UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS

APPROCHES D'APPRENTISSAGE PROFOND POUR LA DÉTECTION ET LA SEGMENTATION DES DÉFAUTS VISUELS DES PONTS EN BÉTON

THÈSE
PRÉSENTÉE
COMME EXIGENCE PARTIELLE
DU DOCTORAT EN SCIENCES ET TECHNOLOGIES DE L'INFORMATION

PAR

DARIUSH AMIRKHANI

SEPTEMBRE 2024

Cette thèse a été évaluée par un jury composé des personnes suivantes :

Prof. Shamsodin Taheri ..... Président du comité

Dr. Abdelhamid Mammeri ..... Membre externe du comité

Dr. Etienne St-Onge ..... Membre du comité

Prof. Mohand Saïd Allili ..... Directeur de recherche

Dr. Jean-François Lapointe ..... Codirecteur de recherche

Thèse acceptée le : 13 Septembre 2024

Dédicace

Remerciements

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations, sigles et acronymes

CL CLASSIFICATION

DT DETECTION

SG SEGMENTATION

SSG Semantic SEGMENTATION

ISG Instance SEGMENTATION

DL DEEP LEARNING

TL TRANSFER LEARNING

SM SEMI-SUPERVISED

WS WEAKLY-SUPERVISED

AD ADVERSARIAL LEARNING

GAN GENERATIVE ADVERSARIAL NETWORK

AT ATTENTION MECHANISM

SF SELF-COLLECTED DATASET

TI TOTAL IMAGES OF DATASET

MD MULTI DEFECT

LM Leung-Malik Filter Bank

UAV Unmanned Aerial Vehicles

Résumé

Cette thèse explore l'application des techniques d'apprentissage profond dans le domaine de la surveillance de l'état des structures, en se concentrant particulièrement sur la détection et la segmentation précises des défauts dans les ponts en béton. Motivé par la nécessité cruciale d'assurer la sécurité et l'intégrité des infrastructures de transport, ce travail présente deux contributions révolutionnaires : la première contribution propose une revue systématique et une application des modèles d'apprentissage profond pour la classification et la détection des défauts des ponts en béton, tandis que la deuxième contribution introduit CrackSight, une nouvelle architecture U-net qui intègre parfaitement les mécanismes d'attention avec la détection et la classification pour une segmentation précise des fissures dans des environnements complexes.

La première partie de cette recherche explore l'application de l'apprentissage profond pour la classification visuelle et la détection des défauts des ponts en béton. À travers un examen minutieux des méthodologies actuelles, cette étude évalue l'efficacité de diverses architectures d'apprentissage profond, soulignant leur potentiel à identifier les défauts des bétons, à surmonter les défis posés par les conditions d'éclairage variables et à différencier les textures nuancées dans des arrière-plans complexes. Les expériences complètes menées sur divers ensembles de données et scénarios réels ont démontré une amélioration notable de la précision de détection des fissures, faisant ainsi progresser significativement l'état de l'art dans ce domaine.

La deuxième partie de cette thèse introduit CrackSight, une méthodologie innovante qui améliore de manière significative la détection et la segmentation des fissures. CrackSight intègre un mécanisme de mise au point linéaire à double attention (DALFM) et une couche de saillance dans le modèle Détection de Fissures Basée sur les Caractéristiques Linéaires (LFB-Net), utilisant la convolution à trous pour traiter les motifs de fissures subtils et complexes. De plus, la perte d'entropie croisée binaire pondérée contextuelle est introduite pour traiter dynamiquement le déséquilibre des classes et intégrer des

informations contextuelles globales, améliorant ainsi les performances de segmentation. Nos expériences ont démontré la robustesse de CrackSight face à plusieurs complexités des arrière-plans et des conditions d'acquisition d'images ainsi qu'à des scénarios pratiques, établissant ainsi une nouvelle norme dans la segmentation des fissures.

En résumé, cette thèse non seulement démontre l'efficacité de l'apprentissage profond pour améliorer les pratiques de surveillance de l'état des structures, mais elle met également en évidence le potentiel de la combinaison des mécanismes d'attention avec les stratégies de détection et de classification pour améliorer la précision de la segmentation. Grâce à ce travail, une contribution significative est apportée au domaine de la surveillance de l'état des structures, offrant de nouvelles perspectives et des approches pratiques pour la détection précoce et l'intervention sur les défauts structurels.

Abstract

This thesis explores the application of deep learning techniques in the domain of structural health monitoring, particularly focusing on the accurate detection and segmentation of defects in concrete bridges. Motivated by the crucial need for ensuring the safety and integrity of transportation infrastructures, this work introduces two groundbreaking contributions: the first contribution presents a systematic review and application of deep learning models for the classification and detection of concrete bridge defects, while the second contribution consists CrackSight, a novel U-net architecture that seamlessly integrates attention mechanisms with detection, and classification for precise crack segmentation in complex backgrounds.

The initial part of this research explores the application of deep learning for visual concrete bridge defect classification and detection. Through a meticulous examination of current methodologies, this study benchmarks the effectiveness of various deep learning architectures, highlighting their potential in identifying concrete defects, overcoming challenges posed by varying lighting conditions, and differentiating nuanced textures across complex backgrounds. The comprehensive experiments conducted across diverse datasets and real-world scenarios demonstrated a marked improvement in crack detection accuracy, significantly advancing the state-of-the-art in this field.

The second part of this thesis introduces CrackSight, an innovative methodology that significantly enhances crack detection and segmentation. CrackSight integrates a dual-attention linear focus mechanism (DALFM) and a saliency layer within the Linear Feature-based Crack Detection (LFB-Net) model, employing atrous convolution to handle subtle and complex crack patterns. Additionally, the Contextual Weighted Binary Cross-Entropy Loss is introduced to dynamically address class imbalance and incorporate global contextual information, thereby improving segmentation performance. Our experiment demonstrated the robustness of CrackSight against several complexities of

the background and image acquisition conditions and practical scenarios, setting a new standard in crack segmentation.

In summary, this thesis not only substantiates the efficacy of deep learning in enhancing structural health monitoring practices but also it highlights the potential of combining attention mechanisms with detection and classification strategies to enhance segmentation accuracy. Through this work, a significant contribution is made to the field of structural health monitoring, offering novel insights and practical approaches for the early detection and intervention of structural defects.

Chapitre 1

Introduction

1.1 Maintenance des infrastructures de ponts

La maintenance et la sécurité des infrastructures de ponts sont cruciales pour la durabilité des réseaux de transport urbains et ruraux [95]. Les ponts, en tant que composants vitaux de ces réseaux, sont continuellement exposés à des facteurs de stress environnementaux tels que les conditions météorologiques, les charges de trafic et les catastrophes naturelles. Ces facteurs peuvent induire diverses formes de dommages structurels au fil du temps, pouvant entraîner une dégradation et une défaillance [148]. Les échecs historiques des ponts, comme l'effondrement du pont Morandi en Italie [23] et du pont I-35W à Minneapolis [141], servent de rappels frappants des conséquences potentielles de la négligence de la maintenance des ponts, allant des pertes économiques aux tragédies humaines.

Une maintenance efficace des ponts assure non seulement la sécurité et la fiabilité de ces structures, mais prolonge également leur durée de vie, offrant ainsi des avantages économiques et sociaux à long terme. En mettant en œuvre des programmes systématiques d'inspection et de maintenance, les autorités de transport peuvent identifier et atténuer les déficiences structurelles dès le début, empêchant les problèmes mineurs de se transformer en défaillances majeures.

1.2 De l'inspection manuelle aux systèmes d'inspection automatisés

Les méthodes traditionnelles d'inspection des ponts reposent principalement sur des examens visuels effectués par des inspecteurs humains. Ces inspecteurs évaluent l'état d'un pont à travers une observation visuelle minutieuse et des tests physiques [144, 109]. Cette approche pratique, fondée sur l'expertise et le jugement des techniciens et ingénieurs expérimentés, constitue le pilier de la surveillance de l'état des structures depuis des décennies. Malgré son efficacité prouvée dans certains contextes, cette méthode présente des défis tels que la nature subjective des évaluations, les risques potentiels pour la sécurité des inspecteurs, et les complexités logistiques liées à l'accès et à l'examen de toutes les parties des structures de ponts étendues.

Les limitations des inspections manuelles ont conduit au développement de systèmes d'inspection automatisés, tirant parti des avancées en matière de technologie de détection et de traitement des données. Des technologies telles que le radar à pénétration de sol, le balayage laser et la photogrammétrie numérique ont amélioré la capacité de détecter et de quantifier les défauts avec une précision accrue et moins de risques pour les humains. De plus, l'intégration de véhicules aériens sans pilote (UAV) et de plateformes robotiques dans les flux de travail d'inspection représente une avancée significative [33]. Ces technologies permettent des inspections complètes et efficaces des structures de ponts grandes et complexes, réduisant le besoin de perturber la circulation et améliorant la sécurité du processus d'inspection.

1.3 Utilisation des drones pour des inspections visuelles améliorées dans la surveillance des ponts

L'avènement des véhicules aériens sans pilote (UAV), communément appelés drones, a considérablement transformé le paysage de la surveillance de l'état des structures, en particulier dans l'inspection des ponts. Les inspections traditionnelles des ponts, qui reposent fortement sur des évaluations visuelles effectuées par des inspecteurs humains, rencontrent souvent de nombreux défis tels que l'accès limité aux zones critiques, les risques pour la sécurité et les complexités logistiques associées à l'utilisation d'équipements tels que les camions inspecteurs. Ces inspections sont non seulement

exigeantes en main-d'œuvre, mais également chronophages et coûteuses, en particulier lorsqu'il s'agit de gérer les perturbations de la circulation et d'assurer la sécurité du personnel impliqué dans le processus d'inspection.

L'intégration des drones dans le flux de travail d'inspection permet de surmonter bon nombre de ces défis en permettant une inspection visuelle à distance des ponts. Les drones sont capables d'accéder à des zones difficiles d'accès, telles que le dessous des tabliers de ponts ou les espaces entre les poutres, sans avoir besoin de déploiements complexes ou de la présence d'inspecteurs dans des lieux potentiellement dangereux. Cette capacité réduit considérablement les risques associés aux méthodes d'inspection traditionnelles, où les inspecteurs pourraient devoir travailler à de grandes hauteurs ou dans des espaces confinés.

Cependant, malgré ces avantages, l'utilisation des drones introduit également des défis spécifiques qui doivent être relevés pour maximiser leur efficacité dans les inspections visuelles. Un défi notable est l'incapacité des drones à s'approcher suffisamment de certaines parties d'un pont en raison des complexités structurelles ou des obstacles. Par exemple, les ponts aux conceptions complexes ou ceux situés dans des zones densément peuplées avec des infrastructures environnantes peuvent limiter la capacité du drone à capturer des images haute résolution à courte distance. Dans de tels scénarios, les drones peuvent ne prendre que des images à distance, ce qui peut réduire la clarté et les détails nécessaires pour une détection précise des défauts.

Cette limitation pose un défi significatif pour les modèles de vision par ordinateur existants, tels que les réseaux de neurones convolutifs (CNN) et les architectures U-net, qui dépendent d'images de haute qualité et de gros plans pour détecter et classer efficacement les défauts structurels. Lorsque les drones sont obligés de capturer des images à distance, la résolution réduite peut entraîner une diminution de la précision de détection, avec un risque de manquer des défauts subtils comme de petites fissures ou des signes précoces de corrosion. Ce défi souligne la nécessité de nouvelles avancées dans les techniques de traitement d'images et les modèles d'intelligence artificielle capables de traiter des images de résolution inférieure sans compromettre la précision de la détection des défauts.

De plus, les facteurs environnementaux tels que le vent, la pluie ou les mauvaises conditions d'éclairage peuvent également affecter la qualité des images capturées par les drones, compliquant davantage le processus d'inspection. Ces facteurs peuvent entraîner un flou de mouvement ou masquer des zones critiques du pont, rendant difficile

pour les algorithmes d'IA d'évaluer avec précision l'état de la structure. Pour atténuer ces problèmes, les développements futurs de la technologie des drones et des systèmes d'inspection pilotés par l'IA devraient se concentrer sur l'amélioration de la stabilisation d'image, le renforcement des performances en basse lumière, et le développement d'algorithmes capables de compenser la perte de détails dans des conditions difficiles.

Les récentes avancées en intelligence artificielle (AI) et en réalité augmentée (AR) ont montré des promesses pour relever certains de ces défis. Par exemple, comme détaillé dans les recherches récentes de Lapointe et al. [107], un système basé sur l'IA et l'AR peut être employé où des drones équipés de caméras transmettent des flux vidéo en direct vers des stations au sol. Ces flux sont ensuite traités en temps réel pour détecter d'éventuels défauts, et les informations résultantes sont superposées à la vue de l'inspecteur via des casques AR. Ce système permet aux inspecteurs de recevoir immédiatement des données contextuelles sur l'état du pont, améliorant leur capacité à prendre des décisions éclairées sur place, même lorsqu'ils doivent composer avec une qualité d'image moins que parfaite.

L'adoption de la technologie des drones dans les inspections de ponts marque un tournant vers des systèmes de surveillance plus automatisés et intelligents. Alors que la technologie continue d'évoluer, la combinaison des UAV avec l'IA et l'AR promet de rationaliser encore davantage les processus d'inspection, de réduire les coûts, et surtout, d'améliorer la sécurité et la fiabilité des infrastructures critiques, tout en relevant les défis uniques qui se posent en cours de route.

1.4 Apprentissage profond pour la surveillance de l'état des structures

Ces dernières années, les technologies d'apprentissage profond ont entraîné un changement significatif dans la maintenance des ponts et l'évaluation de leur sécurité, en s'appuyant sur leur capacité à déchiffrer des motifs complexes dans des ensembles de données étendus pour détecter et évaluer les défauts du béton avec une précision sans précédent. En employant une gamme de techniques allant des réseaux de neurones convolutifs (CNN) [177] aux réseaux de neurones récurrents (RNN) et aux modèles de transformateurs, ces algorithmes avancés excellent à identifier les signatures de défauts nuancées qui varient selon les éléments structurels. Ces technologies répondent au prob-

lème critique de la détection, de la classification et de la segmentation efficaces des diverses formes de défauts du béton, essentielles pour maintenir l'intégrité et la sécurité des infrastructures de ponts. La classification permet d'identifier le type de défaut présent, la détection localise le défaut dans une boîte englobante, et la segmentation délimite le défaut au niveau du pixel, chaque étape étant cruciale pour une évaluation précise et une intervention en temps opportun sur les défauts potentiels du béton.

Les premières recherches dans ce domaine se sont concentrées sur la classification de défauts uniques [2, 179, 225]. D'autres méthodes ont abordé la classification multi-classes des défauts en supposant une instance de défaut par image [21, 83, 235, 240]. Par la suite, des méthodes ont proposé des modèles ciblant la classification multi-étiquettes des défauts dans les images [15, 16]. Les méthodes de détection des défauts du béton, quant à elles, visaient à localiser les défauts soit en utilisant des boîtes englobantes (BB) soit par segmentation. La détection des défauts basée sur BB comprend des méthodes en deux étapes et en une étape. La détection en deux étapes comprend une première étape pour générer des propositions de régions, et une deuxième étape pour affiner les propositions les plus prometteuses pour la classification et la localisation des défauts. Cette technique donne généralement une précision élevée, mais elle entraîne un coût computationnel élevé. La détection en une seule étape intègre la proposition de région, la classification et la localisation en un seul pipeline, ce qui réduit considérablement les exigences en temps d'exécution [37, 35, 88]. Enfin, la segmentation des défauts tente de localiser les défauts au niveau du pixel, ce qui permet d'obtenir des informations telles que les limites des défauts, leur superficie et leur emplacement [87].

La plupart des méthodes proposées dans la littérature ont été conçues pour traiter des défauts spécifiques (par exemple, les fissures), où une bonne précision a été obtenue sur des ensembles de données bien préparés. Cependant, plusieurs défis peuvent entraîner une mauvaise généralisation sur des ensembles de données étendus, qui peuvent être résumés comme suit :

1. **Non-homogénéité des surfaces en béton :** La texture de surface du béton n'a pas une distribution unimodale. Différents matériaux bruts, finitions et exposition aux éléments naturels peuvent donner une variété de couleurs et de textures de béton [138] (voir Fig. 1.2.(a)). De plus, les surfaces en béton peuvent contenir des peintures, de petits trous, des graffitis et des marquages, ce qui augmente encore cette variabilité.

2. **Variabilité des classes de défauts du béton :** Les défauts peuvent avoir différentes tailles, intensités et emplacements dans l'image, ce qui complique leur classification et détection. En outre, une même instance de défaut peut être fragmentée en différentes parties (c'est-à-dire fragmentation du défaut), tandis que plusieurs instances de défauts de classes différentes peuvent se produire au même endroit (c'est-à-dire chevauchement des défauts). Comme il sera présenté dans la section suivante, certaines classes de défauts sont fortement corrélées, ce qui peut expliquer leur chevauchement fréquent (voir Fig.1.2.(b)).

3. **Conditions d'acquisition des images de défauts** : Selon sa taille et son emplacement, un défaut peut être capturé à différentes résolutions d'image, rapports d'aspect et distances par rapport à la caméra. Ces facteurs déterminent le niveau de détail avec lequel les défauts peuvent être observés. Les défauts capturés à grande distance peuvent être petits ou même imperceptibles par rapport aux éléments du pont, tandis que les défauts acquis à très courte distance peuvent manquer certaines parties importantes. Enfin, les changements de points de vue, de conditions météorologiques, d'éclairage ou de méthodologies/dispositifs d'acquisition d'images peuvent augmenter la variabilité de l'apparence du défaut (voir Fig. 1.2.(c)).

4. **Annotation des défauts du béton** : La qualité de l'annotation peut être affectée par plusieurs facteurs. Les défauts peuvent se produire simultanément au même endroit de l'image (c'est-à-dire chevauchement des défauts). De plus, lorsqu'une instance de défaut est discontinue, elle peut être annotée comme plusieurs instances (c'est-à-dire fragmentation du défaut). Cela peut induire une énorme variabilité d'annotation intra- et inter-observateurs, ce qui peut envoyer des signaux incohérents lors de l'entraînement du modèle (voir Fig. 1.2.(d) pour l'illustration).

En réponse aux défis de l'analyse des défauts du béton, les chercheurs ont introduit plusieurs ensembles de données publics annotés. Ceux-ci vont de la concentration sur les défauts uniques, comme les fissures, à l'inclusion d'un large éventail de types de défauts, avec des annotations allant des étiquettes au niveau de l'image à la délimitation détaillée des défauts par des boîtes englobantes ou par segmentation. Il est à noter que les ensembles de données vont au-delà des ponts pour inclure d'autres structures en béton telles que les tunnels et les trottoirs, enrichissant ainsi la diversité des données d'entraînement pour les modèles d'apprentissage profond [130]. Cependant, le pré-traitement de ces

ensembles de données, y compris le nettoyage et le recadrage pour éliminer les éléments non défectueux, soulève des préoccupations quant aux performances des modèles dans des conditions réelles, où un tel pré-traitement contrôlé n'est pas réalisable. Cette lacune souligne la nécessité de développer des modèles robustes capables de naviguer dans la complexité des défauts réels. Pour combler cette lacune, cette thèse présente une technique de segmentation des fissures qui a été développée et qui intègre les principes de classification et de détection. Cette approche est conçue pour améliorer l'adaptabilité et la précision du modèle dans l'identification et la quantification des fissures dans une large gamme de scénarios réels, répondant ainsi au besoin critique d'une analyse précise et automatisée des défauts dans la détection des défauts du béton.

1.5 Objectifs spécifiques

Dans cette thèse, les objectifs spécifiques sont les suivants :

- Réaliser une revue systématique des méthodes d'apprentissage profond appliquées à la détection et à la classification des défauts des ponts en béton, afin d'identifier les principaux défis et opportunités.
- Développer un modèle capable de différencier précisément une véritable fissure d'une fausse fissure (par exemple, distinguer une fissure réelle des traces, graffitis ou autres artefacts non structurels).
- Optimiser la segmentation des fissures dans des images capturées à différentes distances et sous diverses conditions d'éclairage, en tenant compte des contraintes réelles d'acquisition.
- Intégrer des mécanismes avancés d'attention, tels que le mécanisme de mise au point linéaire à double attention (DALFM), pour améliorer la détection et la segmentation dans des environnements complexes.
- Proposer une approche de détection et de segmentation en temps réel adaptée aux plateformes mobiles, notamment pour l'utilisation sur des UAV, et comparer systématiquement ses performances avec celles des architectures existantes.

1.6 Contributions et structure de la thèse

L'objectif principal de cette thèse est d'améliorer la détection et la segmentation des défauts des ponts en béton à travers l'utilisation de l'apprentissage profond. Pour atteindre cet objectif, cette thèse présente deux contributions majeures :

1. ***Revue de la littérature sur la détection et la classification des défauts du béton :*** Une analyse approfondie des développements récents en matière d'apprentissage profond pour la classification et la détection visuelles des défauts des ponts en béton constitue la base de cette recherche. Cette revue répertorie méticuleusement les types de défauts, évalue les ensembles de données existants, et examine l'efficacité des modèles d'apprentissage profond actuels par rapport aux méthodes traditionnelles. La revue identifie plusieurs défis critiques : les géométries et morphologies complexes des fissures, la difficulté de distinguer les défauts dans des arrière-plans complexes, et l'influence des facteurs environnementaux tels que l'éclairage et les ombres. En outre, elle met en évidence les limites des ensembles de données actuels, notamment les déséquilibres de classes et les annotations insuffisantes, qui entravent le développement de modèles robustes. Ces défis soulignent la nécessité de solutions innovantes, préparant ainsi le terrain pour la contribution de cette thèse.

   Notre travail récent dans ce domaine inclut une revue systématique d'Amirkhani et al. [6] sur la classification et la détection des défauts visuels des ponts en béton basée sur l'apprentissage profond, où nous proposons une enquête détaillée sur les avancées dans ce domaine, identifiant à la fois les défis et les pistes prometteuses pour les recherches futures. De plus, notre étude sur l'intégration de l'intelligence artificielle et de la réalité augmentée (AI-AR) pour l'inspection des ponts par drone, présentée par Lapointe et al. [107], élargit les possibilités de détection automatisée des défauts en combinant les technologies AI avec des inspections assistées par drone. Ensemble, ces contributions posent les bases pour relever les défis complexes de la détection automatisée des défauts des ponts.

2. ***CrackSight : Une nouvelle approche pour la segmentation des fissures :*** En réponse à l'un des principaux défis identifiés dans la revue de la littérature — les arrière-plans complexes — cette thèse introduit CrackSight, un modèle d'apprentissage profond innovant qui fait progresser de manière significative l'état

de l'art en matière de segmentation des fissures. En exploitant un mécanisme de mise au point linéaire à double attention (DALFM) et une couche de saillance, CrackSight est conçu pour identifier et localiser avec précision les fissures — un type de défaut prévalent et critique dans les structures en béton — à différentes distances d'observation.

Contrairement aux modèles existants, CrackSight excelle à détecter une large gamme de types de fissures, y compris celles à peine visibles ou à leurs stades initiaux. Il intègre des cartes de caractéristiques combinées issues de notre modèle de détection de fissures basé sur les caractéristiques linéaires (*Détection de Fissures Basée sur les Caractéristiques Linéaires*, LFB-Net) et utilise la convolution à trous pour traiter habilement les motifs de fissures subtils et complexes. Cette approche surmonte efficacement les défis posés par les variations d'éclairage et les arrière-plans complexes. Une autre innovation de cette étude est la perte d'entropie croisée binaire pondérée contextuelle, qui calcule dynamiquement les poids pour traiter le déséquilibre des classes et intègre des informations contextuelles globales grâce aux poids d'attention. La performance supérieure de CrackSight est démontrée par des tests rigoureux sur divers ensembles de données et scénarios réels, soulignant son potentiel pour améliorer les pratiques de maintenance et de sécurité des ponts.

Une caractéristique distinctive de cette recherche est l'application de la technologie des drones pour la collecte de données dans des environnements réels. Cette approche répond à une limitation notable des études actuelles, qui reposent souvent sur de petits ensembles de données contrôlés qui ne capturent pas les complexités rencontrées dans l'infrastructure réelle comme les ponts en béton. En intégrant des images capturées par drones, cette étude non seulement améliore le réalisme et l'applicabilité du modèle CrackSight, mais ouvre également la voie à une détection des défauts évolutive, efficace et précise dans des contextes réels.

Notre travail récent dans ce domaine inclut le développement de CrackSight, qui est détaillé dans un article à venir par Amirkhani et al. [7] sur CrackSight : améliorer la segmentation des fissures à différentes distances, où nous fournissons une analyse approfondie des performances à travers des ensembles de données réels. De plus, notre étude sur l'utilisation de la saillance pour la détection multi-étiquette des défauts des ponts en béton à l'aide d'images capturées par UAV, présentée par Hebbache et al. [76], améliore la détection des défauts dans des

environnements complexes et réels grâce aux données capturées par drone. Ces contributions ouvrent la voie à une détection évolutive et efficace des défauts dans des applications pratiques.

La Figure 1.3 met en évidence les principaux défis de la détection des fissures sur les surfaces en béton à différentes distances, y compris les effets de la texture hétérogène du béton, les conditions d'éclairage variables, et la difficulté à distinguer les fissures des autres caractéristiques linéaires. De plus, les fissures peuvent être obscurcies par la saleté ou les graffitis, se chevaucher avec d'autres défauts, ou être manquées en raison d'un faible contraste, ce qui complique la détection précise.

Cette thèse est organisée en plusieurs chapitres, chacun dédié à un aspect spécifique de la recherche, comme suit : Le chapitre 2 présente une revue de la littérature approfondie, couvrant la taxonomie des défauts du béton, les techniques d'apprentissage profond pour la classification d'images, la détection d'objets et la segmentation, ainsi que les approches de classification et de détection des défauts du béton utilisant l'apprentissage profond. Au chapitre 3, la discussion avance vers l'évaluation comparative et les défis de la détection et de la classification des défauts, y compris un aperçu des ensembles de données, des métriques d'évaluation, des tests comparatifs des modèles, et des limitations actuelles. Le chapitre 4, intitulé "CrackSight : Une nouvelle architecture U-Net pour la segmentation des fissures à différentes distances", introduit un modèle U-Net innovant pour la segmentation des fissures, détaillant l'architecture, les fonctions de perte, les stratégies d'optimisation et l'intégration de mécanismes d'attention avancés. Enfin, le chapitre 5 conclut la thèse en résumant ses contributions à la surveillance de l'état des structures, ses implications pour le domaine, et en suggérant des pistes pour des recherches futures.

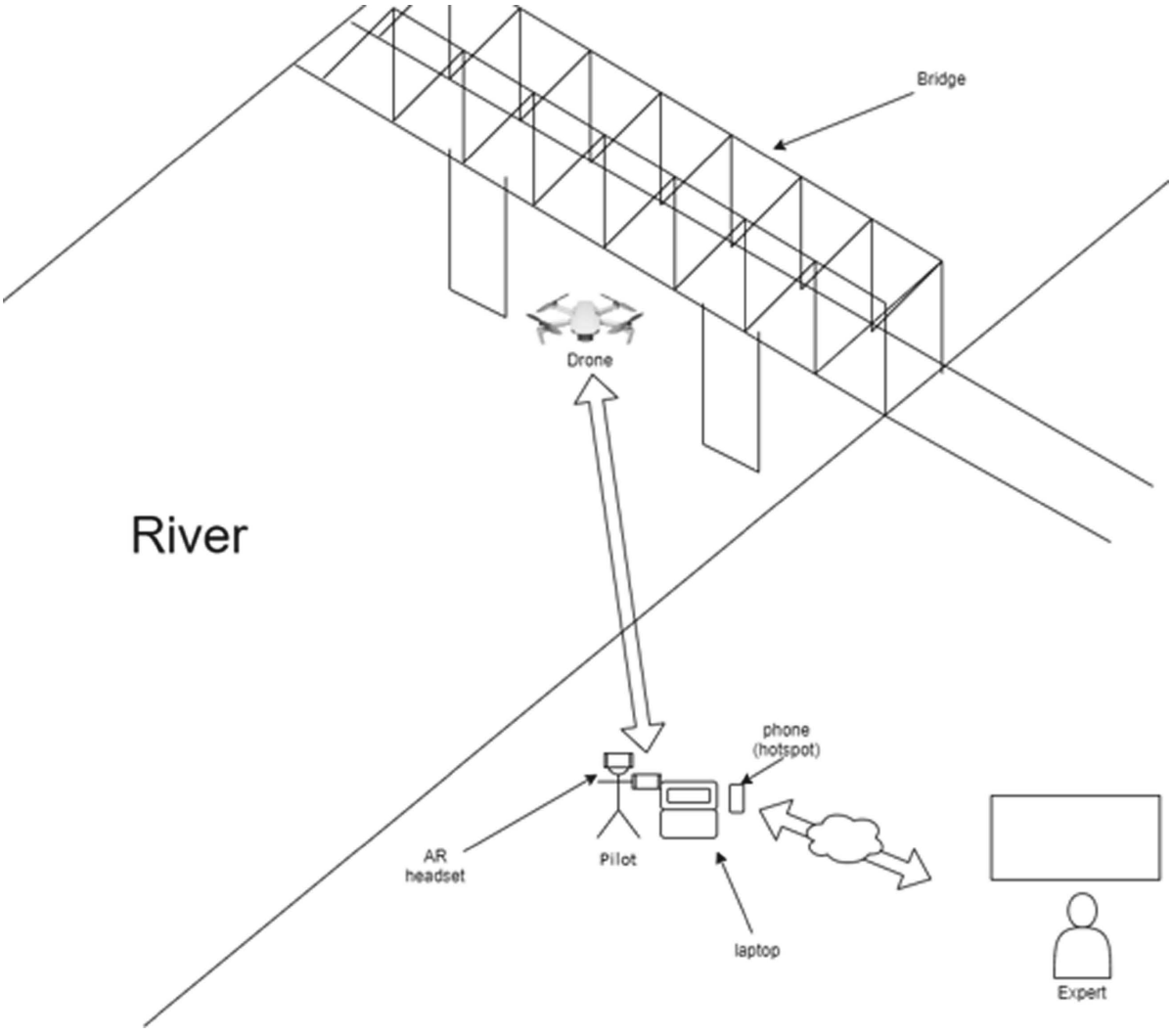


Figure 1.1: Une illustration montrant comment les drones, intégrés aux technologies IA et AR, peuvent faciliter les inspections de ponts en fournissant une détection des défauts en temps réel et des données visuelles améliorées aux inspecteurs au sol [107].

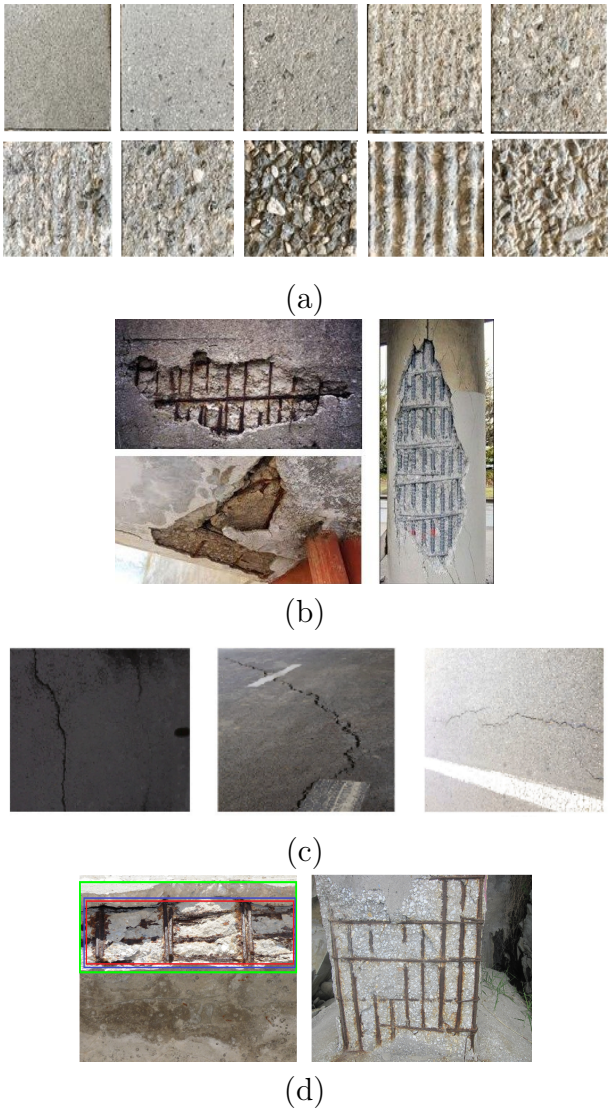


Figure 1.2: Exemples illustrant les défis de la classification et de la détection des défauts : (a) Variabilité de la texture du béton, (b) Variabilité intra-défaut pour l'effritement (c) Conditions d'éclairage variables, et (d) Ambiguïté dans l'annotation (à gauche : chevauchement de l'*Effritement*, la *Corrosion* et des *Barres exposées*), à droite : fragmentation d'une instance de défaut (*Barre exposée*).

Figure 1.3: Défis de la détection des fissures sur les surfaces en béton à différentes distances.

Chapitre 2

Revue de la littérature

2.1 Taxonomie des défauts du béton dans les ponts

2.1.1 Introduction aux défauts du béton

Les ponts en béton, éléments essentiels de l'infrastructure mondiale, exigent une surveillance continue des défauts pour garantir l'intégrité structurelle et la sécurité. Traditionnellement, cette surveillance a été manuelle, reposant sur des inspections visuelles qui sont non seulement exigeantes en main-d'œuvre mais aussi sujettes à des erreurs humaines. L'avènement de l'apprentissage profond a révolutionné ce domaine, offrant des solutions automatisées, précises et efficaces pour la détection et la classification des défauts des ponts en béton. Ce chapitre explore l'application des techniques d'apprentissage profond dans le domaine de la détection des défauts des ponts en béton. Il examine les méthodologies qui ont ouvert la voie à des avancées significatives dans ce domaine, en abordant les défis posés par les divers types de défauts, les conditions environnementales variables, et la nécessité d'une détection et d'une classification précises. En intégrant l'apprentissage profond avec les méthodes d'inspection traditionnelles, nous sommes à l'aube d'une nouvelle ère dans la surveillance de l'état des structures, où la technologie améliore notre capacité à protéger nos infrastructures.

2.1.2 Défauts étudiés dans la littérature

Dans cette section, nous donnons une brève description des défauts les plus couramment étudiés des ponts en béton dans la littérature [146]. Comme l'enquête se concentre

sur la classification et la détection des défauts dans le domaine visuel, nous décrivons principalement les défauts visibles. La Figure 2.1 présente une illustration des défauts les plus étudiés dans la littérature. Pour mieux comprendre la relation entre ces défauts, nous les avons organisés en une hiérarchie à deux niveaux. Le premier niveau contient deux défauts souvent invisibles à leur apparition (rectangles ombrés en bleu) ; mais leur progression peut entraîner d'autres défauts visibles courants (rectangles ombrés en orange) :

1. ***Corrosion des armatures*** : il s'agit de la détérioration des armatures causée par l'électrolyse chimique. Aux premiers stades, la corrosion apparaît sous forme de taches de rouille sur la surface du béton. Une armature fortement rouillée peut être exposée avec le béton environnant, ce qui peut provoquer une délamination ou un éclatement à des stades avancés.

2. ***Réaction alcali-silice (RAS)*** : elle est produite lorsque les granulats réagissent défavorablement avec les alcalis du ciment. La réaction est naturellement lente, mais ne peut pas être arrêtée ou inversée. À ses stades avancés, elle peut provoquer des fissures et des éclatements.

Le deuxième niveau contient des défauts qui sont visibles (à l'exception de la délamination qui est généralement difficile à percevoir à ses premiers stades) :

1. ***Ecaillage*** : il est induit par les effets cumulatifs du gel et du dégel, ainsi que par une mauvaise finition du béton ou un excès de travail. Le béton sans air est plus susceptible de s'écailler, mais le béton contenant de l'air peut également s'écailler s'il est complètement saturé.

2. ***Délamination*** : elle se produit lorsqu'une tranche se sépare du corps du béton, mais sans se détacher complètement. Elle commence généralement par une armature corrodée et des fissures qui s'étendent parallèlement à la surface extérieure du béton si les barres d'armature sont rapprochées.

3. ***Eclatement*** : il se produit lorsque des fragments de béton se détachent de la structure principale. Il peut être causé par la délamination lorsque le béton devient instable en raison de la pression exercée par la corrosion ou la glace.

4. ***Fissuration*** : ce sont des fractures linéaires qui s'étendent partiellement ou complètement à travers la surface du béton. Le béton et les armatures sont initialement

utilisés pour supporter la tension jusqu'à ce que la force dépasse la résistance à la traction du béton. Les fissures peu profondes peuvent également être causées par la délamination.

5. ***Efflorescence*** : elle se compose de la couleur blanche et crayeuse qui apparaît parfois sur la surface du béton. Cela peut être un problème esthétique mineur ou un signe d'infiltration d'humidité qui représente une menace sérieuse pour la structure.

6. ***Barres exposées*** : les armatures métalliques sous forme de barres doivent généralement être encastrées ou recouvertes de béton. Avec l'écaillage ou l'éclatement, ces armatures peuvent être exposées et soumises aux éléments naturels qui peuvent provoquer la rouille.

7. ***Oxydation (taches de corrosion)*** : les taches de rouille sur les surfaces en béton sont généralement une conséquence du métal (en particulier le fer) exposé à l'humidité.

D'autres défauts moins couramment étudiés que l'on peut trouver dans les manuels d'inspection [147] sont les suivants :

1. ***Érosion*** : elle se produit lorsque l'eau, le sable, le gravier ou la glace en mouvement frottent contre la surface du béton. Dans certains cas, l'érosion est accélérée par la réaction de l'air et des polluants présents dans l'eau.

2. ***Désintégration*** : elle se réfère à la désagrégation du béton en petits fragments. Elle commence généralement par un écaillage ou une fissuration, mais elle est considérée comme une désintégration lorsqu'elle dépasse un niveau sévère. Elle peut également être causée par des produits chimiques de déglaçage, des sulfates, des chlorures ou le gel.

3. ***Nid de gravier*** : il s'agit de vides ou de cavités à la surface du béton. La principale cause de ce défaut est une mauvaise vibration et une maniabilité insuffisante du béton.

4. ***Vides*** : ce sont des espaces de surface causés par de l'air emprisonné dans le béton frais. Ils se produisent généralement dans la partie supérieure ou sur les formes inclinées de la surface du béton.

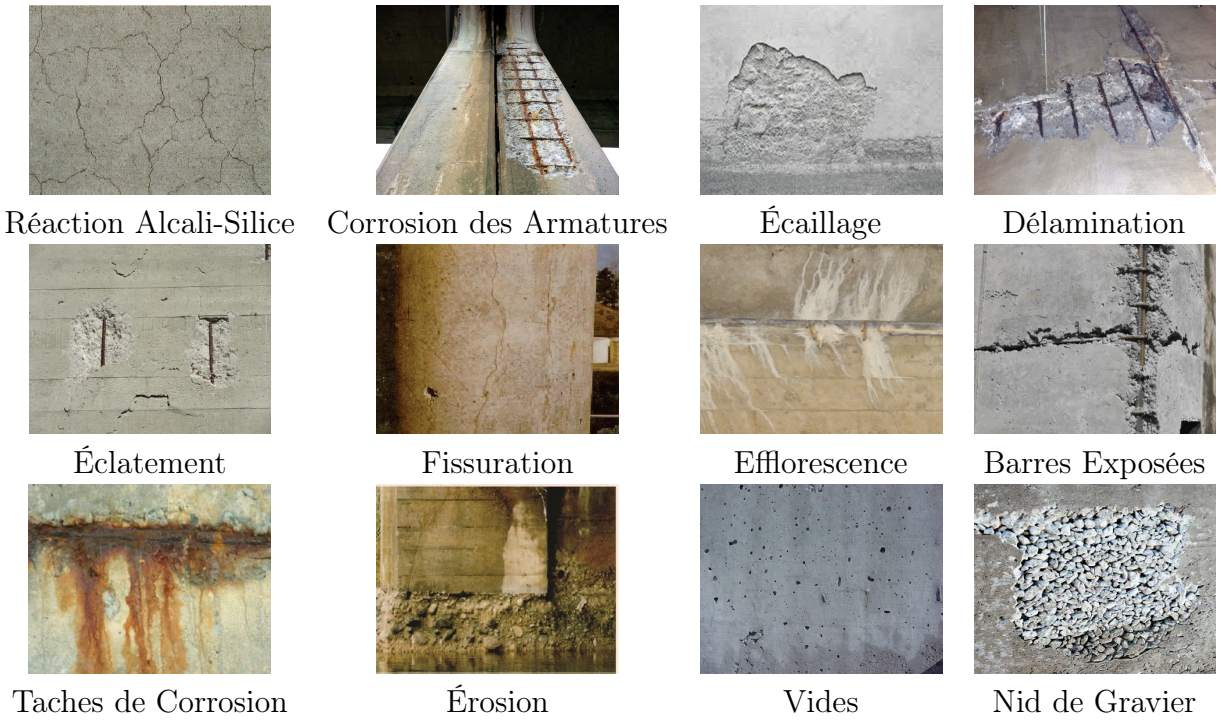


Figure 2.1: Exemples d'images des différentes classes de défauts du béton.

