échos réfléchis par des objets sous-marins. La figure 1.1.B montre un sonar passif, qui, contrairement au sonar actif, n'émet pas d'ondes. Il se contente de recevoir et analyser les bruits produits par des sources sous-marines, tels que les moteurs ou les hélices. Quant à la figure 1.1.C, elle représente un sonar actif spécialisé dans la détection et la localisation de mines sous-marines. La fréquence des ondes émises est plus élevée afin d'obtenir une résolution plus fine pour détecter de petits objets.

Le traitement numérique des signaux fut un élément marquant dans l'évolution technologique apporté aux recherches sous-marines. Les calculateurs ont permis d'améliorer les capacités, mais la limite des sonars passifs fut atteinte et la tendance de recherche s'est inversée pour revenir sur les techniques des sonars actifs. Les conflits à la fin du XX^e siècle ont mis en évidence l'importance de maîtriser les technologies sonar, en particulier l'usage de certaines fréquences permettant une couverture de surface étendue, face aux menaces des sous-marins d'attaque et de la guerre des mines.

Le terme "sonar" est souvent utilisé pour désigner un ensemble de systèmes permettant de détecter ou bien de localiser une cible sous-marine grâce aux ondes acoustiques. Pour effectuer ces tâches, les sonars se basent sur la réception de signaux.

Le sonar passif est basé sur l'écoute et l'analyse des sons présents dans l'environnement sans émettre le moindre signal. Il est donc composé uniquement d'un récepteur qui va capter les signaux provenant de toutes les sources, comprenant les signaux biologiques, bruits ambiants ou bien les sources humaines comme les bateaux ou sous-marins. Il essaye de capter et d'analyser les sons afin d'extraire des informations utiles. Ce type de sonar est utilisé pour détecter des

communications sous-marines, identifier des caractéristiques d'un bateau, ou bien, surveiller la vie maritime, tout en restant discret, car il n'émet pas d'ondes.

Quant au sonar actif, le système émet son propre signal dans l'eau et écoute les échos qui sont réfléchis sur les objets de l'environnement. Ces systèmes ont un émetteur particulier qui génère le signal et un récepteur qui détecte et analyse les échos. Le signal qui est généré a des caractéristiques particulières qui permet au receveur d'analyser l'onde captée, il prend en compte la fréquence, durée d'émission et la modulation. En identifiant les propriétés des échos, le sonar actif peut obtenir des informations sur la cible comme la localisation, la distance, la taille, la forme et d'autres caractéristiques. C'est le type de sonar le plus utilisé lorsqu'il est nécessaire de détecter la présence de cibles ou d'avoir des mesures précises.

Le fonctionnement d'un sonar, tel que décrit par les deux types précédents, est résumé dans le schéma de la figure 1.2 ci-dessous.

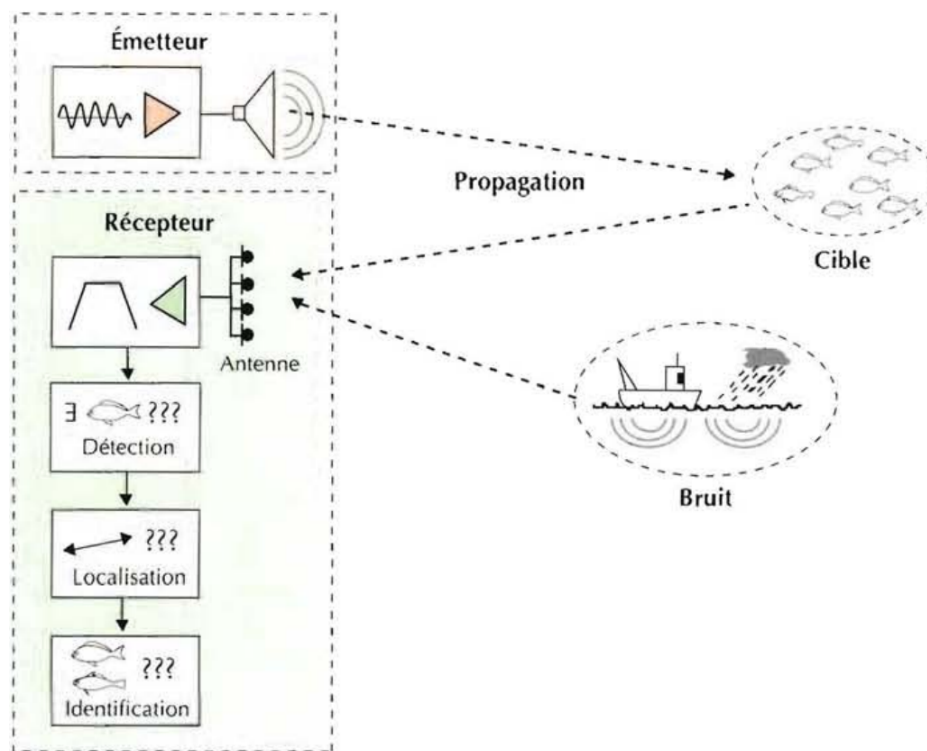


FIGURE 1.2 – Fonctionnement général d'un système sonar [Lurton, 1998].

1.1 Onde acoustique

Le son est une onde de pression qui se propage à travers un milieu qu'il soit solide, liquide ou gazeux. Il a été démontré que le milieu aquatique est propice à la diffusion des ondes acoustiques [Sutton, 1979 ; Grelowska et Kozaczka, 2014] pour de multiples raisons.

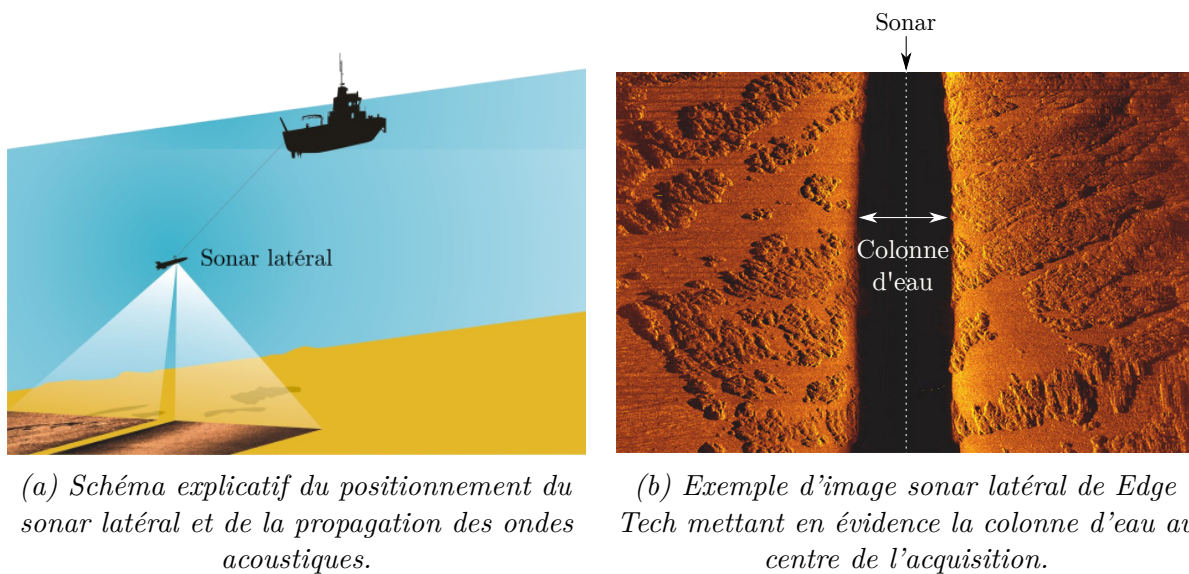


FIGURE 1.3 – Illustration du principe de fonctionnement du sonar à balayage latéral et de la représentation acoustique de la colonne d'eau.

La figure 1.3a illustre le principe d'acquisition d'images par sonar à balayage latéral. Ce capteur, généralement tracté à distance derrière un navire, émet des ondes acoustiques latéralement. Les signaux traversent l'eau et interagissent avec le fond marin. Les échos résultant de ces interactions sont captés par le récepteur du sonar, puis exploités pour former une image acoustique du relief sous-marin.

Sur l'exemple d'image dans la figure 1.3a, la zone située directement sous le sonar, entre l'émetteur et le fond marin, correspond à la colonne d'eau. Celle-ci apparaît en noir ou en faible intensité, en raison de l'absence de réflecteurs denses capables de produire un retour acoustique significatif. Ce contraste traduit l'absence de diffusion notable dans cette région.

L'imagerie sous-marine nécessite une grande précision. Il est donc nécessaire d'acquérir des images de fond marin de haute résolution, ce qui peut être réalisé par les échosondeurs paramétriques ou les sonars à balayage latéral, qui sont tous les deux utilisés pour de la cartographie ou des recherches sous-marines. Ces technologies peuvent traverser la colonne d'eau pour identifier et analyser des caractéristiques détaillées, des objets qui ne sont pas visibles par des moyens optiques. Grâce à ces dispositifs, il est possible de recueillir des données sur l'environnement sous-marin, notamment des mesures de profondeur, de cartographier le fond et de caractériser les sédiments du fond.

Foote [2017] a étudié la précision des mesures acoustiques sous-marines à basses fréquences et analysé l'importance des différentes gammes de fréquences pour la précision des mesures sonar. Les fréquences peuvent être découpées en plusieurs gammes, les basses, les moyennes et les hautes.

Les basses fréquences (1 kHz à 10 kHz) sont des ondes pouvant parcourir des centaines de kilomètres dans des conditions favorables, elles sont utilisées pour la détection et surveillance des eaux profondes. L'étude de Wang et Jiao [2019] traite de la technologie "*Track Before Detect (TBD)*" utilisant un sonar actif à basse fréquence afin de détecter et suivre les sous-marins sur de grandes distances.

Les moyennes fréquences (10 kHz à 100 kHz) sont utilisées dans les systèmes de moyenne portée, couvrant quelques kilomètres à plusieurs dizaines de kilomètres. Elles offrent un bon compromis entre portée et résolution, et sont utilisées dans des applications telles que la navigation,

l'étude de la faune sous-marine [Southall *et al.*, 2023] et la détection de mines.

En ce qui concerne les hautes fréquences (100 kHz à 1 MHz), leur portée est fortement réduite, généralement limitée à quelques centaines de mètres. Elles sont adaptées à l'inspection détaillée des fonds marins, des infrastructures sous-marines et des sites archéologiques. Elles permettent d'identifier de petits objets et de capturer des détails topographiques à proximité de la source sonar. L'étude de Cook *et al.* [2019] a validé l'utilisation d'un système sonar à haute fréquence pour déterminer la taille des poissons et observer leur comportement dans des conditions de turbidité avec une faible luminosité. Au-delà du mégahertz, les très hautes fréquences subissent une atténuation rapide limitant leur portée à moins de cent mètres.

Les ondes acoustiques sont efficaces dans diverses conditions et profondeurs sous-marines. La fréquence des ondes influence la distance à laquelle elles peuvent être utilisées. Il est important de noter que la portée peut être affectée par divers facteurs, notamment la température de l'eau, la salinité et la profondeur, qui influent sur la propagation du son dans l'eau. De manière générale, les hautes fréquences interagissent davantage avec le milieu que les basses fréquences, elles produisent des images plus détaillées, utile pour la cartographie par exemple. En revanche, les basses fréquences permettent de couvrir de plus longues distances, car elles ont moins d'interaction avec le milieu leur permettant de garder leur énergie. De plus, les basses fréquences ont un cône de balayage plus large, ce qui leur permet de mieux détecter en eaux profondes tout en couvrant une zone plus vaste.

Ces différences entre les basses et hautes fréquences, notamment en termes de portée et de largeur du cône de balayage, sont illustrées dans la figure 1.4, mettant en évidence l'impact de la fréquence sur la détection en milieu sous-marin.

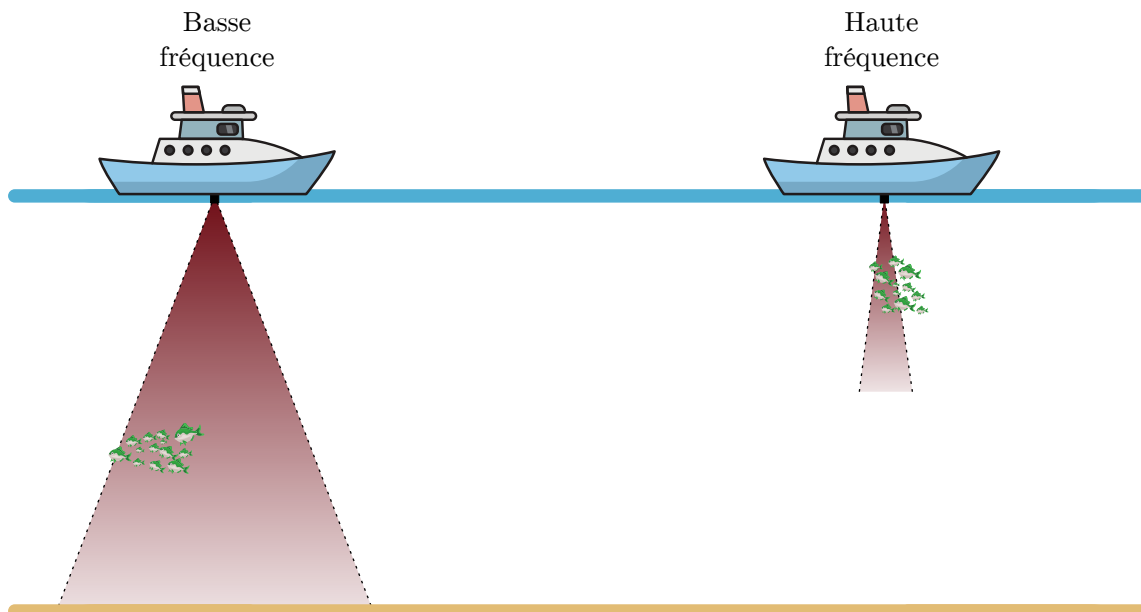


FIGURE 1.4 – Schéma représentant les différences entre les basses et hautes fréquences.

Après avoir vu l'importance des fréquences sonar sur l'utilisation finale d'un système, intéressons-nous maintenant à la manière dont une acoustique se propage à travers le milieu.

1.2 Propagation d'une onde

Dans le domaine de l'acoustique sous-marine, la compréhension des principes de propagation des ondes aide pour mieux appliquer la technologie sonar dans divers domaines. Cette section

aborde les propriétés des ondes sonores lorsqu'elles se propagent dans les environnements aquatiques, leurs interactions avec différents milieux et les implications pour la détection et la caractérisation des objets immergés.

1.2.1 Concepts fondamentaux des ondes acoustiques

Les ondes sonores dans l'eau sont des perturbations mécaniques (compression-dilatation) qui se propagent à travers le milieu. La pression acoustique correspond à l'écart local par rapport à la pression ambiante, elle est mesurée en Pascals ($1 \text{ Pa} = 1 \text{ N.m}^{-2}$).

Dans l'eau, la vitesse du son se propage uniquement par compression. Elle est de l'ordre de 1500 m.s^{-1} soit 5400 km.h^{-1} (à 20°), l'équation permettant de calculer la célérité (c) est la suivante :

$$c = \sqrt{\frac{K}{\rho}} \quad (1.1)$$

avec K étant le module d'élasticité du milieu et ρ sa densité.

Dans la mer, calculer la célérité est plus complexe car elle dépend de plusieurs facteurs, comprenant la profondeur (liée à la pression), la température et la salinité du milieu. Medwin et Clay [1997] ont proposé une équation mettant en lien tous ces paramètres :

$$c(T, S, D) = C_0 + C_T T + C_S S + C_D D + C_{TT} T^2 + C_{TS} TS + C_{TD} TD \quad (1.2)$$

$$c(T, S, D) = 1449.2 + 4.6T - 0.055T^2 + 0.00029T^3 \\ + (1.34 - 0.01T)(S - 35) + 0.016D \quad (1.3)$$

où $c(T, S, D)$ est la vitesse du son en m.s^{-1} , T la température en $^\circ\text{C}$, S la salinité en ‰, D la profondeur en mètres. $C_0, C_T, C_S, C_D, C_{TT}, C_{TS}$ et C_{TD} sont des coefficients empiriques.

Parmi les facteurs influençant la célérité, la profondeur constitue le paramètre dominant, car elle induit les variations les plus significatives de la vitesse de propagation.

La densité (ρ) de l'eau de mer est d'environ 1030 kg.m^{-3} , elle varie en fonction de la température et la salinité. Dans les eaux profondes, l'augmentation de pression peut faire augmenter ρ jusqu'à 1050 kg.m^{-3} . En général, lorsque la densité augmente, la vitesse du son augmente également.

Des études [Dushaw *et al.*, 1992] ont été menées pour comprendre du mieux que possible le comportement de la célérité dans différentes eaux, mais la limitation au niveau matériel de l'époque a poussé à simplifier l'équation NRL II de Del Grosso [Grosso, 1974] qui ne pouvait être correctement modélisée. La figure 1.5 ci-après illustre la célérité en fonction de la profondeur du fond.

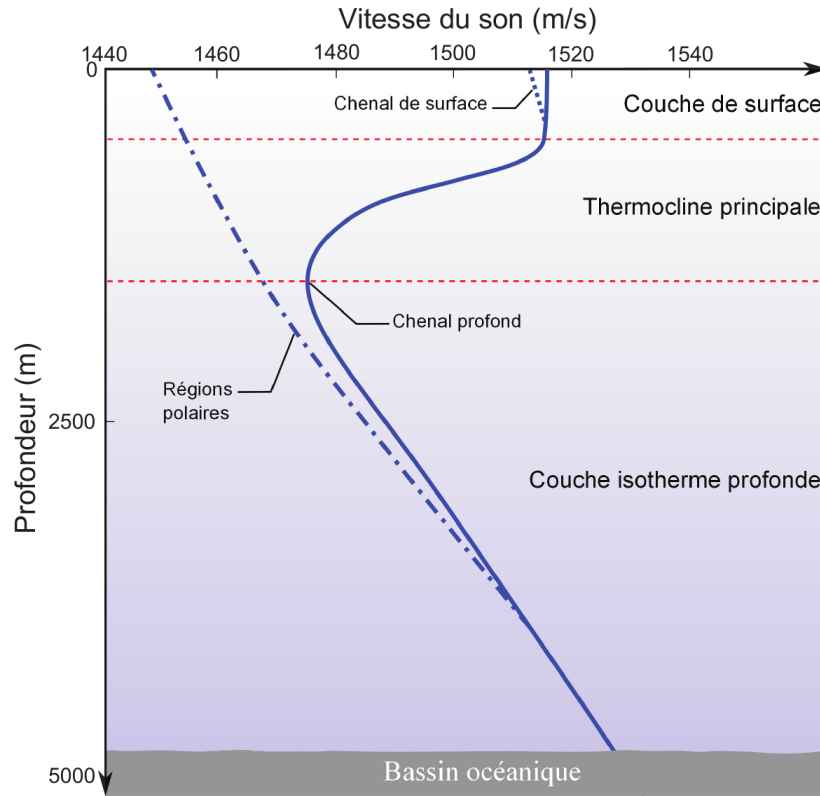


FIGURE 1.5 – Différentes couches d'un profil de célérité (extrait de Le Gall [2015])

La couche de surface, également appelée “zone de mélange” a une température relativement constante, impliquant que la vitesse du son est uniquement dépendante de la profondeur et de la pression. Au-delà d'un certain seuil, les influences thermiques de surface cessent d'agir. Cette zone correspond à la couche thermocline principale, caractérisée par une diminution progressive de la température avec la profondeur, entraînant une réduction de la célérité des ondes acoustiques. La transition entre la thermocline et la couche isotherme profonde marque la profondeur à partir de laquelle la température devient stable. À partir de ce point, la vitesse du son augmente de nouveau, sous l'effet de la pression croissante. Le minimum de vitesse observé à cette interface est appelé le “chenal profond”.

Les acousticiens font l'hypothèse que dans les conditions ordinaires, les phénomènes sonores sont linéaires et qu'il est possible d'étudier les ondes séparément. Une onde plane est une variation sinusoïdale du volume dans laquelle elle se propage, la variation de la pression acoustique l'est aussi, régi par l'équation :

$$p(x, t) = p_0 \sin(\omega t - kx) \quad (1.4)$$

$p(x, t)$ est la pression acoustique à une position x et à un temps t , p_0 est l'amplitude de la pression acoustique (la valeur maximale de la variation de la pression), ω est la pulsation angulaire (en rad.s^{-1}) reliée à la fréquence de l'onde par $\omega = 2\pi f$ et k est le nombre d'ondes en lien avec la longueur d'onde λ par $k = \frac{2\pi}{\lambda}$.

Dans les fluides, cette variation de pression acoustique est à l'origine des ondes sonores que nous percevons, ou qui sont détectées par les sonars dans le cas de la propagation dans l'eau.

Les notions importantes ayant été définies, il est maintenant possible d'approfondir les mécanismes de propagation de l'onde sonore.

1.2.2 Mécanismes de propagation de l'onde

1.2.2.1 Notation logarithmique en sonar

Les grandeurs acoustiques sont notées en décibels (dB), cela facilite le traitement de leur dynamique importante. L'échelle des décibels, une mesure logarithmique, permet une représentation du rapport entre deux intensités : $L_I = 10 \log_{10} \left(\frac{I}{I_0} \right)$ (valable aussi pour des puissances), avec I_0 l'intensité de référence.

Le niveau de pression acoustique est aussi une mesure logarithmique qui exprime la différence entre la pression acoustique d'un son donné et une référence standard. Il est calculé en décibels selon l'équation suivante :

$$P = 20 \log_{10} \left(\frac{p_{eff}}{p_0} \right) \quad (1.5)$$

où p_0 est la pression de référence et p_{eff} représente la valeur quadratique moyenne (RMS) de la pression acoustique instantanée. Elle représente la moyenne des variations de pression dues à l'onde sonore sur une certaine période de temps.

En acoustique sous-marine, il est important de comprendre comment l'énergie sonore diminue avec la distance et comment elle interagit avec divers matériaux. L'échelle logarithmique permet de simplifier les calculs de ces phénomènes, en remplaçant les opérations exponentielles complexes par des additions et soustractions.

Les systèmes sonar doivent souvent détecter des signaux très faibles en présence de bruit de fond. L'échelle des décibels est particulièrement utile pour exprimer le rapport signal sur bruit, "*Signal-to-Noise Ratio (SNR)*", qui est un paramètre servant à évaluer les performances d'un système sonar. Un SNR plus élevé en décibels indique une meilleure capacité de détection.

1.2.2.2 Pertes dans la propagation

La transmission des ondes acoustiques dans l'eau est sujette à des pertes dues aux inhomogénéités environnementales [Akal, 2002] ainsi qu'à la propagation par trajets multiples [Weinberg, 1975]. L'amortissement représente souvent le principal facteur limitant de la propagation acoustique, car le milieu dissipatif absorbe une partie de l'énergie de l'onde sonore, ce sont les pertes par amortissement.

Ce phénomène est quantifié par un coefficient d'absorption α , exprimé en dB.km^{-1} , qui représente la diminution de l'amplitude de l'onde avec la distance. Thorp [1967] fournit une expression pratique, en extension à celle de Horton [1959] sur les basses fréquences, pour déterminer les coefficients d'atténuation du son en fonction de la fréquence f , menant à l'équation (1.6).

$$\alpha(f) = \frac{0.1f^2}{1+f^2} + \frac{40f^2}{4100+f^2} \quad (\text{dB/kyd}) \quad (1.6)$$

$$\alpha(f) = 3.3 \times 10^{-3} + \frac{0.11f^2}{1+f^2} + \frac{43f^2}{4100+f^2} + 2.98 \times 10^{-4}f^2 \quad (\text{dB/km}) \quad (1.7)$$

L'expression (1.7) est la transformation de l'équation (1.6) passant de l'unité kiloyard à l'unité kilomètre. Malgré ses coefficients empiriques, cela permet de déterminer le coefficient α dans le milieu marin. Cette équation est utilisée dans la simulation des modèles précis pour les applications comme le sonar ou les systèmes de communication acoustique.

Il existe d'autres pertes en plus de l'amortissement, notamment celles liées aux trajets multiples étant plus communément appelées "divergence géométrique". Ces pertes de transmission, indépendantes de l'amortissement, résultent de la dispersion de l'énergie dans un milieu homogène.

$$\mathcal{L} = 20 \log_{10}(r) \quad (1.8)$$

avec $\mathcal{L}$ la perte de la transmission en dB et r la distance depuis la source en m.

Lors de la propagation acoustique, les pertes d'intensités totales se composent donc des effets cumulatifs de l'amortissement et de la divergence géométrique, avec une prédominance de l'amortissement.

La figure 1.6 présente les graphiques illustrant les différentes pertes. Le graphique 1.6a montre l'augmentation de l'absorption en fonction de la fréquence utilisée, tandis que le graphique 1.6b illustre la perte liée à la transmission du signal en fonction de la distance parcourue par celui-ci.

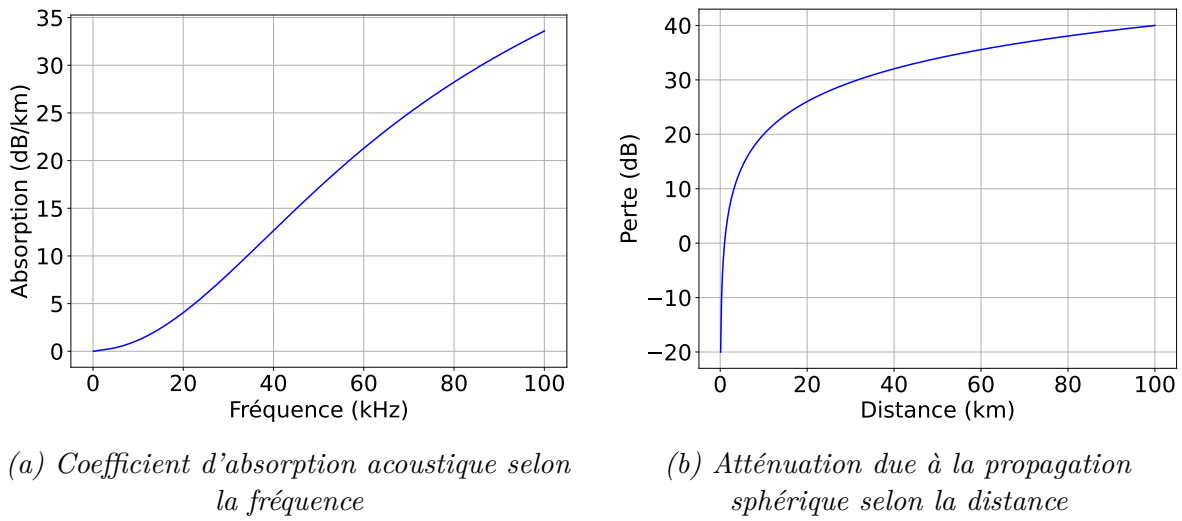


FIGURE 1.6 – Influence de l'absorption acoustique et de la propagation sphérique sur la perte de transmission

Les ondes acoustiques interagissent avec l'eau, mais aussi avec tous les objets immergés. Nous allons maintenant nous concentrer sur les phénomènes de réflexion et de diffusion, conduisant à la génération d'échos et de signaux de rétrodiffusion.

1.2.3 Interactions avec des objets

1.2.3.1 Réflexion

La réflexion se produit lorsqu'une onde sonore rencontre une interface entre deux milieux différents et rebondit dans le milieu d'origine. La loi de la réflexion stipule que l'angle d'incidence (l'angle θ_i , formé par l'onde incidente avec une ligne perpendiculaire à la surface) est égal à l'angle de réflexion (l'angle θ_{re} , formé par l'onde réfléchie avec la même ligne) : $\theta_i = \theta_{re}$, ces angles sont représentés dans la figure 1.7. Ce principe est fondamental pour la technologie des sonars, car il permet de prédire comment les ondes sonores reviennent vers le récepteur après avoir rencontré des objets sous-marins ou le fond marin.

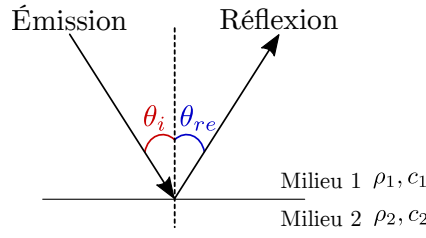


FIGURE 1.7 – Illustration du phénomène de réflexion entre deux milieux.

Plusieurs facteurs influent sur l'intensité de l'onde réfléchie.

Tout d'abord, la rugosité de la surface, les surfaces lisses provoquent une “réflexion spéculaire”, c'est-à-dire que les ondes sonores sont réfléchies dans une seule direction. En revanche, les surfaces rugueuses dispersent les ondes sonores dans de nombreuses directions, un phénomène connu sous le nom de “réflexion diffuse”.

Ensuite, l'angle d'incidence, l'efficacité de la réflexion varie en fonction de l'angle sous lequel l'onde sonore frappe la surface de l'objet. L'étude de Shagapov et Sarapulova [2015] soutient l'idée que l'efficacité de la réflexion des ondes sonores varie avec l'angle d'incidence, comme observé dans les systèmes sonar, où certains angles maximisent l'intensité du signal réfléchi.

Pour terminer, les propriétés des matériaux affectent également la réflexion. L'impédance acoustique, permet de mesurer la résistance d'un milieu à la propagation d'une onde sonore. Plus la différence d'impédance acoustique entre les milieux est importante, plus la réflexion est forte.

La réflexion est une composante importante dans l'imagerie sonar, car elle permet de détecter et de caractériser les objets. Les ondes réfléchies transmettent au récepteur du sonar des informations sur la taille, la forme et la composition de l'objet.

1.2.3.2 Réfraction des ondes

La réfraction est la déviation des ondes sonores lorsqu'elles passent d'un milieu à un autre dans lequel la vitesse du son est différente. La loi de Snell-Descartes décrivant ce phénomène est donnée par :

$$\eta_1 \sin(\theta_1) = \eta_2 \sin(\theta_2) \quad (1.9)$$

où η_1, η_2 sont les indices de réfraction des milieux (liés à la vitesse du son dans chaque milieu), et θ_1, θ_2 les angles d'incidence et de réfraction, respectivement. Il est possible de compléter le schéma 1.7 en ajoutant l'angle de réfraction θ_{ra} (voir figure 1.8).

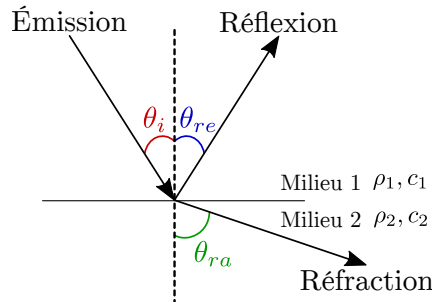


FIGURE 1.8 – Schéma des deux phénomènes : réflexion et réfraction entre deux milieux.

Dans le milieu sous-marin, la réfraction est influencée par les variations de la vitesse du son et donc par les gradients de températures. Une augmentation de la vitesse du son tend à réfracter

l'onde vers l'horizontale, modifiant ainsi sa trajectoire et, par conséquent, les performances du sonar. Un rayon acoustique est une onde caractérisée par sa trajectoire entre deux points, ce phénomène de réfraction est schématisé par le rayon de droite dans la figure 1.9.

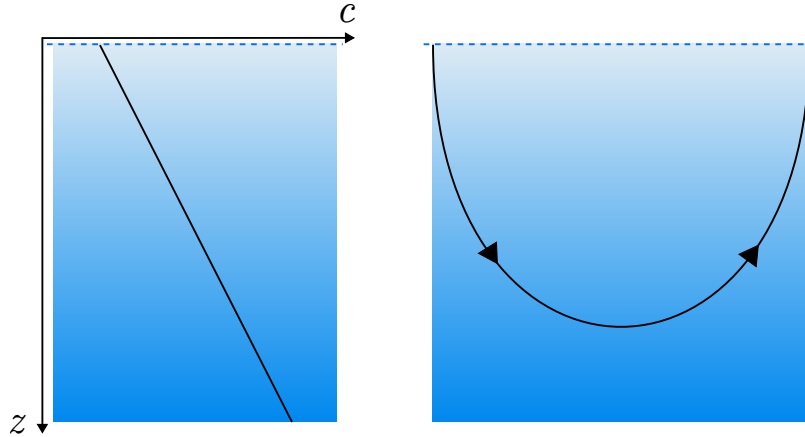


FIGURE 1.9 – Illustration du trajet de la réfraction d'un rayon acoustique par un changement de célérité continu.

Dans le fonctionnement des systèmes sonar, il faut tenir compte de la réfraction, car la modification du trajet des ondes affecte la position et la forme perçue des objets. Ce changement de trajectoire a également un impact direct sur la distance parcourue par les ondes, et donc sur la portée de détection des objets.

1.2.3.3 Rétrodiffusion des ondes d'une cible

L'un des intérêts de l'utilisation des sonars est la détection de cible. Ces cibles peuvent être sous différentes formes comme un banc de poissons pour la pêche, la détection d'épaves pour des raisons écologiques et historiques, ou encore la détection de mines dans le cadre de la guerre de mines.

La rétrodiffusion, c'est la redirection de l'énergie acoustique incidente vers la source en raison de la réflexion sur la cible. Cette section se concentre sur les bases théoriques et les formules mathématiques qui décrivent le phénomène de rétrodiffusion.

Lors de l'émission d'une onde sonar dans l'eau, une partie du son est réfléchi par les objets, c'est ce qui est appelé "l'écho" de la cible. La puissance d'une cible (en décibels) est définie par le rapport des intensités :

$$T_S = 10 \log \frac{I_r}{I_i} \Big|_{r=1\text{m}} \quad (1.10)$$

avec I_i l'intensité incidente, I_r l'intensité réfléchie sur la cible située dans un rayon r d'un mètre de la source.

De manière générale, si toute l'énergie est retransmise par la cible, les intensités incidente et réfléchie sont égales. On obtient alors :

$$I_i \sigma = 4\pi r^2 I_r \quad (1.11)$$

où σ est la section efficace de rétrodiffusion. Cette section représente le degré de dispersion du son sur la cible et dépend de la taille, de la forme et de la réflectivité de cette dernière.

Le ratio entre les intensités est alors donné par :

$$\frac{I_r}{I_i} = \frac{\sigma}{4\pi r^2} \quad (1.12)$$

D'où

$$T_S = 10 \log \left[\frac{\sigma}{4\pi} \right] \quad \text{où } r = 1\text{m} \quad (1.13)$$

Il y a plusieurs facteurs pouvant influencer l'intensité réfléchiée I_r , notamment les caractéristiques physiques de la cible ainsi que celles du signal comme l'angle et la fréquence. La puissance d'une cible dépend de sa forme (convexe, plane...) et de son type (rigide, liquide...), il existe donc différentes équations explicitées dans Hui et Sheng [2022]. Le calcul de la rétrodiffusion par des objets complexes est réalisé par la méthode de l'optique physique (OP), connue sous le nom d'approximation vectorielle de Kirchhoff, et abordée plus en détails dans l'article Klement *et al.* [1988].

1.2.3.4 Perte de réflexion du fond marin

Dans la section 1.2.2.2, nous avons abordé les pertes par amortissement dues à la propagation de l'onde dans l'eau, mais il y a aussi une perte par amortissement due au fond. Lorsqu'une onde sonore frappe le fond marin, une partie de son énergie est réfléchiée, tandis que le reste est absorbé ou transmis dans le fond marin. Cette perte a un impact, car elle détermine la force du signal renvoyé au récepteur. La perte dû à la réflexion peut être modélisée en utilisant le coefficient de réflexion R , qui dépend de l'impédance acoustique de l'eau (Z_e) et du fond marin (Z_f) :

$$R = \left(\frac{Z_f - Z_e}{Z_f + Z_e} \right) \quad (1.14)$$

L'impédance acoustique d'un milieu est définie comme le produit de sa densité (ρ) et de la vitesse du son dans celui-ci (c), c'est-à-dire $Z = \rho c$. Cette équation souligne l'importance des propriétés matérielles du fond marin dans la détermination de la perte de réflexion.

Par exemple, plus le fond marin est dur (comme de la roche), plus l'impédance acoustique du fond (Z_f) est élevée par rapport à celle de l'eau (Z_e). Cela entraîne un coefficient de réflexion R plus proche de 1, signifiant qu'une grande partie de l'énergie sonore sera réfléchiée vers le récepteur. En revanche, si le fond est plus mou ou sédimentaire (comme du sable ou de la vase), l'impédance acoustique du fond sera plus proche de celle de l'eau, ce qui réduira le coefficient de réflexion. Une plus grande part de l'énergie sera alors absorbée ou transmise dans le fond, diminuant la force du signal réfléchi.

Nous venons d'étudier une onde acoustique, sa propagation et ses interactions au sein du milieu marin. Plusieurs types de sonar utilisent ces ondes pour effectuer différentes tâches, c'est ce qui va être abordé dans les prochaines parties.

1.3 Les différents sonars

Bien que les ondes électromagnétiques aient été utilisées dans la communication sous-marine comme détaillé par Yusof et Kabir [2012] et aient été utilisées dans les technologies radar comme le montre Paduan et Washburn [2013], cette partie se concentrera sur les ondes acoustiques. Cette partie commencera par traiter des sonars à balayage latéral, l'une des technologies acoustiques les plus connues. Elle poursuivra ensuite par examiner les capacités et les avancées des sonars multi-faisceaux. Enfin, la discussion se conclura par l'étude des sonars à ouverture synthétique.

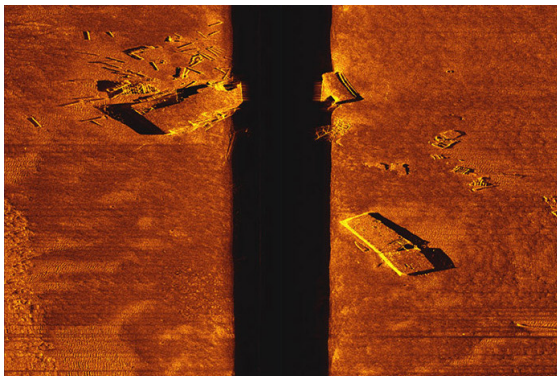
1.3.1 Sondeur latéral

La technologie du sonar à balayage latéral, ou *Side-Scan Sonar (SSS)*, est une forme spécialisée de sonar utilisée dans divers domaines comme l'archéologie marine [Singh *et al.*, 2000], la géologie [Johnson et Helferty, 1990] et l'ingénierie sous-marine [Burguera et Oliver, 2016] grâce à sa capacité à produire des images détaillées pour explorer et surveiller l'environnement sous-marin.

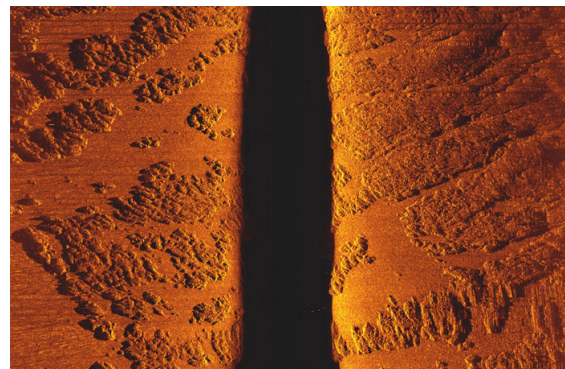
Les systèmes de sonar à balayage latéral utilisent un dispositif sonar remorqué derrière un navire ou un sous-marin, il est appelé "poisson remorqué" (cf. partie C de la figure 1.1). Ce dispositif émet des impulsions en forme d'éventail perpendiculaires à la trajectoire du navire. L'intensité du signal réfléchi du fond marin est enregistrée dans une série de tranches transversales. Lorsque ces tranches sont assemblées dans la direction du mouvement du faisceau, elles créent une image du fond marin dans la zone couverte par le faisceau.

La qualité des images produites par le sonar à balayage latéral est influencée par plusieurs facteurs [Liu *et al.*, 2023], y compris l'altitude du dispositif au-dessus du fond marin, la vitesse du navire d'étude, et la fréquence des ondes sonores. Généralement, les fréquences sonores utilisées sont entre 100 et 500 kHz. L'angle et la force de l'impulsion sonar affectent également la qualité de l'image, avec des impulsions plus fortes fournissant des images plus claires à de plus grandes profondeurs.

La figure 1.10 montre deux exemples d'images pouvant être obtenues grâce à un sonar à balayage latéral.



(a) Barges et débris



(b) Coraux

FIGURE 1.10 – Deux images obtenues avec un sondeur latéral. (Source : Edge Tech)

Le sonar à balayage latéral est une technologie qui a servi à visualiser en détails et mieux interpréter les fonds sous-marins, mais l'évolution a conduit au développement de systèmes sonar à balayage latéral multifaisceaux. Ces systèmes offrent une résolution et une couverture encore plus grande, ils vont être approfondis par la suite.

1.3.2 Sondeur multifaisceaux

Contrairement au sonar latéral, qui reposait sur un seul faisceau pour scanner le fond marin, le sonar à balayage latéral multifaisceaux utilise plusieurs faisceaux émis simultanément, augmentant l'efficacité et le détail de l'imagerie du fond.

Les systèmes de sonar à balayage latéral multifaisceaux utilisent un réseau de faisceaux répartis sur une large bande perpendiculaire à la trajectoire du navire. Cela augmente considérablement l'efficacité de l'acquisition de données, permettant de cartographier de vastes zones

du fond marin en moins de temps, tout en améliorant la résolution. Cela est possible grâce au chevauchement de la couverture des faisceaux, réduisant les lacunes dans les données. Voici une illustration de l'acquisition de données pour un sonar multifaisceaux.

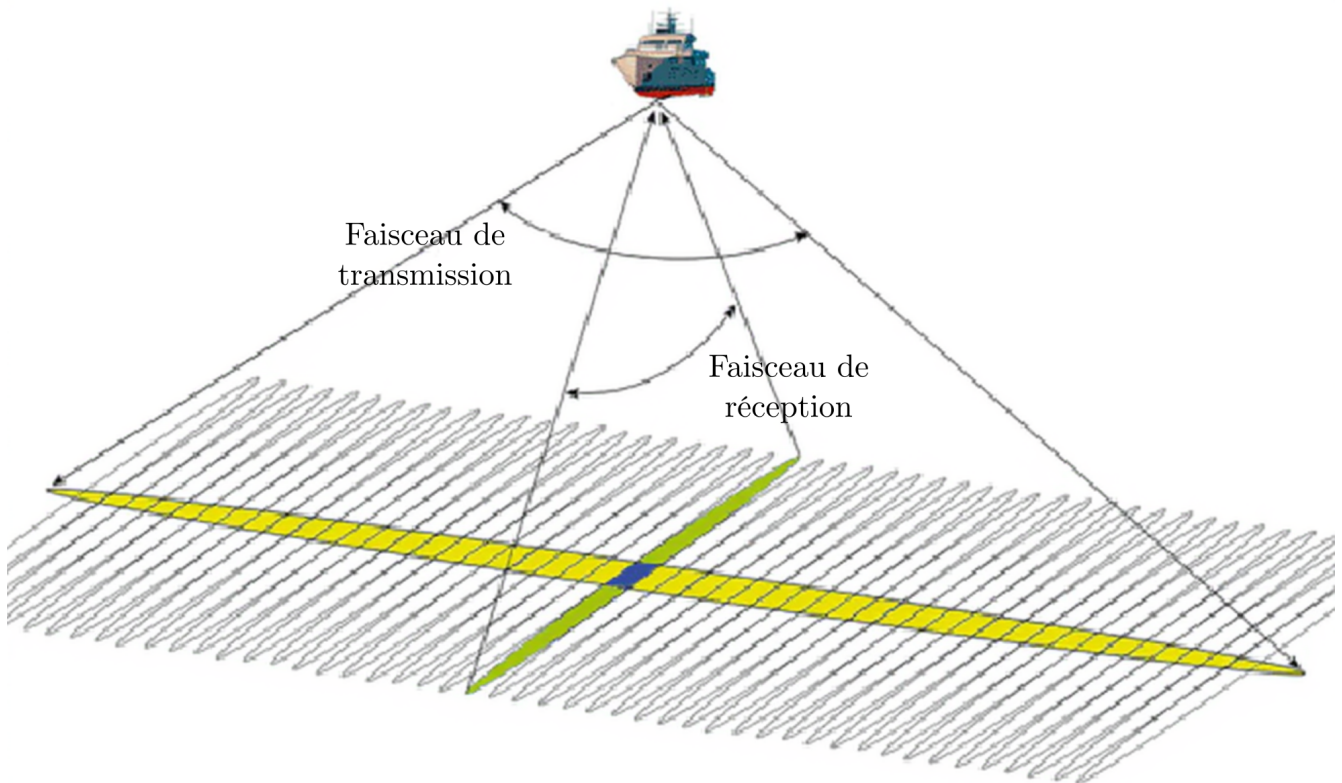


FIGURE 1.11 – Acquisition des données d'un sonar multifaisceau. (Extrait de Bikonis et Demkowicz [2018])

Le sonar à balayage latéral classique produit des images bidimensionnelles, tandis que les systèmes multifaisceaux étendent cette capacité en offrant des données bathymétriques tridimensionnelles. La bathymétrie est l'équivalent sous-marin de la topographie, c'est la mesure de la profondeur de l'eau dans les fonds marins. Ces informations de profondeur permettent de comprendre la structure du fond marin, y compris l'identification de caractéristiques telles que les canaux sous-marins, les monticules et les dangers potentiels pour la navigation.

Les systèmes multifaisceaux ajustent l'angle et la fréquence du faisceau en fonction de la profondeur étudiée, permettant une cartographie flexible et optimisée en fonction de la diversité des fonds. Cette adaptabilité améliore le détail et la précision des images, un exemple de résultat est montré dans la figure 1.12.

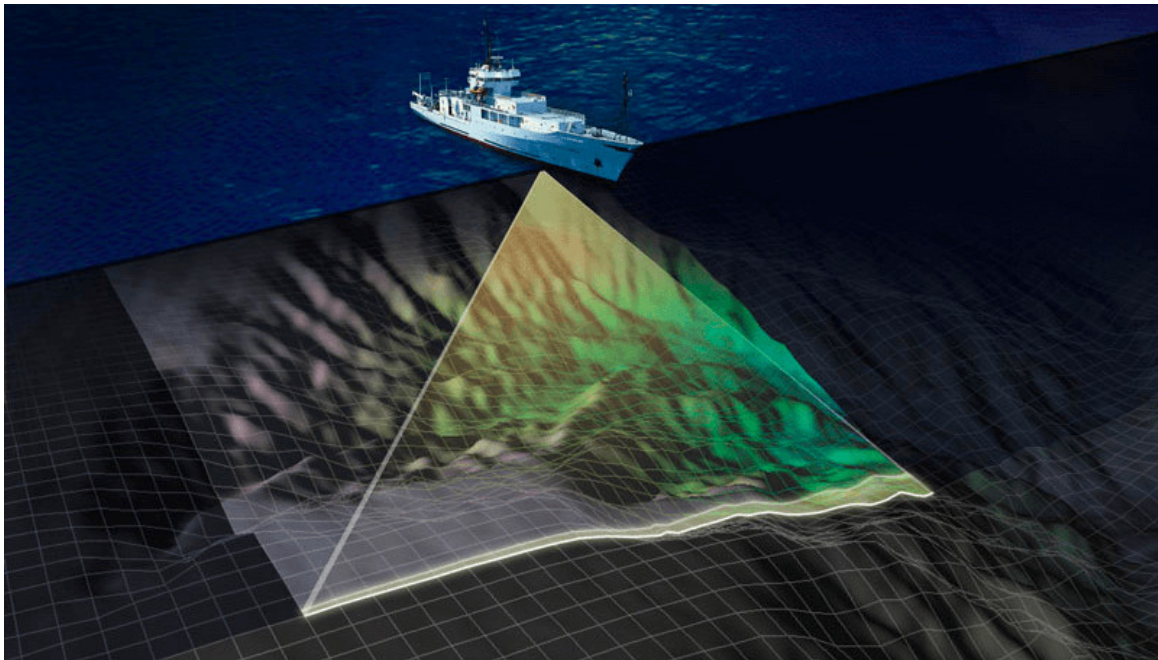


FIGURE 1.12 – Cartographie 3D du fond marin grâce à un sonar multifaisceaux. (Source : RWMT)

L'émission de plusieurs faisceaux implique le traitement de plus de données que les systèmes classiques. Il est nécessaire de corriger la réfraction [Bu *et al.*, 2021] que subissent les ondes en traversant les couches d'eau ayant des densités différentes. Sans cette correction, les variations de célérité et la courbure des trajectoires seraient mal prises en compte par le système, entraînant des erreurs sur la localisation du point de réflexion ainsi que sur l'estimation de la distance de propagation.

L'utilisation du multifaisceaux dans le système sonar a amélioré la cartographie des fonds marins grâce à sa couverture et une résolution accrue, ainsi que ses données bathymétriques.

Cependant, il existe d'autres systèmes, par exemple le Sonar à Ouverture Synthétique, appelé *Synthetic Aperture Sonar (SAS)*, qui utilise le mouvement de la plateforme sonar pour synthétiser une ouverture plus longue. C'est ce que nous allons aborder dans la prochaine partie.

1.3.3 Sonar à ouverture synthétique (SAS)

1.3.3.1 Principe

L'introduction du SAS par Hansen [2011] dans les systèmes sonar repose sur l'utilisation du mouvement de la plateforme pour simuler une antenne ou une ouverture beaucoup plus grande par rapport à ce qui est physiquement possible.

Cette technique permet au SAS d'obtenir des images avec une haute résolution sur de larges étendues. Le principe est de continuer le déplacement tout en illuminant la même zone sur le fond avec plusieurs signaux. Lors du traitement, grâce à la réorganisation cohérente des données de tous les signaux, une image à "ouverture synthétique" est produite avec une résolution indépendante de la portée. En comparaison avec le sonar latéral classique, le SAS peut avoir une meilleure résolution de plusieurs ordres de grandeur.

La figure 1.13 montre la différence entre les deux types de sonar lors de l'acquisition des données d'une zone.

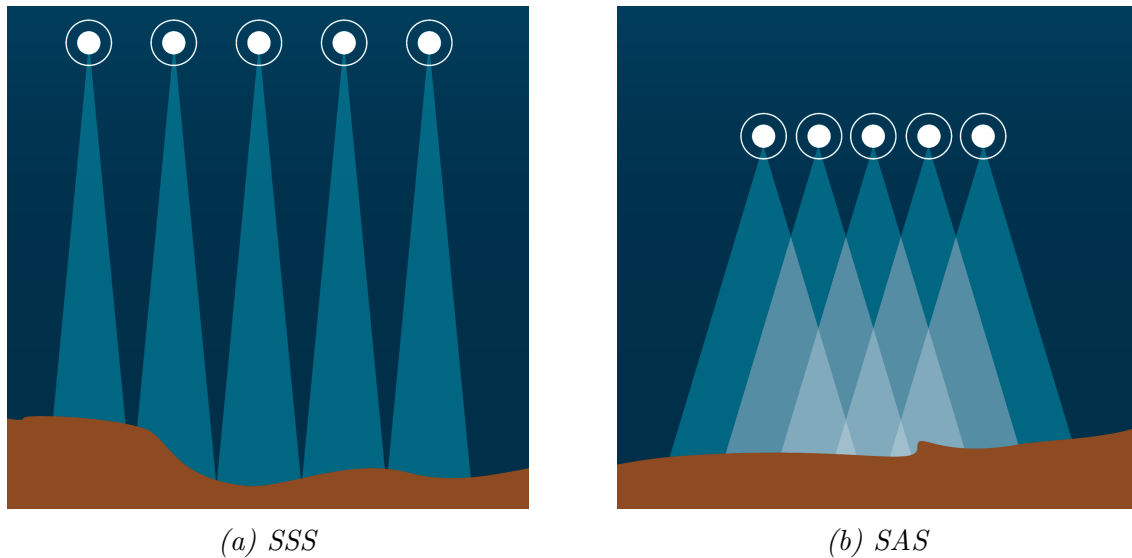


FIGURE 1.13 – Illustration des différences lors de l'acquisition entre les sonars latéraux classiques 1.13a et les SAS 1.13b lors de l'acquisition. (Source : NOAA)

1.3.3.2 Traitement des données

Le traitement des données SAS est une opération complexe transformant les retours des ondes bruts en images haute résolution. Cela implique plusieurs étapes : conditionnement du signal, correction de phase et synthèse d'image, chacune permettant de compenser les distorsions dues à l'environnement sous-marin.

Les algorithmes utilisés pour aligner et combiner les échos sont basés sur les informations de phase. Le résultat est une image claire du fond marin très détaillée, dont certains détails n'étaient pas forcément détectés par les systèmes sonar classiques.

Ces images sont souvent utilisées pour la détection, la localisation et la classification d'objets sur le fond marin. Cependant, les données obtenues grâce à un SAS ont un bruit appelé le “*speckle*”. Il existe des algorithmes de traitement d'images [Chanussot *et al.*, 2002] pour réduire le bruit et supprimer les réflexions parasites. La figure 1.14 illustre une méthode proposée par Firoiu *et al.* [2011] pour retirer le “*speckle*”.

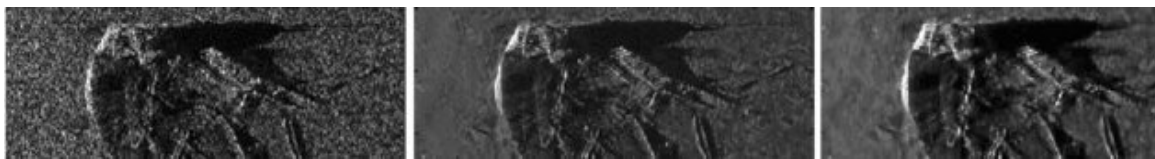


FIGURE 1.14 – Illustration d'un algorithme pour enlever le bruit d'une image sonar en entrée (à gauche) et les résultats avec une méthode à différents niveaux (au milieu et à droite). (Extrait de Firoiu *et al.* [2011])

En résumé, ce chapitre a présenté les principes de l'acoustique sous-marine ainsi que les principaux systèmes sonar pour l'imagerie du milieu marin. Ce chapitre permet de comprendre les données utilisées dans cette thèse, leurs spécificités et leurs limites.

Le chapitre suivant décrira plus en détail les bases de données utilisées, en distinguant la simulation de données et les acquisitions réelles, ainsi que les traitements appliqués pour les rendre exploitables dans un cadre de détection automatique.

Chapitre 2

Présentation des bases de données sonar

Sommaire

2.1	Données simulées (base SIMSON)	26
2.2	Données réelles	29
2.3	Construction des ensembles d'apprentissage	30
2.3.1	Modifications des données	30
2.3.1.1	Développement des images simulées	31
2.3.1.2	Suppression de la colonne d'eau	32
2.3.2	Structuration des bases	35

Ce chapitre présente les bases de données utilisées dans cette thèse. Deux ensembles distincts sont utilisés : une base simulée, abondante et contrôlée, et une base réelle, plus restreinte mais représentative des conditions opérationnelles.

Leur constitution et leurs principales caractéristiques sont décrites, ainsi que les traitements appliqués pour les rendre exploitables et leur structuration en ensembles d'apprentissage.

2.1 Données simulées (base SIMSON)

Au cours d'un projet distinct à la thèse, la Direction Générale de l'Armement (DGA) a confié à SCALIAN la tâche de développer un simulateur d'images sonar. Le but étant de générer une base de données d'images simulées, baptisée SIMulation d'image SONar (SIMSON), qui servirait à l'entraînement et à l'évaluation de modèles d'apprentissage automatique pour la détection d'objets sous-marins. SCALIAN a alors structuré son développement en plusieurs étapes distinctes.

La première étape consiste à générer un environnement 3D complet, dans lequel sont insérés des objets d'intérêt. Les simulations intègrent une diversité de cibles, réparties en cinq types : manta, BRM, rockan, sphère et sphère corne. Chaque type de cible a ses particularités, comme le type "BRM" ayant une forme plus rectiligne et allongée contrairement au type "manta" étant plus pointu. Les orientations locales des objets varient selon plusieurs paramètres : un taux d'enfouissement compris entre 5% et 40%, une rotation autour de l'axe x comprise entre -10° et 10° , suivie d'une rotation autour de l'axe z de 0° à 360° . Chaque objet est répliqué trois fois dans une même scène afin de simuler un environnement plus encombré, facilitant ainsi l'évaluation de la robustesse des méthodes d'analyse, indépendamment de la variabilité entre objets.

Les images sont générées en suivant un ordre de superposition précis : la cible est insérée en premier, suivie des éléments rocheux, puis du fond marin. Les types de fonds simulés sont variés et comprennent des environnements rocheux, du gravier, du sable, des rides de sable et des sédiments.

La figure 2.1 ci-après illustre les différents types d'objets intégrés dans les simulations, ainsi que les fonds marins utilisés. Elle permet de visualiser la diversité des formes et textures des cibles synthétiques, ainsi que la complexité des environnements dans lesquels elles sont insérées.

Cette diversité de configurations permet de tester et d'évaluer les performances des algorithmes de détection dans des conditions à la fois variées et représentatives des scénarios réels.

La deuxième étape consiste à simuler une acquisition sonar, en reproduisant les caractéristiques d'un système de type Klein 5000 faisant partie de la classe des SAS. Chaque cible est visualisée depuis deux altitudes fixes du sonar, à savoir 10 mètres et 20 mètres. Les distances entre le sonar et la cible sont variables, les cibles peuvent apparaître soit en début, milieu ou en fin de portée. Ces positions sont générées aléatoirement selon une distribution normale centrée autour des positions souhaitées, de manière à éviter un alignement artificiel des objets dans les images. Par ailleurs, chaque cible est approchée sous quatre angles d'incidence différents : 0° , 45° , 90° et 135° , afin d'augmenter la variabilité des perspectives et enrichir la diversité des représentations générées.

Pour finaliser le processus et renforcer le réalisme des images simulées, un post-traitement est appliqué afin d'y introduire différents types de bruit. Ce traitement prend en compte le

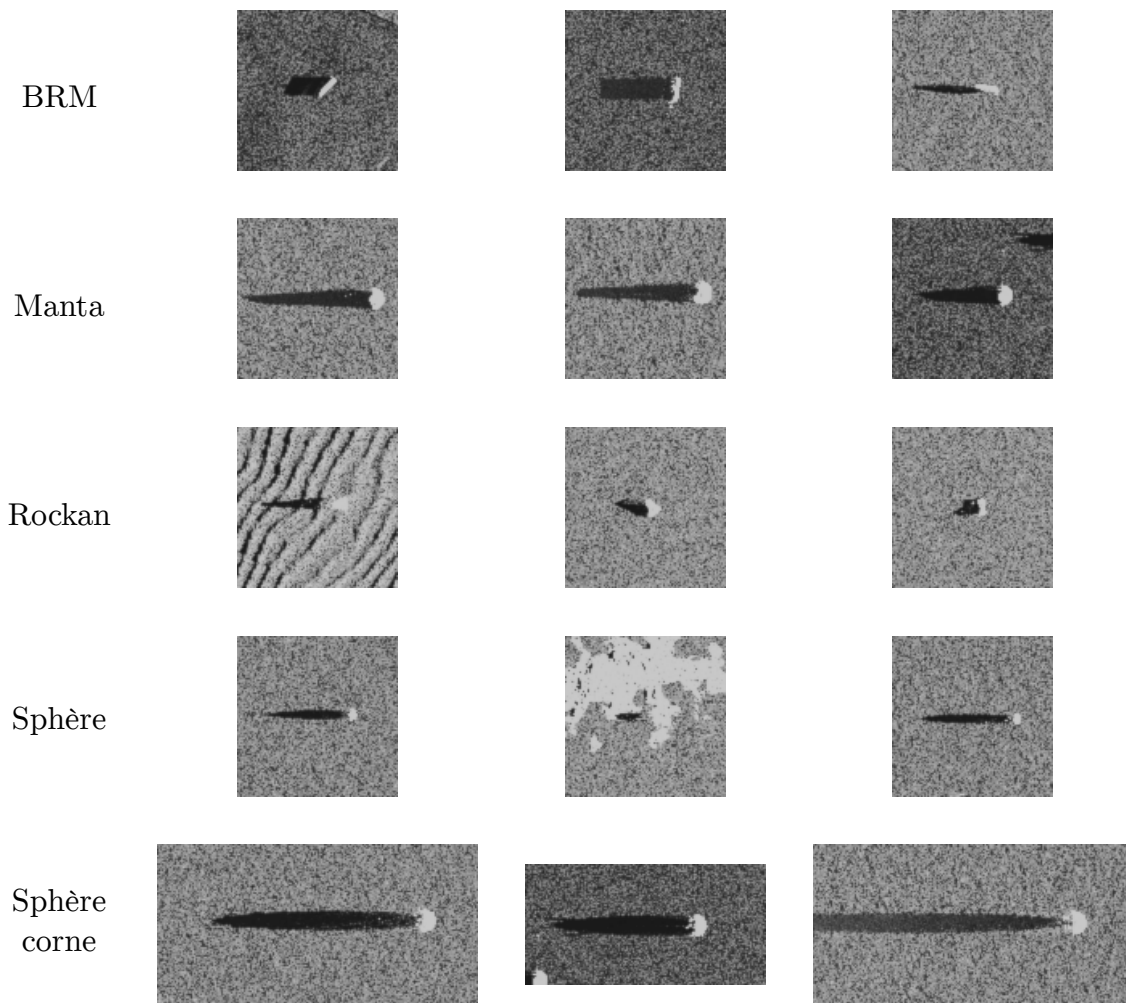


FIGURE 2.1 – Visuel des différents types d'objets sur divers fonds.

bruit thermique, les pertes dues à la transmission dans le milieu marin, ainsi que l'effet des diagrammes d'antenne en émission et en réception.

Afin de simplifier les calculs liés à la propagation acoustique, la célérité du son est supposée constante tout au long de la simulation.

Le simulateur prend en compte de nombreux paramètres, qui sont tous répertoriés dans le tableau 2.1.

TABLE 2.1 – Paramétrage du sonar Klein 5000 pour la génération des images de la base

Paramètre	Valeur
Fréquence	455 kHz
Largeur de bande	20 kHz
Orientation1	-90°
Orientation2	90°
Incidence	10°
Longueur d'antenne Rx/Tx	120 cm
Hauteur d'antenne Rx/Tx	8 cm
Largeur faisceau Rx/Tx	0,076°
Hauteur faisceau Rx/Tx	40°
Directivité Mode Horizontal Rx/Tx	Harris
Directivité Mode Vertical Rx/Tx	
Niveau de source	180 dB
Bruit thermique	75 dB
Portée min	0 m
Portée max	15 m
Résolution across track	10 cm
Résolution along track	10 cm
Vitesse	1,558 m/s

Certaines images simulées ont été conçues pour reproduire au mieux les environnements observés lors d'une campagne d'acquisition réelle menée avec le système sonar Klein 5000, dans le but d'obtenir des rendus visuels aussi réalistes que possible. Pour cela, la DGA a mis à disposition un ensemble d'images issues de campagnes réelles, servant de référence à SCALIAN dans le développement d'un simulateur capable de générer des images fidèles aux conditions attendues sur le terrain.

Afin de faciliter l'organisation et la catégorisation des images simulées, celles-ci sont réparties dans plusieurs sous-dossiers, chacun correspondant à un niveau ou type de difficulté particulier. Ces catégories incluent : la simulation de la campagne réelle, les environnements de récifs rocheux, les plaines de roches, les fonds sédimentaires, ainsi qu'un ensemble mixte combinant plusieurs de ces caractéristiques.

Le tableau 2.2 synthétise les principaux paramètres expliquant la constitution de la base, qui compte au total 22176 images, ainsi que la répartition des différentes configurations qui la composent.

TABLE 2.2 – Tableau des paramètres croisés pour la génération de la base .

	Description du paramètre	Valeur	Commentaire
Sonar	Klein 5000	1	type de sonar
Profil célérité	Célérité constante	1	
Orientations locales	Distance d'enfouissement à partir d'un tirage aléatoire entre 5% et 40% de la taille de la cible ; Angle de rotation autour de l'axe x à partir d'un tirage aléatoire entre -10° et 10° ; Angle de rotation autour de l'axe z à partir d'un tirage aléatoire entre 0° et 360° .	77	orientations locales de la cible
Altitudes porteur	10m, 20m	2	altitudes du porteur
Positions relatives cible	En début, milieu et fin de portée	3	distances relatives
Trajectoires	Trajectoire avec rotation du cap à 0° , 45° , 90° et 135°	4	vues différentes du fond
Dimensions	$\sim 1500 \times 5000$	1	largeur $\times$ hauteur
Combinatoires / fond	Nombre de combinatoires tout paramètres confondus pour un type de fond	1848	combinatoires (= simulations) par type de fond
Itération de cible	Nombre de fois où la même cible est présente dans le rail	3	itérations d'une cible par rail
Configurations cible/fond		5544	vues différentes d'une cible sur un type de fond
Terrains		6	fonds homogènes et hétérogènes
Simulation		11088	nombre de simulations
Rails	babord + tribord	22176	nombre de rails (= images)

La base comprend ainsi un ensemble d'images simulées. Toutefois, ces données ne constituent pas l'unique source de données utilisée dans le cadre de cette thèse, la DGA a également fourni des images issues de réelles campagnes d'acquisition.

2.2 Données réelles

Pour des raisons de confidentialité, aucune information n'a été communiquée par la DGA sur le système utilisé pour l'acquisition, ni sur les paramètres associés ainsi que sur les traitements

effectués sur les données. Il est néanmoins évident que les images sont totalement différentes de celles simulées provenant de la base .

Elles se distinguent notamment par la résolution des images et le bruit qui les composent. Malgré de nombreux post-traitements, les images simulées ont un aspect “parfait” rendant leur lecture plus facile que celles issues d’une acquisition réelle. Les résolutions dans la longueur et dans la largeur du rail sont différentes, tout comme les dimensions globales des images, qui sont d’environ 3000×15000 , soit six fois plus grandes en termes de pixels que les images simulées.

Le nombre total d’images est également très différent de celui des images simulées. La base fournie par la DGA est composée de seulement 432 images, avec les vérités terrain associées. La vérité terrain est donnée sous forme de coordonnées avec le type d’objet, pour chaque image ayant au moins un objet présent.

La figure 2.2 présente des exemples d’images illustrant la difficulté à distinguer, à l’œil nu, les rochers des objets d’intérêt.



FIGURE 2.2 – Exemples d’images provenant de la base réelle avec des objets d’intérêts ainsi que des rochers.

Même si les données réelles sont différentes des images simulées, plus difficiles à traiter et en quantité limitée, elles représentent les conditions réelles et serviront de référence pour évaluer les performances des modèles ainsi que la validation de la fiabilité des approches développées.

2.3 Construction des ensembles d’apprentissage

Cette section présente les processus mis en place pour la création d’ensembles d’apprentissage et de test qui seront utilisées pour les expériences développées dans la partie 5.

2.3.1 Modifications des données

Les données réelles utilisées ont été traitées en amont par la DGA, selon un protocole de développement¹ propriétaire. Ce processus n’étant pas accessible ni modifiable, il n’est pas possible d’intervenir sur les étapes de développement des images sonar.

En revanche, les données simulées, générées indépendamment, offrent une liberté totale quant à leur développement. Les modifications apportées au développement des données simulées vont être abordées dans la prochaine partie et vise à assurer une compatibilité totale avec les algorithmes de traitement d’images qui seront utilisés.

1. Dans le domaine de l’imagerie sonar, le terme *développement* désigne l’ensemble des traitements appliqués aux données brutes pour les transformer en images exploitables. Ces traitements incluent notamment le décodage des signaux, leur mise à l’échelle, ainsi que divers post-traitements visant à améliorer la lisibilité des données.

2.3.1.1 Développement des images simulées

Les données RAW issues des images simulées sont sous forme de nombres complexes, capturant l'amplitude et la phase des échos reçus.

Le calcul du module permet de transformer un signal complexe en une mesure d'intensité, qui est une valeur scalaire positive, servant à la construction et visualisation d'images sonar.

Pour un nombre complexe z , ce module est défini par :

$$\sqrt{\text{Re}(z)^2 + \text{Im}(z)^2} \quad (2.1)$$

avec $\text{Re}(\cdot)$ étant la partie réelle du nombre complexe et $\text{Im}(\cdot)$ la partie imaginaire.

Cependant, les données brutes contiennent les images bâbord et tribord de chaque acquisition, puisque le sonar effectue le lancer d'ondes de part et d'autre. La figure 2.3 donne un exemple de données simulées avant la séparation.

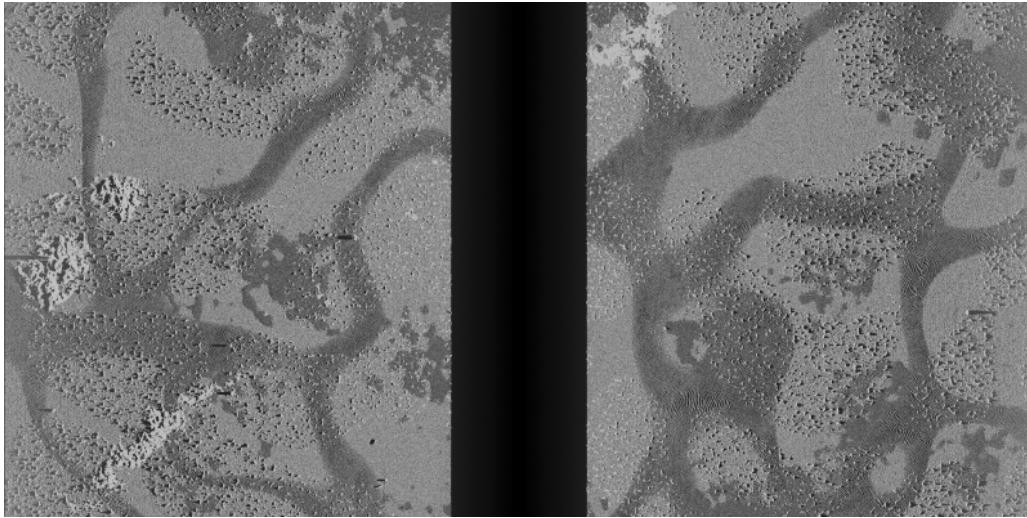


FIGURE 2.3 – Exemple d'une partie de données simulées sans la séparation des images bâbord / tribord. L'image a subi un traitement manuel afin de la visualiser.

L'image est séparée en deux images. En divisant la largeur de l'image en deux pour obtenir le milieu de l'image. Le résultat de cette séparation donne deux images, l'image bâbord (appelée "port") à gauche et l'image tribord (appelée "stbd", pour *starboard* en anglais) à droite.

Les signaux sonar présentent souvent une large dynamique, allant d'échos très faibles à d'autres beaucoup plus puissants. Représenter cette plage sur une échelle linéaire impliquerait de manipuler simultanément des valeurs très petites et très grandes, ce qui complique la visualisation et l'analyse.

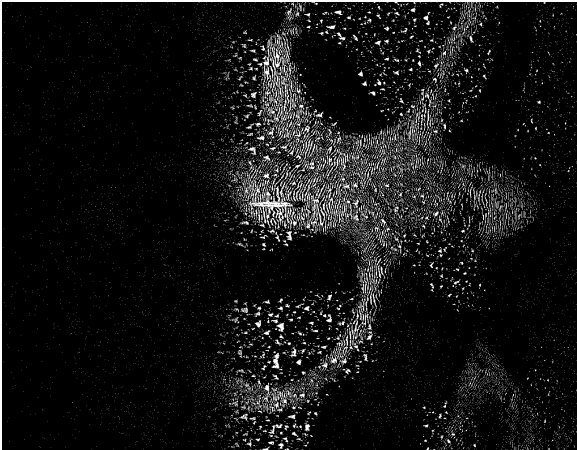
L'utilisation d'une échelle logarithmique, comme celle des décibels, permet de comprimer cette plage dynamique en un intervalle plus restreint. Cette transformation, généralement suivie d'une quantification, facilite la représentation numérique et visuelle des données.

Le module est exprimé en décibels (dB) grâce à l'équation suivante :

$$f(x) = 20 \log_{10} (data) \quad (2.2)$$

où $data$ représente les valeurs brutes des données d'origine et f la fonction permettant le passage en décibels.

La figure 2.4 est une illustration du passage à une échelle logarithmique en décibels sur une image sonar.



(a) Module des données brutes.



(b) Passage en décibels du module.

FIGURE 2.4 – Différence visuelle lors du passage à l'échelle décibels.

Il est évident que le changement d'échelle permet de mieux répartir l'ensemble des valeurs. Par exemple, il devient plus facile de discerner l'intégralité des rides de sable, notamment dans la partie gauche dans l'image 2.4b que dans l'image 2.4a.

La quantification permet de mettre en évidence les valeurs les plus significatives en renforçant leur contraste, facilitant ainsi leur interprétation et leur exploitation.

Les images simulées (), bien que de taille inférieure aux images de la base réelle, conservent des dimensions suffisamment importantes pour rendre leur traitement difficile dans leurs dimensions d'origine. Toutes les données sont encodées en 64 bits, ce qui représente plus de 175 MB par acquisition (comprenant les images bâbord et tribord). Ce format est volumineux à stocker, rendant nécessaire une normalisation ainsi qu'un passage à 16 bits. Cet encodage est un bon compromis entre la qualité de l'image et le stockage nécessaire. Cette transformation est réalisée à l'aide de l'équation suivante :

$$\text{data}_{new} = \lambda \times \left((\text{data}_{old} - \min_{old}) \times \frac{\max_{new} - \min_{new}}{\max_{old} - \min_{old}} + \min_{new} \right), \quad (2.3)$$

avec $\min_{old} = \min(\text{data}_{old})$, $\max_{old} = \max(\text{data}_{old})$, $\min_{new} = 0$ et $\max_{new} = 2^{16} = 65536$.

Une fois ces étapes effectuées, le développement des images simulées est achevé, les images sont stockées sur 16 bits avec une représentation adéquate des plages de valeurs intéressantes, au même titre que les images réelles.

En complément, une découpe des images a été effectuée pour ne conserver qu'une partie des données au sein de chaque image, notamment en retirant la colonne d'eau, visible au centre de la figure 2.3.

2.3.1.2 Suppression de la colonne d'eau

La colonne d'eau est souvent source de bruit et d'interférences, causées notamment aux bulles d'air, aux particules en suspension et aux variations thermiques. La supprimer permet d'éliminer des éléments perturbateurs pouvant altérer la qualité des données.

En retirant les données de la colonne d'eau, les algorithmes de détection peuvent se concentrer uniquement sur les caractéristiques du fond. Cela permet de réduire le temps et les ressources nécessaires au traitement, tout en se focalisant sur les informations essentielles.

Dans le contexte du traitement des images sonar, la suppression de la colonne d'eau peut améliorer la détection d'objets sur le fond marin en atténuant les interférences et artefacts situés entre le transducteur et le fond. Cette opération permet de focaliser l'analyse sur les zones pertinentes, rendant les objets d'intérêt plus visibles et facilitant leur identification.

Néanmoins, cette suppression n'est pas sans conséquence, puisqu'elle entraîne une perte d'informations relatives à la colonne d'eau, comme la présence de particules en suspension, la turbidité ou encore la faune marine. Ces éléments sont importants pour des analyses environnementales ou biologiques, en particulier lorsqu'il s'agit d'étudier la qualité de l'eau ou les conditions de propagation acoustique.

Cependant, dans le cadre de cette thèse, l'objectif n'est pas d'analyser la colonne d'eau ou les écosystèmes sous-marins, mais bien de détecter des objets posés sur le fond. La perte d'information liée à la colonne d'eau reste donc acceptable, car elle permet d'optimiser la détection des cibles sur le fond marin.

Étant donné que les images réelles et simulées sont différentes, les méthodes utilisées pour retirer la colonne d'eau varient également.

Lors de la simulation des images, selon le tableau 2.2, il existe 1848 combinaisons par type de fond. Ces combinaisons ont été divisées en deux groupes égaux de 924 images, correspondant à la distance d'acquisition. Dans le premier groupe, la colonne d'eau mesure exactement 200 pixels de large, tandis que dans le second groupe, elle mesure 400 pixels.

La suppression de la colonne d'eau dans les images simulées consiste simplement en un recadrage de l'image, supprimant ainsi cette colonne. Ce recadrage tient compte de deux éléments : à quel groupe l'image appartient et s'il s'agit d'une image bâbord ou tribord, ce qui détermine de quel côté se situe la colonne d'eau dans les données.

En ce qui concerne les images réelles, les colonnes d'eau n'ont pas toutes la même largeur, ce qui rend leur suppression plus complexe. Par exemple, la figure 2.5 montre l'image sur laquelle le protocole va être appliqué.

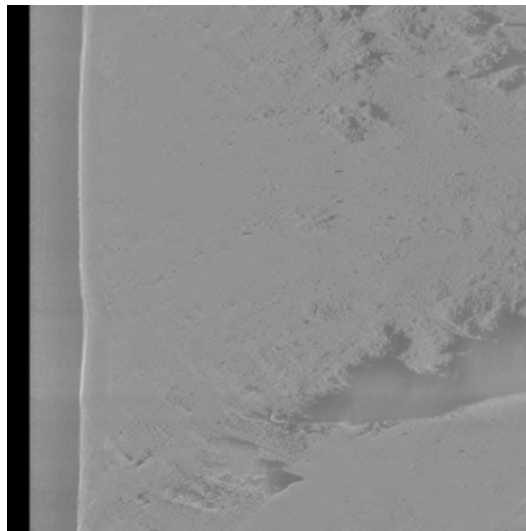


FIGURE 2.5 – Partie d'une image réelle servant d'exemple pour l'illustration du retrait de la colonne d'eau.

L'onde arrivant sous le sonar renvoie un écho très puissant, et c'est cette zone qui est recherchée pour effectuer la découpe. Deux méthodes automatiques ont été testées pour déterminer la meilleure coordonnée afin de supprimer uniquement la colonne d'eau. Ces méthodes sont basées sur l'utilisation d'un gradient et d'une moyenne des valeurs, comme le montrent les courbes dans le graphique de la figure 2.6 ci-dessous.