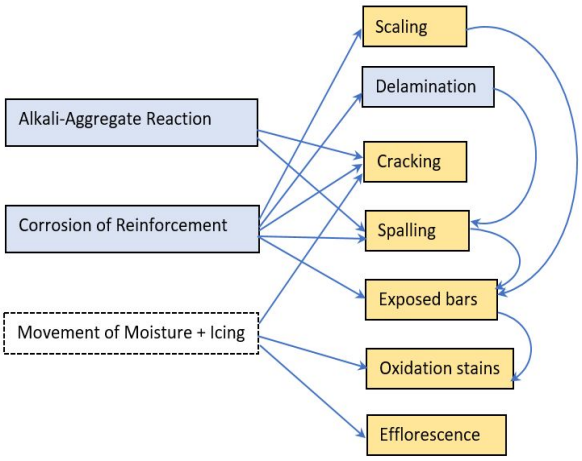


Figure 2.2: Taxonomie des défauts et leurs relations.

2.1.3 Relations entre les défauts

Bien qu'il soit utile de mener une étude approfondie pour définir tous les facteurs causant chacun des défauts mentionnés ci-dessus, cela dépasse le cadre de cette thèse. En effet, la survenue et l'évolution des défauts peuvent être liées à des processus physiques et chimiques complexes dont l'analyse va au-delà de la modalité visuelle. Néanmoins, nous tentons ici de définir certaines relations générales, mais importantes, entre les défauts, afin d'aider à comprendre certains effets causaux et à améliorer l'interprétation et l'évaluation des méthodes de classification et de détection.

Comme le montre la Figure 2.2, les défauts du béton sont intimement liés. L'expansion de la *corrosion des armatures* provoque des contraintes de traction dans le béton, ce qui peut entraîner des *fissures*, une *délamination* (invisible) et un *éclatement*. Les *barres exposées* peuvent être une conséquence de la *délamination*, de l'*écaillage* ou de l'*éclatement* ; dans l'ensemble de données CODEBRIM [138], par exemple, nous avons constaté que les défauts *barres exposées* et *éclatement* se produisaient ensemble plus de 70 % du temps. Les *barres exposées* en contact avec l'humidité peuvent entraîner des *taches de corrosion*. De plus, la *réaction alcali-silice* provoque une perte de résistance, de rigidité et d'imperméabilité du béton, ce qui, à long terme, peut entraîner des *fissures* et des *éclatements*. Enfin, le mouvement de l'humidité et le gel peuvent corroder la surface du béton et entraîner (ou exacerber) un certain nombre de défauts indésirables, tels que *l'efflorescence*, *l'oxydation des barres* et *la fissuration*. Compte tenu de ces observations, il est très probable d'observer plus d'un défaut au même endroit sur une image.

Le domaine de la surveillance de l'état des structures (SHM) englobe un large éventail de techniques et de méthodologies visant à évaluer l'état des infrastructures civiles afin d'assurer leur sécurité et leur intégrité opérationnelle. Au fil des années, ce domaine a connu une évolution significative, motivée par les avancées des technologies de capteurs, des algorithmes de traitement des données et des outils informatiques. L'importance du SHM a été soulignée par le vieillissement croissant des infrastructures à l'échelle mondiale et la nécessité urgente de méthodes efficaces pour prévenir les défaillances structurelles qui peuvent avoir des conséquences catastrophiques. Dans ce contexte, la détection des défauts dans les structures, en particulier les ponts, est devenue un domaine d'intérêt critique, nécessitant l'exploration d'approches innovantes offrant précision, efficacité et évolutivité.

Parallèlement, l'essor de l'apprentissage profond a révolutionné le paysage de la vision par ordinateur et de la reconnaissance des formes, offrant des outils puissants capables de transformer les pratiques de SHM. L'apprentissage profond, caractérisé par sa capacité à apprendre des représentations complexes à partir de grands ensembles de données, s'est avéré exceptionnellement performant pour des tâches telles que la classification d'images, la détection d'objets et la segmentation sémantique. Ces capacités sont particulièrement pertinentes pour relever les défis de la détection des défauts dans les structures civiles, où les méthodes traditionnelles se heurtent à la subjectivité des inspections manuelles et aux difficultés de détection des défauts subtils ou précoces.

Ce chapitre explore les bases théoriques et les développements historiques du SHM et de l'apprentissage profond, en mettant l'accent sur leur application à la détection des défauts dans les ponts en béton. Il débute par un aperçu des techniques SHM conventionnelles, en exposant leurs forces et leurs limites, avant de se pencher sur l'émergence de l'apprentissage profond comme solution prometteuse pour surmonter ces défis. En examinant les efforts de recherche précédents et l'évolution des méthodes de détection des défauts, cette thèse met en évidence les progrès significatifs réalisés dans ce domaine, préparant le terrain pour les chapitres suivants qui détailleront la mise en œuvre et les contributions des approches basées sur l'apprentissage profond développées dans le cadre de cette recherche.

2.2 Aperçu des jeux de données pour la détection des défauts du béton

Les jeux de données annotés sont essentiels pour l'entraînement et l'évaluation des performances des modèles d'apprentissage automatique. Pour la plupart des méthodes proposées de classification et de détection des défauts, les auteurs utilisent souvent leurs propres jeux de données, qui sont souvent prétraités pour éliminer les éléments structurels. Récemment, un certain nombre de jeux de données publics ont été proposés pour la détection et la classification des défauts du béton. Le Tableau 2.1 présente certains de ces jeux de données avec leurs propriétés. En particulier, pour chaque jeu de données, nous présentons : 1) l'année de publication, 2) la taille du jeu de données, 3) la taille de l'image, 4) la structure en béton, 5) la distance d'acquisition (proche, moyenne, large), 6) les défauts acquis et 7) les annotations fournies.

2.2.1 Caractéristiques et limitations des jeux de données existants

- **CODEBRIM [138]**. Les 1590 images haute résolution ont été collectées sur 30 ponts à différentes échelles à l'aide d'une caméra drone. Le jeu de données comprend 5354 boîtes englobantes de défauts annotées (largement avec des défauts se chevauchant) et 2506 boîtes englobantes de fond générées sans chevauchement. Les classes de défauts incluent : fissures (2507), écaillage (1898), efflorescence (833), barres exposées (1507) et taches de corrosion (1559).

- **MCDS [83]**. À partir de 38 408 images brutes, 3607 images de défauts uniques ont été extraites. Les images ont été collectées sur dix ponts autoroutiers et contiennent : fissures (789), efflorescence (311), défauts généraux (264), pas de défauts (452), écaillage (168), écaillage (427), armature exposée (223), pas d'armature exposée (203), tache de rouille (355) et pas de tache de rouille (415).

- **CDS [84]**. Le jeu de données a été collecté à Cambridge et contient 691 images sans défaut et 337 images avec défauts, organisées en deux classes binaires.

- **SDNET [39]**. Les images originales ont été subdivisées en 56 092 images, qui incluent plusieurs effets d'obstruction (par exemple, ombres, rugosité de surface,

Table 2.1: Jeux de données de défauts du béton des ponts.

Dataset	**Year**	**Taille du jeu de données**	**Taille de l'image** ($H \times W$)	**Structure en béton**	Range	**Défauts**	**Tâche**
CODEBRIM [138]	2019	1590	1280×960 to 6000×4000	Pont	Proche, Moyen, Large	Multiples	CL, DT
MCDS [83]	2019	3607	35×47 to 3966×2830	Pont	Proche	Multiples	CL
CDS [84]	2017	1029	299×299	Pont	Proche	Général	CL
SDNET [39]	2018	230	4068×3456	Pont, Mur, Pavé	Proche	Fissures	CL
BCD [209]	2019	2068	1024×1024	Pont	Proche	Fissures	CL
ICCD [114]	2019	1455	4160×3120	Ponts, Tours	Proche	Fissures	CL
CSSC [215]	2017	1232	100×100 to 130×130	Bâtiments	Large	Écaillage, Fissures	CL
Hoang et al. [77]	2019	1240	100×100	Bâtiments	Proche	Écaillage	CL
Song et al. [178]	2019	2068	1024×1024	Pont	Proche	Fissures	CL
METU [22]	2019	458	4032×3024	Pavé	Proche	Fissures	CLDT
DeepCrack [124]	2019	537	544×384	Route	Proche	Fissures	SSG
Ren et al. [160]	2020	409	4032×3016	Tunnels	Proche	Fissures	SSG
CrSpEE [10]	2021	2229	147×288 to 4600×3070	Toutes structures	Proche, Moyen, Large	Fissures, écaillage	ISG
BCL [224]	2021	1400	4032×3204 to 6000×4000	Toutes structures	Moyen	Fissures	SSG
COCO-Bridge [18]	2021	1470	256×256 to 400×360	Pont	Moyen, Large	Multiples	DT
LCW [19]	2021	3817	512×512	Toutes structures	Proche, Moyen, Large	Fissures	SSG
S2DS [12]	2022	743	1024×1024	Surfaces en béton	Proche, Moyen, Large	Multiples	SSG

trous et débris). Connu sous le nom de SDNETv1, le nouveau jeu de données contient 2025 images de fissures et 11 595 images de fond.

- **BCD [209]**. Les images originales ont été subdivisées en images de 512×512, résultant en 6069 images (4058 fissures et 2011 arrière-plans). Les images de fissures incluent des effets tels que des ombres, des taches d'eau et un éclairage intense.

- **ICCD [114]**. Les images originales ont été subdivisées en 60 000 petites images de taille 256×256 avec des distributions égales de fissures et d'images de fond.

- **CSSC [215]**. Composé de 278 images d'écaillage et de 954 images de fissures. Elles ont été subdivisées en 15 950 sous-images d'écaillage et 31 180 fissures en découpant des images en différentes tailles de sous-parties, telles que 100×100 et 130×130.

- **Hoang et al. [77]**. Le nombre d'échantillons d'images dans les catégories non écaillées et écaillées est respectivement de 620 et 620. Pour assurer la diversité, des anomalies d'image telles que des fissures et des taches ont été incluses dans le jeu de données d'images.

- **Song et al. [178]**. Les images originales ont été subdivisées en 5 180 sous-images de taille 256×256 avec des annotations binaires d'images mettant en évidence les fissures dans les images.

- **METU [22]**. Le jeu de données, provenant de divers bâtiments du campus de METU, contient 458 images haute résolution (4032×3024) à partir desquelles 40 000 images couleur de taille 227×227 ont été découpées. Elles sont divisées également en catégories négatives (sans fissures) et positives (fissures) à des fins de classification d'images, capturant une variété de finitions de surface et de conditions d'éclairage.

- **DeepCrack [124]**. Composé de 537 images couleur annotées manuellement pour la segmentation. Toutes les images ont une taille fixe de 544×384.

- **Ren et al. [160]**. Les images originales ont été collectées dans un tunnel sous diverses conditions d'éclairage, puis découpées en 919 images de taille 512×512. Photoshop a été utilisé pour annoter manuellement les images de fissures.

- **CrSpEE [10]**. Le jeu de données est similaire à Common Objects in Context (COCO). Les images ont été étiquetées par l'outil appelé le COCO Annotator, dans lequel les fissures et l'écaillage sont délimités par des polygones.

- **BCL [224]**. Les images originales proviennent de composants structurels de ponts en maçonnerie, en acier et en béton et contiennent 200 fissures. Elles ont été subdivisées en 11 000 sous-images de taille 256×256 pixels.

- **COCO-Bridge [18]**. Ce jeu de données d'images a été présenté pour soutenir un processus d'inspection de pont par véhicule aérien sans pilote (UAV), et il comprenait 774 images et 2583 instances d'objets pour la détection des défauts dans les ponts en béton (avec 322 boîtes englobantes chevauchantes et 2261 boîtes englobantes non chevauchantes) et les ponts en acier.

- **LCW [19]**. Labeled Cracks in the Wild (LCW) contient 3817 images redimensionnées à 512×512 de ponts du monde réel. Ce jeu de données détecte les fissures globales du béton, contrairement à beaucoup d'autres qui identifient des sections locales de béton.

- **S2DS [12]**. Le S2DS contient 743 images (1024×1024) principalement de défauts en béton. Les images ont été segmentées en catégories comprenant fond, fissure, écaillage, corrosion, efflorescence, végétation et point de contrôle.

2.3 Apprentissage profond pour la classification d'images

La classification d'images est une pierre angulaire des tâches de reconnaissance visuelle telles que la classification d'objets, la détection et la segmentation [184]. Elle vise à attribuer l'une des plusieurs étiquettes (ou catégories) à une image en entrée. Alors que les méthodes d'apprentissage automatique traditionnel reposaient sur la conception manuelle de caractéristiques pour la classification d'images [86], elles présentent une faible capacité de généralisation et de portabilité. Actuellement, les *réseaux de neurones convolutionnels* (CNN) et les *transformateurs de vision* (ViT) sont les modèles d'apprentissage profond les plus utilisés pour la classification d'images. Avant d'aborder la classification des défauts du béton, nous proposons un bref aperçu des modèles d'apprentissage profond populaires pour la classification d'images. Les lecteurs intéressés pourront consulter des revues approfondies dans [30, 168].

2.3.1 Aperçu des CNN

Les CNN constituent une classe de réseaux de neurones profonds conçus pour traiter des données structurées en grille, telles que les images. Inspirés par le cortex visuel, ils sont reconnus pour leur capacité à apprendre automatiquement et de manière adaptative des hiérarchies spatiales de caractéristiques à partir d'images en entrée. Les composants principaux d'un CNN comprennent des couches convolutionnelles, des fonctions d'activation, des couches de pooling et des couches entièrement connectées (voir la Figure 2.3).

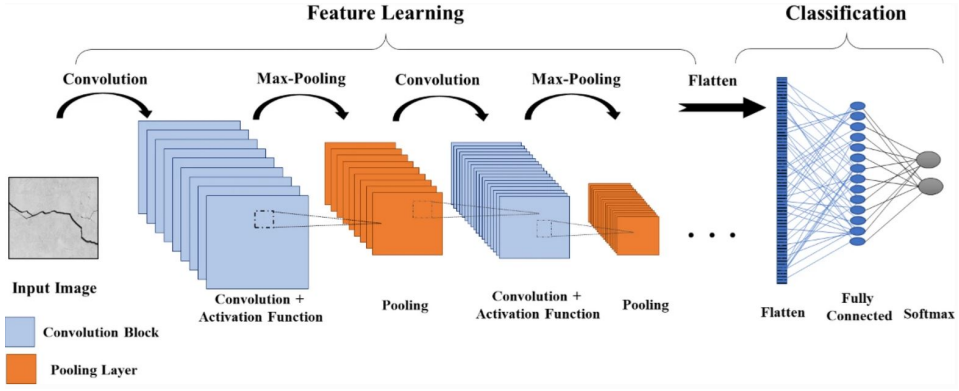


Figure 2.3: Aperçu de l'architecture d'un CNN [99].

Couches Convolutionnelles : La couche convolutionnelle constitue le bloc de construction fondamental d'un CNN. Dans cette couche, des filtres (également appelés noyaux) d'une taille donnée sont appliqués à l'image en entrée pour générer une carte de caractéristiques. Chaque filtre est capable de détecter des caractéristiques spécifiques telles que des bords, des textures et des motifs. L'opération de convolution consiste à faire glisser le filtre sur l'image et à calculer le produit scalaire entre les poids du filtre et les valeurs d'entrée. Cette opération est répétée sur l'ensemble de l'image, produisant ainsi une carte de caractéristiques qui met en évidence la présence de la caractéristique détectée (voir la Figure 2.4).

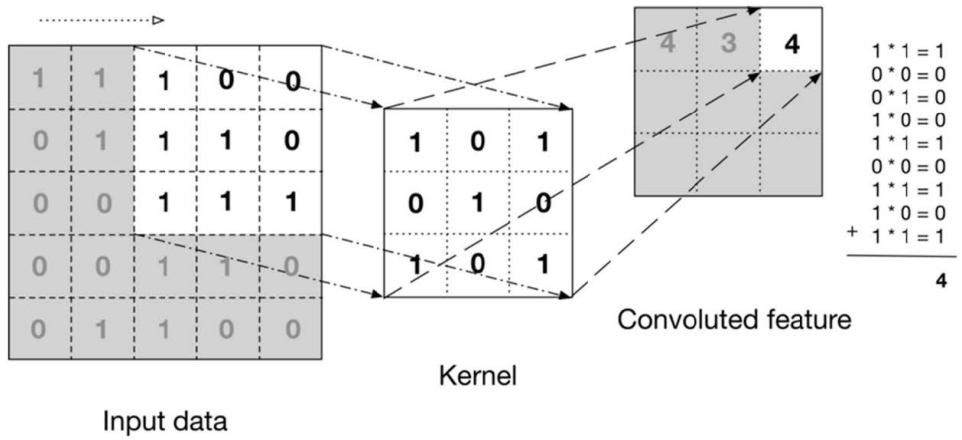


Figure 2.4: Opération de convolution dans un CNN [188].

Pas et Remplissage : Le pas (stride) détermine la distance parcourue par le filtre sur l'image (un pas de 1 signifie un déplacement d'un pixel, tandis qu'un pas de 2 signifie un déplacement de deux pixels). Le remplissage (padding) consiste à ajouter des pixels supplémentaires (généralement des zéros) autour de l'image en entrée afin de contrôler les dimensions spatiales de la carte de caractéristiques de sortie. Le résultat de l'opération de convolution est une carte d'activation indiquant où une caractéristique particulière est présente, qui est ensuite transmise à la couche suivante du réseau.

Fonctions d'Activation : Les fonctions d'activation introduisent la non-linéarité dans le réseau, permettant ainsi d'apprendre des motifs plus complexes. La fonction d'activation la plus couramment utilisée dans les CNN est la Rectified Linear Unit (ReLU).

ReLU :

$$\mathrm{ReLU}(x) = \max(0, x) \tag{2.1}$$

La fonction ReLU remplace les valeurs négatives par zéro tout en conservant la taille de la carte de caractéristiques et en améliorant la capacité du réseau à apprendre à partir des données. Elle est favorisée pour sa simplicité et son efficacité à surmonter le problème de gradient évanescent, fréquent avec des fonctions d'activation telles que le sigmoid et le tanh.

Sigmoid :

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.2}$$

La fonction sigmoid est souvent utilisée dans la couche de sortie des réseaux de classification binaire, car elle produit des valeurs comprises entre 0 et 1, ce qui convient aux modèles qui prédisent des probabilités.

Softmax :

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{C} e^{x_j}} \tag{2.3}$$

La fonction softmax est généralement utilisée dans la couche de sortie des réseaux de classification multi-classes. Elle convertit les logits des couches entièrement connectées en une distribution de probabilités sur les différentes classes.

Couches de Pooling : Les couches de pooling servent à réduire les dimensions spatiales des cartes de caractéristiques, diminuant ainsi le nombre de paramètres et les calculs dans le réseau. Ce processus aide également à rendre la détection des caractéristiques invariante à de petites translations de l'image en entrée. Le Max-Pooling sélectionne la valeur maximale dans une région donnée (souvent de taille 2x2), tandis que l'Average-Pooling calcule la moyenne des valeurs dans la région, ce qui permet de lisser la représentation (voir la Figure 2.5).

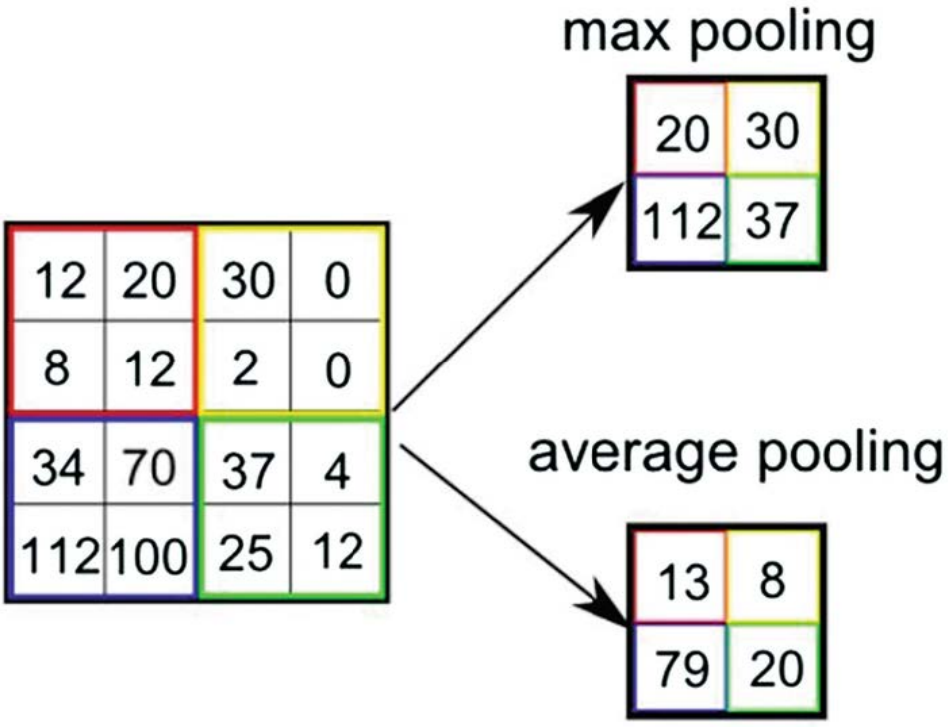


Figure 2.5: Opérations de Max-Pooling et d'Average-Pooling dans les CNN [188].

Le pooling permet de sous-échantillonner les cartes de caractéristiques en conservant les informations essentielles, contribuant ainsi également à contrôler le surapprentissage.

Couches Entièrement Connectées : Après plusieurs couches convolutionnelles et de pooling, le raisonnement de haut niveau dans le réseau est effectué par des couches entièrement connectées. Les cartes de caractéristiques sont alors aplaties en un vecteur unique, qui est utilisé pour produire un score de classification pour chaque classe.

Aplatissement des Caractéristiques : Les cartes de la dernière couche convolutionnelle sont aplaties en un vecteur 1D, servant d'entrée aux couches entièrement connectées.

Classification : Les couches entièrement connectées exploitent ces caractéristiques pour effectuer la prédiction finale, en produisant des probabilités pour chaque classe via une couche softmax.

La Figure 2.6 présente l'architecture de certains des modèles CNN les plus couramment utilisés, illustrant leur évolution et leurs différences clés.

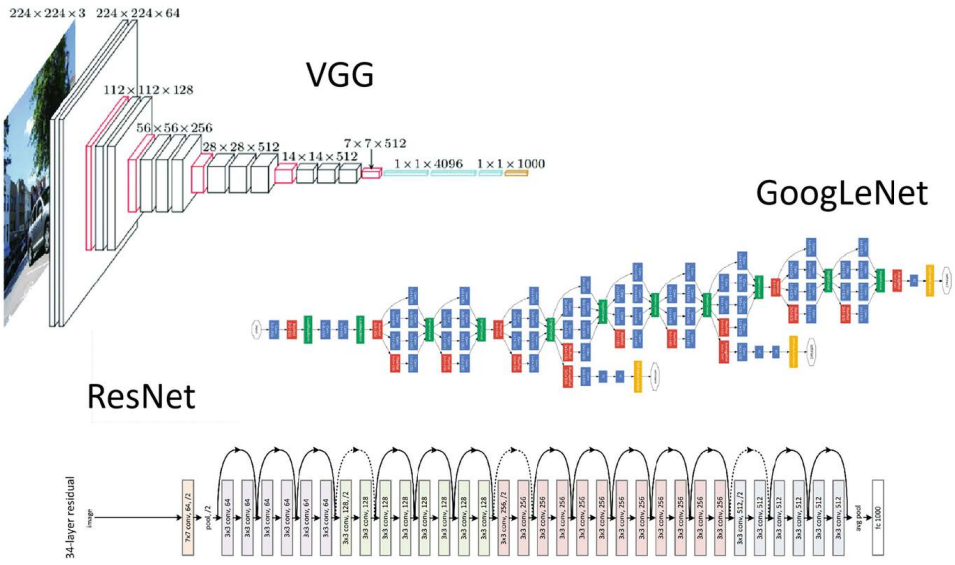


Figure 2.6: Architectures CNN courantes, y compris VGG, GoogLeNet et ResNet [188].

2.3.2 CNN pour la classification d'images

Les CNN (ou ConvNets) s'inspirent des systèmes biologiques où la connectivité entre les neurones ressemble à l'organisation du cortex visuel [110]. Contrairement à l'apprentissage automatique traditionnel, les CNN ont une capacité unique à extraire des caractéristiques hiérarchiques, en alternant les couches de convolution, de pooling

et denses. Depuis la percée d'AlexNet [105] lors du défi ImageNet [85], les architectures CNN ont progressivement amélioré leur précision, leur puissance de calcul et leur efficacité en mémoire. Bien que la tendance de conception ait encouragé l'augmentation de la profondeur des architectures, cela a entraîné l'inconvénient du surapprentissage et du *problème de gradient évanescent* (VGP) [60].

Similaire à AlexNet, VGGNet [176] empile plusieurs couches de convolution, suivies par du max-pooling. GoogLeNet [182] a atténué les problèmes des grands réseaux grâce au module *Inception*, qui incorpore plusieurs chemins d'extraction de caractéristiques avec différentes tailles de filtres. Pour atténuer le VGP, ResNet [73] a utilisé des connexions résiduelles pour améliorer le flux d'informations à travers les couches successives. Cela a permis d'entraîner des réseaux neuronaux extrêmement profonds (par exemple, ResNet100). DenseNet [81] a étendu cette idée en permettant à chaque couche de recevoir des entrées de toutes les couches précédentes, gagnant ainsi en efficacité computationnelle et en mémoire. Xception [32] a combiné les propriétés d'*Inception* et de ResNet. MobileNet [79] est un réseau léger développé pour les appareils de pointe (par exemple, les smartphones), en incorporant des convolutions séparables en profondeur. EfficientNet [185] utilise des goulets d'étranglement résiduels inversés comme blocs de construction de base avec un coefficient de facteur pour évoluer uniformément en profondeur, en largeur et en résolution du réseau.

2.3.3 Transformers de vision (ViT) pour la classification d'images

Les architectures Transformer utilisent l'attention automatique pour apprendre les relations entre les éléments d'une séquence d'entrée [96]. Contrairement aux réseaux récurrents [60] qui ne peuvent se concentrer que sur le contexte à court terme, les Transformers peuvent apprendre des relations à long terme. Le *Vision Transformer* (ViT) a adapté cette idée aux images en découpant une image en une séquence de patchs. Les couches d'attention automatique mettent à jour la représentation de chaque élément de la séquence en agrégeant des informations globales provenant de toute la séquence. De plus, des relations complexes entre différents éléments de la séquence peuvent être obtenues en utilisant l'attention multi-tête. Récemment, les modèles ViT ont gagné en popularité pour la classification d'images [96]. Dosovitskiy et al. [41] ont démontré que ViT peut atteindre des performances élevées lorsqu'il est entraîné sur un ensemble de données d'images suffisant. DeiT [190] a atteint une performance élevée en entraînant

un ViT sur un ensemble de données de taille moyenne (par exemple, 1,2 million d'images d'ImageNet) en utilisant une augmentation de données adéquate, une régularisation et une distillation de l'information. XCiT [44] a incorporé l'attention à travers les canaux de caractéristiques pour gérer les images haute résolution. Pour capturer les détails fins à différentes échelles, des conceptions hiérarchiques de ViT telles que SWIN [127] et Pyramid ViT [198] ont été proposées.

L'architecture du modèle Transformer, comme illustré à la Figure 2.7, consiste en un encodeur qui traite la séquence d'entrée (patchs d'images) pour générer une représentation complète. Cette architecture, initialement développée pour les tâches de traitement du langage naturel (NLP), a été efficacement adaptée pour la vision par ordinateur. Le mécanisme d'attention automatique de l'encodeur lui permet de se concentrer simultanément sur différentes parties de l'image, capturant ainsi les relations globales et locales. Dans la classification d'images, le décodeur est souvent omis, la sortie de l'encodeur étant directement utilisée pour la classification.

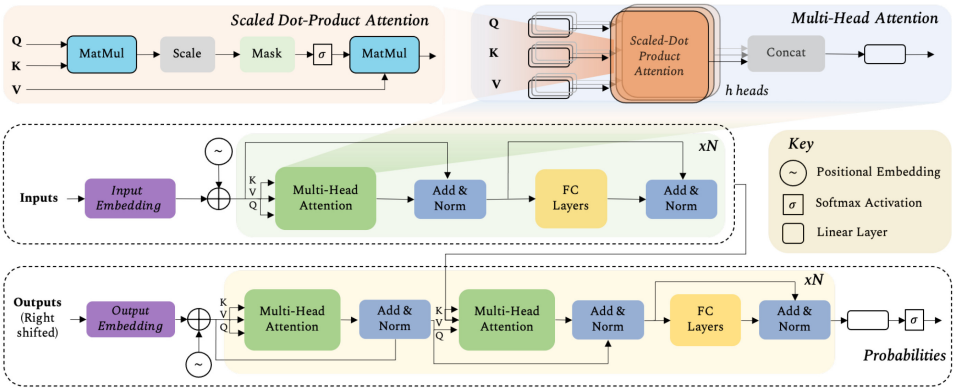


Figure 2.7: Architecture du modèle Transformer [96]. Le modèle traite une image sous forme de séquence de patchs, en intégrant chaque patch et en l'alimentant à travers une série de couches Transformer. La représentation finale est utilisée pour la classification d'images.

L'utilisation des Transformers en vision par ordinateur, en particulier des ViT, marque un changement significatif par rapport aux réseaux neuronaux convolutionnels traditionnels (CNN). Contrairement aux CNN, qui s'appuient sur des filtres de convolution pour détecter des motifs locaux, les ViT utilisent le mécanisme d'attention automatique pour modéliser les dépendances complexes à travers l'image entière. Cette capacité s'est avérée particulièrement bénéfique lorsqu'il s'agit de travailler avec des ensembles de données de grande taille, comme le montrent les améliorations des performances dans divers

benchmarks tels qu'ImageNet et CIFAR-100. En conclusion, les Transformers de Vision représentent une approche puissante pour la classification d'images, capable de capturer des motifs complexes dans les images grâce à leurs mécanismes d'attention automatique. Leur capacité à évoluer et à bien fonctionner sur de grands ensembles de données en fait un choix convaincant pour les tâches modernes de classification d'images.

2.4 Apprentissage profond pour la détection d'objets

Le succès de l'apprentissage profond pour la classification d'images a conduit à des progrès remarquables dans le domaine de la détection d'objets. La détection d'objets par apprentissage profond se divise en deux catégories : 1) la détection en deux étapes, avec une première étape générant des propositions de régions (c'est-à-dire des boîtes englobantes), et une deuxième étape affinant et classifiant les propositions de régions, et 2) la détection en une seule étape où la proposition de région et la classification sont intégrées dans un modèle unique. Le Tableau 2.2 montre les détails des modèles de détection d'objets actuels.

2.4.1 Méthodes de détection d'objets en deux étapes

Le *Réseau de neurones convolutionnel basé sur les régions* (R-CNN) [55] a inauguré l'ère de la détection d'objets en utilisant l'apprentissage profond. Il utilise l'algorithme de Recherche Sélective (SR) [191] pour produire 2000 candidats régionaux sur une image en entrée. Ensuite, chaque candidat est passé à travers un CNN pour extraire des caractéristiques afin d'entraîner un classificateur SVM. Enfin, les candidats d'objets sont affinés par la régression de la boîte englobante (BB). Pour gérer les tailles et les ratios d'aspect arbitraires des images, le SPP-Net [72] a utilisé la couche de *Spatial Pyramid Pooling* (SPP) pour déplacer et regrouper les couches de convolution. *Fast-RCNN* [56] a amélioré R-CNN en utilisant un seul réseau entraînable de bout en bout pour extraire des caractéristiques, sur lequel l'algorithme SR est utilisé pour les propositions de régions. *Faster-RCNN* [159] est allé plus loin en accélérant la détection en remplaçant le module SR par un réseau distinct qui prédit les propositions de régions.

Table 2.2: Résumé des caractéristiques des détecteurs d'objets les plus populaires

Méthode	**Paradigme de détection**	**Année**	**Backbone**	**Génération de propositions**	**Tâche**	**Neck**	**Tête de classification**	**Tête de boîte englobante**
R-CNN [55]	Deux étapes	2014	-	Recherche sélective	DT	AlexNet	Perte d'entropie croisée	Perte L2
Fast-RCNN [56]	Deux étapes	2015	VGG-16	Recherche sélective	DT	Pooling de RoI	Perte d'entropie croisée	Perte L1 lissée
Faster-RCNN [159]	Deux étapes	2015	VGG-16	Basé sur des ancres	DT	FPN	Perte d'entropie croisée	Perte L1 lissée
Mask R-CNN [74]	Deux étapes	2017	ResNeXt-101	Basé sur des ancres	SG	FPN	Perte d'entropie croisée	Perte L1 lissée
YOLO [162]	Une étape	2016	(Modifié) GoogLeNet	Basé sur des ancres	DT	-	Perte d'entropie croisée	Perte L2
SSD [122]	Une étape	2016	VGG-16	Basé sur des ancres	DT	FFDL	Perte d'entropie croisée	Perte L1 lissée
RetinaNet [119]	Une étape	2017	ResNet50	Basé sur des ancres	DT	FPN	Perte focalisée	Perte L1 lissée
EfficientDet [186]	Une étape	2020	EfficientNet	Basé sur des ancres	DT	BiFPN	Perte focalisée	Perte L1 lissée
YOLOv2 [163]	Une étape	2017	DarkNet-19	Basé sur des ancres	DT	-	Perte d'entropie croisée	Perte L1 lissée
YOLOv3 [164]	Une étape	2020	DarkNet-53	Basé sur des ancres	DT	-	Perte focalisée	Perte L2
YOLOv4 [88]	Une étape	2021	CSPDarkNet-53	Basé sur des ancres	DT	-	Perte d'entropie croisée	Perte L1 & Perte L1 lissée
YOLOv5 [58]	Une étape	2022	CSPNet	Basé sur des ancres	DT	-	Perte d'entropie croisée	Perte L1
YOLOX [54]	Une étape	2021	SPP-YOLO	Sans ancres	DT	PAFPN	Perte d'entropie croisée	Perte L1
YOLOv7 [199]	Une étape	2022	SPADE	Basé sur des ancres	DT,SG	-	Perte d'entropie croisée	Perte L1
YOLOv8 [59]	Une étape	2023	CSPDarknet-53	Basé sur des ancres	DT,SG	-	Perte d'entropie croisée	Perte L1 lissée

2.4.2 Méthodes de détection d'objets en une seule étape

You Only Look Once (YOLO) [162] considère la détection d'objets comme un problème de régression, où l'image d'entrée est divisée en plusieurs cellules de grille. Chaque cellule contient plusieurs boîtes englobantes décrites par les attributs suivants : coordonnées du centre, taille et score de confiance. Enfin, les prédictions avec des scores élevés sont conservées. Le modèle YOLO est généralement beaucoup plus rapide que ses homologues à deux étapes. Le *Single Shot MultiBox Detector* (SSD) [122] est un concurrent de YOLO qui égale la précision des détecteurs contemporains à deux étapes tout en permettant une vitesse en temps réel. Cependant, il a des difficultés à détecter de petits objets. En utilisant des boîtes d'ancrage, YOLOv2 [163] a amélioré le modèle YOLO original pour la détection de petits objets. YOLOv3 [164] a amélioré YOLOv2 en utilisant un CNN multi-échelle avec des connexions résiduelles pour couvrir diverses tailles d'objets. YOLOv4 [20] a amélioré YOLOv3 en réorganisant la détection en plusieurs parties distinctes : backbone, neck et head. Actuellement, les variantes YOLO les plus avancées en compétition sont : YOLOX [54], PPYOLOE [211], YOLOv7 [199] et YOLOv8 [59]. L'architecture de divers détecteurs d'objets tels que RCNN, Fast RCNN, Faster RCNN, RFCN, Mask RCNN, YOLO et SSD est illustrée dans la Figure 2.8, adaptée de [126].

2.5 Apprentissage profond pour la segmentation d'images

2.5.1 Principes fondamentaux de la segmentation d'images

En vision par ordinateur, la segmentation d'images est le processus de partitionnement d'une image en plusieurs segments, ou ensembles de pixels, pour simplifier sa représentation et la rendre plus significative pour l'analyse. Il existe différents types de segmentation, chacun servant un but distinct pour identifier et catégoriser les régions d'une image. Les plus couramment discutés sont la segmentation sémantique et la segmentation d'instance.

Segmentation Sémantique est une tâche de classification au niveau des pixels où chaque pixel d'une image est assigné à une étiquette de classe. L'objectif est de classifier chaque pixel dans l'une des catégories prédéfinies, telles que « route », « bâtiment » ou « arbre » dans une scène, ou « fissure » et « fond » dans le contexte de la détection de défauts. Cependant, la segmentation sémantique ne distingue pas entre les différents objets de la même classe. Par exemple, dans une image contenant plusieurs voitures, la

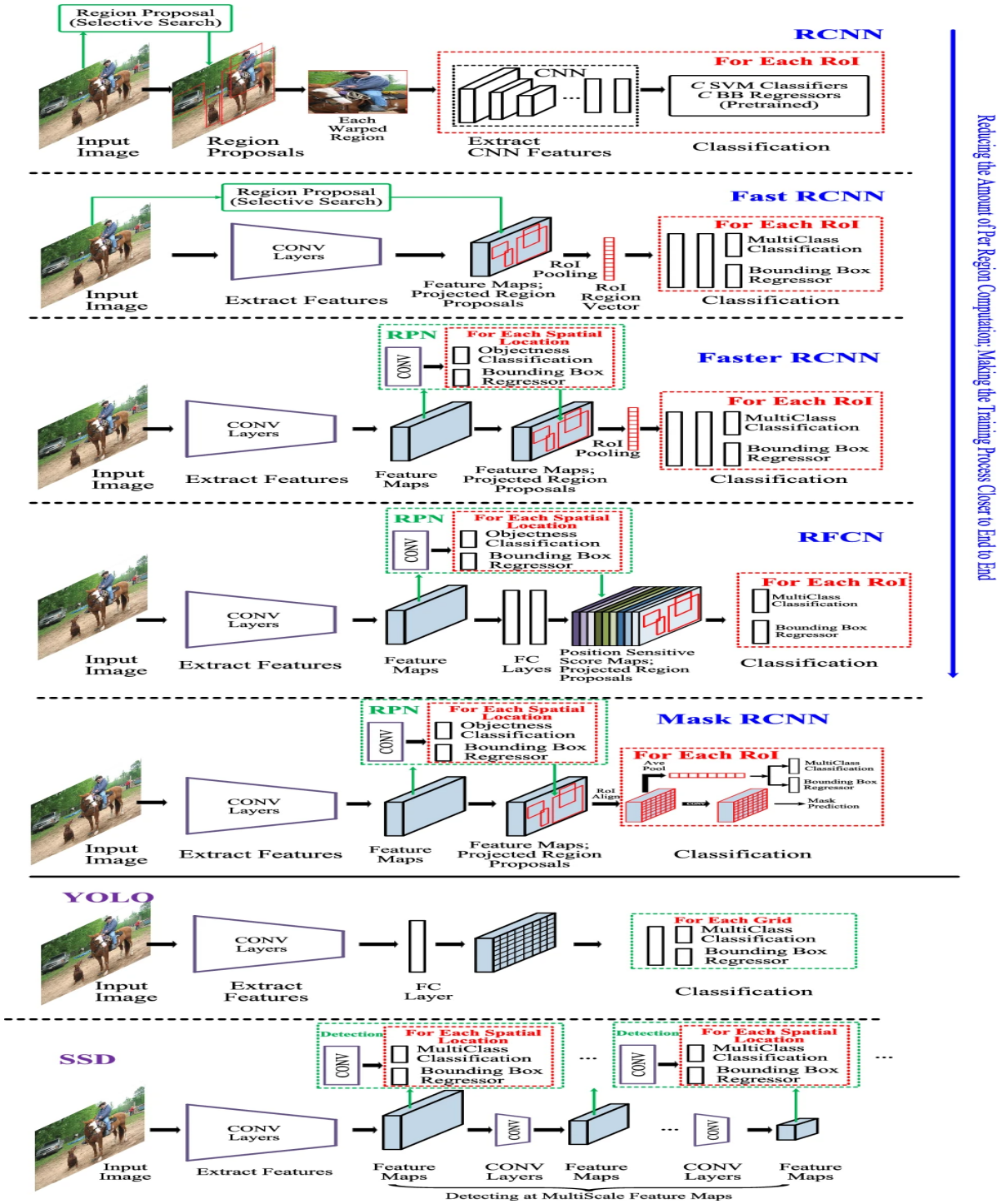


Figure 2.8: L'architecture des détecteurs d'objets populaires, y compris RCNN, Fast RCNN, Faster RCNN, RFCN, Mask RCNN, YOLO et SSD. Adaptée de [126].

segmentation sémantique étiquettera toutes les voitures avec la même catégorie, sans les différencier. Cette limitation la rend moins efficace lorsque la distinction des objets est cruciale.

Segmentation d'Instance étend la segmentation sémantique en classifiant non seulement chaque pixel dans une catégorie, mais aussi en distinguant entre différentes instances du même objet. Par exemple, si une image contient plusieurs fissures, la segmentation d'instance étiquettera séparément chaque fissure en tant qu'instance distincte, permettant une identification et une analyse précises de chaque défaut. Des techniques comme Mask R-CNN [74] ont été développées pour effectuer la segmentation d'instance en prédisant un masque binaire pour chaque objet détecté dans une boîte englobante. Comme illustré dans la Figure 2.9, les problèmes de reconnaissance liés à la détection d'objets comprennent (a) la classification d'objets au niveau de l'image, (b) la détection d'objets générique au niveau de la boîte englobante, (c) la segmentation sémantique au niveau des pixels, et (d) la segmentation sémantique au niveau de l'instance.

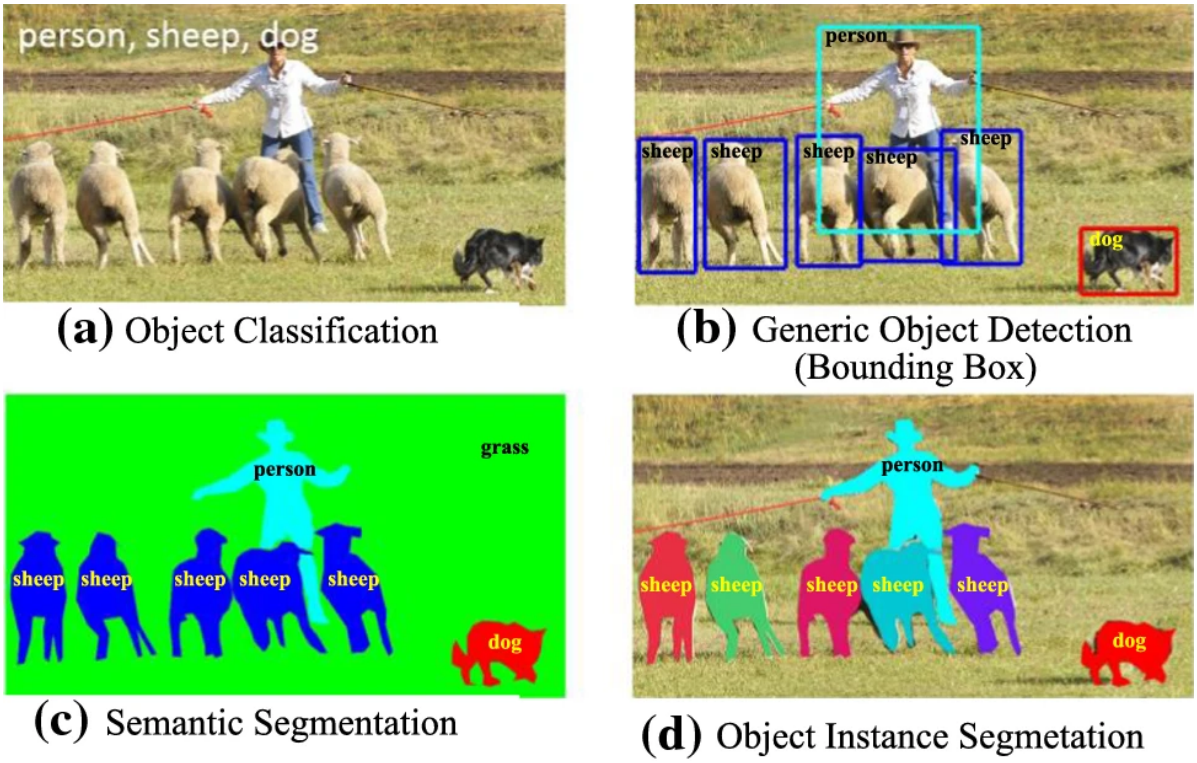


Figure 2.9: (a) Classification d'objets au niveau de l'image, (b) détection d'objets au niveau de la boîte englobante, (c) segmentation sémantique au niveau des pixels, (d) segmentation sémantique au niveau de l'instance [126].

U-Net est une architecture de réseau de neurones convolutionnel (CNN) largement utilisée, conçue principalement pour la segmentation d'images biomédicales mais qui a depuis été adoptée dans divers domaines, y compris la détection de défauts dans les infrastructures [161]. L'architecture U-Net est particulièrement efficace pour les tâches nécessitant une localisation et une segmentation précises des objets dans les images. L'architecture U-Net est nommée en raison de sa forme "U" symétrique, comprenant deux parties principales : le chemin contractant (encodeur) et le chemin expansif (décodeur). L'encodeur est responsable de capturer le contexte de l'image en réduisant progressivement ses dimensions spatiales tout en augmentant la profondeur des cartes de caractéristiques. À l'inverse, le décodeur est conçu pour restaurer la résolution spatiale des cartes de caractéristiques afin de faciliter la localisation précise des objets segmentés. Une caractéristique clé de U-Net est les connexions de saut entre les couches correspondantes de l'encodeur et du décodeur, ce qui permet au réseau de conserver des caractéristiques haute résolution de l'encodeur et d'améliorer la précision de la segmentation. L'architecture de U-Net, comme illustrée dans la Figure 2.10, met en évidence la forme "U" symétrique comprenant le chemin contractant (encodeur) et le chemin expansif (décodeur), qui sont connectés par des connexions de saut pour conserver des caractéristiques haute résolution et améliorer la précision de la segmentation.

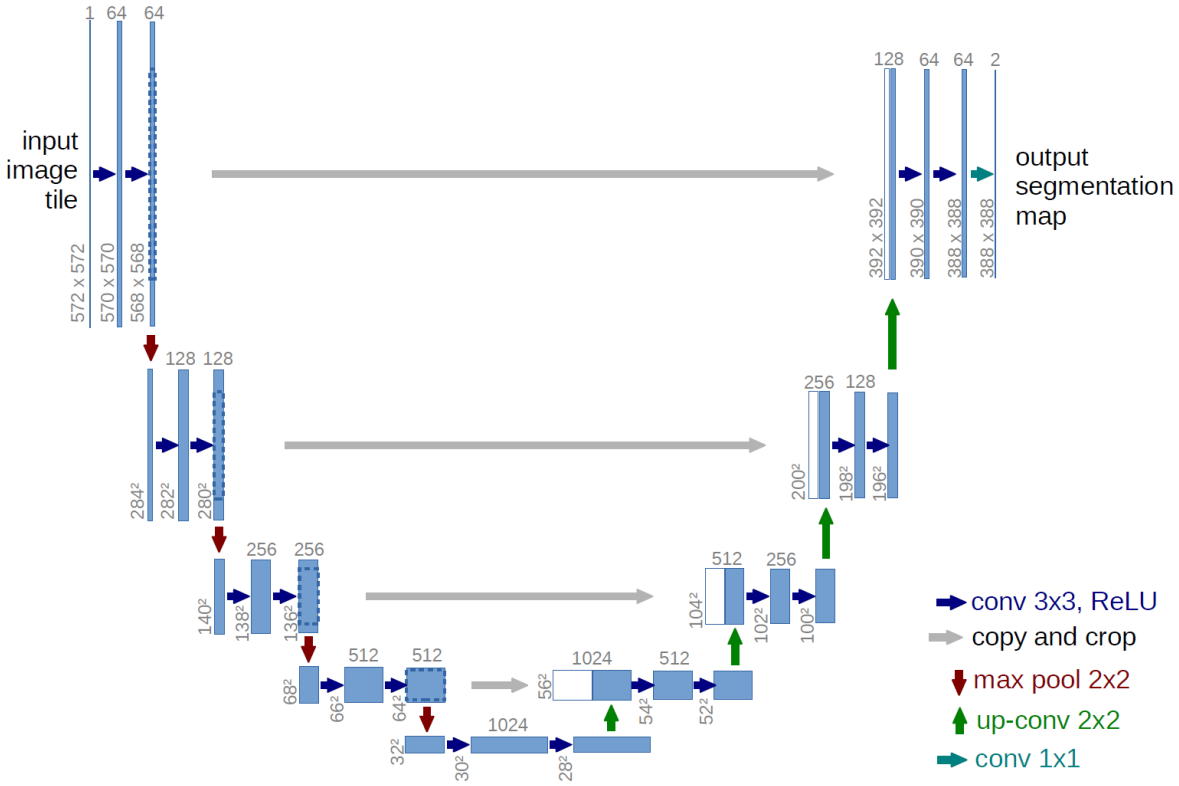


Figure 2.10: L'architecture U-Net, montrant le chemin contractant (encodeur) et le chemin expansif (décodeur), avec des connexions de saut entre les couches correspondantes [161].

Encodeur L'encodeur de U-Net suit la structure d'un réseau convolutionnel typique. Il se compose de blocs répétés de deux couches de convolution 3×3, chacune suivie d'une fonction d'activation ReLU (Rectified Linear Unit). Ces blocs sont entrecoupés de couches de max-pooling 2×2 avec un pas de 2, qui sous-échantillonnent les cartes de caractéristiques, réduisant leurs dimensions spatiales tout en doublant le nombre de canaux. Ce sous-échantillonnage progressif permet au réseau de capturer des caractéristiques de plus en plus abstraites et complexes, essentielles pour une segmentation précise.

Décodeur Le chemin décodeur de U-Net est conçu pour rééchantillonner les cartes de caractéristiques à la résolution d'image d'origine. Il le fait en utilisant des convolutions transposées, également appelées up-convolutions ou déconvolutions. Ces convolutions transposées augmentent la résolution spatiale des cartes de caractéristiques. De plus, le décodeur inclut des connexions de saut qui concatènent les cartes de caractéristiques rééchantillonnées avec les cartes de caractéristiques correspondantes de l'encodeur. Cette concaténation garantit que les détails fins perdus lors du processus

de sous-échantillonnage sont récupérés, ce qui conduit à des segmentations plus précises. Après concaténation, les cartes de caractéristiques subissent deux convolutions 3×3 suivies d'activations ReLU, affinant la sortie de segmentation.

Couche Finale La couche finale du modèle U-Net se compose d'une convolution 1×1 qui mappe les cartes de caractéristiques au nombre souhaité de classes de sortie. Cette couche produit une carte de segmentation où chaque pixel est assigné à une probabilité de classe, facilitant la classification au niveau des pixels.

Convolution Atrous La convolution atrous, également connue sous le nom de convolution dilatée, est une technique puissante utilisée dans les réseaux de neurones convolutionnels (CNN) pour étendre le champ réceptif sans augmenter le nombre de paramètres ou perdre la résolution spatiale. En introduisant des espaces (ou dilatations) entre les poids du filtre, la convolution atrous permet au réseau de capturer un contexte plus large tout en préservant les détails fins. Cette capacité est particulièrement bénéfique pour des tâches telles que la détection de fissures, où la reconnaissance des caractéristiques à plusieurs échelles est cruciale. Dans le contexte de U-Net, l'intégration des convolutions atrous permet l'extraction de caractéristiques multi-échelles en ajustant le taux de dilatation, améliorant ainsi la précision de la segmentation des objets de tailles et de formes variées. La Figure 2.11 démontre l'effet de différents taux de dilatation (1, 2 et 3) sur le champ réceptif du filtre, montrant comment l'augmentation du taux de dilatation élargit la zone de couverture tout en maintenant le nombre de paramètres du filtre constant.

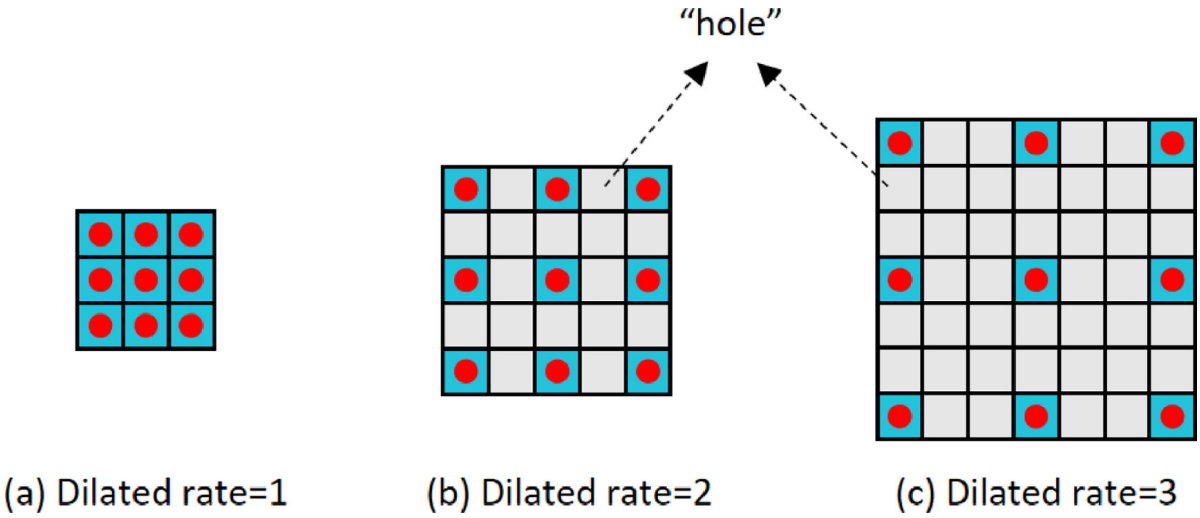
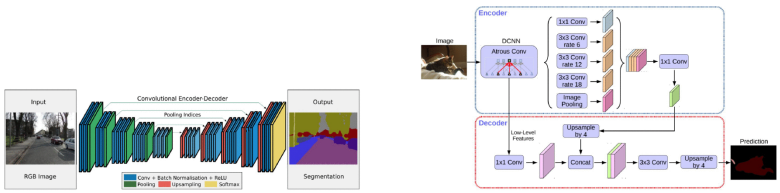
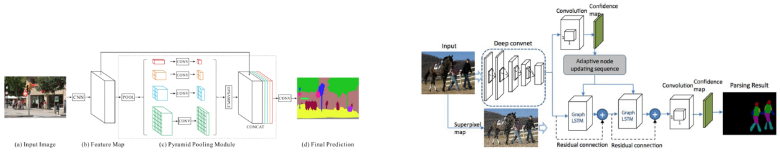


Figure 2.11: Impact de la convolution atrous avec différents taux de dilatation : (a) taux 1 (convolution standard), (b) taux 2 (dilatation modérée), et (c) taux 3 (dilatation augmentée), montrant comment le champ réceptif s'étend avec des taux de dilatation plus élevés tandis que le nombre de paramètres reste constant [70].

Au-delà de U-Net, d'autres architectures populaires dans le domaine de la segmentation d'images incluent SegNet, DeepLab-v3+, MaskLab, PSPNet, Graph-LSTM, et RAN, chacune offrant des approches et des optimisations uniques pour différentes tâches de segmentation, comme illustré dans la Figure 2.12.



(a) SegNet (b) DeepLab-v3+ (c) MaskLab



(d) PSPNet (e) Graph-LSTM (f) RAN

Figure 2.12: Architectures populaires de segmentation d'images : (a) SegNet, (b) DeepLab-v3+, (c) MaskLab, (d) PSPNet, (e) Graph-LSTM, et (f) RAN [136].

La segmentation d'image peut être formulée comme le problème d'attribution d'étiquettes aux pixels (segmentation sémantique), ou de partitionnement de l'image en objets individuels (segmentation d'instances) [136]. La segmentation d'instances étend le champ de la segmentation sémantique en permettant de délimiter chaque objet individuel dans la scène. Les modèles de segmentation sont généralement plus complexes que ceux pour la classification d'images.

Shelhamer et al. [171] ont proposé le *Réseau Convolutionnel Entièrement Convolutif* (FCN), un jalon dans la segmentation sémantique. Le FCN utilise des couches convolutionnelles qui lui permettent de produire une carte de segmentation de la même taille que l'image d'entrée. Chen et al. [26] ont proposé un modèle de segmentation combinant les CNN et les *Champs Aléatoires Conditionnels* (CRF), ce qui a permis d'améliorer la localisation des contours des segments. Chen et al. [27] ont proposé le modèle DeepLabv3+ qui a amélioré la segmentation en utilisant des convolutions dilatées. En combinant un auto-encodeur [60] et des CNN, Badrinarayanan et al. [9] ont proposé le modèle SegNet pour la segmentation sémantique d'images, qui se compose de réseaux d'encodeur et de décodeur suivis d'une couche de classification pixel par pixel. Il utilise les caractéristiques calculées lors de l'étape de max-pooling de l'encodeur pour effectuer un suréchantillonnage non linéaire dans le décodeur. Dans la même veine, Ronneberger et al. [161] ont proposé l'architecture U-Net comprenant un chemin contractant (encodeur) pour capturer le contexte, et un chemin expansif symétrique (décodeur) avec des connexions résiduelles de saut pour permettre une segmentation précise. Un examen approfondi des méthodes de segmentation basées sur l'apprentissage profond peut être trouvé dans [136].

2.6 Classification des défauts du béton en utilisant l'apprentissage profond

L'entraînement des architectures CNN à partir de zéro nécessite généralement une grande quantité d'images annotées. Grâce aux techniques d'apprentissage par transfert et de fine-tuning, les architectures pré-entraînées sur de grands ensembles de données (par exemple, ImageNet) peuvent être efficacement adaptées pour la classification des défauts en utilisant un nombre limité de données étiquetées [168]. Le principe est que les modèles pré-entraînés peuvent extraire des caractéristiques génériques qui peuvent être

utilisées pour accélérer l'entraînement des modèles spécifiques à une application [30]. Ces techniques ont été largement utilisées pour la classification des défauts du béton :

2.6.1 Classification binaire

Fait référence aux méthodes classifiant les images comme étant un défaut ou un arrière-plan. La majorité des méthodes traitent du défaut de fissure [218, 158, 3]. En effet, les fissures sont souvent indicatives de stress ou de faiblesse sous-jacente dans la structure du pont [2]. Yang et al. [218] ont étudié l'apprentissage par transfert en utilisant VGG16 pour la classification des fissures dans les infrastructures civiles. La validation expérimentale sur plusieurs ensembles de données tels que METU, SDNET et BCD a montré une haute précision. Rajadurai et al. [158] ont développé une méthode pour la classification des fissures dans le béton en utilisant un modèle AlexNet pré-entraîné. La méthode a obtenu de bons résultats sur le jeu de données METU. Ali et al. [3] ont comparé quatre architectures d'apprentissage profond pour la classification des fissures de béton sur divers ensembles de données, en se concentrant sur des aspects tels que la taille des données, l'hétérogénéité, la complexité du réseau et les époques d'entraînement. Su et al. [179] ont utilisé EfficientNet pré-entraîné sur ImageNet pour la détection des fissures dans le béton. Pour améliorer les performances, des images de fissures provenant de différents emplacements ont été utilisées pour améliorer la généralisation du modèle. Zhang et al. [225] ont proposé le modèle CrackNet pour la détection des fissures dans les pavés. En supprimant les couches de pooling, cette architecture maintient la taille de l'image inchangée, ce qui a permis au modèle d'atteindre de bonnes performances. Qi et al. [154] ont développé une méthode basée sur CNN pour détecter les micro-fissures dans les structures en béton sous l'eau. Cette méthode aborde des défis tels que l'éclairage inégal et la distorsion des couleurs typiques des environnements sous-marins grâce à des techniques avancées de prétraitement d'images. Cependant, pour que ces méthodes fonctionnent bien, elles supposent généralement que les images sont prises à courte distance ou déjà prétraitées pour éliminer les éléments structurels du pont. La Figure 2.13 illustre les résultats de classification des fissures du travail de Yang et al. [218].

(a) Crack images (b) Non-crack images

Fig. 6. Examples of images contained in the CCIC dataset.

(a) Crack images (b) Non-crack images

Fig. 7. Examples of images contained in the SDNET dataset.

(a) Crack images (b) Non-crack images

Fig. 8. Examples of images contained in the BCD dataset.

Figure 2.13: Résultats de la classification de la détection des fissures en utilisant VGG16 de Yang et al. [218].

2.6.2 Classification de défauts multiples

Les méthodes de classification de défauts multiples se situent dans l'un des deux scénarios suivants : 1) classification multi-classe, qui attribue une étiquette unique à l'image, et 2) classification multi-étiquette, qui peut attribuer plus d'une étiquette à l'image.

Gao et al. [49] ont proposé le concept de Structural ImageNet pour la classification des défauts de béton. Un petit nombre d'images (2000) ont été étiquetées manuellement pour réaliser quatre tâches de reconnaissance : *identification du type de composant* (poutre/colonne ou mur), *vérification de l'état de l'effritement* (effritement ou non), *évaluation du niveau de dommage* (pas de dommage, dommage mineur, et dommage modéré/lourd), et *détermination du type de dommage* (aucun, flexion, cisaillement, et combiné). Ensuite, un VGGNet pré-entraîné est utilisé comme base pour implémenter les quatre tâches en utilisant l'extraction de caractéristiques (apprentissage par transfert) et le fine-tuning. Les modèles obtenus ont montré des résultats prometteurs en classification. Bail et al. [10] ont développé l'approche Mask Regional CNNs pour

classifier automatiquement les défauts de béton. Un ensemble de données similaire à Common Objects in Context (COCO) [120] a été généré pour entraîner les modèles. Les tests effectués sur le jeu de données Phi-Net [51] ont montré de bonnes performances pour la classification des défauts de fissure et d'effritement. Paques et al. [150] ont proposé une approche de classification multi-étiquette utilisant Vision Transformer (ViT). L'article aborde des défis tels que le déséquilibre des classes et l'étiquetage bruyant grâce à l'apprentissage auto-supervisé et aux fonctions de perte équilibrée par classe.

Bukhs et al. [21] ont comparé les modèles VGG16 [176], ResNet50 [73] et InceptionV3 [182] pour la classification multi-classe des défauts de béton. Les modèles ont d'abord été pré-entraînés en utilisant l'apprentissage par transfert inter-domaine ou intra-domaine, puis ajustés pour classifier les défauts. Ils ont conclu que l'apprentissage par transfert inter-domaine donne généralement de meilleures performances. De plus, la combinaison de l'apprentissage inter-domaine et intra-domaine a donné de meilleurs résultats que l'utilisation de leurs homologues seuls. Mundt et al. [138] ont proposé le jeu de données CODEBRIM et comparé deux approches d'apprentissage par renforcement basées sur l'apprentissage méta, MetaQNN et la recherche efficace d'architecture de réseau neuronal, pour trouver des CNN adaptés à la classification multi-classe des défauts. Zhu et al. [235] ont utilisé un modèle InceptionV3 pré-entraîné pour la classification de défauts multiples, ce qui a donné une haute précision sur les images collectées par les auteurs. Cependant, aucun test n'a été effectué sur d'autres ensembles de données. Höthwohl et al. [83] ont proposé une approche hiérarchique en trois étapes basée sur InceptionV3, pour classifier les images comme ne contenant pas de défaut, un défaut ou plusieurs défauts. Shin et al. [175] ont proposé un modèle de classification de défauts multiples (CMDnet) qui étend VGG16 [176] en ajoutant une couche de pooling hybride et un module d'attention. La méthode a montré de hautes performances pour classifier les fissures, l'exposition des armatures et la délamination. Zoubir et al. [240] ont proposé un ensemble de données de plus de 6 900 images présentant trois défauts courants des ponts en béton : fissures, efflorescence et effritement. Ensuite, trois modèles basés sur VGG16 [176] ont été entraînés et comparés pour la classification des défauts. La Figure 2.14 démontre les résultats de classification de défauts multiples utilisant le modèle CMDnet.

Figure 2.14: (a) Intact; (b) Fissure; (c) Exposition des armatures; (d) Délamination; (e) Fuite. Résultats de classification de défauts multiples utilisant le modèle CMDnet de Shin et al. [175].

La classification des défauts du béton permet de classer rapidement les images en catégories de défauts. De plus, les annotations au niveau de l'image sont relativement faciles à obtenir. Cependant, pour réussir, les images doivent être préalablement nettoyées des éléments structurés. Enfin, les méthodes de classification ne fournissent pas d'information sur la localisation des défauts. Le Tableau 2.3 ci-dessous résume certains des principaux algorithmes de classification de défauts multiples présentés dans la littérature.

Table 2.3: Résumé des algorithmes de classification de défauts multiples.

Référence	**Architecture**	**Jeu de données**	**Tâche**	**Année**
Gao et al. [49]	VGGNet (pré-entraîné)	Structural ImageNet	Multi-classe	2018
Bail et al. [10]	Mask R-CNN	COCO-like	Multi-classe	2021
Paques et al. [150]	Vision Transformer	Phi-Net	Multi-étiquette	2023
Bukhs et al. [21]	VGG16/ResNet50/InceptionV3	Divers	Multi-classe	2021
Shin et al. [175]	CMDnet (VGG16 + pooling hybride et attention)	SF (1981 images)	Multi-classe	2020
Zoubir et al. [240]	VGG16	SF (6 900 images)	Multi-classe	2022

2.7 Détection des défauts du béton en utilisant l'apprentissage profond

La détection des défauts vise à estimer l'emplacement des défauts présents dans l'image. Les méthodes existantes ont abordé cet objectif par deux approches principales : 1) Localisation des défauts par des boîtes englobantes (BB) et 2) Localisation des défauts par segmentation :

2.7.1 Détection de défauts de béton basée sur des boîtes englobantes à deux étapes

Kim et al. [97] ont utilisé le R-CNN pour la détection des fissures. Cependant, la méthode peut manquer les fissures non contrastées et est intensivement computationnelle. Haciefendioglu et al. [67] ont utilisé Faster-RCNN pour détecter les fissures de pavé. La méthode a une haute précision pour la détection des fissures, mais elle n'est pas robuste aux changements d'échelle. Yao et al. [222] ont proposé une méthode basée sur GoogleNet [182] pour la détection des trous de boulons. La méthode est capable de détecter des défauts même lorsqu'elle est entraînée sur un nombre limité d'exemples étiquetés et en présence d'illuminations non uniformes et d'ombres. Kang et al. [91] ont proposé une méthode pour la détection, la localisation et la quantification des fissures. Tout d'abord, Faster-RCNN est utilisé pour produire des BB pour les fissures, qui sont ensuite segmentées en parties fissure et arrière-plan. Ensuite, l'épaisseur et la longueur des fissures sont calculées pour évaluer le défaut. Mishra et al. [140] ont développé une méthode en deux étapes pour la détection des fissures basée sur YOLOv5 [58]. Tout d'abord, les fissures sont identifiées et localisées en utilisant des BB, puis la longueur des fissures est calculée.

Xu et al. [209] ont proposé une méthode basée sur Fast-RCNN pour détecter les dommages sismiques sur les surfaces en béton armé (par exemple, fissures, effritement, exposition des armatures et flambement). La méthode offre généralement une bonne précision, mais manque d'efficacité puisque les ancres de BB sont extraites d'une seule image dans chaque lot. Li et al. [111] et Marin et al. [132] ont développé des modèles basés sur Faster-RCNN pour la détection et la localisation des défauts de béton. Ces méthodes ont donné une bonne précision, mais elles nécessitent un temps d'exécution élevé. Wan et al. [196] ont proposé une méthode de détection des défauts du béton

basée sur ViT. La méthode offre une bonne précision de détection, mais nécessite de grands ensembles de données pour l'entraînement du modèle. Fu et al. [48] ont proposé un modèle pour la détection des fissures de béton basé sur EfficientNetV2 avec des améliorations dans ses blocs initiaux, intermédiaires et finaux. Le modèle a été intégré à Faster R-CNN, et a démontré de hautes performances pour la classification et la détection des fissures en utilisant plusieurs ensembles de données.

2.7.2 Détection des défauts de béton par des boîtes englobantes à une étape

Teng et al. [187] et Deng et al. [37] ont proposé des méthodes basées sur YOLOv2 pour la détection des fissures. Ils ont montré de bons résultats pour la détection en temps réel sur des images prises à courte distance. Cui et al. [35] ont proposé une version améliorée de YOLOv3 pour détecter l'érosion sur les surfaces en béton. La méthode a obtenu de bons résultats sur les défauts à courte distance sur des fonds propres. Zhang et al. [227] ont utilisé le YOLOv3 pour la détection multi-classes des défauts de béton (par exemple, fissures, éclatements, écaillages, et barres exposées). Le modèle a obtenu de bonnes performances pour les défauts à courte distance. Wu et al. [204] ont utilisé YOLOv4 pour la détection des fissures de béton en employant une stratégie d'élagage pour surmonter le problème des CNNs sur-paramétrés. Wang et al. [201] ont développé un modèle automatisé de détection des défauts de béton en une seule étape basé sur EfficientNet. Le modèle fusionne des caractéristiques de bas niveau et de haut niveau pour une meilleure détection des défauts.

Jiang et al. [88] ont utilisé une version améliorée de YOLOv3 pour la détection de défauts multiples sur des surfaces en béton. Le modèle combinait EfficientNet-B0 et MobileNetV3, pré-entraînés sur MS-COCO, et des convolutions séparables en profondeur. Kumar et al. [106] ont utilisé YOLOv3 pour détecter les éclatements et les fissures sur les surfaces en béton. La validation de la méthode a été effectuée sur 800 images annotées, déjà recadrées pour faciliter la localisation des défauts. Zou et al. [239] ont utilisé YOLOv4 pour détecter plusieurs types de défauts de béton, tels que les fissures, les éclatements, les barres exposées et les barres fléchies. Ils ont également utilisé des convolutions séparables en profondeur pour réduire le coût de calcul pour l'entraînement et la prédiction du modèle. La plupart des modèles ci-dessus obtiennent généralement de bonnes performances pour les images à courte distance.

La détection des défauts de béton par des boîtes englobantes (BB) présente plusieurs avantages pratiques en ce que les annotations sont relativement faciles à obtenir, et les modèles peuvent effectuer une détection en temps réel. Cependant, ils ont une limitation pour détecter les défauts à différentes distances d'acquisition ou pour localiser de petits défauts sur des fonds encombrés. Enfin, la plupart des méthodes ci-dessus ont été proposées pour localiser des défauts simples et connectés. Par conséquent, elles sont limitées pour traiter les défauts qui se chevauchent et les défauts fragmentés.

2.8 Détection des défauts par segmentation

Pour générer une segmentation grossière des fissures, Dorafshan et al. [40] et Han et al. [69] ont entraîné des classificateurs basés sur CNN pour étiqueter des patchs d'images locales en classes de fissures ou d'arrière-plan. Cependant, étant donné le nombre de patchs classifiés, ces méthodes entraînent un coût computationnel énorme, en plus de leur manque de précision pour identifier les détails fins des fissures. Pour une segmentation fine des fissures, Dung et al. [42], Zhang et al. [228], et Benz et al. [13] ont proposé des méthodes basées sur les modèles FCN [171], SegNet [9] et U-Net [161], respectivement. Ces méthodes ont montré une bonne précision sur des images à courte distance avec des arrière-plans propres. Cependant, elles perdent leur efficacité lorsque les images contiennent des défauts qui se chevauchent ou des arrière-plans encombrés. Mei et al. [134] ont proposé une méthode de segmentation des fissures basée sur un réseau neuronal densément connecté avec fusion de caractéristiques à plusieurs niveaux et une nouvelle fonction de perte prenant en compte la connectivité des pixels. La méthode a donné une bonne précision de segmentation pour des images acquises avec différents réglages de caméra.

Yang et al. [216] et Bhowmick et al. [17] ont proposé des méthodes basées sur les modèles FCN [171] et U-Net [161], respectivement, pour segmenter les fissures sur des images de surfaces en béton. Les deux méthodes ont ajouté des étapes de post-traitement basées sur des opérations morphologiques pour mesurer la longueur, la largeur, la surface et l'orientation des fissures. Meng [133] a proposé le modèle CrackNet pour la segmentation des fissures basé sur une combinaison des architectures FCN et U-Net. Ce modèle a utilisé la colonne vertébrale ResNet101 pour extraire un ensemble de caractéristiques riches pour identifier les détails fins des fissures. Sun et al. [181] ont proposé un modèle basé sur le modèle DeepLabv3+ [27] pour segmenter les défauts de fissures et de trous de

vers sur des surfaces en béton. Grâce à l'utilisation du modèle Xception [32], la méthode a permis une segmentation précise des deux défauts à plusieurs échelles. Cependant, il n'est pas facilement applicable aux défauts qui se chevauchent ou aux défauts acquis à différentes distances.

Li et al. [113] ont proposé une méthode pour la segmentation de plusieurs défauts de béton basée sur le FCN [171] et la colonne vertébrale DenseNet121. La méthode peut segmenter plusieurs types de défauts, y compris les fissures, l'effritement, l'efflorescence et les trous. Zou et al. [238] ont proposé le modèle DeepCrack pour segmenter les fissures de pavage, basé sur une architecture d'auto-encodeur inspirée du modèle SegNet [9]. Les tests effectués sur des ensembles de données de pavage ont montré de bonnes performances. Wang et al. [200] ont comparé cinq modèles de segmentation de fissures de bout en bout utilisant différentes approches, et DeepLabv3+ avec un ResNet101 [27] a obtenu la plus haute précision. Yang et al. [217] ont proposé un réseau de boost pyramidal et hiérarchique (FPHBN) pour la détection des fissures de pavage, qui intègre des informations contextuelles aux caractéristiques de bas niveau. La méthode permet de segmenter précisément les fissures à courte distance. Zhang et al. [229] ont proposé un réseau neuronal à trois flux qui combine des informations spatiales, contextuelles et de bordure pour une segmentation fine des fissures. Al-Huda et al. [4] ont proposé un modèle pour la segmentation sémantique des fissures de pavage qui intègre une fonction de perte hybride renforçant la détection des fissures fines et l'affinement des bordures des fissures. Le modèle a obtenu une haute précision pour plusieurs ensembles de données. Mask R-CNN est un développement de Faster RCNN en prédisant un masque binaire supplémentaire qui identifie les pixels d'objets à l'intérieur des boîtes englobantes [74]. Kim et al. [98] ont développé une méthode utilisant Mask R-CNN pour la segmentation des défauts d'instance qui est réalisée en parallèle avec la détection. La méthode a été appliquée avec succès à la détection de multiples défauts de béton, y compris les fissures, l'efflorescence, les barres exposées et l'effritement, où de bonnes performances ont été obtenues.

La segmentation des défauts de béton a le mérite de localiser précisément les bordures des défauts. Cependant, les annotations pour la segmentation sont difficiles et longues à obtenir. De plus, les méthodes de segmentation ont une limitation pour détecter de petits défauts ou des défauts non contrastés sur des fonds encombrés ou non homogènes. Ces méthodes sont également moins efficaces pour traiter les défauts qui se chevauchent.

Table 2.4: Méthodes représentatives de classification et de détection de défauts concrets basées sur le deep learning (Partie 1). Dans la colonne précision, nous rapportons la précision de la classification, le mAP de détection ou le F1 score de segmentation. Pour les articles avec plusieurs jeux de données, les meilleurs résultats sont notés. Si d'autres métriques sont utilisées, elles sont spécifiées.

Référence	Backbone	Jeu de données pré-entraîné	Architecture	Jeu de données	Classes de défauts	Tâche	Code	Année	Précision (%)
Dorafshan et al.[39]	AlexNet	IMAGENET	CNN	SDNET	Fissure	CL	×	2018	95.5
Su and Wang [179]	EfficientNetB0	IMAGENET	CNN & TL	SDNET, Li & Zhao	Fissure	CL	✓	2020	99.1
Rajadurai and Kang[158]	AlexNet	-	CNN & TL	Concrete Crack Images for Classification	Fissure	CL	×	2021	99.9
Qi et al.[154]	-	-	CNN	SF(TI: 500)	-	CL	✓	2022	93.9
Ali et al.[3]	4 CNN connus	IMAGENET	CNN & TL	SDNET & METU	Fissure	CL	×	2021	95.3
Yang et al.[218]	VGG16	IMAGENET	CNN & TL	METU, SDNET & BCD	Fissure	CL	×	2020	99.8
Zhu et al.[235]	Inception-v3	IMAGENET	CNN & TL	SF(TI: 1458 & 5 classes)	MD	CL	×	2020	97.8
Hötlwohl et al.[83]	Inception-v3	IMAGENET	Trois CNN & TL	MCDS	MD	CL	×	2019	95.6
Bhattacharya et al.[15]	Res2Net[50]	-	iDAAM (AT)	CODEBRIM, SDNET	MD	CL	×	2021	92.7
Zoubir et al.[240]	VGG16	IMAGENET	CNN & TL	SF(TI: 6952 &4 classes)	MD	CL	×	2022	97.1
Bhattacharya et al.[16]	Module inédit	-	MSCA,FGIA (AT)	CODEBRIM	MD	CL	×	2021	86.15
Gao et al.[49]	VGGNet	IMAGENET	CNN & TL	SF(TI: 2,000 & 4 classes)	MD	CL	×	2018	68.8
Mundt et al.[138]	-	-	CNN	CODEBRIM	MD	CL	✓	2020	72.2
Shin et al.[175]	-	-	CMDnet CNN (AT)	SF(TI: 1981 & 5 classes)	MD	CL	×	2019	98.9
Pâques et al.[150]	DINO[24]	IMAGENET	ViT/s8	SOFIA dataset(TI: 53,805 & 13 classes)	MD	CL	×	2023	89.0
Bukhs et al.[21]	VGG16, Inception-v3, et ResNet50	IMAGENET	CNN affiné	CDS, SDNETv1, BCD, ICCD, MCDS, CODEBRIM	MD	CL	✓	2021	90
Marin et al.[132]	3 CNN connus	IMAGENET	Faster RCNN	Li et Zhao [114]	Fissure	DT	×	2021	93.2
Kim et al. [97]	-	IMAGENET, Cifar-10	RCNN	SF(TI: 384)	Fissure	DT	×	2018	-
Hacefendioğlu et al.[67]	VGG-16	SF	Faster R-CNN	SF(TI: 323 & 1128 boîtes englobantes de fissures)	Fissure	DT	✓	2021	70.0
Fu et al.[48]	-	-	StairNet CNN	Fissures routières, dataset de fissures[61] & SF(TI: 674)	Fissure	DT	✓	2023	91.6
Joshi et al.[90]	Mask RCNN	COCO	CNN	SF(TI:3000)	Fissure	DT	×	2022	93.4
Zhong et al.[234]	WGAN-GP	IMAGENET	WGAN-GP, YOLOv3	SF(TI: 2000)	Fissure	DT	✓	2023	81.9
Teng et al.[187]	11 CNN connus	IMAGENET	YOLO-V2	SF(TI: 990)	Fissure	DT	×	2021	89
Deng et al.[37]	VGG16	-	YOLO-V2	SF(TI: 3010)	Fissure	DT	×	2021	76
Wu et al.[204]	CSPDarknet53	-	YOLO-v4 Amélioré	SF(TI: 3176 & 6 classes)	Fissure	DT	×	2022	92.5
Kang et al.[91]	ResNet-50	COCO	Faster-RCNN	SF(TI: 1400)	Fissure	DT	×	2020	95.0
Mishra et al.[140]	Darknet-53	IMAGENET	YOLOv5	SF(TI: 4500)	Fissure	DT	×	2022	95.0
Liu et al.[130]	-	-	DDACDN	CQU-BPDD, RDD2020, CQU-BPMDD	Fissure	DT	×	2023	82.6
Cui et al.[35]	Darknet53	-	YOLO-v3	SF(TI: 1,860)	Érosion	DT	×	2021	75.6
Yao et al.[222]	-	-	DCNN	SF(TI: 116, Fissure & trou d'insecte)	MD	DT	×	2019	96.4
Wang et al. [201]	EfficientNetB0	-	Détecteur nouveau	CODEBRIM	MD	DT	×	2022	83.2
Xu et al. [209]	-	-	Faster-RCNN	SF(TI: 4058)	MD	DT	×	2019	96.3
Li et al.[111]	Réseau ZF	IMAGENET	Faster-RCNN	The Pittsburgh Dataset (TI: 254,000)	MD	DT	×	2018	86.0
Wan et al.[196]	-	-	ViT	SF (TI: 3,500)	MD	DT	×	2023	91.9
Zhang et al.[227]	Darknet-53	COCO	YOLOv3 Modifié	SF(TI: 145)	MD	DT	×	2020	71.8
Jiang et al.[88]	EfficientNet & MobileNetV3	COCO	Fast-YOLO	SF(TI: 14,756 & 4 classes)	MD	DT	×	2021	64.8
Kumar et al.[106]	-	COCO	YOLO-v3	SF(TI: 800 & 2 classes)	MD	DT	×	2021	94.2
Zou et al.[239]	CSPDarknet53	-	YOLO-v4 Amélioré	SF(TI: 924)	MD	DT	×	2022	92.6
Hebbache et al.[76]	ResNet50	COCO	RetinaNet, AT	CODEBRIM	MD	DT	×	2023	99.1
Zhang et al.[231]	ResNet	BCD	CR-YOLO	SF(TI: 800)	Fissure	SG	×	2022	86.2
Wang et al.[198]	-	-	Inception-Resnet-v2	SF OCD (TI: 2,000), METU & SDNET	Fissure	SG	×	2021	98.8
Han et al.[69]	AlexNet	IMAGENET	CNN	SF(TI: 150)	Fissure	SG	×	2022	91.0
Ding et al.[42]	VGG16	IMAGENET	CNN	600 images de METU	Fissure	SG	×	2019	89.3
Lin et al. [121]	-	-	CNN et AT	Maçonnerie & Rissbilder [149]	Fissure	SG	×	2023	76.6

Table 2.5: Méthodes représentatives de classification et de détection de défauts concrets basées sur le deep learning (Partie 2). Dans la colonne précision, nous rapportons la précision de la classification, le mAP de détection, ou le F1 score de segmentation. Pour les articles avec plusieurs jeux de données, les meilleurs résultats sont notés. Si d'autres métriques sont utilisées, elles sont spécifiées.