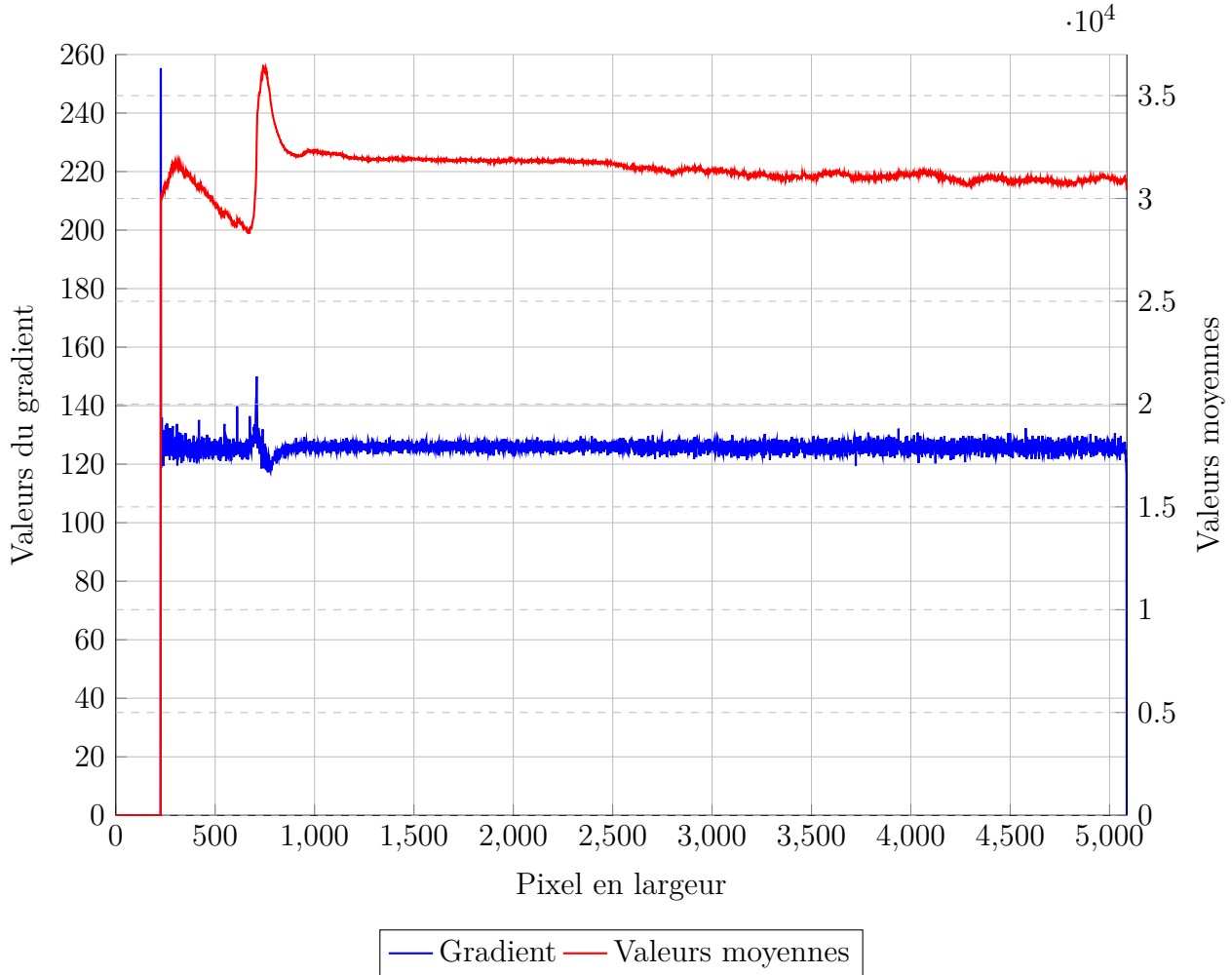


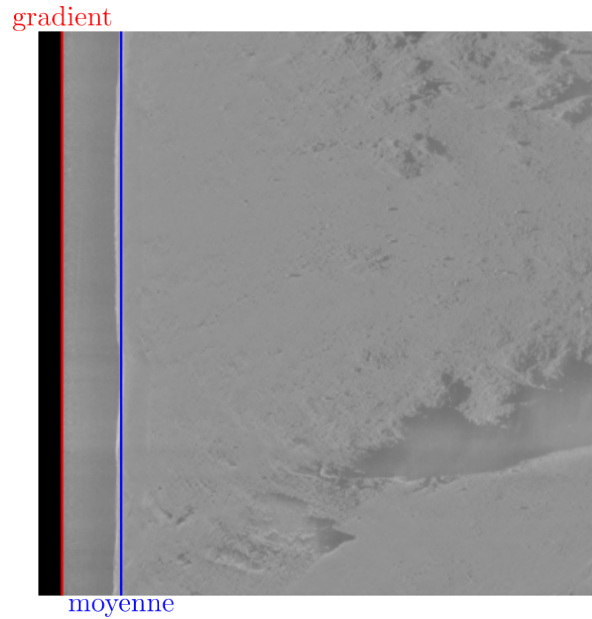
FIGURE 2.6 – Graphique représentant les valeurs du gradient ainsi que les valeurs moyennes en fonction de la largeur de l'image.

Pour automatiser cette sélection, il suffit de prendre le maximum de ces courbes afin d'obtenir la coordonnée permettant de retirer la colonne d'eau. Nous obtenons alors des coordonnées différentes pour chaque méthode : $x_{\text{gradient}} = 226$ et $x_{\text{moyenne}} = 742$, visible sur la figure 2.7a.

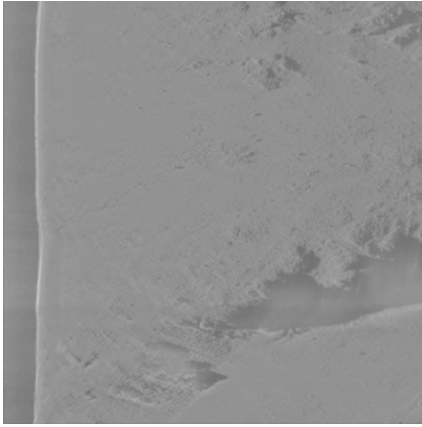
La méthode utilisant un gradient dans l'image vise à identifier la grande variation des valeurs des pixels entre la colonne d'eau (valeurs faibles) et l'écho (valeurs élevées). Le calcul du gradient se fait grâce à la technique "Sobel" de la librairie OpenCV.

La figure 2.7b montre qu'une partie de la colonne d'eau reste à supprimer, cette zone allant jusqu'à la ligne verticale blanche à gauche. Le gradient permet de visualiser cette transition grâce au deuxième pic (arrivant à environ 150), mais il n'est pas certain que cette démarcation corresponde toujours au deuxième pic.

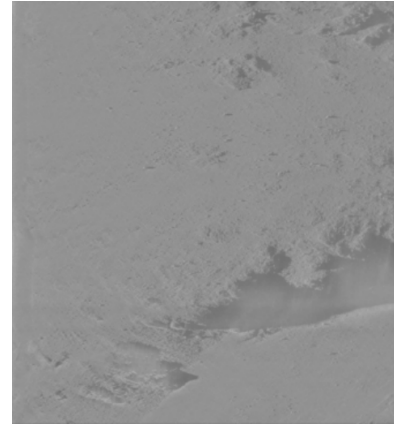
Le deuxième pic obtenu avec la méthode du gradient est proche du pic maximum de la méthode basée sur la moyenne des valeurs de l'image. Le résultat obtenu en découpant l'image avec la méthode basée sur la moyenne est illustré dans la figure 2.7c. Ce résultat est bien plus net et conserve uniquement les données utiles pour la détection des objets.



(a) Localisation des limites de la colonne d'eau pour les différentes méthodes de retrait.



(b) Suppression de la colonne d'eau basée sur le calcul du gradient.



(c) Suppression de la colonne d'eau par la moyenne des valeurs.

FIGURE 2.7 – Méthodes de détection et de suppression de la colonne d'eau dans les images sonar.

Après avoir effectué le développement complet des images et retiré la colonne d'eau, nous obtenons notre base d'images réelles. La dernière étape consiste alors à structurer ces données en différentes bases, qui serviront aux expérimentations menées dans le cadre de cette thèse.

Ainsi, la suppression de la colonne d'eau varie selon la nature des données : un recadrage simple pour les images simulées, où sa largeur est connue, et une détection automatique pour les images réelles, aux caractéristiques plus variables. Cette étape recentre les images sur le fond marin afin de les rendre plus adaptées à la détection d'objets.

La préparation des données se poursuit ensuite par un découpage des images, destiné à optimiser leur traitement et à l'adapter aux contraintes d'apprentissage.

2.3.2 Structuration des bases

Afin de préparer les images pour les algorithmes de détection, les données réelles et simulées ont été découpées en patches de 512×512 pixels. Cette étape réduit la taille des images pour

faciliter l'apprentissage et accroît la proportion relative d'objets, initialement très faible dans les images complètes ($\approx 4.10^{-3}\%$ de l'image).

Les découpes conservent systématiquement les objets présents dans l'image d'origine, tout en incluant également des patches de fond. De légères translations aléatoires sont appliquées afin d'éviter que les objets ne soient systématiquement centrés.

Après ces étapes de préparation, les caractéristiques principales des deux bases sont résumées dans le tableau 2.3.

TABLE 2.3 – Caractéristiques des bases de données réelles et simulées

Base	Total	Objets		% obj.	Réparti- tions	Remarques
		Avec	Sans			
Réelle	9744 (512×512)	864	8880	8.9%	4 dossiers croisés	Découpes groupées par image source
Simulée	66577 (512×512)	28602	37975	42.9%	Mélange complet	

La figure 2.8 illustre les différences entre les deux bases après l'ensemble des étapes de préparation, en présentant quelques exemples d'images simulées et réelles. Les premières se distinguent par un rendu plus “net” et synthétique, tandis que les secondes présentent davantage de bruit et une complexité visuelle accrue.

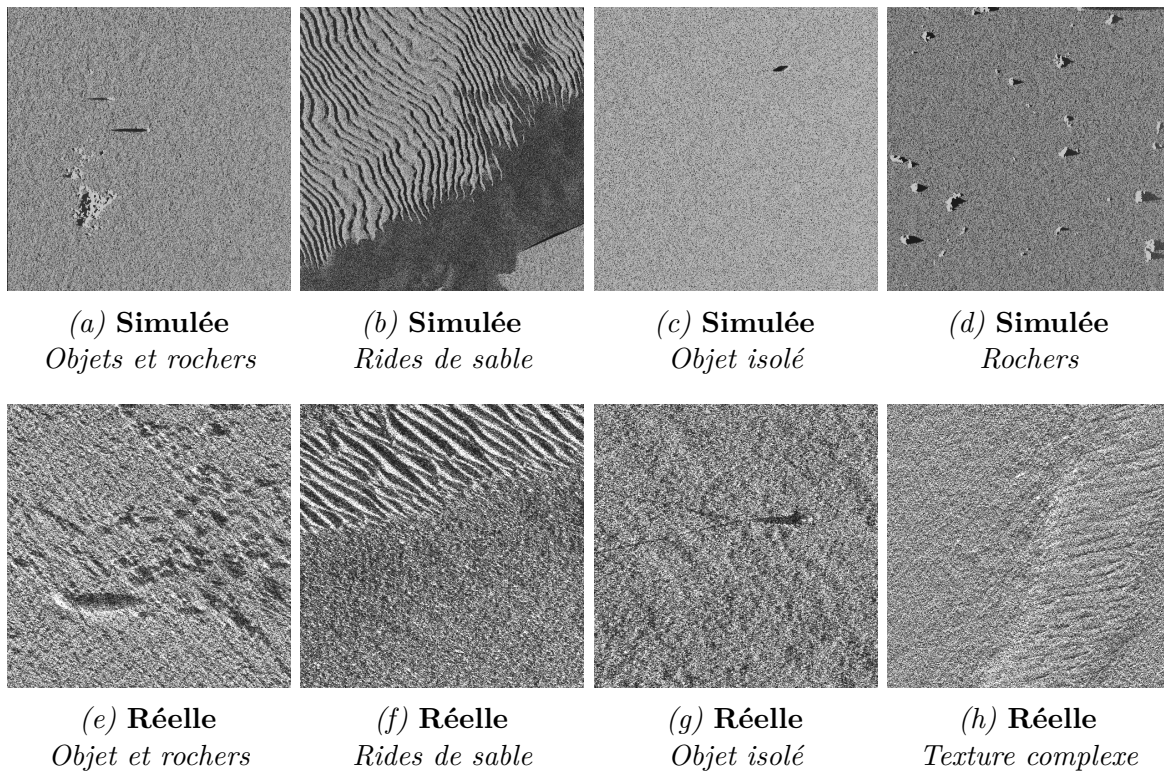


FIGURE 2.8 – Exemples d'images issues des deux bases après préparation. Les patches (a-d) proviennent de la base simulée et (e-h) de la base réelle fournie par la DGA.

Les données réelles étant peu nombreuses, nous avons opté pour un protocole en validation croisée en prenant $k = 4$ dossiers. La répartition des différentes images dans ces dossiers est faite de sorte à diluer le peu d'images contenant des objets parmi les images composées uniquement de fond, qui sont majoritaires.

Chaque dossier de la base réelle contient 2436 images, dont 91,1% sont des images de fond. Les 216 images restantes comportent au moins un objet, ce qui équivaut à un ratio d'environ 1 image avec objet pour 10 images de fond. Lors de la création des bases, pour éviter de biaiser les résultats, les découpes d'images issues d'une même image initiale ont été regroupées dans le même dossier de validation croisée.

Quant à la base simulée, il y a uniquement un dossier où l'ensembles des images avec objet et sans objet sont mélangés. Comme mentionné dans la partie 2.1, il existe cinq types d'objet dans la base simulée. Chaque objet est inséré trois fois par image, faisant un total de quinze objets par image. La proportion d'objets est donc plus élevée que dans les images réelles. La base simulée est composée de 66577 images au total, dont 28602 images contiennent au moins un objet. Une sous-base a été créée pour différents tests, en réservant certaines images pour garantir qu'elles n'ont jamais été utilisées auparavant. Cette sous-base est réduite à 38976 images, dont 16788 avec objet.

Ces images étant la propriété de la DGA et de SCALIAN, les données ne peuvent pas être diffusées publiquement et sont uniquement utilisables dans le cadre de la thèse.

Chapitre 3

Détection d'objets

Sommaire

3.1	Introduction	40
3.2	Principe des détecteurs et RPN (<i>Region Proposal Network</i>)	40
3.2.1	Détection d'objets	40
3.2.2	Approches initiales basées sur la recherche de régions	41
3.2.3	Détecteurs à deux étapes et RPN	42
3.2.3.1	Region-based Convolutional Neural Networks (R-CNN)	42
3.2.3.2	Fast R-CNN	43
3.2.3.3	Faster Region-based Convolutional Neural Networks (Faster R-CNN)	44
3.2.3.4	Region Proposal Networks (RPN)	44
3.3	You Only Look Once (YOLO)	47
3.3.1	Historique et évolutions de YOLO	47
3.3.2	YOLOv5	49
3.3.2.1	Architecture	49
3.3.2.2	Fonctionnement et prédiction	54
3.3.2.3	Fonctions de perte	55
3.4	Détecteur DETR basé sur les ViT	57
3.4.1	Mécanisme d'auto-attention	58
3.4.2	Architecture DETR	58
3.4.3	Fonction de perte et optimisation	59

3.1 Introduction

La détection d'objets est une tâche majeure en vision par ordinateur, visant à localiser et identifier les objets présents dans une image ou une vidéo. Contrairement à la classification d'images, qui attribue une seule étiquette à l'ensemble du contenu, elle doit prendre en compte plusieurs objets de tailles et de positions variables. Cette problématique est présente dans de nombreuses applications, notamment la surveillance vidéo, la conduite autonome ou bien l'analyse d'images médicales.

Au fil du temps, diverses approches ont été développées pour répondre à cette problématique, allant des méthodes fondées sur des caractéristiques visuelles aux modèles modernes basés sur l'apprentissage profond. Ce chapitre étudie les principaux détecteurs d'objets utilisés aujourd'hui, en reprenant leurs principes fondamentaux et leur évolution.

Ce chapitre débutera par l'étude des détecteurs en deux étapes, tels que *Faster R-CNN* [Ren *et al.*, 2016], qui s'appuient sur des réseaux de propositions de régions, les *RPN*, pour générer des régions candidates avant la classification. Ensuite YOLO sera abordé, une approche innovante qui reformule la détection comme un problème de régression unique, permettant une exécution en temps réel. Enfin, les détecteurs basés sur les Transformers, notamment *DETR* [Carion *et al.*, 2020 ; Zhao *et al.*, 2024 ; Lv *et al.*, 2024], qui s'affranchissent des mécanismes traditionnels en exploitant l'attention globale pour localiser et classer les objets.

Cette présentation des détecteurs d'objets servira de base aux développements abordés dans le chapitre suivant, où ces concepts vont être détaillés.

3.2 Principe des détecteurs et RPN (*Region Proposal Network*)

3.2.1 Détection d'objets

En vision par ordinateur, et plus particulièrement dans le cadre des réseaux neuronaux, la classification, la localisation et la détection remplissent des fonctions distinctes mais complémentaires. Ces différentes tâches sont illustrées dans la figure 3.1 ci-après.

La détection (figure 3.1c) consiste à identifier et localiser plusieurs objets dans une image en leur attribuant une étiquette de classe et en définissant leurs limites spatiales à l'aide de boîtes englobantes. Cette tâche combine à la fois la classification (attribution d'une catégorie aux objets) et la localisation (détermination de leur position).

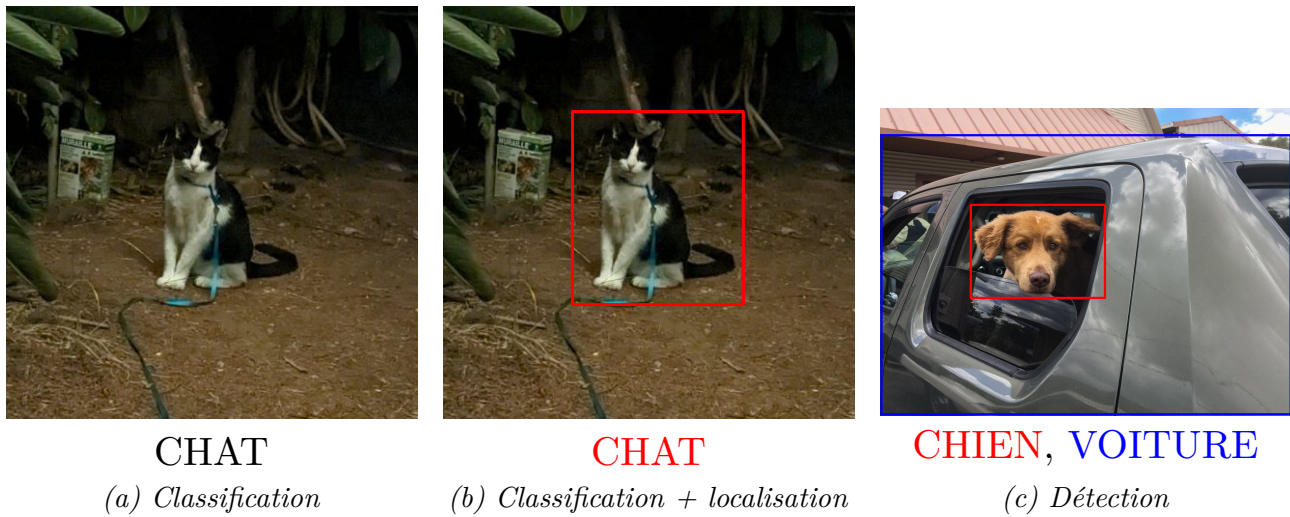


FIGURE 3.1 – Illustration des différences entre les différentes tâches dans le traitement d'images.

La tâche de détection soulève plusieurs défis liés à la variabilité des objets et de la sémantique des images. Un même objet peut apparaître sous différents angles, tailles ou conditions d'éclairage, voire être partiellement occulté par d'autres éléments de l'image. La diversité des contextes et, parfois, l'exigence d'une détection robuste en temps réel renforcent la complexité du problème. L'intégration des réseaux de propositions de régions dans les détecteurs à deux étapes a permis de concevoir des systèmes à la fois plus rapides et plus précis. La section suivante détaillera ces détecteurs et leur fonctionnement.

3.2.2 Approches initiales basées sur la recherche de régions

Avant l'émergence des méthodes d'apprentissage profond, la détection d'objets s'appuyait principalement sur une exploration systématique des images. L'objectif était d'identifier les régions susceptibles de contenir des objets avant d'y appliquer un modèle de classification.

Les premières approches reposaient sur des descripteurs de caractéristiques couplés à des classifieurs. Parmi elles, les histogrammes de gradients orientés, *Histogram of Oriented Gradients (HOG)* [Dalal et Triggs, 2005], les transformées invariantes de caractéristiques locales, *Scale-Invariant Feature Transform (SIFT)* [Lowe, 2004], ainsi que les cascades de Haar [Haar, 1910; Viola et Jones, 2001; Lienhart et Maydt, 2002], étaient fréquemment utilisées en combinaison avec des modèles comme les machines à vecteurs de support, *SVM* [Cortes et Vapnik, 1995]. Ces méthodes permettaient d'extraire des caractéristiques discriminantes pour distinguer les objets des arrière-plans, mais elles reposaient sur des représentations fixes et ne s'adaptaient pas correctement aux variations d'apparence, d'échelle ou d'éclairage des objets détectés.

Ces approches présentaient toutefois des limitations majeures. L'extraction des caractéristiques, manuelle et statique, n'était pas toujours adaptée aux tâches de détection complexes. De plus, les méthodes basées sur les fenêtres glissantes devaient analyser un grand nombre de sous-régions, entraînant un coût computationnel élevé et une inefficacité face aux variations de taille des objets.

Pour pallier ces limitations, des stratégies de sélection automatique des régions d'intérêt ont été développées. Parmi elles, *Selective Search* [Uijlings et al., 2013] repose sur un algorithme de regroupement hiérarchique exploitant des indices de similarité tels que la couleur, la texture et la continuité des contours. Plutôt que de parcourir l'image à plusieurs résolutions avec des

fenêtres glissantes, cette approche génère un ensemble plus restreint et pertinent de régions candidates, réduisant ainsi le coût de calcul.

Les histogrammes de gradients orientés, *HOG* [Dalal et Triggs, 2005], sont une méthode importante pour capturer l'apparence et la structure locale d'un objet. Elle repose sur le calcul des orientations de gradient au sein de cellules locales, formant ainsi des histogrammes représentatifs des motifs de bord.

Une autre approche est la méthode "*Bag of Features (BoF)*" [Csurka *et al.*, 2004 ; Nowak *et al.*, 2006], qui représente une image comme un ensemble de caractéristiques locales indépendantes, agrégées sous forme d'un vocabulaire visuel. Ces descripteurs sont souvent couplés à des classifieurs comme les machines à vecteurs de support, *SVM* [Cortes et Vapnik, 1995], pour effectuer la classification.

Bien que plus performantes que les techniques fondées sur des descripteurs manuels, ces méthodes présentent également des limites. L'absence d'apprentissage guidé par les données restreint leur capacité d'adaptation à des contextes complexes. De plus, la qualité des propositions de régions dépend fortement des heuristiques utilisées, pouvant affecter la précision globale du modèle.

Ces limitations ont conduit au développement d'approches basées sur l'apprentissage profond, où les Convolutional Neural Network (CNN) apprennent directement des représentations adaptées minimisant une tâche donnée et pour cela apprennent une représentation optimale des données et du problème. Ce principe sera approfondi dans la section suivante à travers l'étude des détecteurs en deux étapes, qui marquent une transition vers des méthodes plus précises et performantes.

3.2.3 Détecteurs à deux étapes et RPN

Les détecteurs en deux étapes adoptent une approche séquentielle, dissociant la génération des propositions de régions de leur classification. Cette méthodologie améliore la précision par rapport aux approches traditionnelles en se concentrant uniquement sur les régions d'intérêt.

Dans un premier temps, un module spécifique extrait un ensemble de régions candidates susceptibles de contenir des objets. Ensuite, un second réseau applique un classifieur et affine les boîtes englobantes pour en améliorer la précision.

Ce cadre a été introduit avec R-CNN [Girshick *et al.*, 2014], optimisé par Fast R-CNN [Girshick, 2015], puis entièrement repensé avec Faster R-CNN [Ren *et al.*, 2016], qui intègre un réseau de propositions de régions, *RPN*. Le fonctionnement du R-CNN va être étudié plus en détails ci-après.

3.2.3.1 R-CNN

Le modèle R-CNN [Girshick *et al.*, 2014] se distingue des méthodes traditionnelles en intégrant l'extraction de régions à un CNN.

Cette approche utilise un algorithme tel que *Selective Search* [Uijlings *et al.*, 2013] pour générer un grand nombre de régions candidates dans l'image. Chaque région est redimensionnée puis traitée individuellement par un CNN pré-entraîné, qui en extrait un vecteur de caractéristiques. Ces caractéristiques sont ensuite exploitées par un classifieur, comme un SVM, pour attribuer une catégorie à l'objet détecté. Un modèle de régression vient aussi ajuster la position et la taille de la Bounding Box (BBOX) afin d'améliorer la précision.

Ces différentes étapes sont illustrées au travers de la figure 3.2.

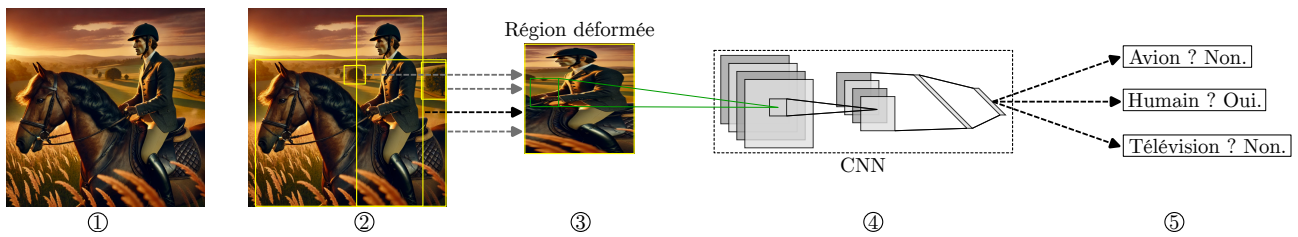


FIGURE 3.2 – Les différentes étapes du R-CNN. Inspiré de l'article de Girshick et al. [2014].

Cependant, cette approche présente plusieurs limitations. Le traitement indépendant de chaque région engendre un temps de calcul important, rendant cette méthode difficilement applicable aux applications en temps réel. De plus, le processus est fragmenté en plusieurs étapes (extraction, classification, régression), nécessitant une exécution séparée pour chacune, limitant son efficacité.

Bien que cette stratégie permette d'éviter un balayage exhaustif de l'image, l'algorithme de *Selective Search* repose sur des heuristiques prédéfinies, qui ne sont pas optimisées pour la tâche de détection. Il génère de nombreuses régions candidates redondantes, dont la pertinence n'est pas toujours garantie. De plus, le traitement indépendant de chacune de ces régions par le réseau induit une importante redondance computationnelle.

Ces limitations ont conduit au développement de Fast R-CNN une version optimisée de R-CNN permettant de traiter les images plus efficacement. Ce modèle va être développé dans la section suivante.

3.2.3.2 Fast R-CNN

Fast R-CNN [Girshick, 2015] a été conçu pour pallier les principales limitations de R-CNN en optimisant l'efficacité et la rapidité du processus.

Contrairement à R-CNN, Fast R-CNN traite l'image en une seule passe à travers un CNN, générant en sortie une *feature map* unique. Plutôt que d'extraire séparément les caractéristiques de chaque région candidate, celles-ci sont directement projetées sur cette carte, réduisant le coût computationnel.

Afin d'uniformiser la taille des régions candidates avant la classification, Fast R-CNN introduit la couche *Region of Interest (RoI) pooling*, permettant d'extraire les zones pertinentes sans recalculer des convolutions. Ces régions sont ensuite traitées simultanément par un classifieur pour l'identification des objets et par un régresseur pour ajuster des BBOX.

Le pipeline de Fast R-CNN repose sur une unique extraction des caractéristiques à partir de l'image complète, la figure 3.3 ci-dessous représente l'entièreté du processus.

En intégrant directement la classification et l'ajustement des BBOX au sein d'un même réseau, Fast R-CNN adopte une approche "bout en bout", éliminant ainsi le recours à un classifieur externe. Cette conception améliore l'efficacité du réseau, sa précision tout en réduisant le temps de calcul.

Bien que Fast R-CNN améliore l'efficacité, il reste limité par l'utilisation de *Selective Search* pour la génération des propositions de régions, ralentissant l'ensemble du modèle. Pour surmonter cette contrainte, Faster R-CNN intègre un mécanisme optimisé qui génère ces propositions de manière plus efficace, réduisant ainsi le temps de calcul tout en améliorant la précision.

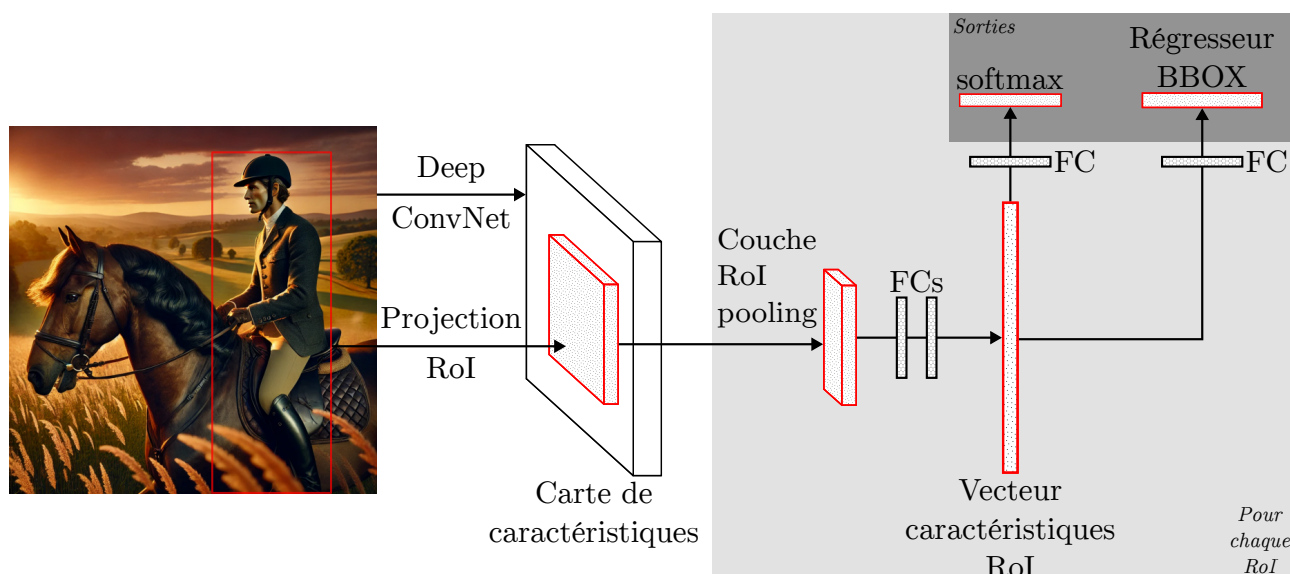


FIGURE 3.3 – Illustration du pipeline de Fast R-CNN. Inspiré de l'article de Girshick [2015].

3.2.3.3 Faster R-CNN

Faster R-CNN [Ren *et al.*, 2016] a été développé pour surmonter la principale limitation de Fast R-CNN : sa dépendance à *Selective Search* pour la génération des propositions de régions. L'architecture de Faster R-CNN reprend les principes de Fast R-CNN pour l'extraction des caractéristiques et la classification des objets. Cependant, au lieu de s'appuyer sur une approche heuristique pour générer des régions d'intérêt, elle intègre un réseau de neurones capable de générer des propositions de régions directement à partir de la carte de caractéristiques extraite du CNN de départ, optimisant ainsi le pipeline global illustré dans la figure 3.4.

La principale contribution de Faster R-CNN est l'intégration du RPN, venant remplacer *Selective Search* par un mécanisme entièrement appris. Le RPN est entraîné conjointement avec le reste du réseau, permettant une meilleure adaptation aux données d'apprentissage, son fonctionnement va être développé dans la prochaine section.

3.2.3.4 Region Proposal Networks (RPN)

L'objectif du RPN est d'identifier efficacement les zones pertinentes contenant des objets, en réduisant le nombre de régions analysées par le classifieur final. D'après Ren *et al.* [2016], le RPN peut être schématisé comme dans la figure 3.5.

Le RPN fonctionne en ajoutant une couche de convolution supplémentaire sur la carte de caractéristiques générée par le *backbone* du réseau. Cette couche agit comme une fenêtre glissante parcourant l'ensemble de la carte, générant des prédictions à chaque position. Dans la figure 3.5, chaque position de cette grille est associée à k boîtes d'ancrage (*anchor boxes*) de différentes tailles et proportions. La couche de régression du RPN produit ainsi $4k$ valeurs ajustant les coordonnées des BBOX, tandis que la couche de classification génère $2k$ scores indiquant la probabilité qu'une boîte contienne un objet ou qu'elle appartienne au fond. L'entraînement repose sur une fonction de perte combinée, intégrant deux composantes. La première est une perte de classification basée sur l'entropie croisée binaire, *Binary Cross-Entropy (BCE)* [Ruby et Yendapalli, 2020], qui utilise les scores de la couche de classification. La seconde est une perte de régression, ajustant les coordonnées des BBOX en minimisant une fonction de type "*Smooth L1*" (eq. [3.2-3.3]).

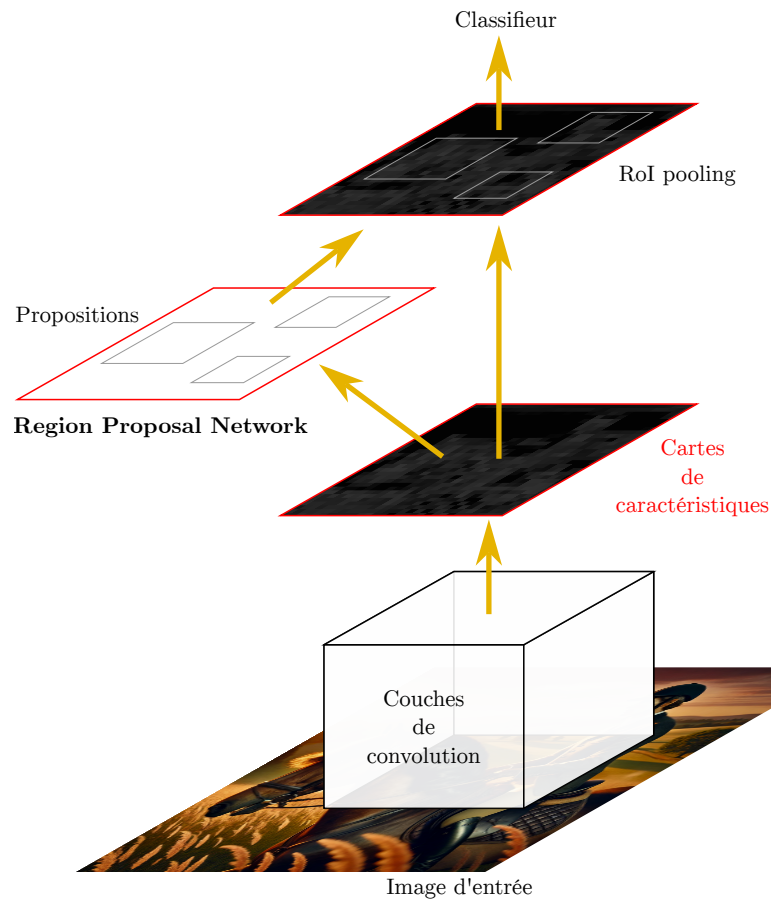


FIGURE 3.4 – Illustration du pipeline de Faster R-CNN incorporant le RPN. Inspiré de l'article de Ren et al. [2016].

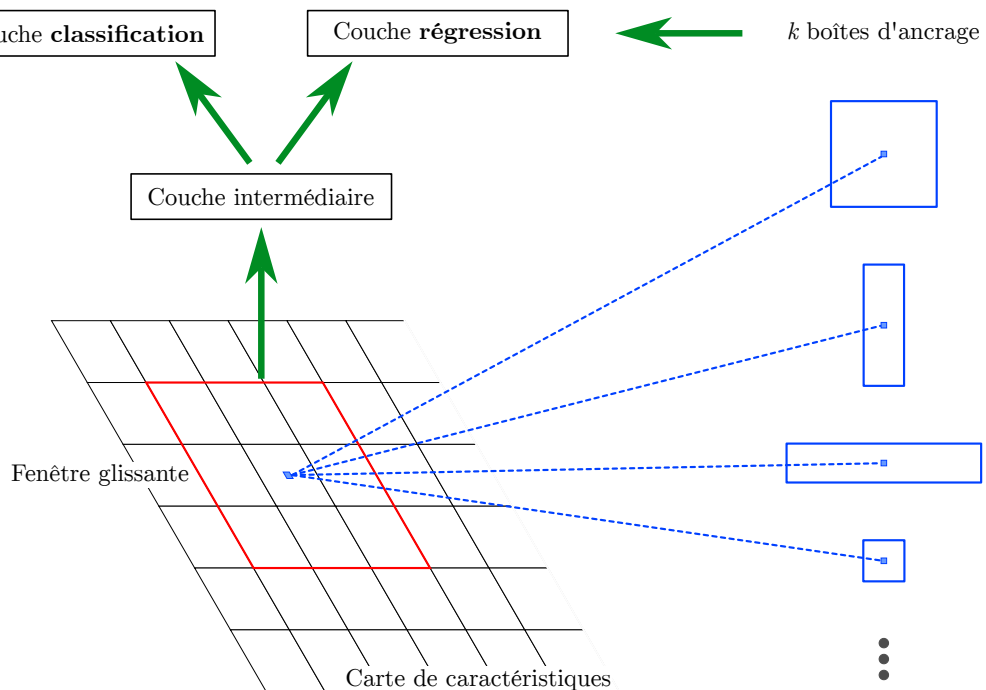


FIGURE 3.5 – Region Proposal Network (RPN) d'après Ren et al. [2016].

L'expression complète de la fonction de perte globale est donnée par :

$$\mathcal{L}(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i \mathcal{L}_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* \mathcal{L}_{reg}(t_i, t_i^*) \quad (3.1)$$

$$\mathcal{L}_{reg}(t_i, t_i^*) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L1}(t_i - t_i^*) \quad (3.2)$$

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{si } |x| < 1, \\ |x| - 0.5 & \text{sinon.} \end{cases} \quad (3.3)$$

soit i l'index d'une ancre et p_i la probabilité prédite de l'ancre i contienne un objet. La vérité terrain p_i^* est égale à 1 si l'ancre est positive, 0 sinon. t_i est le vecteur représentant les 4 coordonnées paramétrées de la BBOX prédite, et t_i^* la vérité de la BBOX associée avec une ancre positive. Le terme $p_i^* \mathcal{L}_{reg}$ indique que la perte de régression est appliquée uniquement si l'ancre est positive ($p_i^* = 1$), et est désactivée dans le cas contraire ($p_i^* = 0$).

L'ensemble de la paramétrisation, incluant la pondération des termes $\lambda, N_{cls}, N_{reg}$ ainsi que des coordonnées t_i , est détaillé dans l'article de Ren *et al.* [2016].

Le RPN génère un grand nombre de propositions de régions. Pour ne conserver que les plus pertinentes, il applique l'algorithme Non-Maximum Suppression (NMS), qui réduit significativement le nombre de régions transmises au détecteur final, améliorant ainsi la rapidité globale du modèle. Un exemple illustrant le fonctionnement de la NMS est présenté dans la figure 3.6.

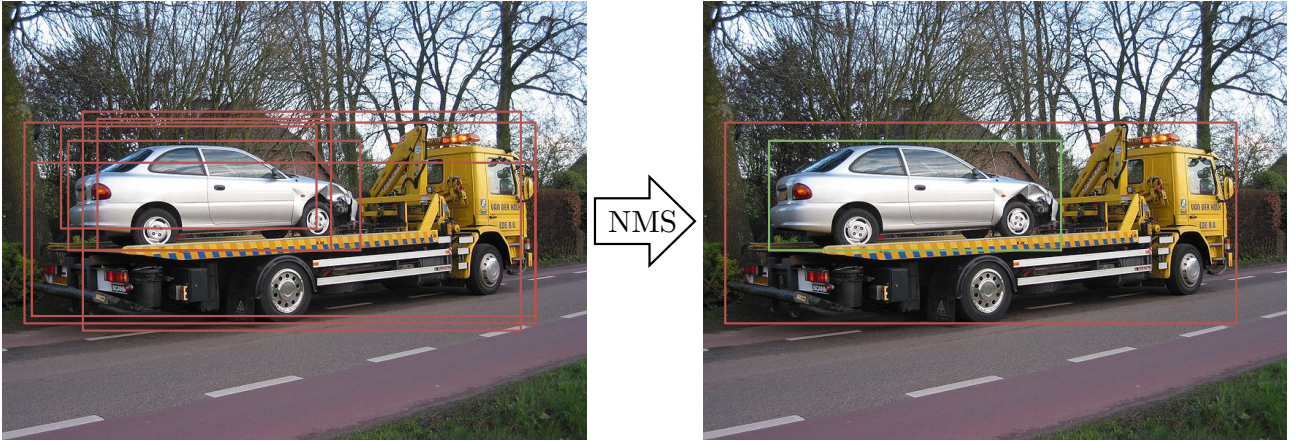


FIGURE 3.6 – Exemple de filtrage des propositions de région par la NMS.

L'intégration du RPN dans Faster R-CNN permet de rendre le processus entièrement exploitable de bout-en-bout, en optimisant et en co-entraînant la génération des régions avec le reste du réseau.

Contrairement aux méthodes heuristiques, cette approche s'adapte mieux aux caractéristiques des données d'apprentissage. En éliminant le calcul séparé des propositions de régions, le RPN permet à Faster R-CNN d'atteindre des performances supérieures en termes de rapidité et de précisions.

Bien que les détecteurs en deux étapes, tels que Faster R-CNN, aient amélioré la précision et l'efficacité de la détection d'objets grâce à l'introduction du RPN, ces architectures restent relativement complexes et coûteuses en calcul. La séparation entre la génération des propositions de régions et la classification entraîne un temps d'inférence plus élevé, limitant leur utilisation dans des applications nécessitant des décisions en temps réel.

Pour surmonter cette restriction, la famille de modèles YOLO a introduit une nouvelle approche, reformulant la détection d'objets comme un problème de régression unique. La section suivante retrace l'évolution de YOLO, de sa première version jusqu'à la version ayant été utilisée dans cette thèse, en mettant en avant les améliorations apportées à chaque version.

3.3 You Only Look Once (YOLO)

Les détecteurs d'objets en une seule étape ont été développés pour offrir un traitement plus rapide et plus efficaces que les approches en deux étapes. Introduit par Redmon *et al.* [2015], YOLO propose une architecture novatrice qui reformule la détection d'objets en un problème de régression unique. Contrairement aux détecteurs à deux étapes, qui génèrent d'abord des propositions de régions avant la classification, YOLO prédit directement les classes et les BBOX en une seule passe à travers le réseau, permettant ainsi un gain de rapidité significatif.

3.3.1 Historique et évolutions de YOLO

YOLO a transformé la détection d'objets en proposant un système unique, rapide et performant. Cette approche a donné lieu à de nombreuses évolutions, comme le montre la frise chronologique de la figure 3.7.

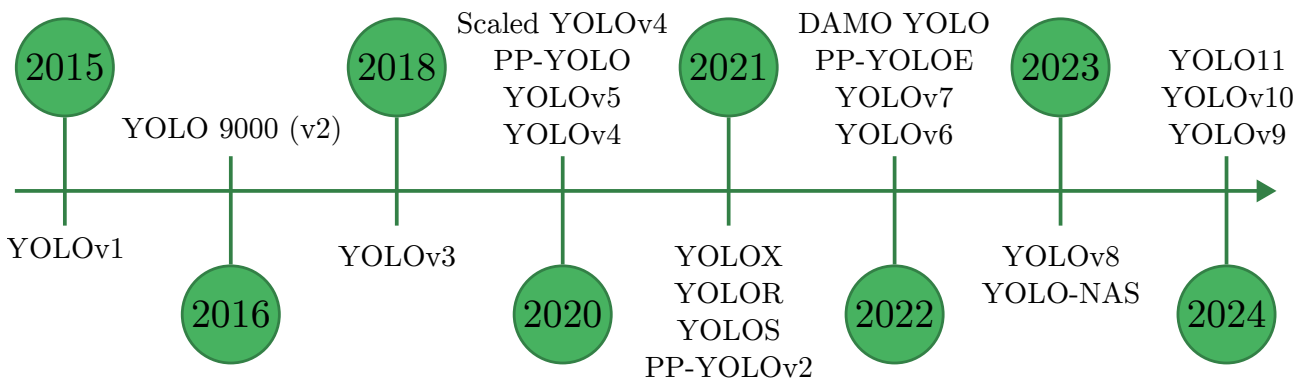


FIGURE 3.7 – Chronologie des versions de YOLO, inspiré de Terven et al. [2023]

La première version, YOLOv1, s'est distinguée des méthodes traditionnelles en introduisant une nouvelle approche, en divisant l'image en une grille et en associant à chaque cellule des prédictions de BBOX et des scores de classes. Même si cette conception a permis un gain significatif en vitesse, elle présentait toutefois des limites en termes de précision, notamment pour la détection des petits objets et ceux superposés.

Avec YOLOv2 [Redmon et Farhadi, 2017] et YOLOv3 [Redmon et Farhadi, 2018], plusieurs améliorations ont été apportées, dont le changement de *backbone* plus performants, tels que Darknet-19 et Darknet-53, optimisant l'extraction des caractéristiques. Par ailleurs, la gestion des objets à différentes échelles a été améliorée grâce à la prédiction multi-niveaux, exploitant le *Feature Pyramid Network (FPN)* [Lin et al., 2017]. Cette approche a renforcé la détection des objets de tailles variées.

Après YOLOv3, le développement de YOLO s'est poursuivi avec YOLOv4 [Bochkovskiy et al., 2020], introduisant l'utilisation d'un nouveau *backbone* dans CSPDarknet-53. Outre ce changement d'architecture, plusieurs améliorations ont été apportées, en particulier sur

l'augmentation de données, avec la technique “*mosaic data augmentation*” qui combine plusieurs images en une seule, renforçant ainsi la robustesse du modèle face aux variations d'échelle et de contexte. Cette approche enrichi la diversité des échantillons vus par le réseau lors de l'entraînement, réduisant ainsi les risques de sur-apprentissage. Par ailleurs, des optimisations en régularisation ont aussi été mises en place. Enfin, YOLOv4 a structuré ses optimisations autour des concepts de *Bag of Freebies* (augmentation mosaïque, régularisation ou le DropBlock) et *Bag of Specials* (SPP, Mish ou modules d'attention), regroupant des techniques d'entraînement et d'inférence visant à améliorer les performances sans augmenter le coût en calcul.

L'aspect segmentation, bien que présent dans YOLOv5, ne sera pas abordé ici. Le développement de YOLOv5 par Ultralytics [Jocher, 2020] a poursuivi l'optimisation de l'architecture pour une meilleure efficacité sur les plateformes modernes. Sa principale contribution est une architecture développée en PyTorch [Paszke *et al.*, 2019] au lieu de Darknet, offrant ainsi une meilleure intégration avec les frameworks d'apprentissage profond. De plus, les différentes tailles de modèles de YOLOv5 ont été conçues avec moins de paramètres, améliorant la rapidité d'inférence et la portabilité sur des appareils aux ressources limitées. D'autres optimisations notables incluent l'*AutoAnchor* pour l'ajustement automatique des ancres, des techniques d'augmentation de données intégrées comme pour la mosaïque ou le MixUp, ainsi que l'utilisation d'une amélioration de la stabilité et la généralisation lors de l'entraînement appelé “*Exponential Moving Average (EMA)*”.

La figure 3.8 synthétise les principales évolutions des différentes versions de YOLO présentées précédemment.

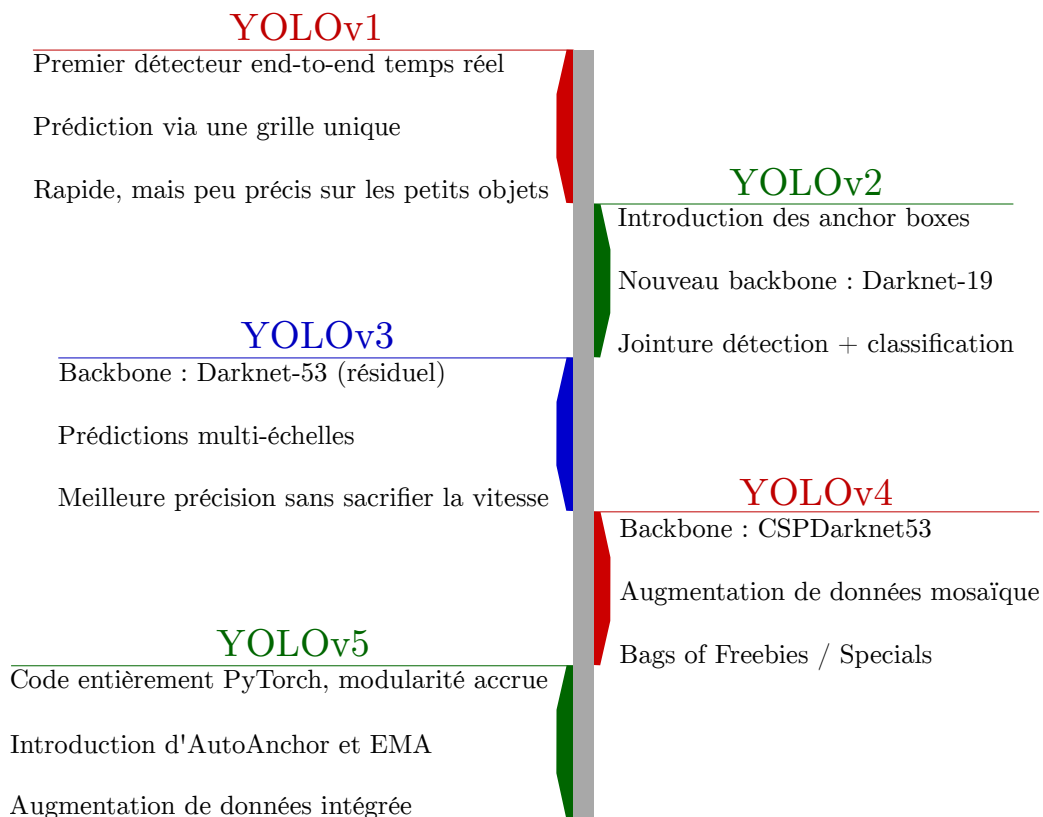


FIGURE 3.8 – Récapitulatif des avancées majeures pour les premières versions de YOLO.

À partir de la version 6, YOLO continue d'évoluer avec l'introduction d'optimisations computationnelles, de structures allégées et de nouvelles techniques de régularisation visant à améliorer la robustesse du modèle. Ces améliorations successives soulignent l'importance de

YOLO dans les applications en temps réel et de son utilisation dans divers domaines. Ultralytics a réalisé un benchmark comparant les dernières versions de YOLO, présenté dans la figure 3.9.

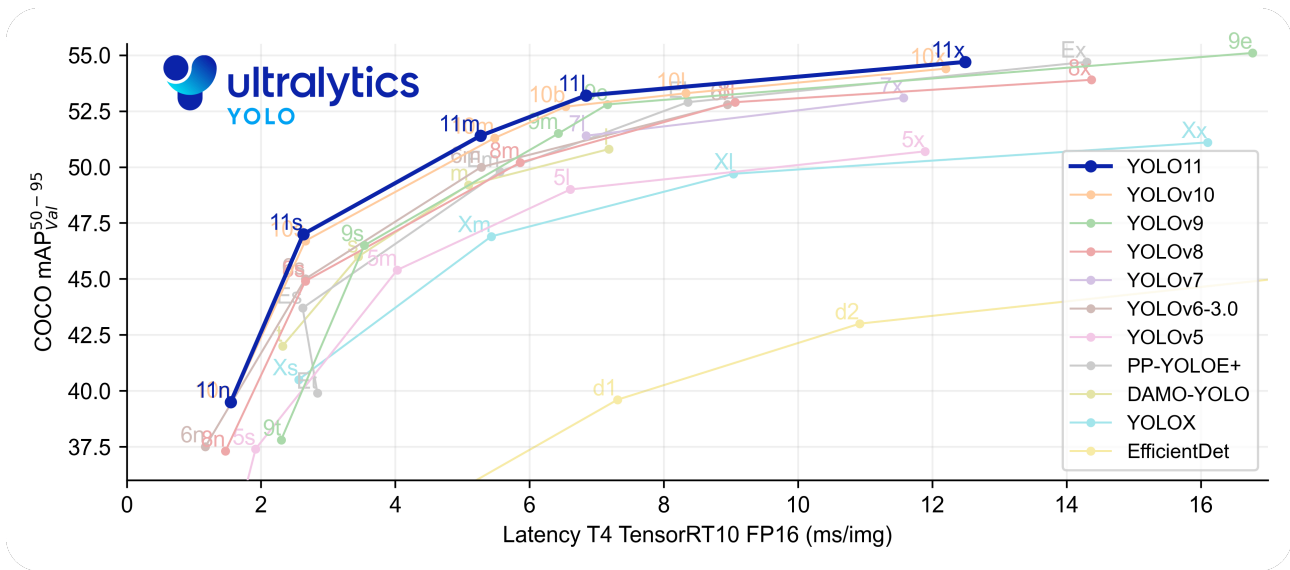


FIGURE 3.9 – Comparaison des performances des dernières versions de YOLO, extrait de Jocher et Qiu [2024]

Bien que les versions récentes de YOLO offrent de meilleures performances, YOLOv5 reste une solution stable. Sa modularité représente un avantage important pour effectuer des modifications approfondies de l'architecture du réseau.

La section suivante présentera le principe de fonctionnement de YOLOv5, en détaillant son architecture et sa méthode de prédiction.

3.3.2 YOLOv5

YOLOv5 s'inscrit dans la continuité des modèles YOLO et est largement utilisé en vision par ordinateur. Il constitue une base pour de nombreuses extensions et applications, que ce soit dans le domaine médical, pour la reconnaissance de structures anatomiques et l'aide au diagnostic [Dipu *et al.*, 2021 ; Ragab *et al.*, 2024], ou dans l'industrie, notamment pour l'analyse d'images issues de drones ou de systèmes embarqués [Jung et Choi, 2022].

Pour comprendre les performances et la flexibilité de YOLOv5, il convient d'analyser son architecture ainsi que les optimisations qui le distinguent des versions antérieures.

3.3.2.1 Architecture

Le modèle de détection YOLOv5 se compose de trois parties principales : le *backbone* (dorsale), le *neck* (cou) et le *head* (tête), chacune ayant son importance dans l'apprentissage du réseau.

Chaque partie étant composée de différents blocs plus ou moins complexes, parfois reliés par des connexions résiduelles. La figure 3.10 illustre plus en détails cette architecture.

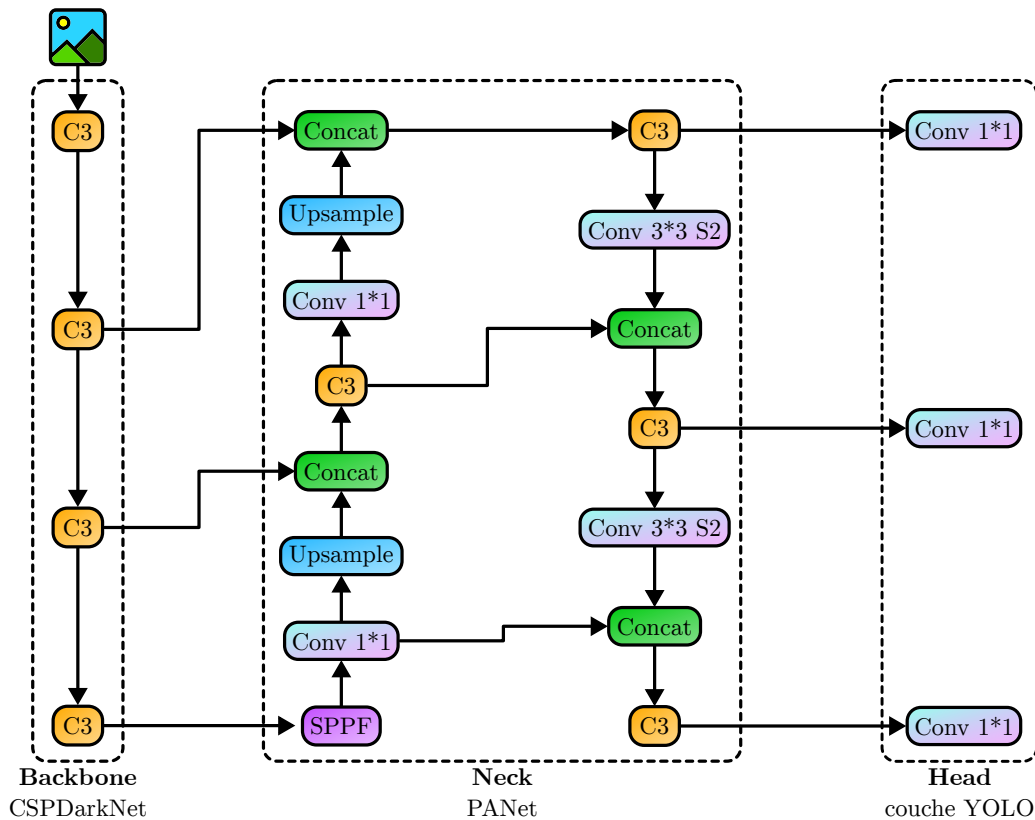


FIGURE 3.10 – Structure complète de YOLOv5.

Pour mieux comprendre le fonctionnement de ce réseau, les différentes parties seront examinées et certains blocs seront approfondis.

3.3.2.1.1 Backbone (Dorsale)

Le *backbone* est chargé d'extraire les caractéristiques de l'image d'entrée. Au fil des versions de YOLO, plusieurs *backbone* ont été utilisés. Dans YOLOv5, le modèle utilise le CSPDarknet53, introduit dans YOLOv4 [Bochkovskiy *et al.*, 2020] une variante du Darknet53 employé dans YOLOv3 [Redmon et Farhadi, 2018], sur laquelle la stratégie *Cross Stage Partial (CSP)* a été appliquée.

YOLO utilise des blocs résiduels et denses pour maintenir un flux d'information, y compris dans les couches les plus profondes du réseau. Cette architecture permet de surmonter les problèmes de disparition du gradient. Cependant, l'utilisation de blocs résiduels et denses introduit un autre problème : la redondance de gradients. CSPNet [Wang *et al.*, 2020] permet d'atténuer cet effet tout en conservant l'avantage de la réutilisation des caractéristiques de DenseNet [Huang *et al.*, 2017]. Il contribue à réduire l'excès d'informations redondantes en tronquant le flux de gradient.

Son application aux différents blocs est illustrée dans la figure 3.11.

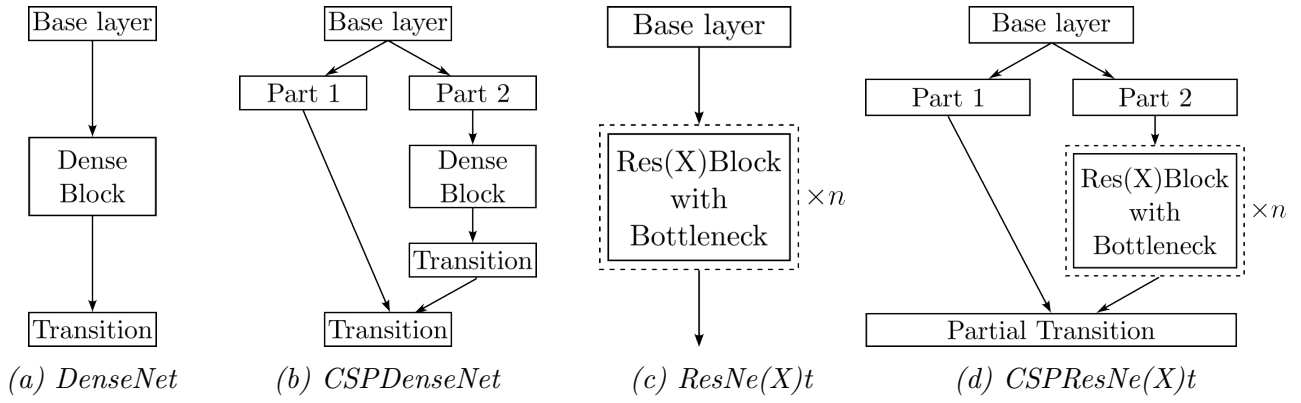


FIGURE 3.11 – CSPNet appliqué à ResNet et DenseNet, inspiré de Wang et al. [2020].

Dans la figure 3.10 le premier bloc est appelé le module C3. Il est précédé d'une couche convolutive suivie de blocs résiduels, il peut être schématisé par la figure 3.12 ci-après.

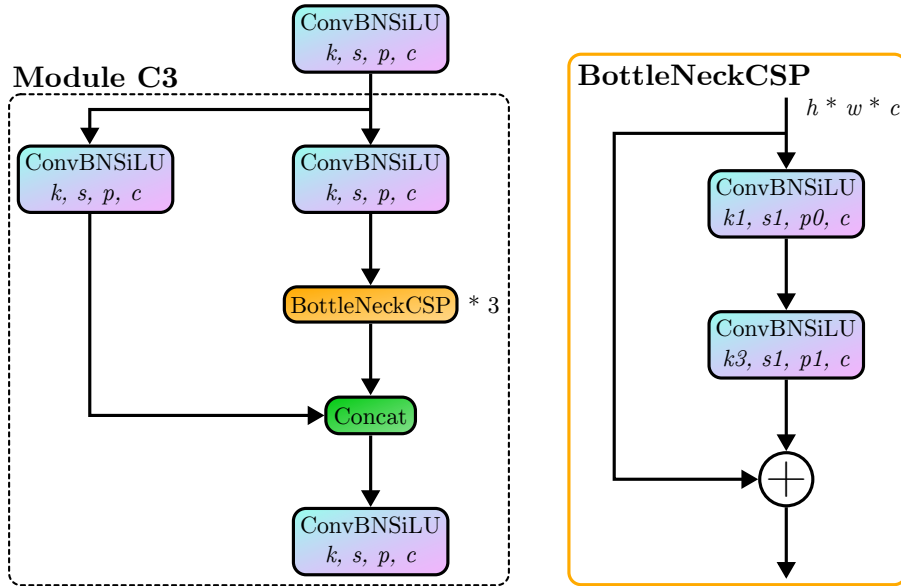


FIGURE 3.12 – Détails du module C3, extrait de Ultralytics [2023].

La stratégie du réseau CSP consiste à diviser la carte de caractéristiques en deux parties. Une partie traverse une série de blocs de convolution, tandis que l'autre partie est directement transmise à la couche suivante sans modification. Ensuite, les deux parties sont fusionnées selon une certaine hiérarchie, généralement une concaténation. Cette approche permet de réduire la redondance des calculs tout en préservant les caractéristiques importantes, améliorant ainsi l'efficacité du réseau.

En conclusion, le *backbone* de YOLOv5, basé sur le CSPDarknet53, est conçu pour extraire efficacement les caractéristiques de l'image d'entrée tout en atténuant les problèmes de disparition de gradient et de redondance de calculs grâce à la stratégie CSP. Cette approche améliore l'efficacité de calcul sans compromettre la qualité des informations extraites qui vont être transmises à la partie suivante, le *neck*.

3.3.2.1.2 Neck (Cou)

Une fois les caractéristiques extraites et les dimensions réduites, ces informations sont transmises à la partie *neck* du réseau. Cette partie intermédiaire prépare les données avant leur

traitement par la tête pour la prédiction finale.

Son objectif principal est d'améliorer la capacité du modèle à détecter des objets à différentes échelles en combinant les informations issues de plusieurs couches du *backbone*.

Il réduit la complexité computationnelle tout en préservant l'efficacité de l'agrégation multi-échelles. Le *Spatial Pyramid Pooling - Fast (SPPF)* constitue une version optimisée du module SPP, initialement introduit par He *et al.* [2014a] et intégré dans YOLOv4. Cette version vise à réduire la complexité computationnelle tout en conservant l'efficacité de l'agrégation d'informations à différentes échelles pour la détection d'objets.

Le processus du SPPF se déroule en plusieurs étapes. La première consiste en un “*max pooling*” à trois niveaux, appliqué via trois couches de 5×5 avec un *stride* de 1 et un *padding* de 2. Chaque couche permet d'avoir une vue d'ensemble de l'image tout en gardant sa dimension d'origine, comme représenté par les équations (3.4-3.5). La deuxième étape correspond à la concaténation des cartes de caractéristiques à différentes échelles. Une fois le regroupement effectué, cette concaténation permet d'intégrer plusieurs niveaux de détails, représenté par l'équation (3.6).

D'un point de vue mathématique, ce processus peut être exprimé à l'aide des équations suivantes :

$$\begin{aligned} F_1 &= P(F_{\text{in}}) \\ F_2 &= P(F_1) \\ F_3 &= P(F_2) \end{aligned} \quad (3.4)$$

avec :

$$P(\cdot) = \text{MaxPool2d}[\text{kernel} = 5, \text{stride} = 1, \text{padding} = 2](\cdot) \quad (3.5)$$

$$F_{\text{concat}} = \mathcal{C}(F_{\text{in}}, F_1, F_2, F_3) \quad (3.6)$$

où F_{in} est l'entrée du module SPPF (une carte de caractéristiques issue du *backbone*) et $\mathcal{C}$ la concaténation des différentes sorties.

Après cette concaténation, le nombre de canaux des caractéristiques est multiplié par quatre. Sans réduction de dimension, cela augmente la charge computationnelle et rend les données trop volumineuses pour être utilisées efficacement dans les couches suivantes. Une convolution 1×1 permet alors de fusionner les informations multi-échelles et de réduire la dimensionnalité du tenseur en conservant l'essentiel des informations.

L'ensemble du processus est illustré dans la figure 3.13.

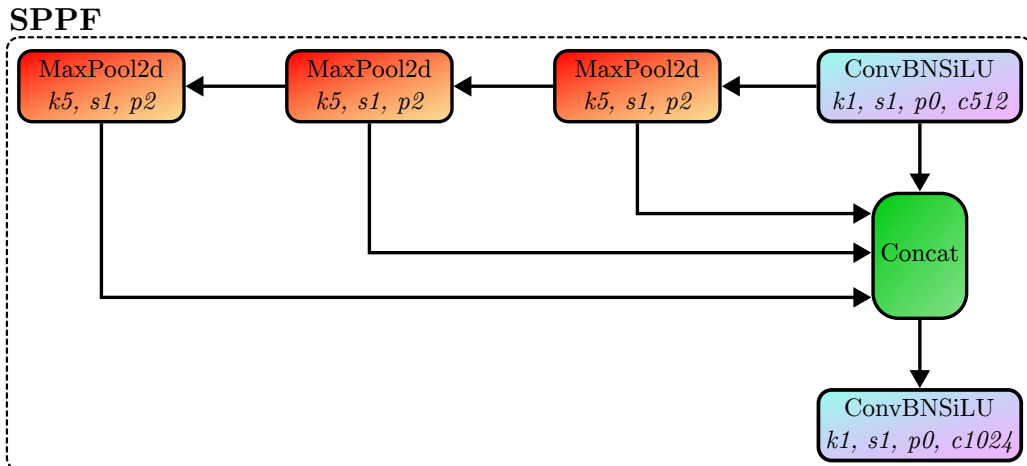


FIGURE 3.13 – Schéma du bloc SPPF au sein de YOLOv5.

Contrairement au SPP classique, où chaque regroupement est appliqué en simultané mais indépendamment sur plusieurs tailles, suivis de plusieurs convolutions, le SPPF applique un même regroupement plusieurs fois séquentiellement. Cela permettant d'accélérer le traitement et optimise l'utilisation de la mémoire, tout en conservant les performances.

Tout comme le *backbone* utilise le CSPDarknet53, le *neck* applique aussi la stratégie CSP au “*Path Aggregation Network (PANet)*” [Liu et al., 2018] pour donner “Cross Stage Partial PAN (CSP-PAN)”.

Le PANet repose sur une propagation plus profonde des caractéristiques extraites grâce au SPPF. Dans un premier temps, la propagation de “bas en haut” transfère les détails de bas niveau, tels que les points, traits ou gradient de couleurs, des couches antérieures vers les couches hiérarchiquement plus haut. Ensuite, la fusion des caractéristiques de “haut en bas” affine les informations sémantiques de haut niveau, comme les roues, fenêtres ou lampes, afin d'améliorer la localisation des objets. La réutilisation des caractéristiques est assurée par des connexions résiduelles qui sont concaténées au travers du réseau. La figure 3.14 montre la structure originale du PANet.

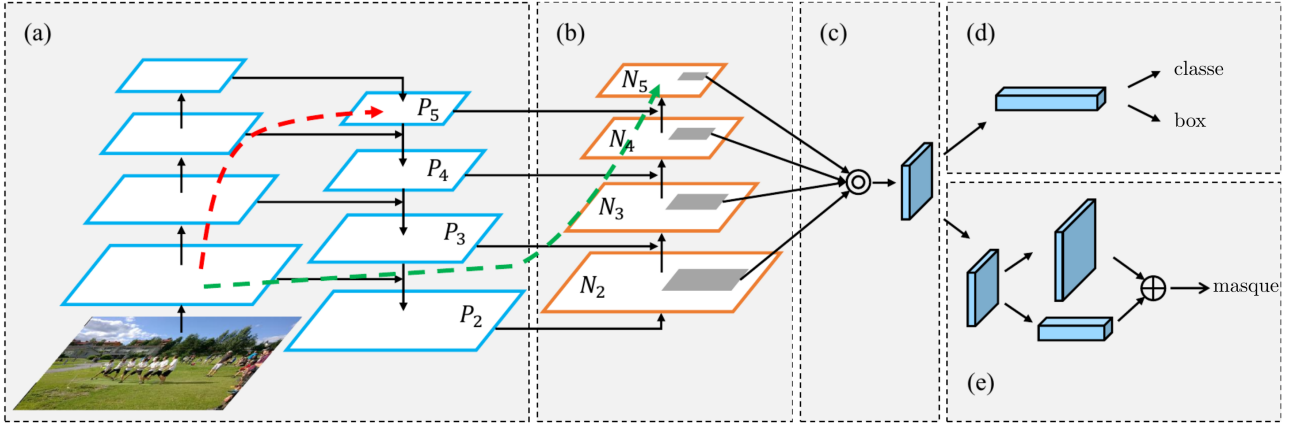


FIGURE 3.14 – Illustration du réseau PANet. (a) FPN [Lin et al., 2017] backbone, (b) augmentation de “bas en haut”, (c) regroupement adaptatif de caractéristiques, (d) branche des boîtes, (e) fusion des Fully Connected (FC).

Grâce à cette architecture pyramidale, le PANet facilite le flux d'informations. Comme dans le *backbone*, le système de CSP est appliquée à ce réseau avec les mêmes objectifs : limiter la redondance des calculs, renforcer le partage d'informations et éviter la fuite de gradient.

3.3.2.1.3 Head (Tête)

YOLOv5 utilise la même structure pour la tête que YOLOv3 et YOLOv4. Cette partie se compose de trois couches convolutives chargées de prédire les BBOX ($x, y, height, width$), ainsi que les scores de confiance et les classes des objets.

La seule différence se situe dans le calcul des coordonnées des BBOX. Le système d'équations (3.7) correspond à celui utilisé dans les versions précédentes, alors que les équations (3.8) sont celles utilisées dans YOLOv5.

$$\begin{aligned} b_x &= \sigma(t_x) + c_x \\ b_y &= \sigma(t_y) + c_y \\ b_h &= p_h \cdot e^{t_h} \\ b_w &= p_w \cdot e^{t_w} \end{aligned} \quad (3.7)$$

$$\begin{aligned} b_x &= (2 \cdot \sigma(t_x) - 0.5) + c_x \\ b_y &= (2 \cdot \sigma(t_y) - 0.5) + c_y \\ b_h &= p_h \cdot (2 \cdot \sigma(t_h))^2 \\ b_w &= p_w \cdot (2 \cdot \sigma(t_w))^2 \end{aligned} \quad (3.8)$$

où b_x, b_y représentent les coordonnées du centre de la BBOX prédite et b_h, b_w sa largeur et sa hauteur. t_x, t_y, t_h, t_w sont les valeurs de sortie du réseau qui doivent être transformées en coordonnées réelles. c_x, c_y désignent les coordonnées du coin supérieur gauche de la cellule d'ancrage dans la grille de détection, tandis que p_h, p_w sont les largeur et hauteur de l'ancre associée à la BBOX prédite. σ est une fonction sigmoïde permettant de contraindre les valeurs entre 0 et 1.

Le fonctionnement de la grille de détection et l'utilisation des ancres seront détaillés dans la section suivante, qui expliquera le processus global du réseau et la manière dont il prédit les BBOX dans les images.

3.3.2.2 Fonctionnement et prédiction

YOLOv5 est un modèle de détection d'objets basé sur l'ancrage. L'image en entrée est divisée en une grille dans laquelle des BBOX ainsi que des probabilités sont prédites en un seul passage. Ce processus se divise en plusieurs étapes permettant d'apprendre au réseau à générer des coordonnées cohérentes.

Le pipeline est le suivant, l'image d'entrée est d'abord redimensionnée, puis divisée en une grille pour en extraire les caractéristiques. Un processus multi-échelles exploite ces caractéristiques afin de prédire des BBOX, qui sont ensuite filtrées pour ne conserver que les plus pertinentes, illustré dans la figure 3.6.

L'image d'entrée est redimensionnée à une dimension fixe de 640×640 pour ensuite être divisée en une grille, chaque cellule de la grille étant responsable de la détection des objets dont le centre se trouve à l'intérieur. Cette division permet au modèle de prédire des objets à plusieurs endroits simultanément.

Contrairement aux précédentes version de YOLO qui s'appuyaient sur une sortie à échelle unique, YOLOv5 utilise une approche de détection multi-échelles, en extrayant des caractéristiques à différentes résolutions, plus efficace pour détecter des objets de différentes tailles.

Les couches de détection fonctionnent à trois échelles différentes : détection à petite échelle (haute résolution, petit pas - *stride*), détection à moyenne échelle (pas intermédiaire) et détection à grande échelle (basse résolution, grand pas). Cette approche multi-résolution améliore la capacité du modèle à reconnaître les objets de petite et de grande taille au sein d'une même image.

Contrairement aux précédentes versions, la fonction d'activation utilisée au sein des couches de convolution est différente du *Leaky Rectified Linear Unit (ReLU)*, YOLOv5 intègre le Sigmoid Linear Unit (SiLU) ou Swish [Ramachandran *et al.*, 2017]. Cette fonction d'activation permet une descente de gradient plus lisse et améliore la stabilité de l'entraînement, tout en évitant le problème de “*dying ReLU*” [Douglas et Yu, 2018].

Une fois les BBOX et les classes prédites, une étape de post-traitement est appliquée pour affiner les détections. Il se peut que certaines prédictions se superposent sur un même objet ; la technique NMS est utilisée afin de filtrer et d'éliminer les détections redondantes.

Lors de l'entraînement, l'optimisation se fait grâce à des fonctions de perte qui vont être décrites ci-après.

3.3.2.3 Fonctions de perte

L'apprentissage des modèles se fait en utilisant une fonction de perte globale qui peut être décomposée en plusieurs sous-fonctions : la loss de localisation ($\mathcal{L}_{loc}$), la loss d'objets ($\mathcal{L}_{obj}$) et la loss de classification ($\mathcal{L}_{cls}$).

Cette décomposition reste valable jusqu'à YOLOv7. À partir de YOLOv8, avec l'adoption d'une approche sans ancres, la composante de perte liée aux objets devient plus implicite. Enfin, avec YOLOv10 et YOLO11, la NMS est retirée, remplaçant cette perte par des mécanismes d'apprentissage et de sélection plus avancés des BBOX.

Pour YOLOv5, l'équation de la fonction de perte totale peut s'écrire de la manière suivante

$$\mathcal{L} = \lambda_{loc}\mathcal{L}_{loc} + \lambda_{cls}\mathcal{L}_{cls} + \lambda_{obj}\mathcal{L}_{obj} \quad (3.9)$$

où λ_{loc} , λ_{cls} , λ_{obj} sont des coefficients d'équilibrage qui pondèrent l'importance de chaque perte.

Les différents termes de cette fonction vont être détaillés par la suite, afin d'expliquer leur rôle et leur mode de calcul.

3.3.2.3.1 Loss de localisation

L'objectif de la loss de localisation est d'optimiser la précision des BBOX en minimisant l'écart entre les prédictions et la vérité terrain.

YOLOv5 utilise une fonction de perte appelée *Complete Intersection over Union (CIoU)* [Zheng et al., 2019], une variante de l'IoU, qui prend en compte non seulement l'IoU lui-même, mais aussi la distance entre les centres des boîtes ainsi que leurs ratios d'aspect :

$$\mathcal{L}_{loc} = 1 - \text{CIoU} \quad (3.10)$$

avec :

$$\text{CIoU} = \text{IoU} - \frac{\rho^2(\mathbf{b}, \mathbf{b}^*)}{c^2} - \alpha v \quad (3.11)$$

L'IoU quantifie le recouvrement entre la BBOX prédite et la BBOX réelle, plus sa valeur se rapproche de 1, plus les deux BBOX sont alignées. Le terme $\rho^2(\mathbf{b}, \mathbf{b}^*)$ représente la distance euclidienne entre les centres des BBOX prédite et réelle, contribuant ainsi à corriger les BBOX mal positionnées. La valeur c^2 correspond à la diagonale du plus petit rectangle englobant les deux BBOX et permet de normaliser la distance des centres dans l'intervalle $[0, 1]$. Le terme v évalue la différence de ratio d'aspect entre la BBOX prédite et la BBOX réelle, empêchant le réseau de générer des BBOX aux proportions incorrectes. Enfin, α est un facteur d'équilibrage ajustant l'influence du ratio d'aspect sur la perte de localisation.

La loss de localisation pénalise les prédictions imprécises, qu'elles concernent la position ou les dimensions des BBOX. En minimisant la perte définie par l'équation (3.10), le modèle est incité à prédire des BBOX aux emplacements et aux tailles correctes, améliorant à la fois la précision, le positionnement des détections et la localisation des objets dans l'image.

3.3.2.3.2 Loss de classification

La perte de classification mesure la capacité du modèle à associer correctement chaque détection à la classe d'objet correspondante. YOLOv5 utilise une fonction de perte BCE, adaptée à la classification multi-classe.

$$\mathcal{L}_{\text{cls}} = - \sum_i^C y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i) \quad (3.12)$$

où C représente le nombre total de classes d'objets, cette perte étant calculée uniquement si $C > 1$. La variable binaire y_i indique si l'objet appartient à la classe i (1 si oui, 0 sinon). Les termes $\log(\hat{p}_i)$ et $\log(1 - \hat{p}_i)$ sont des fonctions logarithmiques qui amplifient l'impact des erreurs de classification : plus une prédiction est incorrecte, plus la perte associée est élevée.

La quantité $\hat{p}_i$ correspond à la probabilité prédite pour la classe i , obtenue en appliquant une fonction d'activation sigmoïde à la sortie produite par le réseau pour cette classe. Sa formulation est donnée par :

$$\hat{p}_i = \sigma(z_i) = \frac{1}{1 + e^{-z_i}} \quad (3.13)$$

où $z_i \in \mathcal{R}$ est la sortie non normalisée (logit) du modèle pour la classe i et σ est la fonction d'activation sigmoïde.

Cette formulation permet d'interpréter chaque sortie comme une probabilité indépendante, ce qui est adapté avec l'utilisation d'une perte BCE et le cadre multi-classe de YOLOv5.

En pratique, l'implémentation utilise la version numériquement stable "*BCE with logits*", qui combine la sigmoïde et la BCE en une seule opération afin d'améliorer la stabilité numérique lors de l'entraînement.

En minimisant cette perte, le réseau améliore la précision de la catégorisation des objets détectés, réduisant ainsi les erreurs de classification.

3.3.2.3.3 Loss d'objet

La perte liée aux objets permet au modèle de distinguer les objets du fond de l'image. Elle pénalise les fausses détections tout en renforçant la confiance des prédictions correctes. Dans YOLOv5, une BCE est appliquée sur le score de confiance de chaque BBOX, avec une cible ajustée en fonction de l'IoU. La forme générale de cette perte est donnée par :

$$\mathcal{L}_{\text{obj}} = \frac{1}{N} \sum_{i=1}^N [t_{\text{obj},i} \log(\hat{p}_i^{\text{obj}}) + (1 - t_{\text{obj},i}) \log(1 - \hat{p}_i^{\text{obj}})] \quad (3.14)$$

avec N le nombre total d'ancres, $\hat{p}_i^{\text{obj}}$ la probabilité prédite par le modèle qu'un objet soit présent à la position i , et $t_{\text{obj},i}$ la cible ajustée, définie comme :

$$t_{\text{obj}} = (1 - gr) + gr \cdot \text{IoU} \quad (3.15)$$

Le paramètre $gr \in [0, 1]$ est un coefficient de pondération qui ajuste l'influence de l'IoU sur la cible de confiance. Cette formulation permet d'introduire une pénalisation plus progressive, atténuant l'impact des prédictions proches de la vérité terrain, ce qui contribue à renforcer la stabilité de l'apprentissage.

Cette fonction de perte incite le réseau à ne prédire des objets que lorsqu'il a un niveau de certitude suffisant, réduisant ainsi le nombre de fausses détections et améliorant la robustesse globale du détecteur.

3.4 Détecteur DETR basé sur les ViT

En 2020, Facebook AI Research a introduit DETR [Carion *et al.*, 2020], une nouvelle approche pour la détection basée sur les Transformers [Vaswani *et al.*, 2017], inspirée des modèles à succès en Traitement du Langage Naturel (TLN) comme *Bidirectional Encoder Representations from Transformers (BERT)* [Devlin *et al.*, 2019] et *Generative Pre-trained Transformer (GPT)* [Radford *et al.*, 2018].

Leur efficacité repose sur le mécanisme d'auto-attention (*self-attention*), dérivé du mécanisme d'attention [Bahdanau *et al.*, 2014], permettant de modéliser les dépendances globales au sein d'une séquence, indépendamment des positions relatives. Cette capacité à capturer les relations globales a motivé leur utilisation en vision par ordinateur, où les convolutions des CNN se limitent à l'extraction de caractéristiques locales.

Cela a conduit à l'introduction des Transformers dans les tâches visuelles, notamment avec *Vision Transformer (ViT)* [Dosovitskiy *et al.*, 2020] ou *Swin Transformer* [Liu *et al.*, 2021], capables d'exploiter l'auto-attention pour modéliser les relations contextuelles et spatiales entre les régions d'une image. Le modèle ViT a montré qu'une architecture basée sur les Transformers pouvait atteindre, voire améliorer, les performances des CNN dans des tâches de classification d'images.

DETR s'inscrit dans cette continuité en étendant les Transformers à la détection d'objets. Plutôt que de produire une seule prédiction par classe, DETR encode l'image et y associe un ensemble de requêtes, permettant une prédiction directe des BBOX et des classes, sans être dépendant d'une structure hiérarchique ni avoir recours à un post-traitement tel que la NMS.

Le domaine de la détection d'objets implique des problèmes spécifiques pour assurer une convergence efficace, notamment un coût computationnel élevé, proportionnel aux dimensions de l'image d'entrée, ainsi qu'un besoin important en données.

DETR représente l'une des premières tentatives d'adaptation des Transformers à la détection d'objets. Son approche bout en bout simplifie considérablement le pipeline de détection en supprimant la nécessité des composants traditionnels tels que les RPN ou les ancres. Contrairement aux modèles CNN qui segmentent le processus en plusieurs étapes (voir figure 3.15), DETR formule la détection comme un problème d'appariement entre les prédictions et les annotations cibles, optimisé via l'algorithme de *Hungarian Matching* [Henriques *et al.*, 2010].

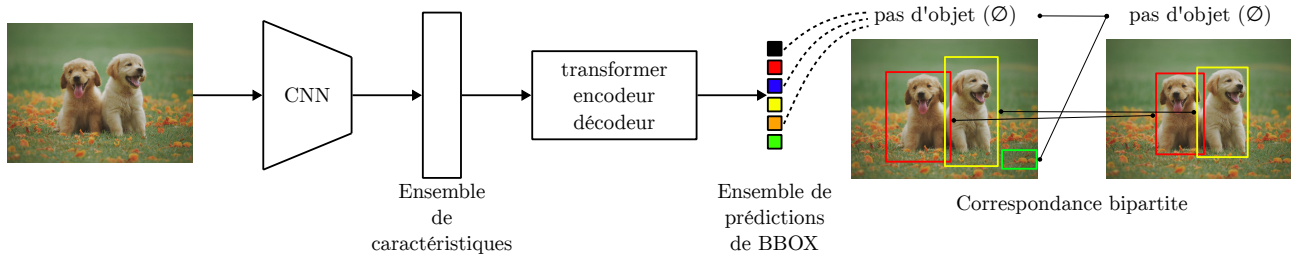


FIGURE 3.15 – Schéma simplifié du processus du DETR, inspiré de Carion *et al.* [2020].

Cette approche améliore la gestion des relations spatiales et contextuelles entre objets, ouvrant la voie à de nouveaux détecteurs basés sur les Transformers.

Pour mieux comprendre cette approche novatrice, la section suivante examinera en détail le mécanisme l'auto-attention des Transformers ainsi que son architecture pour l'adaptation pour la détection d'objets.

3.4.1 Mécanisme d'auto-attention

Les Transformers se différencient des CNN avec ce mécanisme d'auto-attention. Contrairement aux convolutions, qui capturent uniquement l'information locale dans une image à travers un filtre fixe, l'auto-attention permet d'établir des relations entre différentes régions, quelle que soit leur distance spatiale. Cela est particulièrement utile en détection d'objets, où il est important de prendre en compte le contexte global d'une scène pour améliorer la reconnaissance et la localisation des objets.

Le mécanisme d'auto-attention repose sur trois éléments clés : les requêtes (*queries*) Q , les clés (*keys*) K et les valeurs (*values*) V . Chaque élément d'une image peut interagir avec tous les autres en calculant des poids d'attention, définissant ainsi son importance relative par rapport aux autres régions. Ce processus est capturé dans une matrice d'attention, qui ajuste dynamiquement l'importance de chaque région d'entrée en fonction de l'image complète. L'auto-attention est définie par l'équation suivante :

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_K}} \right) V \quad (3.16)$$

où d_K est la dimension des vecteurs clés, généralement définie pour correspondre à celle des vecteurs de requête et de valeur. Le softmax est appliqué sur chaque ligne de la matrice de scores pour produire une distribution de poids.

Ce processus permet d'attribuer à chaque région une pondération basée sur sa similarité avec les autres éléments de l'image, offrant ainsi une meilleure compréhension des relations contextuelles.

Un avantage de l'auto-attention est sa capacité à apprendre des dépendances à longue portée, contrairement aux CNN qui nécessitent plusieurs couches pour capturer des relations globales. Cependant, cette flexibilité a un coût : le calcul des poids d'attention est quadratique en fonction de la taille de l'image d'entrée, ce qui est problématique pour des images haute résolution. Pour atténuer cet aspect, diverses variantes comme l'attention déformable [Zhu *et al.*, 2020] ou l'attention hiérarchique [Yang *et al.*, 2016] ont été développées pour optimiser les performances computationnelles sans compromettre l'efficacité du modèle.

3.4.2 Architecture DETR

L'innovation de DETR repose sur l'intégration d'un Transformer et la reformulation de la détection d'objets comme un problème d'appariement entre les prédictions et les objets annotés dans l'image.

Deux éléments distinguent DETR des autres détecteurs classiques. D'une part, la suppression du RPN en prédisant directement un ensemble de BBOX, sans passer par une phase de génération de propositions. D'autre part, l'optimisation par appariement global, grâce à l'algorithme "Hungarian Matching" (développé dans la partie 3.4.3), qui associe chaque prédiction à un objet réel de manière optimale, simplifiant ainsi la phase d'entraînement.

L'architecture globale du DETR est illustrée dans la figure 3.16. Les différentes composantes du modèle seront détaillées ensuite.

Comme le montre la figure 3.15, le modèle commence par un *backbone* composé d'un CNN, chargé d'extraire des cartes de caractéristiques à partir de l'image d'entrée. Cette représentation est ensuite aplatie en une séquence et enrichie par des encodages positionnels afin de préserver les informations spatiales tout au long du processus.

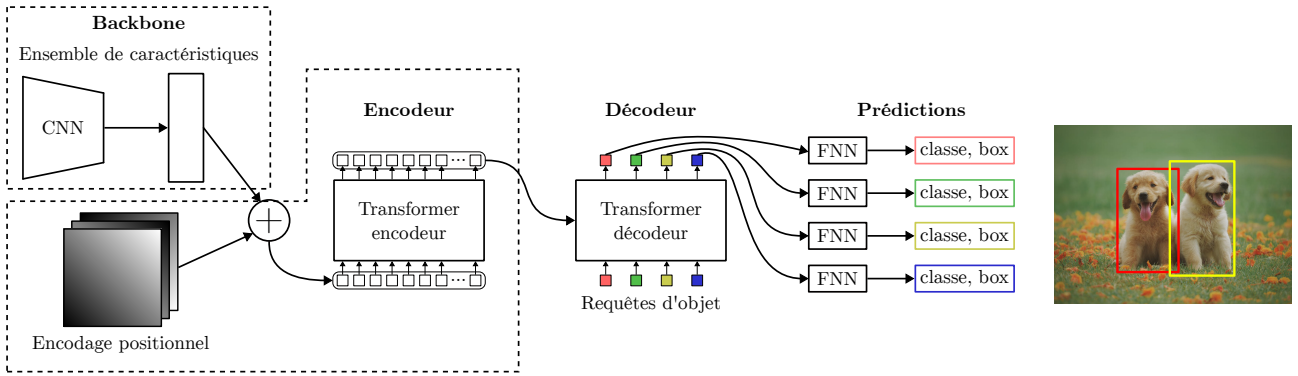


FIGURE 3.16 – Architecture du DETR, inspiré de Carion et al. [2020].

Contrairement aux séquences textuelles où l'ordre est explicite, les images possèdent une organisation spatiale implicite, qui se perd lorsque les cartes des caractéristiques sont converties en séquence. Pour préserver cette organisation spatiale, DETR ajoute à chaque vecteur de caractéristiques un “*embedding positionnel*”. Ces *embeddings*, qu'ils soient sinusoïdaux ou appris, permettent au modèle d'encoder explicitement la position spatiale de chaque élément, rendant ainsi possible la modélisation des relations spatiales au sein de l'image via le mécanisme d'auto-attention.

Le cœur du modèle repose sur l'utilisation du Transformer, présent dans l'encodeur et dans le décodeur, où il applique les mécanismes d'auto-attention pour capturer les interactions entre les différentes régions de l'image.

L'encodeur reçoit les caractéristiques extraites par le CNN et applique plusieurs couches d'auto-attention multi-têtes, permettant à chaque élément de la séquence d'interagir avec l'ensemble des autres. Ce processus améliore la prise en compte des relations à longue distance au sein de l'image.

Le décodeur reçoit en entrée un ensemble de requêtes d'objet, qui sont des vecteurs appris (et non dérivés de l'image). Chaque requête est conçue pour capturer un objet potentiel dans l'image, sans pour autant posséder de structure spatiale explicite. Elles jouent ainsi le rôle de “*slots*” que le modèle remplit en fonction des informations extraites de l'image.

Lors du passage dans le décodeur, chaque requête interagit avec la séquence encodée via l'attention croisée, apprenant à se focaliser sur les régions de l'image les plus pertinentes. À la sortie, chaque requête produit une prédiction composée d'une classe (ou “pas d'objet”) et d'une boîte englobante.

Le dernier module du réseau concerne les prédictions et traite les sorties du décodeur. Chaque sortie est traitée indépendamment par deux branches distinctes : l'une pour la classification et l'autre pour la régression des BBOX.

La branche de classification attribue un score de confiance à la catégorie de l'objet détecté, permettant d'évaluer la probabilité d'appartenance à une classe spécifique.

La branche de régression fournit les coordonnées de la BBOX, exprimées en ratios normalisés par rapport à l'image d'entrée, garantissant l'adaptabilité du modèle à des images de tailles variables.

3.4.3 Fonction de perte et optimisation

Comme dans tous les réseaux d'apprentissage profond, la fonction de perte est un élément essentiel pour obtenir un entraînement efficace et une bonne convergence des prédictions. DETR

se différencie une nouvelle fois des méthodes classiques en n'utilisant pas de méthode heuristique mais une approche basée sur l'appariement optimal entre les prédictions et les objets réels.

L'une des particularités de DETR est qu'il prédit directement un ensemble fixe de BBOX et doit associer chaque boîte prédite à une vérité terrain unique. Cet appariement est réalisé via l'algorithme *Hungarian Match* [Henriques *et al.*, 2010], qui vise à minimiser une fonction de coût définissant la similarité entre chaque prédiction et les objets annotés.

L'algorithme recherche une correspondance optimale entre les BBOX prédites et les vérités terrain en minimisant un coût défini par l'équation suivante :

$$\mathcal{L}_{\text{Hungarian}}(y, \hat{y}) = \sum_{i=1}^N \left[-\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \mathbb{1}_{\{c_i \neq \emptyset\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}(i)}) \right] \quad (3.17)$$

$$\hat{\sigma} = \underset{\sigma \in \mathfrak{S}_N}{\operatorname{argmin}} \sum_i^N \mathcal{L}_{\text{match}}(y_i, \hat{y}_{\sigma(i)}) \quad (3.18)$$

où σ représente la permutation optimale entre les prédictions $\hat{y}$ et les vérités terrain y (équation 3.18). $\hat{p}_{\hat{\sigma}(i)}(c_i)$ est la probabilité prédite pour la classe c_i , utilisée pour la loss de classification. $\mathcal{L}_{\text{box}}$ est la loss de régression des boîtes, appliquée uniquement si l'indicateur $\mathbb{1}_{\{c_i \neq \emptyset\}}$ valide si l'objet c_i est présent dans l'image ($c_i \neq \emptyset$).

Concernant la fonction de perte liée aux BBOX $\mathcal{L}_{\text{box}}$, elle permet d'évaluer la qualité des coordonnées. Contrairement à de nombreux détecteurs qui prédisent les BBOX comme un décalage (Δ) par rapport à des suppositions initiales, DETR génère directement leurs coordonnées.

$$\mathcal{L}_{\text{box}} = \lambda_{\text{iou}} \mathcal{L}_{\text{iou}}(b_i, \hat{b}_{\sigma(i)}) + \lambda_{\ell_1} \|b_i - \hat{b}_{\sigma(i)}\|_1 \quad (3.19)$$

avec λ_{iou} et λ_{ℓ_1} des facteurs de pondération pour ajuster les poids des pertes IoU et ℓ_1 . $\|b_i - \hat{b}_{\sigma(i)}\|_1$ est la perte ℓ_1 qui calcule la différence absolue entre les coordonnées de la BBOX prédite et la vérité terrain.

Bien que cette approche simplifie la mise en œuvre, elle introduit un problème lié à l'échelle des pertes. La perte ℓ_1 , couramment utilisée, présente des échelles différentes pour les petites et grandes BBOX, même si leurs erreurs relatives sont similaires. Pour atténuer cet effet, une combinaison linéaire est employée, associant la perte ℓ_1 et la perte IoU généralisée [Rezatofghi *et al.*, 2019] $\mathcal{L}_{\text{iou}}(\cdot, \cdot)$ qui est insensible aux variations d'échelle.

L'équation (3.17) permet de calculer une perte globale qui peut être utilisée pour entraîner le modèle DETR, en tenant compte à la fois de la classification des objets et de la précision des BBOX.

Deuxième partie

Apprentissage actif dans le cadre de la détection d'objets

Chapitre 4

Apprentissage actif

Sommaire

4.1	Introduction	64
4.2	Stratégies d'apprentissage actif	65
4.2.1	Incertitude	65
4.2.2	Diversité	66
4.2.3	Hybride	67
4.2.4	Autre	67
4.3	Méthodes pour la détection d'objets	68
4.3.1	Incertitude	69
4.3.1.1	MC-Dropout	70
4.3.1.2	Deep Ensembles	72
4.3.1.3	Evidential Deep Learning	74
4.3.2	Diversité	75
4.3.3	Hybride	78

4.1 Introduction

L'apprentissage actif, *Active Learning (AL)*, est une stratégie d'apprentissage supervisé conçue pour maximiser l'efficacité des modèles tout en réduisant le nombre d'exemples annotés nécessaires pour atteindre de bonnes performances. Plutôt que de sélectionner des échantillons annotés de manière aléatoire pour l'apprentissage d'un modèle, cette approche permet de sélectionner activement les exemples les plus pertinents, en se concentrant sur les données les plus incertaines ou les plus représentatives.

Cette méthodologie est particulièrement pertinente lorsque l'annotation est coûteuse ou complexe, comme en imagerie sonar. Dans ce domaine, l'interprétation visuelle peut être particulièrement incertaine pour un humain et l'annotation repose souvent sur des informations complémentaires, comme des balises géo-localisées ou des repères visuels tels que des câbles.

La figure 4.1 illustre le fonctionnement global de l'AL.

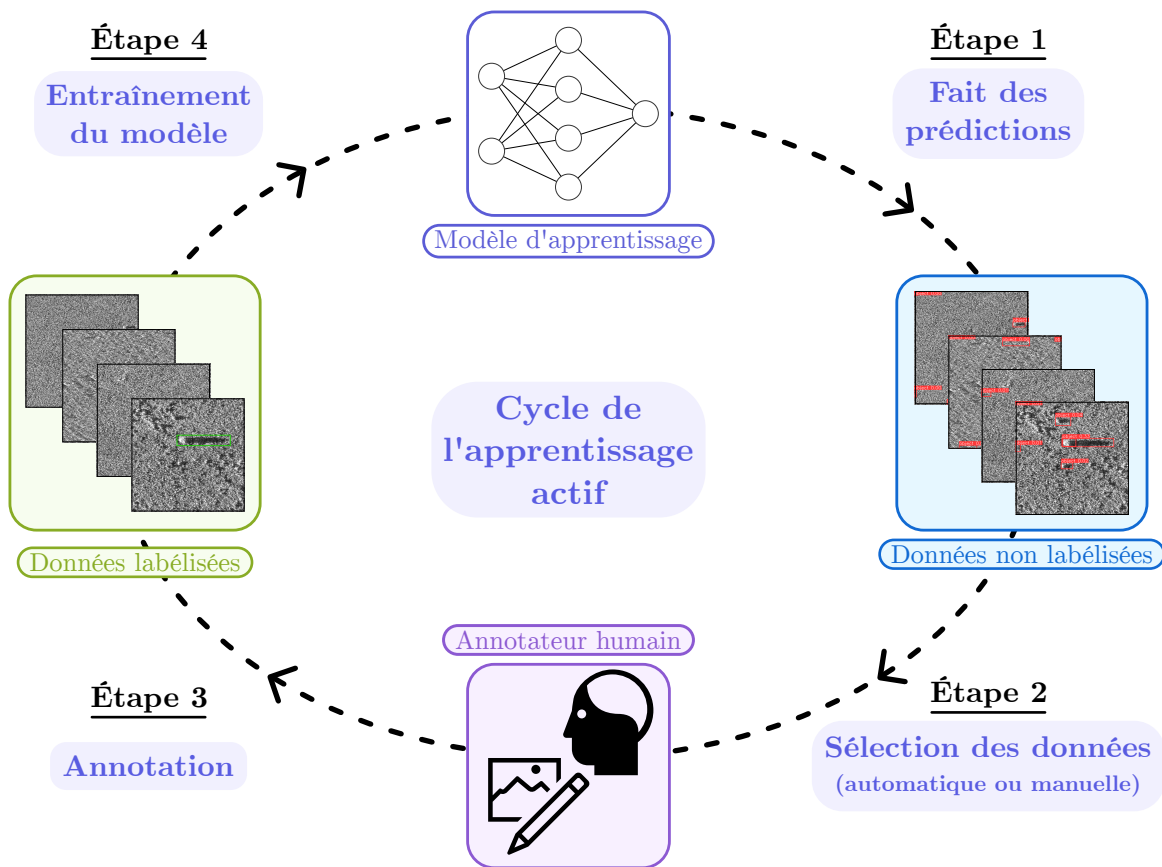


FIGURE 4.1 – Schéma de fonctionnement de l'apprentissage actif. Inspiré de Hvilshøj [2023].

Le processus d'AL débute par la génération de prédictions (étape 1) à partir d'un modèle appliqué à un ensemble non labellisé. Une sélection d'échantillons est ensuite réalisée (étape 2), de manière automatique ou manuelle. Les données retenues sont soumises à un oracle, généralement un annotateur humain, pour être labellisées (étape 3). Une fois annotées, elles sont intégrées au jeu d'apprentissage pour ré-entraîner le modèle initial (étape 4), dans le but d'améliorer ses performances.

En sélectionnant les exemples les plus informatifs, l'AL réduit les coûts d'annotation (temps humain) et accélère la convergence des modèles, optimisant ainsi l'utilisation des ressources

d'apprentissage. Une analyse approfondie permet de structurer les stratégies existantes et d'établir une taxonomie claire des méthodes, notamment l'échantillonnage basé sur l'incertitude, la sélection par diversité et les approches hybrides.

Dans ce chapitre, nous passons en revue l'état de l'art. Nous comparons différentes approches tout en mettant en avant leurs avantages et leurs limites.

La prochaine section présentera les principales catégories d'AL existantes, en expliquant leurs principes fondamentaux. Elle sera suivie d'une section dédiée aux détails de ces différentes méthodes, illustrée par des exemples issus des travaux de recherche récents.

4.2 Stratégies d'apprentissage actif

L'apprentissage actif est une approche efficace pour optimiser la qualité de l'apprentissage en exploitant au mieux les données. Cette section présente les critères d'évaluation permettant de comparer les différentes méthodes d'apprentissage actif, ainsi que les principales stratégies utilisées dans le cadre de la détection d'objet.

L'AL s'appuie sur plusieurs critères de sélection d'échantillons, généralement regroupés en trois catégories : incertitude, diversité et approches hybrides combinant les deux.

4.2.1 Incertitude

Les méthodes basées sur l'incertitude se concentrent sur l'identification des échantillons pour lesquels le modèle est le moins certain [Wang *et al.*, 2016]. Parmi les approches classiques, la stratégie de la moindre confiance (*Least Confidence*, *LC*) sélectionne les échantillons pour lesquels le modèle a le plus faible niveau de confiance (i.e, valeur de LC élevée), définie par :

$$LC(x_i) = 1 - \max_{c \in C} P(c|x_i) \quad (4.1)$$

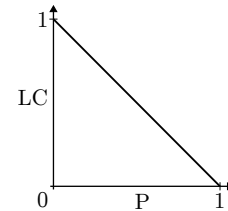


FIGURE 4.2 – Illustration du score de moindre confiance.

où x_i est un échantillon, $P(c|x_i)$ représente la probabilité prédite pour la classe c et C est l'ensemble des classes possibles. La figure 4.2 schématise la courbe de cette stratégie.

Cette stratégie est utilisée dans divers domaines [Vercruyssen *et al.*, 2018], notamment en classification, mais elle a également été adaptée récemment à la détection d'objets. Dans ce cadre, l'évaluation de la confiance est réalisée directement sur la classe prédite de chaque BBOX [Beluch *et al.*, 2018 ; Chitta *et al.*, 2019 ; Yoo et Kweon, 2019 ; Qu *et al.*, 2020 ; Yuan *et al.*, 2021 ; Choi *et al.*, 2021 ; Yu *et al.*, 2022 ; Chen *et al.*, 2023].

Une autre approche couramment utilisée repose sur le calcul de l'entropie des prédictions, initialement étudiée par Cohn *et al.* [1994] et Lewis et Gale [1994], qui évalue le degré de dispersion de la distribution de probabilité des classes [Wu *et al.*, 2022], grâce à l'équation suivante :

$$H(x_i) = - \sum_{c \in C} P(c|x_i) \log P(c|x_i). \quad (4.2)$$

Les exemples présentant la plus forte entropie sont alors sélectionnés, correspondant à des prédictions réparties de manière plus uniforme entre les classes, ce qui traduit une incertitude élevée.

Des études récentes ont démontré l’efficacité de cette méthode dans le cadre de l’AL. Par exemple, Wu *et al.* [2022] intègrent l’entropie au sein de la NMS en introduisant la méthode “*Entropy-based Non-Maximum Suppression (ENMS)*”, une variante permettant d’estimer l’incertitude des prédictions de manière plus robuste. De leur côté, Li *et al.* [2021] proposent un algorithme basé sur l’entropie visant à réduire les coûts d’annotation tout en tenant compte du déséquilibre des classes, améliorant ainsi la représentativité des échantillons sélectionnés. Plus récemment, Ghita *et al.* [2024] appliquent cette approche pour la détection d’objets 3D dans leur logiciel “*ActiveAnno3D*”, intégrant des mécanismes d’apprentissage actif pour optimiser le processus d’annotation. De même, Greer *et al.* [2024] utilisent cette stratégie pour la détection d’objets 3D, spécifiquement dans le contexte de la conduite autonome. Enfin, Wang *et al.* [2024a] explorent une approche combinant l’entropie postérieure avec la saillance des caractéristiques locales, afin d’améliorer l’évaluation de l’incertitude dans la détection d’objets complexes.

L’échantillonnage avec marges (Margin Sampling (MS)), introduit par Tong et Koller [2000], repose sur les propriétés des SVM pour sélectionner les échantillons les plus informatifs [Campbell *et al.*, 2000 ; Schohn et Cohn, 2000 ; Nguyen et Smeulders, 2004]. Cette méthode se concentre sur l’écart entre les deux classes les plus probables [Joshi *et al.*, 2009], une mesure qui capte directement le doute du modèle sur ses prédictions :

$$M(x_i) = P(c_1|x_i) - P(c_2|x_i) \quad (4.3)$$

où $P(c_1|x_i)$ et $P(c_2|x_i)$ sont les probabilités des deux classes les plus probables, respectivement. Cette approche est particulièrement efficace pour identifier les exemples ambigus, souvent situés aux frontières de décision du modèle.

L’un des inconvénients de cette méthode est qu’elle ne permet de sélectionner qu’un seul échantillon de données à chaque itération pour interroger l’oracle, ce qui entraîne un coût computationnel élevé en raison du grand nombre d’itérations nécessaires. Malgré cette limitation, de nombreuses recherches continuent d’exploiter le MS en raison de sa simplicité d’implémentation et de son efficacité dans divers domaines d’application [Lv *et al.*, 2020 ; Jiang et Gupta, 2021 ; Xie, 2021 ; Xie *et al.*, 2022 ; Zhu *et al.*, 2024a].

Certaines méthodes ont déjà été présentées pour la détection d’objets ; d’autres seront abordées dans la section 4.3.1, en particulier celles visant à affiner l’estimation des probabilités.

4.2.2 Diversité

D’autres méthodes reposent sur la **diversité**, en sélectionnant des échantillons couvrant une large variété de cas afin d’améliorer la généralisation du modèle.

Ces approches incluent notamment les stratégies de *clustering*, qui regroupent les échantillons en fonction de leur similarité dans l’espace des caractéristiques. Un algorithme de regroupement vise à révéler des structures latentes dans les données en formant des groupes d’échantillons similaires sans utiliser d’étiquettes. L’objectif est d’aligner ces regroupements, reflétant les relations implicites entre les données, avec les catégories sémantiques recherchées [Xu *et al.*, 2010], afin d’améliorer la sélection d’échantillons utiles pour la classification.

Plusieurs méthodes s’appuient sur des mesures de distance entre représentations afin de promouvoir la diversité lors de la sélection d’échantillons.

Une première approche consiste à exploiter les distances dans les cartes de caractéristiques, en sélectionnant les échantillons les plus éloignés afin de limiter la redondance dans l’espace de représentation.

Une seconde méthode utilise des graphes. Liao *et al.* [2023] propose de modéliser chaque classe sous forme de graphe, où les nœuds représentent les échantillons et les arêtes traduisent leur similarité. Cette structure permet de capturer les relations intra et interclasses, et d’extraire des sous-ensembles représentatifs à partir de la topologie du graphe.

4.2.3 Hybride

La combinaison des approches d’incertitude et de diversité a conduit au développement de méthodes “**hybrides**”, cherchant à maximiser l’information acquise tout en garantissant une diversité suffisante dans les données sélectionnées.

La combinaison des stratégies fondées sur l’incertitude et la diversité a conduit au développement de l’algorithme BADGE [Ash *et al.*, 2019], une approche hybride en apprentissage actif. Cette méthode sélectionne les échantillons à annoter en combinant deux critères : l’incertitude, estimée à partir des gradients de perte, et la diversité, assurée par un regroupement via k-means++ [Arthur et Vassilvitskii, 2007] dans l’espace de ces gradients.

Les gradients sont calculés pour chaque échantillon non annoté en supposant une pseudo-étiquette, et mesurent la sensibilité de la fonction de perte à de faibles variations des paramètres du modèle. Ils reflètent à la fois le potentiel informatif de chaque échantillon et son impact sur l’apprentissage. Les gradients de grande amplitude correspondent généralement aux échantillons les plus incertains. Le regroupement par k-means++ permet ensuite d’assurer une répartition diversifiée des échantillons dans l’espace des gradients, garantissant la complémentarité du batch sélectionné.

Parmi les approches hybrides, celle de He *et al.* [2014b] combine explicitement incertitude, représentativité et diversité pour sélectionner les échantillons à annoter. La méthode s’appuie sur une stratégie en deux étapes : identification des échantillons non étiquetés les plus incertains (via la marge entre les deux classes les plus probables), suivie d’une mesure de représentativité fondée sur la distance aux centres de clusters formés à partir des données non étiquetées.

L’aspect hybride résulte de l’articulation entre ces critères : l’incertitude cible les exemples informatifs, tandis que la représentativité assure leur pertinence vis-à-vis de la distribution des données. Une fonction de pondération combine ces deux scores pour orienter la sélection. La diversité est également prise en compte de manière implicite par le processus de clustering, qui favorise une couverture étendue de l’espace des données.

4.2.4 Autre

Une autre approche intéressante est la méthode **Query By Committee (QBC)**, qui repose sur un ensemble de modèles (“comité”) votant pour identifier les échantillons les plus incertains. Un échantillon est jugé informatif lorsque les prédictions des modèles varient fortement, indiquant un manque de consensus. Une approche repose sur l’entropie des votes [Dagan et Engelson, 1995], qui évalue le désaccord entre les comités en mesurant l’entropie de leurs prédictions, grâce à l’équation suivante :

$$\text{VE}(x) = \sum_i^C \frac{v_i}{k} \log \frac{v_i}{k} \quad (4.4)$$

avec v_i le nombre de votes pour la classe i et k le nombre total de comités, la probabilité de vote pour i s'écrit $\frac{v_i}{k}$.

Toujours dans le type de méthode QBC, une autre approche est d'utiliser la divergence de Kullback-Leibler (KL) [Kullback et Leibler, 1951]. La divergence KL mesure la quantité d'information perdue lorsque Q est utilisée pour approximer P , deux distributions de probabilité, cette différence est régie par :

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)} \quad (4.5)$$

[Hino et Eguchi, 2022] utilise la divergence KL pour un comité de M modèles, l'équation (4.5) devient alors :

$$D_{KL}^{\text{comite}}(x) = \frac{1}{M} \sum_{j=1}^M D_{KL}(P_j(y|x)||\bar{P}(y|x)) \quad (4.6)$$

où $P_j(y|x)$ est la distribution de probabilité du modèle j et $\bar{P}(y|x) = \frac{1}{M} \sum_{j=1}^M P_j(y|x)$ est la moyenne des distributions sur l'ensemble des modèles du comité.

4.3 Méthodes pour la détection d'objets

La section précédente était consacrée aux méthodes de classification, nous abordons à présent la détection d'objets. Contrairement à la classification, qui consiste à attribuer une étiquette à une image entière, la détection requiert à la fois l'identification et la localisation de plusieurs objets au sein d'une même image. Les premières tentatives [Agarwal *et al.*, 2020; Sener et Savarese, 2017; Yoo et Kweon, 2019; Yu *et al.*, 2022; Yuan *et al.*, 2021] se sont principalement appuyées sur des approches issues de la classification. Toutefois, en raison des spécificités de la tâche de détection, ces premières méthodes se sont heurtées à des limites, nécessitant des stratégies d'apprentissage adaptées, qui seront explorées dans les sections suivantes.

Lorsqu'une seule détection est présente dans une image, un score peut lui être directement associé à l'aide des approches présentées en classification (section 7.8b). En revanche, en présence de plusieurs objets détectés, une agrégation des scores est requise, pouvant être réalisée selon différentes stratégies.

Les agrégations possibles comprennent le calcul de la somme, de la moyenne ou du maximum de toutes les détections. Lorsqu'il n'y a aucune détection dans une image, un traitement particulier est appliqué.

L'utilisation de la somme favorise le choix des images à donner à l'annotateur contenant de nombreuses détections. Lors de l'agrégation des détections via une somme, les images ne contenant aucune détection devraient être évaluées à zéro. Attribuer une valeur nulle aux exemples vides, dans une optique de minimisation de la métrique, garantit qu'ils ne seront sélectionnés qu'en absence d'exemples plus informatifs (présentant des détections incertaines) ou lorsque le modèle atteint un niveau de confiance suffisant pour y détecter des objets.

La valeur agrégée grâce à la somme, pour une seule image I , $A_{sum}(I)$ se base sur les détections D_I , elle est calculée de la manière suivante :

$$A_{sum}(I) = \sum_{d \in D_I} v(d) \quad (4.7)$$

avec $v(d)$ étant le score individuel de chaque détection d .

Une autre méthode d'agrégation est la moyenne. Cette méthode équilibre à la fois le nombre et les scores des détections au sein d'un exemple, en favorisant ceux dont les scores moyens sont élevés lorsqu'un critère de maximisation est utilisé. Pour les exemples vides, l'attribution d'une valeur moyenne de zéro reste logique.

La valeur moyenne agrégée, d'une seule image I , $A_{mean}(I)$ est calculée comme ceci :

$$A_{mean}(I) = \frac{1}{|D_I|} \sum_{d \in D_I} v(d). \quad (4.8)$$

La troisième méthode abordée ici est celle de l'agrégation en calculant le maximum. Cette approche évalue chaque exemple en se basant uniquement sur la détection ayant la valeur la plus élevée. Cela peut entraîner une perte d'informations, mais cela peut aussi être bénéfique car elle est plus robuste au bruit provenant de nombreuses détections peu significatives. Tout comme les précédentes méthodes, un score de zéro pour les exemples vides est aussi valable. Le maximum n'est pas affecté par les détections avec un score nul, car le score d'aucune détection ne peut être inférieur à zéro.

$$A_{max}(I) = \max_{d \in D_I} v(d). \quad (4.9)$$

Ces différentes stratégies d'agrégation permettent de calculer une mesure d'incertitude unique pour chaque image, facilitant ainsi la priorisation efficace des données à annoter.

4.3.1 Incertitude

DivProto [Wu *et al.*, 2022] a changé la méthodologie de choix des données en remplaçant les scores de détection par l'incertitude du modèle au sein de la NMS.

Les méthodes exploitant l'incertitude des modèles, abordées dans la section 4.2, doivent être légèrement adaptées pour qu'elles puissent fonctionner dans le domaine de la détection d'objets. Lors de l'adaptation des méthodes d'incertitude basées sur la classification à la détection d'objets, l'introduction d'un seuil devient nécessaire en raison des différences inhérentes entre ces deux tâches [Li et Sethi, 2006 ; Feng *et al.*, 2017].

La détection d'objets produit de multiples prédictions pour chaque image, chacune étant caractérisée par : les coordonnées de la BBOX, un score de confiance ou d'objectivité indiquant la probabilité de contenir un objet, et les probabilités de classe pour les catégories potentielles. Une proportion importante de ces prédictions est généralement associée à des régions d'arrière-plan, correspondant à des valeurs de confiance ou à des scores d'objectivité faibles. Par conséquent, l'application directe des mesures d'incertitude à toutes les prédictions sans différenciation conduit à l'inclusion de détections d'arrière-plan non pertinentes, ce qui risque de surcharger la mesure d'incertitude et de diminuer l'efficacité de l'apprentissage actif.

L'utilisation d'un seuil, en particulier sur le score d'objectivité ou de confiance, permet de résoudre ce problème. En fixant un seuil minimal d'objectivité, on filtre les prédictions de fond probables et peu fiables, ce qui permet de réduire le bruit et de concentrer le calcul de l'incertitude exclusivement sur les détections pertinentes. Le seuil ainsi que l'agrégation de l'incertitude reflète l'incertitude réelle associée aux prédictions d'objets réels plutôt que le bruit de fond.

En outre, le seuil améliore la qualité et l'efficacité du processus de sélection de l'apprentissage actif. Sans lui, les mesures d'incertitude pourraient à tort donner la priorité à des images

comportant de nombreuses prédictions peu fiables qui, en dépit d’une incertitude élevée, ne contiennent pas de données utiles ou informatives. L’introduction d’un seuil d’objectivité garantit que les images sélectionnées sur la base de l’incertitude agrégée sont réellement informatives, ce qui optimise les efforts d’annotation et accélère l’amélioration des modèles.

Il existe de nombreux articles utilisant de l’incertitude au sein de leurs méthodes [Wang *et al.*, 2018 ; Chen *et al.*, 2023 ; Zhang *et al.*, 2024b] pour effectuer de la détection d’objets. Dans les prochaines sections, nous allons recenser les méthodes classiques et ensuite les méthodes plus récentes permettant de faire de la détection d’objets.

Plusieurs approches probabilistes récentes ont été proposées pour améliorer l’estimation des probabilités de sortie des réseaux de neurones dans un contexte d’apprentissage actif appliqué à la détection d’objets. Parmi elles, le “MC-Dropout” [Gal et Ghahramani, 2015] introduit une approximation bayésienne simple consistant à activer le mécanisme de *dropout* [Srivastava *et al.*, 2014] à la fois durant l’entraînement et lors de l’inférence. En réalisant plusieurs passes stochastiques sur un même échantillon, le modèle génère une distribution de prédictions permettant d’estimer des statistiques telles que la moyenne ou l’écart-type. Cette méthode ne redéfinit pas directement une fonction de score, mais fournit des probabilités calibrées plus fiables, sur lesquelles peuvent être appliquées des scores d’incertitude classiques tels que la moindre confiance ou l’entropie. Elle permet en particulier de quantifier l’incertitude épistémique, liée au manque d’exemples similaires dans les données d’entraînement, détectée par la variabilité des prédictions stochastiques.

Une autre proposition récente est l’utilisation des “*Deep Ensembles*” [Lakshminarayanan *et al.*, 2016], où plusieurs modèles, entraînés indépendamment, permettent d’estimer l’incertitude à partir de la variance de leurs prédictions. Enfin, une approche particulièrement prometteuse, “*Evidential Deep Learning*” [Sensoy *et al.*, 2018 ; Park *et al.*, 2023 ; Pandey *et al.*, 2025], modélise directement l’incertitude grâce à des distributions évidentielles, offrant ainsi une estimation explicite et robuste de l’incertitude totale.

4.3.1.1 MC-Dropout

Le Monte Carlo Dropout (MC-Dropout), initialement proposé par Gal et Ghahramani [2015], est une méthode permettant d’obtenir une estimation bayésienne rapide de l’incertitude épistémique des prédictions des réseaux neuronaux. Cette méthode repose sur l’activation du *dropout* non seulement pendant l’entraînement, mais également lors de l’inférence. Alors que le *dropout* est normalement utilisé pour limiter le surapprentissage en désactivant aléatoirement certains neurones pendant l’entraînement, son activation pendant l’inférence permet d’échantillonner plusieurs configurations du modèle à partir d’un seul réseau entraîné.

Lors de l’inférence avec le MC-Dropout, plusieurs passes sont effectuées sur une même image en maintenant le *dropout* activé, générant ainsi plusieurs prédictions légèrement différentes [Gal *et al.*, 2017]. Ces prédictions multiples permettent ensuite de calculer une moyenne, utilisée comme prédiction finale du modèle, ainsi qu’une variance, constituant une estimation directe de l’incertitude épistémique.

Formellement, pour une entrée x , l’incertitude $U(x)$ estimée avec MC-Dropout correspond à la variance empirique des prédictions générées par le modèle lors de T inférences :

$$U(x) = \frac{1}{T} \sum_{t=1}^T (f(x, W_t) - \bar{f}(x))^2 \quad (4.10)$$

où $f(x, W_t)$ désigne la prédiction du modèle avec les poids échantillonnés W_t à la t -ième inférence, et $\bar{f}(x)$ représente la moyenne des prédictions sur les T échantillons.

Dans le contexte spécifique de la détection d’objets, cette variance doit être calculées non seulement sur les scores de classification, mais également sur les coordonnées des BBOX. L’incertitude estimée reflète ainsi à la fois l’identification des objets et la précision de leur localisation spatiale.

Afin d’illustrer concrètement l’application pratique et les implications du MC-Dropout en apprentissage actif pour la détection d’objets, nous présentons ci-dessous plusieurs études récentes ayant mis en œuvre cette méthode.

Choi *et al.* [2021] proposent une méthode visant à réduire significativement les coûts d’annotation en détection d’objets, en combinant apprentissage actif et estimation de l’incertitude. Leur approche repose sur l’intégration du MC-Dropout dans l’architecture “*Single Stage multibox Detector (SSD)*” [Liu *et al.*, 2015], en activant le *dropout* dans certaines couches spécifiques lors de l’inférence. Cette configuration permet de réaliser en moyenne 25 inférences par image, ce qui implique que le modèle génère 25 prédictions distinctes pour chaque image grâce à l’utilisation du MC-Dropout. Cette approche estime d’une manière plus précise l’incertitude, facilitant ainsi la sélection des images les plus informatives pour l’annotation et, par conséquent, la réduction des coûts associés.

À partir de ces prédictions multiples, une mesure d’incertitude est calculée à l’aide de l’entropie moyenne des distributions de classes. Cette stratégie permet une réduction d’environ 30% du nombre d’annotations nécessaires, sans dégradation significative des performances.

Cependant, deux limites sont identifiées. D’une part, le coût computationnel accru dû aux inférences répétées. D’autre part, une forte sensibilité au choix des paramètres de *dropout*, nécessitant un réglage rigoureux.

Deepshikha *et al.* [2021] introduisent une variante du MC-Dropout, nommée “MC-DropBlock”, spécifiquement adaptée à l’architecture YOLO afin d’améliorer la modélisation de l’incertitude spatiale en détection d’objets. Contrairement au MC-Dropout classique, cette méthode repose sur l’application de “DropBlock” [Ghiasi *et al.*, 2018], une technique introduisant de la stochasticité par blocs dans les couches convolutionnelles, activée à la fois durant l’entraînement et l’inférence.

En réalisant plusieurs inférences sur une même image, il devient possible d’estimer de manière précise l’incertitude épistémique. Les résultats expérimentaux montrent une amélioration notable de la capacité de généralisation du modèle, ainsi qu’une estimation plus fiable de l’incertitude spatiale, comparée à celle obtenue avec le MC-Dropout standard.

Cependant, cette approche s’accompagne d’un surcoût computationnel significatif dû aux inférences multiples, tout comme le précédent article.

Plus récemment, Yi *et al.* [2024] étudient l’intégration de l’estimation de l’incertitude dans le cadre de la distillation des connaissances appliquée à la détection d’objets. Leur approche repose sur l’utilisation du MC-Dropout pour évaluer l’incertitude des prédictions produites par un modèle enseignant. Cette estimation est ensuite exploitée pour guider le transfert d’information vers un modèle étudiant, en pondérant davantage les connaissances issues des prédictions les plus fiables.

Les résultats expérimentaux, notamment sur le jeu de données COCO [Lin *et al.*, 2014], montrent une amélioration significative des performances du modèle étudiant en termes de précision moyenne (mAP).

Un réglage précis du taux de *dropout* reste important, bien que cette approche allonge le temps d’entraînement en raison des inférences multiples nécessaires à l’estimation de l’incertitude.

Le temps d'entraînement reste un goulot d'étranglement dans les apprentissages. Brach *et al.* [2020] proposent d'approximer MC-Dropout en une seule passe en utilisant la propagation des moments. Leur approche permet d'estimer directement l'espérance et la variance de la distribution prédictive, réduisant ainsi le temps de calcul tout en conservant les avantages des réseaux bayésiens. Boillet *et al.* [2022] réalisent une comparaison entre différents estimateurs d'incertitude, y compris ceux basés sur MC-Dropout. Bien que certains estimateurs puissent être plus efficaces, MC-Dropout reste une méthode performante pour améliorer les résultats avec moins de données annotées. Dans un domaine différent, Kawaguchi *et al.* [2022] appliquent l'AL avec MC-Dropout pour la classification des signatures de systèmes de détection d'intrusion (IDPS). Grâce à cette méthode, un plus grand nombre de signatures jugées pertinentes sont transférées aux experts, tout en réduisant les déséquilibres au sein des données d'entraînement, améliorant ainsi les performances globales du modèle.

Ces exemples montrent que, malgré ses bénéfices majeurs en matière de réduction d'annotations et d'amélioration de la robustesse des modèles face aux variations de données, l'utilisation du MC-Dropout est systématiquement limitée par une augmentation significative du coût computationnel et une dépendance forte au paramétrage de la méthode (nombre d'inférences, taux de *dropout*...).

Afin de contourner certaines limites du MC-Dropout, d'autres méthodes probabilistes récentes, comme les *Deep Ensembles* et l'*Evidential Deep Learning*, proposent des approches alternatives que nous détaillons dans les sous-sections suivantes.

4.3.1.2 Deep Ensembles

Les *Deep Ensembles*, introduits par Lakshminarayanan *et al.* [2016], est une technique d'ensemble utilisant le modèle choisi en MC-Dropout.

Tout comme le MC-Dropout, l'incertitude globale est alors estimée à partir de la variance des prédictions. Contrairement au MC-Dropout qui exploite la variabilité induite par les inférences stochastiques d'un seul modèle, les *Deep Ensembles* tirent parti de la diversité naturelle entre les modèles.

L'équation de la variance reste similaire, mais elle s'applique ici aux prédictions issues des différents modèles de l'ensemble. Pour une image x , l'incertitude $U(x)$ est ainsi donnée par :

$$U(x) = \frac{1}{M} \sum_{m=1}^M (f_m(x) - \bar{f}(x))^2 \quad (4.11)$$

où M est le nombre total de modèles dans l'ensemble, $f_m(x)$ est la prédiction du modèle m et $\bar{f}(x)$ désigne la moyenne des prédictions fournies par l'ensemble.

Cette approche a été introduite par Lakshminarayanan *et al.* [2016] dans un cadre général, non spécifique à la détection d'objets, dans le but de fournir une estimation simple, robuste et scalable de l'incertitude prédictive dans les réseaux de neurones profonds.

Les auteurs montrent que les *Deep Ensembles* dépassent les méthodes bayésiennes approximatives comme le MC-Dropout, tant en précision (*classification accuracy*), qu'en optimisation de la fonction de coût (*negative log-likelihood*) et en calibration des probabilités, sur des jeux de données standards tels que MNIST.

Laurent *et al.* [2023] se focalisent sur la problématique majeure du coût computationnel élevé inhérent aux *Deep Ensembles* traditionnels. Leur objectif est de concevoir une méthode

plus efficiente en ressources, tout en préservant les atouts fondamentaux de ces ensembles : une haute précision, une bonne calibration des probabilités, une estimation fiable de l'incertitude et une robustesse renforcée face aux données hors distribution.

Pour réduire ce coût computationnel, les auteurs proposent une variante appelée “*Packed-Ensembles (PE)*”. Cette méthode repose sur l'entraînement d'ensembles structurés et allégés, en exploitant les *grouped convolutions* [Krizhevsky *et al.*, 2012] afin de paralléliser plusieurs modèles au sein d'un unique réseau neuronal. Concrètement, ces convolutions permettent de séparer le flux d'information entre plusieurs branches internes du réseau, chaque groupe correspondant à un sous-modèle distinct. En ajustant la largeur de l'espace de représentation (paramètre α) et le nombre de convolutions groupées (paramètre γ), chaque sous-réseau peut conserver une capacité suffisante tout en cohabitant avec d'autres dans la même architecture globale. Grâce à cette conception, chaque sous-réseau peut apprendre de manière indépendante, favorisant ainsi la diversité des prédictions tout en réduisant les coûts mémoire et de calcul comparés aux *Deep Ensembles* classiques.

Les différences entre ces architectures sont illustrées au sein de la figure suivante.

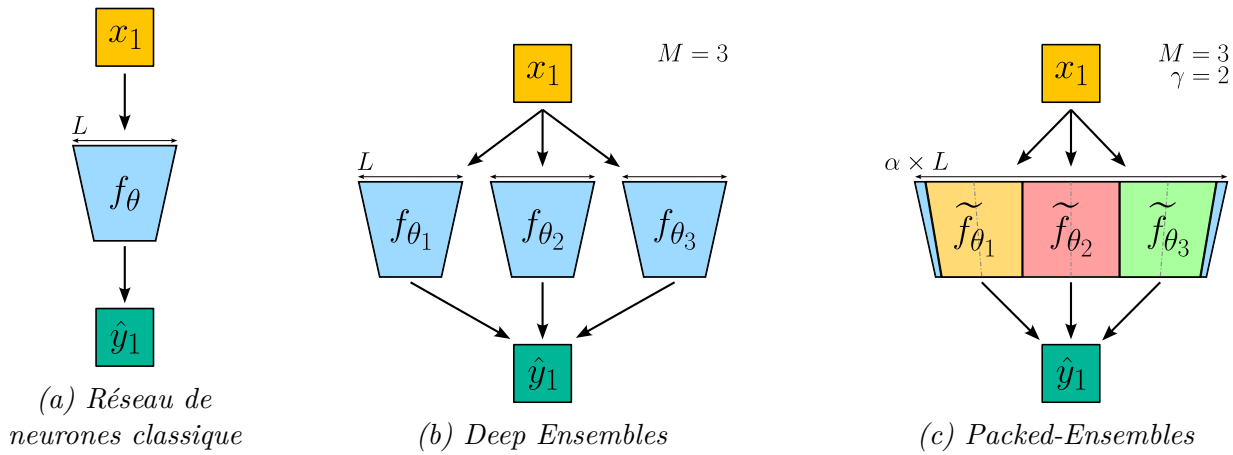


FIGURE 4.3 – Illustration des différentes architectures : réseau classique, deep ensembles, packed-ensembles

D'après leurs résultats expérimentaux, les PE constituent un compromis intéressant entre performance prédictive et efficacité computationnelle. En intégrant plusieurs sous-modèles dans une seule architecture via des convolutions groupées, cette approche permet de réduire significativement le nombre de paramètres (jusqu'à 40 % par rapport aux *Deep Ensembles*) tout en maintenant des performances comparables. L'objectif de rendre les ensembles utilisables sur des systèmes aux ressources limitées est ainsi atteint.

Initialement développée pour la classification, cette approche a été progressivement adaptée à des tâches plus complexes telles que la détection d'objets. Dans ce cadre, plusieurs travaux récents ont exploré l'utilisation des *Deep Ensembles* pour optimiser la sélection des données à annoter, en s'appuyant sur leur capacité à estimer une incertitude précise. Ce qui suit est un aperçu des contributions intégrant cette méthode dans des stratégies d'AL appliquées à la détection.

Schmidt *et al.* [2020] évaluent plusieurs stratégies d'incertitude (entropie, vote majoritaire, variation inter-modèle) pour la détection 2D et 3D, montrant qu'un ensemble de détecteurs permet de réduire jusqu'à 30% les données annotées nécessaires.

Jung *et al.* [2023] proposent une méthode fondée sur des ensembles de snapshots capturés en cours d'optimisation, offrant une bonne diversité sans surcoût d'entraînement, adaptée aux

contraintes computationnelles.

De leur côté, Choi *et al.* [2021], s'appuient sur les réseaux de densité de mélange [Yoo *et al.*, 2019], permettant une estimation de l'incertitude en une seule passe. Ils construisent un ensemble de trois modèles indépendants pour agréger efficacement les incertitudes, tout en maintenant un coût computationnel réduit.

Tang *et al.* [2020] combinent plusieurs détecteurs pour effectuer de la détection d'anomalies. En intégrant le retour d'information humain, ils parviennent à réduire le taux de faux positifs.

Dans une approche d'AL continu, la stratégie opère en flux : le modèle sélectionne les échantillons à annoter de manière itérative au fil du temps, sans repartir d'un ensemble fixe [Perkonigg *et al.*, 2021]. Kim *et al.* [2024] combinent plusieurs techniques d'extraction de données (moindre confiance, entropie, et l'algorithme ATLAS [Wang *et al.*, 2014] fondé sur le transfert d'apprentissage) afin de sélectionner des échantillons à la fois informatifs et diversifiés.

Enfin, Xu *et al.* [2020] étudient deux stratégies d'ensembles distinctes pour améliorer la précision et l'efficacité en détection. La première, l'ensemble NMS, fusionne les sorties de plusieurs détecteurs à l'aide de la NMS, combinant ainsi les résultats de classification et de localisation. La seconde, l'ensemble de caractéristiques, agit plus en amont en concaténant les cartes de caractéristiques extraites par différents modèles avant l'étape de détection, permettant une complémentarité des représentations tout en conservant une seule passe de prédiction.

Ces travaux illustrent clairement l'efficacité des *Deep Ensembles* pour fournir une estimation précise et robuste de l'incertitude épistémique, aussi bien dans des scénarios d'apprentissage actif pour réduire les coûts d'annotation que pour renforcer la robustesse face aux variations de données. Malgré ces bénéfices notables, toutes ces études confirment que l'inconvénient majeur des *Deep Ensembles* reste leur coût computationnel élevé, conséquence directe de l'entraînement de plusieurs modèles indépendants.

4.3.1.3 Evidential Deep Learning

La méthode "*Evidential Deep Learning (EDL)*" a été proposée par Sensoy *et al.* [2018] afin de quantifier l'incertitude dans les réseaux de neurones sans avoir besoin d'échantillonnage, comme dans MC-Dropout.

L'objectif principal est de prédire des classes pour des tâches de classification, de modéliser l'incertitude sur les prédictions, sans recourir à des méthodes bayésiennes coûteuses comme les échantillonnages de poids, Gao *et al.* [2024] présente les fondements et certaines avancées de l'EDL.

EDL repose sur la théorie de l'évidence de Dempster-Shafer, "*Dempster-Shafer Theory of Evidence (DST)*", qui est la généralisation de la théorie bayésienne pour les probabilités subjectives [Dempster, 1968]. Plutôt que de modéliser les probabilités d'appartenance de chaque classe sous forme d'un simple vecteur softmax, la modélisation passe par la forme d'une distribution de Dirichlet [Jsang, 2018].

La distribution de Dirichlet ($\text{Dir}(\boldsymbol{\alpha})$), définie sur des vecteurs de probabilités positifs dont la somme vaut un, est paramétrée par le vecteur $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_k)$. Sa norme reflète la confiance globale du modèle, tandis que sa répartition traduit les croyances entre classes. Cette représentation permet de modéliser une distribution de croyance plutôt qu'une estimation ponctuelle, la dispersion de cette dernière traduisant directement l'incertitude.

En sortie du réseau, le modèle ne produit pas directement des logits pour une fonction softmax, mais génère des valeurs intermédiaires z_k pour chaque classe. Celles-ci sont ensuite transformées par une fonction d'activation positive, typiquement une ReLU, afin d'assurer la positivité des

sorties. Cela conduit à un vecteur de “preuves” $e = (e_1, \dots, e_K)$, avec $e_k = \text{ReLU}(z_k)$, utilisé pour paramétrer une distribution de Dirichlet selon :

$$\alpha_k = e_k + 1 \quad (4.12)$$

Quant aux prédictions, l’estimation de la probabilité prédite pour chaque classe est l’espérance de la distribution Dirichlet :

$$\hat{p}_k = \frac{\alpha_k}{S}, \text{ avec } S = \sum_{i=1}^K \alpha_i \quad (4.13)$$

Quant à l’incertitude u , elle est calculée par rapport au nombre de classes :

$$u = \frac{K}{S} \quad (4.14)$$

Si la somme des α est grande, c’est qu’il y a beaucoup de preuves, l’incertitude est donc faible et le modèle est confiant. En revanche, si la somme est petite, u sera élevé et le modèle incertain.

Lors de l’entraînement avec cette méthode, la fonction de perte pousse le réseau à donner beaucoup de preuves pour les bonnes classes, mais à rester prudent si l’échantillon est ambigu. Cela permet de mieux détecter les éléments considérés comme hors distribution, “*Out-Of-Distribution (OOD)*”.

Cette méthodologie a inspiré des recherches visant à améliorer le domaine de la détection d’objets. Dans l’article Park *et al.* [2023], les auteurs proposent une stratégie d’AL combinant une tête d’évidence (*Model Evidence Head*) et une agrégation hiérarchique des incertitudes, permettant une sélection plus fine des exemples à annoter. Dans “EvCenterNet” [Nallapareddy *et al.*, 2023], l’EDL est utilisé pour enrichir une architecture de détection de centre d’objets avec une estimation d’incertitude, en adaptant la fonction de perte pour améliorer la fiabilité sur des entrées bruitées. Wang *et al.* [2024b] introduisent une méthode pour la détection d’objets en ensemble ouvert, où l’EDL est exploité pour favoriser la séparabilité inter-classes et renforcer la détection d’objets inconnus grâce à une régularisation spécifique.

L’intégration de l’EDL dans les CNN n’est pas très complexe. Pandey *et al.* [2025] l’ont appliquée au DETR [Carion *et al.*, 2020] via une approche bout-en-bout. Leur méthode repose sur des distributions Beta [Forbes *et al.*, 2011] pour la classification et “*Normal Inverse Gamma (NIG)*” [Amini *et al.*, 2020] pour la localisation. Cette dernière permet de modéliser conjointement la moyenne et la variance prédite d’une variable continue, ce qui la rend adaptée aux tâches de régression, telles que la prédiction des coordonnées de BBOX. La fonction de perte est également modifiée afin de prendre en compte le déséquilibre de classes et d’intégrer une régularisation basée sur la calibration du modèle.

4.3.2 Diversité

Contrairement aux approches basées sur l’incertitude qui ciblent principalement les échantillons les plus ambigus pour le modèle, les méthodes fondées sur la diversité cherchent à sélectionner un sous-ensemble d’exemples à la fois représentatifs et variés.

Leur principe repose sur l’idée que l’amélioration des performances ne dépend pas uniquement de l’annotation des cas les plus incertains, mais aussi d’une couverture équilibrée de l’espace de données. En intégrant une diversité structurelle dans les échantillons annotés, ces méthodes réduisent la redondance, assurent une meilleure exploration des distributions sous-jacentes et renforcent la capacité de généralisation du modèle.

Pour identifier des données complémentaires, ces approches s'appuient généralement sur des mesures de similarité ou de dissimilarité entre les échantillons. Ces mesures sont souvent calculées à partir des représentations internes extraites par les réseaux neuronaux, permettant ainsi de guider efficacement le processus de sélection en exploitant les caractéristiques apprises à différents niveaux de l'architecture.

Les méthodes basées sur le *clustering* visent à regrouper les données non annotées en ensembles distincts (ou *clusters*) afin de sélectionner, au sein de chacun, des échantillons représentatifs. C'est ce qui est illustré au sein de la figure 4.4 ci-après.

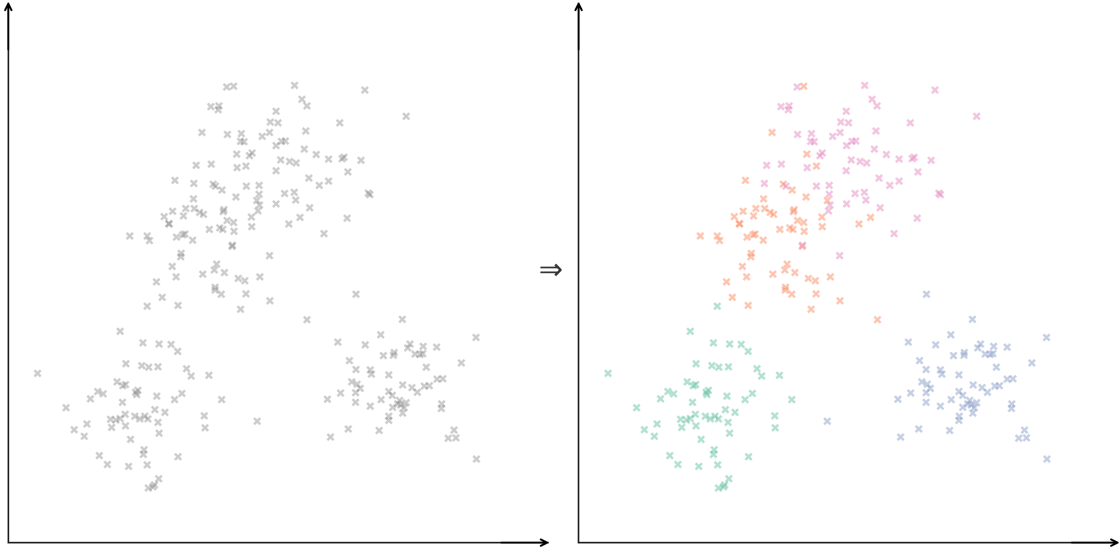


FIGURE 4.4 – Illustration du principe de regroupement par clustering dans l'espace des représentations.

Cette approche permet de maximiser la diversité en assurant une couverture équilibrée de l'espace des données. Certaines variantes intègrent l'incertitude dans le processus de *clustering* [Hwang *et al.*, 2022 ; Yang *et al.*, 2015], ces méthodes hybrides seront abordées dans la section 4.3.3. Néanmoins, les approches classiques reposent sur des algorithmes tels que le *K-means* [MacQueen, 1967] ou le *K-medoids* [Maranzana, 1963 ; Park et Jun, 2009], qui partitionnent les données en un nombre fixe de clusters tout en minimisant la distance intra-cluster. D'autres méthodes, comme le *clustering* hiérarchique [Sokal et Sneath, 1963 ; Szekely *et al.*, 2005 ; Candillier, 2006 ; Dasgupta et Hsu, 2008], construisent une structure arborescente de regroupement, permettant d'analyser les données à différentes échelles de granularité et d'adapter dynamiquement le niveau de regroupement selon les besoins.

Dans ce cadre, la sélection des échantillons à annoter s'effectue généralement en identifiant, pour chaque cluster, les points les plus proches de leurs centres. L'objectif est de choisir des instances représentatives de chaque regroupement afin d'assurer une couverture diversifiée de l'espace des données non annotées $\mathcal{U}$.

$$S = \arg \min_{x \in \mathcal{U}} d(x, c_i) \quad (4.15)$$

où c_i est le centre du *cluster* i et $d(\cdot)$ une fonction de distance ou dissimilarité, typiquement la distance euclidienne ou la distance cosinus, choisie en fonction des propriétés de l'espace de représentation utilisé.

Les approches fondées sur le *clustering* ont été adoptées en raison de leur simplicité de mise en œuvre et de leur efficacité [Nguyen et Smeulders, 2004 ; Yin *et al.*, 2023 ; Mondal et Das, 2021 ; Moses *et al.*, 2023]. Toutefois, leurs performances dépendent fortement de la qualité des regroupements produits comme le soulignent Huang *et al.* [2010]. L'introduction du *clustering* à des fins de sélection d'échantillons représentatifs remonte aux travaux de Xu *et al.* [2003], qui proposent d'utiliser des moyennes de clusters pour guider la sélection dans un cadre d'AL. Cette approche a ensuite été reprise et affinée, notamment dans les méthodes par lots, où les groupes formés sont directement intégrés dans le processus de sélection.

Certaines méthodes d'AL par lots s'appuient implicitement sur le *clustering* sans l'utiliser de manière explicite. Par exemple, Guo [2010] proposent une méthode basée sur la partition de matrice, visant à maximiser l'information mutuelle entre échantillons annotés et non annotés. Cette approche consiste à décomposer une matrice de similarité représentant les relations entre échantillons en sous-matrices associées à des groupes cohérents. L'optimisation de cette partition permet de sélectionner des échantillons représentatifs, capturant la structure globale du jeu de données. Bien qu'indépendante d'un algorithme de *clustering* au sens strict, cette méthode en reproduit les effets en structurant l'espace selon des critères d'information partagée. Dans une perspective similaire, Pinsler *et al.* [2019] introduisent une approche bayésienne nommée “*Active Bayesian CoreSets with Frank-Wolfe*” (ACS-FW), reposant également sur la maximisation de l'information mutuelle. Ce problème est formulé comme une tâche de partition de matrice, et l'objectif est de constituer des lots à la fois diversifiés et informatifs. Plus récemment, Yang *et al.* [2021] ont proposé une méthode hybride baptisée “*Active Learning using Multi-set Clustering*” (ALMC), qui combine un classificateur actif séquentiel pour évaluer l'informativité des échantillons, avec un *clustering K-means* appliqué en second lieu afin de minimiser la redondance au sein des lots tout en garantissant une couverture diversifiée de l'espace des données.

Dans Yan *et al.* [2022], la méthode ALCS (*Active Learning using Clustering-based Sampling*) est introduite. Elle repose sur un algorithme de *clustering* par densité, le FPS-clustering [Yan *et al.*, 2017], qui limite les réglages de paramètres en s'appuyant sur le partage proportionnel de fitness. Deux stratégies de sélection sont combinées : l'une privilégie les échantillons proches du centre des clusters, en fonction de leur représentativité, l'autre cible les zones de chevauchement entre clusters, considérées comme plus incertaines. Cette sélection en frontière s'appuie sur une heuristique géométrique fondée sur la loi des cosinus entre un point et les centres de deux clusters adjacents. Pour réduire la redondance, ALCS intègre une méthode de nichage, FPN [Sareni et Krahenbuhl, 1998], qui regroupe les points en niches locales définies par un rayon de voisinage. Une stratégie de partage de priorité est alors appliquée : après la sélection d'un point, la priorité des voisins proches est diminuée, favorisant une exploration diversifiée de l'espace des données.

Dans Vo *et al.* [2022], une stratégie nommée “*Box-in-Box (BiB)*” est introduite pour améliorer les performances des détecteurs d'objets faiblement supervisés. Elle repose sur l'identification automatique d'images présentant deux boîtes imbriquées (une petite à l'intérieur d'une plus grande) pour une même classe, signe fréquent de confusion chez les modèles peu entraînés (détection partielle d'un objet ou regroupement de plusieurs instances). Ces motifs BiB servent ensuite à orienter la sélection dans l'apprentissage actif : les images contenant ce type d'erreur sont retenues, puis diversifiées à l'aide d'un mécanisme inspiré de “*k-means++*”, appliqué aux paires de boîtes détectées plutôt qu'aux images.

La méthode “ALEC” proposée dans Wang *et al.* [2017] est basée sur le *clustering* par densité [Ester *et al.*, 1996], et introduit plusieurs innovations. Elle s'appuie sur la construction d'un arbre maître, permettant de modéliser les relations entre les instances afin de guider la croissance structurée de l'arbre de recrutement. À cela s'ajoute une stratégie déterministe de sélection

d'échantillons, combinant des critères de densité locale et de distance pour choisir les instances les plus représentatives. Une autre originalité d'ALEC réside dans son tri-partitionnement, qui permet de décider, à chaque itération, si une instance doit être étiquetée, conservée pour un traitement ultérieur, ou ignorée, en fonction de sa position et de son importance dans la structure de données. Cette approche intégrée de *clustering*, d'étiquetage et de classification permet d'optimiser la sélection d'exemples à annoter tout en structurant efficacement l'espace des données.

4.3.3 Hybride

Compte tenu des avantages complémentaires mais distincts des approches basées sur l'incertitude et celles fondées sur la diversité, de nombreuses méthodes dites "hybrides" ont récemment vu le jour. Celles-ci cherchant à tirer parti des deux critères simultanément, en combinant la capacité des mesures d'incertitude à cibler les échantillons les plus informatifs, avec celle des stratégies de diversité à garantir une couverture représentative de l'espace des données. Cette intégration permet d'éviter les limitations propres à chaque approche prise séparément : comme la redondance fréquente dans les sélections fondées uniquement sur l'incertitude, ou le manque de cible des zones d'ambiguïté lorsque seule la diversité est prise en compte. Ainsi, les approches hybrides offrent un compromis efficace, optimisant la sélection des échantillons à annoter pour améliorer plus rapidement la performance du modèle.

Plusieurs méthodes récentes combinent efficacement incertitude et diversité, que ce soit par l'entropie couplée au *clustering* [Wu *et al.*, 2022], l'utilisation d'ensembles neuronaux [Hernández *et al.*, 2020], ou des stratégies plus spécifiques comme BADGE [Ash *et al.*, 2019] et PPAL [Yang *et al.*, 2024], qui seront détaillées ci-dessous.

Dans leur article, Wu *et al.* [2022] présentent une approche hybride novatrice pour l'AL pour la détection d'objets, combinant des mesures d'incertitude basées sur l'entropie avec des critères de diversité. Leur méthode, appelée ENMS, commence par évaluer l'incertitude des images via l'entropie des prédictions, avant d'appliquer une NMS pour éliminer les prédictions redondantes. Pour renforcer la diversité, ils introduisent la stratégie "*Diverse Prototype (DivProto)*", qui repose sur des prototypes spécifiques à chaque classe afin de garantir une diversité intra et inter-classe. L'ensemble du processus permet de sélectionner des échantillons à la fois informatifs et variés, tout en maintenant une complexité computationnelle raisonnable.

Dans le domaine de l'imagerie médicale, Hernández *et al.* [2020] proposent une méthode destinée à prédire les infections associées aux soins de santé (HAI) en milieu hospitalier, notamment dans les Unités de Soins Intensifs (USI). Leur approche combine des techniques d'ensemble avec une stratégie de sous-échantillonnage basée sur le *clustering* pour traiter le déséquilibre des données, un défi fréquent dans les contextes médicaux, où les cas d'infection sont rares comparés aux cas non infectés.

La méthode repose sur l'utilisation de classificateurs d'ensembles, tels que les *Random forest* [Breiman, 2001], le *bagging* [Breiman, 1996] et le *boosting* [Freund *et al.*, 1996], reconnus pour leur robustesse en combinant plusieurs modèles pour améliorer la précision. Afin de contrer le déséquilibre des classes, les auteurs introduisent la technique "clustering-RUS" pour *Random UnderSampling*, qui partitionne l'espace des caractéristiques en régions selon leur niveau de déséquilibre, et n'applique le sous-échantillonnage que là où ce déséquilibre dépasse un certain seuil. Ce phénomène est illustré au travers la figure 4.5, l'image 4.5a montre les clusters originaux présentent différents niveaux de déséquilibre et sur l'image 4.5b, seuls les clusters dépassant un seuil de déséquilibre subissent un sous-échantillonnage aléatoire de la classe majoritaire. Cela permet de préserver les données informatives tout en équilibrant la distribution.

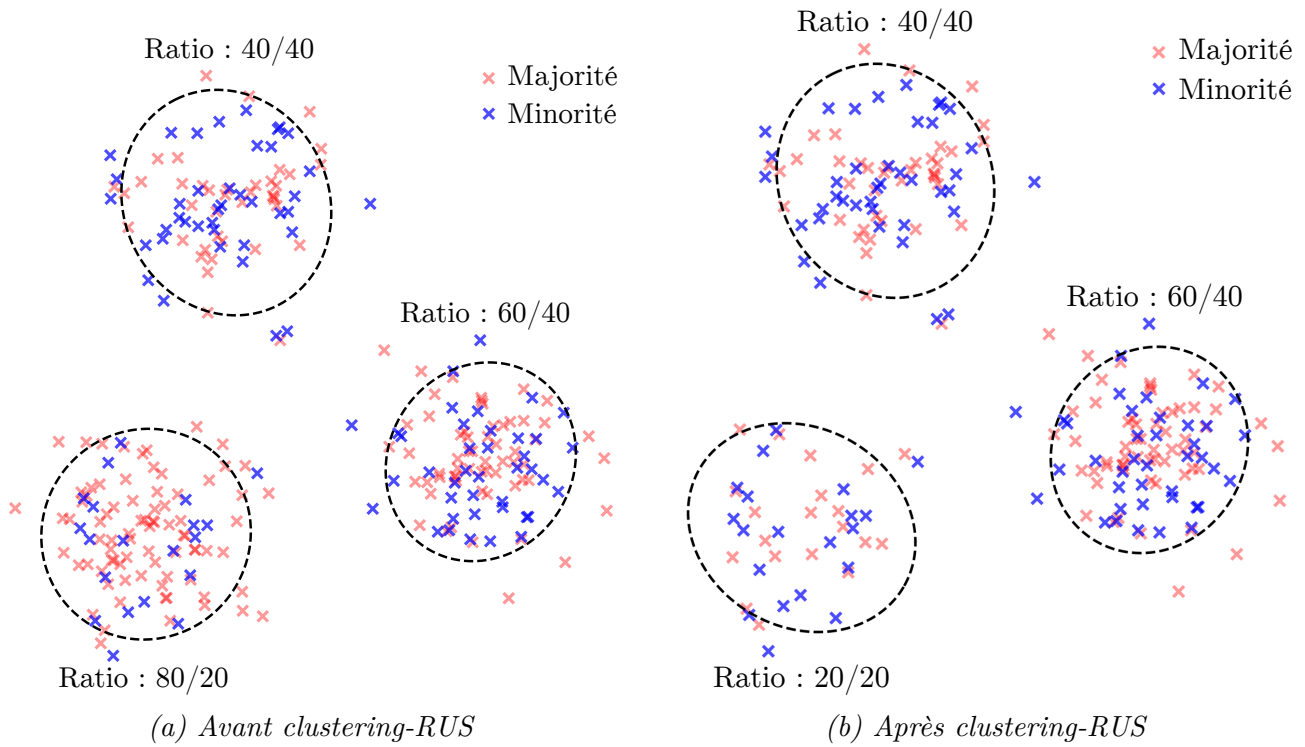


FIGURE 4.5 – Exemple de traitement des clusters déséquilibrés via Clustering-RUS.

L’approche a été validée sur des données issues de patients hospitalisés en USI, parmi lesquels seuls 6.7% ont contracté une infection. Les résultats expérimentaux montrent que la combinaison de classificateurs d’ensemble et de sous-échantillonnage ciblé permet d’améliorer significativement la précision de détection. Cette méthode offre ainsi une solution efficace pour améliorer la détection précoce des infections, contribuant potentiellement à une meilleure prise en charge et à la réduction des coûts en milieu hospitalier.