Référence	Backbone	Jeu de données pré-entraîné	Architecture	Jeu de données	Classes de défauts	Tâche	Code	Année	Précision (%)
Al-Huda et al. [4]	Xception	IMAGENET	KTCAM-Net & AT	DeepCrack, Crack500[218], CFD[172], CrackSC[65]	Fissure	SG	×	2023	96.0
Zhang et al.[228]	VGG16	IMAGENET	deep semantic SG	CFD, TRIMMD & CFTD	Fissure	SG	×	2019	82.5
Benz et al.[13]	VGG16	IMAGENET	TERNAUSNET	UAS	Fissure	SG	✓	2019	84.7
Yang et al. [216]	VGG19	IMAGENET	FCN	SF(TI: 800)	Fissure	SG	×	2018	79.9
Bhowmick et al.[17]	-	-	U-Net	CSSC	Fissure	SG	×	2020	61.2 (IoU)
Liu et al.[124]	VGG-16	-	Deep Hierarchical CNN	DeepCrack	Fissure	SG	×	2019	86.5
Zhang et al.[230]	DC-GAN	-	U-Net, AD	CFD, CGD, Dataset[5]	Fissure	SG	×	2021	91.3
König et al.[104]	ResNet 50 - 152, EfficientNet	-	Ws CNN	CRACK500, CFD, DCD	Fissure	SG	×	2022	90.5
Zhang et al.[229]	-	-	Réseau à trois flux	APD, APD04Crack, APD07Crack	Fissure	SG	×	2022	77.5
Zou et al.[238]	-	-	DCNN	CrackTree260, CRKWH100, CrackLS315, Stone331	Fissure	SG	×	2019	87.0
Zhang et al. [232]	-	VOC[45], SBD[71]	HRNet-W32	SF(TI: 145 & 6 classes)	MD	SG	×	2023	90.47
Kang et al.[91]	ResNet-50	COCO	Faster R-CNN	Jeu de données personnalisé (TI: 1200)	Fissure	SG	×	2020	83.0 (IoU)
Meng [133]	ResNet101	-	CrackNet	SF (1695 & 6 classes)	Fissure	SG	×	2021	96.5 (R)
Mei et al.[134]	-	-	CNN	CFD	Fissure	SG	×	2020	91.9
Shim et al.[174]	-	-	AD, SM	METU	Fissure	SG	×	2020	86.8
Wang and Su[197]	DenseNet121	ImageNet	CNN (AT)	Crack500, Deepcrack, Gaps384[218], Mixed Crack Dataset (MCD)	Fissure	SG	×	2020	69.1 (IoU)
Sun et al [181]	DeepLabv3+	-	DeepLabv3+	SF(TI: 16,662 & Fissure & trou d'insecte)	MD	SG	×	2021	81.8
Li et al. [113]	DenseNet-121	-	CNN	SF	MD	SG	×	2019	84.5 (IoU)
Kim and Cho [98]	ResNet-101	COCO	Mask R-CNN	SF(TI:2750 & 4 classes)	MD	SG	×	2020	87.5 (R)

Cette revue de la littérature établit la base pour comprendre le paysage actuel du SHM et l'impact transformateur du deep learning dans ce domaine. Elle met en lumière la lacune de recherche que cette thèse vise à combler et justifie les nouvelles méthodologies proposées dans les chapitres suivants, les situant dans le contexte plus large des avancées en cours dans ce domaine.

2.8.1 Conclusion du Chapitre

Ce chapitre a fourni une revue exhaustive de la littérature concernant les défauts du béton dans les ponts et l'application des techniques d'apprentissage profond à la classification d'images, à la détection d'objets et à la segmentation d'images dans ce domaine. La taxonomie des défauts explorée dans ce chapitre constitue la base pour comprendre les complexités du suivi de l'état des structures (*Structural Health Monitoring, SHM*). De plus, divers modèles d'apprentissage profond, allant des réseaux de neurones convolutifs (CNNs) aux modèles hybrides comme les systèmes basés sur l'attention et les réseaux adversariaux génératifs, ont été examinés dans le contexte de la détection et de la classification des défauts.

La discussion sur les approches classiques et modernes de l'apprentissage profond révèle les avancées significatives dans l'automatisation de l'analyse des défauts du béton. Cependant, des défis subsistent, notamment en ce qui concerne l'amélioration de la précision, la prise en compte des limitations de l'annotation des données et l'optimisation des performances des modèles pour des applications concrètes dans le monde réel. Ces défis ouvrent la voie aux méthodologies explorées dans le chapitre suivant, qui se concentrera sur l'évaluation comparative des techniques d'apprentissage profond les plus récentes pour détecter et classifier les défauts du béton de manière plus efficace.

Chapitre 3

Évaluation comparative de la détection et de la classification des défauts du béton

3.1 Introduction

Ce chapitre met l'accent sur la transition de la théorie à la pratique, en démontrant comment les techniques de deep learning sont exploitées pour améliorer la précision, l'efficacité et la fiabilité de la détection et de la classification des défauts du béton. À travers une exploration détaillée du développement, de l'entraînement et des processus d'évaluation des modèles, nous illustrons le processus complexe d'adaptation des modèles de deep learning aux exigences spécifiques de la surveillance de la santé structurelle. En présentant diverses architectures de modèles, des stratégies d'utilisation des jeux de données et des références de performance, ce chapitre vise à fournir une vue d'ensemble complète de l'état actuel des applications du deep learning pour la détection des défauts du béton et à souligner le potentiel des avancées futures dans ce domaine.

3.2 Métriques d'évaluation standard pour la classification, la détection et la segmentation

Il existe différentes métriques pour évaluer la classification, la détection basée sur les boîtes englobantes (BB) et la segmentation. La principale différence réside dans les

entités classifiées qui sont considérées. Étant donné que la classification est effectuée au niveau de l'image, les métriques de classification décriront la précision des catégories d'images prédites. Les métriques pour la détection décriront la précision de la localisation des défauts à l'aide soit de boîtes englobantes, soit de segmentation.

Pour simplifier la description des métriques existantes, nous pouvons considérer le cas binaire d'un défaut par rapport à l'arrière-plan; la version multi-classes des métriques peut être construite en moyennant les métriques binaires sur toutes les classes. Dans ce contexte, nous pouvons formuler les métriques en fonction des concepts de *vrais positifs* (TP), *faux positifs* (FP), *vrais négatifs* (TN) et *faux négatifs* (FN), respectivement. Notez que si nous avons N points de données, alors $N = TP + TN + FP + FN$. Les métriques suivantes peuvent être dérivées pour la classification et la détection (les points de données sont des images pour la classification, des boîtes englobantes (BB) pour la détection basée sur BB, et des pixels pour la segmentation). Chacune des métriques donne une perspective spécifique et nuancée sur la performance du modèle :

- *Précision:* $P = \frac{TP}{TP+FP}$
- *Rappel:* $R = \frac{TP}{TP+FN}$
- *Taux de vrais positifs (sensibilité):* $TPR = R$
- *Taux de faux positifs:* $FPR = \frac{FP}{FP+TN}$
- *Taux de vrais négatifs (spécificité):* $TNR = \frac{TN}{TN+FP}$
- *Taux de faux négatifs:* $FNR = \frac{FN}{FN+TP}$
- *Précision:* $Acc = \frac{TP+TN}{TP+TN+FP+FN}$
- *Précision équilibrée:* $BA = \frac{TPR+TNR}{2}$

Pour la détection basée sur BB, soit B_{gt} et B_p les BB de vérité terrain et prédites pour un défaut. La métrique suivante est cruciale pour définir d'autres métriques de détection :

- *Intersection over union:* $IoU_B = \frac{B_p \cap B_{gt}}{B_p \cup B_{gt}}$

Pour un cas multi-classes, on peut moyenner la valeur de IoU_B sur toutes les classes. En utilisant IoU_B, un défaut est un *vrai positif* (TP) lorsque $IoU_B \geq \tau$ et un *faux positif*

(FP) lorsque $IoU_B < \tau$, où τ est un seuil donné (typiquement $\tau = 0.5$). Un *faux négatif* FN signifie que le défaut n'a pas été détecté, tandis qu'un *vrai négatif* (TN) correspond à toute BB pour l'arrière-plan qui n'est pas détectée comme un défaut. Étant donné qu'il existe de nombreuses possibilités pour TN, il n'est souvent pas utilisé dans les métriques. À partir des valeurs TP, FP et FN, on peut calculer la *Précision* (P) et le *Rappel* (R), ainsi que les métriques suivantes :

- *Courbe Précision-Rappel:* est obtenue en faisant varier la valeur du *seuil* pour la métrique IoU_B.
- *Précision moyenne (AP):* est calculée en utilisant *l'aire sous la courbe* (AUC) de la *courbe Précision-Rappel*. Plus l'aire est grande, meilleure est la performance de la méthode pour la détection. En pratique, AP est la *Précision* moyennée sur toutes les valeurs de *Rappel* entre 0 et 1.
- *Précision moyenne (mAP):* est calculée en prenant la moyenne de AP sur toutes les K classes de défauts et/ou sur tous les seuils IoU_B:
- *Précision moyenne:* $mAP = \frac{1}{K} \sum_{k=1}^{K} AP_k$

Dans les scénarios d'entraînement typiques, la métrique mAP, pour les seuils dans la plage 0.5 : 0.95, est utilisée pour sélectionner le meilleur modèle sur l'ensemble de validation. Le seuil de 0.5 est généralement utilisé pour évaluer la méthode sur l'ensemble de test.

Dans la segmentation des défauts, la performance du modèle peut être évaluée par un ensemble de métriques spécialisées calculées au niveau des pixels, chacune fournissant des informations sur différents aspects de la précision de la segmentation :

- *Précision des pixels:* $PA = \frac{TP+TN}{TP+TN+FP+FN}$.
- *Indice de Jaccard:* $J = \frac{TP}{TP+FP+FN}$.
- *Coefficient de Dice:* $D = \frac{2 \times TP}{2TP+FP+FN}$.
- *Intersection sur union moyenne:* $mIoU = \frac{1}{K} \sum_{k=1}^{K} IoU_k$

Le *Coefficient de Dice* (D) et l'*Indice de Jaccard* (J) sont liés comme suit: $D = \frac{2J}{1+J}$. Il est à noter qu'il n'existe pas de métrique universellement applicable, mais qu'il faut plutôt utiliser un sous-ensemble des métriques, et aussi faire attention à l'interprétation

de leurs valeurs. Par exemple, un très haut PA accompagné de faibles valeurs de D et J peut se produire lorsqu'un défaut n'est pas bien segmenté, mais occupe une très petite partie de l'image par rapport à un arrière-plan dominant. Ainsi, le fond correctement classifié dominera tout le numérateur de PA, résultant en un score illégitimement élevé.

3.3 Évaluation comparative des modèles

Il existe des centaines de travaux de recherche sur la classification et la détection des défauts du béton. Cependant, pour la grande majorité d'entre eux, la mise en œuvre n'est pas disponible. De plus, les méthodes utilisent différents jeux de données et conditions expérimentales pour leur validation. Par conséquent, la comparaison directe des résultats obtenus par chaque méthode est impossible.

L'objectif de cette évaluation comparative est de fournir au lecteur une comparaison avec des conditions unifiées, équitables et égales pour les modèles de base dans la classification et la détection des défauts du béton. Ces modèles constituent les bases de la majorité des méthodes de classification et de détection des défauts du béton. Nous avons entraîné ces modèles par apprentissage par transfert et ajustement fin, en utilisant des jeux de données populaires de défauts du béton.

3.3.1 Jeux de données

Pour ce benchmarking, nous avons utilisé les jeux de données CODEBRIM, MCDS, CSSC, BCD, CDS et SDNET pour la classification des défauts. Nous rappelons que CODEBRIM et MCDS sont des jeux de données multi-étiquettes et que CSSC contient des défauts de fissuration et d'écaillage, tandis que BCD, CDS et SDNET traitent uniquement des fissures. Pour la détection des défauts, nous utilisons CODEBRIM qui possède des annotations appropriées sous forme de boîtes englobantes. Enfin, pour la segmentation des défauts, nous avons utilisé les jeux de données CrSpEE, LCW, BCL, DeepCrack et S2DS. Tous les modèles ont été pré-entraînés en utilisant le jeu de données ImageNet.

Table 3.1: Évaluation des modèles de classification des défauts.

			Binaire			**Multi-classes**	**Multi-étiquettes**		
Modèles	#Param	Profondeur	BCD	CDS	SDNET	CSSC	CODEBRIM	MCDS	CODEBRIM + MCDS
ResNet50	25.6M	107	0.99	0.95	0.96	0.99	0.96	0.93	0.95
EfficientNetV2L	119M	-	0.99	0.96	0.97	0.99	0.96	0.94	0.95
MobileNetV3Large	5.4M	263	0.99	0.93	0.96	0.99	0.94	0.90	0.93
DenseNet201	20.2M	402	0.99	0.95	0.96	0.99	0.96	0.94	0.95
InceptionV3	23.9M	189	0.99	0.89	0.95	0.99	0.94	0.89	0.92
Xception	22.9M	81	0.99	0.93	0.96	0.99	0.95	0.89	0.95
VGG19	143.7M	19	0.99	0.88	0.95	0.97	0.92	0.87	0.92
ViT (Transformer)	85.8M	12	0.99	0.83	0.75	0.98	0.91	0.82	0.91

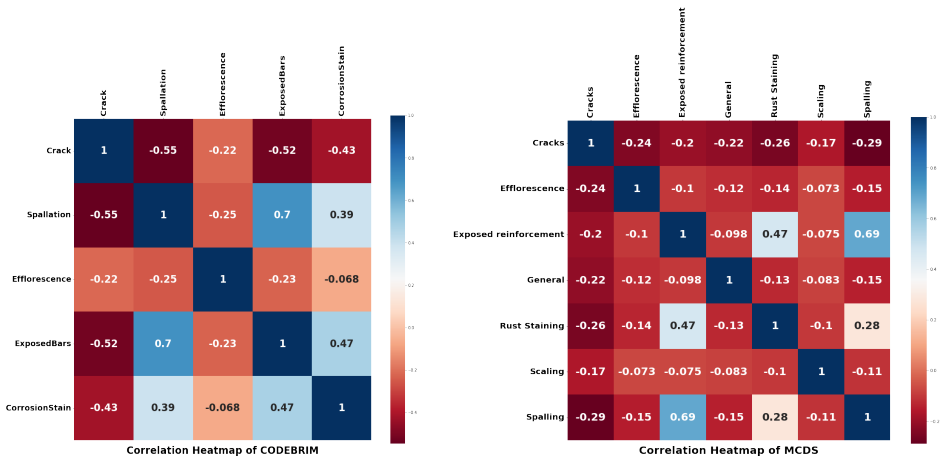
3.3.2 Classification des défauts de béton

Comparaison des modèles de base

Dans cette expérience, nous avons évalué des modèles de classification de défauts dans trois scénarios différents. Le premier scénario concerne la classification binaire pour des jeux de données traitant de la détection de défauts uniques par rapport à l'arrière-plan. Le deuxième scénario concerne la classification multi-classes où une image ne contient qu'une seule étiquette de défaut parmi plusieurs candidats. Le troisième scénario concerne la classification multi-étiquettes où une image peut contenir une ou plusieurs instances de défauts de différentes classes. Pour permettre des sorties multi-étiquettes, nous avons ajusté les réseaux pour prédire plus d'une étiquette en utilisant la fonction d'activation *Sigmoid* au lieu de *Softmax* dans les couches finales.

Nous avons comparé les modèles pré-entraînés suivants : ResNet50 [73], EfficientNetV2L [185], MobileNetV3Large [79], DenseNet201 [81], InceptionV3 [182], Xception [32] et VGG19 [176]. Ces modèles ont été choisis pour leurs diverses propriétés, telles que la taille, la profondeur et la capacité à traiter différentes échelles d'images. Ils ont également été largement utilisés dans la littérature sur la classification et la détection des défauts du béton. Chaque modèle a été pré-entraîné sur ImageNet et ajusté sur les données de détection des défauts du béton. Nous avons également entraîné le modèle Transformer visuel à partir de zéro.

Le tableau 3.1 présente la précision moyenne de classification obtenue sur les jeux de données CODEBRIM, MCDS, CSSC, BCD, CDS et SDNET. Tous les modèles CNN ont bien performé dans les trois scénarios, et les meilleures performances ont été obtenues par EfficientNetV2L et DenseNet201. EfficientNetV2L a la capacité de redimensionner la largeur du réseau, la profondeur et la résolution de l'image, tandis que DenseNet2021 permet des architectures plus profondes grâce à des connexions de saut denses. Ces



(a) (b)

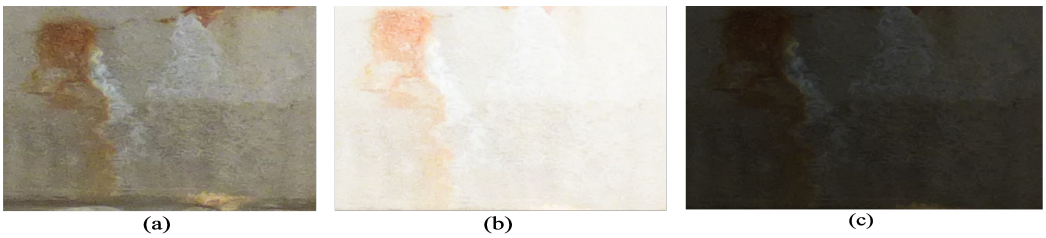
Figure 3.1: Cartes de corrélation des défauts dans les jeux de données CODEBRIM et MCDS.

propriétés permettent aux modèles d'extraire des caractéristiques de défaut plus riches. Notez que les performances les plus faibles ont été obtenues par VGG19, qui a une architecture plus superficielle par rapport aux autres modèles et au ViT. En effet, ViT nécessite généralement un grand nombre de données étiquetées pour assurer des performances équivalentes ou supérieures à celles des CNNs. La Figure 3.2 présente quelques exemples difficiles pour la classification des défauts du béton. Les première, deuxième et troisième lignes illustrent respectivement l'influence des variations d'éclairage, de la portée d'acquisition et du chevauchement des défauts sur la prédiction des défauts. Tous les défauts n'ont pas été bien prédits dans les sorties du modèle.

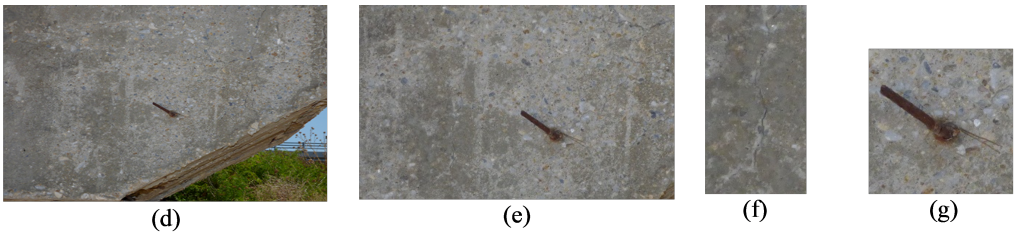
3.3.3 Détection des défauts de béton

Dans cette section, nous présentons les résultats de la détection des défauts de béton en utilisant certains des modèles les plus récents pour la détection d'objets. Ces modèles incluent Faster-RCNN [227], qui est une méthode de détection en deux étapes, et des méthodes de détection en une seule étape : YOLOv3 [88], SSD [88], RetinaNet [119], YOLOX [54] et YOLOR [198]. Nous avons également testé YOLOv7, YOLOv8 et Mask RCNN pour effectuer simultanément la détection et la segmentation d'instances.

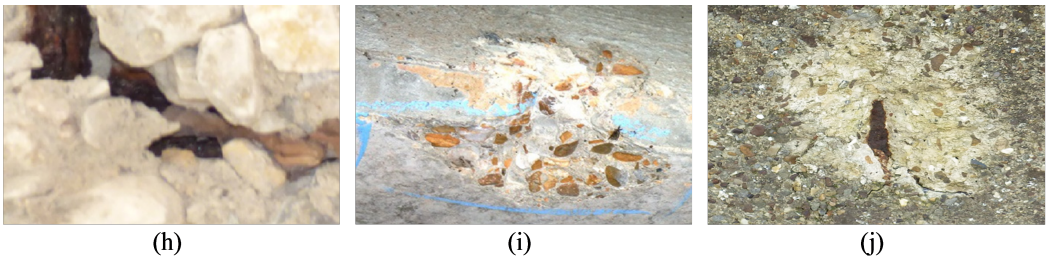
Le tableau 3.2 montre les performances des modèles. Nous présentons également les caractéristiques des modèles, y compris le type de tête de classification, la fonction de perte utilisée dans la détection, et le temps d'inférence. D'après les résultats, YOLOX,



(a) Image échantillon de CODEBRIM avec les étiquettes vraies : {*Fissure*, *Efflorescence*, *Tache de corrosion*}. (b) et (c) sont des versions de (a), avec un éclairage élevé et faible, respectivement. Les étiquettes prédites sont : (a) {*Fissure*, *Efflorescence*, *Tache de corrosion*}, (b) {*Fissure*, *Efflorescence*, *Barres exposées*} et (c) {*Efflorescence*}



(d) Image échantillon de CODEBRIM avec les étiquettes vraies : *Fissure*, *Barres exposées*, acquise à partir d'une vue de portée éloignée. (e) Les mêmes défauts acquis à partir d'une vue de portée moyenne. (f) et (g) sont des images recadrées de (e). Les étiquettes prédites sont : (d) {*Écaillage* et *Barres exposées*} (e) {*Écaillage*, *Barres exposées*}. (f) {*Fissure*} et (g) {*Barres exposées*}.



(h), (i) et (j) sont des images échantillons de MCDS avec des défauts qui se chevauchent avec les étiquettes vraies : (h) {*Écaillage*, *Tache de corrosion*}. (i) {*Écaillage*, *Barres exposées*}, (j) {*Écaillage*, *Barres exposées*, *Tache de corrosion*, *Fissures*}. Les étiquettes prédites sont : (h) {*Tache de corrosion*}, (i) {*Écaillage*, *Efflorescence*, *Barres exposées*, *Tache de corrosion*} (j) {*Écaillage*, *Barres exposées*, *Tache de corrosion*}.

Figure 3.2: Illustration des défis de classification des défauts. Les première, deuxième et troisième lignes montrent respectivement l'effet des changements d'éclairage, des différentes portées d'acquisition et du chevauchement des défauts sur la classification des défauts du béton.

Table 3.2: Comparaison des détecteurs de défauts les plus récents.

Méthode	**mAP @0.5(%)**	**Une étape**	**Deux étapes**	**Perte de classification**	**Perte de régression**	**Temps d'inférence (s)**
YOLOv3[164]	64.8	✓	-	Entropie croisée	L2	0.05
SSD [88]	64.1	✓	-	Entropie croisée	L1 lissée	0.06
YOLOv3 [227]	79.9	✓	-	Focale	L2	0.05
Faster R-CNN [159]	74.4	-	✓	Entropie croisée	L1 lissée	0.14
RetinaNet [119]	88.4	✓	-	Focale	L1 lissée	0.07
YOLOX [54]	91.8	✓	-	Entropie croisée binaire	L1 lissée	0.04
YOLOv5-l [58]	41.7	✓	-	Entropie croisée binaire	L1 lissée	0.02
YOLOv8-l [59]	59.6	✓	-	Entropie croisée binaire	L1 lissée	0.02
(SSD)[88]	64.1	✓	-	Entropie croisée	L1 lissée	0.06
YOLOR [198]	89.2	✓	-	Entropie croisée	L2	0.05

YOLOR et RetinaNet ont donné les meilleurs résultats, tandis que les performances les plus faibles ont été obtenues par YOLOv5. Il est à noter que, parmi les méthodes les plus récentes, YOLOv7 et YOLOv8 se distinguent par leur capacité à extraire des caractéristiques plus fines grâce à une conception optimisée des boîtes d'ancrage et à l'intégration de mécanismes d'attention avancés. Ces améliorations leur permettent d'obtenir de meilleurs résultats en détection, en particulier pour les images acquises à courte portée. Toutefois, dans le contexte de la segmentation, leurs performances restent limitées : ils génèrent souvent un nombre important de faux positifs et manquent certains défauts subtils, notamment lorsque les images présentent des arrière-plans simples ou sont capturées à moyenne ou longue distance. En effet, étant donné que la plupart des modèles sont entraînés sur des défauts acquis à courte portée, il leur est difficile d'acquérir des caractéristiques constantes à différentes échelles, ce qui peut entraîner des performances moins bonnes pour la détection de défauts à différentes distances. La Figure 3.3 montre quelques exemples difficiles pour la détection, où certains défauts sont complètement manqués par les modèles lorsque les images sont acquises à moyenne ou longue portée.

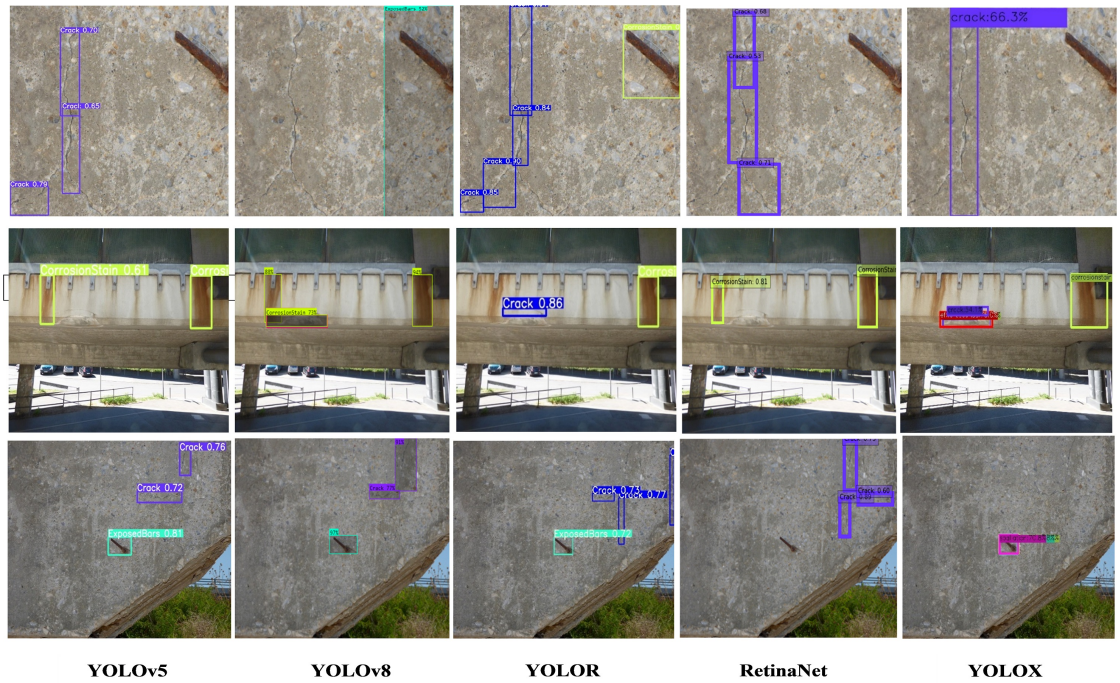


Figure 3.3: Exemples montrant des images difficiles pour la détection des défauts de béton [76].

Table 3.3: Comparaison des modèles de segmentation d'instances de défauts

Modèle	**Boîte mAP@0.5**	**Masque mAP@0.5**	**Temps d'inférence(s)**
YOLOv7	0.93	0.90	0.03
YOLOv8l	0.90	0.87	0.02
Mask RCNN	0.51	0.31	0.2

Table 3.4: Comparaison de l'Exactitude par Pixel des modèles de segmentation des défauts.

	Jeux de données		
Modèle	LCW	BCL	DeepCrack
U-Net	0.99	0.99	0.98
LinkNet	0.99	0.99	0.97
FPN	0.99	0.99	0.97
SegFormer	0.99	0.99	0.96

Table 3.5: Résultats de la perte et du Dice des modèles de segmentation sémantique des défauts sur le jeu de données S2DS

	Corrosion		**Efflorescence**		**Fissures**		**Écaillage**		**Temps d'inférence (s)**
Modèle	Perte	Dice	Perte	Dice	Perte	Dice	Perte	Dice	
U-Net	0.14	0.86	0.12	0.88	0.29	0.72	0.07	0.92	0.04
LinkNet	0.18	0.81	0.10	0.90	0.27	0.72	0.10	0.89	0.06
FPN	0.21	0.78	0.08	0.91	0.39	0.61	0.09	0.92	0.03
PSPNet	0.21	0.79	0.06	0.93	0.52	0.48	0.08	0.92	0.02

Enfin, pour la segmentation d'instances, nous avons comparé YOLOv7, YOLOv8 et Mask-RCNN sur le jeu de données CrSpEE. Le tableau 3.3 montre les résultats obtenus. Clairement, YOLOv7 et YOLOv8 ont obtenu les meilleurs résultats à la fois pour la détection et la segmentation. YOLOv7 a légèrement mieux performé que YOLOv8, tandis que Mask-RCNN a obtenu les résultats les moins bons. Nous avons également implémenté des modèles de base pour la segmentation de fissures dans les jeux de données LCW, BCL et DeepCrack, en utilisant les architectures UNet, LinkNet, FPN et SegFormer. Le tableau 3.4 présente les résultats obtenus, où le modèle UNet a donné les meilleurs résultats en général. La Figure 3.4 présente plusieurs exemples de résultats

(a) (b)

Figure 3.4: Illustration de quelques défis de segmentation des défauts : (a) et (b) montrent la segmentation obtenue par YOLOv7 et YOLOv8, respectivement.

de segmentation utilisant les modèles YOLOv7 et YOLOv8. On peut observer que les images caractérisées par un faible contraste, une prise de vue à longue distance, et de petits défauts dans des arrière-plans complexes posent particulièrement un défi pour les modèles, entraînant une segmentation imprécise. Notez que la segmentation des défauts de béton, tant pour les instances que pour les segments sémantiques, rencontre certains défis similaires.

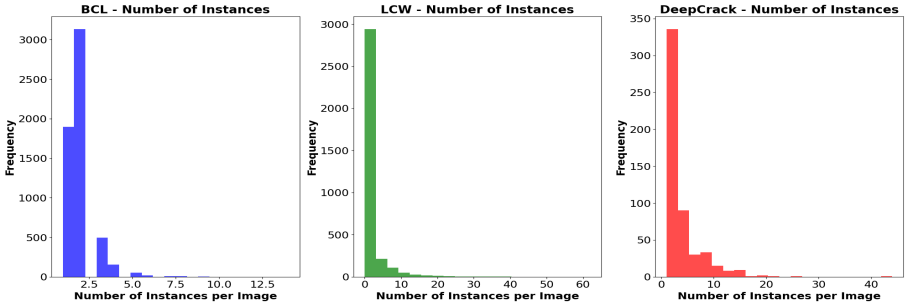


Figure 3.5: Histogrammes du nombre d'instances (masques) par image.

Enfin, identifier et séparer plusieurs instances dans une image par segmentation est un immense défi. Dans plusieurs cas, des instances du même type de défaut peuvent exister dans une image, ce qui rend difficile de les différencier. La Figure 3.5 montre les distributions des instances par image dans les jeux de données BCL, LCW, et Deep-

Crack. Le jeu de données BCL montre une concentration d'images avec un faible nombre d'instances, ce qui suggère des tâches de segmentation plus simples. LCW présente une distribution plus large, indiquant un niveau de complexité de segmentation modéré avec une longue traîne. En revanche, DeepCrack a une distribution plus en queue, reflétant un ensemble de données plus diversifié et potentiellement plus difficile pour la segmentation. Cette figure met en évidence la variance significative de la densité d'instances parmi les jeux de données, ce qui est essentiel pour informer le développement et l'évaluation des modèles de segmentation d'images.

3.3.4 Détection des fissures dans différents matériaux

Dans cette expérience, nous avons mené une analyse de la détection de fissures dans différents matériaux et structures liés aux ponts. Pour cela, nous avons utilisé le modèle de base YOLOv8, entraîné sur le jeu de données CODEBRIM pour la détection de fissures, puis appliqué à la détection de fissures sur route, mur et métal. Initialement, la performance du modèle sur CODEBRIM était modeste (57,2%). En affinant le modèle sur des données d'autres structures, il a montré une remarquable adaptabilité. L'amélioration peut être observée dans le tableau 3.6, où le modèle affiné a surpassé le modèle entraîné à partir de zéro. Nous avons observé que l'affinement non seulement augmentait la précision initiale, mais aussi accélérait la convergence vers la performance optimale, un contraste frappant avec la progression graduelle observée lors de l'entraînement du modèle à partir de zéro.

Pour approfondir notre étude, nous avons employé une approche d'apprentissage semi-supervisé en utilisant le modèle finement ajusté, entraîné sur CODEBRIM, pour générer des pseudo-étiquettes pour des jeux de données secondaires non étiquetés de fissures sur route, mur et métal. Les résultats présentés dans le tableau montrent la performance de l'approche semi-supervisée pour adapter le modèle entraîné dans le domaine source à d'autres matériaux et contextes structurels. Cette expérience met en évidence la polyvalence et l'adaptabilité des modèles pré-entraînés pour identifier et analyser l'intégrité structurelle à travers une diversité de matériaux et de structures.

Table 3.6: Détection des fissures dans différents matériaux

Matériau	**À partir de zéro (%)**	**Affiné(%)**	**Semi-supervisé(%)**
Fissures sur béton	57.2	–	–
Fissures sur métal	91.1	91.6	88.6
Fissures sur route	78.7	79.8	70.7
Fissures sur mur	74.2	75.7	72.1

3.4 Limitations et défis actuels

3.4.1 Limitations des jeux de données

Le manque de grands ensembles de données publics disponibles, avec des annotations appropriées, est une barrière importante pour le développement de modèles fiables pour la classification et la détection des défauts de béton. De plus, le processus d'annotation, en particulier pour la segmentation, est un processus très chronophage et laborieux, et il doit s'appuyer sur des connaissances expertes. En outre, d'autres problèmes majeurs peuvent rendre les modèles moins efficaces dans leur généralisation :

- **Acquisition & annotation des défauts** : Des conditions d'acquisition variables peuvent provoquer une grande hétérogénéité entre différents ensembles de données, et même au sein du même ensemble de données. Celles-ci incluent une illumination non uniforme, des points de vue de la caméra et des distances de prise de vue variables. De plus, en fonction de la structure, les images acquises peuvent inclure différents éléments d'arrière-plan et artefacts. Enfin, les ensembles de données peuvent varier en termes de granularité des défauts ciblés et de niveau d'annotation (par exemple, niveau pixel, niveau bloc ou niveau image), ce qui rend difficile leur fusion en ensembles de données plus grands. L'annotation des défauts, d'autre part, peut être bruyante ou incomplète en raison de plusieurs facteurs liés à la taille du défaut, au contraste, au chevauchement et à la fragmentation. Par exemple, des défauts petits, peu contrastés, chevauchés ou fragmentés peuvent être négligés, ou annotés différemment par les annotateurs. Ces problèmes peuvent envoyer des signaux persistants pendant l'entraînement du modèle, ce qui peut réduire la performance globale du modèle.

- **Ensembles de données déséquilibrés** : Certains défauts sont plus courants que d'autres, ce qui provoque un déséquilibre entre les classes de défauts. En

effet, certains ensembles de données peuvent montrer des distributions longues avec un déséquilibre sévère des classes, où il peut être difficile d'apprendre des caractéristiques représentatives pour les catégories de défauts sous-représentées. La rareté des classes de défauts peut concerner la présence de la classe de défaut au niveau de l'image (c'est-à-dire que peu d'images contiennent la classe de défaut dans l'ensemble de données), ou au niveau des pixels (c'est-à-dire que la classe de défaut occupe peu de pixels par rapport aux autres classes de défaut ou à l'arrière-plan). Cela peut poser un défi significatif pour prédire avec précision ces classes de défauts dans les modèles de classification ou de détection. L'augmentation de données et les fonctions de perte régularisées sont des moyens de mitiger les problèmes de classes déséquilibrées. L'utilisation de l'apprentissage semi-supervisé et de l'entraînement adversarial est une autre approche prometteuse pour atténuer ce problème, où l'exploitation de données non étiquetées abondantes avec quelques données étiquetées peut améliorer la performance du modèle pour les catégories rares.

3.4.2 Variabilité et chevauchement des défauts de béton

La variabilité des défauts de béton peut être causée par plusieurs facteurs tels que la taille, la forme, la couleur et la texture du défaut, qui dépendent de la conception du béton, de l'exposition aux conditions météorologiques et de la gravité du défaut. Par exemple, les fissures peuvent varier de fissures capillaires, à peine visibles, à des fissures larges pouvant compromettre l'intégrité de la structure. De plus, une seule instance de défaut peut se produire à plusieurs endroits proches (c'est-à-dire, fragmentation du défaut). Enfin, l'arrière-plan peut contenir des artefacts (par exemple, des peintures, des éléments structurels) qui peuvent être détectés comme des défauts. Par conséquent, les jeux de données d'images doivent contenir des données suffisamment diversifiées pour refléter la variabilité des défauts de béton.

Le chevauchement des défauts de béton, quant à lui, fait référence à la situation où plusieurs défauts de différentes catégories se produisent au même endroit. Cela peut se produire de manière aléatoire ou comme conséquence d'un effet cumulatif (par exemple, écaillage entraînant des barres d'armature exposées). La Figure 3.1 montre la propension à la cooccurrence des défauts au niveau de l'image sur les jeux de données CODEBRIM et MCDS. On peut noter que certaines classes de défauts comme les bar-

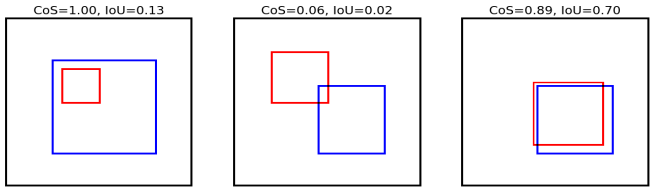


Figure 3.6: Différents degrés de chevauchement des défauts et les valeurs métriques associées CoS et IoU.

res exposées et l'oxydation présentent une forte corrélation. Enfin, notez que certains défauts chevauchants peuvent être très subtils à détecter même pour un modèle bien entraîné (voir la troisième rangée de la Figure 3.2 pour une illustration).