La méthode d’AL par lots, appelée “*Batch Active Learning by Diverse Gradient Embeddings (BADGE)*” [Ash *et al.*, 2019] vise à sélectionner des échantillons à annoter combinant incertitude et diversité. Pour cela, elle exploite les gradients des exemples par rapport aux paramètres de la dernière couche du réseau. La magnitude de ces gradients reflète l’incertitude de la prédiction, plus un gradient est important, plus l’exemple est informatif. Ensuite, pour garantir une sélection variée, la diversité est assurée par l’algorithme “*K-means++*” [Arthur et Vassilvitskii, 2007] sur ces vecteurs de gradients, favorisant ainsi des échantillons à la fois incertains et répartis de manière diversifiée dans l’espace des représentations.

Une méthode pionnière dans ce domaine est celle proposée par Sener et Savarese [2017] formulant l’AL comme un problème de sélection de sous-ensemble central, appelée “*Core-Set*”. L’objectif est de choisir un ensemble représentatif de points de données sur lequel entraîner un modèle capable de généraliser efficacement au reste du jeu non annoté.

Pour cela, les auteurs introduisent une borne théorique reliant la performance du sous-ensemble à la géométrie de l’espace des données, et proposent un algorithme qui minimise cette borne via une solution approximative au problème du “*K-Center*” [Hochbaum et Shmoys, 1985; Gonzalez, 1985]. Cette approche permet de couvrir efficacement l’espace des caractéristiques, assurant une bonne représentativité des données sélectionnées.

Elle se montre particulièrement efficace avec les CNN, notamment dans des contextes d’acquisition par lots, où les approches classiques basées uniquement sur l’incertitude ou la diversité peuvent perdre en efficacité à cause des corrélations entre les échantillons sélectionnés.

À l'inverse, Plug and Play Active Learning (PPAL) [Yang *et al.*, 2024] propose une approche récente et intéressante pour l'AL en détection d'objets, reposant sur une combinaison efficace d'incertitude et de diversité, structurée en deux étapes.

La première, nommée “*Difficulty Calibrated Uncertainty Sampling (DCUS)*”, ajuste les scores d'incertitude des objets à l'aide de coefficients de difficulté spécifiques à chaque catégorie, en tenant compte à la fois de la classification et de la localisation. Cela permet d'identifier un ensemble d'images candidates où le modèle est le plus incertain.

La seconde étape, appelée “*Category Conditioned Matching Similarity (CCMS)*”, évalue la similarité entre les images contenant plusieurs objets en tenant compte de leurs catégories, afin de sélectionner un sous-ensemble final d'images à la fois représentatives et complémentaires.

L'un des atouts majeurs de PPAL réside dans sa génériqueité, elle peut s'appliquer à différents détecteurs d'objets sans nécessiter de modifications architecturales ni de changements dans le pipeline d'entraînement, en faisant une solution à la fois pratique et efficace.

Ces travaux soulignent l'intérêt scientifique et pratique des stratégies hybrides en AL, combinant incertitude et diversité. En tirant parti de ces deux dimensions complémentaires, ces approches permettent une sélection plus judicieuse des données à annoter, favorisant des modèles à la fois plus performants, tout en réduisant le volume d'annotations nécessaires.

Cependant, comme l'illustrent les différentes études, ces méthodes hybrides impliquent souvent un équilibre subtil entre efficacité, coût computationnel et complexité de mise en œuvre. Ce compromis conditionne leur applicabilité dans des scénarios pratiques, notamment ceux contraints en ressources ou en temps réel.

Chapitre 5

Apprentissage actif dans le cas de la détection : une approche de sélection par clustering

Sommaire

5.1	Méthodologie	82
5.2	Définition du critère de sélection automatique	85
5.2.1	Aléatoire	85
5.2.2	Confiance des prédictions	86
5.2.2.1	Utilisation des images ayant les plus petites confiances	86
5.2.2.2	La moyenne des confiances dans chaque image	87
5.2.3	Clustering grâce aux cartes de caractéristiques	89
5.2.4	Choix de la méthode à utiliser	92
5.3	Détails des expérimentations	93
5.3.1	Protocole détaillé	93
5.3.1.1	Entraînement des références	93
5.3.1.2	Création des nouvelles bases d'entraînement	94
5.3.1.3	Finetuning sur les nouvelles bases	95
5.3.1.4	Évaluation des méthodes	96
5.3.2	Résultats	97
5.3.2.1	Comparaison des méthodes : Cluster / Moyenne des confiances	98
5.3.2.2	Comparaison des méthodes : Cluster / Aléatoire	100
5.3.2.3	Analyse des résultats	101

Le chapitre 4 a montré les différentes stratégies possibles (incertitude, diversité ou hybride) pour effectuer de l'AL. Le but est d'optimiser les performances du système tout en minimisant la quantité de données étiquetées nécessaires.

L'AL est utile lorsque les données d'entraînement sont limitées. Comme indiqué dans le chapitre 2, les données utilisées dans cette thèse sont hétérogènes et peu nombreuses. Dong *et al.* [2017] souligne la difficulté d'effectuer correctement une tâche de détection avec un nombre limité de données.

L'utilisation de l'apprentissage actif est une solution possible pour combler ce manque de données et améliorer ainsi les performances en termes de détection d'objets dans l'imagerie sonar.

5.1 Méthodologie

Dans cette thèse, l'un des principaux défis consiste à entraîner efficacement un système de détection d'objets à partir d'un nombre restreint d'annotations, tout en tenant compte des particularités des images sonar. L'apprentissage actif a été étudié comme une solution, en permettant une sélection ciblée des échantillons à annoter selon leur valeur informationnelle. Ce processus intègre naturellement un expert dans la boucle d'annotation, ce qui s'accorde avec les pratiques de la DGA, où un contrôle humain des données contribue à assurer la fiabilité du processus.

Afin de réduire le temps d'annotation de cet expert et de choisir les images les plus pertinentes, la sélection des images sera effectuée automatiquement. Cette approche assure un gain de temps en réduisant la charge de travail humaine, et permet d'obtenir une sélection des données diversifiée.

Le processus d'apprentissage actif (AL) s'appuie sur une base d'entraînement initiale et une base de test, mais il introduit une dynamique particulière : certaines images de la base de test sont sélectionnées comme étant informatives et transférées dans la base d'entraînement (à l'étape suivante). Ce transfert modifie la composition de la base de test au cours du processus, ce qui rend les comparaisons directes entre les performances de différents modèles plus complexes.

Pour structurer l'évaluation, un protocole de validation croisée à k dossiers (ici, $k = 4$) est utilisé, comme détaillé dans la section 2.3.2. À chaque itération, un des dossiers constitue la base d'entraînement (TRN), tandis que les trois autres forment la base de test (TST), conformément au schéma présenté dans la figure 5.1. Ainsi, 2436 images sont initialement utilisées pour l'apprentissage et 7308 pour l'évaluation.

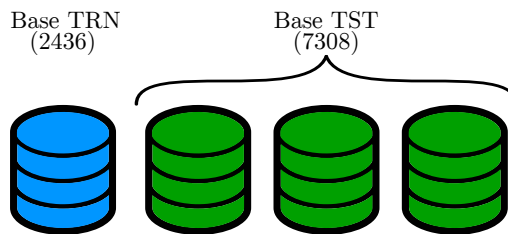


FIGURE 5.1 – Répartition des dossiers pour un entraînement, avec 1 dossier réservé pour l'entraînement en *bleu* et 3 dossiers pour l'évaluation en *vert*.

Ce choix permet de disposer d'une base de test suffisamment vaste pour mesurer les performances du modèle à chaque étape, même après extraction d'images. Toutefois, cette base étant amenée à évoluer afin de se rapprocher d'un contexte d'application réel, où les jeux de données ne sont pas statiques, les résultats doivent être interprétés avec prudence. Les comparaisons

entre différentes stratégies d’annotation reposent ainsi sur un protocole spécifique, détaillé dans la section 5.3.1.4.

Après plusieurs essais, l’utilisation d’une base pour l’entraînement (TRN) et les trois restantes pour la base de test (TST), s’est avérée la configuration la plus adaptée. Ce choix permet, d’une part, d’entraîner un modèle initial suffisamment performant, afin d’effectuer une sélection pertinente d’échantillons, et d’autre part, de conserver une quantité importante d’images non annotées pour alimenter la base d’apprentissage dans les étapes suivantes. Ce compromis garantit que plusieurs milliers d’images restent disponibles pour enrichir le modèle via l’AL, ce qui est essentiel pour observer une amélioration significative du modèle.

La distribution des images présente un fort déséquilibre, la majorité correspond à des fonds marins sans objet d’intérêt, tandis que les occurrences contenant un objet sont peu nombreuses. Cette asymétrie nécessite une stratégie de sélection rigoureuse afin d’optimiser l’apprentissage à partir d’un nombre restreint d’annotations pertinentes.

Dans le contexte de l’exploration sous-marine, l’immensité et la variabilité des fonds rendent impossible une représentation exhaustive des situations possibles. Le ratio retenu, $\frac{2436}{7308} \simeq 0.33$, vise ainsi à garantir un niveau de performance réaliste, tout en préservant une marge d’amélioration.

Enfin, puisque l’évaluation du modèle est réalisée sur les images restantes de la base TST, il faut que cette base conserve une diversité suffisante. Une base trop réduite ou trop simple fausserait les résultats en surévaluant les performances du modèle. Ce protocole a ainsi été conçu pour refléter les conditions d’un déploiement réaliste avec peu de données annotées, tout en garantissant une évaluation rigoureuse et représentative.

Une fois le jeu de données mis en place, il est possible d’entamer la procédure d’apprentissage actif qui se décompose en plusieurs étapes, illustrées dans la figure 5.2. Nous allons présenter globalement le processus qui va être utilisé, il sera approfondi en détails dans la section 5.3.

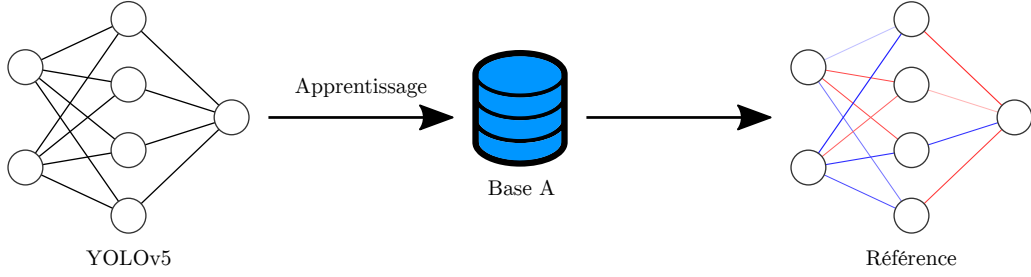
La première étape, illustrée par la figure 5.2a, consiste à entraîner un réseau YOLOv5 sur la base A, en utilisant les poids pré-entraînés fournis du modèle YOLOv5x. Le modèle ainsi obtenu, est désigné comme le modèle de “référence”, car il intègre désormais des poids adaptés la base.

Le modèle de référence étant créé, l’étape suivante est d’évaluer le modèle sur le jeu de données prévu à cet effet, la base T. Cette phase, illustrée par la figure 5.2b, est importante dans le processus d’apprentissage actif, c’est à partir des résultats obtenus sur ces données que les images à annoter seront sélectionnées. Plusieurs critères de sélection peuvent être utilisés pour extraire les images du jeu de données, leurs principes et usages seront détaillés dans la section 5.2.

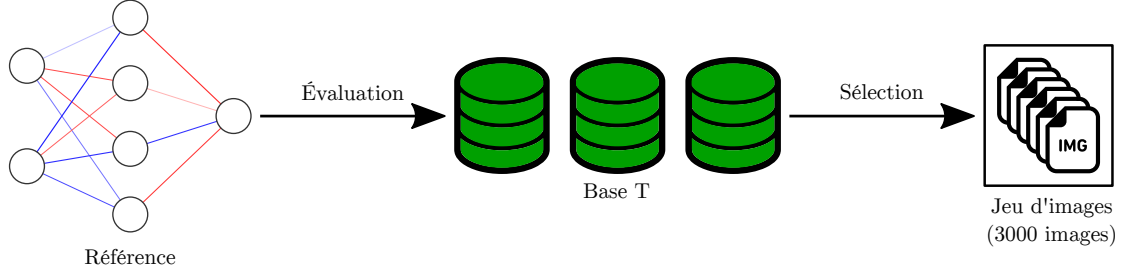
La troisième étape du processus consiste à créer des nouvelles bases d’entraînement et de test, comme illustré dans la figure 5.2c. Les images sélectionnées à l’étape précédente sont ajoutées à la base A, formant ainsi une base B enrichie. En parallèle, elles sont retirées de la base T afin d’éviter tout biais dans l’évaluation. La base U correspond alors aux images restantes de la base T après cette extraction.

L’étape suivante, représentée par la figure 5.2d, consiste à réentraîner le modèle de référence sur la base B. Pour cela, les poids précédemment appris du modèle de référence sont rechargés, puis le réseau est affiné à partir de cette configuration sur la nouvelle base enrichie.

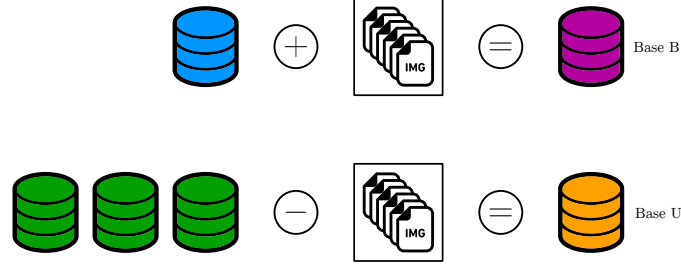
Pour terminer, la dernière étape, illustrée par la figure 5.2e, consiste à évaluer les performances finales du modèle. Le modèle affiné est testé sur la base U, et les résultats sont analysés à l’aide de différentes métriques afin de mesurer les améliorations apportées par l’ajout d’images et de l’AL.



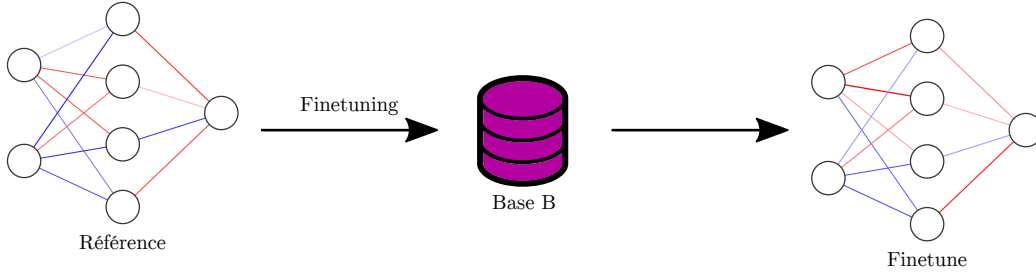
(a) Création du modèle de référence sur grâce à un réseau YOLOv5 apprenant sur la base d'entraînement.



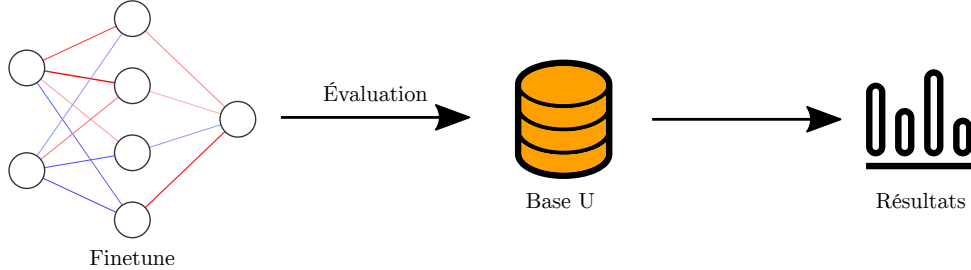
(b) Évaluation du modèle de référence sur la base de test donnant lieu à la sélection d'un jeu d'images.



(c) Création des nouvelles bases, celles d'apprentissage et d'évaluation, grâce au jeu d'images extrait lors de l'étape de sélection.



(d) Finetune du modèle de référence sur la nouvelle base d'entraînement pour obtenir le modèle ajusté.



(e) Évaluation du modèle ajusté pour obtenir les résultats sur la nouvelle base de test dont le jeu d'images a été extrait.

FIGURE 5.2 – Illustrations des différentes étapes du protocole d'apprentissage actif mis en place pour améliorer le modèle de détection dans l'imagerie sonar. Avec (5.2a) la création de la référence, (5.2b) la sélection du jeu de données, (5.2c) la création des nouvelles bases, (5.2d) le finetuning de la référence, (5.2e) l'obtention des résultats.

Dans l'apprentissage actif, il est courant de maintenir une séparation claire entre les données d'entraînement, les données de test et les données non labellisées. Tong et Koller [2000] utilisent cette approche dans leur étude sur la classification de texte, où les données non labellisées sont sélectionnées et réintégrées dans le processus d'apprentissage.

Dans le contexte de cette thèse, le scénario utilisé reflète le fonctionnement réel des tests. Des données sont acquises en mer sur lesquelles le modèle entraîné fait ses prédictions. L'objectif est d'améliorer les performances du modèle spécifiquement sur ce jeu de données, sans chercher nécessairement à généraliser à d'autres ensembles de données. C'est pourquoi les images extraites et annotées par un expert doivent provenir du même jeu de données, celui sur lequel il est évalué.

5.2 Définition du critère de sélection automatique

Il est possible d'utiliser plusieurs critères de sélection pour déterminer quelles données non labellisées seront réintégrées dans le processus d'apprentissage. La diversité des critères permet d'adapter l'apprentissage actif à des contextes variés et de d'améliorer des modèles en fonction des spécificités des jeux de données et des objectifs de l'étude. Trois critères seront étudiés dans cette thèse : la sélection aléatoire, l'utilisation des confiances des prédictions et la sélection via du clustering basé sur les cartes de caractéristiques.

Nous explorerons deux types d'approches : des méthodes non supervisées et une approche supervisée basée sur des seuils. Les méthodes non supervisées permettent une sélection entièrement automatique et sans biais, tandis que la méthode supervisée par seuils offre une alternative qui peut être plus ciblée et précise dans certains cas.

5.2.1 Aléatoire

L'une des méthodes les plus logiques qu'il est possible de mettre en place est celle de la sélection aléatoire. Cette méthode permet un gain de temps durant la phase de sélection en choisissant un certain nombre d'images pour annotation de manière aléatoire.

Il existe plusieurs manières d'effectuer un tirage aléatoire parmi un jeu de données : échantillonnage aléatoire simple [Coker et Tewfik, 2008], échantillonnage stratifié [Gallego, 2005], et échantillonnage systématique [Chan *et al.*, 2013].

Pour un tirage aléatoire, il suffit de générer des nombres avec une distribution uniforme, garantissant un tirage non biaisé. Cette méthode ne reflète pas la distribution des différentes classes de la base initiale. Ainsi, si les données ne sont pas équilibrées, l'apprentissage par le détecteur (dans la deuxième étape) sera fait sur une distribution non équilibrée.

Pour notre base (base test T, figure 5.2b en vert) le ratio global d'images contenant des objets est de 9.6%, comme indiqué dans la section 2.3.2. Ce ratio entre images avec et sans objets, sera approximativement le même pour le jeu d'image sélectionné (figure 5.2b : jeu d'images). Le tirage aléatoire étant réalisé selon une distribution uniforme, la sélection d'environ 2290 images permet de conserver un ratio proche de celui de la base complète. Un tel échantillonnage reproduit la distribution initiale avec une précision de $\pm 1\%$ au niveau de confiance de 95%.

Pour aller au-delà de la simple sélection aléatoire, il est possible d'utiliser des critères plus sophistiqués pour la sélection des images. La prochaine sous-section développe un critère basé sur la confiance des prédictions, permettant de cibler les images susceptibles d'améliorer davantage les performances du modèle.

5.2.2 Confiance des prédictions

Comparée à la méthode de sélection aléatoire, l'utilisation des confiances des prédictions nécessite une étape supplémentaire. Il faut passer en inférence l'ensemble des images de la base d'évaluation (base test T, figure 5.2b en vert) au modèle pour obtenir ses prédictions, auxquelles sont associées des scores de confiance, une position et des dimensions de BBOX, pour chaque objet. Cette approche permet de sélectionner des images en fonction de l'incertitude du modèle. L'hypothèse sous-jacente est que cette sélection permette de se concentrer sur les exemples les plus informatifs.

Deux méthodes distinctes de sélection basées sur la confiance des prédictions peuvent être envisagées. Ces approches seront détaillées dans les sections suivantes, afin de montrer comment exploiter ces informations pour optimiser le processus d'apprentissage actif.

L'objectif est de parvenir à identifier des images contenant soit des objets non détectés correctement (faux négatifs), soit des éléments tels que des rochers ou autres, à tort considérés comme des objets (faux positifs).

5.2.2.1 Utilisation des images ayant les plus petites confiances

La première méthode que nous avons évaluée consiste à trier l'entièreté des prédictions obtenues et de ne garder que les x plus petites confiances enregistrées.

En se concentrant sur les prédictions les moins certaines, il est possible d'identifier les zones où le modèle a besoin d'amélioration. La figure 5.3 montre les 10 images ayant les confiances des prédictions les plus faibles.

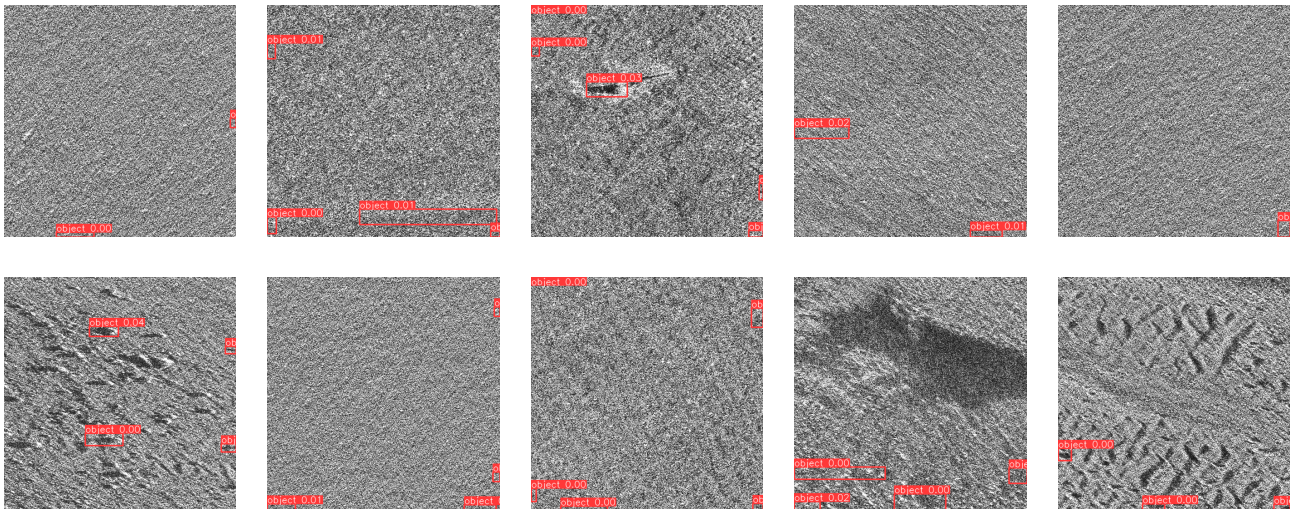


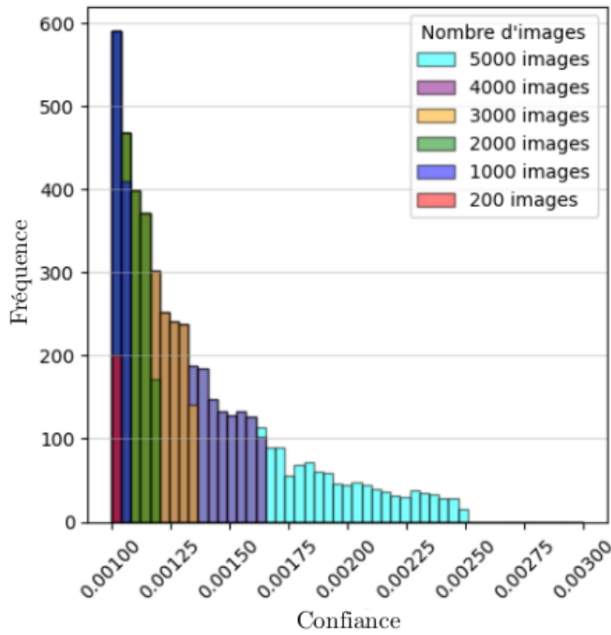
FIGURE 5.3 – Dix exemples d'images où les confiances de prédictions les plus faibles ont été obtenues lors de l'évaluation d'un modèle, l'ordre de grandeur de la confiance est de 0.1%.

Parmi tous ces exemples, la majeure partie ne contient que des bancs de sable ou bien ont quelques rides de sable, avec une seule image contenant un objet d'intérêt (première ligne,

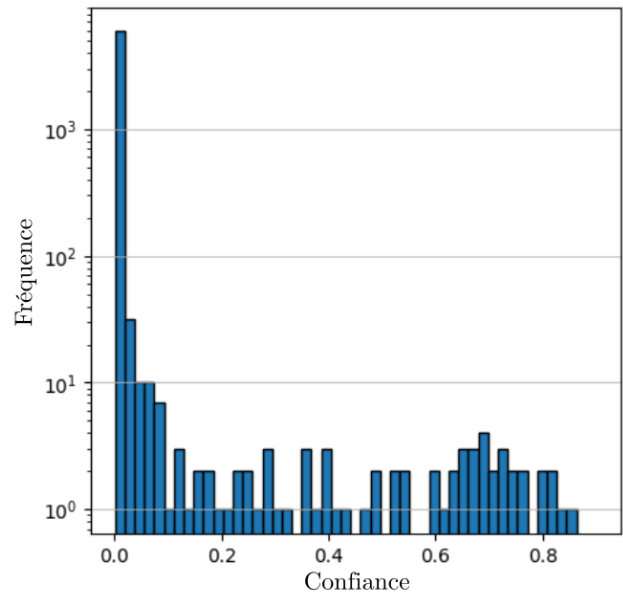
troisième colonne). La proportion d'images contenant un objet d'intérêt est inférieure à 9% en choisissant uniquement les images ayant des confiances de prédictions faibles.

La figure 5.4a illustre l'évolution du nombre d'images extraites en fonction du seuil. La majorité des images sélectionnées présentent des seuils extrêmement faibles, par exemple un seuil de 0.003 pour un ensemble de 5000 images, indiquant que le réseau attribue une très faible confiance à ses prédictions sur ces images, traduisant ainsi une forte incertitude. La concentration de la majorité des images extraites sous le seuil de 0.01 montre que le critère de sélection cible en priorité les cas les plus incertains, ce qui est cohérent avec l'objectif attendu.

Inversement, la figure 5.4b confirme que des prédictions avec des niveaux de confiance bien plus élevés (jusqu'à 0.9) existent dans les données restantes, mais qu'elles ne sont pas retenues par ce critère de sélection.



(a) Analyse du nombre d'échantillons extraits en fonction du seuil de confiance.



(b) Distribution des images en fonction de la confiance pour l'intervalle $[0, 1]$ avec une échelle logarithmique.

FIGURE 5.4 – Évolution du nombre d'images extraites en fonction du seuil avec la méthode utilisant les plus petites confiances.

Une autre méthode d'analyse consiste à regrouper les prédictions au niveau de chaque image, afin de prendre en compte la moyenne des scores par image plutôt que chaque prédiction individuellement. Cette approche sera détaillée dans la prochaine partie.

5.2.2.2 La moyenne des confiances dans chaque image

Il est fréquent que le réseau fasse plus d'une prédiction par image. C'est pour cette raison qu'il est possible de traiter les images directement en regroupant les prédictions obtenues dans chacune d'elles et en moyennant leurs confiances, permettant ainsi d'estimer la confiance du modèle sur l'image globale et non sur une partie où la prédiction est spécifique.

Par rapport aux exemples de la figure 5.3, les images extraites grâce à la moyenne des confiances sur les images n'ont aucun objet, comme le montre la figure 5.5.

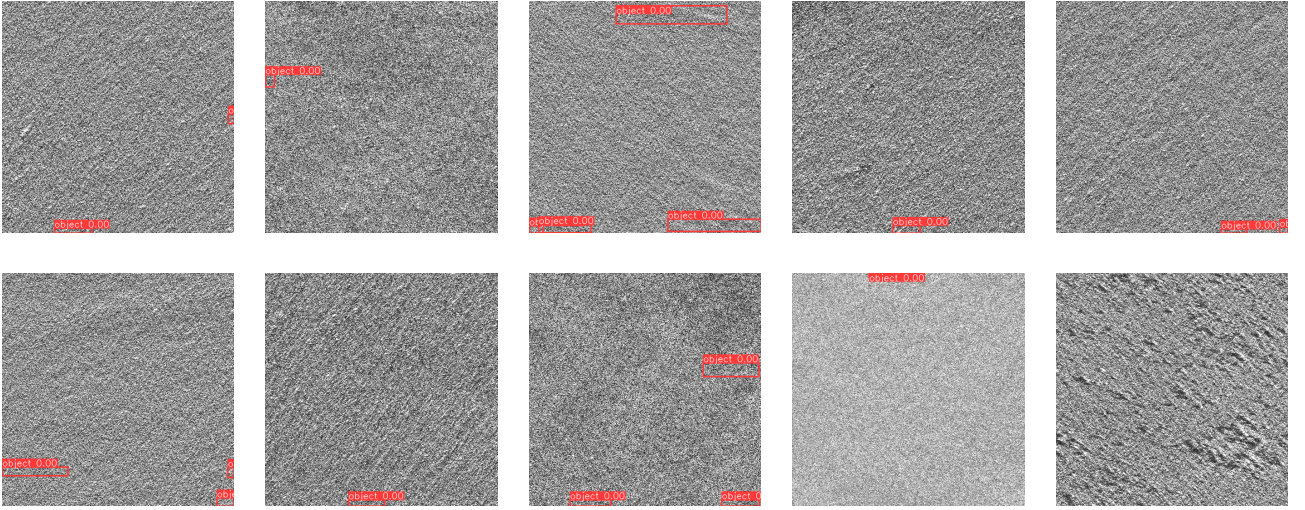


FIGURE 5.5 – Exemples d’images extraites grâce à la méthode de moyenne de confiance par image.

Néanmoins, la figure 5.6a montre une augmentation notable des confiances associées aux images sélectionnées. Alors que la méthode précédente plafonnait à une valeur maximale de 0.003 pour 5000 images, ici, ce seuil est dépassé dès 4000 images extraites, atteignant jusqu’à 0.03, soit dix fois plus.

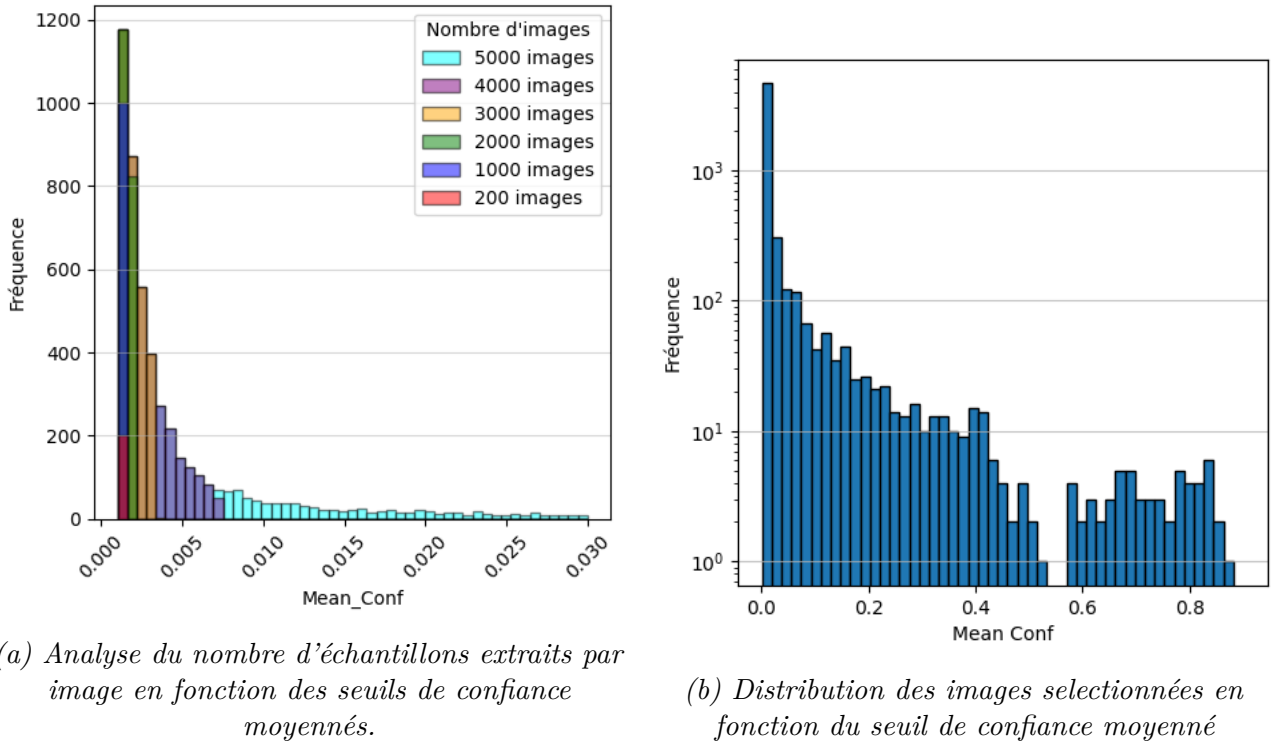


FIGURE 5.6 – Évolution des distributions du nombre d’images extraites, en fonction du seuil, avec la méthode moyennant les confiances par image.

Il est toutefois important de rappeler que la valeur de confiance affichée correspond ici à la moyenne des scores de prédiction pour chaque image. Ainsi, une image contenant plusieurs prédictions à faible confiance, mais également un objet correctement détecté avec une confiance plus élevée, verra sa moyenne globale augmentée. Ce mécanisme tend donc à lisser les scores bas et à élever artificiellement la confiance globale de certaines images, comparé à l’approche

précédente qui analysait chaque prédiction individuellement. Ce phénomène est d'ailleurs visible dans la distribution globale des scores, illustrée par la figure 5.6b.

La méthode de sélection basée sur la moyenne des confiances des prédictions se focalise sur les images où il y a un doute. En choisissant les images où le modèle a globalement moins confiance, cette méthode permet de confirmer au modèle qu'il ne doit pas prédire sur des zones de sable où les objets sont absents. Cependant, cela ne va pas nécessairement améliorer la détection des objets, car les images contenant des objets se trouvent à la fin de la liste de sélection, avec des confiances plus élevées. Le modèle est en effet plus confiant lorsqu'il détecte des objets réels, ce qui explique pourquoi ces images sont moins présentes dans les sélections avec des confiances moyennes faibles.

Pour améliorer les performances d'un réseau de manière significative, il est nécessaire de lui montrer à la fois ce qu'il ne doit pas faire et ce qu'il doit faire. Le critère par moyenne va favoriser la réduction des prédictions incorrectes, car nous allons montrer au modèle des images sans objet. Toutefois, pour que le modèle apprenne à mieux détecter les objets, il est essentiel de lui fournir davantage d'images contenant des objets. La méthode suivante, basée sur le clustering des cartes de caractéristiques, est conçue pour sélectionner de manière non supervisée plus d'images contenant des objets que les méthodes précédentes (aléatoire ou basée sur la confiance).

5.2.3 Clustering grâce aux cartes de caractéristiques

Les premières stratégies de sélection basées sur le niveau de confiance des prédictions, qu'il s'agisse du minimum ou de la moyenne, ont révélé une tendance notable à privilégier l'extraction d'images de fonds sablonneux.

Nous faisons l'hypothèse que, pour améliorer les performances du modèle, en particulier sa capacité de généralisation, il est nécessaire de lui exposer une plus grande variété de contextes visuels : fonds rocaillieux, rides sableuses, colonne d'eau, et bien entendu, images contenant des objets. Autrement dit, la diversité des scènes explorées au cours de l'apprentissage actif pourrait s'avérer plus bénéfique que l'incertitude seule.

Pour explorer cette approche, une méthode de clustering a été mise en place à partir des cartes de caractéristiques extraites par le réseau. Chaque image est projetée dans un espace de représentation via les activations d'une couche intermédiaire de YOLOv5, puis regroupée à l'aide de l'algorithme K-means, en se basant sur la distance euclidienne pour évaluer la similarité entre images.

La sélection des images à annoter s'effectue en prélevant une quantité équivalente d'images dans chaque cluster, assurant ainsi une couverture équilibrée des différents types de scènes. Cette stratégie favorise une plus grande diversité des données sélectionnées, en intégrant des contextes visuels rares ou complexes, souvent sous-représentés par les approches fondées uniquement sur l'incertitude.

Une carte de caractéristiques met en évidence les zones de l'image jugées importantes par le modèle pour la détection et la classification. La figure 5.7 montre des exemples de cartes d'une image avec des rides de sables à différentes couches du réseau. Combinée à l'image initiale, elle permet d'attribuer l'image à un cluster spécifique. Cette approche permet une sélection diversifiée d'images, tout en augmentant la probabilité d'inclure des images contenant des objets, notamment en ciblant des clusters susceptibles d'en regrouper.

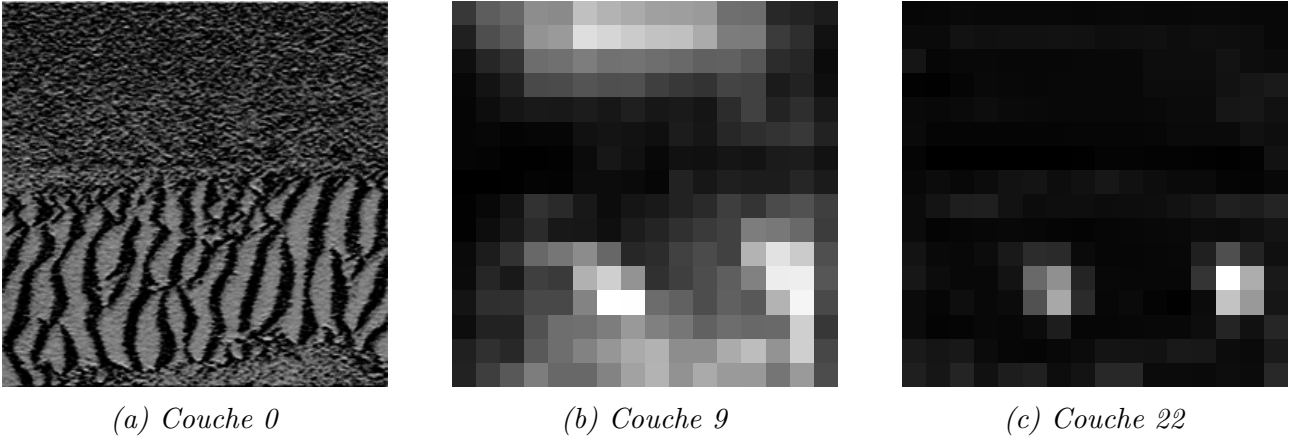


FIGURE 5.7 – Exemple de l'image de rides de sable (figure 5.8b) passée au travers du réseau où certaines cartes de caractéristiques ont été extraites.

La figure 5.8 illustre les différentes typologies d'images que le clustering permet d'isoler.

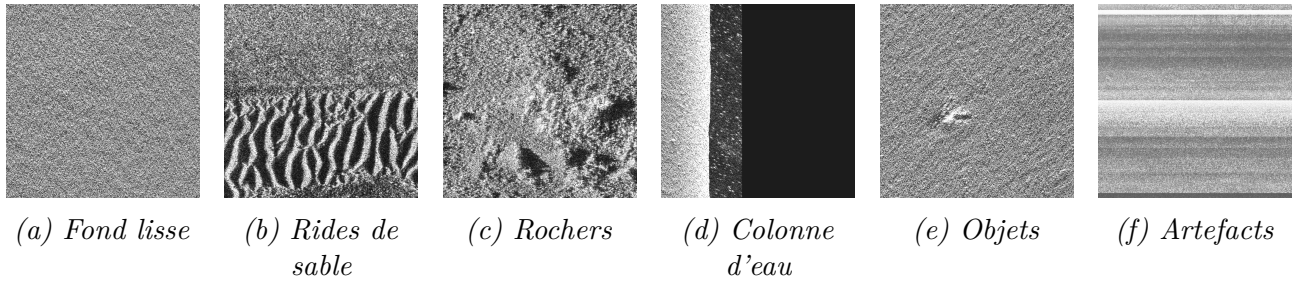


FIGURE 5.8 – Exemples représentatifs des clusters basés sur les cartes de caractéristiques.

La figure 5.9 présente une projection t-SNE des centroïdes issus des 200 clusters construits à partir des cartes de caractéristiques extraites à la couche 9 du réseau. Chaque point correspond à un cluster, avec une taille proportionnelle au nombre d'images associées. Pour faciliter la lecture, ces centroïdes ont été regroupés a posteriori en 10 groupes par clustering hiérarchique, exclusivement à des fins de visualisation, ce regroupement n'intervient pas dans le processus réel de sélection des images.

Des exemples d'images ont été ajoutés manuellement dans certaines zones afin d'illustrer les principales familles de scènes présentes dans les données. La zone centrale rassemble des clusters hétérogènes, incluant des images de rochers, de rides de sable et d'autres structures. Les clusters en bleu clair correspondent principalement à des régions contenant des objets détectés. En revanche, les extrémités de la carte sont occupées par des clusters associés à des fonds marins sans objet ou à des artefacts rares, traduisant une marginalité dans l'espace des représentations.

Dans cette méthode, deux paramètres sont fixés de manière empirique : le nombre de clusters et la couche du réseau utilisée pour l'extraction des cartes de caractéristiques.

Ces choix reposent sur une intuition : lorsque le nombre de clusters est faible, il semble préférable de s'appuyer sur des couches peu profondes, qui offrent des représentations plus génériques et globales de l'image. En revanche, lorsqu'un nombre plus élevé de clusters est utilisé, il est intéressant de s'appuyer sur des couches plus profondes du réseau, dont les activations sont plus spécialisées et sensibles à des détails visuels plus fins.

Toutefois, cette relation entre profondeur des couches et granularité du clustering n'a pas été étudiée de façon systématique. Une exploration plus approfondie serait nécessaire pour appuyer ces observations empiriques.

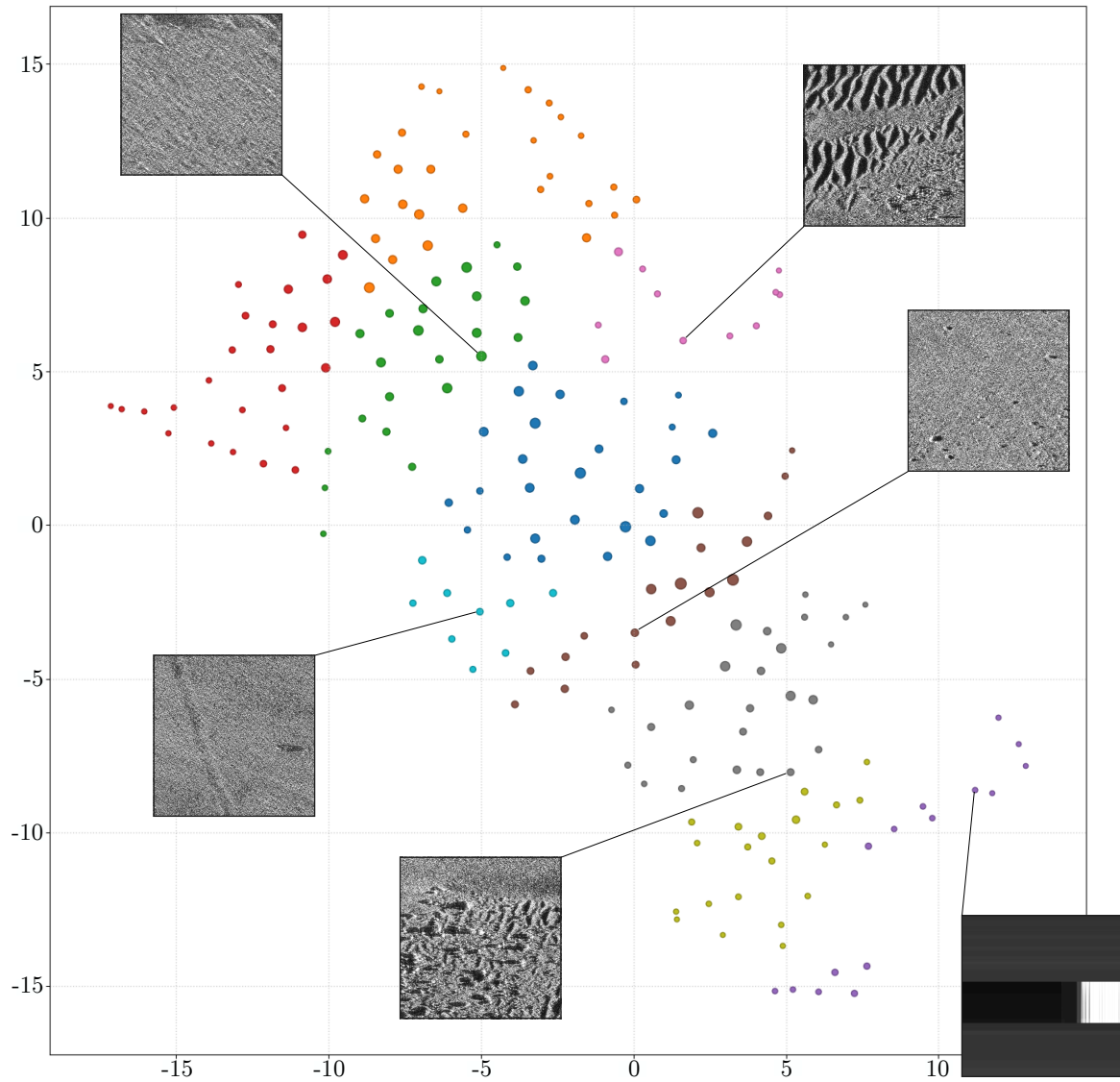


FIGURE 5.9 – Visualisation *t*-SNE des centroïdes des 200 clusters, colorés selon un regroupement hiérarchique effectué à des fins de lisibilité. Des exemples visuels ont été ajoutés manuellement pour illustrer la diversité des scènes.

Pour comparer l'impact des clusters comparé aux méthodes de sélection précédentes, nous avons fait le choix d'utiliser 200 clusters sur les cartes de caractéristiques extraites depuis la couche 9, étant la dernière couche de la partie encodeur. Les résultats obtenus sont recensés dans le tableau 5.1.

TABLE 5.1 – Tableau de l'évolution du ratio des images contenant un objet ou non dans les clusters basés sur les cartes de caractéristiques.

Nombre	Sans objets	Avec objet
200	89.5% (179)	10.5% (21)
1000	88.7% (887)	11.3% (113)
3000	87.6% (2629)	12.4% (371)
4000	88.1% (3526)	11.8% (474)
5000	89.2% (4460)	10.8% (540)
6048	90.4% (5466)	9.6% (582)

Au-delà de 4000 images sélectionnées, le ratio d'images contenant des objets tend à diminuer

et à se rapprocher de celui de la base globale. Ce phénomène s'explique par la diminution progressive des images pertinentes restantes : le nombre d'images contenant des objets devient insuffisant pour maintenir un ratio aussi élevé lors des extractions successives.

Les clusters sont constitués à partir des caractéristiques extraites par le réseau, sans contrainte minimale sur le nombre d'images par cluster. Il en résulte que certains clusters peuvent contenir très peu d'éléments.

Une fois les clusters définis, chaque image est associée à ses prédictions, qui sont ensuite triées par ordre croissant selon la confiance attribuée par le réseau.

Avec un total de 200 clusters, il devient théoriquement possible d'extraire 15 images par cluster pour obtenir une sélection de 3000 images. Dans la pratique, si un cluster contient moins de 15 images, toutes ses images sont retenues, et les images manquantes sont sélectionnées dans d'autres clusters disposant d'un surplus, de manière à atteindre le quota fixé.

Grâce à cette méthode, le ratio d'objets réinjectés dans la base d'apprentissage s'avère significativement plus élevé que celui observé avec les approches précédentes, atteignant jusqu'à 4% supplémentaires pour une extraction de 3000 images. La figure 5.10 illustre quelques exemples d'images contenant des objets, sélectionnées grâce à cette stratégie hybride.

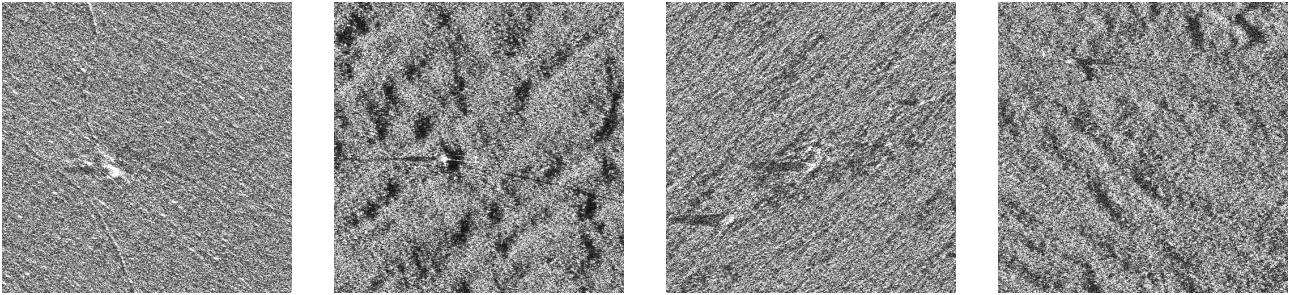


FIGURE 5.10 – Exemples d'images avec objet extraites lors de la sélection des échantillons avec notre hybridation.

Cette approche vise à analyser plus précisément l'impact des images contenant des objets sur les performances du modèle en apprentissage actif. L'évaluation des résultats permettra de déterminer si leur ajout améliore réellement la détection ou si leur contribution demeure limitée.

5.2.4 Choix de la méthode à utiliser

Nous venons de voir différentes méthodes de sélection des images, le choix de la méthodologie vise à atteindre l'objectif initial de la thèse, à savoir, améliorer les performances de détection d'objets dans l'imagerie sonar dans des conditions réelles.

La sélection aléatoire présente l'avantage d'être simple et rapide à mettre en œuvre. Elle permet de conserver une certaine représentativité de la distribution initiale, sans toutefois offrir de contrôle précis sur la proportion d'images contenant des objets, en particulier en présence d'un fort déséquilibre de classes. Cette limitation peut réduire son efficacité dans un contexte d'apprentissage actif, où l'objectif est de privilégier des exemples informatifs. La diversité des données sélectionnées et la présence d'objets annotés restent ainsi difficiles à maîtriser avec une stratégie purement aléatoire.

L'utilisation des confiances des prédictions permet de se concentrer sur les images où le modèle est le moins confiant. Cette méthode est efficace pour identifier les zones d'incertitude

du modèle et améliorer sa précision sur ces cas particuliers. Les zones où le modèle est le moins confiant sont souvent dans des images ne contenant pas d'objets. Par exemple, pour un seuil fixé à 0.5, il y a 92% de chances que l'image ne contiennent pas d'objets, ce qui entraîne une baisse du ratio d'objets réinsérés dans la base d'apprentissage. Il reste possible que certaines images contenant effectivement des objets soient sélectionnées du fait de prédictions faiblement confiantes, mais cela reste marginal. L'équilibre final des bases, en termes de diversité et de présence d'objets annotés, reste insatisfaisant, ce qui limite une fois de plus l'efficacité de cette méthode.

La méthode fondée sur la moyenne des confiances par image correspond une approche plus globale, en agréant les scores de confiance de toutes les prédictions d'une image. Cette vision synthétique permet d'estimer la confiance moyenne du modèle à l'échelle de l'image entière. Toutefois, cette approche tend à privilégier les images homogènes (souvent sans objets), tandis que les images contenant des objets, bénéficiant de prédictions plus confiantes, sont reléguées en fin de sélection. En conséquence, la proportion d'objets intégrés dans la base d'apprentissage est faible, et l'équilibre des classes reste problématique, limitant l'intérêt de cette stratégie dans un cadre actif.

Enfin, la méthode de clustering basée sur les cartes de caractéristiques regroupe les images en fonction de leurs caractéristiques visuelles. Cette méthode permet de sélectionner des images représentatives de différents types de scènes, y compris celles contenant des objets d'intérêt. Les statistiques du tableau 5.1 montrent que cette méthode sélectionne une proportion plus élevée d'images avec objets par rapport aux méthodes précédentes, tout en conservant une diversité d'images.

En conclusion, la méthode de clustering basée sur les cartes de caractéristiques apparaît comme la plus prometteuse pour l'apprentissage actif. Elle permet de maintenir un équilibre entre la sélection d'images sans objet et avec objet, tout en garantissant une diversité des données. Cette approche non supervisée assure une sélection objective et automatisée des images.

Il est pertinent de comparer cette méthode à la sélection aléatoire, qui simplifie fortement le processus tout en maintenant approximativement la répartition entre images avec et sans objets de la base initiale. En complément, la méthode de *clustering* sera également confrontée à celle basée sur la moyenne des confiances, afin d'évaluer l'impact de la présence explicite d'objets dans les échantillons sélectionnés sur les performances du modèle.

5.3 Détails des expérimentations

Dans cette partie, le protocole expérimental va être étudié en détails, comprenant ainsi la définition des bases d'apprentissage, le processus d'entraînement et les caractéristiques des machines utilisées.

Les différentes stratégies sont ensuite comparées en tenant compte des intersections entre les bases d'évaluation, en se concentrant sur l'ensemble des images communes à ces bases.

5.3.1 Protocole détaillé

5.3.1.1 Entraînement des références

La première étape est celle de l'entraînement du modèle dit de "référence" (figure 5.2a). Trois modèles YOLOv5 indépendants sont entraînés séparément, en utilisant les mêmes hyper-

paramètres : nombre d’époques, taux d’apprentissage, taille du lot d’images (ou “*batch*”). Bien qu’entraînés de manière distincte, ces modèles partagent une configuration initiale identique, avec les mêmes poids pré-entraînés fournis par Ultralytics pour le modèle de taille X (86.7 M de paramètres). Ces poids ont été obtenus à partir d’un entraînement préalable sur le jeu de données COCO [Lin *et al.*, 2014].

L’entraînement d’un modèle YOLO implique plusieurs sources d’aléatoire, telles que l’initialisation des paramètres lorsque le modèle n’est pas initialisé à partir de poids pré-entraînés, le mélange aléatoire des données ou l’application de certaines augmentations. Néanmoins, le protocole utilisé n’est pas affecté par l’initialisation aléatoire, les poids pré-entraînés d’Ultralytics étant systématiquement utilisés. Lors du chargement des données ou bien lors de la boucle d’apprentissage, il y a un tirage pseudo aléatoire de l’ordre des données. D’autres fonctions utilisées durant l’apprentissage sont sujettes à de l’aléatoire, notamment l’augmentation de données avec, par exemple, les dimensions de coupure de l’image ou bien l’angle de rotation de l’image. Lors de la boucle d’apprentissage, s’il y a des couches définies avec du “*dropout*” [Srivastava *et al.*, 2014], l’aléatoire joue aussi un rôle dans l’apprentissage de ces couches, en désactivant certains neurones aléatoirement.

Afin d’assurer une évaluation robuste, un protocole de validation croisée à quatre dossiers a été appliqué, comme détaillé en section 2.3.2. À chaque itération, un dossier sert à l’entraînement (base TRN) et les trois autres à l’évaluation (base TST), chaque dossier contenant 2436 images. La base d’entraînement comprend donc 2436 images, contre 7308 pour la base de test.

Pour chaque configuration, trois modèles indépendants sont initialisés à partir des mêmes poids pré-entraînés, mais avec des graines aléatoires différentes, afin de prendre en compte la variabilité induite par les composantes stochastiques de l’apprentissage. Au total, $4 \times 3 = 12$ entraînements sont réalisés, correspondant à 12 modèles de référence.

5.3.1.2 Création des nouvelles bases d’entraînement

En apprentissage actif, le choix des images pour être annotées se fait uniquement à partir de la base de test.

Ainsi, cinq nouvelles bases d’apprentissage seront créées. Toutes ces bases réutiliseront la base d’entraînement initiale, composée de 2436 images, à laquelle seront ajoutées respectivement 500, 1000, 2000 et 3000 nouvelles images. Ce processus est illustré dans la figure 5.11.

Dans le processus de création de la base B et de la base U (voir figure 5.2), nous décidons d’enlever systématiquement 3000 images de la base de test T (pour former la base U) quel que soit le nombre d’images à annoter par l’humain lors de l’étape d’annotation (figure 5.2b). Par contre, la base B (train pour l’étape suivante) aura plusieurs versions (B 500, B 1000, B 2000, B 3000) en fonction du nombre d’annotations effectuées lors de l’étape d’annotation, ce processus est illustré dans la figure 5.11.

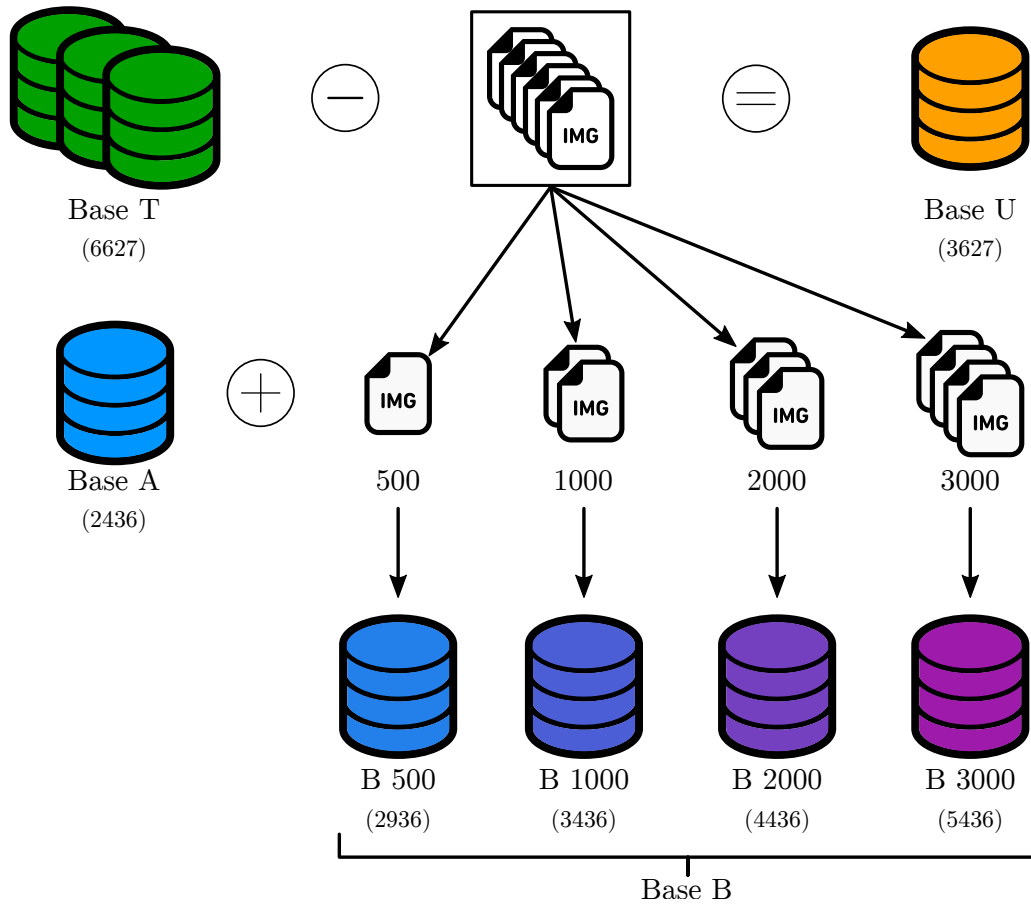


FIGURE 5.11 – Schéma explicitant la création des nouvelles bases pour l'apprentissage actif avec les images ayant été extraites de la base de TST initiale.

À noter que seules 3 000 images ont été retirées de la base test. Les différentes bases B (B 500, B 1000 et B 2000) constituent des sous-ensembles de cet ensemble de 3 000 images.

Cette extraction est réalisée pour chaque dossier de validation croisée. Un ensemble de trois modèles est associé à chaque dossier. La sélection des images s'effectue ensuite de manière indépendante pour chaque dossier, à partir des prédictions fournies par ces trois modèles. Ce protocole permet d'appliquer la phase de finetuning sur l'ensemble des modèles et des bases, comme détaillé dans la suite.

5.3.1.3 Finetuning sur les nouvelles bases

Le protocole, intégrant de multiples réseaux, nécessite un nouveau finetuning pour chacun d'eux. En restant sur le premier dossier Cross Validation (CV)¹ : les trois modèles avec des graines différentes devront apprendre sur les nouvelles bases, incluant les images extraites de la base d'évaluation.

Ainsi, il y a $3 \times 4 = 12$ nouveaux apprentissages à effectuer pour chaque CV (figure 5.12). Sur l'ensemble des dossiers de validation croisée, cela représente $12 \times 4 = 48$ apprentissages. De plus, plusieurs stratégies de sélection d'images sont testées (aléatoire, moyenne des confiances de prédictions par image, et clustering des images), portant le nombre total de tests à 144.

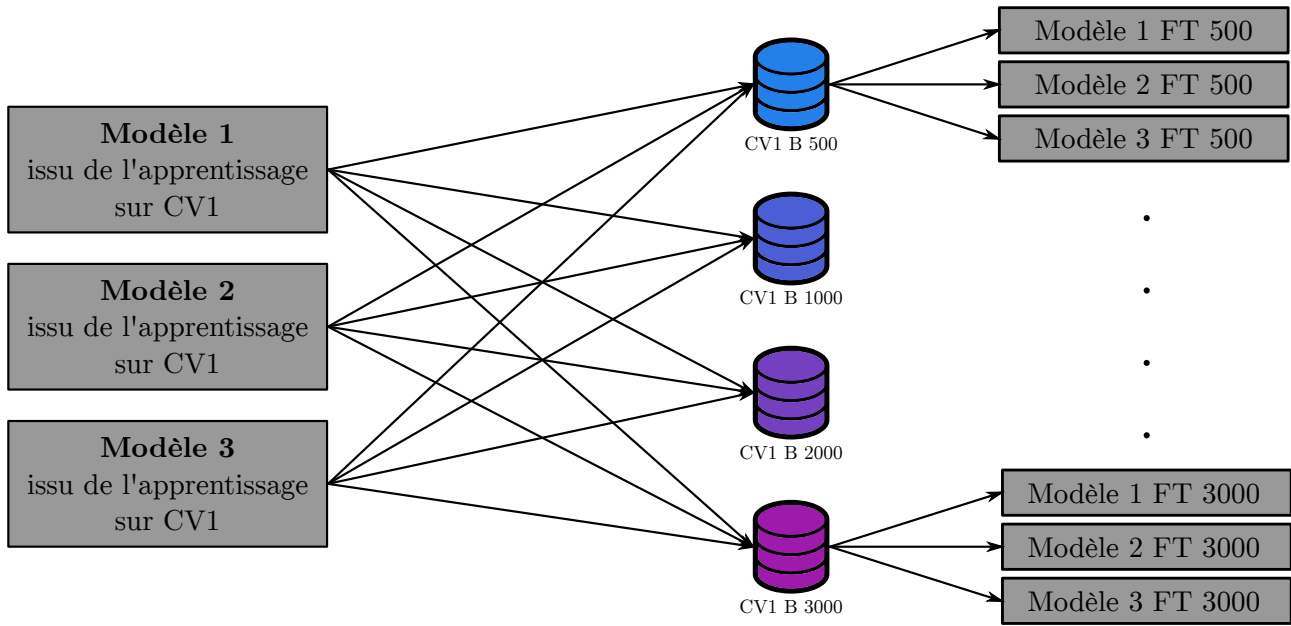


FIGURE 5.12 – Schéma montrant les 12 nouveaux apprentissages à effectuer, uniquement sur le premier dossier de la validation croisée.

Ces nouveaux apprentissages partagent la majorité des paramètres avec les entraînements des références. Cependant, ils se distinguent par les paramètres suivants : une durée de 350 époques avec une patience de 100 époques. Le paramètre de patience en apprentissage profond détermine le nombre d'époques pendant lesquelles l'entraînement peut continuer sans amélioration de ses performances sur la base de validation, avant que le processus ne soit arrêté prématurément pour éviter le sur-apprentissage.

Une fois l'ensemble des tests effectués, il est possible d'obtenir et d'analyser les résultats afin d'avoir une vision plus globale des méthodes utilisées.

5.3.1.4 Évaluation des méthodes

La dernière étape est l'évaluation (figure 5.2e) et permet d'évaluer les différents modèles affinés sur les bases d'entraînement augmentées avec des images sélectionnées par une des 3 méthodes de sélection.

Il y a donc deux aspects à analyser : l'impact du nombre d'images réinsérées dans la base d'apprentissage et l'efficacité relative des différentes méthodes de sélection pour améliorer les performances du modèle de détection.

Il est important de noter que lors de l'étape de sélection des images, les différentes méthodes utilisées (cf. section 5.2) ne sélectionnent pas les mêmes images. Par exemple, la méthode de clustering choisira des images différentes de celles sélectionnées par la méthode aléatoire. Ainsi, lors de l'évaluation, il n'est pas possible d'utiliser la base de test de la méthode aléatoire pour évaluer les modèles entraînés avec la méthode de clustering, et vice versa.

Pour comparer ces deux méthodes, il est nécessaire de prendre l'intersection des deux bases de test, afin de ne conserver que les images n'ayant jamais été réinsérées dans les nouvelles bases d'apprentissage des deux méthodes. En procédant ainsi, nous nous assurons d'éviter tout biais dans les résultats, malgré la réduction du nombre d'images disponibles pour l'évaluation.

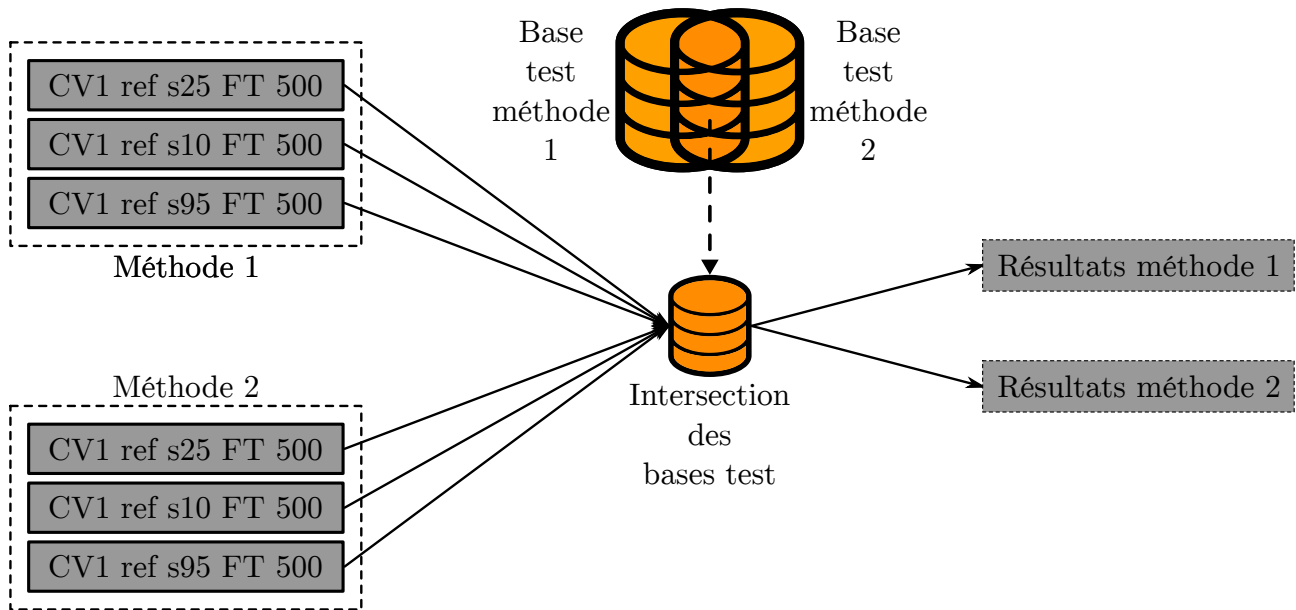


FIGURE 5.13 – Schéma du processus d'évaluation, permettant la comparaison de deux méthodes de sélection d'images, pour un dossier de validation croisée et pour 500 images réinsérées.

Tous les modèles affinés sont évalués sur l'intersection des bases de test de chaque méthode pour obtenir les prédictions finales de tous les ensembles qui seront ensuite utilisées pour établir les résultats. La figure 5.13 illustre le cas précis du premier dossier de validation croisée avec les modèles affinés sur la base d'apprentissage agrégée de 500 images.

Après avoir détaillé les différentes étapes du protocole, nous pouvons maintenant présenter les résultats obtenus pour chaque méthode.

5.3.2 Résultats

Les tableaux présentés pour les comparaisons des méthodes représentent les résultats moyennés sur les différents dossiers de validation croisée. Pour chaque méthode et chaque base d'entraînement, les tableaux affichent la moyenne des résultats ainsi que l'écart-type. La métrique utilisée est l'Area Under Curve (AUC) [Hanley et McNeil, 1982], calculée soit sur l'intégralité de la courbe (de 0 à 1), soit entre 0.5 et 1, car c'est dans cette plage que les prédictions, avec des scores de confiance plus élevés, sont les plus intéressantes.

Les résultats du modèle de référence, réentraîné sur la base initiale sans ajout d'images, sont indiqués dans les titres des tableaux. Ils permettent de vérifier si les gains de performance proviennent des images sélectionnées plutôt que d'un simple apprentissage prolongé.

Pour chaque comparaison, un tableau récapitulatif indique le nombre d'images contenant des objets réinjectées dans la base d'apprentissage pour chaque méthode. La composition de la base de test utilisée pour l'évaluation est précisée dans le titre du tableau, au format : nb img AVEC objet/nb img TOTAL.

Cette présentation facilite l'identification des méthodes sélectionnant le plus d'images pertinentes pour l'apprentissage, tout en permettant d'analyser la significativité statistique des différences observées.

5.3.2.1 Comparaison des méthodes : Cluster / Moyenne des confiances

Dans cette première comparaison, la méthode basée sur notre hybridation est comparée à la méthode avec la moyenne des confiances au sein de chaque image. Voyons tout d'abord le nombre d'images sélectionnées par ces deux méthodes.

TABLE 5.2 – Comparaison du nombre d'images contenant un objet qui sont ajoutées dans les différentes bases grâce aux méthodes de clustering et de moyenne des confiances. Le test initial sans images ajoutées a donné une composition de 130 ± 31 / 2036 ± 46 .

Méthode	500	1000	2000	3000
Moyenne Conf	3 ± 2	6 ± 3	15 ± 8	29 ± 9
Cluster	72 ± 15	145 ± 8	320 ± 26	448 ± 30

Il y a une très grande différence dès la première sélection de 500 images. La méthode avec du clustering arrive en moyenne à extraire 72 images contenant un objet contre seulement 3 pour la méthode moyennant les confiances. Cela se confirme avec l'augmentation du nombre d'images extraites, pour 3000 images, la méthode moyenne n'arrive qu'à extraire seulement une trentaine d'images contre quasiment 450 pour le clustering.

Voyons si ces images ont un impact sur les performances du modèle en sortie grâce aux résultats recensés dans les tableaux ci-après.

TABLE 5.3 – Comparaison des AUCs compris entre 0 et 1 pour les méthodes de clustering et de moyenne de confiance. La référence sans images ajoutées a donné un résultat de 0.451 ± 0.0667 .

Méthode	500	1000	2000	3000
Moyenne Conf	0.448 ± 0.084	0.441 ± 0.0651	0.421 ± 0.0402	0.441 ± 0.0905
Cluster	0.459 ± 0.0633	0.463 ± 0.065	0.51 ± 0.0528	0.567 ± 0.0489

TABLE 5.4 – Comparaison des AUCs compris entre 0.5 et 1 pour les méthodes de clustering et de moyenne de confiance. La référence sans images ajoutées a donné un résultat de 0.0905 ± 0.0316 .

Méthode	500	1000	2000	3000
Moyenne Conf	0.0883 ± 0.0345	0.0869 ± 0.0161	0.0797 ± 0.0146	0.0903 ± 0.0372
Cluster	0.0889 ± 0.0356	0.0993 ± 0.0198	0.116 ± 0.0265	0.144 ± 0.0305

Que l'AUC soit calculé sur l'intervalle $[0, 1]$ ou $[0.5, 1]$, les tendances observées dans le tableau 5.3 restent similaires. La méthode basée sur la moyenne des confiances présente une baisse de performance, probablement en raison de la sélection majoritaire d'images sans objets. Malgré l'ajout de 3000 images, les résultats obtenus restent comparables à ceux de la référence.

En revanche, la méthode de clustering améliore d'environ 11% l'AUC entre 0 et 1 et de plus de 5% lorsque l'AUC est calculé entre 0.5 et 1, comparé au modèle de référence (tableau 5.4).

En valeur absolue, un AUC de 0.567 sur l'intervalle $[0, 1]$ reste modeste, révélant une marge de progression notable. Toutefois, le gain de 11% par rapport à la référence représente une

amélioration conséquente dans un contexte de détection d'objets avec un jeu de données restreint et déséquilibré.

De même, un AUC de 0.144 sur l'intervalle $[0.5, 1]$ peut paraître faible, mais l'augmentation de plus de 5% traduit un progrès tangible sur les prédictions les plus confiantes, généralement les plus exploitables en pratique. Ainsi, bien que les performances obtenues ne soient pas optimales, les résultats confirment que la méthode de *clustering* renforce la capacité discriminante du modèle dans les conditions expérimentales considérées.

La figure 5.14 illustre les courbes ROC issues des différentes méthodes, comparées à la référence. Chaque courbe représente la moyenne des performances obtenues sur les quatre dossiers de cross-validation, tandis que les zones transparentes délimitent l'intervalle compris entre les valeurs minimales et maximales observées.

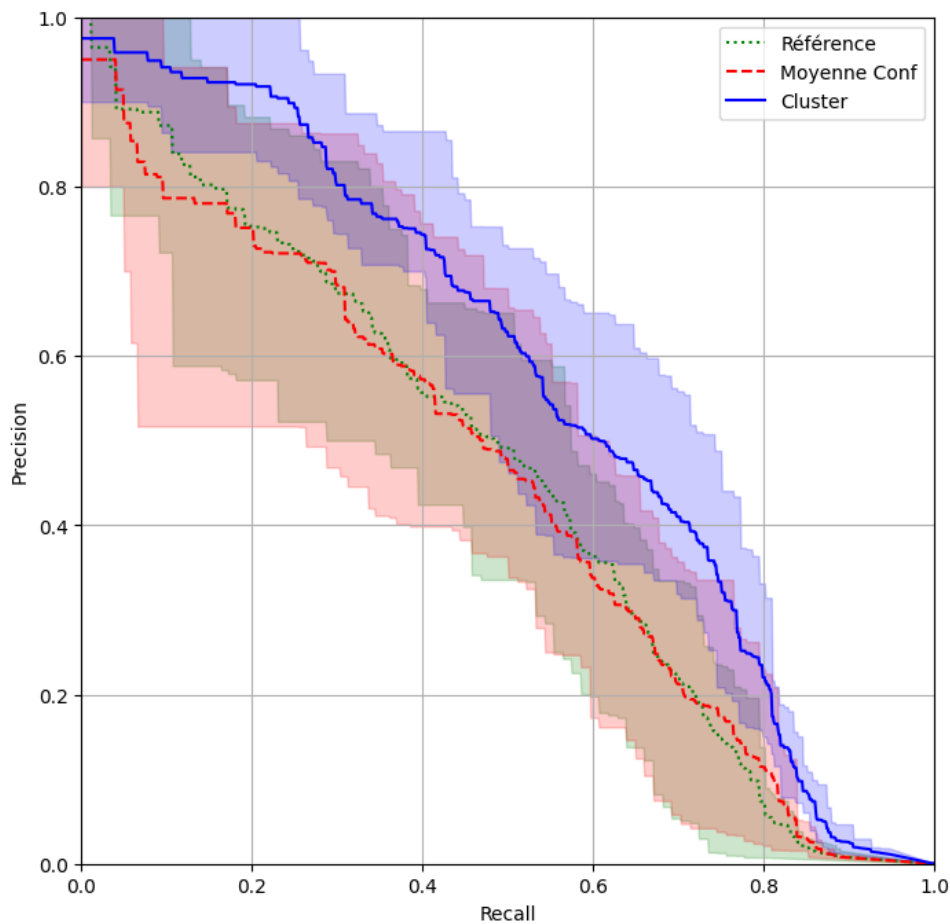


FIGURE 5.14 – Courbes ROC obtenues sur les quatre dossiers de cross-validation après réinjection de 3000 images, comparant la méthode fondée sur la confiance moyenne à notre approche hybride.

La méthode basée sur la moyenne des confiances présente des résultats proches de ceux obtenus avec le modèle de référence, sans réelle amélioration. En revanche, notre méthode hybride dépasse ces deux approches, indiquant une meilleure capacité à renforcer les performances du modèle.

Nous allons à présent examiner les performances de notre méthode comparée à une sélection aléatoire.

5.3.2.2 Comparaison des méthodes : Cluster / Aléatoire

Pour la deuxième comparaison, nous gardons la méthode étant la plus prometteuse et ayant obtenu les meilleurs résultats, la méthode se basant sur les clusters. Elle est maintenant comparée à la sélection d'images aléatoire.

Les résultats présentés dans le tableau 5.5 montre les performances relatives de chacune des méthodes.

TABLE 5.5 – Comparaison des AUCs compris entre 0.5 et 1 pour les méthodes de clustering et aléatoire. La référence sans images ajoutées a donné un résultat de 0.0743 ± 0.039 .