Pour montrer l'étendue du chevauchement des défauts au niveau des boîtes englobantes (BB), nous quantifions empiriquement la propension au chevauchement des défauts dans CODEBRIM. Pour cela, nous avons conçu la métrique *Covering overlap Score* (CoS), qui est donnée entre deux BB B_A et B_B de catégories différentes comme suit :

$$CoS = \frac{|B_A \cap B_B|}{\min(|B_A|, |B_B|)}$$

La métrique CoS est moins sensible à la taille du défaut que les métriques IoU, D ou J. Cela est illustré dans la Figure 3.6, où le premier exemple présente un chevauchement de défaut, mais la métrique IoU a donné un score très faible. La métrique CoS aborde ce problème en se concentrant sur la couverture relative par rapport à la plus petite BB. La Figure 3.7 montre la propension au chevauchement des défauts au niveau des boîtes englobantes (BB) dans le jeu de données CODEBRIM. Le graphique de gauche montre la log-fréquence en fonction des valeurs de la métrique CoS qui varient entre 0 et 1. Le graphique de droite montre dans chaque colonne la distribution de probabilité du chevauchement des défauts conditionnée à chaque catégorie différente. Pour calculer ces probabilités, nous supposons qu'une instance de chevauchement se produit lorsque CoS$>$ 0.75. Comme prévu, nous pouvons observer que l'un des chevauchements de BB de défauts les plus fréquents est entre *ExposedBars* et *Spalling*.

3.4.3 Limites des modèles

Les modèles proposés pour la classification et la détection des défauts sont généralement entraînés sur des images prises de près avec des défauts situés sur des arrière-plans ho-

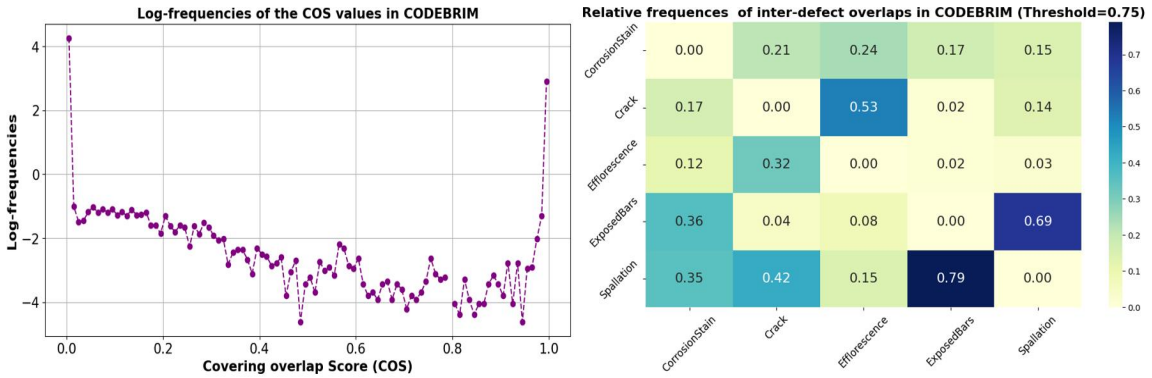


Figure 3.7: Analyse des chevauchements dans CODEBRIM ; Log-fréquences des valeurs COS et fréquences relatives des chevauchements inter-défauts avec un seuil de 0.75.

mogènes et propres. Cela suppose que les images sont prises à proximité de la surface du défaut ou que les données sont prétraitées au préalable (c'est-à-dire, recadrées, nettoyées des éléments structurels). Cependant, dans des scénarios réels (par exemple, lors d'une inspection par drone), les images peuvent être acquises à différentes distances, sous différents angles de vue, et peuvent inclure des éléments structurels de pont et des artefacts d'arrière-plan. Par conséquent, les modèles entraînés sur des images propres peuvent facilement échouer lorsqu'ils sont utilisés dans des systèmes d'inspection en conditions réelles.

Un deuxième problème concerne le phénomène de "dérive des données", où un modèle bien entraîné et validé se comporte mal lorsqu'il est confronté à des changements dans les données d'entrée au fil du temps. La dérive des données pour les défauts peut être liée à la saisonnalité, par exemple, où un modèle entraîné sur des données estivales peut produire de mauvaises prédictions de défauts lorsqu'il est déployé sur des données hivernales. Une solution pour atténuer ce problème est de continuer à affiner le modèle pour l'adapter aux nouvelles données. Cependant, cela peut entraîner une détérioration des performances du modèle sur les données d'origine, un phénomène connu sous le nom d'"oubli catastrophique".

Un troisième problème concerne la conception du modèle. La plupart des modèles proposés ne prennent pas en compte la variabilité et le chevauchement des défauts dans leurs prédictions. Les modèles de classification des défauts peuvent être très sensibles à la taille des défauts et à l'encombrement de l'arrière-plan. Par conséquent, inclure des mécanismes permettant de se concentrer sur les défauts plutôt que sur l'image entière peut permettre une meilleure classification. Pour le chevauchement des défauts,

les modèles peuvent être conçus pour permettre des prédictions multiples de défauts, ainsi que pour utiliser la corrélation entre les défauts pour renforcer les preuves dans la prédiction des défauts. Enfin, les modèles de classification et de détection doivent être déployés sur des systèmes informatiques embarqués montés sur des plateformes mobiles telles que les drones. Par conséquent, ils doivent être économes en mémoire et fonctionner en temps réel avec des délais et des charges de communication faibles. Cela pose le défi de concevoir des modèles qui soient à la fois légers, très précis et avec un temps d'inférence rapide.

3.4.4 Classification des défauts versus détection

Bien que les méthodes de classification et de détection des défauts de béton soient pertinentes pour évaluer les dommages des structures de ponts au niveau de l'image et au niveau local, elles doivent être appliquées avec soin pour exploiter tout leur potentiel. Par exemple, les méthodes de classification et de segmentation des défauts peuvent être inefficaces lorsqu'elles sont appliquées à des images avec de petits défauts enfouis dans des arrière-plans écrasants ou encombrés. D'autre part, l'annotation de la segmentation est une tâche très chronophage et coûteuse, tandis que les annotations de boîtes englobantes (BB) sont relativement faciles à obtenir. De plus, la détection des défauts peut être plus résiliente au bruit local et à la variabilité des défauts que la segmentation, qui agit au niveau des pixels. Par conséquent, la détection par BB devrait être la méthode initiale pour assurer une bonne (mais grossière) localisation des défauts. Elle peut également être appliquée conjointement avec la détection par BB via une supervision faible [104], pour améliorer la convergence et la précision de la segmentation des défauts [90]. Enfin, notez que le processus de détection des défauts de béton en général peut bénéficier de la segmentation des éléments structurels du pont [232], où la détection des défauts peut être plus ciblée sur les parties en béton. Par conséquent, les méthodes de classification et de détection des défauts (utilisant soit BB, soit la segmentation) doivent être appliquées de manière synergique dans toute méthodologie de développement de systèmes d'inspection de ponts.

3.4.5 Vers la résolution de ces défis

Les défis exposés dans ce chapitre soulignent les complexités impliquées dans la détection et la segmentation précises des défauts de béton dans des scénarios réels. Les

problèmes de variabilité des ensembles de données, de chevauchement des défauts, de limitations des modèles et des difficultés à gérer des arrière-plans complexes restent des problèmes ouverts dans le domaine de la surveillance de la santé des structures. Pour relever ces défis, le chapitre suivant présente une approche novatrice appelée CrackSight. CrackSight est conçu pour surmonter les limitations identifiées ici en intégrant des mécanismes d'attention avancés, des fonctions de perte innovantes et des architectures de modèles robustes à la variabilité et aux complexités rencontrées dans les applications pratiques. Plus précisément, CrackSight vise à améliorer la précision de la détection et de la segmentation des fissures dans des conditions diverses, y compris celles impliquant des arrière-plans complexes et encombrés, des conditions d'acquisition d'images variables et des défauts chevauchants.

En relevant ces problèmes ouverts, CrackSight représente une avancée significative dans le domaine, offrant des solutions pratiques qui améliorent l'efficacité des systèmes d'inspection automatisée pour maintenir l'intégrité et la sécurité des infrastructures en béton.

3.4.6 Conclusion du Chapitre

Ce chapitre a présenté une évaluation comparative approfondie des méthodes de détection et de classification des défauts du béton, en mettant en lumière les performances des différents modèles sur des jeux de données variés. Nous avons examiné les métriques d'évaluation standard, les défis liés aux données (variabilité, chevauchement et déséquilibre) et les limites des modèles actuels, notamment leur sensibilité aux conditions d'acquisition et aux arrière-plans complexes.

Ces analyses démontrent que, bien que les techniques actuelles offrent des performances prometteuses, des défis subsistent, notamment dans la généralisation sur des environnements réels et dans la gestion des faux positifs. Ces constats justifient le développement de nouvelles approches, comme CrackSight, qui sera présenté dans le chapitre suivant pour améliorer la précision et la robustesse des systèmes de détection et de segmentation des fissures dans des conditions variées.

Chapitre 4

CrackSight : Une nouvelle architecture U-Net pour la segmentation des fissures à différentes distances

4.1 Introduction

Les ponts en béton constituent une partie essentielle de notre infrastructure de transport, servant de liens vitaux dans le réseau de routes urbaines et interurbaines. Cependant, ces structures sont vulnérables à divers mécanismes de dégradation, les fissures étant l'un des plus importants. Le développement de fissures dans les ponts en béton peut être attribué à plusieurs facteurs, notamment la surcharge des véhicules, le vieillissement intrinsèque des matériaux de la chaussée et les charges de support extrêmes. Les fissures dans les ponts en béton, si elles ne sont pas détectées et traitées, peuvent compromettre l'intégrité structurelle, conduisant à des défaillances potentiellement catastrophiques. Traditionnellement, la détection et la surveillance de ces fissures reposaient sur des inspections manuelles, un processus rempli de défis tels que des problèmes d'accessibilité, des coûts de main-d'œuvre élevés et le risque inhérent d'erreurs humaines. Cela a conduit à la nécessité de méthodes de détection des fissures plus efficaces, précises et automatisées.

À l'intersection de la vision par ordinateur et de l'apprentissage automatique, la tâche d'identifier, de classifier et de segmenter les fissures au sein des infrastructures en béton apparaît comme un défi particulièrement ardu. Le béton, avec sa composition intrinsèquement hétérogène, défie la prévisibilité relative des motifs de fissures que l'on

trouve dans des matériaux comme les métaux ou les plastiques [43]. Cette thèse navigue dans le paysage complexe de l'analyse des fissures de béton, en explorant comment les propriétés variées du matériau, aggravées par les influences environnementales et les contraintes opérationnelles, amplifient la complexité à laquelle sont confrontés les modèles informatiques. Notre exploration vise non seulement à éclairer les obstacles techniques sophistiqués, mais aussi à introduire des approches novatrices qui promettent de raffiner la précision des systèmes de détection des fissures à différentes distances d'observation avec des arrière-plans complexes.

4.2 Technologies et capteurs pour la détection des fissures

En complément de l'utilisation des caméras pour l'inspection visuelle, de nombreuses technologies et capteurs non optiques sont employés pour la détection des fissures dans les structures en béton. Ces capteurs offrent souvent des informations complémentaires permettant d'identifier des défauts qui ne sont pas toujours visibles à l'œil nu ou dans des conditions d'éclairage défavorables. Parmi ces technologies, on retrouve notamment :

- **Capteurs ultrasoniques :** Ces capteurs mesurent la vitesse et l'atténuation des ondes sonores traversant le béton afin de détecter des fissures ou des délaminations internes. Cette technique est particulièrement adaptée pour identifier des défauts subsuperficiels et évaluer l'intégrité structurelle [165].
- **Caméras thermiques :** L'imagerie thermique permet de détecter des anomalies de température sur la surface du béton, lesquelles peuvent indiquer la présence de fissures ou d'autres défauts internes. Cette méthode est utile dans des conditions de faible éclairage ou lorsque des différences de température existent entre des zones endommagées et saines [86].
- **LiDAR et photogrammétrie :** Le LiDAR, grâce à sa capacité à fournir des relevés 3D de haute résolution, est utilisé pour détecter des variations de surface et des déformations structurelles qui peuvent être liées à des fissures. La photogrammétrie, qui combine plusieurs images prises sous différents angles, permet

également de reconstituer des modèles 3D pour une analyse plus précise des défauts [46].

– **Radar à pénétration de sol (GPR) :** Le GPR envoie des ondes électromagnétiques dans le béton et analyse leur réflexion afin de détecter des anomalies internes telles que des fissures, des cavités ou des délaminations. Cette technique est particulièrement efficace pour inspecter la structure interne des ponts et autres infrastructures critiques [131].

Ces technologies, souvent utilisées en complément des inspections visuelles, permettent d'obtenir une évaluation plus complète de l'état des structures. Elles offrent des perspectives nouvelles pour améliorer la fiabilité et l'efficacité des systèmes de surveillance de l'état des infrastructures.

4.3 Défis pour la segmentation des fissures

4.3.1 Complexité du matériau

La composition du béton - un mélange d'agrégats, de ciment et d'eau - entraîne une texture très variable qui complique la tâche de la détection automatisée des fissures. L'irrégularité des surfaces en béton, compliquée par des variables telles que la dégradation environnementale, les graffitis et d'autres anomalies de surface comme les efflorescences ou les taches d'oxydation, brouille considérablement la distinction entre les fissures réelles et les caractéristiques superficielles inoffensives. Cette variance exige des stratégies avancées de traitement d'images, nécessitant des techniques sophistiquées d'extraction de caractéristiques et de classification pour déchiffrer les complexités intrinsèques du matériau.

4.3.2 Acquisition et analyse des images

La qualité des données d'image joue un rôle crucial dans l'efficacité des efforts de détection des fissures dans le béton. Des facteurs tels que la résolution des images, l'éclairage ambiant, et la proximité et l'orientation du dispositif de capture introduisent une gamme de défis, en particulier lors de la surveillance de structures étendues comme les ponts, où des drones ou des outils d'imagerie à distance sont utilisés. Dans des scénarios réels, il n'est souvent pas faisable de s'approcher très près des ponts en raison de préoccupations

de sécurité et d'obstructions physiques. Cela nécessite l'utilisation de drones ou d'autres dispositifs d'imagerie à distance, qui capturent des images à des distances moyennes et longues.

La détection des fissures dans les images prises à des distances moyennes et longues introduit des défis supplémentaires. La variabilité de la texture et de la couleur de la surface, influencée par le vieillissement et les types d'agrégats, complique la distinction entre les fissures et les caractéristiques normales. La saleté et les graffitis peuvent encore plus obscurcir ou imiter les fissures, entraînant des détections erronées. La tâche de segmenter les fissures à partir de ces images, en particulier dans des conditions de lumière ou de météo fluctuantes, nécessite des algorithmes à la fois robustes sur le plan computationnel et capables de discerner les détails complexes des motifs de fissures. Ces défis sont exacerbés dans les images capturées à de plus grandes distances, où les conditions d'éclairage, les variations environnementales et les limitations de résolution du dispositif de capture peuvent avoir un impact significatif sur la qualité et la cohérence des données acquises.

4.3.3 Complexités de l'annotation des données et de l'entraînement des modèles

Le développement de modèles d'apprentissage automatique qui détectent avec précision les fissures dans le béton dépend de la disponibilité de grands ensembles de données annotées de manière méticuleuse. Cependant, le processus d'annotation manuelle des images de fissures est laborieux et subjectif, introduisant un degré de variabilité qui peut potentiellement compromettre l'entraînement des modèles. Le défi est exacerbé par la coexistence de plusieurs types de défauts et la possibilité que les fissures soient obscurcies par la peinture, les graffitis ou se fondent avec d'autres défauts, compliquant ainsi la capacité du modèle à identifier et classifier avec précision la grande variété de motifs de fissures.

En outre, le déséquilibre des données réelles et les pixels bruités entravent les algorithmes de détection. Les méthodes d'apprentissage profond, bien qu'efficaces, nécessitent une puissance de calcul considérable et des paramètres complexes, posant des défis pour le déploiement en temps réel sur des plateformes mobiles comme les UAVs. Cela devient encore plus prononcé dans les images prises à des distances moyennes et longues, où les défis de visibilité et de segmentation sont significativement accrus en raison de la

distance par rapport à la surface en béton et de la probabilité accrue de rencontrer des conditions environnementales diversifiées.

Lorsqu'on le compare à la tâche d'identifier des défauts dans des matériaux plus homogènes, l'effort pour détecter les fissures dans le béton révèle ses complexités uniques. La microstructure variable du béton et les diverses conditions environnementales auxquelles il est exposé nécessitent une approche plus nuancée et sophistiquée de la détection et de la classification des caractéristiques, dépassant la simplicité offerte par des matériaux plus uniformes. La Figure 4.1 illustre divers défis de visibilité et de segmentation des fissures dans le béton.

Figure 4.1: Illustration des défis de visibilité et de segmentation des fissures. La première rangée montre des fissures illustrant la nature hétérogène du béton, avec une texture non uniforme qui peut masquer les fissures. La deuxième rangée commence par une fissure très sévère (épaisse), suivie d'une fissure subtile, et montre ensuite une fissure différente sous trois conditions : normale, très lumineuse et faible luminosité. La troisième rangée représente des fissures avec des structures linéaires comme des joints, mettant en évidence le défi de distinguer les fissures des autres caractéristiques linéaires. La dernière rangée comprend des fissures obscurcies par de la saleté ou des graffitis, entraînant des faux positifs ou des détections manquées, et des fissures chevauchées par d'autres défauts.

4.3.4 Défis de déploiement

Bien que le développement d'algorithmes sophistiqués pour la détection des fissures dans le béton soit crucial, leur application réussie dans des scénarios réels dépend de la facilité de déploiement. Le déploiement englobe non seulement la capacité à implémenter ces algorithmes dans divers environnements, mais aussi l'efficacité opérationnelle et l'adaptabilité des modèles sur différentes plateformes, telles que les appareils mobiles, les UAVs (véhicules aériens sans pilote) ou les systèmes de calcul en périphérie (edge computing). L'un des principaux défis en matière de déploiement est les limitations matérielles des dispositifs utilisés sur le terrain. Par exemple, les UAVs et les appareils mobiles disposent souvent de ressources computationnelles limitées, d'une durée de vie de la batterie restreinte et d'une capacité de stockage réduite, ce qui restreint la complexité des algorithmes pouvant être exécutés en temps réel. Les algorithmes qui fonctionnent bien en laboratoire avec des ressources informatiques importantes peuvent devoir être optimisés pour la vitesse, la consommation d'énergie et l'utilisation de la mémoire lorsqu'ils sont déployés sur de telles plateformes.

Les conditions variées dans lesquelles ces algorithmes doivent opérer compliquent davantage le déploiement. Les modèles de détection des fissures doivent être suffisamment robustes pour gérer différents facteurs environnementaux tels que les variations de l'éclairage, les conditions météorologiques et l'état physique des surfaces en béton examinées. La capacité à s'adapter à ces conditions sans dégradation significative des performances est essentielle pour le déploiement pratique des systèmes de détection des fissures. Un autre aspect du déploiement est l'évolutivité du système de détection. Lorsqu'il est déployé sur des projets d'infrastructure à grande échelle, tels que des réseaux autoroutiers ou des systèmes de ponts étendus, la solution doit être capable de s'adapter sans perte d'efficacité ou de précision. Cela inclut la capacité de traiter de grandes quantités de données rapidement et le besoin potentiel d'une analyse en temps réel, ce qui peut être particulièrement difficile dans des emplacements éloignés ou difficiles d'accès.

Le déploiement exige également que les modèles se généralisent bien à différents ensembles de données et scénarios réels qui n'étaient pas présents lors de la phase d'entraînement. Cela inclut la capacité à détecter des fissures sur différents types de béton, dans diverses conditions environnementales et avec des qualités d'image variées. Un échec de la généralisation peut entraîner de mauvaises performances dans les applications réelles, où les conditions sont souvent moins contrôlées que dans le laboratoire. Enfin, un déploiement réussi nécessite une intégration soigneuse avec les systèmes de

gestion des infrastructures existants et un entretien continu. Cela inclut des mises à jour régulières des algorithmes de détection à mesure que de nouvelles données deviennent disponibles et la garantie que les systèmes restent opérationnels dans le temps. Le processus d'intégration lui-même peut être complexe, car il nécessite souvent une coordination entre différents systèmes techniques et parties prenantes.

Afin d'illustrer certaines des nouvelles méthodes et technologies prometteuses pour la surveillance de la santé structurelle (SHM) des ponts, la Figure 4.2 fournit une représentation graphique de Gkoumas et al. [57] présentant des approches innovantes dans ce domaine. De plus, le concept d'utilisation des UAVs pour l'inspection visuelle à distance des ponts est illustré à la Figure 4.3, tel que démontré par Lapointe et al. [108], mettant en lumière le potentiel de la technologie des drones pour améliorer les processus d'inspection.

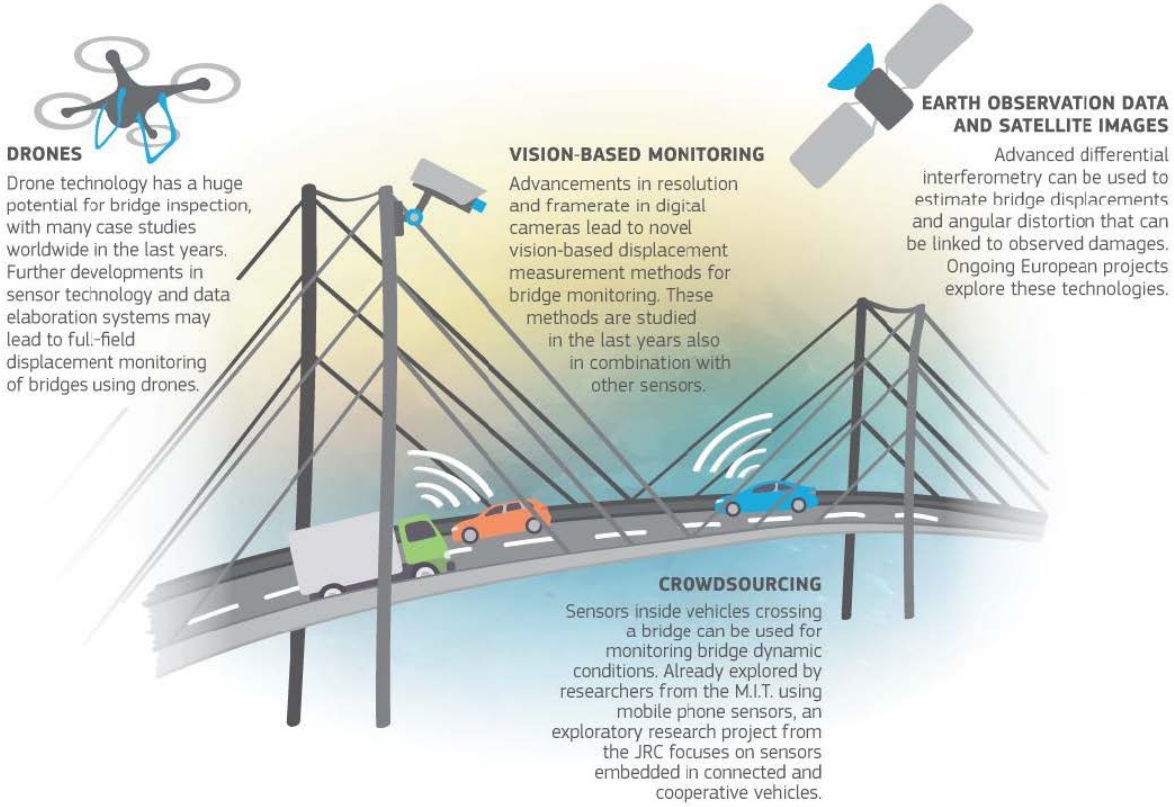


Figure 4.2: Nouvelles méthodes et technologies prometteuses pour la SHM des ponts [57].

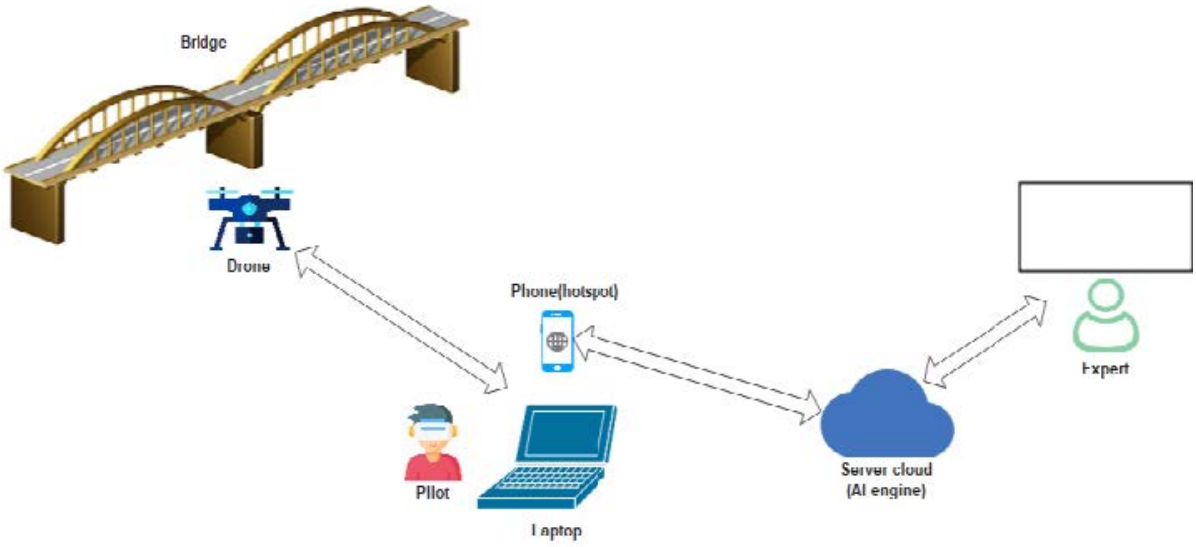


Figure 4.3: Concept d'utilisation des UAVs pour l'inspection des ponts (Hammouche et al., 2022).

4.4 Travaux connexes

Le domaine de la segmentation des fissures dans le béton a considérablement évolué, avec des avancées allant des techniques traditionnelles de traitement d'image aux modèles de pointe d'apprentissage profond. Cette section passe en revue l'évolution des méthodologies, l'impact de divers ensembles de données, et les approches innovantes dans ce domaine, fournissant un aperçu complet de l'état actuel de la recherche et identifiant les domaines à explorer davantage.

4.4.1 Techniques fondamentales de segmentation des fissures

Méthodes de traitement d'image préliminaires

Les premières techniques de détection des fissures dans le béton incluaient la détection des contours, le seuillage et l'analyse de texture, chacune avec des contributions séminales et des limitations. Yamaguchi et Hashimoto [214] ont proposé une méthode basée sur la percolation pour une détection rapide des fissures dans de grandes images de béton. Abdel-Qader et al. [1] ont analysé les techniques de détection des contours, trouvant la transformation rapide de Haar la plus fiable malgré des limitations dans les

arrière-plans texturés. Ayenu et Attoh [8] ont utilisé la décomposition modale empirique bidimensionnelle (BEMD) pour améliorer les performances du détecteur de contours de Sobel, réduisant le bruit par rapport aux méthodes traditionnelles comme le détecteur de Canny. Yamaguchi et al. [213] ont utilisé le seuillage adaptatif pour les surfaces en béton réelles mais ont rencontré des défis liés à l'éclairage et aux arrière-plans. Kirschke et Velinsky [102] ont développé une approche basée sur l'histogramme, efficace pour différentes largeurs de fissures mais sensible au bruit. Zou et al. [237] ont introduit CrackTree, combinant l'analyse de texture avec la reconnaissance des motifs pour une détection précise des fissures dans les chaussées. Petrou et al. [153] ont proposé une méthode basée sur la transformation de Walsh pour les surfaces texturées, efficace dans des environnements contrôlés mais moins dans ceux bruités. Subirats et al. [180] ont utilisé la détection par ondelettes pour améliorer les signatures des fissures, améliorant la précision malgré les variations de texture et d'éclairage. Ces méthodes préliminaires ont mis en évidence le besoin de techniques plus sophistiquées pour gérer les motifs de fissures complexes et diversifiés.

Approches basées sur l'apprentissage automatique

Les récents progrès dans la segmentation des fissures dans le béton ont vu un passage significatif de l'extraction manuelle des caractéristiques aux modèles d'apprentissage automatique, améliorant considérablement la précision de la segmentation. Shi et al. [172] ont proposé une méthode automatique de détection des fissures routières utilisant des forêts structurées aléatoires, démontrant une précision améliorée en exploitant les techniques d'apprentissage automatique pour analyser des motifs de fissures complexes. Cependant, cette méthode peut être intensive en calcul, la rendant moins adaptée aux applications en temps réel. Prasanna et al. [151] ont introduit un système automatisé de détection des fissures pour les ponts en béton utilisant des machines à vecteurs de support (SVM). Ce système a efficacement distingué les fissures des caractéristiques non fissurées, montrant des performances supérieures par rapport aux méthodes traditionnelles, mais nécessitant une grande quantité de données d'entraînement étiquetées pour atteindre une haute précision. Oliveira et Correia [145] ont développé un classificateur bayésien supervisé pour la détection automatique des fissures sur les chaussées, mettant en avant les performances et la précision améliorées des techniques d'apprentissage automatique par rapport aux approches traditionnelles. Néanmoins, cette approche a eu du mal à gérer les conditions d'éclairage variables et a nécessité un prétraitement

intensif des images. Cheng et al. [31] ont utilisé des réseaux neuronaux avec des caractéristiques statistiques pour la sélection de seuils dans la détection des fissures sur les chaussées, obtenant une haute précision dans la classification en entraînant le réseau neuronal à identifier les fissures en se basant sur les valeurs moyennes et de déviation standard des images. Bien que cette méthode soit efficace, elle a rencontré des défis pour traiter les images bruitées et a nécessité des ressources computationnelles importantes pour l'entraînement. Ces études démontrent l'évolution de l'extraction manuelle des caractéristiques vers les modèles d'apprentissage automatique, montrant des améliorations significatives dans la précision et l'efficacité de la segmentation des fissures dans le béton, tout en soulignant la nécessité continue de résoudre les problèmes liés à la complexité computationnelle, aux exigences de données et à la robustesse face aux variations environnementales.

4.4.2 Avancées dans l'apprentissage profond

Réseaux de neurones convolutifs (CNNs)

Qu et al. [155] ont proposé un modèle de réseau VGG16 amélioré pour la détection des fissures sur les pavés en béton. L'amélioration comprenait la modification des couches entièrement connectées et l'ajout de couches de convolution avec des tailles de noyau de 1x1 et 3x3 pour améliorer la précision de détection. Malgré des améliorations significatives des scores F1, le modèle a rencontré des difficultés à détecter avec précision des fissures très fines et dans des arrière-plans complexes. Li et Zhao [115] ont introduit un réseau d'encodage-décodage convolutionnel (CedNet) pour la détection et la mesure automatique des fissures dans les structures en béton. Le CedNet, construit sur DenseNet-121, a atteint une haute précision, une haute sensibilité et un haut rappel, mais a eu du mal à détecter les fissures fines et a nécessité un post-traitement intensif pour la mesure de la largeur et de l'orientation des fissures. Wang et al. [197] ont développé un modèle de segmentation des fissures de pavé basé sur un réseau pyramidal d'attention. Le modèle intégrait DenseNet121 en tant qu'encodeur et un module d'attention pyramidal des caractéristiques, améliorant significativement l'extraction des caractéristiques. Cependant, les performances du modèle pouvaient être impactées par des conditions d'éclairage et environnementales variables, nécessitant des améliorations supplémentaires en termes de robustesse. Li et al. [118] ont proposé une méthode pour la segmentation des fissures fines dans des images de ponts en acier à grande échelle

en utilisant le FCS-Net, qui combine ResNet-50 et un réseau entièrement convolutionnel (FCN). Ils ont introduit la normalisation par lot (Batch Normalization, BN) et la pyramide spatiale de regroupement par trous (ASPP) pour gérer des arrière-plans complexes et des premiers plans fins. Le modèle a atteint une intersection sur union moyenne (mIoU) de 0,7408, surpassant d'autres benchmarks comme LinkNet, DeepLab V3, et CrackSegNet. Cette étude a mis en évidence l'efficacité de l'intégration de BN et d'ASPP pour une meilleure extraction des caractéristiques et une meilleure précision de segmentation. Ces études soulignent collectivement les avancées et les limites des premières architectures CNN dans la segmentation des fissures.

Architectures combinées

Noh et al. [142] ont proposé une méthode innovante pour la détection automatique des fissures sur des images de béton en utilisant la segmentation via le clustering flou C-means. Cette approche a efficacement abordé le bruit et l'encombrement, mais a eu du mal à segmenter précisément les fissures lorsque les images contenaient un bruit important et des conditions d'éclairage variées. Qu et al. [156] ont introduit un algorithme de détection des fissures pour le pavé en béton basé sur un mécanisme d'attention et une fusion multi-caractéristiques. La méthode utilisait des modules Res2Net avec des mécanismes d'attention et des convolutions dilatées en mode cascade et parallèle pour améliorer l'extraction des caractéristiques. Bien qu'efficace pour traiter diverses fissures et des arrière-plans complexes, cela augmentait les besoins computationnels. Qu et al. [157] ont présenté une architecture de fusion de caractéristiques hiérarchique et d'attention connectée pour la détection des fissures de pavé. Ce modèle tirait parti des mécanismes d'attention et de l'extraction des caractéristiques multi-échelle pour améliorer la détection dans des conditions variées. Cependant, la complexité du modèle augmentait les besoins computationnels, rendant les applications en temps réel difficiles. Jin et al. [89] ont proposé une méthode utilisant des réseaux adverses génératifs (GANs) pour établir un ensemble de données d'images de fissures synthétisées pour l'entraînement des réseaux neuronaux profonds (DNNs). Cette approche améliorait la diversité et la richesse des données d'entraînement mais nécessitait des ressources computationnelles importantes et un réglage minutieux pour éviter le surapprentissage. Guo et al. [62] ont développé un réseau de neurones large en cascade pour la classification automatisée des dommages par fissures dans les structures en béton. Cette approche combinait les forces des réseaux de neurones larges en cascade, améliorant la précision de la classification mais nécessi-

tant des données d'entraînement importantes et des ressources computationnelles. Ces études mettent en évidence les récentes percées dans la détection des fissures, soulignant les compromis entre la précision, l'efficacité computationnelle et les exigences en matière de données.

4.4.3 Classifications centrées sur les ensembles de données

Ensembles de données publics standard

Les ensembles de données annotés sont cruciaux pour l'entraînement et l'évaluation des performances des modèles d'apprentissage automatique. Pour la plupart des méthodes proposées de classification et de détection des défauts, les auteurs utilisent souvent leurs propres ensembles de données, qui sont souvent prétraités pour éliminer les éléments structurels. Récemment, un certain nombre d'ensembles de données publics ont été proposés pour la détection et la classification des défauts du béton. Dans nos travaux précédents, nous avons fourni un aperçu détaillé de plusieurs ensembles de données notables, y compris DeepCrack [125], Ren'dataset [160], CrSpEE [10], BCL [224], LCW [19], et C2DS [12]. Ces ensembles de données présentent principalement des images à courte distance et bien éclairées, capturant une variété de types de fissures dans des conditions contrôlées. Cependant, dans les scénarios réels, tels que lorsque des drones ou d'autres véhicules d'inspection sont utilisés pour surveiller des ponts ou de grandes infrastructures, il est souvent impossible de s'approcher des structures de près. Par conséquent, les images prises dans ces conditions peuvent varier considérablement en termes d'éclairage et de distance, posant des défis supplémentaires pour une détection précise des fissures. Tirer parti des images capturées par drones peut relever ces défis en fournissant des données haute résolution et à longue portée, renforçant ainsi la robustesse et l'applicabilité des modèles de détection des fissures à des environnements plus réalistes et diversifiés. Cette intégration est vitale pour le développement de modèles capables d'effectuer des inspections en temps réel et à grande échelle de manière efficace.

Ensembles de données personnalisés et spécialisés

Les ensembles de données personnalisés sont spécifiquement conçus pour capturer des types de fissures uniques et des conditions environnementales spécifiques, comblant les lacunes laissées par les ensembles de données publics standard. Par exemple, Guo et al. [61] ont créé un ensemble de données dérivé de SDNET2018, qui comprend 230 images

de surfaces en béton fissurées et non fissurées segmentées en 56 092 sous-images. Cet ensemble de données a été enrichi par des techniques d'augmentation de données pour créer deux ensembles de données, chacun contenant 10 000 images. Un autre ensemble de données notable se concentre sur les fissures de barrages sous-marins capturées par des caméras robotiques sous-marines, capturant des images haute résolution qui ont ensuite été augmentées pour accroître la taille et la diversité de l'ensemble de données [173]. De plus, l'ensemble de données de Kim et al. [100] comprend des images haute résolution capturées à l'aide d'un smartphone sur la surface d'un pont, manuellement divisées en images plus petites pour un entraînement efficace. Bien que ces ensembles de données haute résolution offrent des avantages significatifs en garantissant que les modèles formés sur eux sont plus robustes et capables de détecter avec précision les fissures dans des conditions variées, ils se concentrent principalement sur des images à courte distance. Cette limitation entrave leur applicabilité dans les scénarios nécessitant une imagerie à longue portée, telle que les inspections basées sur des drones de grandes infrastructures. Par conséquent, le développement d'ensembles de données personnalisés incluant des images à longue portée est crucial pour faire progresser les technologies de détection des fissures. Cette lacune met en évidence le besoin de recherches et de développement d'ensembles de données supplémentaires pour capturer des images de fissures à longue portée, permettant ainsi des modèles de détection des fissures plus efficaces et complets dans les applications réelles.

Ensembles de données synthétiques et augmentés

L'utilisation de données synthétiques et de techniques d'augmentation est cruciale pour améliorer la robustesse des modèles de détection des fissures. Des techniques comme les réseaux adverses génératifs (GANs) ont été utilisées pour créer des ensembles de données synthétiques, qui simulent divers types de fissures et conditions souvent sous-représentées dans les ensembles de données réels. Par exemple, Jin et al. [89] ont utilisé une approche basée sur les GANs pour générer des images de fissures synthétisées et des annotations correspondantes, démontrant l'efficacité de ces images pour l'entraînement des réseaux neuronaux profonds (DNNs) à la détection des fissures. Les méthodes d'augmentation des données, telles que l'augmentation spatiale adaptative, enrichissent encore ces ensembles de données en introduisant des variations dans la direction et l'épaisseur des fissures, abordant ainsi les limites des ensembles de données réels [100]. Les évaluations ont montré que les modèles entraînés sur des ensembles de données augmentés offrent

des performances comparables à ceux entraînés sur des ensembles de données réels, réduisant significativement les taux de fausses détections et améliorant la précision des modèles. Cependant, ces ensembles de données synthétiques et augmentés se concentrent principalement sur des conditions de courte portée et bien contrôlées. Ils ne parviennent souvent pas à reproduire la complexité et la variabilité des images réelles à longue portée capturées par des drones ou d'autres véhicules d'inspection. Cette lacune limite la généralisabilité des modèles formés sur ces ensembles de données à des applications pratiques impliquant des inspections d'infrastructures à grande échelle. Par conséquent, l'intégration de données synthétiques à longue portée et de techniques d'augmentation avancées pour simuler plus précisément les conditions réelles améliorerait considérablement la robustesse et l'applicabilité des modèles de détection des fissures dans des environnements diversifiés.

4.4.4 Autres stratégies d'apprentissage dans la segmentation des fissures

Apprentissage semi-supervisé

Les récents progrès de l'apprentissage semi-supervisé pour la segmentation des fissures se sont largement concentrés sur l'amélioration de l'adaptabilité à des conditions environnementales variées telles que l'éclairage, les conditions météorologiques et la texture. Shim et al. [174] et Mohammed et al. [137] utilisent des mélanges de données étiquetées et non étiquetées, démontrant des améliorations notables de la précision de la segmentation. Shim et al. ont atteint une intersection sur union moyenne de 88,936 % avec leur méthode basée sur l'apprentissage multi-échelle par adversaires, tandis que Mohammed et al. ont réduit les temps d'entraînement et amélioré la précision de 20 % en utilisant une architecture U-Net modifiée. Li et al. [116] et Xiang et al. [205] contribuent également en intégrant l'apprentissage par adversaires pour exploiter efficacement les images non étiquetées, Li et al. atteignant une précision de détection de 95,91 % en n'utilisant que la moitié des données étiquetées. Ces méthodes soulignent le potentiel de réduire la charge d'annotation et d'augmenter la robustesse des modèles dans des conditions variées. Cependant, elles soulignent également la dépendance critique à la qualité des données non étiquetées et les complexités de l'apprentissage par adversaires, ce qui pourrait limiter l'application pratique sans une intégration minutieuse et une validation des données dans les régimes d'entraînement. Ces défis mettent en évidence la nécessité de

trouver un équilibre dans l'entraînement des modèles et le risque de surapprentissage, qui doit être géré pour exploiter pleinement les avantages de l'apprentissage semi-supervisé dans les applications réelles.