Méthode	500	1000	2000	3000
Clustering	0.0794 ± 0.0443	0.0908 $\pm \mathbf{0.0379}$	0.113 $\pm \mathbf{0.034}$	0.15 $\pm \mathbf{0.0423}$
Aléatoire	0.084 $\pm \mathbf{0.0436}$	0.0823 ± 0.0371	0.094 ± 0.0336	0.101 ± 0.0362

TABLE 5.6 – Comparaison du nombre d'images contenant un objet qui sont ajoutées dans les différentes bases grâce aux méthodes de clustering et aléatoire. Le test initial sans images ajoutées a donné une composition de 72 ± 15 / 2162 ± 48 .

Méthode	500	1000	2000	3000
Clustering	72 ± 15	145 ± 8	320 ± 26	448 ± 30
Aléatoire	50 ± 4	102 ± 8	196 ± 14	295 ± 9

La méthode utilisant du clustering obtient des valeurs d'AUC plus élevées à mesure que le nombre d'images augmente, améliorant ainsi les performances du modèle grâce à sa sélection. Pour ce qui est de la méthode aléatoire, bien qu'elle soit meilleure que la référence, n'augmente pas autant les performances que la méthode avec du clustering. Il y a une amélioration de 1.7% pour la méthode aléatoire contre 7% pour le clustering.

Quant au nombre d'images réinjectées dans les bases d'apprentissage recensé dans le tableau 5.6, les deux méthodes arrivent à sélectionner des images avec des objets contrairement à la méthode moyennant les confiances. Cependant, la méthode clustering sélectionne davantage d'objets car elle sélectionne en fonction des caractéristiques de l'image et non pas aléatoirement.

La méthode aléatoire parvient tout de même à extraire des images contenant des objets, mais sans spécificité pour ce type d'image. Elle se base uniquement sur la distribution de la base initiale, ce qui permet d'obtenir un certain nombre d'images avec des objets.

Contrairement à la figure 5.14, qui montrait que la méthode fondée sur la confiance moyenne n'apportait pas d'amélioration significative par rapport à la référence, la sélection aléatoire présente, quant à elle, une légère amélioration. Toutefois, comme l'illustre la figure 5.15, dans le cas précis de la réinjection de 3000 images, notre méthode hybride permet d'obtenir de meilleures performances que la référence et que la sélection aléatoire.

Grâce à ces différents éléments, il est maintenant possible d'avoir une analyse globale sur le fonctionnement de l'apprentissage actif ainsi que des différentes méthodes.

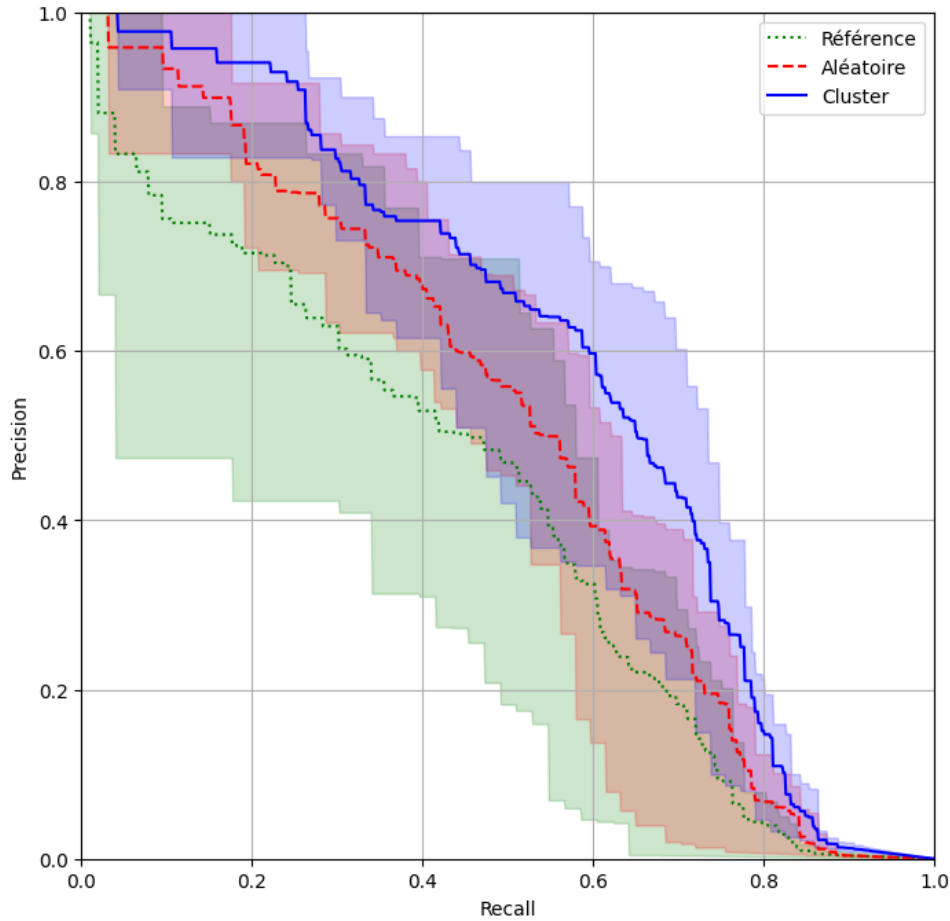


FIGURE 5.15 – Courbes ROC obtenues sur les quatre dossiers de cross-validation après réinjection de 3000 images, comparant la méthode aléatoire à notre approche hybride.

5.3.2.3 Analyse des résultats

D'autres tests ont été réalisés utilisant un petit nombre d'images annotées lors de la phase de *finetune*, de l'ordre de grandeur de quelques unités à quelques dizaines. Cependant, ces essais n'ont pas montré d'amélioration significative des performances. C'est pourquoi le nombre d'images annotées est assez conséquent et a été progressivement augmenté jusqu'à atteindre 3000. Bien que ce soit un nombre conséquent, cela permet d'obtenir une vision globale de l'amélioration apportée par la réintégration d'un grand volume d'images.

La base d'apprentissage initiale compte 2 436 images, et la création de la base B 3000 double cette taille pour la phase de *finetuning*. Cette configuration implique un effort d'annotation important, tant lors de l'initialisation que pour le *finetuning*. Toutefois, même avec une annotation plus limitée, environ 500 images, des gains de performance sont déjà observés.

Parmi les méthodes qui ont été mises en concurrence, la méthode moyennant les confiances au sein de chaque image a démontré l'importance des images contenant un objet pour améliorer la détection. Il est possible d'en déduire que le modèle de référence est capable de ne pas faire trop de faux positifs au sein des images composées uniquement de fond. Le fait de ne lui donner quasiment que des images sans objet n'améliore pas ses performances de détection.

Lorsque les images sont tirées aléatoirement, la distribution est aussi respectée, il y a donc davantage d'images avec objet. Grâce à quoi, une légère amélioration, d'environ 3% sur l'AUC entre 0.5 et 1, est notable. Cette méthode montre donc l'intérêt majeur d'extraire des images

autre que du fond pour apprendre au modèle de référence à mieux détecter et se focaliser sur autre chose que du banc de sable.

La dernière méthode utilisant du clustering afin de regrouper les images similaires entre elles, pour ensuite en extraire un certain nombre, vient confirmer totalement cette hypothèse. Comparé aux performances de la méthode aléatoire, cette méthode augmente les performances en AUC de 5%, ce qui fait un gain de quasiment 8% en AUC comparé à la référence.

L'objectif était d'améliorer les performances d'un modèle de détection dans un cadre de petites bases annotées. L'apprentissage actif, au travers différentes méthodes de sélection, permet d'accomplir cet objectif. Grâce à différents tests, il a été mis en évidence l'importance des images avec un objet, forçant ainsi à avoir une méthode de sélection efficace pour avoir un apprentissage actif réussi.

Pour compléter les comparaisons basées sur l'AUC, une analyse plus détaillée a été réalisée en examinant la répartition des prédictions correctes (True Positive (TP)), manquées (False Negative (FN)) et erronées (False Positive (FP)). Cette approche permet d'évaluer l'influence des différentes méthodes de sélection sur la nature des erreurs et des détections correctes.

Les paramètres d'évaluation ont été uniformisés pour toutes les méthodes et toutes les quantités d'images réinjectées, avec IoU = 0.4, NMS = 0.4 et un seuil de confiance fixé à 0.01.

5.3.2.3.1 Comparaison Cluster vs Moyenne des confiances

TABLE 5.7 – Résultats moyens ($\pm$ écart-type) pour IoU = 0.4, NMS = 0.4 et seuil de confiance = 0.01, comparant la méthode par clustering et la méthode par moyenne des confiances pour les trois métriques TP, FN et FP. Les meilleures valeurs sont indiquées en gras.

	Méthode	500	1000	2000	3000
TP	Cluster	116.2 $\pm$ 33.6	117.2 $\pm$ 35.0	117.8 $\pm$ 35.5	118.5 $\pm$ 32.5
	Moy. Conf	115.5 $\pm$ 34.7	114.2 $\pm$ 33.5	115.5 $\pm$ 34.2	116.5 $\pm$ 34.8
FN	Cluster	10.5 $\pm$ 3.7	10.2 $\pm$ 4.5	9.5 $\pm$ 1.7	10.0 $\pm$ 3.6
	Moy. Conf	11.2 $\pm$ 1.7	10.5 $\pm$ 5.8	11.5 $\pm$ 3.8	9.8 $\pm$ 2.8
FP	Cluster	1915.5 $\pm$ 1187.8	1391.8 $\pm$ 609.8	1020.2 $\pm$ 184.4	886.5 $\pm$ 235.1
	Moy. Conf	1902.8 $\pm$ 1003.7	1527.2 $\pm$ 770.6	1404.8 $\pm$ 402.1	1295.8 $\pm$ 133.0

Les résultats du tableau 5.7 et de la figure 5.16 montrent que les deux méthodes maintiennent un nombre de TP stable, le clustering conservant un léger avantage.

Le nombre moyen de FN est inférieur avec le clustering, ce qui traduit une meilleure détection des objets présents. Sur l'ensemble des dossiers de cross-validation, le nombre moyen d'images contenant des objets est de 129.75 ± 30.74 , soit un taux d'objets non détectés de 7.7% pour le clustering contre 8.5% pour la moyenne des confiances. Cette différence, exprimée en pourcentage, reste sensible à une variation d'une seule image, en raison du nombre limité d'images contenant des objets.

L'écart le plus marqué concerne les FP : avec l'augmentation du volume d'images réinjectées, le clustering réduit leur nombre de près de 1900 à moins de 900, tandis que la moyenne des confiances demeure élevée. L'écart-type des FP diminue également fortement, traduisant une plus grande stabilité sur l'ensemble des dossiers de cross-validation.

Ces résultats suggèrent que la sélection par clustering améliore à la fois les performances moyennes et la régularité des prédictions.

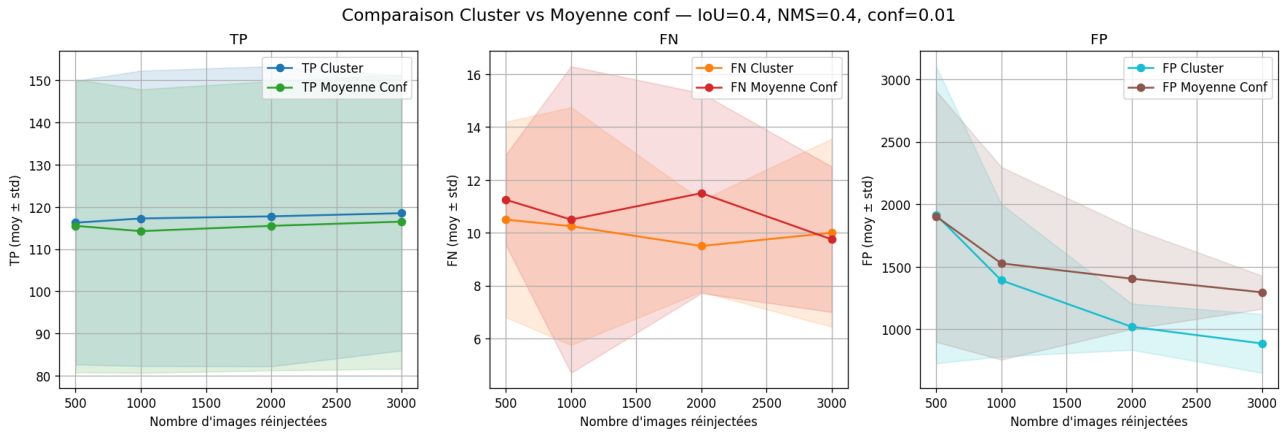


FIGURE 5.16 – Évolution des TP, FN et FP pour $\text{IoU} = 0.4$, $\text{NMS} = 0.4$ et seuil de confiance = 0.01, comparant la méthode par clustering et la méthode par moyenne des confiances. Les zones colorées représentent l'écart-type observé sur les plis de validation croisée.

5.3.2.3.2 Comparaison Cluster vs Aléatoire

TABLE 5.8 – Résultats moyens ($\pm$ écart-type) pour $\text{IoU} = 0.4$, $\text{NMS} = 0.4$ et seuil de confiance = 0.01, comparant la méthode par clustering et la méthode aléatoire pour les trois métriques TP, FN et FP. Les meilleures valeurs sont indiquées en gras.

	Méthode	500	1000	2000	3000
TP	Cluster	61.2 $\pm$ 19.0	61.5 $\pm$ 18.4	63.5 $\pm$ 18.3	63.5 $\pm$ 16.1
	Aléatoire	61.8 $\pm$ 18.1	60.8 $\pm$ 19.8	64.0 $\pm$ 17.3	63.5 $\pm$ 17.9
FN	Cluster	6.5 $\pm$ 1.0	6.2 $\pm$ 1.0	5.8 $\pm$ 2.8	6.5 $\pm$ 1.3
	Aléatoire	5.2 $\pm$ 1.7	6.2 $\pm$ 2.1	6.0 $\pm$ 1.8	5.2 $\pm$ 1.5
FP	Cluster	1365.8 $\pm$ 1022.1	888.5 $\pm$ 449.0	607.2 $\pm$ 133.0	545.5 $\pm$ 138.3
	Aléatoire	1044.0 $\pm$ 544.0	856.2 $\pm$ 428.4	1334.0 $\pm$ 840.5	947.0 $\pm$ 382.7

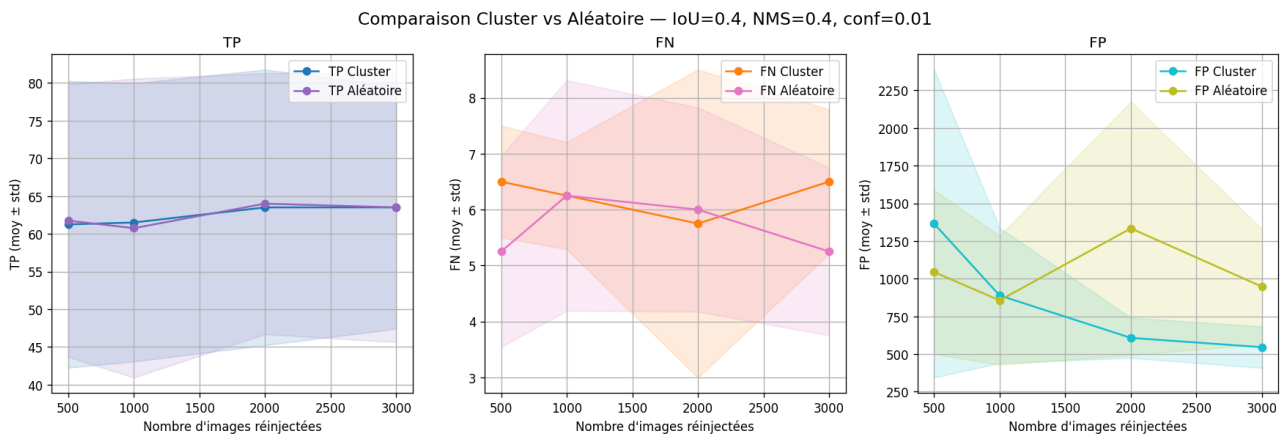


FIGURE 5.17 – Évolution des TP, FN et FP pour $\text{IoU} = 0.4$, $\text{NMS} = 0.4$ et seuil de confiance = 0.01, comparant la méthode par clustering et la méthode aléatoire. Les zones colorées représentent l'écart-type observé sur les plis de validation croisée.

L'examen du tableau 5.8 et de la figure 5.17 indique que les deux méthodes produisent un nombre de TP comparable, avec de faibles variations selon le volume d'images réinjectées. Les FN restent également proches, bien que l'approche aléatoire présente ponctuellement des

valeurs légèrement inférieures. La comparaison repose sur une moyenne de $71,75 \pm 15,37$ images contenant des objets, ce qui correspond à un taux de 8.5% lorsque 6 objets ne sont pas détectés, contre 7% pour 5 objets manquants. Les performances restent donc proches de celles observées dans la comparaison précédente, malgré un nombre total d'objets plus faible.

Les écarts apparaissent surtout sur les FP : le clustering maintient un volume plus faible de fausses détections à partir de 2000 images réinjectées et réduit nettement l'écart-type (de ± 1022 à ± 138), tandis que l'aléatoire conserve une variabilité élevée et une évolution moins régulière.

Ces résultats suggèrent qu'avec des performances comparables pour les TP et FN, le clustering présente un avantage en termes de maîtrise des FP et de stabilité au travers de la cross-validation.

5.3.2.3.3 Conclusion sur les comparaisons FN/FP/TP

La comparaison des méthodes selon les TP, FN et FP montre que l'approche par clustering reste robuste face aux différentes stratégies testées. Comparée à la moyenne des confiances, elle réduit les FN et FP tout en limitant leur variabilité. Face à la sélection aléatoire, les résultats en TP et FN sont similaires, mais le clustering conserve un avantage pour la réduction et la stabilisation des FP. Dans les deux cas, ces observations confirment que la sélection par clustering améliore la qualité des détections et la régularité des performances en apprentissage actif.

Au-delà des différences observées sur les FP, FN et TP, ces résultats doivent être replacés dans la perspective de l'application visée. L'objectif principal d'un système de détection n'est pas seulement de limiter les fausses détections, mais avant tout de reconnaître correctement le plus grand nombre d'objets présents dans les images. Dans cette optique, les TP et FN constituent des indicateurs clés : chaque objet manqué correspond à une information critique potentiellement perdue, ce qui peut s'avérer problématique dans des contextes opérationnels sensibles.

Les résultats montrent que la méthode de clustering maintient, voire améliore légèrement, le nombre de TP, tout en réduisant globalement les FN par rapport à la moyenne des confiances. Comparée à la sélection aléatoire, la différence sur les TP et FN apparaît plus limitée, ce qui nuance les gains apportés par le clustering. En revanche, sa capacité à contrôler les FP représente un avantage important, car l'accumulation de fausses détections peut rapidement affecter la fiabilité opérationnelle d'un système.

Ainsi, sans garantir une augmentation systématique du nombre d'objets détectés, le clustering favorise un compromis plus équilibré entre maximisation des détections et réduction des erreurs, critère essentiel pour l'efficacité globale d'un système de détection.

Il convient de souligner que l'amélioration obtenue par l'ajout de 3000 images annotées s'accompagne d'un coût notable. Ce volume correspond à un doublement de l'ensemble initial, impliquant un effort d'annotation important qui doit être mis en balance avec les gains observés.

Malgré cette exigence, le niveau de détection atteint reste significatif dans un contexte où l'identification visuelle d'objets demeure particulièrement complexe. Bien qu'imparfaite, cette performance illustre la capacité de l'approche à compléter l'analyse humaine, souvent sujette à des incertitudes et à des omissions, et à renforcer ainsi la robustesse du processus d'identification.

Troisième partie

Génération contrôlée d'images

Chapitre 6

Réseaux génératifs

Sommaire

6.1	Adaptation de domaine	108
6.1.1	Définition et enjeux	108
6.1.2	Décalage de distribution et particularités des images sonar	108
6.1.3	Stratégies générales d'adaptation	109
6.1.4	Apprentissage non-supervisé de la translation inter-domaines, sans appariement	111
6.1.5	Vers l'usage de modèles génératifs	112
6.2	Principe	112
6.3	CycleGAN	115
6.3.1	Architecture	115
6.3.2	Apprentissage	115
6.3.2.1	Loss adversaire (<i>adversarial loss</i>)	116
6.3.2.2	Loss de consistance cyclique (<i>cycle consistency loss</i>)	116
6.3.2.3	Objectif global	117
6.3.3	Conclusion	118
6.4	StarGAN	118
6.4.1	Architecture	119
6.4.1.1	Le générateur	119
6.4.1.2	Le réseau de cartographie	120
6.4.1.3	L'encodeur de style	120
6.4.1.4	Le discriminant	121
6.4.2	Apprentissage	121
6.4.2.1	Loss adversariale	121
6.4.2.2	Loss pour la reconstruction de style	121
6.4.2.3	Loss pour la diversification de style	121
6.4.2.4	Loss pour préserver les caractéristiques de l'image source	122
6.4.3	Objectif global	122

Les modèles génératifs ont pour objectif de produire de nouvelles données similaires aux exemples de l'ensemble d'entraînement. Ces modèles sont parfois utilisés pour augmenter la quantité de données d'entraînement, ce qui contribue à améliorer la robustesse et les performances des modèles d'apprentissage.

Dans les tâches où les données disponibles sont limitées ou coûteuses à obtenir, les modèles génératifs pourrait ainsi offrir une alternative intéressante pour enrichir l'ensemble de données en créant des variations simulant des exemples réalistes basées sur des informations importantes. Cette hypothèse est particulièrement prometteuse dans les domaines nécessitant des annotations manuelles fastidieuses, comme la détection d'objets dans des images complexes, où la diversité des données est importante pour améliorer les performances du modèle.

La suite de ce chapitre examine comment des modèles génératifs, entraînés à partir d'images synthétiques, peuvent être exploités pour enrichir les ensembles de données.

6.1 Adaptation de domaine

6.1.1 Définition et enjeux

La rareté des données annotées constitue l'une des principales motivations de l'adaptation de domaines, en transférant les connaissances acquises dans un domaine source richement annoté vers un domaine cible pour lequel peu ou pas d'annotations sont disponibles. C'est le cas dans certains scénarios spécifiques, tels que la conduite autonome [Hoffman *et al.*, 2018] ou bien l'imagerie médicale [Kamnitsas *et al.*, 2017].

Formellement, l'adaptation de domaines s'inscrit dans le cadre plus général de l'apprentissage par transfert (*Transfer Learning*). Elle repose sur l'hypothèse que les données du domaine source et celles du domaine cible suivent des distributions différentes, ce qui constitue un obstacle majeur à la généralisation des modèles d'apprentissage supervisé. Ce décalage, désigné par le terme *domain shift*, limite la capacité de généralisation des modèles, qui tendent à capter les spécificités du domaine source sans les transposer efficacement vers le domaine cible.

L'objectif est donc d'exploiter au mieux les données du domaine source tout en atténuant les effets du décalage entre les domaines afin d'optimiser les performances dans le domaine cible. Les stratégies adoptées varient selon le niveau d'annotation disponible dans ce dernier, distinguant l'adaptation supervisée, semi-supervisée et non supervisée. Le cas le plus réaliste est celui où il n'y a aucune annotation dans le domaine cible [Ganin *et al.*, 2016].

Dans notre cas, le domaine source est constitué d'images sonar simulées, tandis que le domaine cible correspond à des données sonar réelles. Cette configuration pose des défis spécifiques, liés à l'hétérogénéité des distributions mais également à la difficulté de reproduire fidèlement les caractéristiques du domaine réel. Les images simulées et réelles présentent souvent des écarts significatifs, compromettant ainsi le transfert efficace des connaissances.

6.1.2 Décalage de distribution et particularités des images sonar

Le décalage de distribution est l'un des principaux obstacles à l'adaptation de domaines. Ce décalage peut être visuel (textures, bruit, résolution), structurel (variations de forme, de position ou présence/absence de certaines classes), ou statistique (distribution marginale ou conjointe

des données et des étiquettes). Il provient fréquemment de différences liées aux conditions de capture, aux capteurs utilisés ou aux protocoles expérimentaux, comme documenté dans plusieurs travaux [Quinonero-Candela *et al.*, 2009 ; Ganin *et al.*, 2016 ; Tzeng *et al.*, 2017].

Dans ce contexte, le décalage entre domaines est fortement accentué par la nature des données : le domaine source repose sur des images sonar simulées, tandis que le domaine cible correspond à des acquisitions réelles. Bien que les simulations s'appuient sur des modèles acoustiques capables de représenter diverses configurations, comme présenté dans la section 2.1, les images générées restent idéalement construites. En particulier, les interactions acoustiques complexes, comme les réflexions multiples, diffusion de l'onde, interférences structurales, sont difficilement modélisables de manière réaliste. Le bruit simulé est de nature gaussienne, ce qui diffère sensiblement du bruit réel, généralement non stationnaire, hétérogène et dépendant de l'environnement sous-marin.

Les images réelles, quant à elles, sont affectées par de nombreux phénomènes non modélisés, tels que la réverbération, la rugosité du fond, les interférences mécaniques du système d'acquisition, ou encore les variations locales de propagation ou des gradients de température et de salinité. Ces effets génèrent une forte variabilité dans l'apparence des scènes et l'intensité des retours. De plus, les objets simulés apparaissent souvent nets et bien définis, contrairement aux objets réels, qui peuvent être partiellement masqués, faiblement contrastés ou altérés par des artefacts.

Cette dissymétrie complique la reproduction fidèle des caractéristiques du domaine réel. Par conséquent, les modèles entraînés sur des données simulées rencontrent des difficultés à obtenir de bons résultats sur les images réelles. Ce phénomène, connu sous le nom de *reality gap*, est bien établi dans les domaines de la robotique et de la vision par ordinateur [Tobin *et al.*, 2017 ; Tremblay *et al.*, 2018], et constitue un défi majeur lors de la simulation de données.

Afin de combler le peu de données réelles disponibles, l'usage d'images simulées apparaît alors comme une solution pertinente, en raison de leur accessibilité en grande quantité. Toutefois, cela nécessite le développement de méthodes capables de transférer efficacement l'information du domaine simulé vers le domaine réel, tout en intégrant les disparités inhérentes à la qualité des données.

La section suivante présente les principales stratégies d'adaptation envisageables dans ce cadre, en exposant les contraintes spécifiques liées à l'usage de transformations d'images.

6.1.3 Stratégies générales d'adaptation

Pour réduire le décalage de distribution entre un domaine source et un domaine cible, plusieurs stratégies d'adaptation ont été proposées dans la littérature. Ces méthodes peuvent être regroupées en plusieurs grandes familles. Certaines approches se distinguent, notamment celles fondées sur l'alignement des caractéristiques latentes et celles reposant sur la transformation des données d'entrée, bien que d'autres stratégies, telles que l'auto-apprentissage ou l'alignement au niveau des sorties, aient également été proposées.

Les méthodes *feature-based* visent à apprendre un espace de représentation commun dans lequel les données des deux domaines sont alignées, permettant ainsi à un modèle entraîné dans cet espace de généraliser aux deux domaines. Ces approches incluent notamment des techniques d'alignement explicite par minimisation de divergence (par exemple MMD [Gretton *et al.*, 2009 ; Long *et al.*, 2015]) ou par apprentissage adversaire sur les représentations [Tzeng *et al.*, 2017 ; Ganin *et al.*, 2016]. La figure 6.1 schématise ce type de méthodes.

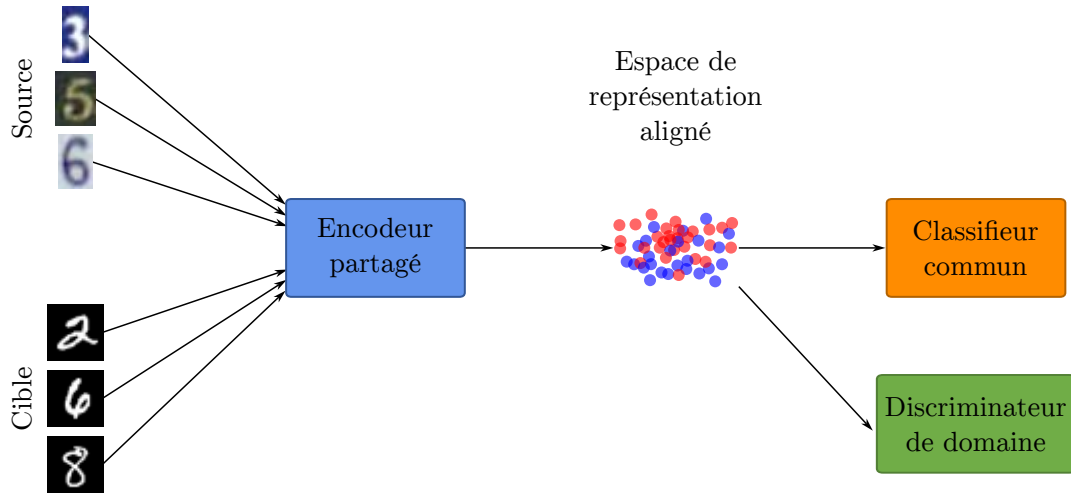


FIGURE 6.1 – Illustration du paradigme d'adaptation de domaine au niveau des représentations.

Les données des domaines source et cible sont encodées dans un espace de représentation commun à l'aide d'un encodeur partagé. Cet espace est aligné de façon à permettre à un classifieur unique de généraliser aux deux domaines. Dans le cadre d'un apprentissage adversaire, cet alignement est renforcé par l'introduction d'un discriminateur de domaine.

À l'inverse, les approches dites *input-level* cherchent à transformer les données du domaine source afin qu'elles adoptent l'apparence visuelle du domaine cible [Bousmalis *et al.*, 2017; Hoffman *et al.*, 2018], ce principe est illustré dans la figure 6.2.

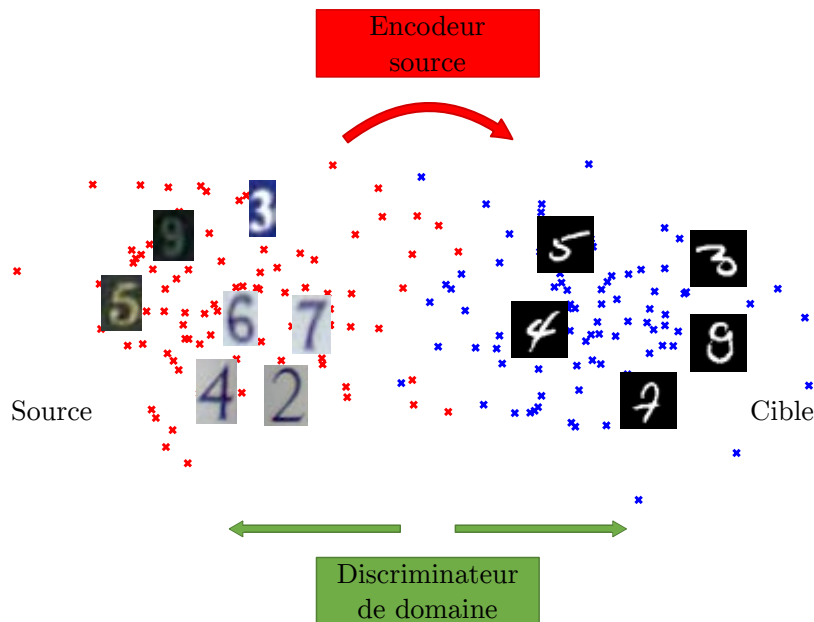


FIGURE 6.2 – Adaptation de domaine au niveau de l'entrée par translation visuelle des données sources vers le domaine cible.

Ce paradigme de translation d'image est intéressant car il ne nécessite pas de modification du modèle aval (tel qu'un détecteur d'objets), qui peut ainsi être entraîné directement sur les images transformées tout en exploitant les annotations initialement disponibles dans le domaine source [Shrivastava *et al.*, 2017]. Ce type d'approche est particulièrement pertinent dans les contextes où les données sources sont synthétiques, comme dans le cas d'images simulées, et où l'objectif est d'en rapprocher l'apparence de celle des données réelles [Tobin *et al.*, 2017;

Tremblay *et al.*, 2018].

Dans le cadre de la détection, et pour la famille *input-visual*, la transformation source $\rightarrow$ cible ne doit pas altérer la structure de l'image, et la position des objets d'intérêt doit être prévisible. Autrement dit, bien que le style visuel soit modifié pour correspondre au domaine cible, les objets doivent, soit rester inchangés en forme, taille, position et identité sémantique, soit ces paramètres doivent être prévisibles, afin de conserver les annotations du domaine source. Cette exigence implique un processus de génération contrôlé et cohérent. Une fois que le générateur a appris, il est alors possible de générer des images ressemblantes aux images de la base cible dont on a des annotations.

Il est important de distinguer les approches fondées sur des données *appariées* de celles reposant sur des données *non appariées*.

Dans le cadre apparié, chaque image du domaine source est associée à une image correspondante du domaine cible, représentant la même scène ou le même objet dans un style différent. Cette correspondance permet d'entraîner des modèles de translation supervisée, tels que Pix2Pix [Isola *et al.*, 2018], en calculant directement une perte entre l'image générée et la cible réelle.

À l'inverse, dans de nombreux cas pratiques, notamment pour les images sonar, il est impossible d'obtenir un tel appariement : les scènes simulées et réelles diffèrent en géométrie, en contexte et en bruit d'acquisition. Ces situations relèvent du cadre *non supervisé*, où aucune correspondance explicite entre domaines n'est disponible. Le CycleGAN [Zhu *et al.*, 2017] est un réseau développé pour ce type de problématique.

Dans notre scénario, la maîtrise sur les générations des objets est particulièrement critique. En effet, la perception des objets est difficile et la localisation précise des objets est essentielle. Toute altération géométrique introduite lors de la transformation source $\rightarrow$ cible compromettrait directement la vérité terrain utilisée pour l'entraînement ou l'évaluation d'un détecteur.

6.1.4 Apprentissage non-supervisé de la translation inter-domaines, sans appariement

La plupart des approches d'adaptation de domaine reposent sur l'hypothèse qu'il existe une correspondance, explicite ou implicite, entre les distributions source et cible. Dans certains cas favorables, des paires d'images appariées peuvent être obtenues, comme dans les jeux de données synthétiques générés à partir de moteurs de rendu 3D, où chaque image peut être associée à sa version réaliste [Wang et Deng, 2018]. Cependant, dans la majorité des cas réels, et en particulier dans le contexte du transfert simulation à réalité, une telle correspondance n'est pas disponible. Ce cadre est désigné sous le nom d'*adaptation de domaines non supervisée avec données non appariées*.

Dans le cas de l'imagerie sonar, il n'existe pas d'image réelle correspondant précisément à une image simulée donnée. Non seulement les objets représentés diffèrent, mais les conditions de prise de vue, les fonds marins, les bruits acoustiques et les paramètres des capteurs varient également. Cette absence de correspondance directe rend inutilisables les approches supervisées ou faiblement supervisées fondées sur une supervision pixel-à-pixel.

Par ailleurs, pour des raisons de sécurité et de confidentialité, l'accès aux paramètres exacts du sonar réel, tels que la résolution, la fréquence d'acquisition ou la dynamique de réponse, est souvent restreint, accentuant ainsi la dissymétrie entre les domaines simulé et réel.

Dans notre contexte, il est nécessaire de recourir à des méthodes capables d'apprendre une transformation entre domaines sans supervision explicite, tout en générant des échantillons visuellement plausibles dans le domaine cible à partir de données simulées. Il est également essentiel que ces méthodes préservent la structure des scènes d'origine et leur contenu sémantique afin de maintenir la validité des annotations, notamment dans des tâches supervisées comme la détection d'objets.

Ce type de problématique a conduit au développement de méthodes basées sur des modèles génératifs capables de capturer et transférer les distributions visuelles d'un domaine à l'autre sans nécessiter d'alignement direct entre les données. Ces approches, et en particulier les GAN, seront détaillées dans la section suivante.

6.1.5 Vers l'usage de modèles génératifs

L'apprentissage d'une transformation réaliste entre un domaine source et un domaine cible, en l'absence d'images appariées, constitue une tâche complexe. Il ne s'agit pas seulement d'ajuster une distribution statistique globale, mais de générer des images visuellement compatibles avec le style du domaine cible, tout en conservant le contenu sémantique de l'image source. Ce problème relève de l'apprentissage génératif conditionnel, qui nécessite des modèles capables de modéliser la distribution cible à partir d'exemples non appariés, tout en respectant des contraintes structurelles ou sémantiques.

Les GAN, introduits par Goodfellow *et al.* [2020], ont établi les bases de l'apprentissage génératif adversaire. Si le modèle initial n'était pas conditionnel, plusieurs extensions, comme les *conditional GANs* [Mirza et Osindero, 2014], ont introduit un contrôle explicite sur la sortie, rendant possible des tâches telles que la translation d'images entre domaines. L'entraînement conjoint des deux réseaux permet d'apprendre à générer des images réalistes, respectant à la fois les caractéristiques du domaine cible et les contraintes définies par l'entrée.

Pour l'adaptation de domaines, les GAN ont été étendus à la translation d'images entre domaines non appariés, en intégrant des contraintes assurant la préservation du contenu structurel notamment la consistance cyclique, comme dans CycleGAN [Zhu *et al.*, 2017]. Ces approches permettent de transformer des images simulées dans le style du domaine réel, tout en maintenant la validité des annotations, condition indispensable aux tâches de détection aval.

La section suivante présente les principes fondamentaux des GAN, leur architecture, leur mécanisme d'apprentissage, ainsi que les principaux défis liés à leur entraînement.

6.2 Principe

Le principe du *Generative Adversarial Networks (GAN)* repose sur l'apprentissage adversaire. C'est un modèle composé de deux réseaux de neurones distincts : un générateur et un discriminateur. Sa finalité consiste à générer des images ressemblant le plus possible aux images ayant servi lors de l'entraînement. Ceci est rendu possible grâce à un apprentissage compétitif entre le générateur et le discriminateur.

Le générateur récupère un bruit aléatoire en entrée, et essaye de générer des données réelles. Il doit donc apprendre à transformer le bruit initial, en données ressemblant aux données

d'apprentissage, au travers d'une série de couches convolutives ou récurrentes [Rumelhart *et al.*, 1986]. Initialement, le générateur produit des échantillons de très mauvaise qualité, mais il s'améliore progressivement au cours de l'apprentissage.

Quant au discriminateur, sa tâche consiste à être capable de discerner les vraies images provenant de la base d'entraînement, de celles produites par le générateur. Au cours de l'entraînement, il apprend à classifier ses entrées selon si elles proviennent de la base cible ou si elles sont générées. Tout comme le générateur, ceci se réalise grâce au passage de ces données au travers d'une série de couches. Au début, le discriminateur peut réaliser des erreurs, mais étant entraîné avec le générateur, il améliore sa capacité à différencier les "vraies" et les "fausses" images.

Ces deux réseaux jouent ainsi un jeu de type "min-max", où le générateur tente de minimiser la capacité du discriminateur à classer correctement les échantillons fournis, tandis que le discriminateur tente de maximiser sa précision.

Grâce à cette relation antagoniste, le générateur apprend à générer des échantillons les plus réalistes possibles, tandis que le discriminateur distingue de mieux en mieux les données réelles des données fausses. Ce processus se poursuit de manière itérative jusqu'à ce que le générateur produise des échantillons indiscernables de données réelles, selon le discriminateur. Ce fonctionnement est représenté dans la figure 6.3

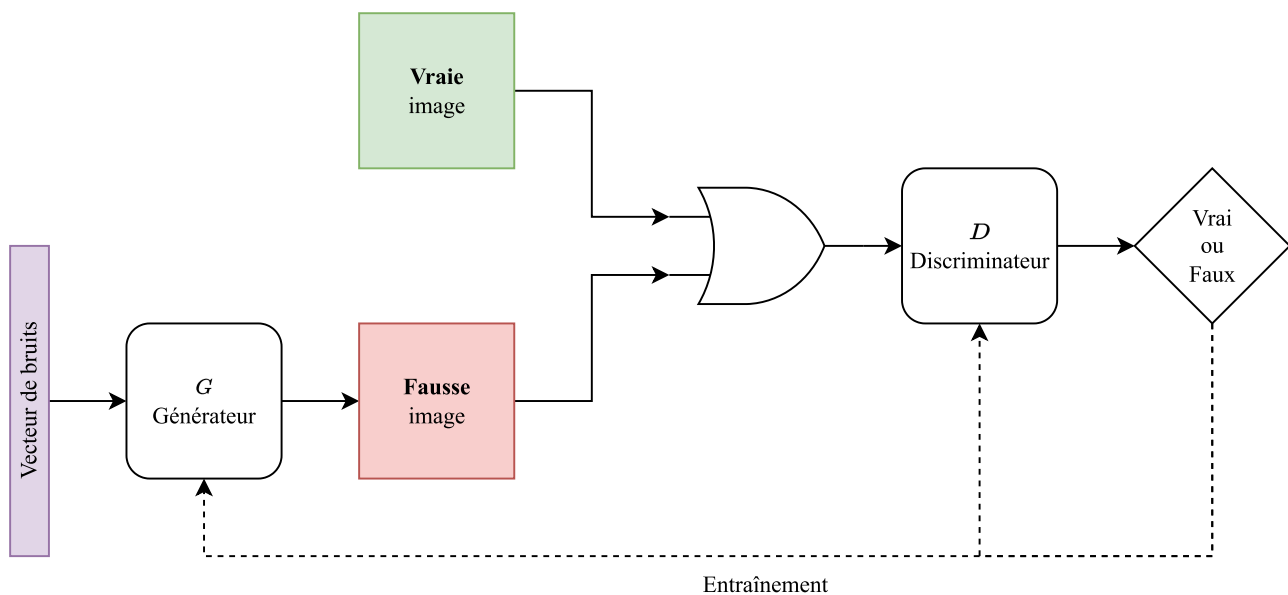


FIGURE 6.3 – Schéma représentant le cycle d'apprentissage d'un GAN

Toutefois, en pratique, les entraînements sont plus complexes. Les GAN étant connus pour être instables, les difficultés consistent essentiellement à faire converger les deux réseaux adversaires correctement. Leur compétition, peut mener à des oscillations voire même un effondrement global de l'entraînement [Zhai *et al.*, 2019], car le générateur ne parvient pas à capturer la distribution complète des données. Cette instabilité résulte de la nature non convexe de la fonction objectif des GAN.

Les conséquences de cet effondrement sont telles que le générateur ne produit que des échantillons avec une variation limitée, et ne parvient pas à apprendre l'intégralité de la diversité de la distribution. Le cas peut se produire lorsque le générateur converge vers quelques modes dominants ou motifs favorisés par le discriminateur.

L'évaluation des performances des GAN est une fonction extrêmement complexe. Concernant les tâches d'apprentissage supervisé, comme la détection d'objets (vu précédemment dans la

section 3.2), il existe des métriques d'évaluation bien définies, alors que les GAN ne disposent d'aucune mesure quantitative directe sur les performances. Évaluer la qualité et la diversité des échantillons générés repose souvent sur un jugement subjectif humain.

Des métriques telles que la *Frechet Inception Distance (FID)* [Heusel *et al.*, 2017] et l'*Inception Score (IS)* [Salimans *et al.*, 2016] ont tout de même été proposées pour évaluer la qualité des images générées par des modèles adversaires. Elles visent à quantifier automatiquement à la fois la fidélité des échantillons (proximité avec les données réelles) et leur diversité (absence de redondance ou d'effondrement des modes).

L'IS mesure la qualité des images générées en combinant deux éléments : une prédiction sûre pour chaque image, signe de clarté visuelle, et une diversité élevée des classes prédites sur l'ensemble du jeu, indiquant une bonne variété des échantillons. Il repose sur la divergence KL [Kullback et Leibler, 1951] entre la distribution conditionnelle $p(y|x)$, où x désigne une image générée et y la classe prédite par un classifieur pré-entraîné, et la distribution marginale $p(y)$ sur l'ensemble des images générées. Toutefois, cette métrique présente plusieurs limites : elle ne tient pas compte des données réelles, ignore la diversité intra-classe et dépend fortement du classifieur utilisé.

Le FID, introduit comme alternative plus robuste, compare les statistiques (moyenne et covariance) des représentations latentes extraites des données réelles et générées. Il mesure la distance entre ces distributions via la distance de Fréchet [Fréchet, 1906], et présente une meilleure corrélation avec l'évaluation humaine, en intégrant à la fois qualité visuelle et proximité avec la distribution réelle. Néanmoins, cette mesure reste limitée par l'hypothèse de normalité et par son insensibilité à certains artefacts structurels ou sémantiques.

Comme le soulignent plusieurs études [Chong et Forsyth, 2020 ; Borji, 2018], ni le FID ni l'IS ne permettent de capturer l'ensemble des dimensions pertinentes pour évaluer les GAN. Ils ignorent, entre autres, la cohérence structurelle des objets, la fidélité au contenu sémantique initial (notamment en traduction d'image) et la préservation des relations spatiales. Ils ne fournissent pas d'indications sur l'impact des images générées sur les performances du modèle dans la tâche cible, pourtant déterminantes en contexte applicatif. Ces métriques, bien que largement utilisées, doivent donc être interprétées avec précaution et idéalement complétées par des évaluations qualitatives ou orientées tâche.

Un autre problème majeur provient des capacités de calcul nécessaires. Le processus d'entraînement implique généralement plusieurs itérations, où chaque itération nécessite des allers-retours au travers des réseaux. Globalement, entraîner un GAN, sur un ensemble de données à grande échelle, demande d'importantes ressources informatiques ainsi qu'un temps considérable. Creswell *et al.* [2018] ; Oncevay *et al.* [2021] ont discuté des problématiques liées aux grandes bases de données, les ressources nécessaires et autres sujets majeurs liés à l'entraînement des GAN.

De manière générale, les GAN présentent certaines limites, que la communauté scientifique cherche à surmonter en intégrant des contraintes sémantiques [Fu *et al.*, 2019], en améliorant la stabilité de l'entraînement ou en explorant de nouvelles architectures.

Les GAN ne sont pas faciles à utiliser, notamment en raison de leur instabilité d'entraînement et de leur sensibilité aux hyperparamètres [Arjovsky et Bottou, 2017 ; Mescheder *et al.*, 2018], mais ils ont néanmoins permis des avancées majeures dans de nombreux domaines tels que l'audio [Donahue *et al.*, 2019], l'imagerie médicale [Kazemini *et al.*, 2020], la restauration ou la génération d'images [Ledig *et al.*, 2017 ; Karras *et al.*, 2019], ou encore la synthèse de visages hyperréalistes dans le cadre des “*deepfakes*” [Nirkin *et al.*, 2019].

Nous allons aborder ci-après quelques exemples de résultats significatifs.

6.3 CycleGAN

La transformation Image-to-Image translation (I2I) a fortement progressé avec l'introduction des GANs. Toutefois, de nombreuses approches reposaient sur des données d'entraînement appariées. Pour dépasser cette contrainte, cette section s'intéresse au “*Cycle Generative Adversarial Network*” (CycleGAN) [Zhu *et al.*, 2017], une méthode conçue pour la traduction d'images non appariées, tout en allégeant les exigences des GAN traditionnels.

6.3.1 Architecture

L'architecture du CycleGAN est composée de deux éléments principaux : les générateurs et les discriminants. Il y a deux générateurs : G et F , où G traduit les images du domaine X au domaine Y , et F fait l'inverse. De même, il y a deux discriminants D_X et D_Y , utilisés pour différencier les images réelles des images traduites, pour les domaines X et Y , respectivement (voir figure 6.4).

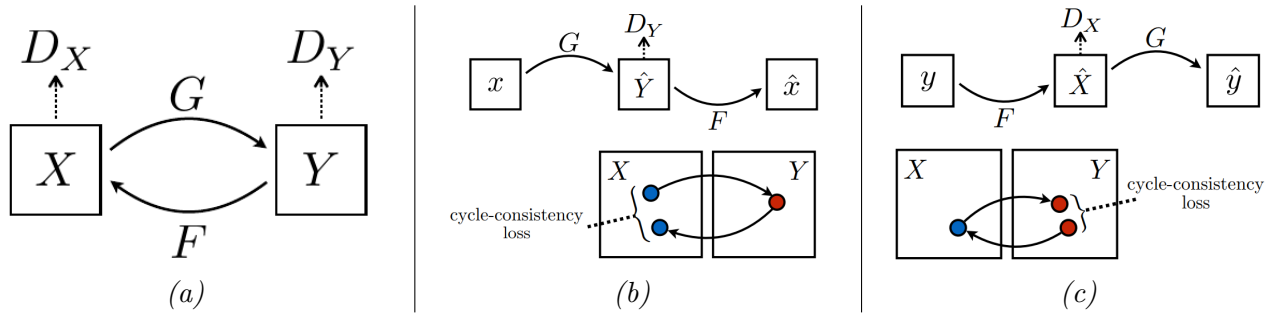


FIGURE 6.4 – Les différents composants et fonctionnements du CycleGAN, extrait de l'article Zhu et al. [2017]

6.3.2 Apprentissage

Le modèle CycleGAN repose sur deux fonctions de transformation principales, $G : X \rightarrow Y$ et $F : Y \rightarrow X$, chacune associée à un discriminateur adversaire (D_Y et D_X) chargé de distinguer les images générées des images réelles dans leur domaine respectif. L'entraînement s'appuie sur une perte adversaire incitant les générateurs à produire des images visuellement réalistes, difficilement distinguables des données réelles par les discriminants.

En l'absence de paires d'images appariées entre les domaines, cette contrainte ne suffit toutefois pas à préserver le contenu structurel des images d'origine. Une perte de consistance cyclique est donc introduite : elle impose qu'une image traduite vers un autre domaine puis reconvertie dans son domaine initial reste proche de sa version d'origine.

Contrairement à d'autres architectures génératives, telles que StyleGAN [Karras *et al.*, 2019], qui intègrent explicitement un vecteur latent pour modéliser la diversité, CycleGAN adopte une approche déterministe : pour une image donnée, le générateur produit une sortie unique, sans composante aléatoire.

Cette caractéristique simplifie l'apprentissage en restreignant l'espace des sorties possibles, mais limite la capacité du modèle à représenter la diversité du domaine cible, notamment en présence de structures multimodales. Elle rend d'autant plus nécessaire l'usage de contraintes

structurelles fortes, comme la consistance cyclique, afin de maintenir une correspondance stable entre les domaines.

Ces deux composantes, adversaire et cyclique, constituent le cœur de l'apprentissage du CycleGAN. Elles sont décrites dans les sous-sections suivantes.

6.3.2.1 Loss adversaire (*adversarial loss*)

Des loss sont appliquées sur les deux transformations. Concernant, $G : X \rightarrow Y$ et son discriminant D_Y , l'objectif se définit comme l'équation (6.1a). Par analogie, il est possible de l'appliquer sur l'autre transformation $F : Y \rightarrow X$ et D_X , dans l'équation (6.1b). Ces deux équations sont les suivantes :

$$\mathcal{L}_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x)))] \quad (6.1a)$$

$$\mathcal{L}_{GAN}(F, D_X, Y, X) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D_X(x)] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log(1 - D_X(F(y)))] \quad (6.1b)$$

G tente de générer des images $G(x)$ qui ressemblent aux images du domaine Y , tandis que D_Y essaye de distinguer les échantillons $G(x)$ des échantillons réels y . G cherche à minimiser cet objectif contre un adversaire D qui tente, lui, de le maximiser, donnant lieu aux équations suivantes :

$$\min_G \max_{D_Y} \mathcal{L}_{GAN}(G, D_Y, X, Y) \quad (6.2a)$$

$$\min_F \max_{D_X} \mathcal{L}_{GAN}(F, D_X, Y, X) \quad (6.2b)$$

6.3.2.2 Loss de consistance cyclique (*cycle consistency loss*)

Avec un volume de données suffisant, un réseau peut apprendre une correspondance arbitraire entre les images d'entrée et celles du domaine cible. Une telle correspondance, bien que statistiquement plausible, ne garantit pas que chaque image source x_i soit transformée en une image cible y_i pertinente. Les pertes adversaires, à elles seules, ne permettent donc pas d'imposer une relation précise entre chaque entrée et la sortie attendue.

Pour réduire davantage l'espace des possibilités des fonctions, ces fonctions de transformations doivent être cohérentes de manière cyclique, comme illustré dans la figure 6.4b. Dans chaque image x du domaine X , le cycle de transformation de l'image doit pouvoir ramener x à l'image originale, à savoir l'équation (6.3), appelé “*forward cycle consistency*”.

$$F(G(x)) \approx x \quad \forall x \in X \quad (6.3)$$

De même, comme l'illustre la figure 6.4c, pour chaque image y du domaine Y , G et F doivent également satisfaire le “*backward cycle consistency*”, correspondant à l'équation (6.4).

$$G(F(y)) \approx y \quad \forall y \in Y \quad (6.4)$$

Ce comportement est encouragé grâce à la loss de consistance cyclique (*cycle consistency*) :

$$\mathcal{L}_{cyc}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1] \quad (6.5)$$

Les auteurs mentionnent qu'ils ont essayé de modifier la norme $L1$ par une loss adversariale entre $F(G(x))$ et x , et entre $G(F(y))$ et y , mais sans observations d'améliorations des performances.

Le comportement induit par la loss de consistance cyclique peut être décrit dans la figure 6.5 ci-après. Les images reconstruites $F(G(x))$ finissent par correspondre étroitement aux images d'entrée x .

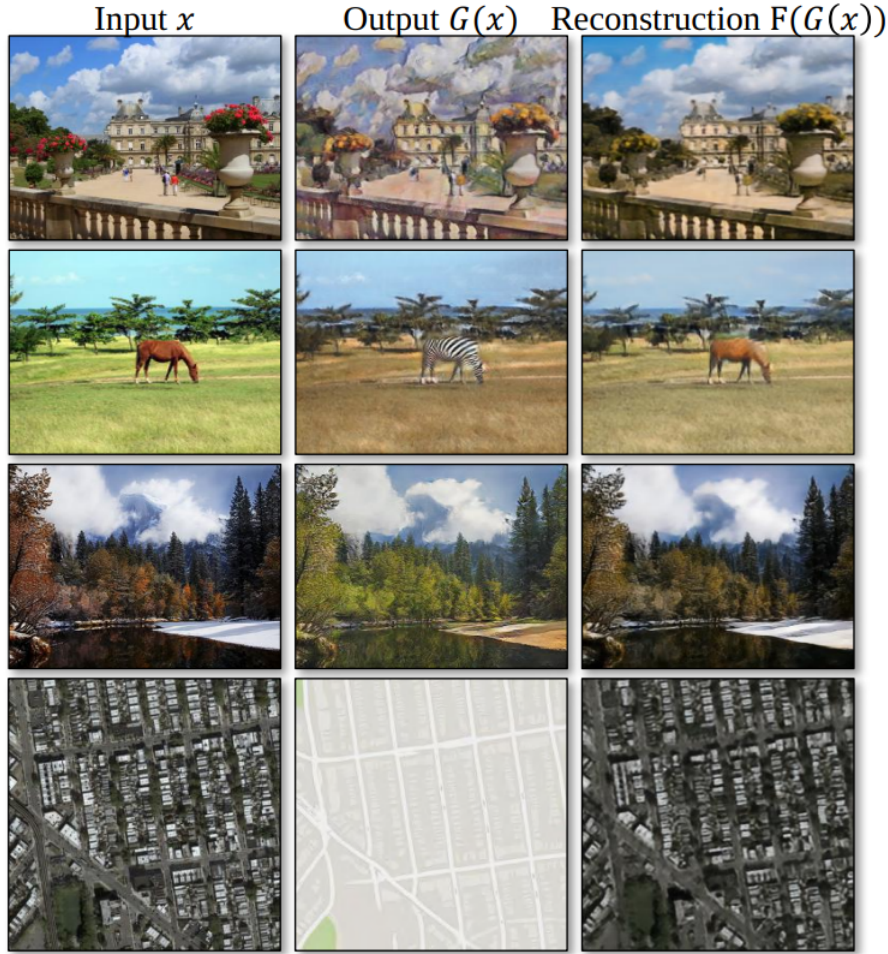


FIGURE 6.5 – Exemple de reconstruction des images grâce au CycleGAN.

Ces transformations proviennent de diverses expériences. Les colonnes de gauche à droite correspondent à l'image d'entrée x , les image de sortie de $G(x)$ et les images reconstruites $F(G(x))$. Pour les lignes de haut en bas, les transformations sont les suivantes : photo $\leftrightarrow$ Cézanne ; cheval $\leftrightarrow$ zèbre ; hiver $\leftrightarrow$ été Yosemite ; photo aérienne $\leftrightarrow$ Google maps.

6.3.2.3 Objectif global

L'équation globale régissant l'entraînement peut s'écrire de la manière suivante :

$$\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{GAN}(G, D_Y, X, Y) + \mathcal{L}_{GAN}(F, D_X, Y, X) + \lambda \mathcal{L}_{cyc}(G, F) \quad (6.6)$$

avec λ permettant de contrôler l'importance relative des deux objectifs cherchant à résoudre :

$$G^*, F^* = \arg \min_{G, F} \max_{D_X, D_Y} \mathcal{L}(G, F, D_X, D_Y) \quad (6.7)$$

Il est possible de considérer le modèle comme l'apprentissage de deux “autoencodeurs” [Hinton et Salakhutdinov, 2006], appris conjointement.

$$F \circ G = X \rightarrow X \quad (6.8a)$$

$$G \circ F = Y \rightarrow Y \quad (6.8b)$$

Il convient néanmoins de souligner que ces autoencodeurs présentent des structures internes différentes. Ils associent une image à elle-même via une représentation intermédiaire correspondant à une transformation dans un espace distinct. Cette configuration peut être vue comme un cas particulier d’un “autoencodeur adversaire” [Makhzani *et al.*, 2016], où une perte adversaire est utilisée pour régulariser l’espace latent, en contraignant la représentation intermédiaire à suivre une distribution cible définie a priori.

Dans le cas du CycleGAN, la transformation $X \rightarrow X$, réalisée à travers le cycle $x \rightarrow G(x) \rightarrow F(G(x))$, peut être interprétée comme un autoencodeur dont le goulot d’étranglement implicite (l’image générée dans le domaine Y) est régularisé par une perte adversaire afin de se conformer aux caractéristiques du domaine cible. Le domaine Y joue ainsi le rôle de distribution de régularisation pour la reconstruction dans le domaine source.

Dans l’article de Zhu *et al.* [2017], les auteurs effectuent une comparaison entre leur approche et des variations de l’objectif global, incluant l’utilisation exclusive de la loss adversaire $\mathcal{L}_{GAN}$, ainsi que l’utilisation unique de la loss de consistance cyclique $\mathcal{L}_{cyc}$. Leur analyse empirique démontre clairement que ces deux objectifs jouent un rôle fondamental dans l’atteinte de résultats de bonne qualité. Ils ont également effectué une évaluation de leur méthode en utilisant une loss cyclique dans une seule direction, mettant en évidence qu’un seul cycle ne suffit pas à régulariser l’apprentissage pour ce problème.

6.3.3 Conclusion

En optimisant conjointement ces deux loss, le CycleGAN s’assure que les traductions ne sont pas seulement réalistes mais qu’elles conservent également la structure et le contenu inhérents aux images originales. Cette méthodologie permet diverses applications, du transfert de style à l’adaptation de domaine, sans qu’il soit nécessaire d’utiliser des échantillons d’entraînement appariés, constituant ainsi une avancée significative dans le domaine de la traduction d’image à image.

6.4 StarGAN

Parmi les différentes architectures de GAN, StarGAN [Choi *et al.*, 2018] a su démontrer son utilité, notamment pour la transformation d’images dans divers domaines.

Le StarGAN a été conçu afin de permettre la traduction d’une image vers plusieurs domaines à l’aide d’un unique modèle, évitant ainsi l’entraînement de réseaux distincts pour chaque paire de domaines. Ce point est une avancée significative, car les modèles précédents nécessitaient d’effectuer un apprentissage séparé, pour chaque domaine de translation, comme Pix2Pix [Isola *et al.*, 2018], DiscoGAN [Kim *et al.*, 2017], DualGAN [Yi *et al.*, 2017]. Le gain de temps d’entraînement est considérable lors de l’utilisation du StarGAN.

En 2020, les auteurs de StarGAN développent et mettent au point une nouvelle version appelée le StarGANv2 [Choi *et al.*, 2020]. Globalement, l’objectif reste inchangé, à savoir pouvoir utiliser un seul modèle, permettant de faire les transformations dans plusieurs domaines. Cette seconde version vise toutefois à améliorer la qualité visuelle ainsi que la diversité des images

générées. Nous nous concentrerons donc sur StarGANv2 dans la suite de ce manuscrit. Les principales évolutions entre les deux versions portant sur l'architecture du réseau, celle-ci est détaillée ci-après.

6.4.1 Architecture

Le fonctionnement global de StarGANv2 est représenté schématiquement dans la figure 6.6.

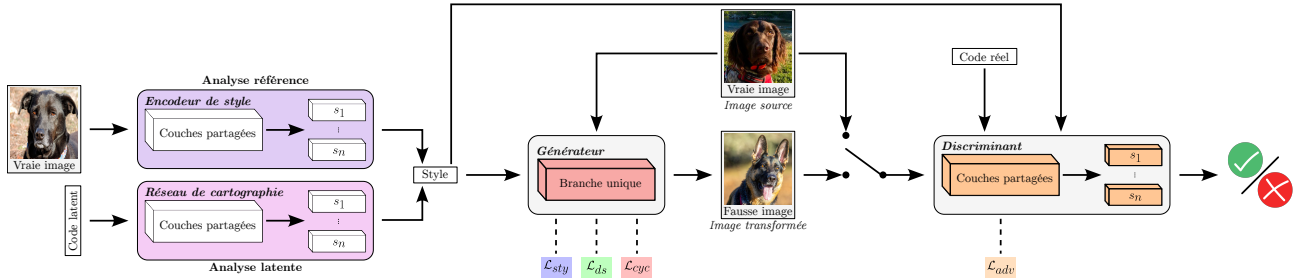


FIGURE 6.6 – Vue d'ensemble du StarGANv2, inspiré de l'article Grassucci et al. [2022]

La figure 6.6 présente une vue d'ensemble du fonctionnement de StarGANv2, illustrant le flux de données depuis l'image source jusqu'à l'image transformée, ainsi que les interactions entre les différents modules.

Deux modalités d'extraction du style y sont représentées : à partir d'une image de référence via un encodeur, ou à partir d'un vecteur latent via un réseau de cartographie. Le style ainsi obtenu est injecté dans le générateur afin de produire une image stylisée, tout en préservant le contenu sémantique de l'image d'origine.

L'image de référence exploitée par l'encodeur de style est indépendante de l'image source. Elle n'a pas pour rôle d'être transformée, mais uniquement de fournir un style visuel (textures, couleurs, motifs...) que le générateur applique ensuite à l'image source.

La figure met également en évidence les différentes fonctions de perte mobilisées : perte de style, perte de diversification et perte de cohérence cyclique, toutes calculées à partir des images générées.

Enfin, le discriminant évalue la vraisemblance des images produites et contribue à la perte adversaire. Cette représentation synthétique permet de visualiser les interactions entre les blocs fonctionnels et la dynamique d'apprentissage antagoniste qui en découle.

Les différents modules composant le StarGANv2 sont représentés dans la figure 6.7 et explicités ci-après.

Soit $\mathcal{X}$ et $\mathcal{Y}$ les ensembles d'images et de domaines possibles, respectivement. Étant donné une image $\mathbf{x} \in \mathcal{X}$ et un domaine arbitraire $y \in \mathcal{Y}$, le but de StarGANv2 est d'entraîner un seul générateur G (figure 6.7a), capable de générer diverses images de chaque domaine y correspondant à l'image $\mathbf{x}$. Des vecteurs de style spécifiques au domaine, sont générés dans l'espace de style (figure 6.7c) appris de chaque domaine, et G est un générateur conditionné par le style et est entraîné à refléter les vecteurs de style.

6.4.1.1 Le générateur

Le générateur G (figure 6.7a) reçoit en entrée une image $\mathbf{x}$ ainsi qu'un code de style s , et produit une image traduite $G(\mathbf{x}, s)$ correspondant à un domaine cible. Le style s est une représentation vectorielle compacte des caractéristiques visuelles globales d'un domaine (texture, éclairage, motifs dominants), obtenue soit à partir d'un vecteur latent projeté par un réseau de cartographie, soit à partir d'une image de référence encodée par un encodeur de style. Ces deux modules sont présentés dans les sous-sections suivantes.

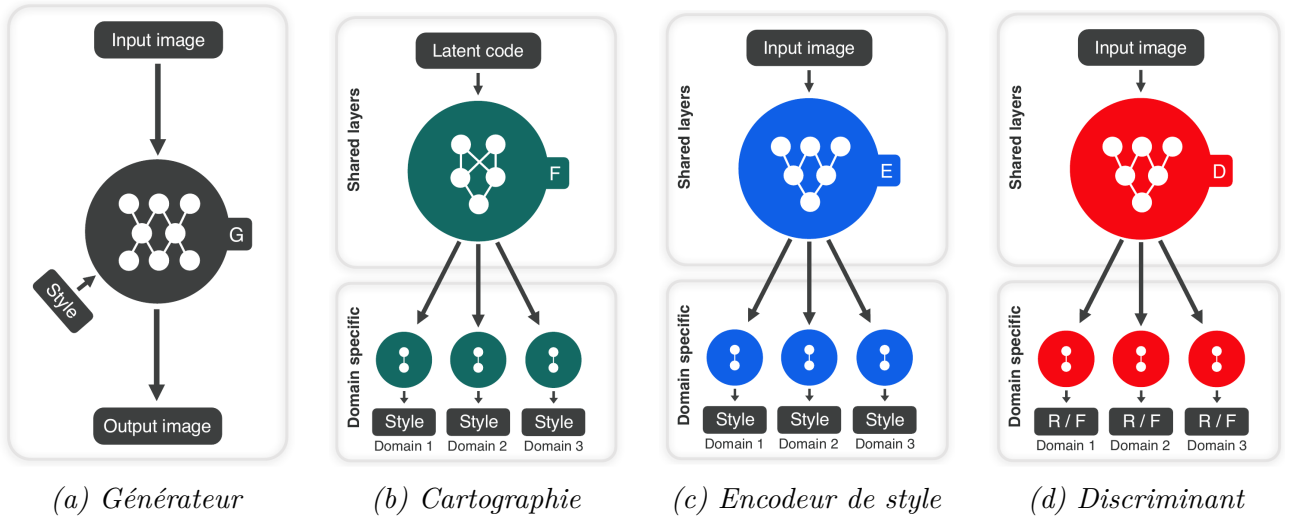


FIGURE 6.7 – Composition du StarGANv2 [Choi et al., 2020] avec ses quatre principaux modules.

Pour injecter efficacement le style dans le générateur, G utilise la technique Normalisation d'Instance Adaptative (AdaIN) [Huang et Belongie, 2017; Karras *et al.*, 2019], qui ajuste dynamiquement les statistiques de normalisation des activations en fonction du code de style. Cette opération permet de modifier l'apparence de l'image générée tout en préservant sa structure de contenu, ce qui est adapté aux tâches de transfert de style.

Contrairement à d'autres approches telles que StarGAN [Choi *et al.*, 2018], le générateur n'a pas besoin de connaître explicitement le label de domaine cible y , le style s encode déjà l'apparence caractéristique de ce domaine. Cela permet à G de synthétiser des images dans n'importe quel domaine en manipulant uniquement les codes de style.

6.4.1.2 Le réseau de cartographie

Le réseau de cartographie F (figure 6.7b) permet, à partir d'un code latent $\mathbf{z}$ et un domaine y , de générer un style $s = F_y(\mathbf{z})$, où $F_y(\cdot)$ désigne une sortie de F correspondant au domaine y . F est un Multi-Layer Perceptron (MLP) [Rumelhart *et al.*, 1986; Rosenblatt, 1958] avec de multiples branches de sortie, permettant de fournir des codes de style pour tous les domaines existants dans $\mathcal{Y}$. De ce fait, F peut produire divers codes de style, en échantillonnant le vecteur latent $\mathbf{z} \in \mathcal{Z}$ et le domaine $y \in \mathcal{Y}$ de manière aléatoire. L'architecture étant multi-tâches, il donne la possibilité à F d'apprendre efficacement les représentations de style de la totalité des domaines.

6.4.1.3 L'encodeur de style

L'encodeur de style E (figure 6.7c) capte en entrée une image $\mathbf{x}$ et son domaine correspondant y , pour extraire le style $s = E_y(\mathbf{x})$ de $\mathbf{x}$. Ici, $E_y(\cdot)$ désigne la sortie de E , correspondant au domaine y . Comme pour F , l'encodeur de style E bénéficie de la configuration de l'apprentissage multi-tâches, afin d'apprendre efficacement sur l'ensemble des domaines. E peut produire divers codes de style, en utilisant différentes images de références, permettant ainsi à G de synthétiser une image de sortie qui reflète le style s d'une image de référence $\mathbf{x}$.

6.4.1.4 Le discriminant

Le discriminant D (figure 6.7d) est également multi-tâches. Il existe de nombreux points positifs à ce que le discriminant soit multi-tâches, notamment une meilleure généralisation du modèle, éviter un sur-apprentissage grâce au phénomène de régularisation [Aggarwal *et al.*, 2017] et donc une meilleure convergence [Liu, Huang, Mallya, Karras, Aila, Lehtinen, et Kautz, 2019; Mescheder, Geiger, et Nowozin, 2018]. Le discriminant possède plusieurs branches de sortie, chaque branche D_y apprenant une classification binaire, qui détermine si une image $\mathbf{x}$ est une image réelle de son domaine y , ou une fausse image $G(\mathbf{x}, s)$ ayant été produite par G .

6.4.2 Apprentissage

L'entraînement du modèle peut se décomposer en plusieurs parties avec des objectifs propres à chacune. Afin d'obtenir un rendu esthétique des images de sortie, tout en intégrant les différents styles demandés, il convient d'avoir des fonctions objectifs se focalisant sur les rendus. La diversification est un point mis en avant par les auteurs; une perte de diversification de style est introduite pour compléter la perte de reconstruction de style.

6.4.2.1 Loss adversariale

Pendant l'entraînement, un code latent $\mathbf{z} \in \mathcal{Z}$ est échantillonné ainsi qu'un domaine cible $\tilde{y} \in \mathcal{Y}$ de manière aléatoire. Un code de style cible est ensuite généré $\tilde{s} = F_{\tilde{y}}(\mathbf{z})$. Le générateur G prend une image $\mathbf{x}$ et $\tilde{s}$ comme données en entrée, et apprend à générer une image de sortie $G(\mathbf{x}, \tilde{s})$, grâce à la fonction de perte adversariale suivante :

$$\mathcal{L}_{adv} = \mathbb{E}_{\mathbf{x}, y}[\log D_y(\mathbf{x})] + \mathbb{E}_{\mathbf{x}, \tilde{y}, \mathbf{z}}[\log(1 - D_{\tilde{y}}(G(\mathbf{x}, \tilde{s})))] \quad (6.9)$$

où $D_y(\cdot)$ désigne la sortie de D correspondant au domaine y . Le réseau de cartographie F apprend à fournir le code de style $\tilde{s}$, similaire au domaine cible $\tilde{y}$, et G apprend à utiliser $\tilde{s}$ et à générer une image $G(\mathbf{x}, \tilde{s})$, ne pouvant être distinguée des images réelles du domaine $\tilde{y}$.

6.4.2.2 Loss pour la reconstruction de style

Dans un but de forcer le générateur G à utiliser un style de code $\tilde{s}$ pendant la génération de l'image $G(\mathbf{x}, \tilde{s})$, la perte de reconstruction de style a été utilisée et est la suivante :

$$\mathcal{L}_{sty} = \mathbb{E}_{\mathbf{x}, \tilde{y}, \mathbf{z}}[\|\tilde{s} - E_{\tilde{y}}(G(\mathbf{x}, \tilde{s}))\|_1] \quad (6.10)$$

Cette approche se base sur de précédents travaux [Huang *et al.*, 2018; Zhu *et al.*, 2018] utilisant plusieurs encodeurs, afin d'apprendre à transformer une image en son code latent. La principale différence consiste dans le fait que le StarGANv2, n'a qu'un seul encodeur E , pour créer de la diversité dans les résultats de plusieurs domaines. Lors du test, l'encodeur E entraîné permet à G de transformer une image d'entrée, en reflétant le style d'une image de référence.

6.4.2.3 Loss pour la diversification de style

Pour inciter le générateur G à produire des images stylistiquement variées tout en préservant le contenu, une régularisation est introduite via la *diversity-sensitive loss* [Mao *et al.*, 2019; Yang *et al.*, 2019] :

$$\mathcal{L}_{ds} = \mathbb{E}_{\mathbf{x}, \tilde{y}, \mathbf{z}_1, \mathbf{z}_2}[\|G(\mathbf{x}, \tilde{s}_1) - G(\mathbf{x}, \tilde{s}_2)\|_1] \quad (6.11)$$

où l'image source est la même $\mathbf{x}$, mais avec deux codes de style différents $\tilde{s}_1$ et $\tilde{s}_2$, obtenus à partir des vecteurs latents $\mathbf{z}_1$ et $\mathbf{z}_2$ via :

$$\tilde{s}_i = F_{\tilde{y}}(\mathbf{z}_i), \quad i \in \{1, 2\}.$$

Cette fonction de perte mesure la dissimilarité entre les deux images générées. En maximisant cette régularisation, le générateur est contraint d'explorer l'espace des styles et d'en extraire des caractéristiques distinctives, favorisant ainsi une plus grande diversité dans les images synthétisées.

6.4.2.4 Loss pour préserver les caractéristiques de l'image source

Pour garantir que l'image générée $G(\mathbf{x}, \tilde{s})$ préserve les caractéristiques essentielles de l'image source $\mathbf{x}$, telles que la structure ou certaines propriétés visuelles (par exemple, la couleur des cheveux dans le cas de visages humains), une contrainte de cohérence cyclique est introduite [Zhu *et al.*, 2017; Kim *et al.*, 2017; Choi *et al.*, 2018].

Ce principe repose sur l'idée qu'une translation de style vers un domaine cible $\tilde{s}$, suivie d'une translation inverse vers le domaine source $\hat{s}$, doit restituer une image proche de l'originale. Cette idée se formalise par la perte suivante :

$$\mathcal{L}_{cyc} = \mathbb{E}_{\mathbf{x}, y, \tilde{y}, \mathbf{z}} = [\|\mathbf{x} - G(G(\mathbf{x}, \tilde{s}), \hat{s})\|_1] \quad (6.12)$$

où $\tilde{s}$ désigne le style cible et $\hat{s}$ le style source associé à $\mathbf{x}$. L'utilisation de la norme ℓ_1 favorise la préservation des structures fines par rapport à une norme quadratique.

Cette perte est cruciale afin de garantir que les attributs intrinsèques de l'image source, restent intacts durant le processus de transformation. Cette perte garantit, malgré que le style ou les caractéristiques spécifiques au domaine de l'image soient transformés, que le contenu fondamental ou les caractéristiques inhérentes de l'image source sont conservés.

En minimisant cette perte, le modèle vise à produire des images transformées qui, non seulement adoptent le style ou les attributs du domaine souhaité, mais préservent également les caractéristiques uniques de l'image originale, réalisant ainsi un équilibre entre la transformation et la conservation.

Cet équilibre est essentiel pour produire des transformations d'images réalistes et cohérentes conservant à la fois l'identité et le contexte de la source, tout en reflétant les caractéristiques du domaine cible.

6.4.3 Objectif global

La fonction objectif global peut être synthétisée en fusionnant toutes les composantes vues précédemment. Cela s'apparente au jeu min-max, dû à la dynamique d'entraînement du réseau, StarGANv2 étant contradictoire entre le générateur G et le discriminateur D . Dans ce cadre théorique, le générateur G et le discriminateur D ont des objectifs opposés, et ils sont entraînés itérativement de manière à ce que chacun essaie de surpasser l'autre.

Le "min", dans le jeu min-max, fait référence à la tentative du générateur de minimiser sa fonction de perte (ou la différence entre ses images générées et les images réelles), tandis que le "max" fait référence à la tentative du discriminateur de maximiser sa capacité à classer correctement les images, comme étant réelles ou générées.

L'équation permettant de régir l'entraînement du réseau complet, peut s'écrire de la manière suivante :

$$\min_{G,F,E} \max_D \mathcal{L}_{adv} + \lambda_{sty} \mathcal{L}_{sty} - \lambda_{ds} \mathcal{L}_{ds} + \lambda_{cyc} \mathcal{L}_{cyc} \quad (6.13)$$

avec λ_{sty} , λ_{ds} et λ_{cyc} des hyperparamètres pour chaque terme. En ajustant les valeurs de ces coefficients, il est possible d'affiner le comportement du modèle, en mettant l'accent sur certains aspects du processus de transformation d'images plutôt que sur d'autres. Permettant ainsi un processus personnalisé, garantissant que le modèle produit des résultats, se rapprochant au plus près des résultats souhaités.

Le terme de diversité de style $\mathcal{L}_{ds}$ est intégré avec un signe négatif car il est formulé comme une fonction que le réseau cherche à maximiser afin de générer des images variées à partir d'un même contenu. Dans la formulation globale, ce signe négatif transforme cette maximisation en contribution à minimiser dans la fonction de coût totale, garantissant ainsi que la diversité des styles est favorisée tout en respectant les autres contraintes de la perte.

La figure 6.6 est structurée de manière à illustrer le flux de données et les interactions entre les différents composants du modèle.

Un code de style est indispensable pour le bon fonctionnement du réseau. Pour l'obtenir, deux possibilités existent. La première consiste à fournir une image à l'encodeur de style, capable de générer un code de style. La seconde possibilité réside dans le fait de donner un code latent au réseau de cartographie, qui va lui aussi obtenir un code de style.

En combinant une image réelle ainsi que le style fourni, le générateur va alors pouvoir créer une nouvelle image en gardant le contenu sémantique de l'image originale tout en appliquant le style demandé. C'est à ce moment que les pertes de style (équation 6.10), de diversification (équation 6.11) et de conservation de contenu (équation 6.12) sont calculées pour l'entraînement.

Concernant le discriminant, correspondant à la dernière partie du processus d'entraînement, il prend en entrée une image, qui peut être soit une vraie image, soit une image produite par le générateur. Son rôle est d'essayer de déterminer l'origine de l'image sans se tromper. Grâce à sa réponse, la perte adversariale (équation 6.9) est alors calculée.

La dynamique contradictoire entre toutes les parties du réseau, se poursuit de manière itérative, chaque composante s'améliorant en réponse à l'autre, ce qui conduit à un scénario dans lequel le générateur devient de plus en plus compétent pour produire des traductions de haute qualité, et le discriminateur devient meilleur pour distinguer les images réelles des images générées. L'équilibre de ce jeu est atteint, lorsque le générateur produit des images que le discriminateur ne peut plus distinguer de manière fiable, des images réelles, aboutissant à un équilibre harmonisé dans le processus de formation contradictoire.

Chapitre 7

Adaptation des GAN aux contraintes imposées

Sommaire

7.1	Modifications apportées	126
7.1.1	CycleGAN	126
7.1.2	StarGAN	129
7.1.2.1	Utilisation d'un masque	130
7.1.2.2	Ajout d'un détecteur YOLO	130
7.1.2.3	Changement de l'encodeur de style	131
7.2	Résultats	132
7.2.1	Cycle GAN	132
7.2.2	StarGANv2	135

7.1 Modifications apportées

Cette section détaille les différentes modifications apportées aux réseaux vus précédemment afin de les adapter à notre scénario : générer des images avec des objets qui soient aux bons endroits et correctement générés.

7.1.1 CycleGAN

Dans le cadre de l'amélioration du CycleGAN, l'attention a été portée sur les discriminants plutôt que sur les générateurs. Bien que l'entraînement repose sur l'interaction entre ces deux composants, renforcer les discriminants permet d'évaluer plus finement les sorties des générateurs et de mieux orienter leur apprentissage.

Une modification pertinente consiste à ajouter un masque binaire en entrée du générateur (F ou G), en complément de l'image initiale. Ce masque encode explicitement la position attendue de l'objet cible dans l'image générée à partir de la vérité terrain. Cette contrainte vise à orienter la génération de manière à faire apparaître l'objet à l'emplacement spécifié, tout en respectant le style visuel du domaine cible. La figure 7.1 illustre cette intégration.

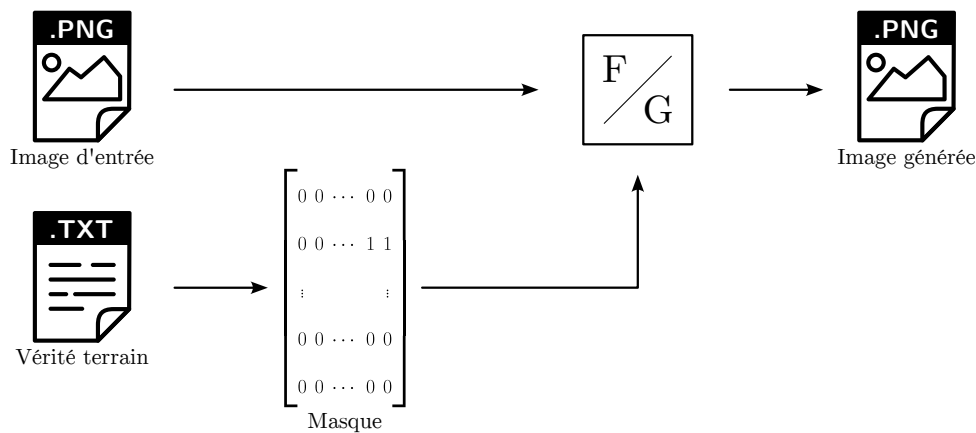
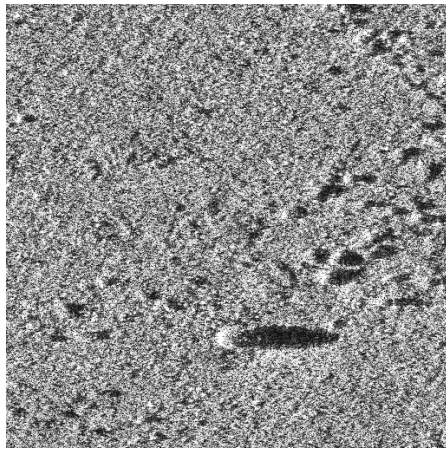


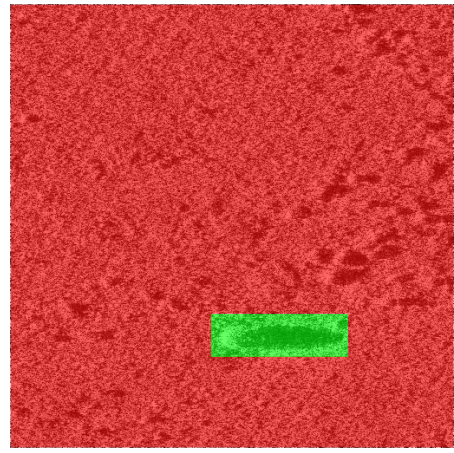
FIGURE 7.1 – Schéma illustratif du CycleGAN montrant l'intégration du masque.

La figure 7.2 montre l'exemple d'un masque (figure 7.2b) surimprimé sur une image originale, qui est utilisé durant l'apprentissage.

Cette approche s'applique facilement à chaque image, dans la mesure où elle repose sur la vérité terrain associée.



(a) Image originale.



(b) Masque correspondant.

FIGURE 7.2 – Illustration d’un masque utilisé pour l’apprentissage du CycleGAN. La zone en rouge correspond aux valeurs mises à 0, et celle en vert aux valeurs à 1.

De nouveaux discriminants ont été ajoutés, pour chaque sens de transformation, intégrant un réseau U-Net [Ronneberger *et al.*, 2015], une architecture largement utilisée pour les tâches de segmentation d’images. L’utilisation d’un U-Net permet de réaliser une prédiction par pixel.

Cette prédiction va servir pour calculer une nouvelle perte. Cette perte utilise un masque appliqué sur les images analysées pour vérifier que les objets présents dans l’image initiale sont également présents dans l’image générée. Cela permet de guider le générateur pour qu’il conserve les caractéristiques essentielles des objets d’intérêt lors de la génération des images.

Les pertes composant le CycleGAN classique sont détaillées dans la partie 6.3.2. La perte que nous avons ajoutée est la Categorical Cross-Entropy (CCE) entre la prédiction du nouveau discriminant et le masque dont les valeurs sont à 1 dans la zone de l’objet et 0 tout autour. L’équation est la suivante :

$$\mathcal{L}_{\text{CCE}} = - \sum_i \sum_{c=1}^C y_{i,c} \log(p_{i,c}) \quad (7.1)$$

où $\mathcal{L}_{\text{CCE}}$ est la perte CCE, $y_{i,c}$ est la valeur du masque pour le pixel i et la classe c . Dans notre cas $y_{i,c}$ vaudra 1 si le pixel i appartient à la zone de l’objet et 0 sinon. $p_{i,c}$ est la probabilité prédite par le discriminateur que le pixel i appartienne à la classe c , et C est le nombre total de classes.

Dans notre cas, la tâche consiste uniquement à distinguer la présence ou l’absence d’un objet, sans différencier les types d’objets. Le problème se ramène donc à une classification binaire, pour laquelle la fonction de perte adaptée est la BCE :

$$\mathcal{L}_{\text{BCE}} = - \frac{1}{N} \sum_{i=1}^N y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (7.2)$$

où N désigne le nombre total de pixels, $y_i \in \{0, 1\}$ indique la présence de l’objet au pixel i , et p_i la probabilité prédite que ce pixel appartienne à la classe “objet”. Cette formulation simplifie l’apprentissage en imposant uniquement la contrainte objet/fond, sans distinction de catégorie.

Un défi majeur réside dans le fort déséquilibre des données, les images ne contenant que du fond étant largement majoritaires. Pour réduire la perte, le générateur pourrait alors privilégier la génération de fond, en omettant l’objet.

Afin de limiter ce biais, une pondération a été appliquée au niveau des échantillons, plutôt qu'aux pixels ou aux classes, attribuant ainsi davantage de poids aux images contenant des objets lors de l'entraînement.

Une fois l'apprentissage terminé, le générateur, ayant appris la transformation du domaine simulé vers le domaine réel, est utilisé pour produire une nouvelle base de données.

La figure 7.3 illustre plusieurs exemples de cette transformation. La ligne supérieure correspond aux images originales issues du domaine simulé, tandis que la ligne inférieure présente les images traduites dans le domaine réel.

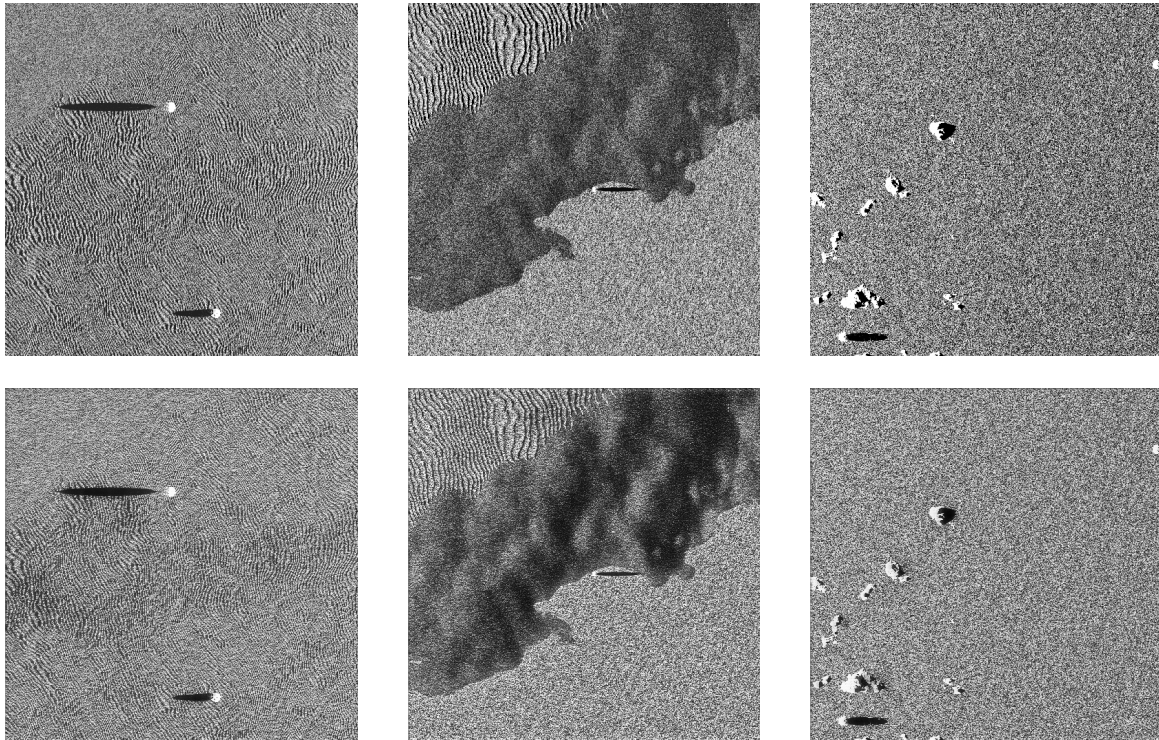


FIGURE 7.3 – Exemples de transformations encourageantes effectuées par un CycleGAN.

Bien que certains résultats soient visuellement satisfaisants, d'autres présentent des défauts, difficilement discernables à l'œil. La figure 7.4 montre plusieurs exemples de transformations problématiques, avec les artefacts indiqués en rouge. Les grands encadrés rouges correspondent à un zoom sur les zones indiquées par les petits encadrés, afin de mettre en évidence les artefacts générés.

Ces artefacts, difficilement contrôlables dans l'architecture CycleGAN, altèrent la qualité des images générées. Comme illustré sur la figure 7.4 (flèches rouges), plusieurs défauts sont observés : à gauche, des ombres artificielles apparaissent dans des zones qui ne devraient pas en contenir ; au centre, une dégradation des textures rocheuses rend la surface incohérente en plus d'une accentuation de l'ombre ; à droite, les ombres sont plus sombres.

Ces erreurs résultent du fonctionnement du CycleGAN, qui effectue principalement une translation de style global sans contrainte explicite sur la cohérence locale ou physique de la scène. Les pertes de cycle assurent une correspondance statistique entre les domaines, mais ne préservent pas fidèlement les structures fines telles que les ombres ou les textures. En imagerie sonar, où les données réelles sont rares et variables, le générateur apprend souvent des correspondances approximatives, favorisant ainsi l'apparition d'artefacts.

La qualité visuelle ne constitue pas l'unique critère d'évaluation. Dans un pipeline de détection, les ombres et les textures jouent un rôle essentiel en tant qu'indices discriminants

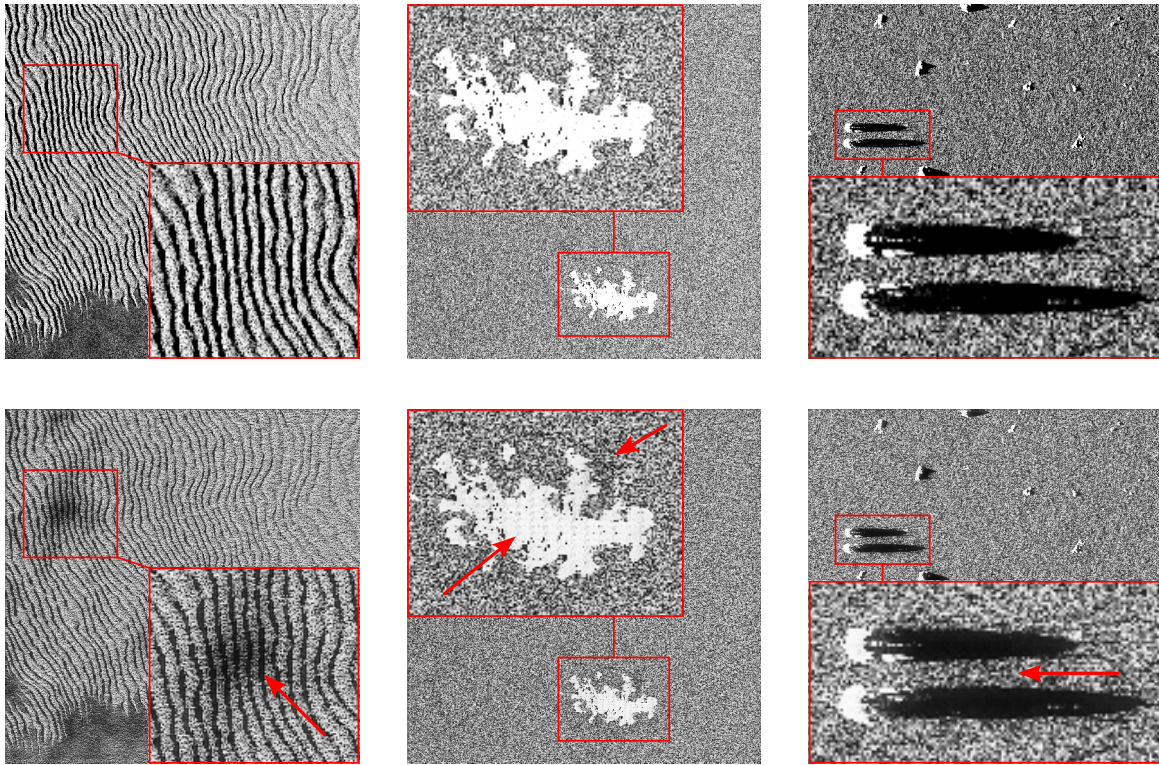


FIGURE 7.4 – Exemples de transformations problématiques effectuées par un CycleGAN.

pour la classification et la localisation des objets. Des défauts apparemment mineurs pour un observateur humain peuvent ainsi provoquer d'importantes erreurs lors de l'apprentissage d'un détecteur automatique.

Afin d'explorer des architectures plus flexibles, des modifications ont également été appliquées au réseau StarGAN, dont la structure modulaire offre un plus grand potentiel d'adaptation.

7.1.2 StarGAN

Les modifications apportées dans le CycleGAN ont également été reprises dans le StarGANv2, auquel s'ajoutent des ajustements structurels visant à pallier certaines limites des approches de translation non supervisée. Le CycleGAN génère typiquement une seule image cible par image source, sans mécanisme explicite de contrôle sur le style, ce qui restreint la diversité des sorties et la possibilité de guider l'apparence cible. Alors que le StarGANv2 introduit un encodeur de style ainsi qu'un générateur conditionnel, permettant de spécifier le style souhaité et de produire des traductions plus variées. Ces choix cherchent à dépasser les contraintes liées à la contrainte de cycle tout en préservant les avantages de la translation non appariée.

Les travaux présentés dans la suite de cette section proposent diverses extensions du StarGANv2. Bien que certaines visent à améliorer la fidélité visuelle, la stabilité d'entraînement ou l'adaptabilité à des contextes spécifiques, leur évaluation reste limitée.

Les sections suivantes constituent un travail prospectif et préliminaire, présentant quelques pistes d'adaptation possibles. Ces propositions ne relèvent pas de contributions établies et n'ont pas été évaluées, elles doivent donc être considérées avec prudence.

7.1.2.1 Utilisation d'un masque

Comme mentionné précédemment, le calcul du masque repose uniquement sur la position de l'objet dans la vérité terrain, ce qui en fait une opération simple. Étant donné que cette vérité terrain reste constante pour chaque image, les masques ont été pré-calculés et intégrés aux données, afin de limiter la charge de calcul durant l'entraînement.

Contrairement au CycleGAN, un masque est fourni en entrée du générateur afin de focaliser la transformation sur une zone ciblée de l'image.

Ce masque est exploité à différentes résolutions et injecté à plusieurs étapes du générateur, comme illustré dans la figure 7.5.

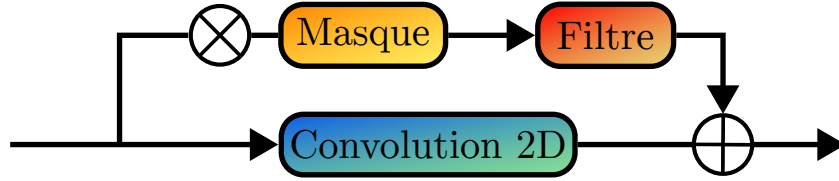


FIGURE 7.5 – Illustration d'un bloc utilisant le masque d'une image.

Tout comme pour le CycleGAN, la multiplication avec le masque permet d'isoler l'information pertinente, en l'occurrence la représentation de l'objet. Le filtre utilisé dans ce bloc est celui du StarGANv2 d'origine, un filtre passe-haut conçu pour faire ressortir les détails fins et les contours.

Ce mécanisme vise à contraindre le générateur, malgré la profondeur de son architecture convolutionnelle, à se concentrer sur une région spécifique de l'image source. Afin de s'assurer que cette zone soit correctement restituée en sortie, un détecteur peut être ajouté, comme présenté dans la section suivante.

7.1.2.2 Ajout d'un détecteur YOLO

L'intégration d'un détecteur dans le processus d'apprentissage vise à garantir la cohérence sémantique et structurelle des objets générés. Dans le cadre de la génération d'images destinée à renforcer les performances d'un modèle de détection, il est important que les objets d'intérêt soient à la fois réalistes, correctement localisés et conformes aux catégories cibles. Cette contrainte permet de préserver la vérité terrain des images simulées tout en transférant l'apparence du domaine réel.

Le réseau ajouté est un détecteur de type YOLO, pré-entraîné sur des données réelles et capable de détecter les objets dans la majorité des images. En complément des pertes de StarGANv2 décrites en 6.4.2, la perte globale du détecteur YOLO (voir la section 3.3.2.3) est incorporée, pondérée par un facteur λ contrôlant son impact sur l'apprentissage.

La fonction de perte globale est alors définie comme suit :

$$loss = \lambda_{sty} \times loss_{sty} - \lambda_{ds} \times loss_{ds} + \lambda_{cyc} \times loss_{cyc} + \lambda_{yolo} \times loss_{yolo} + loss_{adv} \quad (7.3)$$

Le détecteur YOLO utilisé dans l'approche est maintenu fixe durant l'entraînement : ses poids ne sont pas mis à jour et il intervient uniquement en inférence pour fournir une contrainte externe au générateur. Le générateur ne peut ainsi pas s'appuyer sur un détecteur adaptatif

qui corrigerait ses erreurs, mais doit produire des objets réalistes et détectables par un modèle indépendant.

Le détecteur choisi est un YOLOv5 pré-entraîné sur le jeu d'images réelles sonar. Sur ce jeu, il atteint une Mean Average Precision (mAP)@0.5 de 0.67, une mAP@[0.5:0.95] de 0.34, une précision de 0.63 et un rappel de 0.69. Son rôle est de vérifier la présence, la localisation et la détectabilité des objets dans les images synthétiques, sans intervenir directement dans l'optimisation du GAN, en dehors de la pénalisation induite par la perte associée.

L'ajout de la perte YOLO permet de vérifier que l'image générée contient l'objet attendu, sans introduire d'objets non présents dans la vérité terrain. Il permet également de s'assurer que l'objet généré est détectable comme une entité réelle. Dans le cas contraire, la perte associée pénalise l'échec de détection afin d'inciter le générateur à produire des objets de qualité suffisante.

Cette approche présente toutefois une limite notable. L'efficacité de la perte YOLO dépend des performances du détecteur employé, lequel doit être suffisamment entraîné dans le domaine cible pour produire des prédictions fiables. Or, atteindre ce niveau nécessite un volume conséquent d'annotations réelles, en contradiction avec l'objectif de réduire la dépendance aux données annotées. Cette contrainte induit une circularité : l'amélioration de la générativité repose sur un détecteur robuste, dont l'entraînement est lui-même freiné par la rareté des annotations.

L'intégration d'un détecteur d'objets de type YOLO dans un entraînement conjoint avec un GAN introduit une complexité supplémentaire. Le générateur doit alors satisfaire, en plus des contraintes adversariales et de reconstruction, des contraintes sémantiques induites par les prédictions du détecteur. Cette interaction rend l'équilibre de l'apprentissage, déjà délicat dans un GAN, encore plus difficile à atteindre, en particulier dans un contexte de données limitées et de forte variabilité tel que l'imagerie sonar.

Pour atténuer cette instabilité et limiter les risques de divergence, une simplification de l'encodeur de style est envisagée. Cette adaptation vise à réduire le nombre de paramètres et la complexité des gradients, tout en conservant la capacité à moduler le style dans les traductions inter-domaines. Elle s'inscrit dans une démarche d'allègement du modèle afin d'en favoriser la convergence.

7.1.2.3 Changement de l'encodeur de style

Dans StarGANv2, l'encodeur de style joue un rôle central dans la manipulation des attributs visuels d'une image, tout en préservant sa structure et son contenu. Il extrait un vecteur latent à partir de l'image d'entrée, capturant des informations stylistiques. Ce vecteur guide ensuite le générateur pour produire des images présentant les attributs souhaités, permettant une transformation contrôlée et flexible. En dissociant style et contenu, StarGANv2 réalise des traductions d'image à image variées.

Remplacer l'encodeur de style par des statistiques propres à chaque image pourrait améliorer à la fois les performances et la stabilité du réseau. Des mesures telles que la moyenne, l'écart-type, la médiane, la kurtosis, ou la skewness permettent de caractériser finement la distribution des intensités. L'intégration de fonctions de distribution cumulative (CDF) et de matrices de co-occurrence enrichit cette représentation en capturant des informations spatiales et contextuelles [Mroueh *et al.*, 2017 ; Nataraj *et al.*, 2019 ; Barni *et al.*, 2020].

Ce type de caractérisation, directement calculée à partir des données, réduit la complexité architecturale, pourrait améliorer la stabilité de l'apprentissage, et favoriser une meilleure généralisation face aux variations d'entrée.

Pour renforcer cette approche, l'ajout d'une perte basée sur la "*Structural Similarity Index Measure (SSIM)*" permettrait de préserver la structure perceptuelle des images en complément des statistiques extraites.

L'intégration de la SSIM pour évaluer la qualité des images générées s'appuie sur sa capacité à mesurer les altérations structurelles en termes de luminance, de contraste et de corrélation locale, en accord avec la perception humaine. Dans le cas de l'imagerie sonar, cette métrique est particulièrement adaptée, car elle repose sur l'hypothèse que le bruit doit conserver localement des propriétés statistiques similaires (moyenne et variance). La SSIM, calculée sur des fenêtres locales, permet ainsi de vérifier la cohérence du bruit généré.

Contrairement à la "*Mean Square Error (MSE)*", qui évalue uniquement l'écart quadratique moyen entre intensités pixel à pixel sans considérer la structure statistique du bruit, la SSIM prend en compte des informations perceptuelles et structurelles. Elle permet notamment de détecter des dégradations telles que le flou ou la perte d'ombres caractéristiques, ce qui en fait un outil plus pertinent pour juger de la fidélité visuelle des images sonar.

Son utilisation dans la fonction de perte, notamment pour les pertes de cycle et au sein des discriminateurs, permet d'améliorer la fidélité perceptuelle des images générées, tout en limitant la complexité de l'apprentissage.

7.2 Résultats

Deux ensembles distincts ont été constitués pour chaque domaine (réel et simulé) : un ensemble d'entraînement et un ensemble de validation, ce dernier servant également à l'évaluation finale.

L'ensemble réel compte 7308 images pour l'entraînement et 2436 pour la validation, tandis que l'ensemble simulé en contient 37968 pour l'entraînement et 1008 pour la validation. Le peu d'images simulées dans la partie validation est dû au fait que les résultats sur les images réelles sont plus importants. Cette organisation préserve le déséquilibre initial, avec une proportion d'images simulées nettement supérieure à celle des images réelles.

7.2.1 Cycle GAN

La figure 7.6 illustre une restitution correcte des grandes structures de la scène, mais des détails peu réalistes subsistent : un lissage excessif du bruit et des ombres mal positionnées ou artificielles. Ces limites montrent que l'amélioration de la structure globale ne s'accompagne pas d'une convergence satisfaisante vers le style visuel du domaine cible.

Une analyse qualitative des images générées met en évidence plusieurs cas de figure. Certaines images présentent des résultats satisfaisants : les objets sont correctement positionnés, sans déplacement notable, et leur forme globale est préservée.

En revanche, des éléments visuels tels que les ombres, les transitions d'intensité ou le bruit sonar ne sont pas correctement restitués. Ces défauts localisés, visible dans la figure 7.7 peuvent néanmoins compromettre l'exploitation des images pour des tâches de détection.

Cette figure illustre un autre défaut : la génération incorrecte des échos. L'écho associé aux objets est parfois absent ou mal reproduit. Or, cet écho est essentiel à la détection, car il permet de distinguer l'objet de son environnement et d'attester de sa présence. Une restitution dégradée ou incohérente compromet donc directement l'utilité des images générées pour l'entraînement, même lorsque la position globale de l'objet est conservée.

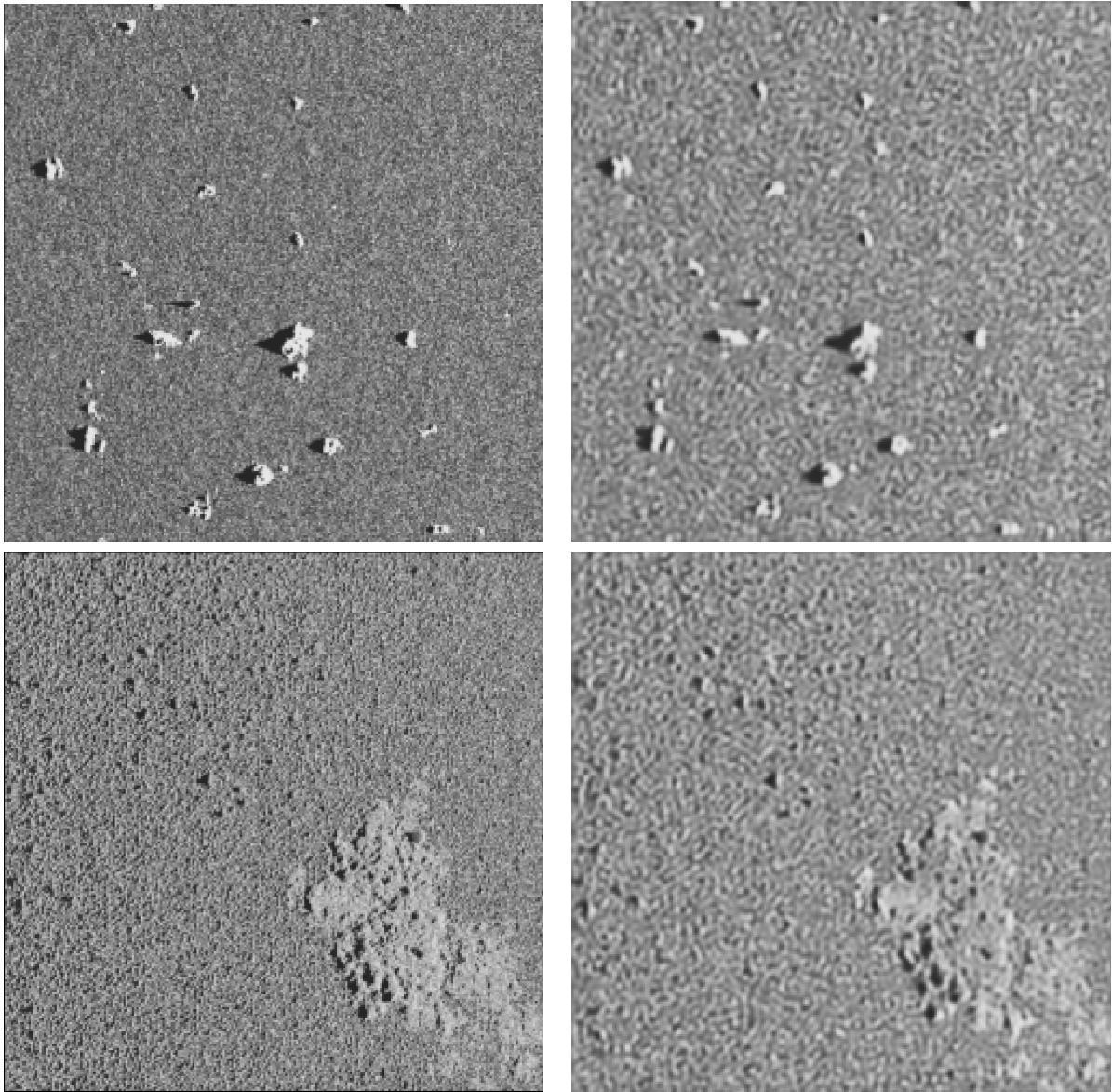
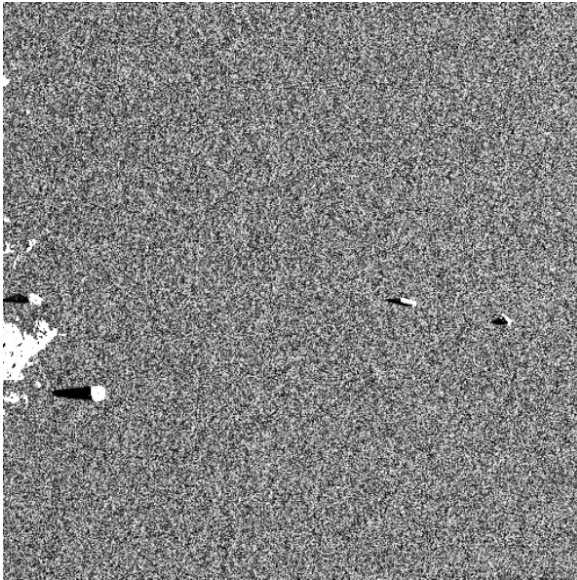


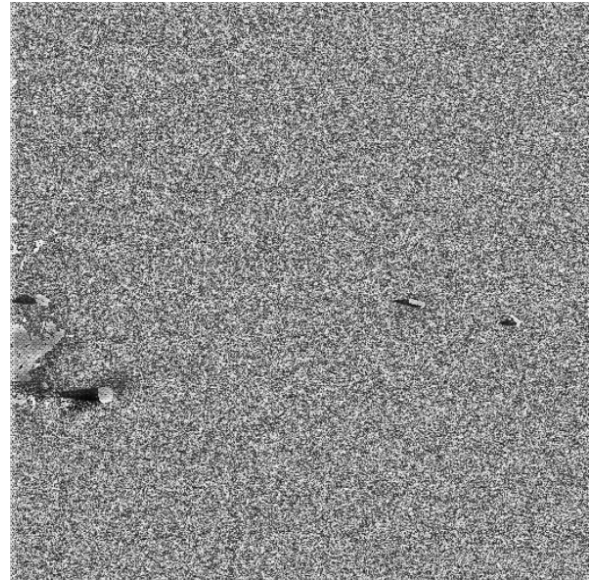
FIGURE 7.6 – Images originales et générées par le CycleGAN (400 étapes).

Dans d'autres cas, les résultats se dégradent nettement. Certaines images traduites présentent des artefacts visuels importants, sans correspondance plausible avec le domaine cible. La figure 7.8 illustre ce phénomène : une scène simulée avec un fond sableux est transformée en une image incohérente, contenant des structures artificielles non interprétables. De tels défauts rendent l'image inutilisable dans un pipeline de détection.

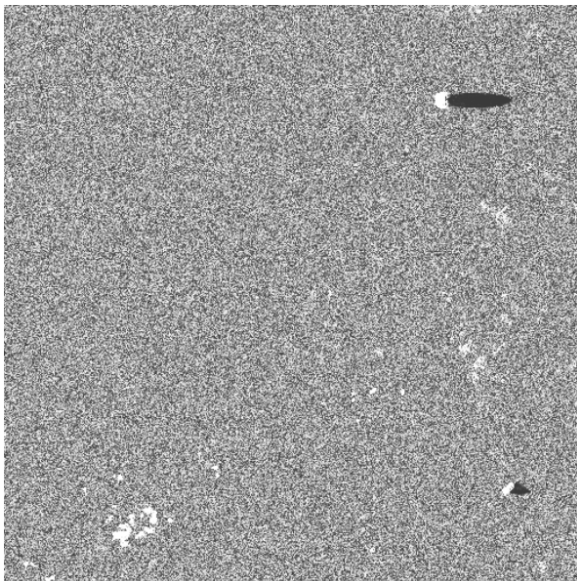
Pour évaluer l'utilité des images générées dans un cadre applicatif, des expériences ont été conduites en les intégrant à l'ensemble d'apprentissage d'un détecteur YOLOv5. Ces essais ont révélé une dégradation rapide des performances. En présence de ces images synthétiques, le détecteur diverge progressivement et perd la capacité à identifier les objets qu'il détectait auparavant sur les données simulées non transformées. Ce comportement indique que les images générées sont perçues comme hors distribution, malgré une certaine ressemblance visuelle avec les données réelles. L'apprentissage semble alors se focaliser sur la distinction entre images générées et images simulées, plutôt que sur leur contenu sémantique, compromettant ainsi l'objectif de transfert.



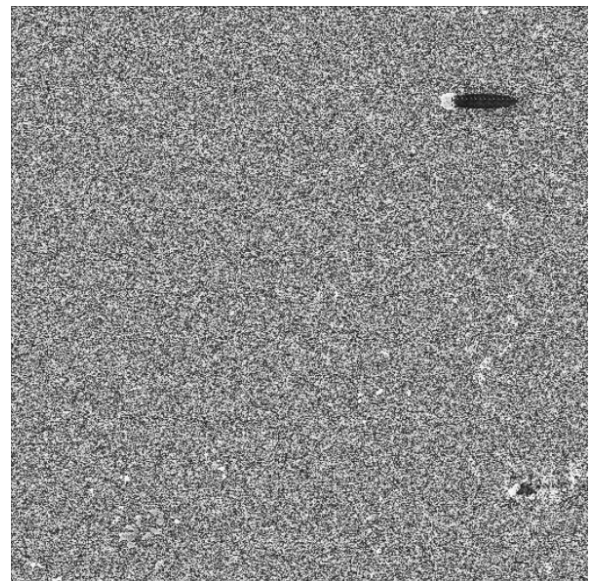
(a) Image 1 originale



(b) Image 1 générée

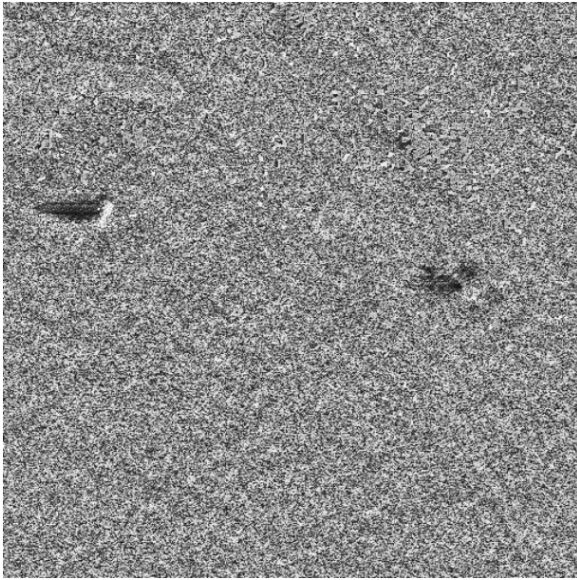


(c) Image 2 originale

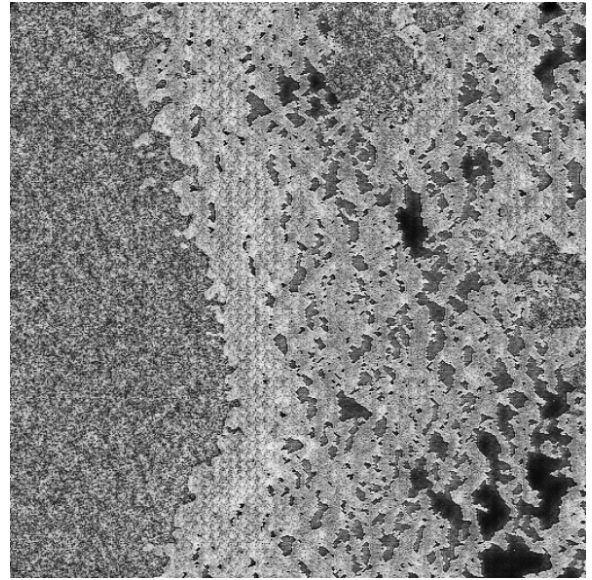


(d) Image 2 générée

FIGURE 7.7 – Exemples d’images générées par le CycleGAN où la position des objets est correcte, mais les ombres sont mal reconstruites.



(a) Image originale



(b) Image générée

FIGURE 7.8 – Exemple d’image traduite présentant un artefact sévère, altérant totalement la structure visuelle de la scène.

Bien que les performances qualitatives soient dégradées, ces résultats permettent d’identifier les limites des GAN dans un contexte aussi contraint que celui de l’imagerie sonar. Ils montrent que la conservation de la position des objets, bien qu’essentielle, ne suffit pas à assurer l’utilisabilité des images générées. La fidélité du bruit sonar, la structure des ombres et la cohérence des textures apparaissent comme des facteurs déterminants dans la perception d’une image par un détecteur automatique.

Ces observations mettent en évidence les limites des modèles adversariaux classiques, y compris modifiés, face aux spécificités visuelles et statistiques des images sonar réelles. Elles motivent l’exploration d’architectures plus flexibles, telles que StarGANv2, intégrant des contraintes de style et structurelles plus fines dans le processus de génération.

7.2.2 StarGANv2

L’architecture StarGANv2 a été explorée pour dépasser les limites observées avec CycleGAN, notamment en termes de contrôle du style, de fidélité visuelle et de stabilité d’apprentissage. Nous avons ajouté plusieurs modules : un encodeur de style permettant de moduler l’apparence des images générées, l’intégration de masques pour contraindre spatialement la génération, l’ajout d’un détecteur YOLO dans la fonction de perte afin de garantir la validité sémantique des objets, ainsi que l’utilisation de la SSIM comme perte de structure perceptuelle.

Contrairement à CycleGAN, qui génère une seule sortie par image, StarGANv2 autorise la spécification d’un style cible, permettant ainsi de produire plusieurs variantes d’une même image source. La figure 7.9 illustre ce principe dans un cadre de transformation *sim2real*¹. Chaque image de la première ligne provient du simulateur, tandis que chaque image de la première colonne correspond à une image réelle. Leur combinaison produit une image synthétique simulée adoptant le style visuel du domaine réel.

1. “sim2real” est le nom de la transformation partant d’une image simulée pour être transformée en une image avec le style réel.

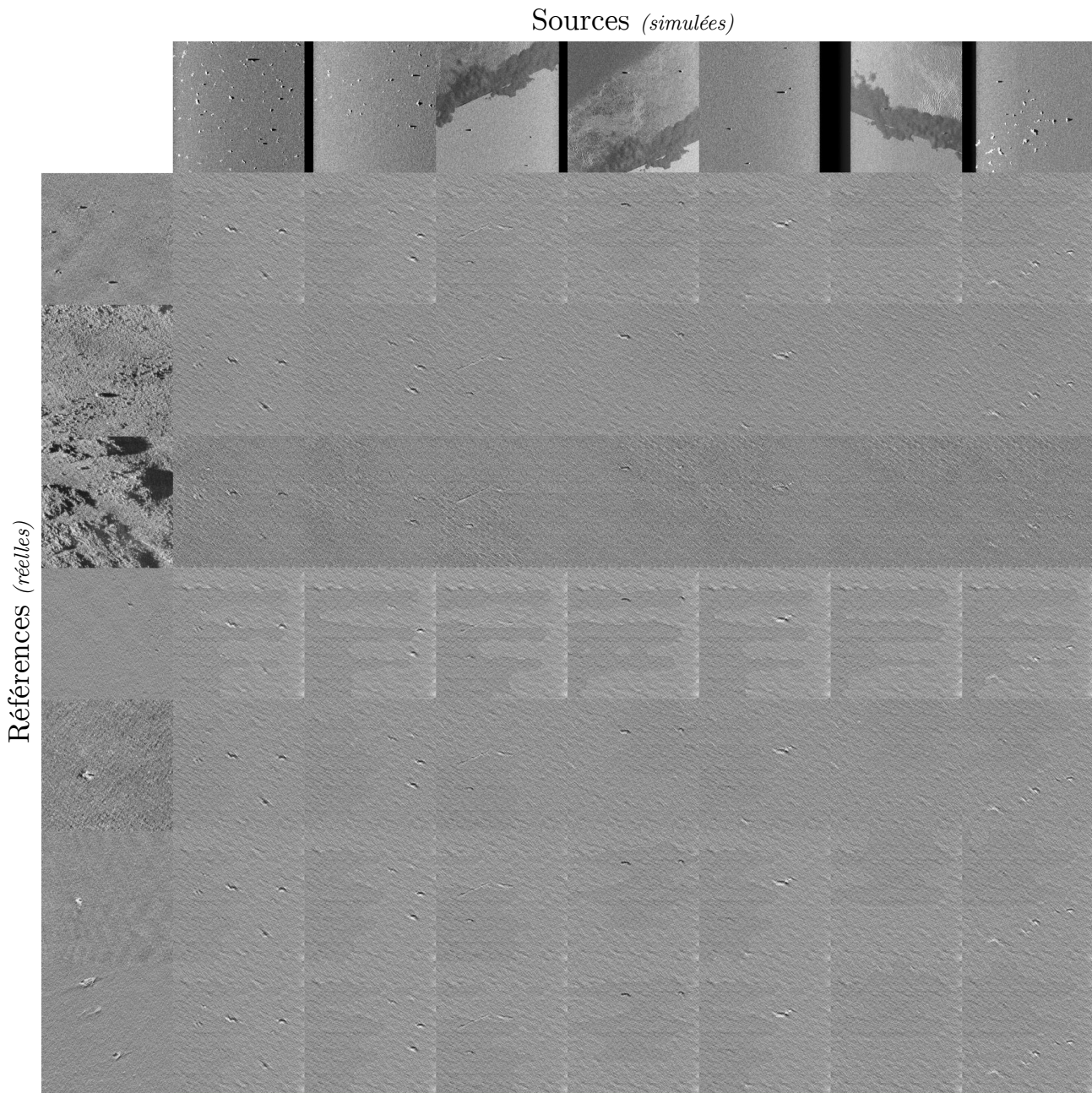


FIGURE 7.9 – Résultats du StarGANv2 dans le cas *sim2real*.

L'analyse qualitative de cette matrice de résultats met en évidence plusieurs cas représentatifs des difficultés observées.

Dans la figure 7.10, aucun objet n'est visible, ou seulement de manière très dégradée, dans l'image générée, alors qu'il est clairement identifiable dans l'image simulée d'origine. Ce type de disparition est particulièrement problématique lorsque la génération doit préserver la vérité terrain en vue d'une tâche de détection. Il peut refléter une instabilité locale du générateur ou une influence excessive du style sur la structure de l'image.

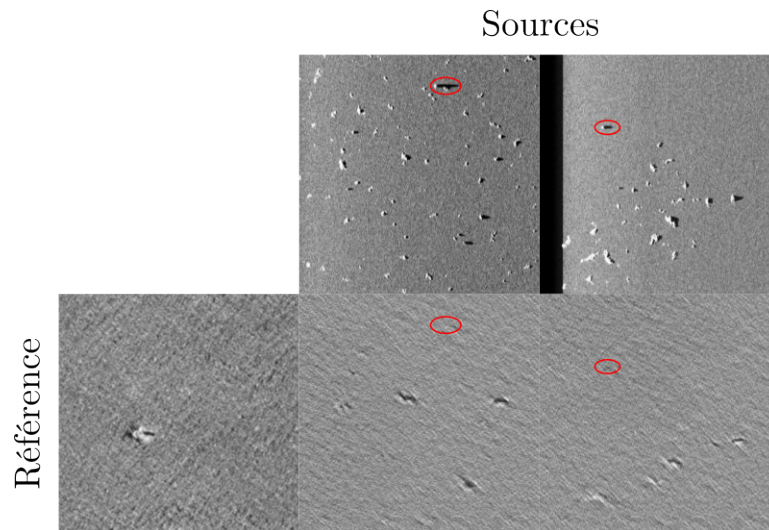


FIGURE 7.10 – Disparition de l'objet dans l'image générée.

La figure 7.11 illustre l'échec du transfert d'un style fortement texturé, issu d'un fond rocheux complexe. Malgré un style de référence visuellement distinct, l'image générée reste lisse et dépourvue de structures comparables. Ce cas met en évidence la difficulté du réseau à apprendre et restituer des styles riches, en particulier lorsqu'ils s'écartent du style dominant dans les données d'entraînement.

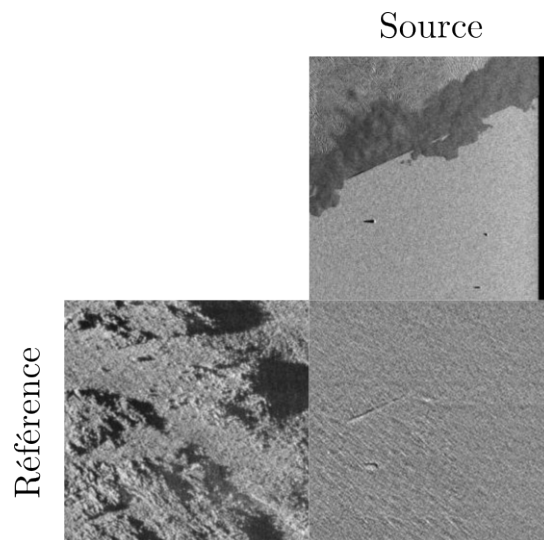


FIGURE 7.11 – Échec du transfert d'un style visuellement complexe.

La figure 7.12 présente un résultat visuellement dégradé : l'image générée adopte un style incorrect et une texture incohérente, la rendant inutilisable. Le style de l'image de référence n'est pas respecté. Ce type de sortie reflète une défaillance conjointe de l'encodeur de style et du générateur, illustrant les risques d'instabilité associés à un apprentissage sans supervision explicite de la traduction.

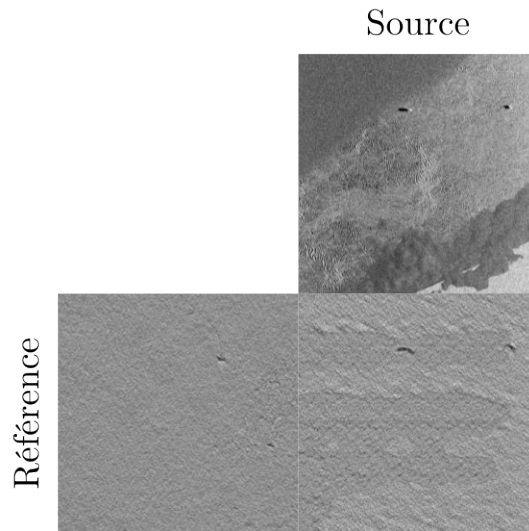


FIGURE 7.12 – *Style incohérent rendant l'image inutilisable.*

La figure 7.13 montre deux exemples où la position de certains objets est correctement préservée, mais le style généré reste incohérent avec celui de la référence. Plusieurs essais avec des styles de référence différents aboutissent à des résultats similaires, ce qui suggère une extraction ou une application insuffisante du style. Cela indique que l'encodeur de style peut ne pas capturer les caractéristiques pertinentes de la référence, ou que le générateur n'y répond pas de manière adéquate.

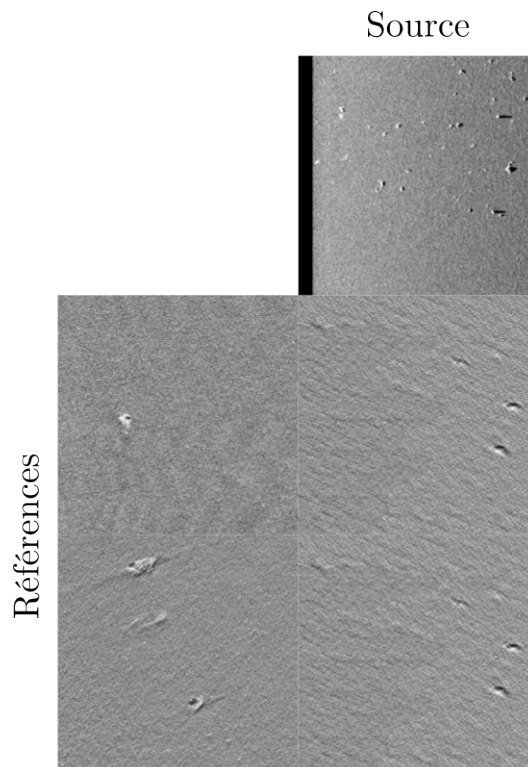


FIGURE 7.13 – *Bon positionnement partiel mais transfert de style incorrect.*

Les images traduites conservent globalement la structure de la scène d'origine ainsi que la position des objets. Toutefois, des défauts stylistiques subsistent : absence ou mauvaise orientation des ombres, restitution imparfaite des contrastes, apparition de flous selon les styles

appliqués. Ces altérations peuvent compromettre l’exploitation directe des images en détection, en particulier lorsque les ombres contribuent à la distinction des objets.

L’entraînement du StarGANv2 s’est avéré coûteux en temps de calcul, nécessitant environ une semaine sur plusieurs Graphics Processing Unit (GPU) pour atteindre un état stable générant des images visuellement exploitables. Cette durée reflète la complexité de l’architecture et la diversité des contraintes à satisfaire, en particulier lorsque le détecteur YOLO est intégré à la fonction de perte. Dans certains cas, l’interaction entre les pertes adversariales, perceptuelles (SSIM) et sémantiques (YOLO) a entraîné une instabilité, voire une divergence de l’apprentissage. Cette difficulté est accentuée par la rareté des exemples dans le domaine réel, qui limite l’apprentissage du style cible.

Une évaluation selon la métrique *FID* a été menée pour quantifier la proximité entre les images générées et celles du domaine cible, dans deux configurations : génération à partir de vecteurs latents aléatoires ou d’images de référence, et à deux étapes d’apprentissage (50k et 100k itérations). Les résultats sont présentés dans le tableau 7.1.

TABLE 7.1 – Résultats *FID* pour StarGANv2 modifié sur les scénarios *sim2real* et *real2sim*, à différentes itérations et selon la source du style.

Scénario	50k - Latent	50k - Réf.	100k - Latent	100k - Réf.
sim2real	223.6	230.6	68.7	109.6
real2sim	126.4	194.6	93.0	142.5
Moyenne	175.0	212.6	80.9	126.0

Les résultats montrent une amélioration marquée entre 50k et 100k itérations, en particulier sur le style latent dans le cadre du *sim2real*. Toutefois, les scores demeurent élevés, indiquant un écart de distribution toujours présent malgré une convergence partielle. Ce constat rejoint les observations visuelles : la génération reste partiellement biaisée, avec un réalisme insuffisant pour permettre une intégration fiable des objets dans un pipeline de détection.

Une seconde métrique, la *Learned Perceptual Image Patch Similarity (LPIPS)*, est utilisée en complément du FID. Contrairement à ce dernier, qui évalue la proximité globale entre distributions d’images, LPIPS mesure la similarité perceptuelle locale entre paires d’images, à partir de représentations internes issues de réseaux convolutifs. Cette mesure est généralement mieux corrélée à la qualité visuelle perçue par un observateur humain.

Les résultats LPIPS sont présentés dans le tableau 7.2. Comme pour le FID, les scores sont calculés pour les scénarios *sim2real* et *real2sim*, en fonction de deux sources de style (vecteur latent ou image de référence) et à deux étapes de l’apprentissage (50k et 100k itérations).

TABLE 7.2 – Résultats *LPIPS* pour StarGANv2. Plus la valeur est faible, plus la similarité perceptuelle est grande.

Scénario	50k - Latent	50k - Réf	100k - Latent	100k - Réf
sim2real	0.252	0.268	0.299	0.276
real2sim	0.660	0.414	0.643	0.486
Moyenne	0.456	0.341	0.471	0.381

Ces résultats mettent en évidence plusieurs tendances.

Les performances sont meilleures pour le scénario *sim2real* que pour *real2sim*, ce qui confirme les observations qualitatives selon lesquelles la transformation d’images simulées vers un style réaliste (même partiel) est plus aisée que l’inverse.

Par ailleurs, les scores LPIPS restent relativement élevés pour *real2sim*, indiquant une variabilité perceptuelle plus marquée entre les images générées et leur référence.

À 100k itérations, les styles latents présentent des résultats plus contrastés : s'ils peuvent produire des correspondances satisfaisantes en *sim2real*, leur efficacité reste limitée en *real2sim*. À l'inverse, l'usage d'une image de référence se montre légèrement plus robuste dans le cas *real2sim*, bien que les scores absolus, supérieurs à 0.4, traduisent une reconstruction encore imparfaite du style local.

Combinée au score FID, la métrique LPIPS permet de mieux caractériser les erreurs du générateur. Une baisse du FID peut refléter une cohérence globale satisfaisante des images, tandis qu'un LPIPS élevé indique un manque de fidélité stylistique ou de détails visuels fins. Cette dissociation met en évidence les limites du modèle à concilier structure d'ensemble et précision locale dans un contexte à faible supervision.

Les résultats obtenus soulignent plusieurs points. L'architecture StarGANv2, même enrichie, ne parvient pas à restituer l'ensemble de la richesse visuelle du domaine réel sonar, en particulier lorsque la vérité terrain doit être préservée. En outre, les difficultés à équilibrer les différentes formes de supervision (adversariale, perceptuelle, sémantique) montrent que chacune prise séparément reste insuffisante, ce qui souligne l'intérêt de stratégies hybrides ou semi-supervisées combinant génération, régularisation et post-traitement.

Conclusion & Perspectives

Ce chapitre synthétise les travaux réalisés au cours de cette thèse et en expose les principales conclusions. Il présente également les limites identifiées ainsi que plusieurs pistes de recherche à explorer dans la continuité, directe ou indirecte, de ces contributions.

Conclusion

Ce manuscrit traite de l'amélioration des performances des modèles de détection d'objets rares dans des images sonar, en tenant compte des contraintes propres au domaine : rareté des données ainsi que des annotations, variabilité des conditions d'acquisition et difficulté de transfert entre données simulées et réelles. Cette thèse se décompose autour de trois parties.

Dans la première partie nous avons introduit le contexte global de la thèse en présentant les principes de l'imagerie sonar, en détaillant la physique de la propagation acoustique sous-marine, ainsi qu'en décrivant les deux bases de données utilisées : une base simulée générée à partir d'un simulateur propriétaire, et une base réelle issue de campagnes d'acquisition menées par la DGA. Cette partie a également permis d'introduire les détecteurs d'objets grâce à une revue des approches majeures (RPN, YOLO, DETR).

Dans la deuxième partie nous avons présenté l'intégration de stratégies d'apprentissage actif dans le contexte de la détection d'objets. L'objectif est de réduire le nombre d'annotations tout en maximisant leur utilité, en sélectionnant les images les plus susceptibles d'améliorer les performances du modèle. Une méthode hybride, combinant des mesures d'incertitude à un regroupement dans l'espace latent, a été proposée. Les expérimentations menées ont montré des performances supérieures à celles obtenues par des sélections aléatoires ou fondées uniquement sur la confiance des prédictions.

Enfin, dans la dernière partie nous avons exploré l'utilisation de modèles génératifs, tels que CycleGAN et StarGAN, pour atténuer le décalage entre domaines simulé et réel, dans le but d'utiliser les images simulées. Les résultats préliminaires ont révélé des limites, en particulier concernant la préservation des objets et la stabilité des transformations, mais ouvrent des perspectives pour un enrichissement contrôlé des données du domaine réel.

Perspectives

Les travaux présentés dans cette thèse ont conduit à une méthodologie visant à améliorer la détection d'objets dans les images sonar par l'intégration de stratégies d'apprentissage actif. L'utilisation des GAN ouvre quant à elle de nouvelles perspectives sur l'exploitation des données simulées. Bien que les résultats obtenus soient encourageants, plusieurs limitations subsistent.

Les perspectives proposées s’articulent autour de trois axes principaux, visant à les dépasser et à explorer de nouvelles pistes d’amélioration.

Renforcement des stratégies d’apprentissage actif

L’approche développée dans cette thèse combine des mesures d’incertitude et un regroupement par clustering pour sélectionner les échantillons à annoter. Bien que cette méthode ait donné des résultats intéressants, plusieurs pistes peuvent être envisagées pour en améliorer les performances.

Pour renforcer la stratégie d’apprentissage actif, deux axes méritent un approfondissement. Le premier repose sur l’utilisation de réseaux bayésiens profonds, *Bayesian Deep Learning (BDL)*, afin d’estimer rigoureusement l’incertitude épistémique. À ce titre, Melo *et al.* [2024] introduisent une approche hybride, nommée *Bayesian Active Learner for Preference Modeling (BAL-PM)*, qui combine l’incertitude issue d’un modèle bayésien de préférences à une mesure d’entropie. Cette méthode améliore significativement les performances par rapport à des approches classiques telles que *Bayesian Active Learning by Disagreement (BALD)*, en réduisant de 33 à 68% le nombre d’exemples à annoter.

D’autre part, Thomas et Houssineau [2024] proposent une alternative aux réseaux bayésiens profonds, en s’appuyant sur une généralisation possibiliste des processus gaussiens bayésiens [Williams et Rasmussen, 2006]. Leur méthode intègre la théorie des probabilités à celle des possibilités [Zadeh, 1978 ; Dubois et Prade, 2015], afin de mieux modéliser l’incertitude épistémique dans un contexte d’apprentissage actif.

Le second axe concerne l’EDL, une méthode d’estimation d’incertitude détaillée dans la partie 4.3. Cette approche présente ainsi un avantage computationnel non négligeable par rapport aux méthodes bayésiennes classiques, ce qui la rend attractive dans le cadre d’un apprentissage actif en détection d’objets.

Néanmoins, des travaux récents ont mis en lumière certaines limites de l’EDL. Shen *et al.* [2024] montrent notamment que l’estimation d’incertitude fournie par les approches EDL peut être mal calibrée, en particulier dans des cas de forte incertitude épistémique ou en présence de données ambiguës. Ils soulignent que les mécanismes internes de l’EDL, tels que les fonctions d’activation et la régularisation implicite, peuvent conduire à une sous-estimation ou une sur-estimation systématique de l’incertitude, ce qui nuit à la fiabilité des décisions prises sur la base de cette information.

Afin de pallier ces limites, Yoon et Kim [2024] proposent une variante appelée “*Density-Aware EDL*”, qui introduit une estimation explicite de la densité des données dans l’espace latent. Cette information est ensuite utilisée pour pondérer l’incertitude produite par le modèle, permettant une meilleure différenciation entre les régions densément peuplées (où l’on peut avoir plus confiance dans les prédictions) et les régions isolées ou hors distribution. Cette approche améliore la capacité du modèle à détecter les données hors distribution et pourrait ainsi constituer une avancée pertinente dans un cadre d’apprentissage actif où l’objectif est précisément d’identifier des échantillons rares, atypiques ou incertains.

Par ailleurs, l’intégration d’informations contextuelles lors de la sélection, via des mécanismes attentionnels ou la modélisation des relations spatiales entre objets, pourrait affiner la pertinence des échantillons choisis. Il conviendrait également d’analyser l’influence de l’espace latent utilisé pour le regroupement, en explorant d’autres méthodes de projection ou de mesure de distance.

Génération d’images guidée par la structure pour l’adaptation de domaine

Les données simulées peuvent être projetées dans l’espace des données réelles, à condition de préserver la structure des objets d’intérêt lors de la génération d’images en conditions réelles. Plusieurs travaux récents montrent que cette transition est à la fois réalisable et bénéfique, notamment pour des tâches de suivi ou de détection.

Zhang *et al.* [2024a] proposent SDA-DETR, une méthode d’adaptation de domaine pour la détection d’objets en conditions de visibilité dégradée, structurée en trois étapes. L’approche combine une projection image-à-image non appariée, un alignement progressif des caractéristiques globales et locales, et un auto-encodeur masqué.

L’une des contributions principales réside dans l’introduction d’un domaine auxiliaire intermédiaire, dit “target-like”, généré à partir du domaine source. Ce domaine sert de passerelle visuelle vers le domaine cible, atténuant le décalage entre distributions et facilitant une adaptation plus progressive et plus stable.

L’intégration d’un auto-encodeur masqué (t-MAE) dans la dernière étape renforce l’apprentissage sémantique des caractéristiques du domaine cible, tout en limitant le biais du modèle vers le domaine source. En contrôlant l’application des masques pendant l’entraînement, le modèle est amené à se concentrer sur des régions informatives, même dans des contextes visuels dégradés. Testée dans des scénarios de conduite nocturne ou par mauvais temps, la méthode dépasse nettement les approches classiques d’adaptation, soulignant l’efficacité d’une stratégie visuelle hiérarchisée et guidée par la sémantique.

Pour la segmentation sémantique, Liming *et al.* [2024] introduisent DS-DWTGAN, un modèle de projection inter-domaine non supervisée basé sur un GAN structuré en double espace : spatial et fréquentiel. L’architecture intègre une contrainte explicite de préservation sémantique, visant à stabiliser le contenu des objets lors du transfert entre domaines visuellement distincts, notamment en imagerie satellitaire.

Un aspect central de cette approche repose sur l’utilisation de la transformée en ondelettes discrètes (*Discrete Wavelet Transformation (DWT)*), permettant de décomposer les images en composantes fréquentielles tout en conservant les structures fines. En opérant la projection de style dans ce domaine, le modèle préserve mieux les motifs sémantiques caractéristiques (formes, contours, textures), limitant ainsi les pertes d’information souvent observées dans les méthodes agissant uniquement dans l’espace image. Cette représentation multi-niveau améliore la robustesse aux variations de style tout en maintenant la cohérence des objets cibles.

Ces mécanismes permettent à DS-DWTGAN d’améliorer le mIoU de 2 à 3% sur plusieurs paires de bases satellites, montrant sa capacité à effectuer une translation inter-domaine fidèle sans compromettre la segmentation des objets d’intérêt. Ce type d’architecture pourrait constituer une base intéressante pour le transfert de données simulées vers des images sonar réelles, où la conservation des structures est essentielle.

Dans un cadre plus généraliste, Li *et al.* [2024] présentent UniHDA, une architecture multi-modale unifiée pour l’adaptation hybride de domaine. Leur approche s’appuie sur un générateur d’images pré-entraîné, tel que StyleGAN [Karras *et al.*, 2019], adapté à un domaine cible tout en préservant la structure sémantique.

L’un des apports principaux réside dans l’introduction d’une fonction de perte *Cross-domain Spatial Structure (CSS)*, conçue pour maintenir la cohérence des structures spatiales fines entre domaines. Cette régularisation rapproche les représentations de régions correspondantes et éloigne celles de régions non correspondantes, assurant ainsi la conservation de la topologie et

de la disposition des éléments d'origine malgré un changement de style.

Ce mécanisme permet à UniHDA de générer des images réalistes dans le domaine cible tout en préservant la structure des objets, ce qui constitue une piste pertinente lorsque la fidélité sémantique est nécessaire.

Ces travaux récents montrent qu'une projection inter-domaine efficace ne repose pas uniquement sur la qualité visuelle des images générées, mais sur l'intégration de mécanismes explicites assurant la préservation des structures sémantiques. Qu'il s'agisse d'un encadrement progressif via un domaine intermédiaire, d'une décomposition fréquentielle par ondelettes ou d'une régularisation des structures spatiales, ces approches démontrent la possibilité de transférer des images issues de domaines éloignés tout en maintenant la cohérence des objets d'intérêt.

Parmi les approches génératives récentes, les modèles de *diffusion* [Holderrieth et Erives, 2025] et de *flow matching* [Lipman *et al.*, 2022] apparaissent comme des alternatives crédibles aux GAN pour la traduction inter-domaines.

Les premiers reposent sur une estimation itérative du bruit inverse afin de reconstruire des images réalistes à partir de bruit pur, mais au prix d'un processus d'échantillonnage lent et coûteux.

Les seconds apprennent directement un champ de transport optimal entre deux distributions, sans recourir à une chaîne de débruitage, ce qui accélère la génération et stabilise l'apprentissage.

Diff2Flow [Schusterbauer *et al.*, 2025] combine ces deux paradigmes en transférant la richesse des trajectoires génératives issues d'un modèle de diffusion (comment le bruit devient image) vers un modèle de flow plus efficace. L'apprentissage repose sur l'alignement des chemins d'échantillonnage dans l'espace latent, associé à une régularisation locale pour préserver la consistance structurelle. Cette stratégie atteint une qualité visuelle comparable à celle des modèles de diffusion, tout en réduisant le coût de génération d'un facteur 5 à 10 dans des tâches de transfert de style.

Contrastive Flow Matching [Stoica *et al.*, 2025] enrichit la formulation classique du *flow matching* par l'introduction d'une perte contrastive conditionnelle. Celle-ci renforce la séparation entre flux associés à différentes conditions (classes, domaines, styles), tout en améliorant leur cohérence interne. Dans des scénarios de génération conditionnelle multi-domaines, notamment en imagerie biomédicale et satellitaire, cette approche améliore la préservation des structures propres à chaque domaine tout en assurant une différenciation claire entre contenus visuellement proches.

Ces approches récentes pourraient constituer une base méthodologique solide pour la génération inter-domaine en imagerie sonar. Leur capacité à modéliser finement les trajectoires de transformation, tout en préservant la cohérence sémantique et la diversité visuelle, en fait des candidates sérieuses pour une adaptation de domaine plus stable, plus rapide, et mieux contrôlée.

Dans le cadre de cette thèse, l'usage d'un simulateur sonar pour générer des données artificielles pourrait être renforcé par de tels mécanismes. Une projection guidée, reposant sur des contraintes structurelles, permettrait d'enrichir le jeu de données réel sans altérer la localisation ni la nature des objets.

Par ailleurs, un pré-alignement statistique des distributions d'intensité (valeurs de pixels) entre domaines, associé à la génération, peut réduire le décalage visuel et faciliter l'adaptation inter-domaine en amont du transfert de style.

Cette perspective ouvre la voie à un couplage entre génération contrôlée et apprentissage actif, dans l'objectif de réduire le coût d'annotation tout en maintenant une précision élevée sur des objets rares et peu représentés.

Couplage actif-génératif dans un pipeline unifié

Une perspective plus ambitieuse, non explorée dans cette thèse, consisterait à coupler de manière itérative les deux axes précédemment étudiés : la génération de données par GAN et l'apprentissage actif. Un tel pipeline traduirait d'abord les images simulées vers un domaine réaliste, avant de les filtrer en fonction de leur utilité estimée pour le détecteur. Seules les plus pertinentes seraient retenues et intégrées au processus d'apprentissage supervisé. Cette stratégie combinerait diversification artificielle et sélection guidée, dans l'objectif de maximiser l'efficacité des annotations humaines, voire d'envisager une automatisation partielle du cycle d'apprentissage.

Plusieurs travaux récents confirment la faisabilité de cette approche. Zhu *et al.* [2024b] proposent un pipeline nommé “*Batch-Selection Generative Active Learning (BSGAL)*”, conçu pour les contextes à distribution déséquilibrée, caractérisés par la présence de nombreuses classes rares (*long-tailed distribution*). Leur méthode s'appuie sur un générateur produisant en continu des images synthétiques, associé à une stratégie d'échantillonnage guidée par les gradients, permettant d'évaluer en ligne l'utilité de chaque image. Seuls les exemples les plus pertinents sont conservés pour l'entraînement du détecteur, conduisant à des gains notables sur les classes rares.

Dans une approche complémentaire, Fan *et al.* [2024] introduisent DiverGen, un cadre de génération visant à élargir la distribution des données d'entraînement. Le modèle génère des instances synthétiques plus diverses, améliorant la couverture du domaine cible et la robustesse du détecteur. Bien que non centré sur l'apprentissage actif, ce travail souligne l'intérêt de maîtriser la distribution des données générées pour renforcer les performances de manière ciblée.

Dans le contexte sonar, cette approche couplée pourrait être envisagée une fois la génération inter-domaine maîtrisée. Un pipeline itératif, dans lequel les images traduites sont sélectionnées en fonction de leur contribution estimée, permettrait de constituer un ensemble d'entraînement à la fois riche, diversifié et optimisé pour la détection. En fonction des contraintes du cadre applicatif, ce filtrage pourrait être entièrement automatisé ou supervisé par un expert, afin d'assurer la pertinence des données retenues. Un tel couplage représenterait une avancée notable dans la réduction du coût d'annotation et l'exploitation des simulateurs sonar.

Liste des publications

Revues internationales

- [Murrieta-Mendoza *et al.*, 2020a] Alejandro Murrieta-Mendoza, Ruxandra Botez, Hugo Ruiz, and Sonya Kessaci. Particle Swarm Optimization with Required Time of Arrival Constraint for Aircraft Trajectory. *SAE International Journal of Aerospace*, 13(2):269-291, 2020. DOI : 10.4271/01-13-02-0020.

Conférences internationales

- [Murrieta-Mendoza *et al.*, 2017a] Alejandro Murrieta-Mendoza, Hugo Ruiz, and Ruxandra M. Botez. Vertical Reference Flight Trajectory Optimization with the Particle Swarm Optimisation. The 36th IASTED International Conference on Modelling, Identification and Control (MIC 2017), Innsbruck, Austria, 2017. DOI : 10.2316/P.2017.848-032.
- [Murrieta-Mendoza *et al.*, 2017b] Alejandro Murrieta-Mendoza, Hugo Ruiz, Sonya Kessaci, and Ruxandra Mihaela Botez. 3D Reference Trajectory Optimization Using Particle Swarm Optimization. The 36th IASTED International Conference on Modelling, Identification and Control (MIC 2017), Innsbruck, Austria, 2017. DOI : 10.2514/6.2017-3435.
- [Murrieta-Mendoza *et al.*, 2017c] Alejandro Murrieta-Mendoza, Hugo Ruiz, Sonya Kessaci, and Ruxandra Mihaela Botez. Reference Trajectory Optimization Using the Particle Swarm Optimization. Council of European Aerospace Societies (CEAS) Conference, 2017.
- [Murrieta-Mendoza *et al.*, 2020b] Alejandro Murrieta-Mendoza, Hugo Ruiz, and Ruxandra Mihaela Botez. Horizontal flight trajectory optimization considering RTA constraints. *MATEC Web of Conferences*, 314:02002, 2020. DOI : 10.1051/mateconf/202031402002.
- [Ruiz *et al.*, 2021b] Hugo Ruiz, Marc Chaumont, Mehdi Yedroudj, Frédéric Comby, et Gérard Subsol. Analyse de l'évolutivité d'un réseau d'apprentissage profond pour la stéganalyse d'images. *CORESA 2021 - 21e colloque COMpression et REprésentation des Signaux Audiovisuels*, Sophia Antipolis, France, novembre 2021. HAL : lirmm-03819818.
- [Ruiz *et al.*, 2021c] Hugo Ruiz, Mehdi Yedroudj, Marc Chaumont, Frédéric Comby, and Gérard Subsol. LSSD : A Controlled Large JPEG Image Database for Deep-Learning-Based Steganalysis "Into the Wild". In Alberto Del Bimbo, Rita Cucchiara, Stan Sclaroff, Giovanni Maria Farinella, Tao Mei, Marco Bertini, Hugo Jair Escalante, and Roberto Vezzani (eds.), *Pattern Recognition. ICPR International Workshops and Challenges*, pages 470-483. Springer International Publishing, Cham, 2021. DOI : 10.1007/978-3-030-68780-9_38

- [Ruiz *et al.*, 2021a] Hugo Ruiz, Marc Chaumont, Mehdi Yedroudj, Ahmed Oulad Amara, Frédéric Comby, and Gérard Subsol. Analysis of the Scalability of a Deep-Learning Network for Steganography "Into the Wild". In Alberto Del Bimbo, Rita Cucchiara, Stan Sclaroff, Giovanni Maria Farinella, Tao Mei, Marco Bertini, Hugo Jair Escalante, and Roberto Vezzani (eds.), Pattern Recognition. ICPR International Workshops and Challenges, pages 439-452. Springer International Publishing, Cham, 2021. DOI : 10.1007/978-3-030-68780-9_36

Conférences nationales

- [Ruiz *et al.*, 2021b] Hugo Ruiz, Marc Chaumont, Mehdi Yedroudj, Frédéric Comby, et Gérard Subsol. Analyse de l'évolutivité d'un réseau d'apprentissage profond pour la stéganalyse d'images. CORESA 2021 - 21e colloque COMpression et REprésentation des Signaux Audiovisuels, Sophia Antipolis, France, novembre 2021. HAL : lirmm-03819818.

Bibliographie

- Sharat AGARWAL, Himanshu ARORA, Saket ANAND et Chetan ARORA : Contextual diversity for active learning. *In* Andrea VEDALDI, Horst BISCHOF, Thomas BROX et Jan-Michael FRAHM, éditeurs : *Computer Vision – ECCV 2020*, pages 137–153, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58517-4.
- H. AGGARWAL, M. MANI et M. JACOB : Modl : Model-based deep learning architecture for inverse problems. *IEEE Transactions on Medical Imaging*, 2017. URL <https://dx.doi.org/10.1109/TMI.2018.2865356>.
- Tuncay AKAL : Effects of environmental variability on acoustic propagation loss in shallow water. 2002. URL <https://api.semanticscholar.org/CorpusID:127148891>.
- Alexander AMINI, Wilko SCHWARTING, Ava SOLEIMANY et Daniela RUS : Deep evidential regression. *In* H. LAROCHELLE, M. RANZATO, R. HADSELL, M.F. BALCAN et H. LIN, éditeurs : *Advances in Neural Information Processing Systems*, volume 33, pages 14927–14937. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/aab085461de182608ee9f607f3f7d18f-Paper.pdf.
- Jinwon AN et Sungzoon CHO : Variational autoencoder based anomaly detection using reconstruction probability. 2015. URL <https://api.semanticscholar.org/CorpusID:36663713>.
- Martin ARJOVSKY et Léon BOTTOU : Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.
- David ARTHUR et Sergei VASSILVITSKII : k-means++ : the advantages of careful seeding. *In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07*, page 10271035, USA, 2007. Society for Industrial and Applied Mathematics. ISBN 9780898716245.
- Jordan T. ASH, Chicheng ZHANG, Akshay KRISHNAMURTHY, John LANGFORD et Alekh AGARWAL : Deep batch active learning by diverse, uncertain gradient lower bounds. *CoRR*, abs/1906.03671, 2019. URL <http://arxiv.org/abs/1906.03671>.
- Dzmitry BAHDANAU, Kyunghyun CHO et Yoshua BENGIO : Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- Mauro BARNI, Kassem KALLAS, Ehsan NOWROOZI et Benedetta TONDI : Cnn detection of gan-generated face images based on cross-band co-occurrences analysis. *arXiv preprint arXiv:2007.12909*, 2020.
- William BELSON : Matching and prediction on the principle of biological classification. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 8(2) :65–75, 1959.