Apprentissage avec peu d'exemples (Few-shot learning)

Les techniques d'apprentissage avec peu d'exemples sont cruciales pour entraîner des modèles de segmentation des fissures avec des données étiquetées limitées, offrant une solution aux exigences de labellisation souvent étendues associées aux applications d'apprentissage profond. Katsamenis et al. [92] ont utilisé un U-Net résiduel récurrent amélioré avec des mécanismes d'attention, démontrant la capacité du modèle à s'adapter rapidement avec peu de données grâce à un réentraînement avec peu d'exemples. Liu et al. [129] ont introduit MCSNet, qui exploite une nouvelle méthode de fusion des caractéristiques pour généraliser efficacement sur divers types de fissures dans des environnements multi-scènes avec peu d'exemples. De même, Yao et al. [223] ont développé CrackNex, un modèle qui utilise la théorie de Retinex pour une segmentation robuste des fissures en basse lumière, démontrant une adaptabilité significative dans des conditions d'éclairage variées. Ces approches soulignent le potentiel de l'apprentissage avec peu d'exemples pour réduire les besoins en données et améliorer la polyvalence des modèles, bien que la qualité et la représentativité des exemples limités restent cruciales pour leur succès.

Fusion de données multimodales pour la segmentation

La fusion de données multimodales, intégrant des informations visuelles, thermiques et de profondeur, améliore significativement la précision de la segmentation des fissures et l'adaptabilité dans des conditions variables. Li et al. [117] ont développé un réseau combinant convolution résiduelle récurrente avec des encodeurs de contexte, améliorant la précision de la détection et la robustesse dans des environnements complexes avec une précision et un mIoU de 98,62 % et 80,93 %, respectivement. Xu et al. [209] ont utilisé une fusion des données visuelles et de profondeur pour aborder la segmentation dans des conditions de faible luminosité pour les fissures des caissons en acier, tandis que Yang et al. [219] ont amélioré la segmentation des fissures en béton grâce à des images capturées par des drones combinées à l'informatique en périphérie, abordant efficacement les variabilités environnementales. Ces approches soulignent les avantages de la fusion multimodale pour surmonter les limitations des seules données visuelles,

bien qu'elles introduisent également des complexités dans le traitement des données et le développement des algorithmes.

4.5 Méthode Proposée

Nous avons introduit une nouvelle approche pour la détection et la segmentation des fissures qui intègre l'architecture U-Net avec un Mécanisme de Focalisation Linéaire à Double Attention (DALFM) amélioré par des cartes de saillance. Notre méthode combine de manière unique des mécanismes d'attention spatiale et par canal avec des améliorations basées sur la saillance dans les deux domaines pour améliorer la précision de segmentation à différentes distances d'observation. L'architecture proposée, nommée Réseau de Détection de Fissures Basé sur les Caractéristiques Linéaires (LFB-Net), fonctionne comme un détecteur de fissures qui emploie une combinaison de cartes de caractéristiques et de mécanismes d'attention pour aborder la complexité de la détection des fissures dans les structures en béton. Les principales innovations incluent le DALFM, qui améliore la pertinence des caractéristiques ; une couche de saillance qui met en évidence les régions critiques dans les dimensions spatiales et par canal ; la convolution atrous pour une information contextuelle multi-échelle ; une Perte d'Entropie Croisée Binaire Pondérée Contextuellement pour aborder le déséquilibre des classes et améliorer les performances ; et le LFB-Net pour gérer efficacement les échelles d'image variées. De plus, notre méthode introduit une pipeline intégrée de détection et de segmentation qui permet un traitement fluide, excelle dans la segmentation d'images avec des arrière-plans complexes, et est conçue comme un modèle léger adapté au déploiement dans des environnements à ressources limitées. Cette approche globale aborde les défis inhérents à la détection des fissures, y compris la variabilité des textures de surface, les conditions d'éclairage, et le besoin d'une segmentation précise à différentes distances, améliorant significativement la précision et la fiabilité des systèmes de détection et de segmentation des fissures. L'architecture globale de la méthode proposée est illustrée à la Figure 4.4.

L'épine dorsale de notre modèle est l'architecture U-Net, reconnue pour son succès dans la segmentation d'images médicales. CrackSight comprend une structure encodeur-décodeur avec des connexions de saut qui préservent l'information spatiale à travers les couches. L'encodeur réduit progressivement l'échantillonnage de l'image d'entrée pour extraire des caractéristiques de haut niveau, tandis que le décodeur rééchantillonne ces caractéristiques pour reconstruire la sortie segmentée.

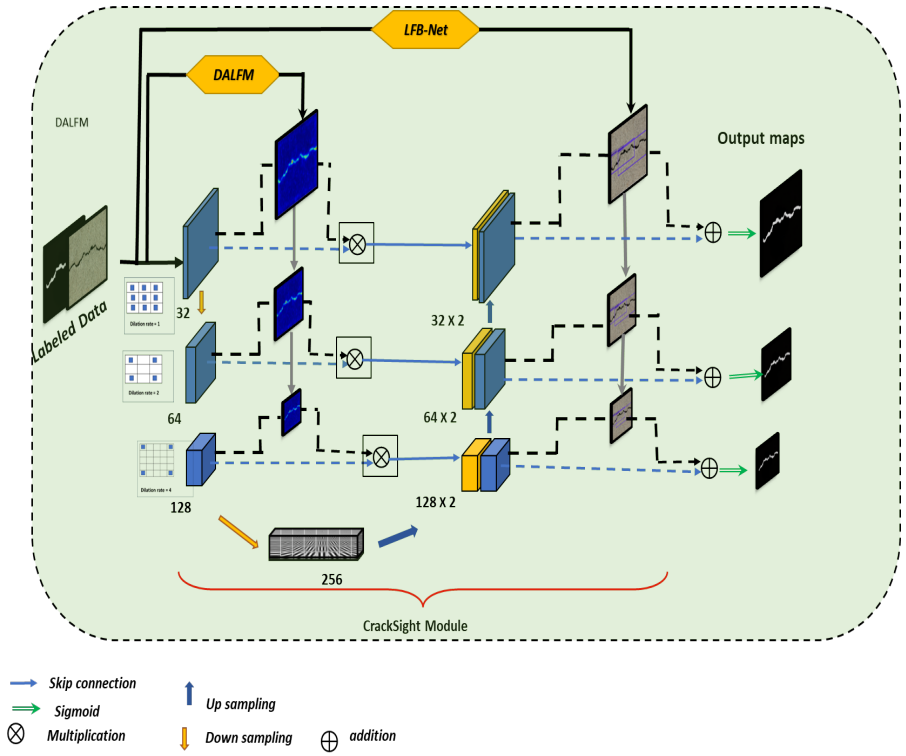
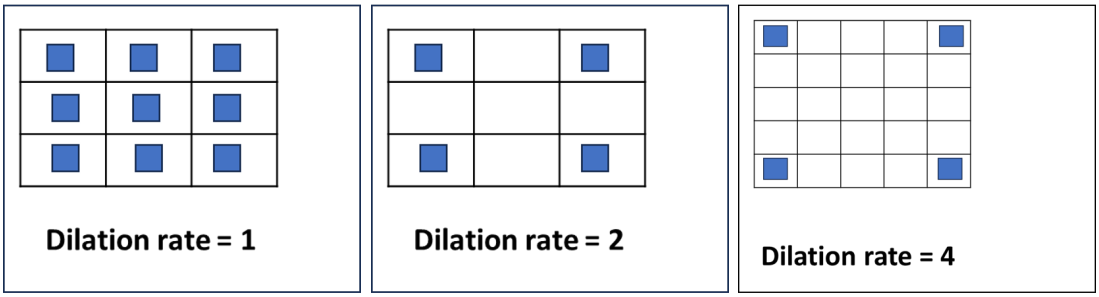


Figure 4.4: Architecture proposée de CrackSight

Pour capturer des informations contextuelles multi-échelles, nous employons des convolutions atrous, également connues sous le nom de convolutions dilatées, dans l'encodeur. Plus précisément, nous avons appliqué des convolutions atrous avec différents taux de dilatation à divers niveaux de l'encodeur. Dans le premier bloc de l'encodeur, nous avons utilisé un taux de dilatation de 1, permettant à la convolution de se comporter comme une convolution standard. Dans le deuxième bloc, nous avons appliqué un taux de dilatation de 2, élargissant le champ réceptif tout en maintenant la résolution spatiale. Enfin, dans le bloc central de l'encodeur, nous avons utilisé un taux de dilatation de 4, ce qui a encore élargi le champ réceptif. Ces taux de dilatation variés permettent à notre modèle d'agréger des informations multi-échelles, améliorant sa capacité à détecter des motifs de fissures subtils et complexes dans des conditions d'éclairage variables et des arrière-plans divers.

L'effet de ces taux de dilatation est illustré à la Figure 4.5, où une série d'images montre visuellement comment différents taux de dilatation affectent le champ réceptif. Dans la première image (a), représentant un taux de dilatation de 1, toutes les parties de l'image sont remplies, démontrant une convolution standard où chaque pixel est pris

en compte. La deuxième image (b) illustre un taux de dilatation de 2, où le motif de remplissage saute une étape, élargissant effectivement le champ réceptif. La troisième image (c) représente un taux de dilatation de 4, le processus de remplissage sautant trois étapes, ce qui entraîne un champ réceptif encore plus large. Ces images aident à comprendre comment l'augmentation du taux de dilatation permet au modèle de capturer un contexte plus large sans sacrifier le détail spatial, améliorant ainsi sa capacité à identifier des caractéristiques complexes et subtiles.



(a) (b) (c)

Figure 4.5: Illustration des différents taux de dilatation à l'aide d'une série d'images. (a) montre une convolution standard avec un taux de dilatation de 1. (b) représente un taux de dilatation de 2, avec une étape sautée. (c) montre un taux de dilatation de 4, avec trois étapes sautées.

Dans la partie encodeur de notre CrackSight, les couches de convolution initiales utilisent un faible taux de dilatation, capturant des détails fins. Au fur et à mesure que nous avançons plus profondément dans le réseau, les taux de dilatation augmentent, permettant au modèle de capturer des informations contextuelles plus larges. Cette stratégie permet à notre réseau de maintenir un équilibre entre la capture des détails fins et la compréhension du contexte structurel global, essentiel pour une détection précise des fissures.

Notre architecture CrackSight démontre une efficacité significative en termes de paramètres par rapport à l'U-Net original. Alors que l'U-Net original comprend 31 031 810 paramètres, notre modèle n'en inclut que 283 483. Cette réduction drastique des paramètres est réalisée sans compromettre les performances du modèle. En intégrant des mécanismes d'attention avancés, y compris le Mécanisme de Focalisation Linéaire à Double Attention (DALFM) et des couches de saillance, notre modèle concentre les ressources computationnelles sur les caractéristiques les plus critiques, améliorant ainsi

la précision de la segmentation des fissures. La gestion efficace des paramètres rend CrackSight adapté au déploiement dans des environnements à ressources limitées.

La structure de CrackSight intègre le DALFM, qui combine des mécanismes d'attention spatiale et par canal pour améliorer la pertinence des caractéristiques, et une couche de saillance pour mettre en évidence les régions critiques. L'utilisation de la Perte d'Entropie Croisée Binaire Pondérée Contextuellement aborde le déséquilibre des classes et améliore les performances, tandis que le Détecteur de Défauts Multi-étiquettes Basé sur la Saillance (SMDD-Net) gère efficacement les échelles d'image variées. Le nombre de paramètres de ce modèle est significativement réduit grâce à l'utilisation optimisée des ressources, tandis que la combinaison de CBAM et des cartes de saillance améliore encore la capacité du modèle à détecter des fissures fines et subtiles.

4.5.1 Augmentation de données

L'augmentation de données est une technique cruciale utilisée pour augmenter la diversité de l'ensemble de données d'entraînement sans réellement collecter de nouvelles données. En appliquant une série de transformations, nous pouvons simuler des variations dans les conditions réelles, améliorant ainsi la capacité du modèle à généraliser des données d'entraînement aux données non vues. Dans cette thèse, nous avons utilisé des techniques d'augmentation de données pour améliorer la robustesse et la précision des modèles de segmentation des fissures, en mettant l'accent sur la préservation des détails des fissures.

1. ***Rotation :*** Les images peuvent être tournées selon des angles θ, où $\theta \in \{90°, 180°, 270°\}$. Cette technique simule différentes orientations des fissures, en tenant compte de leurs apparences directionnelles arbitraires. La représentation mathématique de la rotation est :

$$R_\theta(x, y) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

   Cependant, bien que des angles de rotation extrêmes tels que 90°, 180° et 270° augmentent la variabilité, ils peuvent altérer la perception de la largeur et de la continuité des fissures. Pour une segmentation optimale des fissures, il est recommandé d'utiliser des angles plus petits ($\pm 15°$ à $\pm 30°$) car ils préservent les caractéristiques des fissures tout en offrant une diversité d'orientation suffisante.

2. ***Mise à l'échelle :*** La mise à l'échelle est une technique courante d'augmentation de données utilisée pour simuler des variations dans les tailles des fissures, en tenant compte des différences naturelles rencontrées dans les scénarios réels. La transformation pour la mise à l'échelle est représentée mathématiquement par la matrice :

$$S_s(x, y) = \begin{pmatrix} s & 0 \\ 0 & s \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Bien qu'elle soit bénéfique pour représenter des fissures de différentes tailles, la mise à l'échelle peut modifier des détails essentiels tels que la largeur et la texture des fissures. La mise à l'échelle uniforme dans une plage restreinte (par exemple, de 0,9 à 1,1) est généralement recommandée pour éviter de générer des caractéristiques de fissures irréalistes, assurant que les détails essentiels soient préservés pour une segmentation précise.

3. ***Ajustements de luminosité et de contraste :*** Pour simuler des variations dans les conditions d'éclairage, les discussions théoriques sur l'augmentation des données incluent souvent des ajustements aléatoires de la luminosité et du contraste des images. Cette approche améliore la robustesse d'un modèle face aux changements d'illumination, qui sont courants dans les scénarios réels. Bien que les ajustements de la luminosité et du contraste puissent aider le modèle à s'adapter à divers paramètres d'acquisition d'images, les altérations extrêmes doivent être évitées. Une calibration appropriée est cruciale pour maintenir la visibilité des fissures et s'assurer qu'elles soient distinguables du fond, sans les obscurcir ni exagérer leurs caractéristiques.

4. ***Réseau Génératif Adversarial (GAN) :*** Dans les tâches de segmentation des fissures, un défi majeur est la rareté des données annotées, essentielles pour l'entraînement de modèles efficaces. L'annotation des données exige généralement un effort manuel pour étiqueter les fissures dans les images, un processus à la fois chronophage et sujet à des erreurs humaines. Pour résoudre ce problème, une approche basée sur un GAN a été utilisée afin de générer de nouvelles images d'arrière-plan tout en préservant les positions et les détails des fissures existantes. Cette méthode augmente efficacement l'ensemble de données, éliminant ainsi la nécessité de créer manuellement de nouveaux masques, ce qui simplifie le processus de préparation des données.

L'architecture de cette approche est basée sur une structure GAN, composée de deux composants principaux : un Générateur d'Arrière-Plan Amélioré et un Discriminateur. Le Générateur d'Arrière-Plan Amélioré est conçu pour produire des images d'arrière-plan réalistes avec des fissures intégrées de manière transparente. Il y parvient en employant une série de couches convolutives et de blocs résiduels, connus pour leur capacité à capturer et à maintenir les caractéristiques complexes des images, telles que les détails fins des fissures. Le générateur prend en entrée des images de fissures, des masques qui délimitent les fissures, et des images d'arrière-plan aléatoires. Cette combinaison garantit que les fissures restent aux mêmes positions dans les arrière-plans générés, préservant leur intégrité structurelle.

L'architecture de cette approche repose sur deux étapes principales :

1. Générer des images d'arrière-plan factices à partir des images d'arrière-plan d'entrée sans fissures. 2. Transférer les fissures des images contenant des fissures $\mathbf{x}$ vers les arrière-plans générés factices à l'aide des masques correspondants $\mathbf{m}$.

Le Discriminateur évalue si les images générées, avec les fissures transférées sur les arrière-plans factices, sont réalistes par rapport aux vraies images contenant des fissures $\mathbf{y}$.

5. *Étape 1 : Génération d'Images d'Arrière-Plan Factices :* Le Générateur prend d'abord des images d'arrière-plan sans fissures $\mathbf{z}$ comme entrée et génère des images d'arrière-plan factices. Ces arrière-plans générés sont structurellement similaires aux images d'arrière-plan d'entrée mais peuvent contenir des variations dues au processus génératif. Le processus de génération de ces images d'arrière-plan factices peut être décrit par :

$$\hat{\mathbf{z}} = G_{\text{background}}(\mathbf{z})$$

Dans cette équation, $\mathbf{z}$ désigne l'image d'arrière-plan sans fissures, et $G_{\text{background}}$ est la partie du Générateur responsable de la création des images d'arrière-plan factices $\hat{\mathbf{z}}$.

6. *Étape 2 : Transfert des Fissures sur les Images d'Arrière-Plan Factices :* Une fois les images d'arrière-plan factices $\hat{\mathbf{z}}$ générées, le Générateur transfère les fissures des images contenant des fissures $\mathbf{x}$ sur ces arrière-plans factices générés. Le masque

$\mathbf{m}$ est utilisé pour spécifier les emplacements des fissures dans $\mathbf{x}$, et les fissures sont appliquées aux images d'arrière-plan factices $\hat{\mathbf{z}}$. Le processus de transfert des fissures sur les images d'arrière-plan factices est décrit comme suit :

$$G(\hat{\mathbf{z}}, \mathbf{m}, \mathbf{x}) = (1 - \mathbf{m}) \odot \hat{\mathbf{z}} + \mathbf{m} \odot \mathbf{x}$$

Dans cette équation, $\hat{\mathbf{z}}$ représente l'image d'arrière-plan factice générée à partir de l'image d'arrière-plan d'entrée, $\mathbf{x}$ représente l'image contenant des fissures, et $\mathbf{m}$ est le masque qui spécifie où les fissures de $\mathbf{x}$ doivent être placées sur $\hat{\mathbf{z}}$. Le terme $(1 - \mathbf{m}) \odot \hat{\mathbf{z}}$ préserve l'arrière-plan factice là où il n'y a pas de fissures, et $\mathbf{m} \odot \mathbf{x}$ transfère les fissures de $\mathbf{x}$ aux emplacements correspondants sur $\hat{\mathbf{z}}$, générant une nouvelle image avec les fissures intégrées de manière transparente dans l'arrière-plan factice généré.

Fonctions de Perte : Notre GAN utilise deux fonctions de perte principales : la perte adversariale, qui garantit que l'image générée semble réaliste, et une perte au niveau pixel pour garantir que les fissures sont correctement transférées.

(a) *Perte Adversariale :* La perte adversariale encourage le Générateur à produire des images que le Discriminateur ne peut pas distinguer des vraies images contenant des fissures. Elle est définie comme suit :

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}_{\mathbf{y} \sim p_{\text{data}}}[\log D(\mathbf{y})] + \mathbb{E}_{\hat{\mathbf{z}} \sim p_{\text{gen}}}[\log(1 - D(G(\hat{\mathbf{z}}, \mathbf{m}, \mathbf{x})))]$$

Dans cette équation, $\mathbf{y}$ fait référence aux vraies images contenant des fissures provenant de l'ensemble de données, $\hat{\mathbf{z}}$ désigne les images d'arrière-plan factices générées, $G(\hat{\mathbf{z}}, \mathbf{m}, \mathbf{x})$ représente la sortie du Générateur, qui transfère les fissures de $\mathbf{x}$ vers $\hat{\mathbf{z}}$ en fonction du masque $\mathbf{m}$, et D est le Discriminateur, qui a pour tâche de distinguer les images réelles des images générées.

(b) *Perte de Transfert des Fissures (Perte au Niveau Pixel) :* Pour s'assurer que les fissures des images originales sont transférées avec précision dans les images générées, une perte au niveau pixel est appliquée. Le masque de fissures $\mathbf{m}$ est utilisé pour guider où les fissures doivent apparaître. La perte au niveau pixel est définie comme suit :

$$\mathcal{L}_{\text{fissure}} = \mathbb{E}[\|\mathbf{m} \odot \mathbf{x} - \mathbf{m} \odot G(\hat{\mathbf{z}}, \mathbf{m}, \mathbf{x})\|^2]$$

Dans cette équation, $\mathbf{x}$ représente l'image originale contenant des fissures, $\mathbf{m}$ est le masque qui isole les zones de fissures, et $G(\hat{\mathbf{z}}, \mathbf{m}, \mathbf{x})$ représente l'image générée avec les fissures transférées vers le nouvel arrière-plan. La multiplication élément par élément $\odot$ garantit que la perte est calculée uniquement pour les zones de fissures. La perte d'erreur quadratique moyenne (MSE) minimise la différence entre les détails des fissures et leurs positions dans la sortie générée par rapport au masque original.

(c) *Perte Totale :* La perte totale combine à la fois la perte adversariale et la perte de transfert des fissures :

$$\mathcal{L}_{\text{totale}} = \mathcal{L}_{\text{GAN}} + \lambda \mathcal{L}_{\text{fissure}}$$

Dans cette équation, λ est un hyperparamètre qui équilibre la contribution de la perte adversariale (pour garantir le réalisme) et de la perte de transfert des fissures (pour garantir la précision du positionnement des fissures).

Cette approche d'entraînement garantit que le Générateur d'Arrière-Plan Amélioré produit des images d'arrière-plan réalistes avec des positions et des détails de fissures cohérents, tandis que le Discriminateur améliore continuellement la qualité et le réalisme des sorties en distinguant les images réelles des images synthétiques. L'utilisation de la perte MSE permet de maintenir efficacement les positions et les détails des fissures, éliminant ainsi la nécessité de créer de nouveaux masques. Cela permet non seulement de gagner du temps, mais aussi de garantir que les images générées sont idéales pour entraîner des modèles de segmentation, améliorant ainsi les performances des algorithmes de détection des fissures.

La Figure 4.6b illustre le processus et les résultats de l'utilisation du GAN pour la segmentation des fissures, montrant l'image originale de fissure, le masque généré, l'arrière-plan synthétisé avec la fissure, et l'image finale combinée.

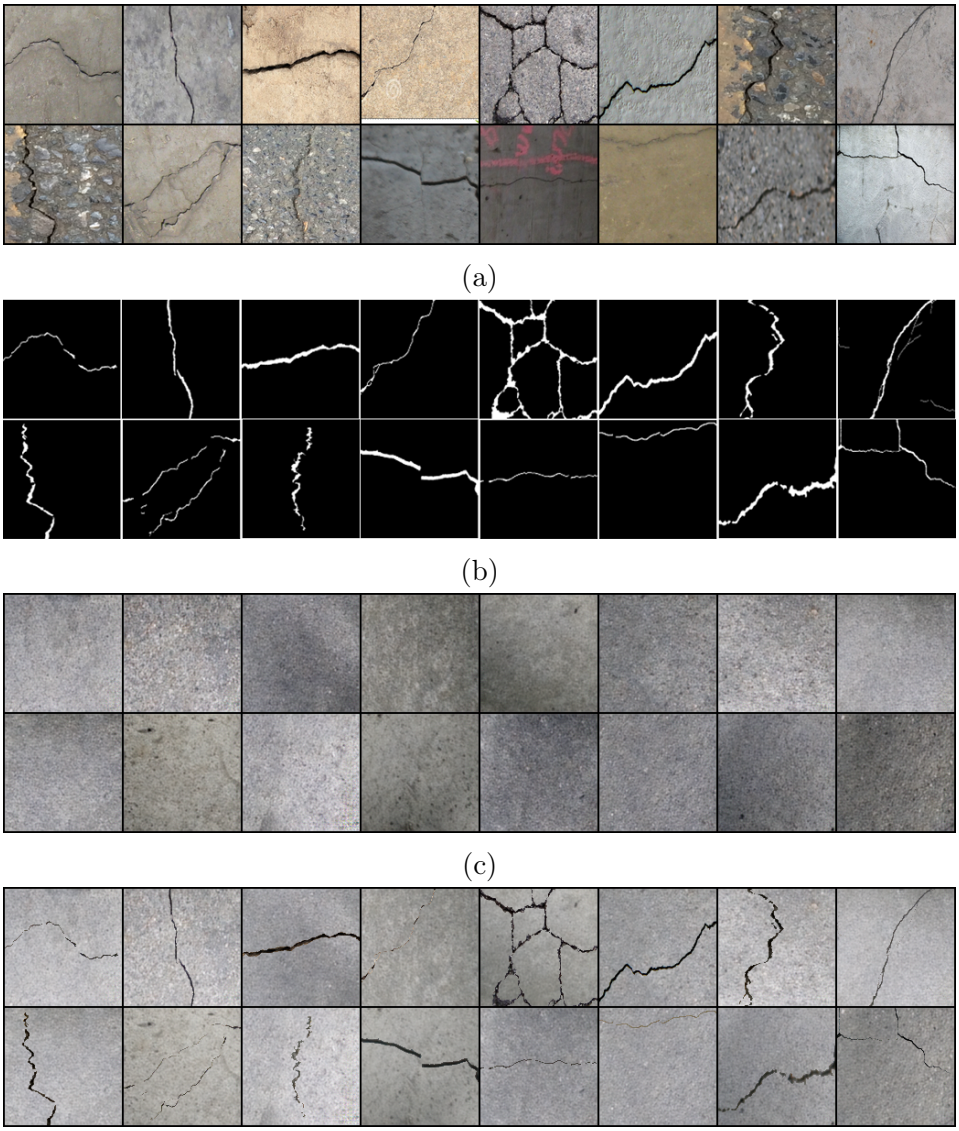


Figure 4.6: Résultats de l'utilisation du GAN pour la segmentation des fissures : (a) Images de fissures originales, (b) Masques, (c) Arrière-plans générés, et (d) Images combinées avec fissures et nouveaux arrière-plans.

4.5.2 Génération de données synthétiques : GANs vs méthodes classiques

Les méthodes classiques d'augmentation, telles que la rotation, la mise à l'échelle, le flou ou encore l'augmentation par des transformations mathématiques (par exemple,

les perturbations statistiques et les modèles algorithmiques de variation), offrent une approche simple pour augmenter artificiellement un ensemble de données. Cependant, ces techniques traditionnelles appliquent des transformations fixes qui ne capturent pas la complexité et la variabilité des caractéristiques des fissures. Elles produisent des images qui, bien qu'augmentées, ne ressemblent pas toujours à des fissures réelles, car elles manquent de la richesse des textures, du bruit et des variations environnementales présentes dans les images authentiques.

En revanche, les réseaux génératifs adversariaux (GANs) apprennent la distribution sous-jacente des fissures à partir d'un grand nombre d'images réelles. Cette capacité permet aux GANs de générer des données synthétiques qui présentent des fissures avec des détails réalistes, intégrant des variations de forme, de texture et d'éclairage similaires à celles observées dans des conditions réelles. Ainsi, les GANs fournissent non seulement une augmentation quantitative du jeu de données, mais améliorent également la qualité et la diversité des images générées, ce qui renforce la robustesse et la capacité de généralisation des modèles d'apprentissage profond pour la détection des fissures.

4.6 Intégration de Mécanismes d'Attention Avancés

Les avancées récentes en apprentissage profond, en particulier l'architecture U-Net, ont marqué des progrès significatifs dans ce domaine. Cependant, la quête d'une précision et d'un détail accrus dans la segmentation a conduit à l'exploration de divers mécanismes d'attention connus pour leur capacité à affiner l'extraction et le traitement des caractéristiques. Cette section examine l'intégration de méthodes sophistiquées basées sur l'attention—à savoir, les blocs Squeeze-and-Excitation (SE), les blocs Selective Kernel Network (SKNet), les blocs Non-local, le Dual Attention Network (DANet), le Spatial-Channel-Depthwise Convolutional Set Transformer (SETR), le Criss-Cross Attention Network (CCNet), le bloc Self-Attention (SA), et l'Aggregation de Couches Profondes (DLA) utilisant des mécanismes d'attention—dans l'architecture U-Net. La contribution unique de chaque méthode à la capture d'informations contextuelles globales et locales promet d'améliorer substantiellement les performances du modèle dans la détection et la segmentation des fissures à travers diverses plages d'observation et conditions.

1. ***Squeeze-and-Excitation (SE) :*** Les blocs Squeeze-and-Excitation (SE) introduisent un mécanisme d'attention par canal qui recalibre de manière adaptative

les réponses des canaux en modélisant explicitement les interdépendances entre les canaux. L'intégration des blocs SE dans l'architecture U-Net pourrait considérablement améliorer sa capacité de représentation [80]. Pour la surveillance de l'état des structures, cela signifie que le U-Net amélioré pourrait mieux différencier les caractéristiques des fissures et le bruit de fond, même dans des images complexes. Les blocs SE permettraient à U-Net de se concentrer davantage sur les caractéristiques informatives grâce à un réajustement apprenante de l'importance des canaux, ce qui pourrait conduire à une segmentation des fissures plus précise et robuste.

2. ***Selective Kernel Network (SKNet) :*** SKNet introduit un mécanisme de sélection dynamique entre des noyaux de différentes tailles au sein de ses unités Selective Kernel (SK), permettant au réseau d'ajuster de manière adaptative le champ réceptif en fonction de la caractéristique d'entrée [112]. Cette adaptabilité rend SKNet particulièrement adapté à la gestion des hiérarchies spatiales dans les images, comme celles trouvées dans des fissures de largeurs et de longueurs variées. En intégrant des blocs SKNet dans U-Net, l'architecture pourrait acquérir la capacité de se concentrer dynamiquement sur différentes échelles de caractéristiques de fissures au sein de la même image, améliorant ainsi la précision de la segmentation à travers une large gamme de tailles et de configurations de fissures. La nature sélective des blocs SKNet pourrait améliorer la flexibilité d'U-Net dans l'extraction des caractéristiques, offrant une approche plus nuancée et adaptable à la détection des fissures dans les évaluations de l'état des structures.

3. ***Bloc Non-local :*** Le bloc Non-local est un mécanisme d'auto-attention qui capture les dépendances à longue portée au sein d'une image en calculant la réponse à une position comme une somme pondérée des caractéristiques à toutes les positions [195]. L'intégration des blocs Non-local dans l'architecture U-Net peut considérablement améliorer sa capacité à considérer les informations contextuelles globales, cruciales pour identifier et segmenter les fissures qui couvrent de grandes zones de l'image ou qui sont subtiles et nécessitent un contexte plus large pour être détectées avec précision.

4. ***Dual Attention Network :*** DANet incorpore à la fois des mécanismes d'attention spatiale et par canal pour capturer des relations contextuelles riches au sein des images [47]. En ajoutant DANet à l'architecture U-Net, le modèle amélioré peut se

concentrer plus efficacement sur les caractéristiques pertinentes pour la segmentation des fissures, tant en termes d'emplacement (attention spatiale) qu'en termes d'importance des caractéristiques (attention par canal), conduisant à des résultats de segmentation plus précis et détaillés.

5. ***Intégration du SETR (Spatial-Channel-Depthwise Convolutional Set Transformer) :*** SETR exploite les modèles de transformateurs pour traiter les images, en se concentrant sur la capture de relations spatiales-canales complexes par le biais de l'auto-attention [193]. L'intégration de SETR dans U-Net pourrait offrir une approche révolutionnaire pour traiter la segmentation des fissures, en fournissant un mécanisme pour comprendre en profondeur les dépendances spatiales et canales des caractéristiques de fissures, améliorant la précision de la segmentation dans des scénarios divers et complexes.

6. ***U-Net avec CCNet (Criss-Cross Attention Network) :*** CCNet met en œuvre un mécanisme d'attention en croix efficace pour recueillir des informations contextuelles de manière plus efficace sur le plan computationnel que les modèles d'attention dense traditionnels [82]. En intégrant CCNet dans U-Net, le réseau peut atteindre une compréhension contextuelle complète avec une charge computationnelle réduite, améliorant sa capacité à segmenter les fissures avec précision tout en étant plus économe en ressources.

7. ***Self-Attention (SA) :*** Les blocs de Self-Attention, en tant que composant fondamental des modèles de transformateurs, permettent un pondération dynamique de l'importance des caractéristiques à travers toute l'image, en mettant l'accent sur les caractéristiques les plus pertinentes pour la tâche à accomplir [193]. Ajouter des blocs SA à U-Net enrichit sa capacité à se concentrer dynamiquement sur les caractéristiques cruciales des fissures dans différentes parties d'une image, améliorant la capture des détails et la précision de la segmentation.

8. ***Aggregation de Couches Profondes (DLA) avec mécanismes d'attention :*** DLA se concentre sur l'agrégation des caractéristiques de différentes couches d'un réseau pour améliorer la capacité de représentation du modèle. En utilisant des mécanismes d'attention pour guider le processus d'agrégation dans U-Net, DLA peut garantir que les caractéristiques pertinentes pour la détection des fissures sont mises en avant à travers la profondeur du réseau, conduisant à une combinai-

son de caractéristiques plus nuancée et efficace, et finalement, à des résultats de segmentation supérieurs.

4.6.1 Mécanisme de Mise au Point Linéaire Double-Attention (DALFM)

Pour affiner davantage la représentation des caractéristiques, nous intégrons un mécanisme d'attention avancé dans notre modèle. Ce mécanisme utilise à la fois des mécanismes d'attention par canal et spatiale, et incorpore des cartes de saillance pour se concentrer sur les régions les plus pertinentes de l'image d'entrée.

Le mécanisme d'attention par canal capture les dépendances complètes par canal en appliquant à la fois un poolage global par moyenne (GAP) et par maximum (GMP) aux cartes de caractéristiques d'entrée. Ces caractéristiques mises en pool sont traitées par des couches denses pour générer une carte d'attention par canal :

$$M_c(F) = \sigma(\text{MLP}(\text{GAP}(F)) + \text{MLP}(\text{GMP}(F)))$$

De plus, la carte de saillance S est interpolée pour correspondre aux dimensions de la sortie de l'attention par canal et est utilisée pour moduler cette carte d'attention :

$$M_c(F) = M_c(F) \otimes S$$

Cela garantit que l'attention se concentre sur les canaux les plus critiques pour la détection des fissures. La sortie de l'attention par canal modulée est ensuite utilisée pour redimensionner les caractéristiques d'entrée :

$$F' = F \otimes M_c(F)$$

Le mécanisme d'attention spatiale utilise des cartes de saillance pour mettre en avant les régions clés de l'image d'entrée. Ces cartes sont redimensionnées pour correspondre aux dimensions des cartes de caractéristiques d'entrée. Le mécanisme combine ces cartes de saillance redimensionnées avec les caractéristiques mises en pool par maximum et par moyenne le long de l'axe des canaux :

$$F_{avg}^{spatial} = \text{Mean}(F, \text{dim} = 1), \quad F_{max}^{spatial} = \text{Max}(F, \text{dim} = 1)$$

$$F_{concat} = [F_{avg}^{spatial}; F_{max}^{spatial}]$$

Une couche convolutionnelle traite les caractéristiques concaténées pour produire une carte d'attention spatiale :

$$M_s(F) = \sigma(\text{Conv}(F_{concat}))$$

La carte de saillance interpolée est également utilisée pour moduler la carte d'attention spatiale :

$$M_s(F) = M_s(F) \otimes S$$

Cela garantit que l'attention se concentre sur les régions spatiales les plus cruciales. La sortie de l'attention spatiale est ensuite utilisée pour redimensionner la carte des caractéristiques affinée par canal :

$$F'' = F' \otimes M_s(F)$$

Les sorties des mécanismes d'attention par canal et spatiale sont combinées par une multiplication élément par élément. Ce mécanisme d'attention intégré permet au modèle de mettre en avant dynamiquement les caractéristiques les plus pertinentes à la fois au niveau des canaux et au niveau spatial, améliorant ainsi la représentation globale des caractéristiques et la performance de la segmentation. La Figure illustre l'architecture du DALFM (Figure 4.7).

Nous avons testé divers mécanismes d'attention à intégrer dans notre architecture U-Net, et les meilleurs résultats ont été obtenus avec DALFM. Des définitions détaillées de toutes les méthodes testées et de leurs résultats sont fournies en annexe, accompagnées d'un tableau récapitulatif des performances. Le module DALFM, avec un total de 1,220 paramètres, s'ajoute aux paramètres globaux du modèle, pour un total combiné de 284,703 paramètres. Cette intégration améliore considérablement la capacité du modèle à détecter des motifs de fissures subtils et complexes dans des conditions variées, garantissant que le modèle se concentre sur les caractéristiques les plus informatives, tant sur le plan spatial qu'à travers les canaux. Cela conduit à des performances supérieures dans les tâches de détection des fissures.

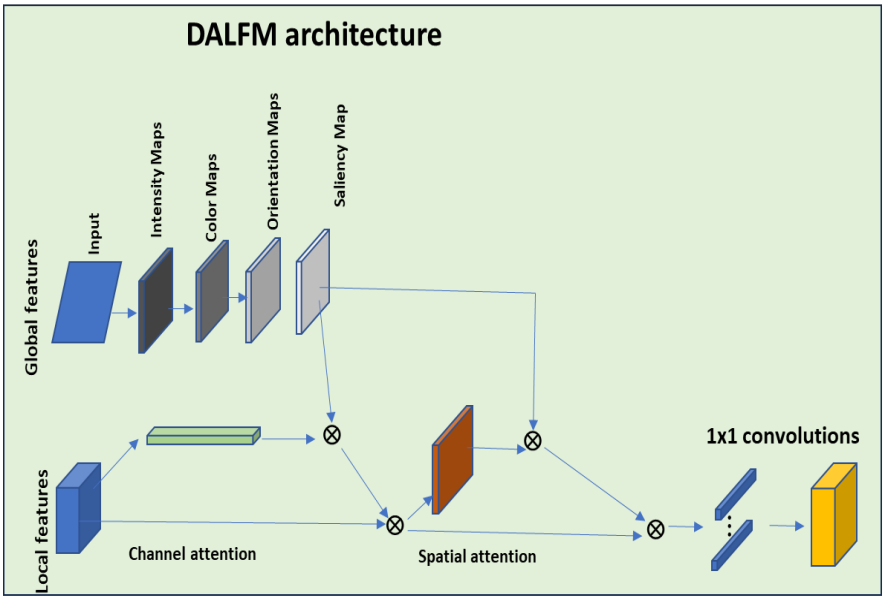


Figure 4.7: Architecture DALFM Proposée

4.6.2 Comparaison des Performances avec d'Autres Modèles d'Attention

La performance de divers mécanismes d'attention intégrés à l'architecture U-Net a été évaluée et comparée. Pour une comparaison équitable, tous les autres paramètres et réglages ont été maintenus constants ; seul le mécanisme d'attention a été varié. Le Tableau 4.1 résume les paramètres et la performance en termes de l'Intersection sur Union moyenne (mIoU) pour chaque mécanisme d'attention :

Mécanisme d'Attention	**G**	**C**	**P**	**R**	**F**	**AUC**	**D**	**IoU**
Squeeze-and-Excitation (SE)	0.9840	0.9742	0.6620	0.9638	0.7831	0.9742	0.7831	0.6463
Selective Kernel Network (SKNet)	0.9859	0.9801	0.6931	0.9739	0.8078	0.9801	0.8078	0.6799
Bloc Non-local	0.9855	0.9727	0.6879	0.9591	0.7978	0.9727	0.7978	0.6671
Dual Attention Network (DANet)	0.9865	0.9754	0.6918	0.9636	0.8006	0.9754	0.8006	0.6723
Spatial-Channel-Depthwise Convolutional Set Transformer (SETR)	0.9798	0.9556	0.6155	0.9305	0.7322	0.9556	0.7322	0.5857
Criss-Cross Attention Network (CCNet)	0.9884	0.9671	0.7403	0.9444	0.8269	0.9671	0.8269	0.7084
Self-Attention (SA)	0.9876	0.9734	0.7059	0.9583	0.8088	0.9734	0.8088	0.6841
Deep Layer Aggregation (DLA)	0.9854	0.9735	0.6828	0.9608	0.7945	0.9735	0.7945	0.6623
Dual-Attention Linear Focus Mechanism (DALFM)	**0.9930**	**0.9414**	**0.9133**	**0.8859**	**0.8994**	**0.9414**	**0.8994**	**0.8172**

Table 4.1: Comparaison des performances des mécanismes d'attention dans la segmentation des fissures. Métriques : Précision Globale (G), Précision Moyenne par Classe (C), Précision (P), Rappel (R), F-score (F), Surface sous la Courbe (AUC), Coefficient de Dice (D), Intersection sur Union (IoU).