- William H. BELUCH, Tim GENEWEIN, Andreas NURNBERGER et Jan M. KOHLER : The power of ensembles for active learning in image classification. pages 9368–9377, 2018.
- Yoshua BENGIO, Pascal LAMBLIN, Dan POPOVICI et Hugo LAROCHELLE : Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19, 2006.
- Yoshua BENGIO, Patrice SIMARD et Paolo FRASCONI : Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2) :157–166, 1994.
- Florentin BIEDER, Robin SANDKÜHLER et Philippe C CATTIN : Comparison of methods generalizing max-and average-pooling. *arXiv preprint arXiv:2103.01746*, 2021.
- Krzysztof BIKONIS et Jerzy DEMKOWICZ : Mems technology quality requirements as applied to multibeam echosounder. *Polish Maritime Research*, 25 :59–64, 12 2018.
- Alexey BOCHKOVSKIY, Chien-Yao WANG et Hong-Yuan Mark LIAO : Yolov4 : Optimal speed and accuracy of object detection, 2020.
- Mélodie BOILLET, Christopher KERMORVANT et Thierry PAQUET : Confidence estimation for object detection in document images. *Pattern Recognit. Lett.*, 166 :31–37, 2022. URL <https://api.semanticscholar.org/CorpusID:249167643>.
- Ali BORJI : Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179 :41–65, 2018. URL <https://dx.doi.org/10.1016/j.cviu.2018.10.009>.
- Konstantinos BOUSMALIS, Nathan SILBERMAN, David DOHAN, Dumitru ERHAN et Dilip KRISHNAN : Unsupervised pixel-level domain adaptation with generative adversarial networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3722–3731, 2017.
- Kai BRACH, Beate SICK et Oliver DÜRR : Single shot mc dropout approximation. *ArXiv*, abs/2007.03293, 2020. URL <https://api.semanticscholar.org/CorpusID:220381176>.
- Leo BREIMAN : Bagging predictors. *Machine learning*, 24 :123–140, 1996.
- Leo BREIMAN : Random forests. *Machine learning*, 45 :5–32, 2001.
- Leo BREIMAN, Jerome FRIEDMAN, Richard A OLSHEN et Charles J STONE : *Classification and regression trees*. Chapman and Hall/CRC, 2017.
- Xianhai BU, Fanlin YANG, Mingzhen XIN, Kai ZHANG et Yue MA : Improved calibration method for refraction errors in multibeam bathymetries with a wider range of water depths. *Applied Ocean Research*, 114 :102778, 2021. URL <https://api.semanticscholar.org/CorpusID:237660809>.
- Antoni Burguera BURGUERA et Gabriel OLIVER : High-resolution underwater mapping using side-scan sonar. *PLoS ONE*, 11, 2016. URL <https://api.semanticscholar.org/CorpusID:4486966>.
- Colin CAMPBELL, Nello CRISTIANINI, Alex SMOLA *et al.* : Query learning with large margin classifiers. *In ICML*, volume 20, 2000.
- Laurent CANDILLIER : *Contextualisation, Visualisation et Evaluation en Apprentissage Non Supervisé*. Theses, Université Charles de Gaulle - Lille III, septembre 2006. URL <https://theses.hal.science/tel-00617420>.

- Nicolas CARION, Francisco MASSA, Gabriel SYNNAEVE, Nicolas USUNIER, Alexander KIRILLOV et Sergey ZAGORUYKO : End-to-end object detection with transformers, 2020. URL <https://arxiv.org/abs/2005.12872>.
- Stanley H. CHAN, Todd E. ZICKLER et Yue M. LU : Monte carlo non-local means : Random sampling for large-scale image filtering. *IEEE Transactions on Image Processing*, 23 :3711–3725, 2013. URL <https://api.semanticscholar.org/CorpusID:3117518>.
- Jocelyn CHANUSSOT, Frédéric MAUSSANG et Alain HÉTET : Scalar image processing filters for speckle reduction on synthetic aperture sonar images. *OCEANS '02 MTS/IEEE*, 4 :2294–2301 vol.4, 2002. URL <https://api.semanticscholar.org/CorpusID:52087959>.
- Sijin CHEN, Yingyun YANG et Yan HUA : Semi-supervised active learning for object detection. *Electronics*, 12(2), 2023. ISSN 2079-9292. URL <https://www.mdpi.com/2079-9292/12/2/375>.
- Kashyap CHITTA, José Manuel ÁLVAREZ, Elmar HAUSSMANN et Clément FARABET : Less is more : An exploration of data redundancy with active dataset subsampling. *ArXiv*, abs/1905.12737, 2019. URL <https://api.semanticscholar.org/CorpusID:170078806>.
- Jiwoong CHOI, Ismail ELEZI, Hyuk-Jae LEE, Clément FARABET et José Manuel ÁLVAREZ : Active learning for deep object detection via probabilistic modeling. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10244–10253, 2021. URL <https://api.semanticscholar.org/CorpusID:232417789>.
- Yunjey CHOI, Minje CHOI, Munyoung KIM, Jung-Woo HA, Sunghun KIM et Jaegul CHOO : Stargan : Unified generative adversarial networks for multi-domain image-to-image translation, 2018.
- Yunjey CHOI, Youngjung UH, Jaejun YOO et Jung-Woo HA : Stargan v2 : Diverse image synthesis for multiple domains, 2020.
- Min Jin CHONG et D. FORSYTH : Effectively unbiased fid and inception score and where to find them. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3783–3792, 2020. URL <https://dx.doi.org/10.1109/cvpr42600.2020.00611>.
- Junyoung CHUNG, Caglar GULCEHRE, KyungHyun CHO et Yoshua BENGIO : Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.
- David A. COHN, Les E. ATLAS et Richard E. LADNER : Improving generalization with active learning. *Machine Learning*, 15 :201–221, 1994. URL <https://api.semanticscholar.org/CorpusID:8483688>.
- Jonathan D. COKER et Ahmed H. TEWFIK : Random sampling strategies in multistatic sar. *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2461–2464, 2008. URL <https://api.semanticscholar.org/CorpusID:18927774>.
- D. COOK *et al.* : Validation of fish length estimations from a high frequency multi-beam sonar (aris) and its utilisation as a field-based measurement technique. *Fisheries Research*, 2019.
- Corinna CORTES et Vladimir VAPNIK : Support-vector networks. *Machine Learning*, 20 (3) :273–297, 1995.

- Danilo COSTARELLI et Renato SPIGLER : Approximation results for neural network operators activated by sigmoidal functions. *Neural networks : the official journal of the International Neural Network Society*, 44C :101–106, 03 2013.
- Antonia CRESWELL, Tom WHITE, Vincent DUMOULIN, Kai ARULKUMARAN, Biswa SENGUPTA et Anil A. BHARATH : Generative adversarial networks : An overview. *IEEE Signal Processing Magazine*, 35(1) :53–65, 2018. URL <https://dx.doi.org/10.1109/MSP.2017.2765202>.
- Gabriella CSURKA, Christopher DANCE, Lixin FAN, Jutta WILLAMOWSKI et Cédric BRAY : Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004.
- Ido DAGAN et Sean P. ENGELSON : Committee-based sampling for training probabilistic classifiers. In Armand PRIEDITIS et Stuart RUSSELL, éditeurs : *Machine Learning Proceedings 1995*, pages 150–157. Morgan Kaufmann, San Francisco (CA), 1995. ISBN 978-1-55860-377-6. URL <https://www.sciencedirect.com/science/article/pii/B978155860377650027X>.
- Navneet DALAL et Bill TRIGGS : Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.
- Nibaran DAS, Ayatullah Faruk MOLLAH, Sudip SAHA et Syed Sahidul HAQUE : Handwritten arabic numeral recognition using a multi layer perceptron, 2010.
- Sanjoy DASGUPTA et Daniel HSU : Hierarchical sampling for active learning. In *Proceedings of the 25th International Conference on Machine Learning, ICML '08*, page 208215, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. URL <https://doi.org/10.1145/1390156.1390183>.
- Kumari DEEPSHIKHA, Sai Harsha YELLEN, P. K. SRIJITH et Chalavadi Krishna MOHAN : Monte carlo dropout for modelling uncertainty in object detection. *ArXiv*, abs/2108.03614, 2021. URL <https://api.semanticscholar.org/CorpusID:236957383>.
- Arthur P DEMPSTER : A generalization of bayesian inference. *Journal of the Royal Statistical Society : Series B (Methodological)*, 30(2) :205–232, 1968.
- Jia DENG, Wei DONG, Richard SOCHER, Li-Jia LI, Kai LI et Li FEI-FEI : Imagenet : A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- Jacob DEVLIN, Ming-Wei CHANG, Kenton LEE et Kristina TOUTANOVA : Bert : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 4171–4186, 2019.
- Nadim Mahmud DIPU, Sifatul Alam SHOHAN et K. M. A. SALAM : Deep learning based brain tumor detection and classification. In *2021 International Conference on Intelligent Technologies (CONIT)*, pages 1–6, 2021.
- Chris DONAHUE, Julian MCAULEY et Miller PUCKETTE : Wavegan : A generative model for raw audio. In *ICLR*, 2019.
- Xuanyi DONG, Liang ZHENG, Fan MA, Yi YANG et Deyu MENG : Few-example object detection with model communication. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41 :1641–1654, 2017. URL <https://api.semanticscholar.org/CorpusID:3320338>.

- Alexey DOSOVITSKIY, Lucas BEYER, Alexander KOLESNIKOV, Dirk WEISSENBORN, Xiaohua ZHAI, Thomas UNTERTHINER, Mostafa DEHGHANI, Matthias MINDERER, Georg HEIGOLD, Sylvain GELLY, Jakob USZKOREIT et Neil HOULSBY : An image is worth 16x16 words : Transformers for image recognition at scale. *arXiv preprint*, 2020.
- S. DOUGLAS et Jiutian YU : Why relu units sometimes die : Analysis of single-unit error backpropagation in neural networks. In *52nd Asilomar Conference on Signals, Systems, and Computers*, pages 864–868, 2018.
- Didier DUBOIS et Henri PRADE : Possibility theory and its applications : Where do we stand ? In *Springer handbook of computational intelligence*, pages 31–60. Springer, 2015.
- Vincent DUMOULIN et Francesco VISIN : A guide to convolution arithmetic for deep learning, 2018.
- Brian DUSHAW, Peter F. WORCESTER, Bruce D. CORNUELLE et Bruce M. HOWE : On equations for the speed of sound in seawater. *Journal of the Acoustical Society of America*, 93 :255–275, 1992. URL <https://api.semanticscholar.org/CorpusID:208868483>.
- Martin ESTER, Hans-Peter KRIEGEL, Jörg SANDER, Xiaowei XU *et al.* : A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.
- Chengxiang FAN, Muzhi ZHU, Hao CHEN, Yang LIU, Weijia WU, Huaqi ZHANG et Chunhua SHEN : Divergen : Improving instance segmentation by learning wider data distribution with more diverse generative data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3986–3995, 2024.
- David FAUST : Shrink your storage needs with c-treeace data compression. URL <https://www.faircom.com/insights/shrink-your-storage-needs-with-c-treeace-data-compression>. Accessed : August 31, 2023.
- Chen FENG, Ming-Yu LIU, Chieh-Chi KAO et Teng-Yok LEE : Deep active learning for civil infrastructure defect detection and classification. pages 298–306, 06 2017.
- Ioana FIROIU, Corina NAFORNITA, Dorina ISAR et Alexandru ISAR : Bayesian hyperanalytic denoising of sonar images. *IEEE Geoscience and Remote Sensing Letters*, 8 :1065–1069, 2011. URL <https://api.semanticscholar.org/CorpusID:332664>.
- Kenneth G. FOOTE : Ensuring fidelity of underwater acoustic measurements at low frequencies. In *OCEANS 2017 - Anchorage*, pages 1–5, 2017.
- Catherine FORBES, Merran EVANS, Nicholas HASTINGS et Brian PEACOCK : *Statistical distributions*. John Wiley & Sons, 2011.
- Yoav FREUND, Robert E SCHAPIRE *et al.* : Experiments with a new boosting algorithm. In *icml*, volume 96, pages 148–156. Citeseer, 1996.
- M. Maurice FRÉCHET : Sur quelques points du calcul fonctionnel. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 22(1) :1–72, décembre 1906. ISSN 0009-725X. URL <https://doi.org/10.1007/BF03018603>.
- Huan FU, Mingming GONG, Chaohui WANG, K. BATMANGHELICH, Kun ZHANG et D. TAO : Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping. In *CVPR*, 2019. URL <https://dx.doi.org/10.1109/CVPR.2019.00253>.

- Yarin GAL et Zoubin GHAHRAMANI : Dropout as a bayesian approximation : Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, 2015. URL <https://api.semanticscholar.org/CorpusID:160705>.
- Yarin GAL, Riashat ISLAM et Zoubin GHAHRAMANI : Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 11831192. JMLR.org, 2017.
- Francisco Javier GALLEGO : Stratified sampling of satellite images with a systematic grid of points. *Isprs Journal of Photogrammetry and Remote Sensing*, 59 :369–376, 2005. URL <https://api.semanticscholar.org/CorpusID:121489521>.
- Yaroslav GANIN, Evgeniya USTINOVA, Hana AJAKAN, Pascal GERMAIN, Hugo LAROCHELLE, François LAVIOLETTE, Mario MARCHAND et Victor LEMPITSKY : Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59) :1–35, 2016.
- Junyu GAO, Weili WANG, Yexin ZHANG, Liang LIN, Peng WU et Hanwang ZHANG : A comprehensive survey on evidential deep learning and its applications. *arXiv preprint arXiv:2409.04720*, 2024. URL <https://arxiv.org/abs/2409.04720>.
- Golnaz GHIASI, Tsung-Yi LIN et Quoc V. LE : Dropblock : A regularization method for convolutional networks. In *Neural Information Processing Systems*, 2018. URL <https://api.semanticscholar.org/CorpusID:53111490>.
- Ahmed GHITA, Bjørk ANTONIUSSEN, Walter ZIMMER, Ross GREER, Christian CRESS, Andreas MØGELMOSE, Mohan M. TRIVEDI et Alois C. KNOLL : Activeanno3d - an active learning framework for multi-modal 3d object detection. *2024 IEEE Intelligent Vehicles Symposium (IV)*, pages 1699–1706, 2024. URL <https://api.semanticscholar.org/CorpusID:267413028>.
- Ross GIRSHICK : Fast r-cnn, 2015.
- Ross GIRSHICK, Jeff DONAHUE, Trevor DARRELL et Jitendra MALIK : Rich feature hierarchies for accurate object detection and semantic segmentation, 2014.
- Teofilo F. GONZALEZ : Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.*, 38 :293–306, 1985. URL <https://api.semanticscholar.org/CorpusID:205092276>.
- Ian GOODFELLOW, Yoshua BENGIO et Aaron COURVILLE : *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- Ian GOODFELLOW, Jean POUGET-ABADIE, Mehdi MIRZA, Bing XU, David WARDE-FARLEY, Sherjil OZAIR, Aaron COURVILLE et Yoshua BENGIO : Generative adversarial networks, 2020.
- Eleonora GRASSUCCI, Luigi SIGILLO, Aurelio UNCINI et Danilo COMMINIELLO : Hypercomplex image- to- image translation. In *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, jul 2022. URL <https://doi.org/10.1109/IJCNN55064.2022.9892119>.
- Ross GREER, Bjørk ANTONIUSSEN, Mathias Viborg ANDERSEN, Andreas MØGELMOSE et Mohan M. TRIVEDI : The why, when, and how to use active learning in large-data-driven 3d object detection for safe autonomous driving : An empirical exploration. *ArXiv*, abs/2401.16634, 2024. URL <https://api.semanticscholar.org/CorpusID:267320855>.
- Grayna GRELOWSKA et Eugeniusz KOZACZKA : Underwater acoustic imaging of the sea. *Archives of Acoustics*, vol. 39(No 4) :439–452, 2014. URL http://journals.pan.pl/Content/101471/PDF/02_paper.pdf.

- Arthur GRETTON, Alex SMOLA, Jiayuan HUANG, Marcel SCHMITTFULL, Karsten BORGWARDT, Bernhard SCHÖLKOPF *et al.* : Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4) :5, 2009.
- Vincent A. Del GROSSO : New equation for the speed of sound in natural waters (with comparisons to other equations). *Journal of the Acoustical Society of America*, 56 :1084–1091, 1974. URL <https://api.semanticscholar.org/CorpusID:124990188>.
- Yuhong GUO : Active instance sampling via matrix partition. In J. LAFFERTY, C. WILLIAMS, J. SHAWE-TAYLOR, R. ZEMEL et A. CULOTTA, éditeurs : *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/f29c21d4897f78948b91f03172341b7b-Paper.pdf.
- Alfréd HAAR : Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69 :331–371, 1910.
- Song HAN, Jeff POOL, John TRAN et William J. DALLY : Learning both weights and connections for efficient neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS’15, page 11351143, Cambridge, MA, USA, 2015. MIT Press.
- James A HANLEY et Barbara J MCNEIL : The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1) :29–36, 1982.
- Roy Edgar HANSEN : Introduction to synthetic aperture sonar. In N. Z. KOLEV, éditeur : *Sonar Systems*, chapitre 1. IntechOpen, Rijeka, 2011. URL <https://doi.org/10.5772/23122>.
- Kaiming HE, Xiangyu ZHANG, Shaoqing REN et Jian SUN : Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision – ECCV 2014*, pages 346–361. Springer International Publishing, 2014a. URL https://doi.org/10.1007/978-3-319-10578-9_23.
- Kaiming HE, Xiangyu ZHANG, Shaoqing REN et Jian SUN : Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- Tianxu HE, Zhang KUI, Jie XIN, Pengpeng ZHAO, Jian WU, Xuefeng XIAN, Chunhua LI et Zhiming CUI : An active learning approach with uncertainty, representativeness, and diversity. *The Scientific World Journal*, 2014 :827586, 08 2014b.
- João F. HENRIQUES, Rui CASEIRO et Jorge P. BATISTA : Tracking in streamed video by updating globally optimal matchings. In *2010 IEEE International Conference on Image Processing (ICIP)*, pages 81–84, 2010.
- Fernando Sánchez HERNÁNDEZ, J. HERRÁEZ, M. KRAIEM, M. S. BARBA et M. GARCÍA : Predictive modeling of icu healthcare-associated infections from imbalanced data. using ensembles and a clustering-based undersampling approach. *arXiv.org*, 2020.
- Martin HEUSEL, Hubert RAMSAUER, Thomas UNTERTHINER, Bernhard NESSLER et Sepp HOCHREITER : Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL <https://papers.nips.cc/paper/2017/hash/6d1b10674c1681d22ddfdb2ab5982a5a-Abstract.html>.

- Hideitsu HINO et Shinto EGUCHI : Active learning by query by committee with robust divergences. *arXiv preprint arXiv:2211.10013*, 2022.
- G. E. HINTON et R. R. SALAKHUTDINOV : Reducing the dimensionality of data with neural networks. *Science*, 313(5786) :504–507, 2006. URL <https://www.science.org/doi/abs/10.1126/science.1127647>.
- Geoffrey E HINTON et Richard ZEMEL : Autoencoders, minimum description length and helmholtz free energy. In J. COWAN, G. TESAURO et J. ALSPECTOR, éditeurs : *Advances in Neural Information Processing Systems*, volume 6. Morgan-Kaufmann, 1993. URL https://proceedings.neurips.cc/paper_files/paper/1993/file/9e3cfc48eccf81a0d57663e129aef3cb-Paper.pdf.
- Dorit S. HOCHBAUM et David B. SHMOYS : A best possible heuristic for the k-center problem. *Math. Oper. Res.*, 10 :180–184, 1985. URL <https://api.semanticscholar.org/CorpusID:17379599>.
- Sepp HOCHREITER : The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6 :107–116, 04 1998.
- Judy HOFFMAN, Eric TZENG, Taesung PARK, Jun-Yan ZHU, Phillip ISOLA, Kate SAENKO, Alexei A. EFROS et Trevor DARRELL : Cycada : Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2018.
- Peter HOLDERRIETH et Ezra ERIVES : An introduction to flow matching and diffusion models. *arXiv preprint arXiv:2506.02070*, 2025.
- J. W. HORTON : Fundamentals of sonar. 1959. URL <https://api.semanticscholar.org/CorpusID:58527395>.
- Gao HUANG, Zhuang LIU, Laurens van der MAATEN et Kilian Q WEINBERGER : Densely connected convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- Sheng-jun HUANG, Rong JIN et Zhi-Hua ZHOU : Active learning by querying informative and representative examples. In J. LAFFERTY, C. WILLIAMS, J. SHAWE-TAYLOR, R. ZEMEL et A. CULOTTA, éditeurs : *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/5487315b1286f907165907aa8fc96619-Paper.pdf.
- Xun HUANG et Serge BELONGIE : Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.
- Xun HUANG, Ming-Yu LIU, Serge BELONGIE et Jan KAUTZ : Multimodal unsupervised image-to-image translation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 172–189, 2018.
- Junying HUI et Xueli SHENG : *Active Sonar Target Channel*, pages 193–202. Springer Nature Singapore, Singapore, 2022. ISBN 978-981-19-0774-6. URL https://doi.org/10.1007/978-981-19-0774-6_7.

- Frederik HVILSHØJ : Active learning in machine learning : Guide & strategies, septembre 2023. URL <https://encord.com/blog/active-learning-machine-learning-guide/>. Consulté le 12 mai 2025.
- Sekjin HWANG, Jinwoo CHOI et Joonsoo CHOI : Uncertainty-based selective clustering for active learning. *IEEE Access*, 10 :110983–110991, 2022. URL <https://api.semanticscholar.org/CorpusID:253124668>.
- Hydro INTERNATIONAL : Multibeam sonar reveals ship that warned titanic of icebergs. <https://www.hydro-international.com/content/news/multibeam-sonar-reveals-ship-that-warned-titanic-of-icebergs>, 2024. Consulté le 10 décembre 2024.
- Sergey IOFFE et Christian SZEGEDY : Batch normalization : Accelerating deep network training by reducing internal covariate shift. *In International conference on machine learning*, pages 448–456. pmlr, 2015.
- Phillip ISOLA, Jun-Yan ZHU, Tinghui ZHOU et Alexei A. EFROS : Image-to-image translation with conditional adversarial networks, 2018.
- Heinrich JIANG et Maya R. GUPTA : Bootstrapping for batch active sampling. *In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 30863096, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383325. URL <https://doi.org/10.1145/3447548.3467076>.
- Jianbo JIAO, Wei-Chih TU, Shengfeng HE et Rynson W. H. LAU : Formresnet : Formatted residual learning for image restoration. *In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017. URL <https://dx.doi.org/10.1109/CVPRW.2017.140>.
- Glenn JOCHER : YOLOv5 by Ultralytics, mai 2020. URL <https://github.com/ultralytics/yolov5>.
- Glenn JOCHER et Jing QIU : Ultralytics yolo11, 2024. URL <https://github.com/ultralytics/ultralytics>.
- Harry P. JOHNSON et M. G. HELFERTY : The geological interpretation of sidescan sonar. *Reviews of Geophysics*, 28 :357–380, 1990. URL <https://api.semanticscholar.org/CorpusID:54639682>.
- Ajay J JOSHI, Fatih PORIKLI et Nikolaos PAPANIKOLOPOULOS : Multi-class active learning for image classification. *In 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2372–2379. IEEE, 2009.
- Audun JSANG : *Subjective Logic : A formalism for reasoning under uncertainty*. Springer Publishing Company, Incorporated, 2018.
- Hyun-Ki JUNG et Gi-Sang CHOI : Improved yolov5 : Efficient object detection using drone images under various conditions. *Applied Sciences*, 2022. URL <https://api.semanticscholar.org/CorpusID:250935498>.
- Seohyeong JUNG, Sanghyun KIM et Juho LEE : A simple yet powerful deep active learning with snapshots ensembles. *In International Conference on Learning Representations*, 2023. URL <https://api.semanticscholar.org/CorpusID:259298619>.

- Konstantinos KAMNITSAS, Wenqi BAI, Enzo FERRANTE et et AL. : Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. *In International Conference on Information Processing in Medical Imaging*, pages 597–609. Springer, 2017.
- Tero KARRAS, Samuli LAINE et Timo AILA : A style-based generator architecture for generative adversarial networks. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.
- Hidetoshi KAWAGUCHI, Yuichi NAKATANI et Shogo OKADA : Idps signature classification based on active learning with partial supervision from network security experts. *IEEE Access*, 10 :105713–105725, 2022. URL <https://api.semanticscholar.org/CorpusID:252700552>.
- Shayan KAZEMINIA, Christoph BAUR, Arjan KUIJPER, Bram van GINNEKEN, Nassir NAVAB, Shadi ALBARQOUNI et Abhijit MUKHOPADHYAY : Gans for medical image analysis. *Artificial Intelligence in Medicine*, 109 :101938, 2020.
- Eden KIM, SeungChul SON et Seok-Gap KO : Continual active learning with ensemble method. *2024 15th International Conference on Information and Communication Technology Convergence (ICTC)*, pages 2235–2240, 2024. URL <https://api.semanticscholar.org/CorpusID:275547782>.
- Taeksoo KIM, Moonsu CHA, Hyunsoo KIM, Jung Kwon LEE et Jiwon KIM : Learning to discover cross-domain relations with generative adversarial networks. *In International conference on machine learning*, pages 1857–1865. Pmlr, 2017.
- Diederik P. KINGMA et Jimmy BA : Adam : A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL <https://api.semanticscholar.org/CorpusID:6628106>.
- D. KLEMENT, J. PREISSNER et V. STEIN : Special problems in applying the physical optics method for backscatter computations of complicated objects. *IEEE Transactions on Antennas and Propagation*, 36(2) :228–237, 1988.
- Ivan KRASIN, Tom DUERIG, Neil ALLDRIN, Vittorio FERRARI, Sami ABU-EL-HAJJA, Alina KUZNETSOVA, Hassan ROM, Jasper UIJLINGS, Stefan POPOV, Shahab KAMALI, Matteo MALLOCI, Jordi PONT-TUSET, Andreas VEIT, Serge BELONGIE, Victor GOMES, Abhinav GUPTA, Chen SUN, Gal CHECHIK, David CAI, Zheyun FENG, Dinesh NARAYANAN et Kevin MURPHY : Openimages : A public dataset for large-scale multi-label and multi-class image classification. 2017.
- Alex KRIZHEVSKY, Ilya SUTSKEVER et Geoffrey E HINTON : Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.
- Solomon KULLBACK et Richard A. LEIBLER : On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1) :79–86, 1951.
- Balaji LAKSHMINARAYANAN, Alexander PRITZEL et Charles BLUNDELL : Simple and scalable predictive uncertainty estimation using deep ensembles. *In Neural Information Processing Systems*, 2016. URL <https://api.semanticscholar.org/CorpusID:6294674>.
- Paul LANGEVIN : Procédés et appareils d’émission et de réception des ondes élastiques sous-marines à l’aide des propriétés piézo-électriques du quartz, brevet francais, n 505 703 du 17 septembre 1918. online, 1918. URL http://rcin.org.pl/Content/195816/PDF/WA35_226863_8818_Art33.pdf.

- Olivier LAURENT, Adrien LAFAGE, Enzo TARTAGLIONE, Geoffrey DANIEL, Jean-Marc MARTINEZ, Andrei BURSUC et Gianni FRANCHI : Packed-Ensembles for Efficient Uncertainty Estimation. In *11th International Conference on Learning Representations*, Kigali, Rwanda, mai 2023. URL <https://hal.science/hal-04042523>.
- Yann LE GALL : *Problèmes inverses en acoustique sous-marine : prédiction de performances et localisation de source en environnement incertain*. Thèse de doctorat, 10 2015.
- Y. LECUN, B. BOSER, J. S. DENKER, D. HENDERSON, R. E. HOWARD, W. HUBBARD et L. D. JACKEL : Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4) :541–551, 1989.
- Y. LECUN, L. BOTTOU, Y. BENGIO et P. HAFFNER : Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.
- Yann LECUN, John S. DENKER, Donald HENDERSON, Richard E. HOWARD, William HUBBARD et Lawrence D. JACKEL : Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*. Citeseer, 1990.
- Yann LECUN, D TOURESKY, Geoffrey HINTON et Terrance SEJNOWSKI : A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pages 21–28, 1988.
- Christian LEDIG, Lucas THEIS, Ferenc HUSZÁR, Jose CABALLERO, Andrew CUNNINGHAM, Alejandro ACOSTA, Andrew AITKEN, Alykhan TEJANI, Johannes TOTZ, Zehan WANG et Wenzhe SHI : Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017.
- David D. LEWIS et William A. GALE : A sequential algorithm for training text classifiers. In Bruce W. CROFT et C. J. van RIJSBERGEN, éditeurs : *SIGIR '94*, pages 3–12, London, 1994. Springer London. ISBN 978-1-4471-2099-5.
- Hengjia LI, Yingcong LI, Wenhan ZHANG et Feng LIU : Unihda : A unified and versatile framework for multi-modal hybrid domain adaptation. *arXiv preprint arXiv:2401.12596*, 2024. URL <https://arxiv.org/abs/2401.12596>.
- Mingkun LI et I.K. SETHI : Confidence-based active learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8) :1251–1261, 2006.
- Panle LI, Xiaohui HE, Xijie CHENG, Mengjia QIAO, Dingjun SONG, Mingyang CHEN, Tao ZHOU, Jiamian LI, Xiaoyu GUO, Shaokai HU et Zhihui TIAN : An improved categorical cross entropy for remote sensing image classification based on noisy labels. *Expert Systems with Applications*, 205 :117296, 05 2022.
- Ying LI, Binbin FAN, Weiping ZHANG, Weiping DING et Jianwei YIN : Deep active learning for object detection. *Inf. Sci.*, 579 :418–433, 2021. URL <https://api.semanticscholar.org/CorpusID:238667993>.
- Zhi LI, Shuihua WANG, Rui FAN, Gang CAO, Yudong ZHANG et Ting GUO : Teeth category classification via seven-layer deep convolutional neural network with max pooling and global average pooling. *International Journal of Imaging Systems and Technology*, 29(4) :529–539, 2019.

- Xiaolong LIAO, Bing TU, Jun LI et Antonio PLAZA : Class-wise graph embedding-based active learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 61 :1–13, 2023.
- Rainer LIENHART et Jochen MAYDT : An extended set of haar-like features for rapid object detection. In *International Conference on Image Processing (ICIP)*, pages 900–903, 2002.
- Wang LIMING, Liang ZHI, Xiong PENG, Zhang XIAOXIAO et Zhang QIANG : Ds-dwtgan : Dual-space wavelet-transform gan for unsupervised cross-domain semantic segmentation. *Scientific Reports*, 14 :74781, 2024. URL <https://www.nature.com/articles/s41598-024-74781-y>.
- Tsung-Yi LIN, Piotr DOLLÁR, Ross GIRSHICK, Kaiming HE, Bharath HARIHARAN et Serge BELONGIE : Feature pyramid networks for object detection, 2017.
- Tsung-Yi LIN, Michael MAIRE, Serge BELONGIE, James HAYS, Pietro PERONA, Deva RAMANAN, Piotr DOLLÁR et C Lawrence ZITNICK : Microsoft coco : Common objects in context. In *Computer Vision—ECCV 2014 : 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.
- Y. LIPMAN *et al.* : Flow matching for generative modeling. *ArXiv*, 2022.
- Jia LIU, Yan PANG, Lengheng YAN et Hanhao ZHU : An image quality improvement method in side-scan sonar based on deconvolution. *Remote. Sens.*, 15 :4908, 2023. URL <https://api.semanticscholar.org/CorpusID:264091707>.
- Ming-Yu LIU, Xun HUANG, Arun MALLYA, Tero KARRAS, Timo AILA, Jaakko LEHTINEN et Jan KAUTZ : Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10551–10560, 2019.
- Shu LIU, Lu QI, Haifang QIN, Jianping SHI et Jiaya JIA : Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018.
- Wei LIU, Dragomir ANGUELOV, Dumitru ERHAN, Christian SZEGEDY, Scott E. REED, Cheng- Yang FU et Alexander C. BERG : SSD : single shot multibox detector. *CoRR*, abs/1512.02325, 2015. URL <http://arxiv.org/abs/1512.02325>.
- Ze LIU, Yutong LIN, Yue CAO, Han HU, Yixuan WEI, Zheng ZHANG, Stephen LIN et B. GUO : Swin transformer : Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021.
- Mingsheng LONG, Yue CAO, Jianmin WANG et Michael JORDAN : Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015.
- David G. LOWE : Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2) :91–110, 2004.
- Xavier LURTON : *Acoustique sous-marine*. IFREMER, 07 1998. URL <https://archimer.ifremer.fr/doc/00017/12790/>.
- Wenyu LV, Yian ZHAO, Qinyao CHANG, Kui HUANG, Guanzhong WANG et Yi LIU : Rt-detr2 : Improved baseline with bag-of-freebies for real-time detection transformer, 2024. URL <https://arxiv.org/abs/2407.17140>.

- Xiaoming LV, Fa jie DUAN, Jia jia JIANG, Xiao FU et Lin GAN : Deep active learning for surface defect detection. *Sensors (Basel, Switzerland)*, 20, 2020. URL <https://api.semanticscholar.org/CorpusID:213186633>.
- James MACQUEEN : Some methods for classification and analysis of multivariate observations. *In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1 : Statistics*, volume 5, pages 281–298. University of California press, 1967.
- Alireza MAKHZANI, Jonathon SHLENS, Navdeep JAITLEY, Ian GOODFELLOW et Brendan FREY : Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2016.
- Qi MAO, Hsin-Ying LEE, Hung-Yu TSENG, Siwei MA et Ming-Hsuan YANG : Mode seeking generative adversarial networks for diverse image synthesis. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1429–1437, 2019.
- Francesco E. MARANZANA : On the location of supply points to minimize transportation costs. *IBM Systems Journal*, 2(2) :129–135, 1963.
- Jonathan MASCI, Ueli MEIER, Dan CIRESAN et Jürgen SCHMIDHUBER : Stacked convolutional auto-encoders for hierarchical feature extraction. pages 52–59, 06 2011. ISBN 978-3-642-21734-0.
- Warren S. MCCULLOCH et Walter PITTS : A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, 1943. ISSN 1522-9602. URL <https://doi.org/10.1007/BF02478259>.
- Herman MEDWIN et Clarence S. CLAY : *Fundamentals of Acoustical Oceanography*. Academic Press, 1997.
- Luckeciano C. MELO, Gustavo L. BATISTA, Flavio S. DEL FIOL, Rafael S. OLIVEIRA, Luiz H. ANDRADE et João P. PAPA : Deep bayesian active learning for preference modeling in large language models. *In Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL <https://arxiv.org/abs/2406.10023>.
- Lars MESCHEDER, Andreas GEIGER et Sebastian NOWOZIN : Which training methods for gans do actually converge? *In International conference on machine learning*, pages 3481–3490. PMLR, 2018.
- Zhongqi MIAO *et al.* : Insights and approaches using deep learning to classify wildlife. *Scientific Reports*, 9(8137), 2019.
- Umberto MICHELUCCI : An introduction to autoencoders. *arXiv preprint arXiv:2201.03898*, 2022.
- Marvin MINSKY et Seymour PAPERT : *Perceptrons*. Perceptrons. M.I.T. Press, Oxford, England, 1969.
- Mehdi MIRZA et Simon OSINDERO : Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.
- Anindya MONDAL et Mayukhmali DAS : Moving object detection for event-based vision using k-means clustering. *In 2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pages 1–6, 2021.

- Aimee MOSES, Christopher BOGART, Kelsey O’HAIRE, Lidia SOLORZANO et Elinor YEO : Diversity-based active learning : creating a representative object detection dataset in 3d point clouds. In *Defense + Commercial Sensing*, 2023. URL <https://api.semanticscholar.org/CorpusID:259492831>.
- Youssef MROUEH, Chun-Liang LI, Tom SERCU, Anant RAJ et Yu CHENG : Sobolev gan. *arXiv preprint arXiv:1711.04894*, 2017.
- A. MURRIETA-MENDOZA, R. BOTEZ, H. RUIZ et S. KESSACI : Particle swarm optimization with required time of arrival constraint for aircraft trajectory. *SAE Int. J. Aerosp.*, 13(2) :269–291, 2020a. URL <https://doi.org/10.4271/01-13-02-0020>.
- Alejandro MURRIETA-MENDOZA, Hugo RUIZ et Ruxandra M. BOTEZ : Vertical reference flight trajectory optimization with the particle swarm optimisation. In *Proceedings of the 36th IASTED International Conference on Modelling, Identification and Control (MIC 2017)*, Innsbruck, Austria, February 2017a. URL <https://zenodo.org/record/1315609>.
- Alejandro MURRIETA-MENDOZA, Hugo RUIZ et Ruxandra Mihaela BOTEZ : Horizontal flight trajectory optimization considering rta constraints. In *International Cross-Industry Safety Conference (ICSC) – International Symposium on Aircraft Technology, MRO and Operations (ISATECH)*, volume 314 de *MATEC Web of Conferences*, page 02002, 2020b. URL <https://doi.org/10.1051/matecconf/202031402002>.
- Alejandro MURRIETA-MENDOZA, Hugo RUIZ, Sonya KESSACI et Ruxandra Mihaela BOTEZ : 3d reference trajectory optimization using particle swarm optimization. In *AIAA Aviation Technology, Integration, and Operations Conference*, USA, 2017b. URL <https://arc.aiaa.org/doi/10.2514/6.2017-4208>.
- Alejandro MURRIETA-MENDOZA, Hugo RUIZ, Sonya KESSACI et Ruxandra Mihaela BOTEZ : Reference trajectory optimization using the particle swarm optimization. In *Proceedings of the Council of European Aerospace Societies (CEAS) Conference*, Europe, 2017c. URL <https://zenodo.org/record/1315609>.
- Monish R. NALLAPAREDDY, Kshitij SIROHI, Paulo L. J. DREWS, Wolfram BURGARD, Chih-Hong CHENG et Abhinav VALADA : Evcenternet : Uncertainty estimation for object detection using evidential learning. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5699–5706, 2023.
- NASA : Linear and nonlinear acoustic wave propagation in the atmosphere. Rapport technique, NASA, 1988. URL <https://ntrs.nasa.gov/api/citations/19880015884/downloads/19880015884.pdf>.
- NASA : Anatomy of an electromagnetic wave, 2010. URL http://science.nasa.gov/ems/02_anatomy. Accessed : 07/12/2023.
- Lakshmanan NATARAJ, Tajuddin Manhar MOHAMMED, Shivkumar CHANDRASEKARAN, Arjuna FLENNER, Jawadul H. BAPPY, Amit K. ROYCHOWDHURY et B. S. MANJUNATH : Detecting gan generated fake images using co-occurrence matrices. *Electronic Imaging 2019 (Media Watermarking, Security, and Forensics)*, 2019.
- Hieu T NGUYEN et Arnold SMEULDERS : Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79, 2004.
- Yuval NIRKIN, Yosi KELLER et Tal HASSNER : Fsgan : Subject agnostic face swapping and reenactment, 2019. URL <https://arxiv.org/abs/1908.05932>.

- Eric NOWAK, Frédéric JURIE et Bill TRIGGS : Sampling strategies for bag-of-features image classification. In *Computer Vision–ECCV 2006 : 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006, Proceedings, Part IV 9*, pages 490–503. Springer, 2006.
- Arturo ONCEVAY, Marco AVILA et Doo Young KWON : A survey of the recent architectures of deep convolutional neural networks. *Database : The Journal of Biological Databases and Curation*, 2021, 2021. URL <https://dx.doi.org/10.1093/database/baab012>.
- Jeffrey D. PADUAN et Libe WASHBURN : High-frequency radar observations of ocean surface currents. *Annual review of marine science*, 5 :115–36, 2013. URL <https://api.semanticscholar.org/CorpusID:207582017>.
- Tejas PANDEY, Nick PEARS, William A P SMITH et John Alexander McDERMID : E-DETR : Evidential deep learning for end-to-end uncertainty estimation in object detection, 2025. URL <https://openreview.net/forum?id=tdV1GRkCpZ>.
- Bumjun PARK, Songhyun YU et Jechang JEONG : Densely connected hierarchical network for image denoising. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019. URL <https://dx.doi.org/10.1109/CVPRW.2019.00263>.
- Hae-Sang PARK et Chi-Hyuck JUN : A simple and fast algorithm for k-medoids clustering. *Expert systems with applications*, 36(2) :3336–3341, 2009.
- Younghyun PARK, Wonjeong CHOI, Soyeong KIM, Dong-Jun HAN et Jaekyun MOON : Active learning for object detection with evidential deep learning and hierarchical uncertainty aggregation. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=MnEjsw-vj-X>.
- Razvan PASCANU, Tomas MIKOLOV et Yoshua BENGIO : On the difficulty of training recurrent neural networks. *International Conference on Machine Learning*, pages 1310–1318, 2013.
- Adam PASZKE, Sam GROSS, Francisco MASSA, Adam LERER, James BRADBURY, Gregory CHANAN, Trevor KILLEEN, Zeming LIN, Natalia GIMELSHEIN, Luca ANTIGA, Alban DESMAISON, Andreas KÖPF, Edward YANG, Zach DEVITO, Martin RAISON, Alykhan TEJANI, Sasank CHILAMKURTHY, Benoit STEINER, Lu FANG, Junjie BAI et Soumith CHINTALA : Pytorch : An imperative style, high-performance deep learning library, 2019.
- Manoranjan PAUL, Shah M. E. HAQUE et Subrata CHAKRABORTY : Human detection in surveillance videos and its applications - a review. *EURASIP Journal on Advances in Signal Processing*, 2013(1) :176, novembre 2013. ISSN 1687-6180. URL <https://doi.org/10.1186/1687-6180-2013-176>.
- Matthias PERKONIGG, Johannes HOFMANNINGER et Georg LANGS : Continual active learning for efficient adaptation of machine learning models to changing image acquisition, 2021. URL <https://arxiv.org/abs/2106.03351>.
- Robert PINSLER, Jonathan GORDON, Eric NALISNICK et José Miguel HERNÁNDEZ-LOBATO : Bayesian batch active learning as sparse subset approximation. In H. WALLACH, H. LAROCHELLE, A. BEYGELZIMER, F. d'ALCHÉ-BUC, E. FOX et R. GARNETT, éditeurs : *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/84c2d4860a0fc27bcf854c444fb8b400-Paper.pdf.

- Zhenshen QU, Jingda DU, Yong CAO, Qiuyu GUAN et Pengbo ZHAO : Deep active learning for remote sensing object detection. *ArXiv*, abs/2003.08793, 2020. URL <https://api.semanticscholar.org/CorpusID:213176093>.
- Joaquin QUINONERO-CANDELA, Masashi SUGIYAMA, Anton SCHWAIGHOFER et Neil D. LAWRENCE : *Dataset Shift in Machine Learning*. MIT Press, 2009.
- Alec RADFORD, Karthik NARASIMHAN, Tim SALIMANS et Ilya SUTSKEVER : Improving language understanding by generative pre-training. *arXiv preprint*, 2018.
- Mohammed Gamal RAGAB, Said Jadid ABDULKADIR, Amgad MUNEER et Hitham ALHUSSIAN : A comprehensive systematic review of yolo for medical object detection (2018 to 2023). *arXiv preprint arXiv:2401.12345*, 2024.
- Prajit RAMACHANDRAN, Barret ZOPH et Quoc V. LE : Swish : a self-gated activation function. *arXiv : Neural and Evolutionary Computing*, 2017. URL <https://api.semanticscholar.org/CorpusID:196158220>.
- Ramin RANJBARZADEH, Abbas BAGHERIAN KASGARI, Saeid JAFARZADEH GHOUSHCHI, Shokofeh ANARI, Maryam NASERI et Malika BENDECHACHE : Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images. *Scientific Reports*, 11(1) :10930, mai 2021. ISSN 2045-2322. URL <https://doi.org/10.1038/s41598-021-90428-8>.
- T. K. RAO, M. LINGAM, M. PRATEEK et E. RAJAN : Subsurface imaging using ground penetrating radar for underground oil and minerals exploration. *International Journal of Engineering Science Technologies*, 4(6) :139, 2020. URL <https://dx.doi.org/10.29121/ijoest.v4.i6.2020.139>.
- Joseph REDMON, Santosh Kumar DIVVALA, Ross B. GIRSHICK et Ali FARHADI : You only look once : Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015. URL <http://arxiv.org/abs/1506.02640>.
- Joseph REDMON et Ali FARHADI : Yolo9000 : better, faster, stronger. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.
- Joseph REDMON et Ali FARHADI : Yolov3 : An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.
- Shaoqing REN, Kaiming HE, Ross GIRSHICK et Jian SUN : Faster r-cnn : Towards real-time object detection with region proposal networks, 2016.
- Hamid REZATOFIGHI, Nathan TSOI, JunYoung GWAK, Amir SADEGHIAN, Ian REID et Silvio SAVARESE : Generalized intersection over union : A metric and a loss for bounding box regression. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.
- Boxfish ROBOTICS : Multibeam sonar accessories for boxfish rov. <https://www.boxfishrobotics.com/products/boxfish-rov/accessories/multibeam-sonar/>, 2024. Consulté le 10 décembre 2024.
- Olaf RONNEBERGER, Philipp FISCHER et Thomas BROX : U-net : Convolutional networks for biomedical image segmentation, 2015. URL <https://arxiv.org/abs/1505.04597>.

- Frank ROSENBLATT : The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386–408, 1958.
- Usha RUBY et Vamsidhar YENDAPALLI : Binary cross entropy with deep learning technique for image classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9, 10 2020.
- Hugo RUIZ, Marc CHAUMONT, Mehdi YEDROUDJ, Ahmed Oulad AMARA, Frédéric COMBY et Gérard SUBSOL : Analysis of the scalability of a deep-learning network for steganography “into the wild”. In Alberto DEL BIMBO, Rita CUCCHIARA, Stan SCLAROFF, Giovanni Maria FARINELLA, Tao MEI, Marco BERTINI, Hugo Jair ESCALANTE et Roberto VEZZANI, éditeurs : *Pattern Recognition. ICPR International Workshops and Challenges*, pages 439–452, Cham, 2021a. Springer International Publishing. ISBN 978-3-030-68780-9.
- Hugo RUIZ, Marc CHAUMONT, Mehdi YEDROUDJ, Frédéric COMBY et Gérard SUBSOL : Analyse de l'évolutivité d'un réseau d'apprentissage profond pour la stéganalyse d'images. In *CORESA 2021 – 21e colloque COmpression et REprésentation des Signaux Audiovisuels*, Sophia Antipolis, France, November 2021b. URL <https://hal-lirmm.ccsd.cnrs.fr/lirmm-03819818v1>.
- Hugo RUIZ, Mehdi YEDROUDJ, Marc CHAUMONT, Frédéric COMBY et Gérard SUBSOL : Lssd : A controlled large jpeg image database for deep-learning-based steganalysis “into the wild”. In Alberto DEL BIMBO, Rita CUCCHIARA, Stan SCLAROFF, Giovanni Maria FARINELLA, Tao MEI, Marco BERTINI, Hugo Jair ESCALANTE et Roberto VEZZANI, éditeurs : *Pattern Recognition. ICPR International Workshops and Challenges*, pages 470–483, Cham, 2021c. Springer International Publishing. ISBN 978-3-030-68780-9.
- David E RUMELHART, Geoffrey E HINTON et Ronald J WILLIAMS : Learning representations by back-propagating errors. *Nature*, 323 :533–536, 1986. URL <https://dx.doi.org/10.1038/323533a0>.
- Tim SALIMANS, Ian J. GOODFELLOW, Wojciech ZAREMBA, Vicki CHEUNG, Alec RADFORD et Xi CHEN : Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016. URL <https://papers.nips.cc/paper/2016/hash/8a3363abe792db2d8761d6403605aeb7-Abstract.html>.
- Bruno SARENI et Laurent KRAHENBUHL : Fitness sharing and niching methods revisited. *IEEE transactions on Evolutionary Computation*, 2(3) :97–106, 1998.
- Sebastian SCHMIDT, Qing RAO, Julian TATSCH et Alois KNOLL : Advanced active learning strategies for object detection. *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 871–876, 2020. URL <https://api.semanticscholar.org/CorpusID:231600435>.
- Greg SCHOHN et David COHN : Less is more : Active learning with support vector machines. In *Icml*, volume 2, page 6. Citeseer, 2000.
- Johannes SCHUSTERBAUER, Ming GUI, Frank FUNDEL et Björn OMMER : Diff2flow : Training flow matching models via diffusion model alignment. *ArXiv*, 2025.
- Ozan SENER et Silvio SAVARESE : Active learning for convolutional neural networks : A core-set approach. *arXiv : Machine Learning*, 2017. URL <https://api.semanticscholar.org/CorpusID:3383786>.

- Murat SENSOY, Lance KAPLAN et Melih KANDEMIR : Evidential deep learning to quantify classification uncertainty. In S. BENGIO, H. WALLACH, H. LAROCHELLE, K. GRAUMAN, N. CESA-BIANCHI et R. GARNETT, éditeurs : *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/a981f2b708044d6fb4a71a1463242520-Paper.pdf.
- Irhum SHAFKAT : Intuitively understanding convolutions for deep learning. URL <https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1>. Accessed : January 22, 2024.
- V. SHAGAPOV et V. SARAPULOVA : Refraction and reflection of sound at the boundary of a bubbly liquid. *Acoustical Physics*, 61 :37–44, 2015.
- Maohao SHEN, Yichi ZHANG, Henghui ZHAO, Yifan YANG, Gregory W. WORNELL et Peng XU : Are uncertainty quantification capabilities of evidential deep learning a mirage? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL <https://openreview.net/forum?id=P6nVDZRZRB>.
- Ashish SHRIVASTAVA, Tomas PFISTER, Oncel TUZEL, Josh SUSSKIND, Wenda WANG et Russ WEBB : Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2107–2116, 2017.
- Hanumant SINGH, Jonathan ADAMS, David A. MINDELL et Brendan FOLEY : Imaging underwater for archaeology. *Journal of Field Archaeology*, 27 :319 – 328, 2000. URL <https://api.semanticscholar.org/CorpusID:129534607>.
- Robert R SOKAL et Peter HA SNEATH : Principles of numerical taxonomy. 1963.
- Brandon L. SOUTHALL *et al.* : Behavioural responses of fin whales to military mid-frequency active sonar. *Royal Society Open Science*, 10, 2023.
- Nitish SRIVASTAVA, Geoffrey HINTON, Alex KRIZHEVSKY, Ilya SUTSKEVER et Ruslan SALAKHUTDINOV : Dropout : A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1) :1929–1958, jan 2014. ISSN 1532-4435.
- George STOICA, Vivek RAMANUJAN, Xiang FAN, Ali FARHADI, Ranjay KRISHNA et Judy HOFFMAN : Contrastive flow matching. *ArXiv*, 2025.
- J.L. SUTTON : Underwater acoustic imaging. *Proceedings of the IEEE*, 67(4) :554–566, 1979.
- Gabor J SZEKELY, Maria L RIZZO *et al.* : Hierarchical clustering via joint between-within distances : Extending ward’s minimum variance method. *Journal of classification*, 22(2) : 151–184, 2005.
- Xuning TANG, Yihua Shi ASTLE et Craig FREEMAN : Deep anomaly detection with ensemble-based active learning. *2020 IEEE International Conference on Big Data (Big Data)*, pages 1663–1670, 2020. URL <https://api.semanticscholar.org/CorpusID:232374190>.
- Yulin TANG, Liming WANG, Shaohua JIN, Jianhu ZHAO, Chao HUANG et Yongcan YU : Auv-based side-scan sonar real-time method for underwater-target detection. *Journal of Marine Science and Engineering*, 11(4), 2023. ISSN 2077-1312. URL <https://www.mdpi.com/2077-1312/11/4/690>.

- Juan Terven, Diana-Margarita Córdova-Esparza et Julio-Alejandro Romero-González : A comprehensive review of yolo architectures in computer vision : From yolov1 to yolov8 and yolo-nas. *Machine learning and knowledge extraction*, 5(4) :1680–1716, 2023.
- Jake Thomas et Jeremie Houssineau : Improving active learning with a bayesian representation of epistemic uncertainty. *arXiv preprint arXiv:2412.08225*, 2024. URL <https://arxiv.org/abs/2412.08225>.
- W. Thorp : Analytic description of the low-frequency attenuation coefficient. *The Journal of the Acoustical Society of America*, 42(1) :270–270, 1967.
- Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba et Pieter Abbeel : Domain randomization for transferring deep neural networks from simulation to the real world. *arXiv preprint arXiv:1703.06907*, 2017.
- Simon Tong et Daphne Koller : Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.*, 2 :45–66, 2000. URL <https://api.semanticscholar.org/CorpusID:7806109>.
- Jonathan Tremblay, Ankur Prakash, Daniel Acuna, Michael Brophy, Varun Jampani, Chetan Anil, Tien-Ju To, Sam Cameracci, Sidney Bochoon et Stan Birchfield : Training deep networks with synthetic data : Bridging the reality gap by domain randomization. *arXiv preprint arXiv:1804.06516*, 2018.
- Turing : Necessity of bias in neural networks. <https://www.turing.com/kb/necessity-of-bias-in-neural-networks#what-is-bias-in-a-neural-network>, 2024. Accessed : 2024-02-19.
- Eric Tzeng, Judy Hoffman, Kate Saenko et Trevor Darrell : Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7167–7176, 2017.
- J. Uijlings, K. V. D. Sande, T. Gevers et A. Smeulders : Selective search for object recognition. *International Journal of Computer Vision*, 104(2) :154–171, 2013.
- Ultralytics : YOLOv5 tutorials : Architecture description, 2023. URL https://docs.ultralytics.com/fr/yolov5/tutorials/architecture_description/.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, ukasz Kaiser et Illia Polosukhin : Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.
- Vincent Vercauteren, Wannes Meert, Gust Verbruggen, Koen Maes, Ruben Baumer et Jesse Davis : Semi-supervised anomaly detection with an application to water analytics. *Industrial Conference on Data Mining*, 2018.
- Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio et Pierre-Antoine Manzagol : Stacked denoising autoencoders : Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, 11 :33713408, dec 2010. ISSN 1532-4435.
- Paul Viola et Michael Jones : Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 511–518, 2001.

- Huy V VO, Oriane SIMÉONI, Spyros GIDARIS, Andrei BURSUC, Patrick PÉREZ et Jean PONCE : Active learning strategies for weakly-supervised object detection. *In European conference on computer vision*, pages 211–230. Springer, 2022.
- Chien-Yao WANG, Hong-Yuan Mark LIAO, Yueh-Hua WU, Ping-Yang CHEN, Jun-Wei HSIEH et I-Hau YEH : Cspnet : A new backbone that can enhance learning capability of cnn. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391, 2020.
- Jun WANG et J. JIAO : Track before detect for low frequency active towed array sonar. *In 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*. IEEE, 2019.
- Keze WANG, Xiaopeng YAN, Dongyu ZHANG, Lei ZHANG et Liang LIN : Towards human-machine cooperation : Self-supervised sample mining for object detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- Keze WANG, Dongyu ZHANG, Yanning LI, Ruimao ZHANG et Liang LIN : A new active labeling method for deep learning. *In Proceedings of the 2016 ACM on Multimedia Conference*, pages 157–161, 2016.
- Mei WANG et Weihong DENG : Deep visual domain adaptation : A survey. *ArXiv*, abs/1802.03601, 2018. URL <https://api.semanticscholar.org/CorpusID:3654323>.
- Meng WANG, Guobao LIU et Haipeng LIU : Object detection via active learning strategy based on saliency of local features and posterior probability. *IEEE Access*, 12 :35462–35474, 2024a. URL <https://api.semanticscholar.org/CorpusID:268150702>.
- Min WANG, Fan MIN, Zhi-Heng ZHANG et Yan-Xue WU : Active learning through density clustering. *Expert Systems with Applications*, 85 :305–317, 2017. ISSN 0957-4174. URL <https://www.sciencedirect.com/science/article/pii/S095741741730369X>.
- Ruofan WANG, Rui-Wei ZHAO, Xiaobo ZHANG et Rui FENG : Towards evidential and class separable open set object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(6) :5572–5580, Mar. 2024b. URL <https://ojs.aaai.org/index.php/AAAI/article/view/28367>.
- Xuezhi WANG, Tzu-Kuo HUANG et Jeff SCHNEIDER : Active transfer learning under model shift. *In International Conference on Machine Learning*, pages 1305–1313. PMLR, 2014.
- H. WEINBERG : Multipath expansion for computing acoustic propagation loss in horizontally stratified media. *Journal of the Acoustical Society of America*, 58, 1975.
- Paul WERBOS : Beyond regression : New tools for prediction and analysis in the behavioral sciences. *PhD thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA*, 1974.
- Christopher K. WILLIAMS et Carl Edward RASMUSSEN : *Gaussian Processes for Machine Learning*, volume 2. MIT Press Cambridge, MA, 2006.
- Jiaxi WU, Jiaxin CHEN et Di HUANG : Entropy-based active learning for object detection with progressive diversity constraint. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9397–9406, 2022.

- Ming XIE, Yuxi LI, Yabiao WANG, Zekun LUO, Zhenye GAN, Zhongyi SUN, Mingmin CHI, Chengjie WANG et Pei WANG : Learning distinctive margin toward active domain adaptation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7983–7992, 2022. URL <https://api.semanticscholar.org/CorpusID:247411022>.
- Xijiong XIE : Sampling active learning based on non-parallel support vector machines. *Neural Processing Letters*, 53 :2081–2094, 2021.
- Jie XU, Wei WANG, Hanyuan WANG et Jinhong GUO : Multi-model ensemble with rich spatial information for object detection. *Pattern Recognit.*, 99, 2020. URL <https://api.semanticscholar.org/CorpusID:208609904>.
- Y. XU, X. ZHU et A. K. QIN : Cluster-based active learning for compact image classification. In *2010 20th International Conference on Pattern Recognition*, pages 3224–3227. IEEE, 2010.
- Zhao XU, Kai YU, Volker TRESP, Xiaowei XU et Jizhi WANG : Representative sampling for text classification using support vector machines. In *Advances in Information Retrieval : 25th European Conference on IR Research, ECIR 2003, Pisa, Italy, April 14–16, 2003. Proceedings 25*, pages 393–407. Springer, 2003.
- Jun YAN, Aiyu QU, H. CAI, D. XIE et Shuaifeng WU : Application of geological radar method in exploration of reservoir complex pipelines. *Advances in Technology and Engineering*, 1 (1) :316, 2021. URL <https://dx.doi.org/10.3233/atde210316>.
- Xuyang YAN, Abdollah HOMAIFAR, Shabnam NAZMI et Mohammad RAZEGHI-JAHROMI : A novel clustering algorithm based on fitness proportionate sharing. In *2017 IEEE International conference on systems, man, and cybernetics (SMC)*, pages 1960–1965. IEEE, 2017.
- Xuyang YAN, Shabnam NAZMI, Biniam GEBRU, Mohd ANWAR, Abdollah HOMAIFAR, Mrinmoy SARKAR et Kishor Datta GUPTA : Mitigating shortage of labeled data using clustering-based active learning with diversity exploration. *ArXiv*, abs/2207.02964, 2022. URL <https://api.semanticscholar.org/CorpusID:250334591>.
- Chenhongyi YANG, Lichao HUANG et Elliot J. CROWLEY : Plug and play active learning for object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17784–17793, 2024.
- Dingdong YANG, Seunghoon HONG, Yunseok JANG, Tianchen ZHAO et Honglak LEE : Diversity-sensitive conditional generative adversarial networks. *arXiv preprint arXiv:1901.09024*, 2019.
- Yazhou YANG, Xiaoqing YIN, Yang ZHAO, Jun LEI, Weili LI et Zhe SHU : Batch mode active learning based on multi-set clustering. *IEEE Access*, 9 :51452–51463, 2021. URL <https://api.semanticscholar.org/CorpusID:233197319>.
- Yi YANG, Zhigang MA, Feiping NIE, Xiaojun CHANG et Alexander G HAUPTMANN : Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113 :113–127, 2015.
- Zichao YANG, Diyi YANG, Chris DYER, Xiaodong HE, Alex SMOLA et Eduard HOVY : Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 1480–1489, 2016.

- Mehdi YEDROUDJ : *Steganalysis and steganography by deep learning*. Thèse de doctorat, 2019. URL <http://www.theses.fr/2019MONT095>. Thèse de doctorat dirigée par Chaumont, Marc et Comby, Frédéric Informatique Montpellier 2019.
- Junfei YI, Jianxu MAO, Tengfei LIU, Mingjie LI, Hanyu GU, Hui ZHANG, Xiaojun CHANG et Yaonan WANG : Teaching with uncertainty : Unleashing the potential of knowledge distillation in object detection. *ArXiv*, abs/2406.06999, 2024. URL <https://api.semanticscholar.org/CorpusID:270379766>.
- Zili YI, Hao ZHANG, Ping TAN et Minglun GONG : Dualgan : Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017.
- Chunyang YIN, Shuangshuang CHEN et Zhichao YIN : Clustering-based active learning classification towards data stream. *ACM Trans. Intell. Syst. Technol.*, 14(2), février 2023. ISSN 2157-6904. URL <https://doi.org/10.1145/3579830>.
- Donggeun YOO et In-So KWEON : Learning loss for active learning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 93–102, 2019. URL <https://api.semanticscholar.org/CorpusID:148571749>.
- Jaeyoung YOO, Geonseok SEO et Nojun KWAK : Mixture-model-based bounding box density estimation for object detection. *ArXiv*, abs/1911.12721, 2019. URL <https://api.semanticscholar.org/CorpusID:208513322>.
- Taeseong YOON et Heeyoung KIM : Uncertainty estimation by density aware evidential deep learning. *arXiv preprint arXiv:2409.08754*, 2024. URL <https://arxiv.org/abs/2409.08754>.
- Weiping YU, Sijie ZHU, Taojiannan YANG et Chen CHEN : Consistency-based active learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3951–3960, June 2022.
- Tianning YUAN, Fang WAN, Mengying FU, Jianzhuang LIU, Songcen XU, Xiangyang JI et Qixiang YE : Multiple instance active learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5330–5339, June 2021.
- Mohd Ansor Bin YUSOF et Shahid KABIR : An overview of sonar and electromagnetic waves for underwater communication. *IETE Technical Review*, 29 :307 – 317, 2012. URL <https://api.semanticscholar.org/CorpusID:62475852>.
- Lotfi A ZADEH : Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1 (1) :3–28, 1978.
- Mengyao ZHAI, Lei CHEN, Frederick TUNG, Jiawei HE, Megha NAWHAL et Greg MORI : Lifelong gan : Continual learning for conditional image generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 285–294, 2019. URL <https://dx.doi.org/10.1109/ICCV.2019.00285>.
- Gege ZHANG, Luping WANG et Zengping CHEN : A step-wise domain adaptation detection transformer for object detection under poor visibility conditions. *Remote Sensing*, 16(15) :2722, 2024a. URL <https://www.mdpi.com/2072-4292/16/15/2722>.

- Yanyan ZHANG, Fei ZHAO, Sheng CHANG, Mingliang LIU et Robert WANG : An innovative synthetic aperture radar design method for lunar water ice exploration. *Remote Sensing*, 14 (9) :2148, 2022. URL <https://dx.doi.org/10.3390/rs14092148>.
- Zhipeng ZHANG, Wenting MA, Xiaohang YUAN, Yuan HAO, Meng GUO, Hongyi TANG, Zhiheng ZHOU et Zhenjie YAO : Instance-aware uncertainty for active learning in object detection. In *2024 IEEE International Conference on Image Processing (ICIP)*, pages 298–304, 2024b.
- Yian ZHAO, Wenyu LV, Shangliang XU, Jinman WEI, Guanzhong WANG, Qingqing DANG, Yi LIU et Jie CHEN : Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16965–16974, 2024.
- Zhaohui ZHENG, Ping WANG, Wei LIU, Jinze LI, Rongguang YE et Dongwei REN : Distance-iou loss : Faster and better learning for bounding box regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12993–13000, 2019.
- Dongjun ZHU, Chengjie GU, Junjun ZHANG, Yuyou YAO et Dayu TAN : Global-margin uncertainty and collaborative sampling for active learning in complex aerial images object detection. *IEEE Geoscience and Remote Sensing Letters*, 21 :1–5, 2024a.
- Jun-Yan ZHU, Taesung PARK, Phillip ISOLA et Alexei A EFROS : Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.
- Jun-Yan ZHU, Richard ZHANG, Deepak PATHAK, Trevor DARRELL, Alexei A EFROS, Oliver WANG et Eli SHECHTMAN : Toward multimodal image-to-image translation. *Advances in neural information processing systems*, 30, 2018.
- Muzhi ZHU, Chengxiang FAN, Hao CHEN, Yang LIU, Weian MAO, Xiaogang XU et Chunhua SHEN : Generative active learning for long-tailed instance segmentation. In *International Conference on Machine Learning (ICML)*, 2024b. URL <https://arxiv.org/abs/2406.02435>.
- Xizhou ZHU, Weijie SU, Lewei LU, Bin LI, Xiaogang WANG et Jifeng DAI : Deformable detr : Deformable transformers for end-to-end object detection. *arXiv preprint*, 2020.

Annexes

Notions fondamentales sur les réseaux de neurones

NN

Dans cette section, nous présentons les balbutiements des Réseaux de Neuronaux Artificiels (RNA), “*Artificial Neural Network (ANN)*”, et les principes fondamentaux ayant permis de créer de tels modèles. Goodfellow *et al.* [2016] est un livre qui effectue une revue détaillée de ce domaine et pour davantage d’informations, il est conseillé de s’y référer. Ce développement a aussi été inspiré de la thèse Yedroudj [2019] portant sur du traitement d’images.

Réseaux neuronaux artificiels (RNA)

Ces réseaux représentent une avancée dans l’exploitation de la puissance de calcul, pour traiter l’information, raisonner et apprendre d’une manière s’apparentant à la cognition humaine.

Les RNA s’éloignent grandement des modèles informatiques conventionnels, en imitant le réseau interconnecté de neurones et de synapses du cerveau, permettant de créer des systèmes dynamiques et adaptables.

Ces réseaux de neurones, organisés en couches, créent une hiérarchie permettant au réseau, de modéliser des relations complexes au sein des données (cf. la figure 14).

Un RNA est une succession de L couches, chaque couche $l \in \{0, \dots, L\}$ étant composée de M_l neurones (en rouge dans la figure 14), et les couches reliées entre elles le sont par des connexions pondérées. Ainsi, le neurone $m_j^{(l)}$ de la l -ième couche est connecté, au travers d’une connexion pondérée $w_{j,k}^{(l),(l+1)}$, au neurone $m_k^{(l+1)}$ de la $l+1$ -ième couche, où $j \in \{1, \dots, M_l\}$, $k \in \{1, \dots, M_{l+1}\}$ sont les index des neurones dans une couche et l l’index de la couche. L’ensemble des poids $w_{j,k}^{(l),(l+1)}$ correspond aux paramètres que le réseau doit mettre à jour lors de l’apprentissage. Dans la figure 14, des connexions sont notées en violet, elles représentent les liens entre le neurone m_1 de la couche $l=0$ avec l’ensemble des autres neurones de la couche $l=1$.

De manière générale, les RNA sont composés de trois types de couches.

1. La première est la couche d’entrée, située au début du réseau neuronal. Les neurones de la couche d’entrée (représentée en vert dans la figure 14) sont prêts à recevoir l’ensemble de données initial, représentant les données du problème. Dans le contexte du traitement d’images, ces neurones sont reliés aux pixels des images.
2. La deuxième est la couche cachée (représentée en rouge dans la figure 14) caractérisée par l’ensemble des couches, situées entre la couche d’entrée et de sortie. Ces couches sont chargées d’exécuter des calculs complexes et de favoriser la transmission transparente des

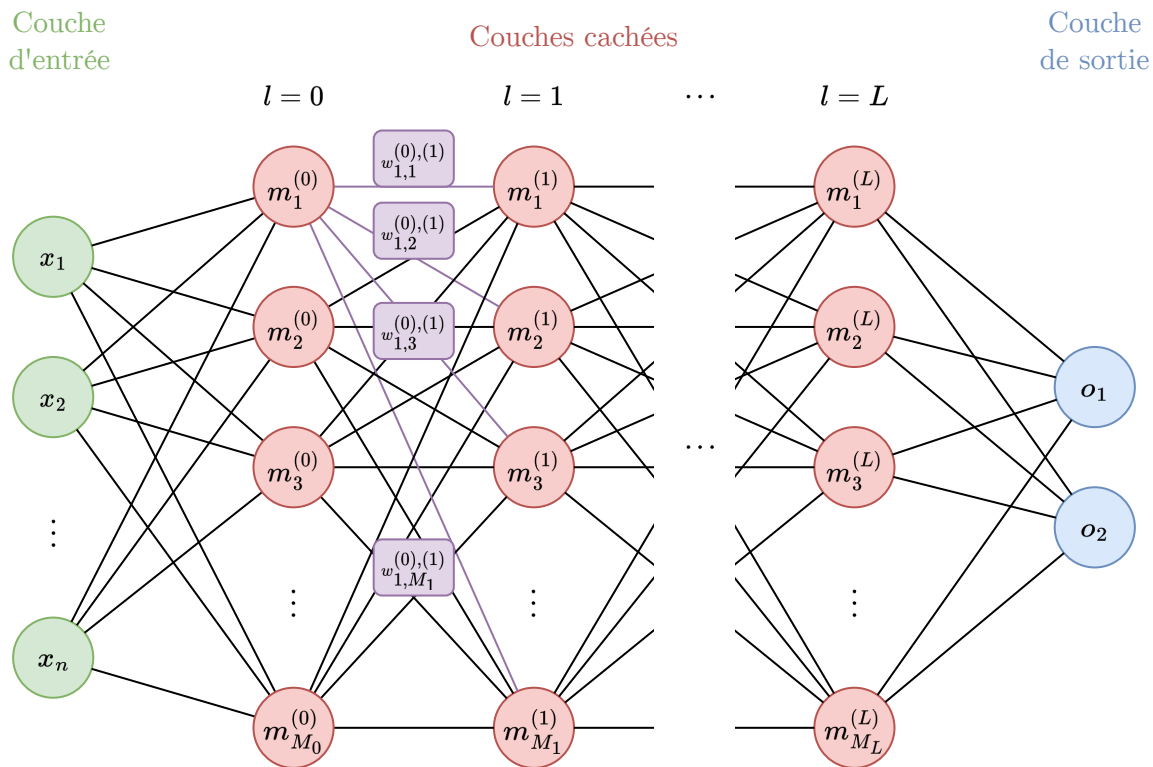


FIGURE 14 – Illustration d'un réseau de neurones dit "classique"

informations dans la couche d'entrée vers la couche de sortie finale. La détermination du nombre optimal de couches cachées au sein d'un RNA est un facteur très important, bien que ce choix ne soit pas évident en soi et qu'il nécessite un calibrage méticuleux. En règle générale, le niveau de complexité, inhérent à un modèle de réseau neuronal, est étroitement lié au nombre de ses couches cachées, une mesure appelée profondeur du réseau.

3. La dernière est la couche de sortie, située tout à la fin du réseau (en **bleu** dans la figure 14). Ces couches sont composées de neurones artificiels donnant les résultats obtenus par le réseau. Chaque neurone correspond à une classe cible distincte, le nombre de neurones au sein de cette couche, correspond au nombre de classe ($N_L = c$). Pour la détection d'objects, les classes correspondent aux différents concepts que le réseau est amené à classer.

Différentes architectures de réseaux neuronaux

Depuis de nombreuses années, l'utilisation des RNA a poussé la mise au point de nombreuses architectures différentes. L'objet de cette section n'est pas d'être exhaustive mais de présenter les architectures classiques ayant fondé les premiers RNA.

Multi-Layer Perceptrons (MLP)

Après avoir montré qu'un simple perceptron n'était pas suffisant pour résoudre des problèmes relativement complexes, la recherche d'architectures de RNA plus sophistiquées et plus polyvalentes, a conduit au développement du MLP. Le MLP apparaît comme une extension du perceptron original, conçu pour répondre à des tâches complexes exigeant une reconnaissance des formes nuancée.

Le MLP est connu dans le domaine de l'apprentissage profond comme étant un graphe orienté d'une succession de couches de neurones artificiels. Chaque neurone de la l -ième couche est connecté à tous les neurones de la $(l + 1)$ -ième couche. C'est pour cette raison que le MLP fait aussi référence à ce qu'est communément appelé le réseau entièrement connecté. Afin de mieux comprendre son apprentissage la figure 15 illustre un MLP traitant une image en entrée.

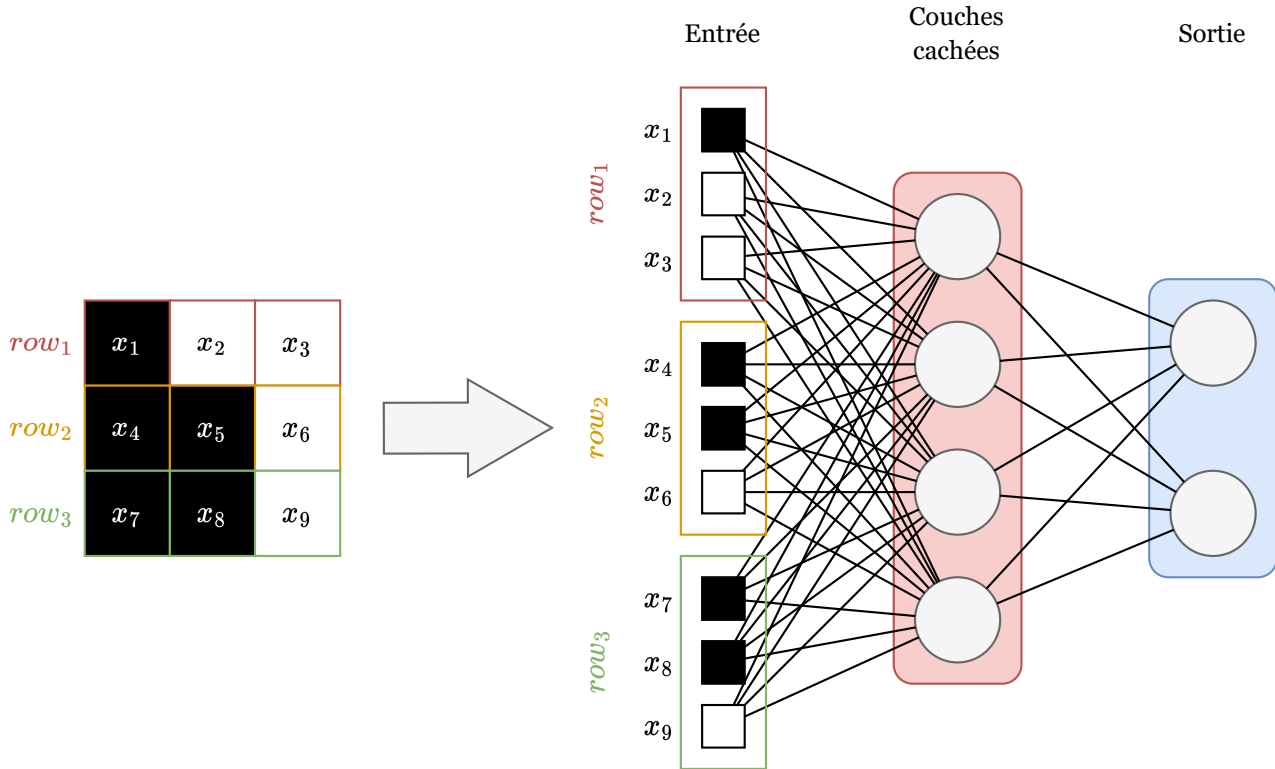


FIGURE 15 – Schéma exemple d'un Multi-Layer Perceptron (MLP)

Lorsqu'un échantillon de données est fourni en entrée d'un MLP, chaque neurone $m_k^{(0)}$, où $k \in \{1, \dots, M_0\}$, est associé à tous les éléments des données de l'entrée (voir figure 15). Le modèle suivant, calcule alors systématiquement la sortie de chaque neurone, en progressant à travers toutes les couches, jusqu'à atteindre la couche finale (L -ième). En prenant à titre d'exemple un neurone k dans la $(l + 1)$ -ième couche, sa sortie est déterminée par la formule suivante :

$$n_k^{(l+1)} = \varphi \left(\sum_{j=1}^{N_l} n_j^{(l)} \times w_{j,k}^{(l),(l+1)} + b_k^{(l+1)} \right) \quad (4)$$

avec $\varphi(\cdot)$ une fonction d'activation non linéaire (détaillée dans la partie 7.8b), $w_{j,k}^{(l),(l+1)}$ la connexion pondérée entre le neurone de la l -ième couche ($n_j^{(l)}$) et le neurone de la $(l + 1)$ -ième couche ($n_k^{(l+1)}$), tandis que N_l correspond au nombre de neurones dans la l -ième couche.

L'entraînement d'un MLP consiste à ajuster l'ensemble des paramètres $(\mathbf{w}, b)$ pour une problématique particulière en fonction des données d'entraînement fournis. L'ajustement des poids est modélisé par un problème d'optimisation. Rumelhart *et al.* [1986] ; LeCun *et al.* [1988] ont démontré la possibilité d'utiliser la descente de gradients, au travers d'une rétropropagation² [Werbos, 1974] afin de réaliser l'entraînement.

2. Pour plus de détails sur le fonctionnement de la "backpropagation", le site suivant explique en détails le

L'espace latent est une représentation mathématique des caractéristiques intrinsèques des données sous une forme plus compacte et abstraite [Hinton et Zemel, 1993]. Cet espace multidimensionnel est abstrait car il contient des valeurs de caractéristiques qui ne sont pas interprétables directement, mais permettant d'encoder une représentation interne significative des données en faisant abstraction de certaines informations. L'apprentissage d'un espace latent aide le modèle à mieux comprendre les données observées que les données elles-mêmes, les données étant un espace plus grand pour l'apprentissage. Le fait d'encoder les données d'entrée dans une dimension inférieure, c'est ce qui est appelé la "compression".

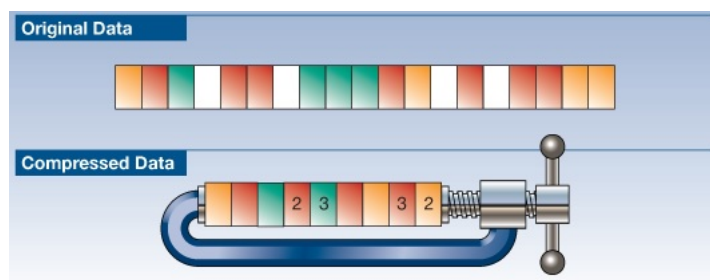


FIGURE 16 – Illustration de la compression de données. (Source Faust)

Cependant, le modèle entraîné doit être capable de reconstruire les données compressées, l'espace latent doit apprendre à ne garder que les informations intéressantes en évitant le bruit, permettant ainsi de se focaliser seulement caractéristiques importantes.

Par exemple, pour l'extraction de caractéristiques des images, les couches cachées permettent d'extraire de l'image en entrée les caractéristiques les plus importantes, permettant ensuite de détecter la présence d'objets dans différentes conditions. Sans espace latent, en travaillant directement avec les pixels de l'image, c'est un tâche difficile et coûteuse.

Les espaces latents peuvent être définis avec les termes "haute dimension" et "basse dimension", cela détermine le degré généralité ou de spécificité que cet espace doit apprendre. Un espace latent de petite dimension vise à ne garder que les caractéristiques les plus importantes pour représenter les données. Quant à un espace de grande dimension, il est sensible à des caractéristiques plus spécifiques des données d'entrée, ce qui peut impliquer un sur-apprentissage lorsqu'il n'y a pas suffisamment de données pour l'entraînement.

Auto-Encodeur (AE)

Dans le domaine de l'apprentissage non supervisé, l'Auto-Encoder (AE) a suscité une attention particulière, en raison de son potentiel permettant de construire des caractéristiques latentes dans des ensembles de données complexes. Dans l'architecture des réseaux neuronaux, les AE constituent une approche pratique pour : la réduction de dimension, l'apprentissage des caractéristiques, le débruitage, avec des applications couvrant divers domaines comme par exemple l'identification des codes postaux écrits manuellement LeCun *et al.* [1989]. Les travaux de Michelucci [2022] couvrent l'aspect mathématique et les concepts fondamentaux des AE, leur architecture, les fonctions d'activation, l'erreur de reconstruction ainsi que les applications comme la classification ou bien la détection.

Le principe fondamental des AE consiste à encoder les données d'entrée, dans un espace latent de dimension inférieure aux données en entrée, puis à les décoder pour reconstruire l'entrée

principe : <https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd>
ou bien cette vidéo : https://www.youtube.com/watch?v=isPiE-DBagM&list=PL3FW7Lu3i5Jsnh1rnUwq_Tcy1Nr7EkRe6

d'origine (cf. la figure 17).

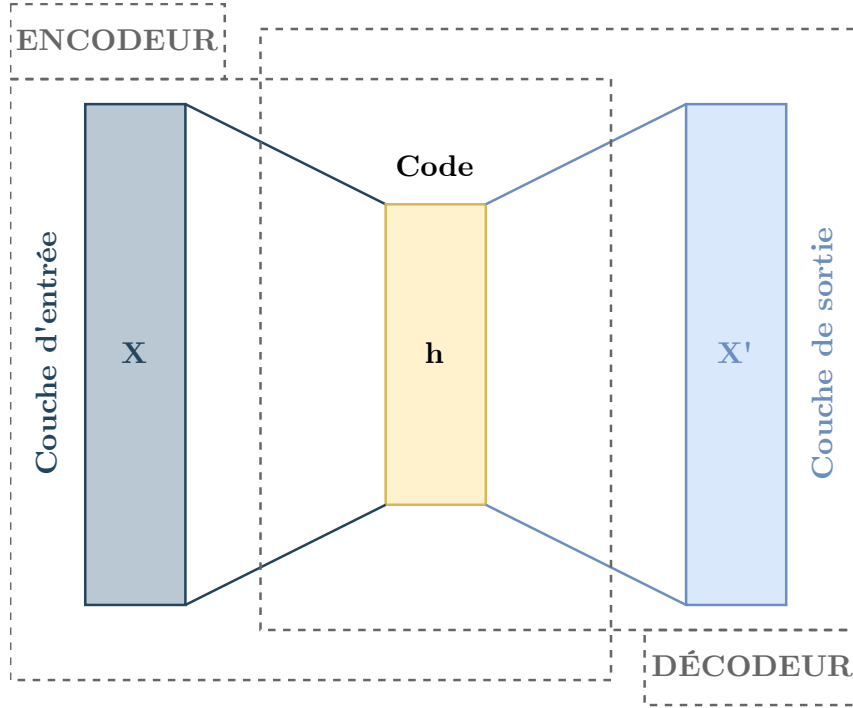


FIGURE 17 – Schéma d'un Auto-Encoder (AE)

Ce processus d'apprentissage est régi par des paramètres qu'il est possible d'apprendre et qui sont optimisés au cours des itérations [Bengio *et al.*, 2006].

Il existe diverses architectures des AE, souvent adaptées aux spécificités des données. Les AE convolutifs sont utiles pour les données images, en utilisant les couches convolutives pour capturer efficacement les informations spatiales [Masci *et al.*, 2011]. Les AE récurrents, quant à eux, sont adaptés aux données séquentielles, permettant de capturer et reconstruire les dépendances temporelles [Chung *et al.*, 2014].

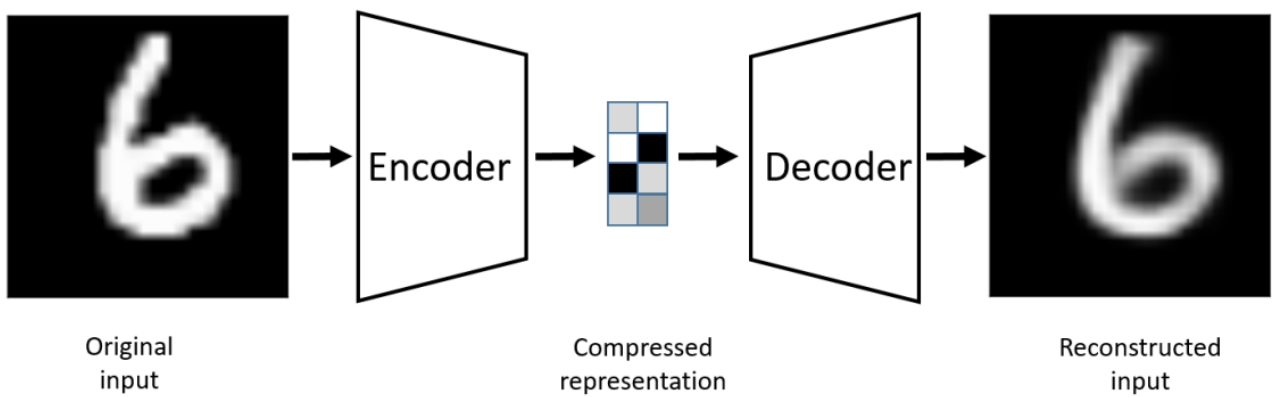


FIGURE 18 – Exemple de l'utilisation d'un AE. L'image est encodée en une représentation compressée, puis décodée.

En partant d'une image en entrée $\mathbf{x}$, l'encodeur calcule un vecteur encodé $\mathbf{h}$ de dimension N_h , nombre de neurones dans les couches cachées, de la manière suivante :

$$h_j = \varphi \left(\sum_{k=1}^m w_{k,j}^{(1)} \times x_k + b_j^{(1)} \right) \quad (5)$$

avec $w^{(1)}$, une matrice de poids de taille $N_h \times m$, et $b^{(1)}$ le vecteur de biais de taille N_h . Quant à $\varphi(\cdot)$ il représente la fonction d'activation.

Une fois le vecteur encodé calculé, le décodeur va essayer de reconstruire le vecteur x depuis h . Le résultat reconstruit, est noté $\hat{x}$ et calculé grâce à l'équation :

$$\hat{x}_j = \varphi \left(\sum_{k=1}^{N_h} w_{k,j}^{(2)} \times h_k + b_j^{(2)} \right) \quad (6)$$

avec $w^{(2)}$ une matrice de poids de taille $m \times N_h$ et $b^{(2)}$ un vecteur de biais avec m éléments.

Les AE sont utilisés dans la détection des anomalies, où les écarts, par rapport aux données reconstruites, mettent en évidence les anomalies [An et Cho, 2015]. En outre, la réduction de la dimensionnalité obtenue par les AE, est particulièrement bénéfique pour les tâches en aval, telles que la classification et le regroupement [Han *et al.*, 2015]. Leur capacité à débruiter les données, a des implications dans les domaines où de la réduction du bruit est nécessaire [Vincent *et al.*, 2010].

L'objectif d'un AE est de réduire l'erreur de reconstruction entre x et $\hat{x}$. Pour ce faire, l'AE utilise la “*MSE*” en utilisant les paramètres $\theta = \{\mathbf{w}^{(2)}, b^{(2)}, \mathbf{w}^{(1)}, b^{(1)}\}$ avec :

$$MSE(x, \hat{x}) = \|x - \hat{x}\| \quad (7)$$

L'apprentissage des AE implique généralement des algorithmes d'optimisation, tels que la descente de gradient stochastique, couplée à la rétropropagation [Rumelhart *et al.*, 1986]. Bien que des problèmes, tels que le sur-apprentissage et la sélection d'architectures optimales, persistent, des variations innovantes et des techniques de régularisation continuent d'être explorées, pour atténuer ces problématiques [Kingma et Ba, 2014 ; Srivastava *et al.*, 2014].

L'AE est une approche fondamentale de l'apprentissage non supervisé, permettant d'offrir de nouvelles perspectives en matière de découverte de caractéristiques dans des données relativement complexes. Avec des implications dans de nombreux domaines, son évolution dynamique à travers des architectures différentes montre des transformations, comparé aux ANN pour une meilleure utilisation de tout type de données. L'augmentation du nombre de couches cachées au sein du réseau, introduit la notion de profondeur d'un ANN découlant sur le domaine de l'apprentissage profond, appelé plus communément le “*Deep Learning (DL)*”.

CNN

Les balbutiements des RNA (*ANN*) se base sur le modèle informatique d'un neurone inspiré du monde biologique [McCulloch et Pitts, 1943]. Rosenblatt [1958] a mis au point le neurone artificiel le plus connu, le “Perceptron”. Il s'agit d'un modèle extrêmement simple, composé d'un seul neurone, et ne pouvant traiter que des problèmes linéairement séparables. Ce manque de capacité d'abstraction rend le perceptron inapte à modéliser des relations complexes, comme dans le cas bien connu de la fonction XOR³ [Minsky et Papert, 1969], qui nécessite une frontière de décision non linéaire.

3. La fonction XOR (OU exclusif) trouve son origine dans l'algèbre et la logique booléennes. En algèbre booléenne, la fonction XOR prend deux entrées binaires (0 ou 1) et renvoie une sortie binaire : $XOR(0,0) = 0$, $XOR(0,1) = 1$, $XOR(1,0) = 1$, et $XOR(1,1) = 0$.

Pour pallier cette limitation, le domaine de l'intelligence artificielle a fait évoluer ce concept vers des architectures plus complexes, notamment avec le développement des réseaux de neurones multicouches [Rumelhart *et al.*, 1986] (ou *MLP*), introduisant plusieurs couches de neurones entre l'entrée et la sortie. L'ajout d'au moins une couche cachée munie d'une fonction d'activation non linéaire permet au réseau d'approximer des fonctions plus complexes, et notamment de résoudre le problème XOR.

Cette évolution structurelle n'a été rendue pleinement efficace qu'avec la découverte de la rétropropagation du gradient, une méthode d'apprentissage supervisé qui permet d'ajuster automatiquement les poids dans chaque couche, abordée plus en détails dans la section III. L'introduction de cette méthode a permis de construire des réseaux profonds performants dans divers domaines.

C'est dans ce contexte qu'ont émergé les “*CNN*” [LeCun *et al.*, 1989], prolongeant le principe du MLP mais en tirant parti des structures spatiales présentes dans les images. Grâce à la rétropropagation, les CNN ont permis des avancées majeures en TLN, en vision par ordinateur [LeCun *et al.*, 1990 ; Lecun *et al.*, 1998 ; Das *et al.*, 2010] et en reconnaissance vocale.

La vision par ordinateur est un domaine visant à permettre aux machines d'interpréter et de comprendre les informations visuelles du monde. Cela englobe diverses tâches telles que la classification d'images, la détection d'objets et la segmentation sémantique. Les approches existantes s'appuient de plus en plus sur des modèles d'apprentissage profond. Les travaux de Krizhevsky *et al.* [2012] utilisant des CNN pour la tâche de classification d'images sur la base de données ImageNet [Deng *et al.*, 2009] ou bien l'introduction des réseaux résiduels par He *et al.* [2016] ont fait progresser le domaine de manière significative. Voici plus en détails les principales tâches du domaine de la vision par ordinateur.

La classification a pour objectif de catégoriser les images en fonction de leurs contenus. Il existe de nombreuses applications comme le recensement à distance des animaux pour estimer les populations et les densités au cours du temps. Les auteurs de Miao *et al.* [2019], à partir d'image capturée dans la nature, arrivent à extraire les motifs, les formes et les couleurs de chaque espèce, permettant ainsi de reconnaître les différentes espèces et de les classer correctement.

La tâche de segmentation consiste à étiqueter chaque pixel dans des images ou des trames vidéo. Le modèle U-Net [Ronneberger *et al.*, 2015], utilisé dans le domaine médical, a été appliqué à la segmentation d'images cérébrales. Ranjbarzadeh *et al.* [2021] utilise le même type d'images issues d'IRM pour distinguer les différents éléments qui la composent (muscles, os, organes...). Cette approche permet d'isoler les zones d'intérêt et d'identifier d'éventuelles anomalies.

Une autre tâche est la détection d'objets, cherchant à détecter et à localiser la position exacte des objets. Dans un système de surveillance devant détecter des personnes dans des séquences vidéos [Paul *et al.*, 2013], le système traite chaque image de la vidéo pour identifier et localiser les silhouettes humaines. Il s'agit de reconnaître la forme d'un corps humain ou bien des mouvements indiquant une présence humaine.

Les principales notions permettant de comprendre le fonctionnement des CNN vont être abordées dans la section suivante.

Dans cette section, nous allons aborder les concepts clés des réseaux neuronaux : la convolution pour l'extraction des caractéristiques, la réduction de dimension (“pooling”) pour réduire la taille des données, le biais pour ajuster les sorties, les fonctions d'activation pour la non-linéarité, la normalisation pour la standardisation des données, rétropropagation pour l'apprentissage du réseau, fonctions de perte pour mesurer les erreurs de prédiction et les métriques pour évaluer les performances du modèle. Ces éléments sont essentiels pour comprendre le fonctionnement et

l'efficacité des réseaux neuronaux.

Convolution

Le terme “convolution” est largement employé dans le traitement d'images avec des réseaux de neurones. Une convolution est un concept fondamental, tout comme l'addition ou la multiplication, contrairement à elles, la convolution (notée “ $*$ ”) ne s'applique pas directement sur les nombres mais sur des listes de nombres (vecteurs ou matrices) ou bien lors de combinaison de fonctions.

Des masques de convolution, aussi appelés noyaux (“*kernels*”), sont appliqués sur les pixels des images. Cela permet de transformer l'image afin de faire émerger des propriétés, comme d'accentuer les contours, d'atténuer le bruit.

Définition mathématique

Du point de vue mathématique, le produit de convolution est noté comme suit :

$$(f * g)(i) = \sum_{k=-\infty}^{+\infty} f(k) \cdot g(i - k) \quad (8)$$

où f et g sont les deux séquences convoluées, i l'indice auquel la séquence résultante est évaluée, et k la variable d'indexation représentant le décalage.

Cette définition discrète est la plus adaptée au traitement d'images numériques, contrairement à la version continue souvent rencontrée en traitement du signal. La convolution possède plusieurs propriétés fondamentales telles que la commutativité (9), l'associativité (10) et la distributivité (11) :

$$f * g = g * f \quad (9)$$

$$(f * g) * h = f * (g * h) \quad (10)$$

$$f * (g + h) = (f * g) + (f * h) \quad (11)$$

Cependant, dans le cadre des CNN, ces propriétés ne s'appliquent plus directement. Le pipeline de traitement inclut des opérations supplémentaires telles que les fonctions d'activation non linéaires (voir section III), le padding, le stride ou la normalisation, qui modifient la linéarité et influencent le comportement global du système. Par conséquent, bien que la convolution reste mathématiquement bien définie, son intégration dans les CNN s'inscrit dans un cadre plus complexe, altérant certaines de ses propriétés algébriques.

Utilisation des convolutions en apprentissage profond

Dans cette partie, nous allons étudier la convolution classique utilisée dans deux dimensions uniquement et ensuite nous introduirons comment les réseaux d'apprentissage profond font pour traiter des images en couleur qui peuvent être vus comme un cube en trois dimensions (partie III).

Convolution 2D classique

Dans le contexte des réseaux de neurones convolutifs, cette opération s'applique généralement à des matrices multidimensionnelles, telles que les images. Le noyau de convolution (ou filtre) est alors déplacé localement sur l'image pour générer une carte de caractéristiques. L'équation de convolution est étendue de manière analogue aux dimensions concernées, notamment en deux dimensions.

Pour une image I et un noyau K , la convolution 2D est donnée par :

$$(I \times K)[i, j] = \sum_m \sum_n I[m, n] \times K[i - m, j - n] \quad (12)$$

avec I l'image d'entrée, K le noyau de convolution, i et j les indices de la position dans la sortie et m et n les indices de sommation qui parcourent les dimensions du noyau.

L'étude se limite au cas bidimensionnel : une matrice 2D en entrée est combinée à un noyau 2D pour produire une sortie 2D. Les figures illustrant cet exemple sont inspirées de Dumoulin et Visin [2018]. Le noyau utilisé est une petite matrice de poids, définie comme suit :

0	1	2
2	2	0
0	1	2

FIGURE 19 – Noyau de convolution de dimension 3×3 .

La figure 20 montre une réduction des dimensions entre l'entrée et la sortie, causée par l'absence de *padding* (p). Cette technique consiste à ajouter des pixels, généralement nuls, autour de l'image pour préserver ses dimensions après convolution, ce qui est utile dans certaines tâches de traitement d'images.

Le *stride* (s) définit le pas de déplacement du noyau. Un stride de 1 conserve un déplacement pixel par pixel, tandis qu'un stride plus grand réduit les dimensions de la sortie, diminuant ainsi le coût computationnel.

Dans la figure 20, la grille **bleue** est appelée la “carte de caractéristiques d'entrée”, elle a une dimension 5×5 , les chiffres en indice sur la zone **bleu foncé** représentent les valeurs du noyau, la flèche en **rouge** montre le déplacement du filtre appliqué sur les données d'entrée afin d'obtenir la carte de sortie en **vert**. Avec ce schéma, cela permet de mieux comprendre comment sont calculées les valeurs lors d'une convolution discrète.

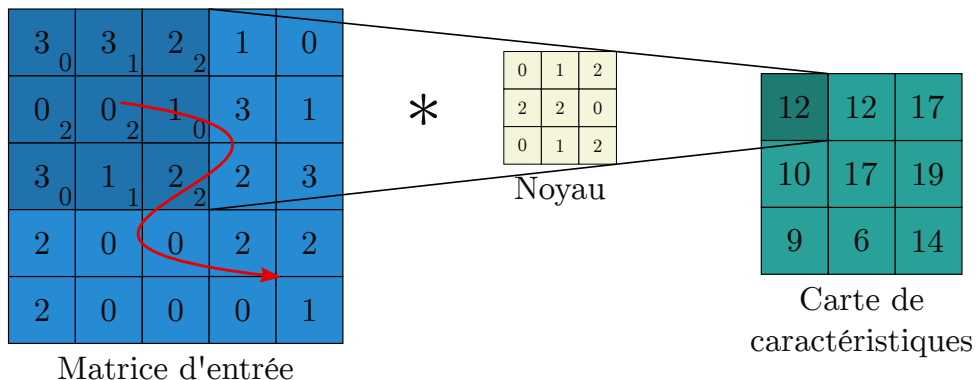


FIGURE 20 – Illustration d'une convolution discrète.

La figure 21 illustre une image ayant une padding $p = 1$ ainsi qu'un stride $s = 2$. Il est possible de voir que le padding ajoute des zéros tout autour de la carte d'entrée et que le stride fait se déplacer le noyau de deux entre chaque calcul. En combinant ces deux modifications, la carte de sortie a les mêmes dimensions que la figure 20 mais n'ont pas les mêmes valeurs.

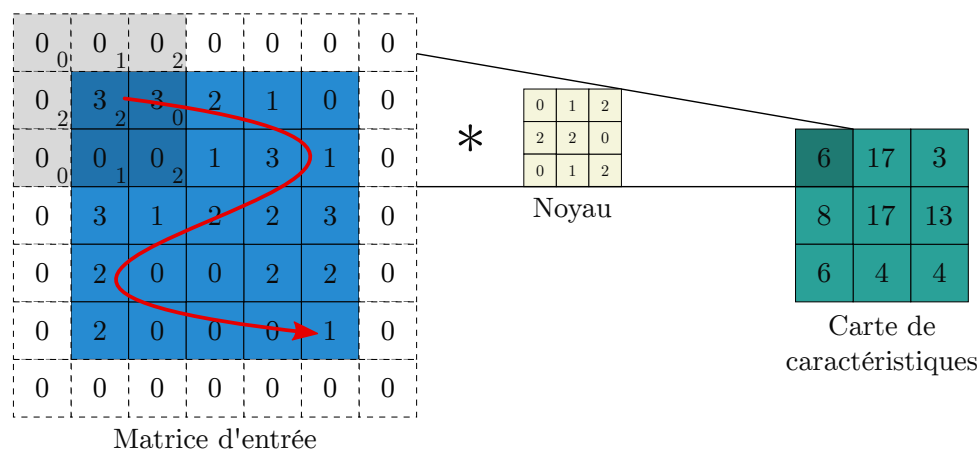


FIGURE 21 – Exemple de calcul d’une convolution discrète avec $p = 1$ et $s = 2$.

Cette méthode équivaut à la convolution théorique en deux dimensions, mais en pratique, la plupart des images ont plusieurs canaux et sont donc tridimensionnelles. Leur traitement diffère de ce qui a été décrit avant ; c’est ce que nous allons détailler ci-après.

Convolution en apprentissage profond

Dans les applications d’apprentissage profond utilisant des images, les données sont souvent représentées en 3D plutôt qu’en 2D. Par exemple, les images en couleur sont composées de trois canaux : Rouge, Vert et Bleu (RVB). Les canaux ajoutent une dimension représentant l’intensité de la couleur aux dimensions spatiales 2D (largeur et hauteur) donnant ainsi une structure globale en 3D. Ce format 3D est important car il permet aux réseaux de neurones de traiter l’ensemble des informations contenues dans l’image, y compris les motifs de couleur et les textures des différents canaux.

Dans le fonctionnement des réseaux de neurones convolutifs, il existe une distinction importante entre les termes “noyau” et “filtre”.

Un noyau désigne une matrice 2D de poids, appliquée à un seul canal d’entrée lors d’une opération de convolution. En revanche, un filtre correspond à un ensemble de noyaux, chacun appliqué à un canal différent de l’entrée multicanale (par exemple, R, G, B pour une image en couleur).

Les résultats de chaque noyau sont ensuite sommés, et un biais est ajouté, pour produire une carte de caractéristiques unique, comme illustré dans la figure 23. Ainsi, dans un CNN, un filtre produit une carte de sortie à partir de l’ensemble des canaux d’entrée, comme illustré dans la figure 22.

Dans le cas d’une entrée monocanal (comme une image en niveaux de gris), le noyau et le filtre peuvent être confondus. Mais en apprentissage profond, le filtre est toujours une structure 3D formée de plusieurs noyaux 2D, un pour chaque canal d’entrée.

Avant d’illustrer le fonctionnement multicanal avec la figure 22, il convient de formaliser le calcul d’une sortie convolutive sur une zone locale.

Soit une région locale Ω , la sortie convolutive associée à un filtre donné s’exprime comme une somme de produits terme à terme entre les valeurs d’entrée $f(i, j, k)$ et les poids du noyau

$g(i, j, k)$:

$$g(\text{centre de } \Omega) = \sum_{(i,j,k) \in \Omega} f(i, j, k) \times g(i, j, k) \quad (13)$$

où (i, j) parcourent les dimensions spatiales et k l'indice des canaux de l'image. Cette expression constitue le principe de l'opération convolutive multicanal, appliquée à chaque position de l'image en faisant glisser le filtre.

La figure suivante montre comment, dans le cas d'une image en couleur, chaque canal est associé à un noyau distinct, et comment leurs sorties sont ensuite combinées pour produire une carte de caractéristiques unique.

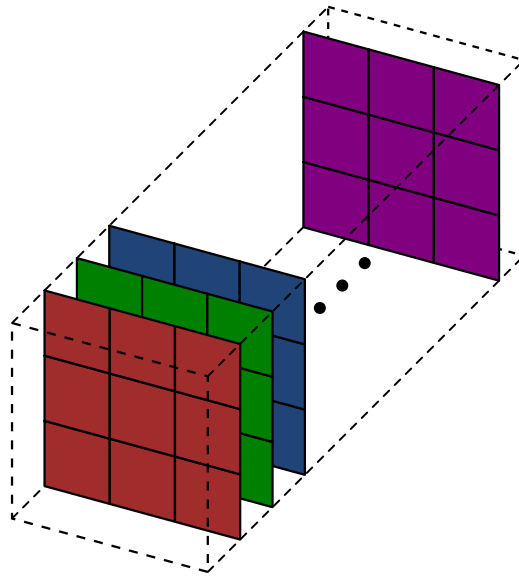


FIGURE 22 – Convolution pour chaque canal d'entrée avec son propre noyau pour obtenir des cartes de caractéristiques contenant les informations de chaque canal.

Le biais, propre à chaque filtre, est ensuite ajouté au canal de sortie afin de moduler la réponse de la fonction d'activation, en particulier en présence de données d'entrée faibles ou peu informatives. Il contribue à l'adaptabilité du réseau face à la variabilité des entrées, en améliorant le traitement des cas moins discriminants.

Ce raisonnement s'étend à l'ensemble de la couche convolutive. Chaque filtre opère localement sur l'entrée, équivalant à un neurone connecté à une région restreinte de l'image, réduisant ainsi la connectivité.

L'utilisation de plusieurs filtres génère autant de cartes de caractéristiques, qui sont ensuite concaténées pour constituer la sortie globale de la couche. Le nombre de cartes en sortie correspond ainsi au nombre total de convolutions distinctes, et non au nombre de neurones au sens strict.

Ce fonctionnement est directement lié à l'opération de convolution décrite précédemment. En effet, chaque cellule de la carte de caractéristiques de sortie est le résultat d'un calcul neuronal appliqué sur une petite région locale de l'image en entrée.

Ainsi, le calcul de la sortie convolutive à une position donnée correspond exactement au schéma d'un neurone artificiel, comme illustré dans la figure 24, avec ses différents composants et l'ordre dans lequel ils sont utilisés.

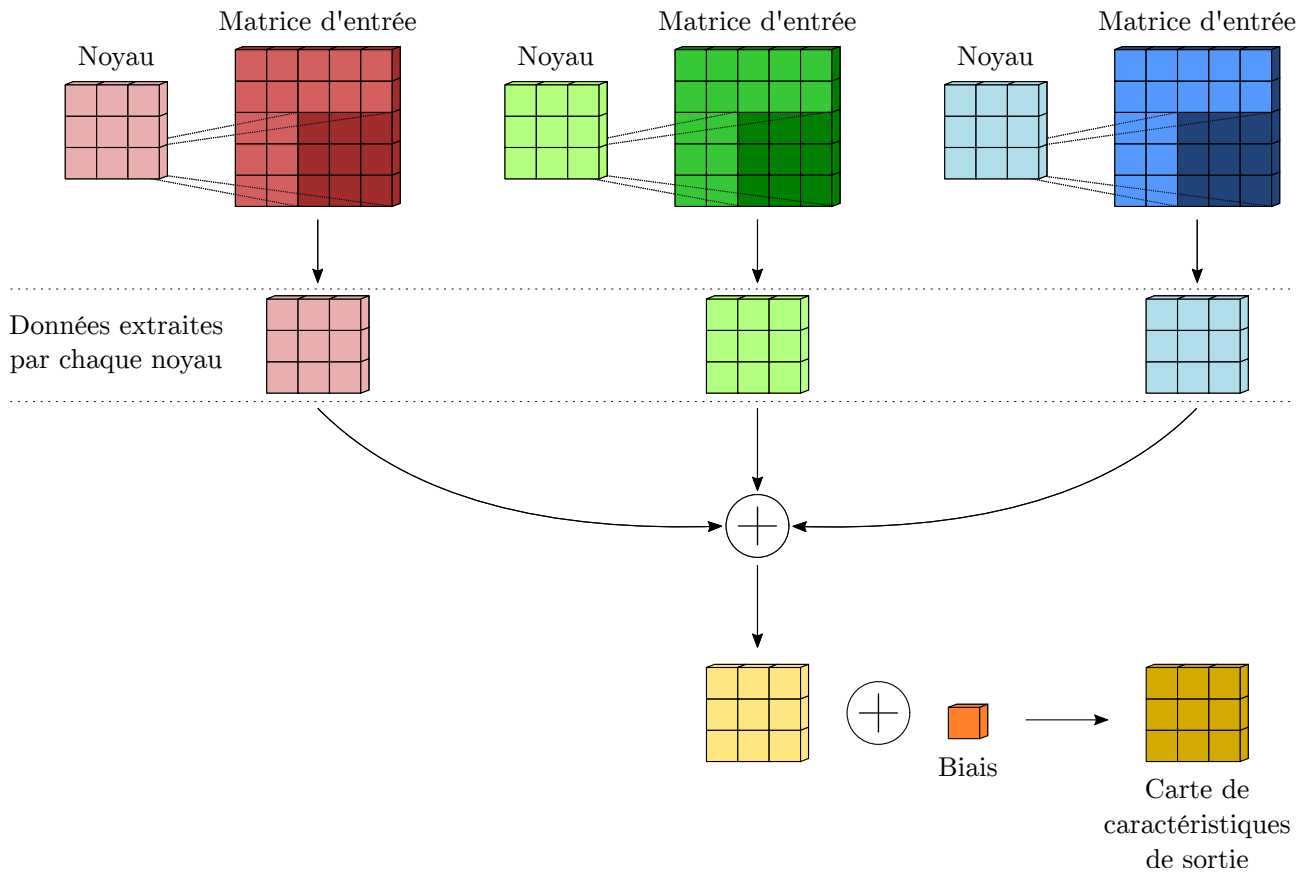


FIGURE 23 – Illustration d'un filtre composé de plusieurs noyaux différents pour chaque canal des données en entrée, où les sorties seront sommées entre elles ainsi qu'avec le biais pour obtenir les caractéristiques extraites. Inspiré de Shafkat.

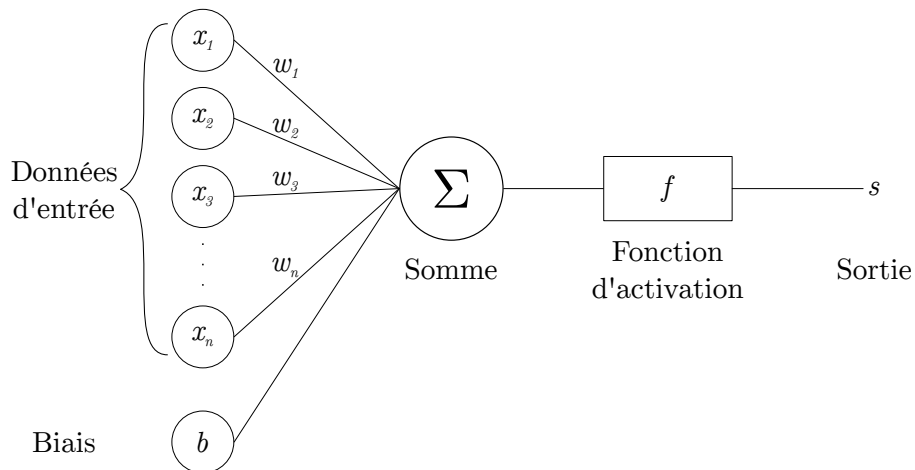


FIGURE 24 – Schéma du processus global d'un neurone artificiel.

L'introduction d'un biais permet de réguler les sorties des neurones avant l'application de la fonction d'activation. Cette dernière correspond à une opération non linéaire appliquée aux sorties afin d'introduire de la non-linéarité dans le réseau, condition nécessaire à l'approximation de fonctions complexes (voir section III).

Sans biais, la sortie d'un neurone résulte uniquement d'une combinaison linéaire des entrées et des poids, limitant ainsi la capacité d'approximation du réseau. Il autorise notamment des sorties non nulles même lorsque les entrées sont nulles, contribuant ainsi à limiter des phénomènes indésirables comme la disparition du gradient durant l'apprentissage.

La figure 26b illustre cet effet pour la fonction d'activation ReLU, en montrant l'impact de deux valeurs de biais distinctes sur la sortie d'un neurone.

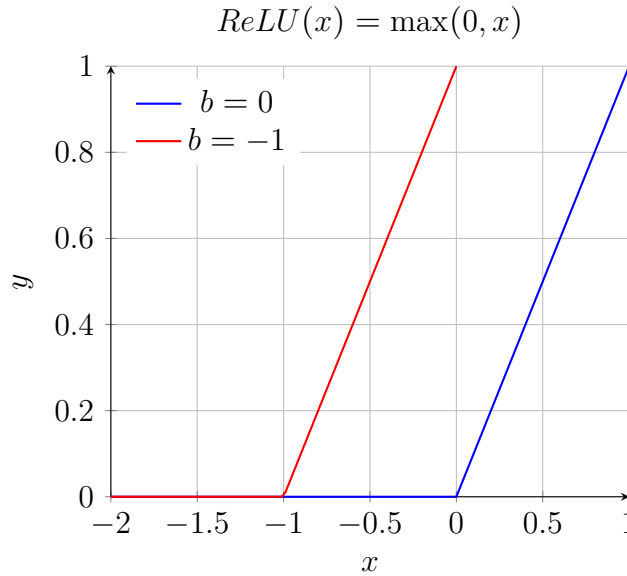


FIGURE 25 – Exemple de valeurs de sorties pour la fonction d'activation ReLU avec deux biais différents. Inspiré de Turing [2024].

Les biais sont mis à jour au cours du processus d'apprentissage d'une manière analogue aux poids, grâce aux méthodes de rétropropagation et de descente de gradient, ce qui garantit que les poids et les biais sont optimisés pour minimiser la fonction de perte (partie III). Ce processus d'optimisation permet au réseau d'apprendre les valeurs de biais les plus efficaces en union avec les poids, améliorant ainsi les performances globales et la capacité de généralisation du modèle.

Le fait d'avoir une équation linéaire pour la sortie d'un neurone a ses limites. Un réseau neuronal linéaire composé de n couches cachées avec m neurones ainsi qu'une fonction d'activation linéaire est équivalent à un réseau sans couches cachées. Voici la démonstration mathématique :

$$\begin{aligned}
 y = f(x) &= b_n + w_n (b_{n-1} + w_{n-1} (\dots (b_1 + w_1 x) \dots)) \\
 &= b_n + w_n b_{n-1} + w_n w_{n-1} b_{n-2} + \dots + w_n w_{n-1} \dots w_1 x \\
 &= b' + w' x
 \end{aligned} \tag{14}$$

D'après l'équation précédente, sans non-linéarité, un réseau de neurones se comporterait essentiellement comme un modèle de régression linéaire, sans pouvoir modéliser des relations plus complexes, alors qu'un réseau composé de deux couches avec de la non-linéarité est capable d'approximer toutes les fonctions [Costarelli et Spigler, 2013].

Il est donc nécessaire d'intégrer de la non-linéarité au sein des réseaux pour approximer ces relations, et ainsi avoir un intérêt d'empiler les couches cachées. Cet ajout se fait grâce aux fonctions d'activation non-linéaires qui vont être abordées par la suite.

Fonction d'activation

Le concept de “fonction d'activation” provient du domaine de la biologie avec “potentiel d'activation”, étant le seuil de stimulation nécessaire pour déclencher une réponse du neurone.

Il existe de nombreuses fonctions d'activation, certaines sont abordées dans la suite, notamment sigmoïde, tangente hyperbolique, ReLU et SiLU dont leurs graphiques sont tracés dans la

figure 26 ci-dessous.

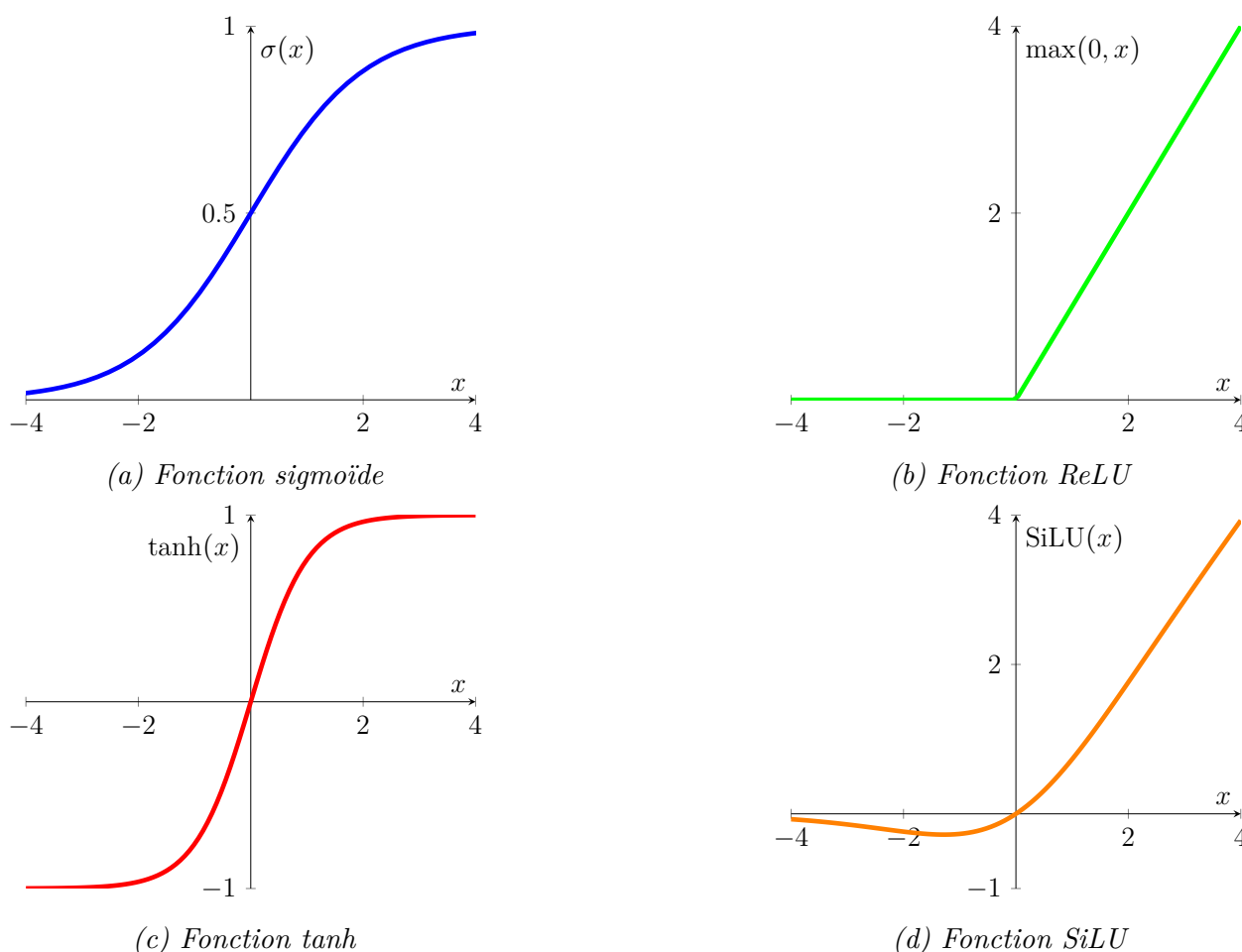


FIGURE 26 – Représentations graphiques des fonctions d’activation majeures.

La fonction d’activation en elle-même est importante lors de son analyse des données en entrées, mais elle est aussi utile pour la rétropropagation (voir partie III), plus précisément pour calculer les gradients, qui servent à l’apprentissage d’un réseau de neurones. Si une fonction n’est pas différentiable ou si sa dérivée présente un comportement problématique, cela peut gêner le processus d’apprentissage du réseau. Ces comportements problématiques incluent notamment la disparition ou l’explosion du gradient, deux phénomènes majeurs pouvant empêcher la mise à jour efficace des poids du réseau.

Le premier est celui de la disparition de gradient (“*vanishing gradient*”), c’est un problème connu dans le domaine sur lequel de nombreux chercheurs ont travaillé [Hochreiter, 1998 ; Park *et al.*, 2019 ; Jiao *et al.*, 2017]. Il se produit lors de la phase de rétropropagation, quand les gradients calculés au travers de l’ensemble des couches devient trop petit. Un exemple de fonction d’activation ayant ce problème est la fonction “sigmoid”.

Le second problème est l’inverse de la disparition de gradient, c’est celui de l’explosion de gradient (“*exploding gradient*”). Celui-ci intervient aussi lors de la phase de rétropropagation, mais, à l’inverse de la disparition de gradient, les gradients calculés à travers les couches augmentent exponentiellement, devenant très grands. Ce problème peut conduire à des mises à jour de poids massives lors de l’apprentissage, provoquant de l’instabilité au sein du réseau et empêchant le modèle de converger correctement Pascanu *et al.* [2013] ; Goodfellow *et al.* [2016] ; Bengio *et al.* [1994]. Cela peut arriver avec la fonction d’activation ReLU.

Pour limiter les phénomènes de disparition ou d’explosion du gradient, plusieurs mécanismes de régulation ont été introduits. Parmi eux, la normalisation par lot (*Batch Normalization*) [Ioffe et Szegedy, 2015] et l’injection de bruit stochastique via le *Dropout* [Srivastava *et al.*, 2014] sont largement utilisés dans les architectures modernes. Ces techniques complètent les fonctions d’activation en régularisant les activations intermédiaires et en facilitant l’apprentissage dans les réseaux profonds.

Cependant, le comportement du gradient dépend aussi de la forme mathématique des fonctions d’activation. L’analyse de leurs dérivées, impliquées directement dans le calcul des gradients lors de la rétropropagation, permet de mieux comprendre leur influence sur la stabilité de l’apprentissage. Les dérivées de ces fonctions sont représentées dans la figure 27.

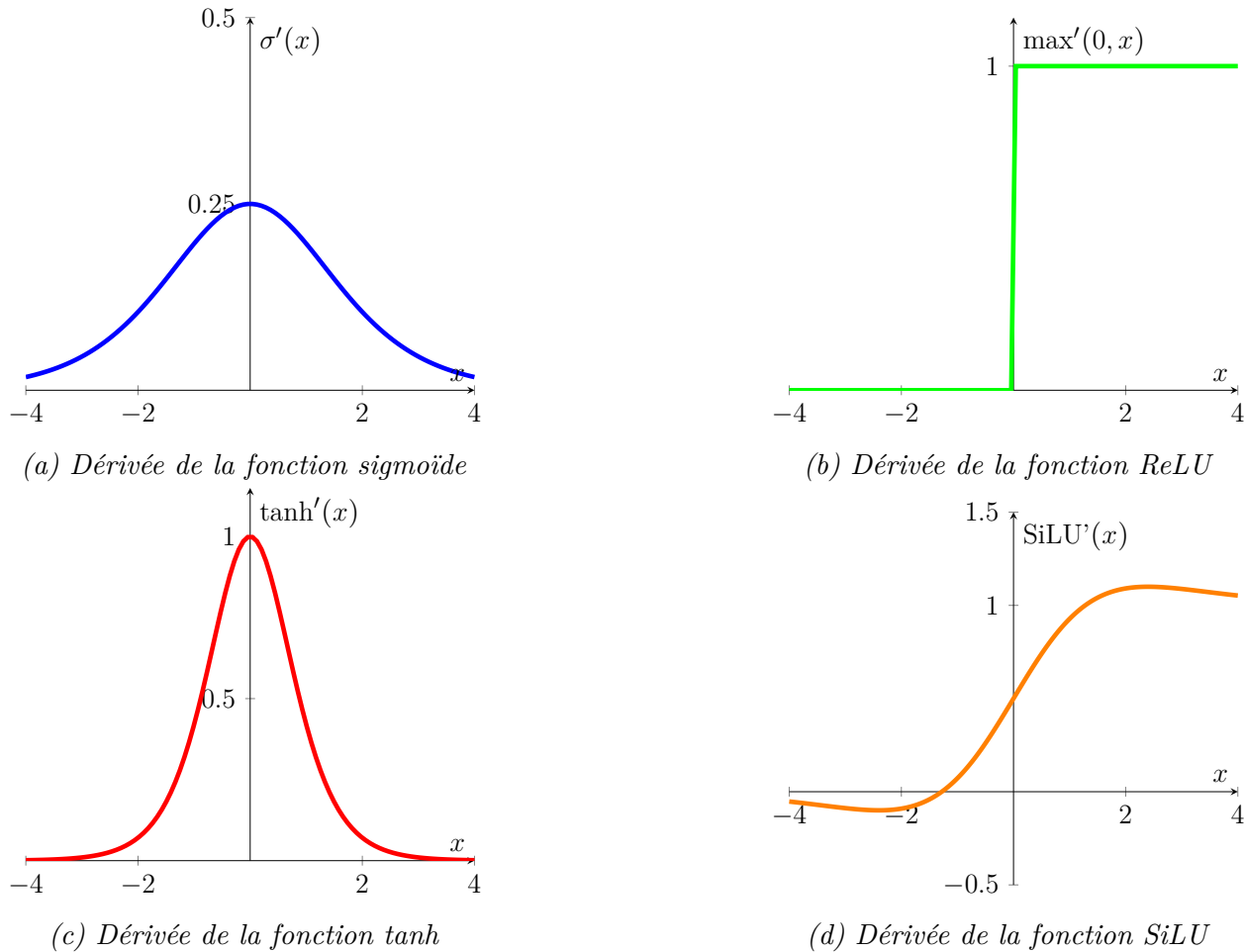


FIGURE 27 – Représentations graphiques des dérivées des fonctions d’activation majeures.

Pooling

L’opération dite de “pooling” effectue une réduction des dimensions des données en entrée en utilisant les valeurs d’un voisinage au sein de celles-ci, le but étant de réduire la complexité de calcul et le nombre de paramètres du modèle pour éviter le sur-apprentissage. Nous allons illustrer ce fonctionnement en prenant comme exemple les deux principaux types de réduction : “*Max Pooling (MP)*” et “*Average Pooling (AP)*”. Le fonctionnement peut être assimilé à celui de la convolution, car un filtre est utilisé pour extraire les informations d’une région de la carte de caractéristiques en entrée, mais ce filtre ne possède pas de poids, seulement une taille, un pas et une opération associée.

Afin d'illustrer ces différences, la figure 28 représentant les deux sorties de ces stratégies.

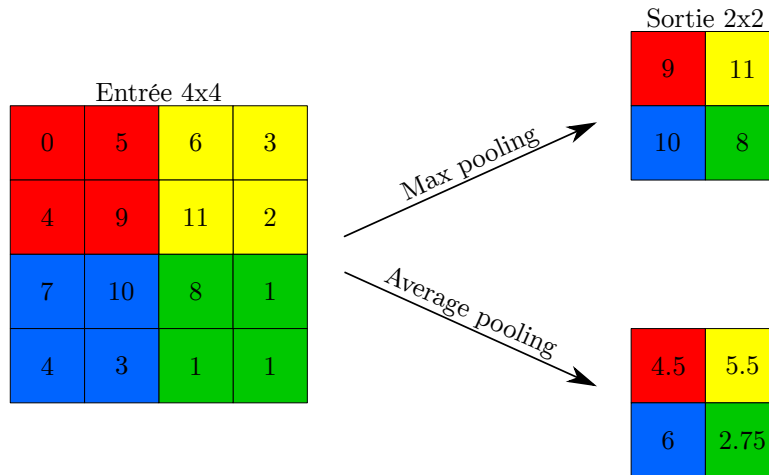


FIGURE 28 – Illustration de la stratégie “max pooling” et “average pooling”. Les données en entrée sont de dimension $H = L = 4$, avec un filtre de $k = s = 2$ et obtenir des résultats de dimension $H' = L' = 2$.

Dans la figure 28, la carte de caractéristiques d’entrée a une dimension de 4×4 (16 données), avec une fenêtre de regroupement de 2×2 . En sortie, la carte de caractéristiques est de taille 2×2 (4 données), donnant un facteur de réduction, pour ces données 2D, de 4.

Les méthodes de regroupement sont choisies en fonction de leur capacité à réduire les dimensions spatiales tout en préservant les informations importantes. Leurs utilisations et leurs performances sont difficilement comparables, cela dépend du contexte de leurs applications (tâche, type de données...).

Par exemple, dans le cas de classification de dents, Li *et al.* [2019] a comparé les deux méthodes et c’est le MP qui a obtenu les meilleures performances. Bieder *et al.* [2021] compare différentes stratégies de réduction de dimensions, dont MP et AP. Les performances sont similaires en termes de précision de classification sur les données de Google Open Images v5 [Krasin *et al.*, 2017].

Dans la section suivante, nous allons aborder les fonctions de perte qui font le lien entre le traitement des données et l’optimisation du modèle.

Fonction de pertes

Les fonctions de perte (“*loss functions*”) permettent de formuler l’apprentissage comme un problème de minimisation. Elles mesurent la distance entre les prédictions du modèle et la vérité terrain, définie comme l’ensemble des données considérées comme correctes. Leur minimisation, réalisée par ajustement itératif des paramètres du réseau, guide l’apprentissage du modèle vers des prédictions proches de la vérité terrain.

Selon leur formulation, certaines fonctions de perte peuvent influencer le comportement global du modèle, en le rendant plus robuste à certaines données ou en accentuant l’impact de certaines erreurs.

Deux fonctions de perte couramment utilisées dans les réseaux neuronaux sont étudiées ici. L’erreur quadratique moyenne (*MSE*), associée aux tâches de régression, et l’entropie croisée (*Cross-Entropy*), liée à la classification, peuvent également intervenir dans des tâches plus complexes comme la détection d’objets, où les deux types de sorties sont présents.

Erreur quadratique moyenne (MSE)

La MSE est l'une des fonctions de perte les plus connues car elle est relativement simple à comprendre. Elle se calcule grâce à l'équation suivante :

$$\mathcal{L}_{\text{MSE}} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (15)$$

La MSE correspond à la différence entre les prédictions du modèle $\hat{y}$ et la vérité terrain y sur l'ensemble des exemples du jeu de données n . Cette différence est élevée au carré, ce qui fait que la MSE n'est jamais négative. Grâce à cette mise au carré, il est possible de détecter les prédictions aberrantes du modèle, car ces erreurs seront fortement mises en évidence. Cependant, cela peut poser problème si le modèle fait une seule très mauvaise prédiction, car l'erreur sera amplifiée. Or, de manière générale, ces quelques prédictions aberrantes n'ont pas de valeur importante, c'est plutôt un modèle bien équilibré qui est recherché, capable de donner des bons résultats dans la majorité des cas.

La MSE offre un moyen clair de quantifier la variance des prédictions d'un modèle, ce qui facilite l'identification de la distance moyenne entre les prédictions et les valeurs réelles. Son utilisation est également due à ses propriétés mathématiques, telles que la différentiabilité, qui simplifie le processus de recherche de l'erreur minimale au cours de l'apprentissage du modèle.

Voyons maintenant l'autre fonction spécialisée pour la classification, l'entropie croisée.

Entropie croisée (Cross-entropy)

Cette fonction de perte est utilisée dans les tâches de classification, car elle permet de quantifier la différence entre deux distributions de probabilités, celles des probabilités prédites et la distribution réelle des données.

Pour une classification binaire, la formule de la “BCE” est :

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (16)$$

avec y la vérité terrain et $\hat{y}$ est la prédiction pour tous les échantillons N .

La BCE est souvent couplée à la fonction d'activation sigmoïde, son objectif étant de permettre au modèle de différencier deux classes en minimisant la dissimilarité entre les prédictions et la vérité terrain.

Dans certains scénarios, la tâche de classification implique la distinction de plus de deux classes, c'est le domaine de la classification multiclass, comme dans le défi ImageNet Deng *et al.* [2009]. Pour palier cette différence, il est nécessaire de changer de fonction de pertes, car la perte binaire n'est plus appropriée. La fonction de perte utilisée est alors “CCE”, Li *et al.* [2022] l'utilise pour améliorer la robustesse des réseaux.

Rétropropagation

L'apprentissage des réseaux de neurones repose sur la rétropropagation du gradient (*backpropagation*), un algorithme permettant de calculer efficacement les gradients de la fonction de perte par rapport aux paramètres du réseau (poids et biais), en appliquant la règle de dérivation en chaîne. Ces gradients sont ensuite exploités par une méthode d'optimisation, comme la descente de gradient, afin d'ajuster les paramètres.

Le processus d'apprentissage commence par une phase de propagation avant (*forward pass*) au cours de laquelle les données d'entrée traversent les couches du réseau pour produire une prédiction. Une fonction de perte est ensuite évaluée en comparant cette prédiction à la vérité terrain.

L'erreur obtenue est ensuite propagée en sens inverse dans le réseau lors de la phase de rétropropagation (*backward pass*). À chaque couche, le gradient de la fonction de perte par rapport aux paramètres est calculé à l'aide de la règle de la chaîne, qui exprime le gradient total comme un produit de dérivées partielles successives. Par exemple, pour un poids w_i :

$$\frac{\partial \mathcal{L}}{\partial w_i} = \frac{\partial \mathcal{L}}{\partial a_i} \cdot \frac{\partial a_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial w_i} \quad (17)$$

où z_i désigne l'entrée de la fonction d'activation, a_i sa sortie, et $\mathcal{L}$ la fonction de perte.

Une fois les gradients calculés, les paramètres sont mis à jour par descente de gradient selon la règle suivante :

$$x_{n+1} = x_n - \eta \cdot \nabla f(x_n) \quad (18)$$

où η désigne le taux d'apprentissage (*learning rate*), $\nabla \mathcal{L}(x_n)$ le gradient de la fonction de perte $\mathcal{L}$ par rapport aux paramètres courants x_n , et x_{n+1} les paramètres mis à jour.

Ce mécanisme ajuste les poids du réseau afin de réduire progressivement l'erreur entre les prédictions et les valeurs attendues, ce qui améliore la capacité de généralisation. Grâce à sa formulation récursive, le calcul de rétropropagation est particulièrement efficace dans les architectures en couches et constitue encore aujourd'hui le fondement de l'apprentissage supervisé des réseaux de neurones profonds.

Métriques

Les techniques de DL permettent de traiter diverses tâches en vision par ordinateur, telles que la classification d'images, la segmentation ou la détection d'objets. Les métriques jouent un rôle central dans l'évaluation des performances, en fournissant une mesure explicite de la fonction à optimiser durant l'entraînement. Le choix des métriques doit donc être adapté au contexte, afin de mieux interpréter le comportement du modèle et guider les ajustements nécessaires.

Les contre-performances peuvent résulter de plusieurs facteurs : architecture inadéquate, volume de données insuffisant, déséquilibre entre classes, stratégie de prétraitement inappropriée ou métrique mal définie. Une analyse rigoureuse de ces aspects permet d'identifier des pistes d'amélioration, qu'il s'agisse d'ajuster les hyperparamètres, de revoir la stratégie d'apprentissage ou d'introduire des techniques de régularisation.

Terminologie de l'évaluation des prédictions

Pour évaluer les performances d'un modèle de classification binaire, quatre notions fondamentales comparent les prédictions à la vérité terrain : les vrais positifs, vrais négatifs, faux positifs et faux négatifs.

Un **vrai positif** (TP) désigne une instance correctement identifiée comme appartenant à la classe positive. Un **faux positif** (FP) correspond à une instance incorrectement prédite comme positive alors qu'elle est négative.

De manière symétrique, un **vrai négatif** (*True Negative* (TN)) est une instance correctement classée comme négative, tandis qu'un **faux négatif** (FN) désigne une instance positive classée à tort comme négative.

Ces quatre catégories servent de fondement à la construction des métriques d'évaluation telles que la précision, le rappel ou le F1-score, permettant de quantifier la capacité d'un modèle à distinguer correctement les classes. Elles seront réutilisées ultérieurement pour l'évaluation des modèles appliqués aux tâches de détection d'objets.

Matrice de confusion

La matrice de confusion est une représentation concise des prédictions du modèle, par rapport à la vérité terrain. Cette matrice fournit une répartition claire des résultats pour les différentes classes, permettant d'évaluer les performances du modèle.

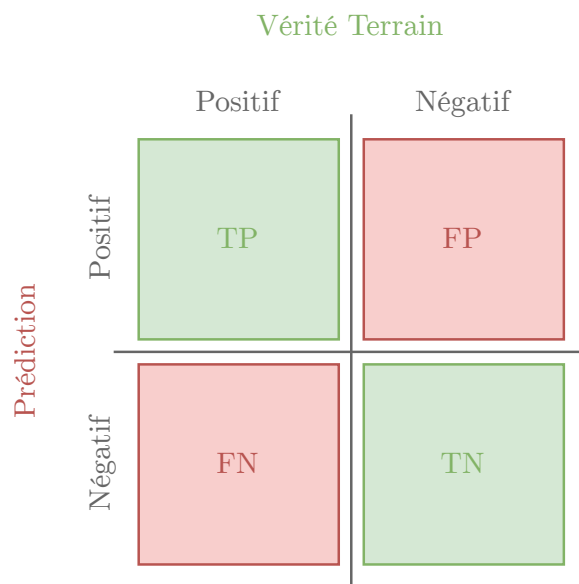


FIGURE 29 – Schéma exemple d'une matrice de confusion

La matrice de confusion est un outil d'évaluation qui détaille les différentes erreurs (FN et FP) des résultats de performance relativement complexes, dans un format facilement compréhensible.

Les éléments de la matrice permettent de calculer des diverses mesures en traitement d'images, que nous allons étudier dans les prochaines parties.

Exactitude, précision, rappel, F1 score, spécificité

En général, il est possible d'obtenir les mesures d'évaluation quantitatives suivantes à partir de la matrice de confusion des classes binaires :

- L'exactitude ("accuracy") : le nombre d'échantillons correctement classés parmi tous les échantillons présents dans l'ensemble de test :

$$\text{exactitude} = \frac{TP + TN}{TP + FP + TN + FN}. \quad (19)$$

L'exactitude est une métrique, reflétant la proportion de vraies détections positives par rapport à toutes les prédictions positives faites par le modèle. Elle indique le degré d'exactitude avec lequel le modèle classifie, tout en évitant les FP. Une exactitude élevée, signifie un faible taux de FP.

- La précision ("precision") : le nombre d'échantillons appartenant effectivement à la classe positive sur l'ensemble des échantillons prédits comme appartenant à la classe positive par le modèle :

$$\text{précision} = \frac{TP}{TP + FP}. \quad (20)$$

- Le rappel ("recall" ou sensibilité) : le nombre d'échantillons prédits correctement comme appartenant à la classe positive sur l'ensemble des échantillons, qui appartiennent effectivement à la classe positive :

$$\text{rappel} = \frac{TP}{TP + FN}. \quad (21)$$

Le rappel démontre la capacité de l'algorithme à détecter la plupart des vrais cas positifs et à ne pas les manquer.

- Le F1 score : la moyenne harmonique des scores de précision et de rappel :

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (22)$$

- La spécificité : le nombre d'échantillons prédits correctement comme appartenant à la classe négative sur l'ensemble des échantillons de l'ensemble de données, qui appartiennent effectivement à la classe négative :

$$\text{spécificité} = \frac{TN}{TN + FP}. \quad (23)$$

Receiver Operating Characteristic (ROC)

Une courbe Receiver Operating Characteristic (ROC) correspond à l'application qui relie le "taux de vrais positifs" (*True Positive Rate (TPR)*) au "taux de faux positifs" (*False Positive Rate (FPR)*). Les différents points de cette courbe sont calculés en variant un seuil avec la confiance. Les courbes ROC sont généralement définies pour un modèle de classification binaire, bien qu'elles puissent être étendues à un environnement multi-classes. Elles sont utilisées pour comparer des architectures et pour calculer l'AUC.

La définition du "True Positive Rate (TPR)" coïncide exactement avec le paramètre rappel (équation 21). La formule de calcul du TPR est donc simple :

$$TPR = \frac{TP}{TP + FN} = \text{rappel} \quad (24)$$

Le “False Positive Rate (FPR)” est défini comme le nombre d’échantillons de la classe négative prédits à tort comme appartenant à la classe positive (c’est-à-dire les faux positifs), sur l’ensemble des échantillons de l’ensemble de données appartenant en réalité à la classe négative. La formule associée est la suivante :

$$\begin{aligned} FPR &= \frac{FP}{FP + TN} \\ &= 1 - \frac{TN}{FP + TN} \\ &= 1 - \text{spécificité} \end{aligned} \quad (25)$$

Il convient de noter que, mathématiquement, le FPR est égal à l’opposé de la spécificité (équation (23)) + 1.

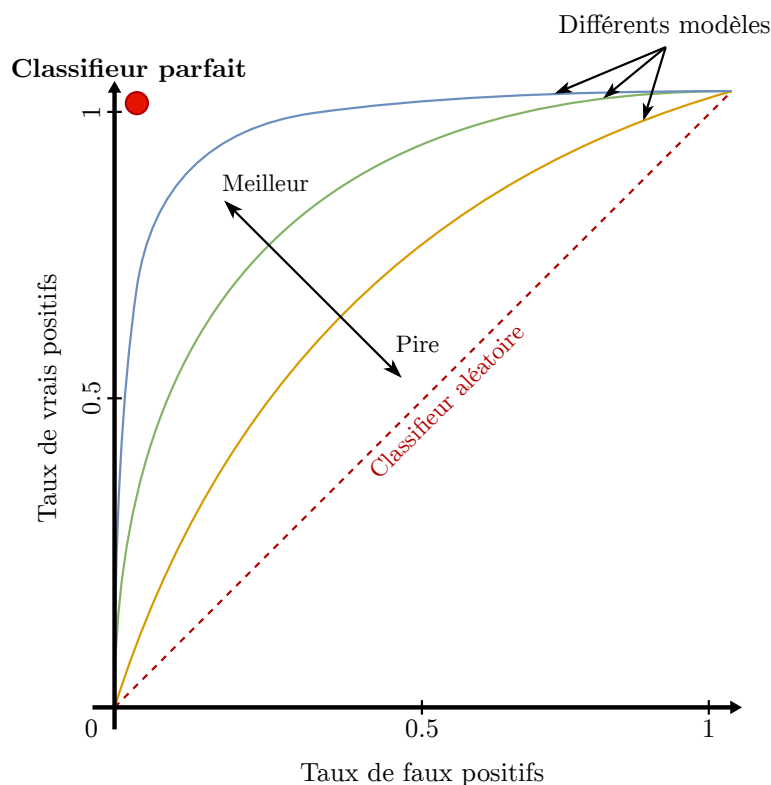


FIGURE 30 – Illustration d’une courbe ROC en comparaison à un classifieur aléatoire. Inspiré d’une image du site medium.com

Courbe précision-rappel (PR)

La courbe PR est une représentation graphique, donnant des indications précieuses sur les performances des modèles de classification et de détection, en particulier dans les scénarios où il y a un déséquilibre entre les classes ou lorsque l’accent est mis sur les instances positives. La courbe PR illustre le compromis entre la précision et le rappel, à différents seuils de décision, mettant en lumière la capacité d’un algorithme à classer correctement (TP) tout en minimisant les erreurs (FP).

Pour construire une courbe PR, il faut faire varier le seuil de décision du modèle de classification. Quand le seuil est bas, un plus grand nombre de prédictions est effectué, augmentant

le rappel mais peut entraîner une diminution de la précision. Inversement, quand le seuil est élevé, le nombre de prédictions est moindre, pouvant augmenter la “precision” au prix d’un rappel plus faible. Voici un exemple de courbe PR.

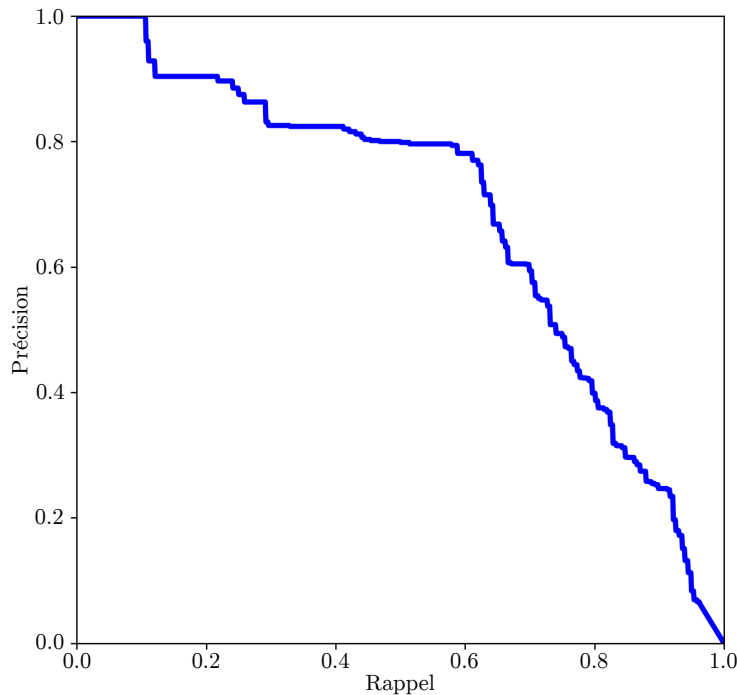


FIGURE 31 – Exemple d’une courbe PR

Une courbe PR bien formée maintient des valeurs de précision élevées sur une large plage de rappels, ce qui indique que le modèle parvient à détecter efficacement de nombreuses instances positives, tout en limitant le nombre de FP. L’AUC, l’aire sous la courbe PR est une mesure couramment utilisée pour quantifier les performances globales, en fonction de différents seuils de décision. Une AUC plus élevée, implique que le modèle est plus performant, avec un bon équilibre entre la précision et le rappel.

Formellement, l’AUC pour la courbe PR se calcule de la manière suivante :

$$\text{AUC-PR} = \int_0^1 \text{précision}(\text{rappel}) \, d(\text{rappel}). \quad (26)$$

Intersection over Union (IoU)

Afin de pouvoir calculer l’IoU, il est nécessaire de définir ces opérations mathématiques faisant partie de la théorie des ensembles. Les définitions se basent sur la relation entre plusieurs ensembles et pour simplifier la compréhension, voici les deux ensembles qui vont être utilisés par la suite pour les illustrer.

Intersection

L’intersection entre deux ensembles, désignée par $A \cap B$, est un ensemble contenant tous les éléments communs aux ensembles A et B .

$$A \cap B = \{x \mid x \in A \text{ et } x \in B\} \quad (27)$$



FIGURE 32 – Illustration de deux ensembles utilisés pour les définitions de l’IoU

Cela peut être applicable avec n ensembles, notés $A_1, \dots, A_n$, où la formule est :

$$\begin{aligned} \bigcap_{i=1}^n A_i &= A_1 \cap \dots \cap A_n \\ &= \{x \mid x \in A_1 \text{ et } \dots \text{ et } x \in A_n\} \end{aligned} \quad (28)$$

En reprenant les ensembles de la figure 32, l’intersection obtenue est illustrée en vert dans la figure suivante :

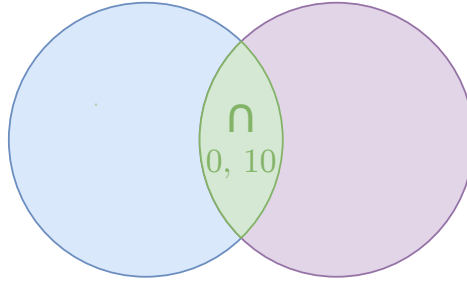


FIGURE 33 – Illustration de l’intersection avec les ensembles A et B

Union

L’union entre deux ensembles, $A \cup B$, est l’ensemble des éléments des deux ensembles A et B sans duplication.

$$A \cup B = \{x \mid x \in A \text{ ou } x \in B\} \quad (29)$$

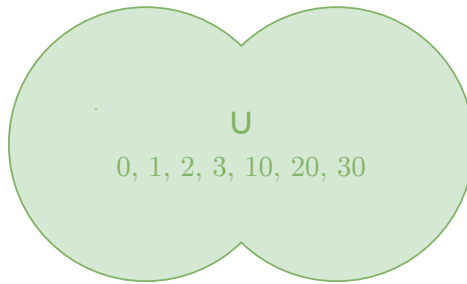
Tout comme l’intersection, l’union peut s’étendre à n ensembles :

$$\begin{aligned} \bigcup_{i=1}^n A_i &= A_1 \cup \dots \cup A_n \\ &= \{x \mid x \in A_1 \text{ ou } \dots \text{ ou } x \in A_n\} \end{aligned} \quad (30)$$

En reprenant les ensembles de la figure 32, l’union obtenue est illustrée en vert dans la figure suivante :

Formule et exemple

Dans le domaine de la classification, l’IoU se calcule seulement entre deux régions, une étant la prédiction, l’autre étant la Vérité Terrain (VT). L’IoU est alors calculée en divisant l’intersection des aires des deux régions par l’union de celles-ci. La formule en est la suivante :


 FIGURE 34 – Illustration de l'union avec les ensembles A et B

$$\text{IoU} = \frac{\text{Aire de l'Intersection}}{\text{Aire de l'Union}} \quad (31)$$

L'IoU est une valeur comprise entre 0 et 1, la valeur 0 signifiant qu'il n'y a aucune intersection entre les ensembles alors que la valeur 1 indique que les ensembles sont identiques. Cette mesure est souvent utilisée pour comparer les prédictions du modèle à la vérité terrain, pour évaluer la qualité des résultats obtenus.

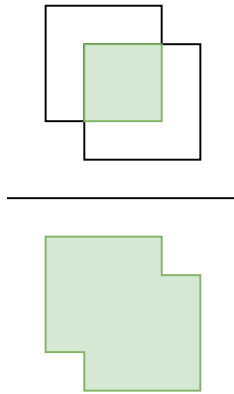
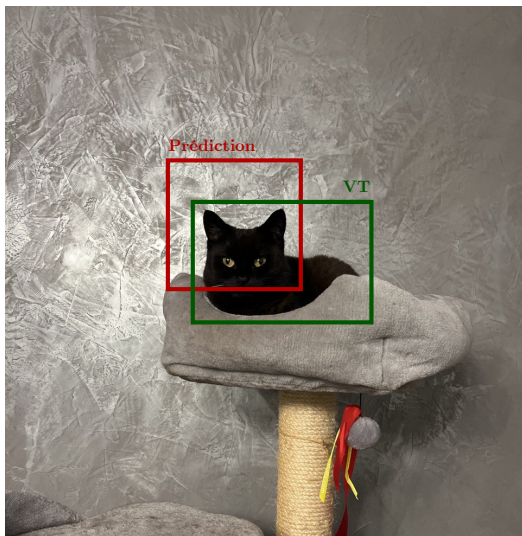
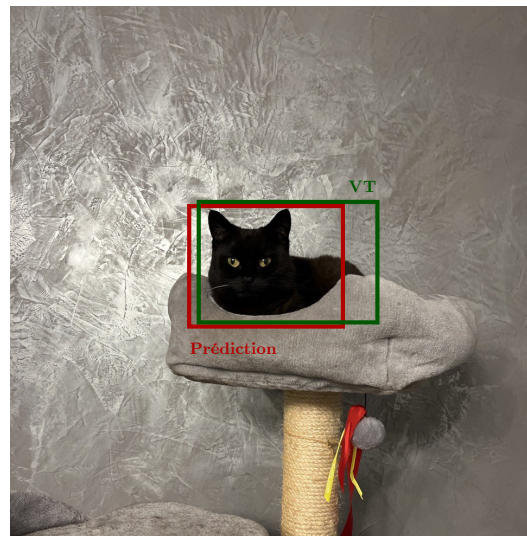

 FIGURE 35 – Illustration de l'IoU avec les aires qui sont utilisées en *vert*

 (a) Exemple d'un False Positive (FP) avec un $\text{IoU} \simeq 0.33$

 (b) Exemple d'un True Positive (TP) avec un $\text{IoU} \simeq 0.73$

FIGURE 36 – Exemples de résultats lorsque le seuil de l'IoU est fixé à 0.5

La figure 36 ci-dessus est un exemple illustrant les résultats, en fonction de leur IoU et l'importance du seuil, fixé dans cet exemple à 0.5. La prédiction dans la figure 36a a un IoU d'environ 0.3 par rapport à la VT et est considérée comme un FP. En revanche, la prédiction de la figure 36b est classée comme TP parce qu'elle possède un IoU d'environ 0.8 avec la VT.

Average Precision (AP) / Mean Average Precision (mAP)

La précision moyenne, *Average Precision (AP)*, correspond à l'aire sous la courbe PR, c'est-à-dire la moyenne pondérée des précisions à différents niveaux de rappel. La pondération reflète l'augmentation du rappel par rapport au seuil précédent. La mAP est ensuite obtenue en moyennant l'AP calculée pour chaque classe du jeu de données.

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^N \text{AP}_i \quad (32)$$

avec AP_i l'AP de la classe i et N le nombre de classes total.

Pour calculer l'AP, il convient de suivre les étapes suivantes :

1. pour chaque BBOX, obtenir les scores de prédiction pour chaque classe ;
2. calculer la précision et le rappel pour différents seuils de confiance, pour chaque classe ;
3. calculer l'AUC de la courbe PR, donnant ainsi l'AP (voir figure 37).

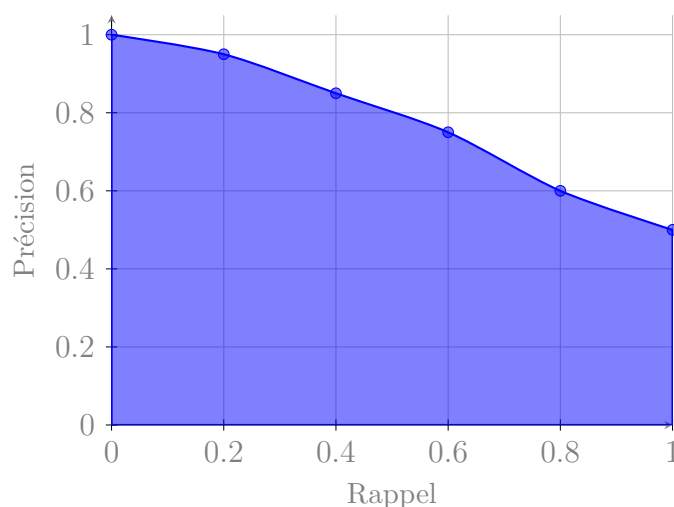


FIGURE 37 – Illustration de la courbe Précision - Rappel (PR) et de l'aire sous la courbe correspondant à l'AP.

Dans certains contextes, la définition de l'AP peut varier. Par exemple, dans le protocole d'évaluation de COCO [Lin *et al.*, 2014], l'AP correspond à la moyenne des AP mesurées à plusieurs seuils d'IoU, ce qui est assimilé à leur mAP interne.

La mAP reflète le compromis entre précision et rappel, tout en prenant en compte à la fois les FP et les FN. Cette caractéristique en fait une métrique robuste et adaptée à la majorité des tâches de détection.

Résumé

Dans cette thèse appliquée au domaine de l'imagerie sonar, nous étudions la problématique de la détection automatique d'objets rares à partir de données visuelles sonar acquises en milieu sous-marin. L'objectif des travaux proposés est d'améliorer la précision des systèmes de détection basés apprentissage dans un contexte où les données réelles sont à la fois peu nombreuses, coûteuses à annoter et fortement variables.

Pour ce faire, nous nous intéressons d'abord à l'intégration de stratégies d'apprentissage actif dans le cadre de la détection d'objets. Nous étudions notamment une méthode combinant des mesures d'incertitude avec des techniques de regroupement (clustering) afin de sélectionner de manière pertinente un sous-ensemble réduit mais informatif d'images à annoter. Cette approche vise à optimiser le processus d'annotation en maximisant l'utilité des données.

Nous explorons ensuite, en perspective, l'utilisation d'images sonar simulées, plus faciles à produire et à contrôler que les images réelles. Dans cette optique, nous proposons l'usage de modèles génératifs, tels que les réseaux antagonistes génératifs (GAN), pour rapprocher les représentations visuelles simulées des caractéristiques du domaine réel, tout en préservant les classes et localisation des objets d'intérêt. Cette stratégie permettrait d'exploiter efficacement des données synthétiques dans l'entraînement de détecteurs, malgré le décalage initial entre domaines.

Les résultats expérimentaux obtenus mettent en évidence les difficultés liées au traitement de l'imagerie sonar, notamment en présence de variations inter-domaines marquées. Néanmoins, les approches proposées permettent d'améliorer les performances de détection tout en réduisant la dépendance à un grand volume d'annotations manuelles.

Mots-clés : *Détection d'objets · imagerie sonar · apprentissage actif · apprentissage profond · réseaux de neurones convolutifs · modèles génératifs · adaptation de domaine*

Abstract

In this thesis, applied to the field of sonar imaging, we address the problem of automatic detection of rare objects from sonar visual data acquired in underwater environments. The objective of the proposed work is to improve the accuracy of learning-based detection systems in a context where real data is both scarce, costly to annotate, and highly variable.

To this end, we first focus on the integration of active learning strategies within the object detection framework. In particular, we investigate a method that combines uncertainty-based measures with clustering techniques to select a relevant and informative subset of images for annotation. This approach aims to optimize the annotation process by maximizing the utility of the selected data.

We then explore, as a perspective, the use of simulated sonar images, which are easier to generate and control than real ones. In this context, we propose the use of generative models—such as Generative Adversarial Networks (GANs)—to bring simulated visual representations closer to the characteristics of the real domain, while preserving the classes and locations of the objects of interest. This strategy would enable the efficient exploitation of synthetic data for training detectors, despite the initial domain gap.

The experimental results highlight the challenges associated with sonar image processing, particularly in the presence of strong inter-domain variations. Nevertheless, the proposed approaches lead to improved detection performance while reducing the dependency on large volumes of annotated data.

Keywords: *Object detection · sonar imaging · rare data · active learning · deep learning · convolutional neural networks (CNN) · generative models (GAN) · domain adaptation*