4.7 Fonction de Perte Pondérée Contextuelle

Dans la segmentation des jeux de données de fissures, les données sont fortement déséquilibrées. Les fissures occupent généralement une très petite portion de chaque image, tandis que la majorité de l'image est constituée du fond. Par conséquent, le choix d'une fonction de perte appropriée est crucial pour une segmentation efficace. Pour résoudre ce problème, nous proposons deux améliorations significatives : *Compensation* et *Attention Contextuelle*.

Compensation

Pour atténuer le déséquilibre des classes, nous introduisons d'abord le terme *Compensation*. Les poids sont calculés dynamiquement en fonction du ratio des pixels de fissures par rapport aux pixels non fissurés dans les masques d'entraînement. Cette approche garantit que le modèle accorde plus d'attention aux pixels de fissures moins fréquents, améliorant ainsi sa capacité à identifier correctement les fissures. La fonction de perte avec Compensation est définie comme suit :

$$\mathcal{WL} = -\sum_{i=1}^{N} \left(w_0 \cdot (1 - y_{\text{true},i}) \cdot \log(1 - y_{\text{pred},i}) + w_1 \cdot y_{\text{true},i} \cdot \log(y_{\text{pred},i})\right) \tag{4.1}$$

Ici, $y_{\text{true},i}$ désigne l'étiquette réelle du i-ème pixel (indiquant la présence ou l'absence d'une fissure), $y_{\text{pred},i}$ est la probabilité prédite pour le i-ème pixel, et w_0 et w_1 sont les poids pour les classes non fissure et fissure, respectivement, afin de traiter le déséquilibre des classes.

Attention Contextuelle

En s'appuyant sur la Compensation, nous améliorons encore la fonction de perte en intégrant l'*Attention Contextuelle*. Les mécanismes d'attention peuvent fournir une vue d'ensemble de l'image, permettant au modèle de tirer parti des informations contextuelles pour se concentrer plus efficacement sur les régions de fissures.

Dans notre approche, les poids d'attention sont dérivés des cartes de caractéristiques combinées de LFB-Net, qui intègrent à la fois des cartes de classification et de régres-

sion. Ces poids d'attention $A_{i,j}$ sont utilisés pour moduler la perte pour chaque pixel, garantissant que le modèle se concentre davantage sur les zones critiques de l'image.

La fonction de perte mise à jour est donnée par :

$$\mathcal{CWL} = -\sum_{i=1}^{N} ((w_0 \cdot (1 - y_{\text{true},i}) \cdot \log(1 - y_{\text{pred},i}) + w_1 \cdot y_{\text{true},i} \cdot \log(y_{\text{pred},i})) \cdot A_{i,j}) \quad (4.2)$$

Ici, $\mathcal{CWL}$ représente la Perte Pondérée Contextuelle, $A_{i,j}$ est le poids d'attention pour le pixel à la position (i, j) dans l'image, et N est le nombre total de pixels dans l'image (ou le lot d'images).

En intégrant ces améliorations, le double mécanisme de *Compensation* et d'*Attention Contextuelle* améliore considérablement la performance du modèle en segmentant avec précision les fissures, en traitant les limitations des fonctions de perte traditionnelles dans la gestion du déséquilibre des classes et de la variabilité contextuelle dans les tâches de segmentation des fissures.

4.7.1 Évaluation des Différentes Fonctions de Perte

Pour évaluer la performance de notre fonction de perte pondérée contextuelle basée sur l'entropie croisée binaire, nous avons mené des expériences en la comparant à plusieurs autres fonctions de perte : Focal Loss, Dice Loss, Tversky Loss, Jaccard Loss (IoU Loss) et l'entropie croisée binaire. Pour une comparaison équitable, nous avons utilisé notre modèle CrackSight et seul la fonction de perte a été modifiée, en veillant à ce que tous les autres paramètres restent identiques. Les résultats de ces comparaisons sont présentés dans le Tableau 4.2 ci-dessous.

Fonction de Perte	**G**	**C**	**P**	**R**	**F**	**AUC**	**D**	**IoU**
Focal Loss	0.9895	0.8509	0.9265	0.7035	0.7971	0.8509	0.7971	0.6656
Dice Loss	0.9818	0.9789	0.6236	0.9758	0.7586	0.9789	0.7586	0.6135
Tversky Loss	0.9814	0.9752	0.6193	0.9687	0.7529	0.9752	0.7529	0.6073
Jaccard Loss (IoU Loss)	0.9780	0.9787	0.5866	0.9793	0.7296	0.9787	0.7296	0.5789
Entropie Croisée Binaire	0.9899	0.8680	0.9246	0.7379	0.8198	0.8680	0.8198	0.6962
Fonction de Perte Pondérée Contextuelle Basée sur l'Entropie Croisée Binaire	**0.9930**	**0.9414**	**0.9133**	**0.8859**	**0.8994**	**0.9414**	**0.8994**	**0.8172**

Table 4.2: Comparaison des différentes fonctions de perte. Métriques : Précision Globale (G), Précision Moyenne par Classe (C), Précision (P), Rappel (R), F-score (F), Surface sous la Courbe (AUC), Coefficient de Dice (D), Intersection sur Union (IoU).

4.7.2 Réseau de Détection des Fissures Basé sur les Caractéristiques Linéaires (LFB-Net)

Pour améliorer encore la performance de notre modèle sur différentes échelles d'image, nous introduisons le Réseau de Détection des Fissures Basé sur les Caractéristiques Linéaires (LFB-Net). Ce modèle est spécifiquement conçu pour la détection des fissures en exploitant la banque de filtres Leung-Malik (LM), qui est adaptée à l'isolement et à l'amélioration des structures linéaires dans les images. LFB-Net capture efficacement les caractéristiques cruciales pour détecter les fissures, améliorant ainsi la localisation des défauts dans les images acquises à différentes distances (Fig. 4.8). Nous avons construit LFB-Net en modifiant l'architecture RetinaNet pour y intégrer ces techniques d'extraction de caractéristiques linéaires et les mécanismes d'attention, le rendant ainsi plus efficace pour la détection des fissures.

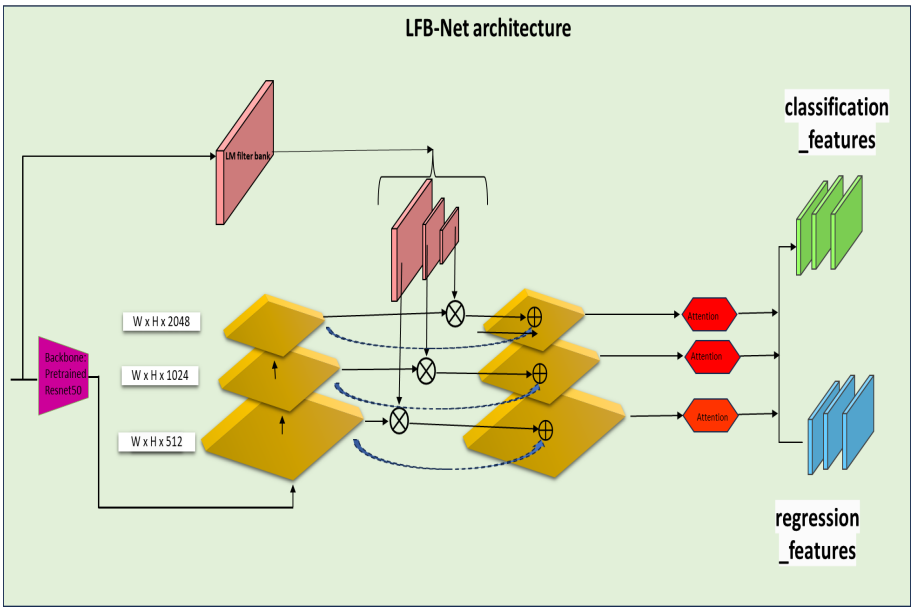


Figure 4.8: Architecture LFB-Net

Traitement Linéaire des Caractéristiques à Double Voie

LFB-Net utilise une approche à double voie pour le traitement des caractéristiques en utilisant la banque de filtres LM, assurant une détection complète des fissures. Cette approche implique :

– Extraction des caractéristiques linéaires à l'aide de la banque de filtres LM, qui améliore la détection des structures linéaires telles que les fissures.

– Extraction des caractéristiques pyramidales pour capturer des caractéristiques multi-échelles, ensuite affinées par un mécanisme d'attention.

De manière plus formelle, considérons une image d'entrée I qui peut contenir une ou plusieurs instances de fissures. L'image est traitée à l'aide de la banque de filtres LM pour produire un ensemble d'images filtrées $L_1, ..., L_n$. La sortie filtrée pour la partie P_i de la carte de caractéristiques après application du filtre LM L_i est donnée par :

$$M_i = L_i \cdot F_i$$

Où F_i représente les cartes de caractéristiques extraites de l'image d'entrée par l'épine dorsale ResNet au sein de LFB-Net. La multiplication élémentaire M_i améliore les caractéristiques linéaires pertinentes pour les fissures dans l'image.

Mécanisme d'Attention avec le Module Squeeze-and-Excitation (SE) Pour affiner davantage les caractéristiques extraites, LFB-Net utilise un module Squeeze-and-Excitation (SE) qui recalibre les cartes de caractéristiques M_i en fonction de leur importance. La carte de caractéristiques recalibrée est donnée par :

$$F_i' = SE(M_i)$$

La carte d'activation finale produite pour la partie P_i est :

$$A_i = ReLU(F_i')$$

Une carte d'activation complète A_{linear} est construite pour l'ensemble de l'image I en assemblant les cartes d'activation A_i générées pour les parties P_i à leurs positions correspondantes dans l'image I.

Intégration avec le Réseau Pyramidale de Caractéristiques Les cartes de caractéristiques améliorées sont intégrées dans le Réseau Pyramidale de Caractéristiques (FPN) utilisé dans LFB-Net. La combinaison des caractéristiques linéaires avec les cartes de caractéristiques pyramidales est définie comme suit :

$$F_{\text{final}} = F_{\text{pyramidal}} \oplus (F_{\text{pyramidal}} \otimes \text{ReLU}\,(A_{\text{linear}}))$$

Où $F_{\text{pyramidal}}$ représente les cartes de caractéristiques multi-échelles générées par le FPN, et A_{linear} est la carte d'activation recalibrée du module SE, se concentrant sur les caractéristiques linéaires améliorées.

Le processus de détection final de LFB-Net implique les modèles de régression et de classification, qui prédisent les boîtes englobantes et les scores de classe pour les fissures potentielles dans l'image. Les filtres LM, combinés au mécanisme d'attention, améliorent la capacité du modèle à détecter les fissures plus petites et à réduire les faux positifs en se concentrant sur les caractéristiques linéaires les plus pertinentes.

Cette méthode exploite une épine dorsale ResNet pré-entraînée, qui a été ensuite formée sur un ensemble de données de détection des fissures pour la spécialiser dans cette tâche spécifique. En formant l'épine dorsale ResNet sur des données spécifiques aux fissures, nous avons amélioré sa capacité à reconnaître les motifs de fissures de manière plus efficace, allant au-delà de sa conception initiale pour la classification d'images générales.

En plus de la formation de l'épine dorsale, nous avons considérablement amélioré la performance du modèle en affinant l'ensemble de données d'entraînement lui-même. Nous avons réduit la taille des boîtes englobantes, permettant ainsi au modèle de détecter des fissures plus petites et plus subtiles avec une plus grande précision. Cette modification a permis au modèle de se concentrer plus efficacement sur les caractéristiques des fissures, réduisant l'incidence des faux positifs et améliorant sa capacité à détecter les fissures même dans des environnements très complexes et bruyants.

De plus, en intégrant la banque de filtres Leung-Malik (LM), le modèle excelle dans l'isolement des structures linéaires, ce qui est essentiel pour une détection précise des fissures. Les modules d'attention Squeeze-and-Excitation (SE) affinent ensuite ces caractéristiques, garantissant que le modèle se concentre sur les aspects les plus pertinents des données d'entrée. Grâce à ces innovations complètes, notre Réseau de Détection des Fissures Basé sur les Caractéristiques Linéaires (LFB-Net) est non seulement capable de détecter les fissures avec une précision exceptionnelle, mais est également suffisamment robuste pour fonctionner de manière fiable dans une grande variété de scénarios réels difficiles. Qu'il s'agisse de traiter des défauts subtils dans des textures complexes ou des petites fissures qui étaient auparavant indétectables, LFB-Net montre une nette amélioration en termes de précision et de fiabilité par rapport aux approches traditionnelles.

Intégration des Caractéristiques LFB-Net

Notre modèle principal, CrackSight, tire parti des puissantes capacités d'extraction de caractéristiques de LFB-Net, en intégrant le mécanisme de mise au point linéaire à double attention (DALFM) avec la convolution à trous dans l'encodeur et une couche de saillance spécialisée pour améliorer la détection et la segmentation des fissures à toutes les échelles d'image. En combinant les cartes de caractéristiques de classification et de régression de LFB-Net et en les utilisant dans le décodeur, CrackSight garantit que le modèle se concentre sur les régions les plus pertinentes de l'image. De plus, la fonction de perte pondérée contextuelle basée sur l'entropie croisée binaire traite dynamiquement le déséquilibre des classes et intègre les informations contextuelles globales, améliorant ainsi la précision et l'efficacité de la détection des fissures au niveau des pixels dans diverses distances d'observation et scénarios réels.

Différenciation entre fissures réelles et fausses fissures

Pour répondre aux défis liés à la différenciation entre fissures réelles et fausses fissures (par exemple, des marques ou des dessins sur le pont), notre approche intègre plusieurs mécanismes de filtrage. D'une part, l'étape de détection utilise de petites boîtes englobantes qui isolent précisément les régions d'intérêt, réduisant ainsi la complexité du fond et limitant les risques de capturer des artefacts non pertinents. D'autre part, ces petites BB permettent de "piéger" uniquement les fissures authentiques en excluant les éléments qui ne présentent pas la continuité et la linéarité caractéristiques des véritables fissures.

De plus, le processus de traitement intégré dans LFB-Net exploite un filtrage basé sur des critères géométriques (tels que l'aspect, la taille et la continuité) et contextuels. Ce filtrage permet de distinguer les fissures réelles, qui présentent des caractéristiques linéaires cohérentes, des faux positifs résultant d'irrégularités de fond ou d'artefacts superficiels. En combinant ces deux stratégies – réduction de la complexité du fond par de petites boîtes englobantes et application de critères géométriques rigoureux – notre méthode parvient à éliminer efficacement les faux positifs et à se concentrer sur les fissures authentiques.

4.8 Validation Expérimentale

Pour évaluer notre méthode, nous préparerons d'abord le jeu de données pour la détection des défauts de pont, puis nous réaliserons des expériences pour comparer nos résultats avec les méthodes précédentes. Nous fournirons également des détails importants sur la mise en œuvre de notre méthode.

4.8.1 Jeux de Données

Pour évaluer les performances de notre réseau, nous avons effectué des expériences en utilisant trois jeux de données publics pour la segmentation (Masonry, Rissbilder et DeepCrack) et un pour la détection (CODEBRIM). Pour assurer la cohérence lors de l'entraînement, toutes les images pour la segmentation ont été redimensionnées à 448 $\times$ 448 pixels, tandis que les images pour la détection ont été redimensionnées à 640 $\times$ 640 pixels. Nous avons appliqué diverses techniques d'augmentation pour améliorer la robustesse du réseau. Chaque jeu de données a été divisé en ensembles de 70 % pour l'entraînement, 20 % pour la validation, et 10 % pour le test, assurant ainsi une évaluation complète à toutes les phases.

Masonry [36]: Ce jeu de données comprend 240 images de fissures dans des structures en maçonnerie prises avec des téléphones mobiles ou des appareils photo reflex numériques dans diverses structures de la région de Groningen aux Pays-Bas. En raison de la sismicité induite, ces structures ont subi des dommages, nécessitant des méthodes d'évaluation des dommages fiables. Les images sont annotées au niveau des pixels pour une classification binaire. Des techniques d'augmentation telles que des rotations, des retournements et des ajustements de luminosité ont augmenté le total à 1440 images. L'ensemble d'entraînement comprend 864 images, et les ensembles de validation et de test contiennent chacun 288 images. Ce jeu de données soutient le développement de solutions d'IA pour la détection automatique des fissures dans les structures en maçonnerie.

Rissbilder [149]: Ce jeu de données se compose de 2736 images de fissures dans le béton prises à l'aide de divers appareils à Florian, en Allemagne. Le jeu de données comprend principalement des fissures de mur, avec quelques images de fissures au sol. La résolution d'image d'origine de 448 $\times$ 448 a été préservée. Comme pour le jeu de données Masonry, des techniques d'augmentation telles que des rotations, des retournements et

des ajustements de luminosité ont été appliquées. L'ensemble d'entraînement comprend 1642 images, tandis que les ensembles de validation et de test incluent chacun 547 images.

DeepCrack [124]: Ce jeu de données se compose de 537 images en couleur annotées manuellement pour la segmentation. Les images ont une taille fixe de 544 × 384, redimensionnées à 448 × 448 pour la cohérence. Ce jeu de données comprend une variété de types de fissures, fournissant un ensemble complet pour l'entraînement et l'évaluation.

La Fig. 4.9 illustre des images d'exemples des jeux de données que nous avons utilisés, montrant des images de fissures et leurs étiquettes correspondantes. Le jeu de données Masonry contient principalement des images de fissures épaisses, dont certaines présentent une variation considérable de largeur. Le jeu de données Rissbilder est plus complexe, avec des fissures fines aux structures complexes et de nombreux motifs entrelacés. Le jeu de données DeepCrack comprend à la fois des fissures épaisses et fines, offrant un large éventail de types et de conditions de fissures, ce qui est idéal pour tester la robustesse des modèles de segmentation de fissures.

CODEBRIM [138]: Ce jeu de données comprend 1590 images haute résolution collectées à partir de 30 ponts à différentes échelles à l'aide d'une caméra drone. Le jeu de données contient des boîtes englobantes annotées pour les défauts. Pour la détection des fissures, nous avons utilisé 605 images de ce jeu de données. Nous avons corrigé les étiquettes en réduisant la taille des boîtes englobantes pour mieux répondre aux exigences de notre modèle. Nous avons également corrigé les étiquettes mal placées ou mal étiquetées dans le jeu de données CODEBRIM. La Fig. 4.10 montre quelques exemples d'images avec les boîtes englobantes corrigées.

4.8.2 Prétraitement : Lissage des Bords des Fissures

Une étape critique de prétraitement dans cette étude est le lissage des bords des fissures dans les annotations du jeu de données. Ce processus vise à améliorer la précision de la segmentation du modèle en réduisant la netteté des bords des fissures, permettant ainsi au modèle de mieux généraliser les données d'entraînement aux images non vues. Le processus est décrit comme suit en pseudocode :

1. Charger `image_path` en niveaux de gris, convertir en binaire en utilisant un seuil de 127, et éroder avec un noyau elliptique 3x3.

2. Soustraire l'image érodée pour définir les contours, puis appliquer un flou gaussien avec un noyau de (5,5) et un sigma de 0.

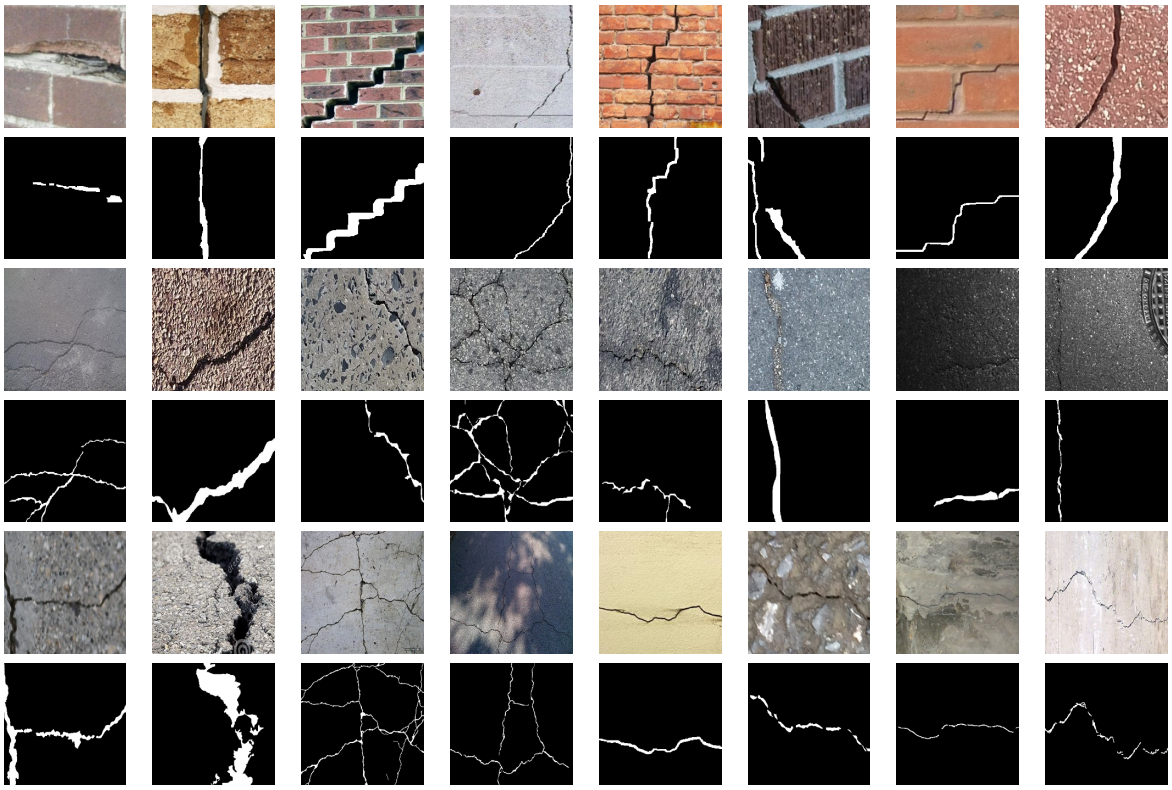


Figure 4.9: Images d'exemples des jeux de données que nous avons utilisés, montrant des images de fissures et leurs étiquettes correspondantes. La première rangée montre des images du jeu de données Masonry, et la deuxième rangée montre leurs masques. La troisième rangée contient des images du jeu de données Rissbilder, et la quatrième rangée montre leurs masques. La cinquième rangée inclut des images du jeu de données DeepCrack, et la sixième rangée montre leurs masques.

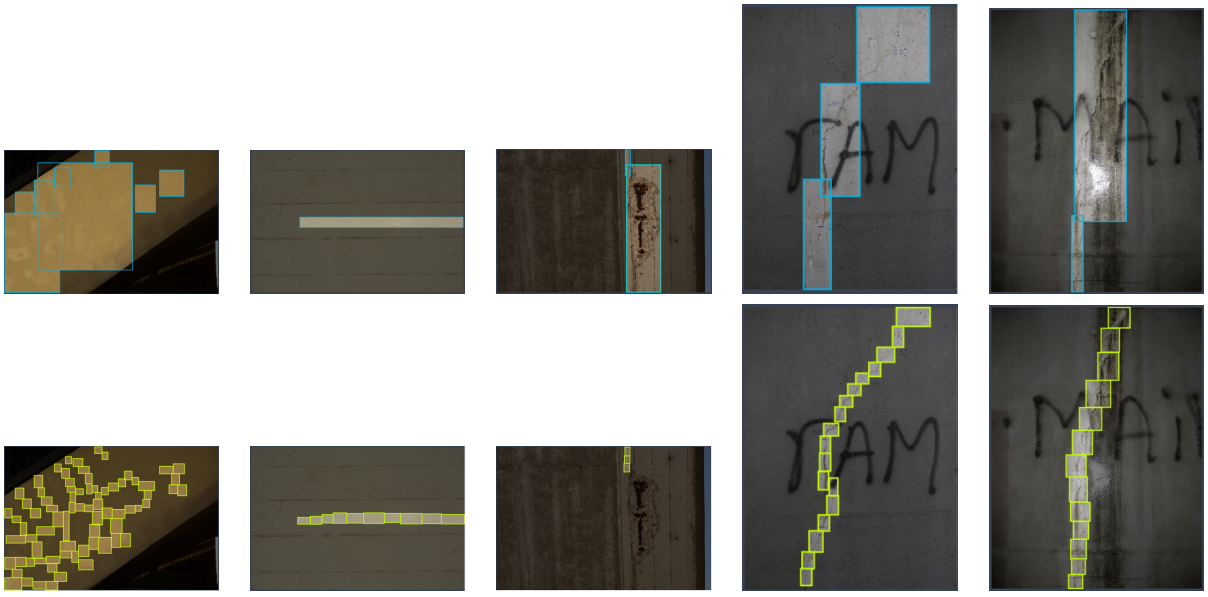


Figure 4.10: Exemples d'images du jeu de données CODEBRIM avec les boîtes englobantes corrigées. La rangée du haut montre les images originales, et la rangée du bas montre les images avec les boîtes englobantes corrigées.

3. Ajouter les contours flous à l'image binaire pour obtenir le masque final avec des bords lissés.

4.8.3 Raisonnement Derrière le Lissage des Bords

Le raisonnement derrière le processus de lissage des bords est double. Premièrement, il atténue l'impact des petites inexactitudes dans les annotations manuelles, ce qui peut conduire à un surapprentissage si le modèle apprend à reproduire ces incohérences. Deuxièmement, lisser les bords des fissures aide le modèle à se concentrer sur la forme générale et la présence des fissures plutôt que sur une délimitation pixel par pixel. Cette approche est particulièrement bénéfique pour l'apprentissage semi-supervisé, où la capacité du modèle à généraliser des données étiquetées limitées à un éventail plus large de données non étiquetées est cruciale. En lissant les bords des fissures, le modèle est formé pour reconnaître les caractéristiques essentielles des fissures, améliorant ainsi sa robustesse et sa précision dans les tâches de segmentation en conditions réelles.

L'étape de prétraitement du lissage des bords des fissures joue donc un rôle important dans l'amélioration des performances du modèle, le rendant plus apte à segmenter les fissures sur des images variées. Cette méthodologie souligne l'importance d'un pré-

traitement réfléchi pour atteindre une haute précision dans les tâches de segmentation des fissures.

4.8.4 Stratégie de Validation

La validation du modèle impliquera de tester ses performances sur un ensemble de validation distinct du jeu de données d'entraînement, afin de s'assurer que le modèle peut se généraliser efficacement. Cette approche permet de surveiller le comportement du modèle sur de nouvelles données, visant à confirmer la reproductibilité des résultats d'entraînement. Les principales métriques pour la validation incluront l'Intersection sur Union (IoU) et le coefficient de Dice, essentiels pour évaluer la qualité de la segmentation.

4.8.5 Comparaison avec les Méthodes de Pointe

Les performances de CrackSight ont été comparées à d'autres méthodes de pointe en utilisant trois jeux de données : Masonry, Rissbilder et DeepCrack. Les tableaux ci-dessous résument les résultats. Le tableau 1 présente les performances sur le jeu de données Masonry, le tableau 2 montre la comparaison de différents réseaux sur le jeu de données Rissbilder, et le tableau 3 fournit la comparaison des performances de diverses méthodes sur le jeu de données DeepCrack.

Table 4.3: Comparaison des Performances sur le Jeu de Données Masonry

Méthode	**Précision [%]**	**Exactitude [%]**	**F1-score [%]**	**Rappel [%]**	**MIoU [%]**
Masonry [36]	65.93	97.48	53.30	49.09	40.82
U-net-VGG [161]	65.40	97.38	48.87	43.44	36.94
U-net [161]	88.34	97.56	57.71	47.44	44.97
SegNet [9]	75.02	80.56	65.43	62.87	50.38
DeepLabv3+ [27]	65.93	97.29	46.12	40.86	48.53
DeepCrack [238]	86.99	96.15	64.87	54.91	51.51
UneXt [192]	86.06	96.74	68.47	62.34	56.41
DeepCrackAT [121]	88.62	97.41	76.67	71.24	64.44
CrackSight	**92.41**	**98.53**	**81.2**	**72.45**	**68.46**

Les résultats montrent que CrackSight surpasse significativement les méthodes existantes en matière de segmentation des fissures, excellant en termes de précision, rappel, F1-score et IoU. En intégrant DALFM, la convolution atrous et une couche de saillance au sein de LFB-Net, CrackSight améliore la détection et la segmentation à toutes les échelles d'image. L'utilisation de la perte binaire pondérée contextuelle (Contextual

Table 4.4: Comparaison des Différents Réseaux sur le Jeu de Données Rissbilder

Méthodes	**Précision [%]**	**Exactitude [%]**	**F1-score [%]**	**Rappel [%]**	**MIoU [%]**
Masonry [36]	41.20	33.01	33.23	30.48	35.80
U-net-VGG [161]	45.80	34.24	28.96	22.75	30.03
U-net [161]	65.16	92.26	39.25	31.97	26.76
SegNet [9]	56.21	81.82	59.42	67.82	43.66
DeepLabv3+ [27]	28.59	96.91	36.18	52.72	22.44
DeepCrack [238]	64.37	97.15	55.43	50.56	42.26
UneXt [192]	60.84	96.94	56.09	55.17	40.50
DeepCrackAT [121]	67.31	97.25	60.18	55.90	44.27
CrackSight	**76.47**	**97.85**	**72.35**	**68.65**	**56.68**

Table 4.5: Comparaison des Performances des Différentes Méthodes sur le Jeu de Données DeepCrack

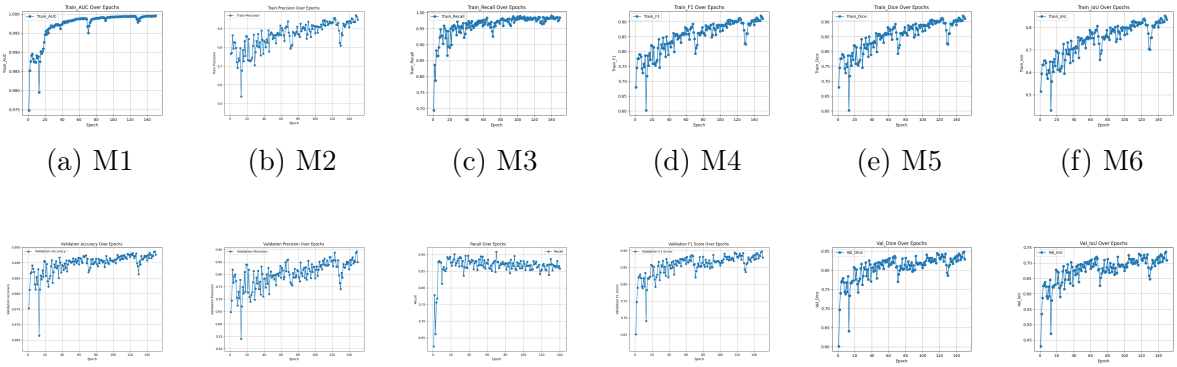
Méthode	**P [%]**	**R [%]**	**F1 [%]**	**IoU [%]**
HED [207]	78.78	88.12	83.19	71.21
RCF [123]	79.36	89.14	83.97	72.37
DeepCrack [124]	79.63	87.92	83.57	71.77
U-Net [161]	79.15	90.29	84.35	72.94
SegNet [9]	79.43	88.31	83.63	71.88
PSPNet [233]	69.50	82.87	75.60	60.77
Deeplabv3+ [27]	75.80	91.21	82.79	70.64
TransUNet [28]	78.04	91.00	84.02	72.45
SegFormer [208]	73.58	86.11	79.35	65.78
DMFNet [11]	76.71	90.56	83.06	71.03
CrackFormer [128]	81.15	91.81	86.15	75.68
GGMNet [202]	83.63	90.93	87.13	77.19
CrackSigh	**91.33**	**88.59**	**89.94**	**81.72**

Weighted Binary Cross-Entropy Loss) permet de résoudre le déséquilibre des classes et d'incorporer le contexte global, augmentant ainsi la précision et l'efficacité dans diverses distances d'observation et scénarios réels.

4.8.6 Courbes d'apprentissage pour l'entraînement et la validation

Les résultats présentés ci-dessous correspondent au modèle DeepCrack, entraîné avec une répartition de 70% d'entraînement, 20% de validation et 10% de test. La Figure 4.11 présente une image unique regroupant les courbes d'apprentissage pour six métriques clés, disposées en deux lignes : la première ligne montre les courbes pour l'ensemble

d'entraînement (Accuracy, Precision, Recall, F1, Dice et IoU), et la deuxième ligne présente ces mêmes métriques sur l'ensemble de validation. Ces courbes permettent d'observer la convergence du modèle ainsi que son aptitude à généraliser sans surapprentissage.



(a) M1 (b) M2 (c) M3 (d) M4 (e) M5 (f) M6

(g) M1 (h) M2 (i) M3 (j) M4 (k) M5 (l) M6

Figure 4.11: Évolution des métriques pour le modèle DeepCrack. La première ligne présente les courbes d'apprentissage pour l'Accuracy, Precision, Recall, F1, Dice et IoU sur l'ensemble d'entraînement, tandis que la deuxième ligne montre ces mêmes métriques sur l'ensemble de validation (70% des données pour l'entraînement, 20% pour la validation et 10% pour le test).

4.8.7 Analyse des Résultats

Les résultats indiquent que notre modèle proposé, CrackSight, surpasse les méthodes de pointe existantes en matière de segmentation des fissures sur presque toutes les métriques. La performance supérieure de CrackSight peut être attribuée à son design architectural innovant, qui se concentre sur la capture des caractéristiques pertinentes sans le surcoût computationnel associé à des modèles plus complexes. Plus précisément, CrackSight a atteint une précision de 92,41 %, 76,47 % et 91,33 % sur les jeux de données Masonry, Rissbilder et DeepCrack, respectivement, ce qui est nettement supérieur aux autres méthodes. La précision globale a atteint 98,53 %, 97,85 % et 88,59 %, démontrant la fiabilité du modèle dans la prédiction des étiquettes correctes sur différents jeux de données. Un F1-score de 81,2 %, 72,35 % et 89,94 % indique que CrackSight maintient un équilibre robuste entre précision et rappel, identifiant efficacement les vrais positifs tout en minimisant les faux négatifs. Avec des rappels de 72,45 %, 68,65 % et 88,59 %, le modèle prouve son efficacité à identifier la majorité des cas positifs, ce qui est

essentiel pour la maintenance des infrastructures. Les scores d'Intersection over Union (IoU) de 68,46 %, 56,68 % et 81,72 % soulignent la capacité de CrackSight à segmenter avec précision les régions de fissures. De plus, le coefficient de Dice de 89,94 % confirme l'exactitude du modèle dans la prédiction des zones fissurées, attestant de sa capacité à gérer efficacement des motifs de fissures complexes. Enfin, la précision moyenne par classe (Class Average Accuracy) et l'AUC ont atteint des valeurs élevées, indiquant la solide performance du modèle sur diverses classes, indépendamment des variations dans le jeu de données.

Les performances supérieures de CrackSight proviennent de plusieurs facteurs clés : il privilégie les caractéristiques de bas niveau, réduisant ainsi la complexité computationnelle et évitant l'extraction de caractéristiques non pertinentes. Les convolutions atrous dans l'encodeur capturent des informations contextuelles multi-échelles sans perte de résolution spatiale, permettant de détecter des fissures de tailles et de complexités variées. L'intégration du mécanisme DALFM avec des cartes de saillance optimise la focalisation sur les caractéristiques importantes, améliorant ainsi la précision et le rappel. Le design léger de CrackSight permet de réduire le nombre de paramètres par rapport aux architectures U-Net traditionnelles, le rendant adapté aux applications en temps réel dans des environnements à ressources limitées. De plus, la perte binaire pondérée contextuelle (Contextual Weighted Binary Cross-Entropy Loss) traite efficacement le déséquilibre des classes, améliorant la détection des fissures même dans des jeux de données déséquilibrés. En comparant CrackSight avec d'autres méthodes telles que DeepCrack, UNeXt, Masonry, U-Net, DeepLabv3+ et SegNet, il apparaît clairement que, malgré les points forts de ces approches, elles se révèlent insuffisantes dans certains aspects. Par exemple, DeepCrack et UNeXt, malgré leur fusion de caractéristiques multi-échelles et leur réduction de complexité computationnelle, n'atteignent pas la précision et le rappel de CrackSight. De même, DeepLabv3+ et SegNet, bien qu'efficaces dans des tâches générales de segmentation, peinent à relever les défis spécifiques de la détection des fissures, en raison de leur focalisation sur les caractéristiques profondes et de leur architecture complexe qui peut conduire à un surajustement ou à une extraction de caractéristiques superflues. En résumé, l'architecture de CrackSight, qui met l'accent sur l'extraction des caractéristiques pertinentes, le contexte multi-échelles et une gestion efficace des paramètres, surpasse les méthodes existantes. Sa haute précision, son rappel et son exactitude en font une solution idéale pour la surveillance et la maintenance des infrastructures, où la performance en temps réel et la fiabilité sont essentielles.

4.8.8 Résultats Qualitatifs

En plus de l'évaluation quantitative, nous fournissons des résultats qualitatifs pour illustrer les performances de notre réseau dans la détection et la segmentation des fissures. Des images d'exemples issues des jeux de données, capturées à courte distance, avec leurs prédictions correspondantes sont présentées dans la Fig. 4.12. Ces exemples montrent la capacité de notre modèle à identifier et segmenter avec précision les fissures dans différentes conditions.

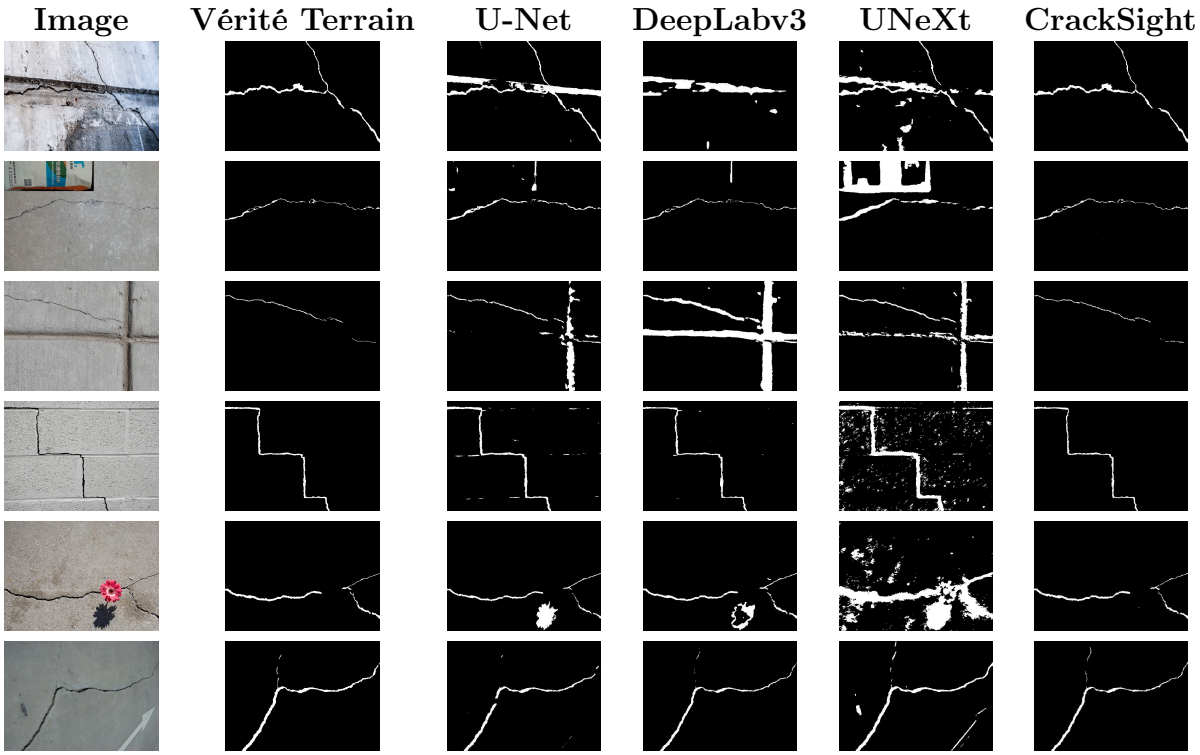


Figure 4.12: Exemples de résultats qualitatifs issus des jeux de données. Les images montrent l'image originale, la vérité terrain, et les prédictions d'U-Net, DeepLabv3, UNeXt, et CrackSight.

Pour les distances moyennes et éloignées, nous avons également évalué notre modèle sur des images complexes contenant des graffitis, des fissures très subtiles, une variabilité dans la texture et la couleur des surfaces, ainsi que d'autres complexités. Les figures 4.13 et 4.14 présentent de telles images, chacune contenant l'image originale, le résultat de la détection par LFB-Net, la version affinée de la détection et l'image segmentée finale.

Utilisation d'images haute résolution pour la détection à longue portée. Bien que notre approche soit principalement entraînée sur des images prises à courte distance, l'utilisation d'images haute résolution est essentielle pour la détection à moyenne et longue portée. En effet, des images haute résolution, comme celles du jeu de données CODEBRIM, permettent de conserver des détails fins et d'améliorer la détection des fissures même lorsque la distance d'acquisition est importante. Cependant, même avec des images haute résolution, des défis subsistent, notamment en raison des variations d'éclairage et de texture qui peuvent impacter la performance de la détection.

Origine des faux positifs et stratégies de mitigation. Les faux positifs résultent principalement de plusieurs facteurs :

- **Variabilité de la texture de surface :** Les différences entre le béton rugueux et lisse, dues aux matériaux, au vieillissement ou aux conditions météorologiques, peuvent masquer ou altérer la visibilité des fissures.
- **Variabilité d'éclairage :** Les ombres, les conditions de faible luminosité et les reflets peuvent générer des signaux erronés, conduisant à la détection d'artefacts comme des fissures.
- **Bruit environnemental :** Des éléments tels que la saleté, le graffiti ou d'autres marquages linéaires sur la surface peuvent imiter l'apparence des fissures.
- **Défauts chevauchants :** Les fissures se superposant à des taches, de la corrosion ou d'autres défauts rendent difficile leur segmentation précise.

Pour atténuer ces problèmes, notre approche utilise LFB-Net, qui intègre la banque de filtres Leung-Malik pour isoler les structures linéaires, combinée à des modules d'attention Squeeze-and-Excitation et à des convolutions atrous. Ces éléments permettent de distinguer plus efficacement les vraies fissures des artefacts et d'améliorer la robustesse du modèle face aux variations de texture et d'éclairage.

4.8.9 Améliorations apportées par CrackSight par rapport à U-Net

CrackSight intègre un mécanisme de focalisation linéaire à double attention (DALFM) et une couche de saillance directement dans l'encodeur, permettant ainsi d'extraire de

manière optimisée les caractéristiques essentielles pour la détection des fissures. Notre approche s'appuie sur le modèle de détection basé sur les caractéristiques linéaires (LFB-Net), qui améliore RetinaNet en intégrant la banque de filtres Leung-Malik (LM) et des modules Squeeze-and-Excitation (SE) pour détecter les fissures dans de petites boîtes englobantes de façon précise.

Par ailleurs, l'utilisation de convolutions atrous dans l'encodeur et le décodeur permet de capter des informations contextuelles multi-échelles sans perte de résolution spatiale. Ces améliorations permettent de générer des boîtes englobantes plus précises et d'obtenir une segmentation fine des fissures, en réduisant efficacement les faux positifs.

Ainsi, en combinant DALFM et la capacité de LFB-Net à isoler les caractéristiques linéaires via la banque LM et SE, tout en exploitant des convolutions atrous pour une meilleure agrégation multi-échelle, CrackSight surpasse la structure U-Net traditionnelle dans la détection et la segmentation des fissures.

4.8.10 Résumé

Ce chapitre a présenté CrackSight, une approche innovante conçue pour améliorer la segmentation des fissures à différentes distances en répondant aux principaux défis liés à la complexité des matériaux, à l'acquisition d'images et à l'annotation des données. En intégrant des mécanismes d'attention avancés, tels que le mécanisme de mise au point linéaire à double attention (DALFM) et le réseau de détection de fissures basé sur les caractéristiques linéaires (LFB-Net), ainsi qu'en utilisant une fonction de perte pondérée contextuelle, nous avons démontré des avancées significatives en termes de précision et d'efficacité dans la détection des fissures. Les résultats expérimentaux, à la fois quantitatifs et qualitatifs, soulignent la robustesse de notre méthode par rapport aux techniques actuelles de pointe.

Dans le dernier chapitre, nous conclurons en résumant les principales contributions de cette thèse, en discutant de ses implications pour la surveillance de l'état des structures (SHM) et en exposant les limites de notre approche actuelle. De plus, nous explorerons les pistes de recherche futures, en mettant l'accent sur la manière dont les résultats de ce travail peuvent contribuer à de nouveaux progrès dans la détection automatisée des défauts et les systèmes SHM.

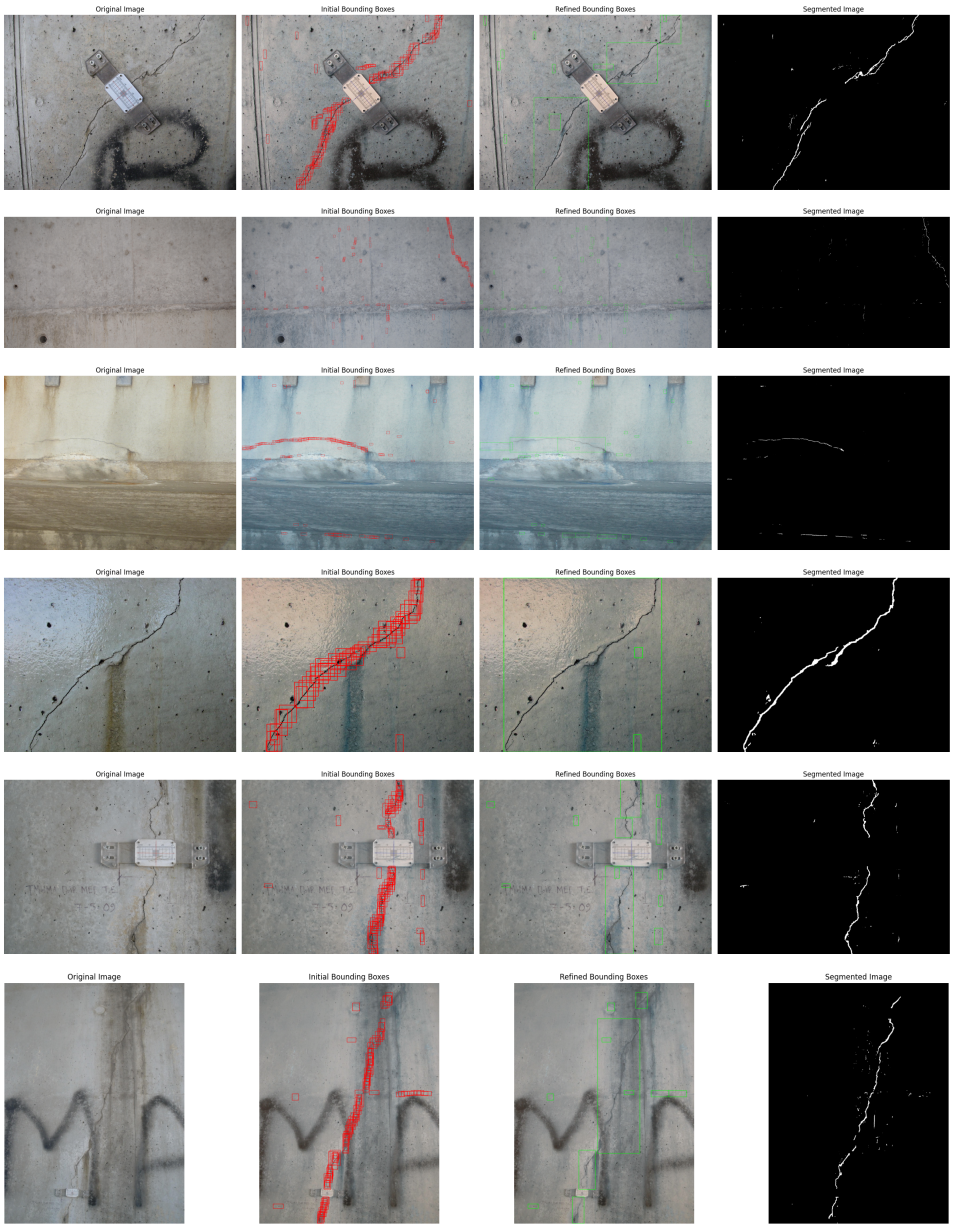


Figure 4.13: Exemples de résultats qualitatifs des jeux de données capturés à des distances moyennes et éloignées (Partie 1). Chaque image montre l'image originale, la détection LFB-Net, la détection affinée, et l'image segmentée dans un format combiné.

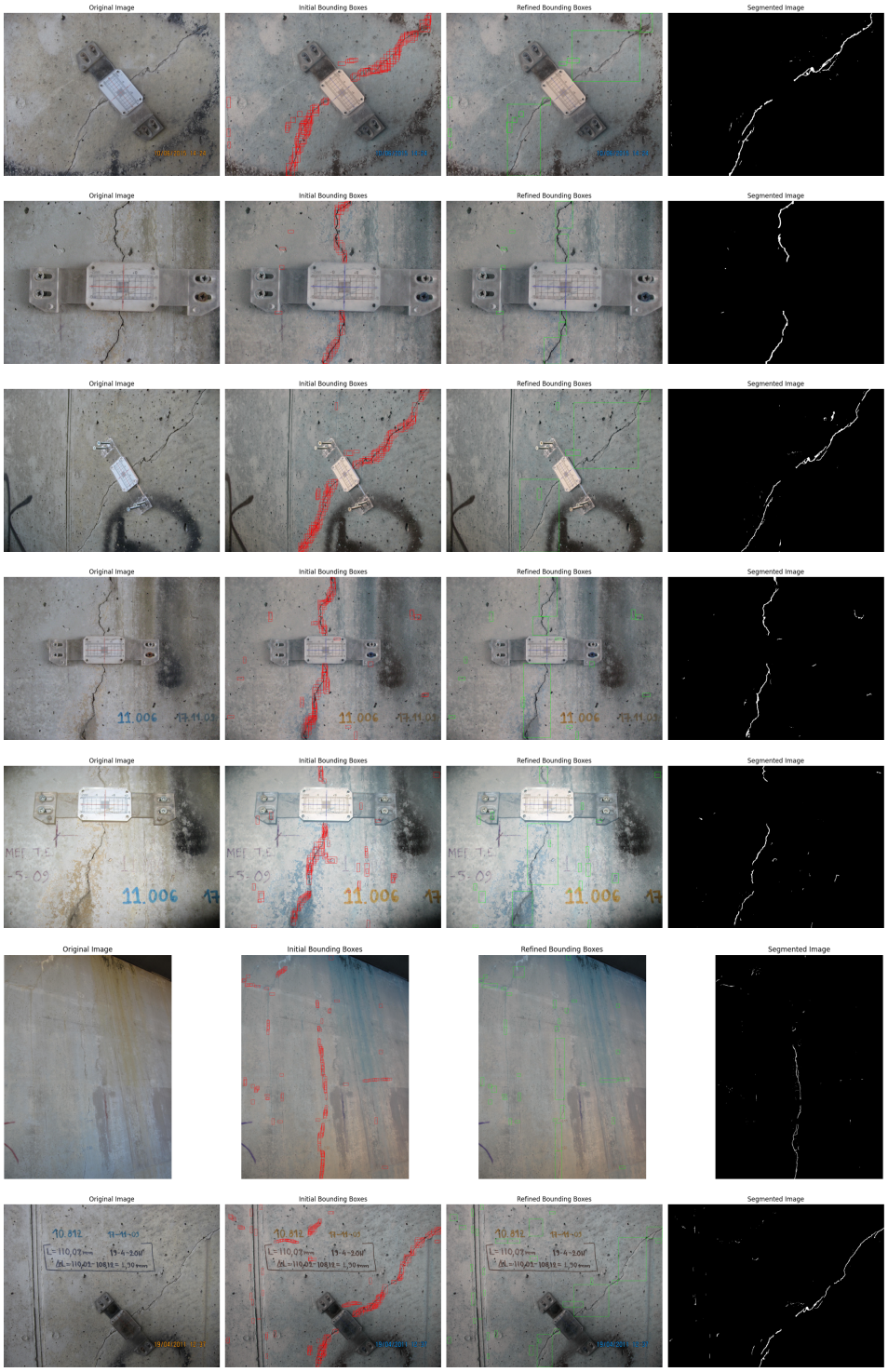


Figure 4.14: Exemples de résultats qualitatifs des jeux de données capturés à des distances moyennes et éloignées (Partie 2). Chaque image montre l'image originale, la détection LFB-Net, la détection affinée, et l'image segmentée dans un format combiné.

Chapitre 5

Conclusion

5.1 Récapitulation des Contributions de la Thèse

Cette thèse a apporté des contributions significatives dans le domaine de la surveillance de l'état des structures (SHM) en abordant les défis pressants de la détection et de la classification des défauts dans les infrastructures de ponts en béton. Grâce à l'application de techniques avancées d'apprentissage profond, la recherche fournit à la fois des contributions théoriques et pratiques qui repoussent les limites des méthodologies actuelles. Ce travail s'ancre dans le besoin de systèmes plus fiables, efficaces et automatisés pour surveiller et maintenir les infrastructures critiques, essentiels pour assurer la sécurité publique et prolonger la durée de vie de ces structures. En explorant systématiquement les limitations des approches existantes et en introduisant des solutions innovantes, cette thèse non seulement approfondit notre compréhension des complexités impliquées dans la détection des défauts, mais offre également des outils innovants qui peuvent être adaptés à une large gamme d'applications SHM. Les contributions réalisées ici posent une base solide pour les recherches et développements futurs, favorisant l'avancement de systèmes SHM plus résilients et évolutifs capables de répondre aux exigences croissantes de la gestion des infrastructures. Les contributions clés incluent :

- **Revue de la Littérature Complète :** Une revue détaillée de l'état de l'art de la détection visuelle des défauts des ponts en béton a été réalisée, identifiant les lacunes et défis existants. Cette revue a fourni la base pour développer des méthodologies nouvelles visant à combler ces lacunes, en particulier dans le contexte des systèmes d'inspection automatisés et de l'utilisation de modèles

d'apprentissage profond pour la détection et la classification des défauts. Ce travail a mis en évidence les limitations des méthodes traditionnelles et le potentiel d'intégration de techniques informatiques avancées pour améliorer les pratiques SHM.

- **Développement de CrackSight pour la Segmentation des Fissures :** L'introduction de CrackSight représente une avancée significative dans le domaine. Ce modèle intègre un mécanisme de focalisation linéaire à double attention (DALFM) et des couches de saillance dans une architecture U-Net modifiée. L'utilisation innovante des cartes de caractéristiques combinées du modèle de détection des fissures basé sur les caractéristiques linéaires (LFB-Net), ainsi que de la convolution atrous, permet à CrackSight de gérer efficacement des motifs de fissures complexes dans des conditions variées. L'introduction de la fonction de perte binaire pondérée contextuelle (Contextual Weighted Binary Cross-Entropy Loss) améliore encore la précision de la segmentation en traitant dynamiquement le déséquilibre des classes. À travers des expérimentations approfondies, CrackSight a démontré des performances supérieures par rapport aux modèles de référence, montrant sa robustesse dans les applications SHM réelles.

- **Validation Expérimentale et Étalonnage :** Des expériences rigoureuses ont été menées en utilisant divers jeux de données pour valider les modèles proposés. Des analyses comparatives ont démontré l'efficacité de CrackSight, en particulier dans des scénarios impliquant des arrière-plans complexes et encombrés. Les résultats ont souligné la capacité du modèle à détecter et segmenter avec précision les fissures, même dans des conditions difficiles, avançant ainsi l'état de l'art en matière de détection des défauts dans le béton.

- **Application à la Surveillance de l'État des Structures (SHM) :** La recherche a démontré l'application pratique des modèles développés dans le SHM, mettant en lumière leur potentiel à automatiser les processus de détection des défauts. En améliorant la précision et l'efficacité de la détection des fissures, ce travail contribue au développement de systèmes SHM plus fiables et évolutifs, qui sont cruciaux pour assurer la sécurité et la longévité des infrastructures.

5.2 Discussion et Implications pour la Surveillance de l'État des Structures

Les résultats de cette thèse ont plusieurs implications importantes pour le domaine du SHM, en particulier dans le contexte de l'automatisation et de l'amélioration de la précision de la détection des défauts dans les structures en béton.

- **Détection Automatisée et Précise des Défauts :** Les avancées présentées dans ce travail montrent le potentiel des modèles d'apprentissage profond, en particulier l'architecture U-Net améliorée utilisée dans CrackSight, pour améliorer significativement la précision et l'automatisation de la détection des fissures. Cela réduit la dépendance aux inspections manuelles, qui sont chronophages, coûteuses et souvent subjectives. L'automatisation de ce processus par des modèles fiables pilotés par l'IA, comme CrackSight, peut conduire à des évaluations plus cohérentes et objectives, garantissant que les défauts potentiels sont détectés tôt, évitant ainsi des défaillances structurelles plus graves.

- **Soutien à la Maintenance Préventive :** La détection précoce et précise des défauts structurels est essentielle pour mettre en œuvre des stratégies de maintenance préventive efficaces. La capacité de CrackSight à identifier même des fissures subtiles dans des arrière-plans complexes soutient les équipes de maintenance dans la mise en œuvre d'interventions en temps opportun, prolongeant ainsi la durée de vie des ponts et autres infrastructures critiques. Cela non seulement garantit la sécurité publique, mais offre également des avantages économiques significatifs en réduisant le besoin de réparations coûteuses et en minimisant les temps d'arrêt.

- **Évolutivité et Adaptabilité :** L'un des points forts des méthodes développées dans cette thèse est leur adaptabilité à différentes conditions d'imagerie et contextes structurels. La recherche a démontré que les modèles pouvaient être appliqués dans divers scénarios, allant des inspections par drone aux installations de caméras fixes, ce qui les rend adaptés à un large éventail d'applications SHM. Cette évolutivité est essentielle pour déployer ces technologies à grande échelle, transformant potentiellement la manière dont les infrastructures sont surveillées et maintenues à l'échelle mondiale.

– **Intégration avec les Technologies Émergentes :** La recherche ouvre également des voies pour l'intégration des modèles développés avec des technologies émergentes, telles que les drones, la robotique et la réalité augmentée (AR). Par exemple, la combinaison de CrackSight avec des UAVs peut permettre une surveillance complète et en temps réel de structures de grande taille, améliorant l'efficacité des inspections et réduisant les risques humains. De plus, l'intégration de ces modèles avec l'AR pourrait fournir aux inspecteurs des données visuelles augmentées lors des inspections, améliorant encore la précision et la rapidité de la détection des défauts.

5.3 Limitations et Directions Futures de Recherche

Bien que cette thèse ait réalisé des progrès substantiels dans l'avancement des pratiques SHM, certaines limitations subsistent, et plusieurs voies pour la recherche future ont été identifiées.

5.3.1 Limitations du Travail Actuel

– **Exigences en matière de Post-Traitement :** Les résultats obtenus avec CrackSight, bien qu'impressionnants, ne sont pas totalement parfaits. Dans certains cas, les sorties segmentées peuvent nécessiter un post-traitement pour affiner les contours des défauts ou pour éliminer le bruit introduit par des arrière-plans complexes. Bien que cette étape supplémentaire soit gérable, elle met en évidence le besoin d'avancées supplémentaires en matière de précision du modèle pour minimiser le besoin d'intervention manuelle.

– **Limitations des Jeux de Données et Généralisation du Modèle :** La performance des modèles d'apprentissage profond, y compris CrackSight, peut être affectée par la qualité et la diversité des jeux de données d'entraînement. Le manque de grands jeux de données publics disponibles avec des annotations complètes reste une barrière significative. Cette limitation affecte la capacité du modèle à se généraliser à différents types de structures et motifs de défauts. Les recherches futures devraient se concentrer sur l'expansion et la diversification des jeux de données pour améliorer la robustesse du modèle.

5.3.2 Axes de Recherche Future

- **Explorer la Généralisation Inter-Domaines :** Un axe clé pour la recherche future est l'exploration de l'adaptabilité des modèles à travers divers matériaux structurels et types de défauts. L'extension de l'approche actuelle pour gérer les défauts dans des matériaux tels que l'asphalte, le métal et le plastique pourrait élargir considérablement l'applicabilité des modèles développés. Cette généralisation inter-domaines est cruciale pour créer des outils polyvalents qui peuvent être déployés dans différents contextes SHM.

- **Aborder la Dérive des Données et l'Apprentissage Continu :** Le phénomène de dérive des données, où la performance du modèle se dégrade au fil du temps en raison de changements dans les données d'entrée, est un problème critique dans les applications SHM à long terme. Les travaux futurs devraient se concentrer sur le développement de modèles capables d'apprentissage continu, leur permettant de s'adapter aux nouvelles données sans perdre leur efficacité sur les scénarios précédemment rencontrés. Des techniques telles que l'apprentissage par transfert et l'apprentissage par faible nombre d'exemples offrent des approches prometteuses pour relever ces défis.

- **Intégration avec les Technologies Émergentes :** Des recherches supplémentaires sont nécessaires pour explorer l'intégration de CrackSight avec des technologies émergentes telles que les drones, la robotique et la réalité augmentée. Cela pourrait impliquer le développement de capacités de traitement en temps réel permettant une analyse en direct lors des inspections, ainsi que l'amélioration de l'interface matériel-logiciel pour renforcer l'efficacité et l'efficience globales des systèmes SHM.

- **Étendre CrackSight à d'Autres Défauts :** Bien que cette thèse se soit concentrée sur la détection des fissures, les méthodologies développées pourraient être étendues à d'autres types de défauts couramment rencontrés dans les infrastructures, tels que l'écaillage, la corrosion et les vides. Élargir la portée de CrackSight pour gérer ces types de défauts supplémentaires améliorerait son utilité dans le SHM et en ferait un outil plus complet pour la surveillance des infrastructures.

- **Améliorer les Techniques de Post-Traitement et de Raffinement :** Les recherches futures pourraient également se concentrer sur la réduction du besoin

de post-traitement en améliorant la précision des segmentations initiales. Cela pourrait impliquer le développement de fonctions de perte plus sophistiquées ou l'intégration du post-traitement directement dans le pipeline d'apprentissage profond, rendant le processus plus fluide et réduisant l'intervention manuelle.

– **Réduction de la Complexité pour le Déploiement sur UAVs :** Une autre direction importante est l'optimisation de la complexité de notre méthode pour permettre son déploiement sur des plateformes mobiles telles que les UAVs. Cela pourrait inclure l'exploration de techniques de compression de modèle, de quantification et d'optimisation de l'inférence afin de réduire la charge computationnelle et la consommation d'énergie. L'objectif est de garantir que le modèle conserve ses performances en temps réel tout en étant suffisamment léger pour être intégré dans des systèmes embarqués sur UAVs, ce qui est essentiel pour les applications de surveillance en conditions réelles.

5.4 Résumé

En résumé, bien que cette thèse ait réalisé des avancées significatives dans le domaine de la surveillance de l'état des structures, en particulier dans la détection et la segmentation des fissures dans les structures en béton, il est clair qu'il reste encore des défis à relever. Aborder les limitations identifiées, notamment en termes de diversité des jeux de données, de généralisation des modèles, et de nécessité de post-traitement, sera crucial pour s'assurer que les méthodes développées ici puissent être appliquées de manière fiable dans des scénarios réels.

Les recherches futures qui se construiront sur ce travail ont le potentiel d'améliorer considérablement la robustesse et l'adaptabilité des systèmes SHM, en les rendant plus efficaces pour maintenir la sécurité et la longévité des infrastructures critiques. En continuant à affiner ces modèles et à les intégrer aux technologies émergentes, la prochaine génération de systèmes SHM pourra offrir une précision, une efficacité et une évolutivité encore plus grandes, contribuant ainsi à des pratiques de gestion des infrastructures plus résilientes et durables.

Bibliography

[1] I. Abdel-Qader, O. Abudayyeh, M.E. Kelly. Analysis of edge-detection techniques for crack identification in bridges. *Journal of computing in civil engineering.*, vol. 17, no. 4, pp. 255–263, 2003.

[2] D. Ai, G. Jiang, S.K. Lam, P. He, and C. Li. Computer Vision Framework for Crack Detection of Civil Infrastructure—A Review. *Engineering Applications of Artificial Intelligence*, vol. 51, pp. 10547–10557, 2023.

[3] L. Ali, F. Alnajjar, H.A. Jassmi, M. Gocho, W. Khan, M.A. Serhani. Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures. *Sensors*, vol. 21, no. 5, pp. 1688–1699, 2021.

[4] Z. Al-Huda, B. Peng, R.N. Algburi, M.A. Al-antari, A.J. Rabea, D. Zhai. A Hybrid Deep Learning Pavement Crack Semantic Segmentation. *Engineering Applications of Artificial Intelligence*, vol. 122, p. 106142, 2023.

[5] R. Amhaz, S. Chambon, J. Idier, V. Baltazart. Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection. *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2718-29, 2016.

[6] D. Amirkhani, M. S. Allili, L. Hebbache, N. Hammouche, and J. -F. Lapointe. Visual Concrete Bridge Defect Classification and Detection Using Deep Learning: A Systematic Review. *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 9, pp. 10483-10505, 2024, doi: 10.1109/TITS.2024.3365296.

[7] D. Amirkhani, L. Hebbache, M. S. Allili, N. Hammouche, and J. -F. Lapointe. CrackSight: Advancing Crack Segmentation Across Varying Ranges. submitted to *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[8] A. Ayenu-Prah, N. Attoh-Okine. Evaluating pavement cracks with bidimensional empirical mode decomposition. *EURASIP Journal on Advances in Signal Processing.*, vol. 2008, pp. 1–7, 2008.

[9] V. Badrinarayanan, A. Kendall, and R. Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[10] Y. Bai, H. Sezen, and A. Yilmaz. Detecting Cracks and Spalling Automatically in Extreme Events by End-to-End Deep Learning Frameworks. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 161–168, 2021.

[11] S. Bai, L. Yang, Y. Liu, and H. Yu. DMF-Net: A Dual-Encoding Multi-Scale Fusion Network for Pavement Crack Detection. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[12] C. Benz, and V Rodehorst. Image-Based Detection of Structural Defects Using Hierarchical Multi-scale Attention. *DAGM German Conference on Pattern Recognition*, pp 337–353, 2022.

[13] C. Benz, P. Debus, H. K. Ha, and V. Rodehorst. Crack Segmentation on Uas-based Imagery Using Transfer Learning. *Int'l Conf. on Image and Vision Computing New Zealand*, pp. 1–6, 2019.

[14] M. Berman, A.R. Triki, M.B. Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. *IEEE Conf.on computer vision and pattern recognition*, pp. 4413–4421, 2018.

[15] G. Bhattacharya, B. Mandal, and N.B. Puhan. Interleaved Deep Artifacts-Aware Attention Mechanism for Concrete Structural Defect Classification. *IEEE Trans.s on Image Processing*, vol. 30, pp. 6957–6969, 2021.

[16] G. Bhattacharya, B. Mandal, and N.B. Puhan. Multi-Deformation Aware Attention Learning for Concrete Structural Defect Classification. *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 31, no. 9, pp. 3707–3713, 2021.

[17] S. Bhowmick, S. Nagarajaiah, and A. Veeraraghavan. Vision and Deep Learning-Based Algorithms to Detect and Quantify Cracks on Concrete Surfaces from UAV Videos. *Sensors*, vol. 20, no. 21, p. 6299, 2020.

[18] E. Bianchi, and M. Hebdon. COCO-Bridge 2021+ Dataset. *University Libraries, Virginia Tech.* 2021.

[19] E. Bianchi, and M. Hebdon. Labeled cracks in the wild (lcw) dataset, *University Libraries, Virginia Tech*, 2021.

[20] A. Bochkovskiy, C.Y. Wang, and H.Y. Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv:2004.10934 [cs.CV], Technical report*, 2020.

[21] Z.A. Bukhs, N. Jansen and A. Saeed. Damage Detection Using In-Domain and Cross-Domain Transfer Learning. *Neural Computing and Applications*, vol. 33, no. 24, pp. 16921–16936, 2021.

[22] F.O. Çağlar, R. "Ozgenel. Concrete Crack Images for Classification. *Mendeley data*, 2019.

[23] G.M. Calvi, M. Moratti, G.J. O'Reilly, N. Scattarreggia, R. Monteiro, D. Malomo, P. Martino, and P. Rui. Once upon a Time in Italy: The Tale of the Morandi Bridge. *Structural Engineering International*, vol. 29, no. 2, pp. 198–217, 2019.

[24] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin. SEmerging properties in self-supervised vision transformers. *IEEE Int'l conf. on computer vision*, pp. 9650–9660, 2021.

[25] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sensing*, vol. 13, p. 4712, 2021.

[26] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834-848, 2018.

[27] L.C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *European Conf. on Computer Vision*, pp. 833–851, 2018.

[28] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A.L. Yuille, and Y. Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021.

[29] C. Chen, H. Seo, C. Jun, and Y. Zhao. A Potential Crack Region Method to Detect Crack Using Image Processing of Multiple Thresholding. *Signal, Image and Video Processing*, vol. 16, pp. 1673–1681, 2022.

[30] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sensing*, vol. 13, p. 4712, 2021.

[31] H.D. Cheng, J. Wang, Y.G. Hu, C. Glazier, X.J. Shi ,and X.W. Chen. Novel approach to pavement cracking detection based on neural network. *Transportation research record.*, vol. 1764, no. 1, pp. 119–127, 2001.

[32] F. Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1800–1807, 2017.

[33] J.K. Chow, K.f. Liu, P.S. Tan, Z. Su, J. Wu, Z. Li, and Y-H. Wang. Automated Defect Inspection of Concrete Structures. *Automation in Construction*, Vol. 132, p. 103959, 2021.

[34] C. Chun and, S.K. Ryu. Road Surface Damage Detection using Fully CNNs and Semi-Supervised Learning. *Sensors*, vol. 19, no. 24, p. 5501, 2019.

[35] X. Cui, Q. Wang, J. Dai, R. Zhang, and S. Li. Intelligent Recognition of Erosion Damage to Concrete Based on Improved YOLOv3. *Materials Letters*, vol. 302, p. 130363, 2021.

[36] D. Dais, I. E. Bal, E. Smyrou, and V. Sarhosis. Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Automation in Construction*, vol. 125, p. 103606, 2021.

[37] J. Deng, Y. Lu, V.C.S. Lee. Imaging-Based Crack Detection on Concrete Surfaces Using You Only Look Once Network. *Structural Health Monitoring*, vol. 20, pp. 484–499, 2021.

[38] Z. Dong, J. Wang, B. Cui, D. Wang, and X. Wang. Patch-Based Weakly Supervised Semantic Segmentation Network for Crack Detection. *Construction and Building Materials*, vol. 258, p. 120291, 2020.

[39] S. Dorafshan, R. J. Thomas, and M. Maguire. Sdnet2018: An Annotated Image Dataset for Non-Contact Concrete Crack Detection Using Deep Convolutional Neural Networks. *Data in brief*, vol. 21, pp. 1664–1668, 2018.

[40] S. Dorafshan, R. J. Thomas, and M. Maguire. Comparison of Deep Convolutional Neural Networks and Edge Detectors for Image-based Crack Detection in Concrete. *Construction and Building Materials*, vol. 186, pp. 1031–1045, 2018.

[41] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, and J. Uszkoreit. An Image is Worth $16x16$ Words: Transformers for Image Recognition at Scale. *Int'l Conf. on Learning Representations*, pp. 1–22, 2021.

[42] C.V. Dung. Autonomous Concrete Crack Detection Using Deep Fully Convolutional Neural Network. *Automation in Construction*, vol. 99, pp. 52–58, 2019.

[43] F. Ellyin. Fatigue Damage, Crack Growth and Life Prediction. *Springer Science & Business Media*, 2012.

[44] A. Ali, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jégou. . "XCiT: Cross-Covariance Image Transformers. *Neural Information Processing Systems*, pp. 1–14, 2021.

[45] M.Everingham,L. Van Gool ,C.K. Williams ,J. Winn , A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, vol. 88, no. 2, pp. 303-38, 2010.

[46] H. Feng, W. Li, Z. Luo, Y. Chen, S. N. Fatholahi, M. Cheng, C. Wang, J. M. Junior, and J. Li. "GCN-based pavement crack detection using mobile LiDAR point clouds." *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11052-11061, 2021.

[47] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu . Dual attention network for scene segmentation. *the IEEE/CVF conf. on computer vision and pattern recognition*, pp. 3146–3154, 2019.

[48] R. Fu, M. Cao, D. Novak, X. Qian, NF. Alkayem. Extended Efficient Convolutional Neural Network for Concrete Crack Detection with Illustrated Merits. *Automation in Construction*, vol. 156, p. 105098, 2023.

[49] Y. Gao, and K.M. Mosalam. Deep Transfer Learning for Image-Based Structural Damage Recognition. *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, pp. 748–768, 2018.

[50] S.H. Gao, M.M. Cheng, K. Zhao, X.Y. Zhang, Yang MH, Torr P. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 2, pp. 652-62, 2019.

[51] Y. Gao, and K.M. Mosalam. PEER Hub ImageNet: A Large-Scale Multiattribute Benchmark Data Set of Structural Images. *Journal of Structural Engineering*, vol. 146, no. 10, 2020.

[52] X. Gao, C. Huang, S. Teng and G. Chen. A Deep-Convolutional-Neural-Network-Based Semi-Supervised Learning Method for Anomaly Crack Detection. *Applied Sciences*, vol. 12, no. 18, p. 9244, 2022.

[53] Y. Gao, H. Li and W. Fu. Few-Shot Learning for Image-Based Bridge Damage Detection. *Engineering Applications of Artificial Intelligence*, Vol. 126, Part C, p. 107078, 2023.

[54] Z. Ge, S. Liu, F. Wang, Z. Li and J. Sun. Yolox: Exceeding Yolo Series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

[55] R. Girshick, J. Donahue, T. Darrell, J. Malik . Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[56] R. Girshick. Fast R-CNN. *IEEE Int'l Conf. on computer vision*, pp. 1440–1448, 2015.

[57] K. Gkoumas, F.L. Marques Dos Santos, M. Van Balen, A. Tsakalidis, A. Ortega Hortelano, M. Grosso, G. Haq, and F. Pekár. Research and innovation in bridge maintenance, inspection and monitoring. *Publications Office of the European Union*, 2019.

[58] J. Glenn. YOLOv5 release v6.1. `https://github.com/ultralytics/yolov5/releases/tag/v6.1`, 2022.

[59] J. Glenn. Ultralytics YOLOv8. `https://github.com/ultralytics/ultralytics`, 2023.

[60] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[61] Guo L, Li R, Jiang B, Shen X. Automatic Crack Distress Classification from Concrete Surface Images Using a Novel Deep-width Network Architecture. *Neurocomputing*, vol. 397, pp. 383–92, 2020.

[62] L. Guo, R. Li ,and B. Jiang. A cascade broad neural network for concrete structural crack damage automated classification. *IEEE Transactions on Industrial Informatics.*, vol. 17, no. 4, pp. 2737–2742, 2020.

[63] J. Guo, Q. Wang and Y. Li. Semi-Supervised Learning Based on CNN and Uncertainty Filter for Façade Defects Classification. *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 3, pp. 302–317, 2021.

[64] M.H. Guo,T.X. Xu, J.J. Liu, Z.N. Liu, P.T. Jiang, T.J. Mu, S.H. Zhang, R.R. Martin,M.M. Cheng, S.M. Hu ., Attention mechanisms in computer vision: A survey. *Computational Visual Media*, vol. 8, no. 3, pp. 331–368, 2022.

[65] F. Guo, Y. Qian, J. Liu, H. Yu. Pavement Crack Detection Based on Transformer Network. *Automation in Construction*, vol. 145, p. 104646, 2023.

[66] P. Gupta, and M. Dixit. Image-Based Crack Detection Approaches: A Comprehensive Survey. *Multimedia Tools and Applications*, vol. 581, pp. 40181–40229, 2022.

[67] K. Hacıefendioğlu, Kemal and H.B. Başağa. Concrete Road Crack Detection using Deep Learning-Based Faster R-CNN Method. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, vol. 46, no. 2, pp. 1621–33, 2022.

[68] Y. Hamishebahar, H. Guan, S. So and J. Jo. A Comprehensive Review of Deep Learning-Based Crack Detection Approaches. *Applied Sciences*, vol. 12, no. 3, p. 1374, 2022.

[69] X. Han, Z. Zhao, L. Chen, X. Hu, Y. Tian, C. Zhai, L. Wang, and X. Huang. Structural Damage-causing Concrete Cracking Detection Based on a Deep-learning Method. *Construction and Building Materials*, vol. 337, p. 127562, 2022.

[70] H. Hao, Y. Zhou, and Y. Guo, "A brief survey on semantic segmentation with deep learning," *Neurocomputing*, vol. 406, pp. 302-321, Sep. 2020.

[71] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik. Semantic contours from inverse detectors. *IEEE Int'l conf. on computer vision*, pp. 991-998, 2011.

[72] K. He, X. Zhang, S. Ren, J. Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence* vol. 37, no. 9, pp. 1904–1916, 2015.

[73] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[74] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask R-CNN. *IEEE Int'l Conf. on Computer Vision*, pp. 2961–2969, 2017.

[75] X. He, Z. Chang, L. Zhang, H. Xu, H. Chen and Z. Luo. A Survey of Defect Detection Applications Based on Generative Adversarial Networks. *IEEE Access*, vol. 10, pp. 113493–113512, 2022.

[76] L. Hebbache, D. Amirkhani, M.S. Allili, N. Hammouche, J-F. Lapointe. Leveraging Saliency in Single-Stage Multi-Label Concrete Defect Detection Using Unmanned Aerial Vehicle Imagery. *Remote Sensing*, vol. 15, no. 5. p. 1218, 2023.

[77] N.-D. Hoang, Q.-L. Nguyen, and X.-L. Tran. Automatic Detection of Concrete Spalling using Piecewise Linear Stochastic Gradient Descent Logistic Regression and Image texture Analysis. *Complexity*, vol. 2019, 2019.

[78] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden and A. Peste. Sparsity in Deep Learning: Pruning and Growth for Efficient Inference and Training in Neural Networks. *Journal of Machine Learning Research*, vol. 23, pp. 1–124, 2021.

[79] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv preprint arXiv:1704.04861*, 2017.

[80] J. Hu, L. Shen and G. Sun. Squeeze-and-excitation networks. *In Proceedings of the IEEE Conf. on computer vision and pattern recognition*, pp. 7132–7141, 2018.

[81] G. Huang, Z. Liu, and K. Q. Weinberger. Densely Connected Convolutional Networks. *IEEE Conf.on Computer Vision and Pattern Recognition*, pp. 2261–2269, 2017.

[82] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, W. Liu. Ccnet: Criss-cross attention for semantic segmentation. *IEEE/CVF international conference on computer vision*, pp. 603–612, 2019.

[83] P. H"uthwohl, R. Lu, and I. Brilakis. Multi-Classifier for Reinforced Concrete Bridge Defects. *Automation in Construction*, vol. 105, p. 102824, 2019.

[84] P. H"uthwohl. Cambridge Bridge Inspection Dataset. 2017. *https://doi.org/10.17863/CAM.13813*, 2017.

[85] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, and A.C. Berg. ImageNet Large Scale Visual Recognition Challenge. *Int'l J. of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[86] M.R. Jahanshahi, J.S. Kelly, S.F. Masri, and G.S. Sukhatme. A Survey and Evaluation of Promising Approaches for Automatic Image-Based Defect Detection of Bridge Structures. *Structure and Infrastructure Engineering*, vol. 5, no. 6, pp. 455–486, 2009.

[87] S.B. Jha and R.F. Babiceanu. Deep CNN-based Visual Defect Detection: Survey of Current Literature. *Computers in Industry*, vol. 148, p. 103911, 2023.

[88] Y. Jiang, D. Pang, and C. Li. A Deep Learning Approach for Fast Detection and Classification of Concrete Damage. *Automation in Construction*, vol. 128, p. 103785, 2021.

[89] T. Jin, X.W. Ye ,and Z.X. Li. Establishment and evaluation of conditional GAN-based image dataset for semantic segmentation of structural cracks. *Engineering Structures.*, vol. 285, p.116058, 2023.

[90] D. Joshi, T.P. Singh, G. Sharma. Automatic surface crack detection using segmentation-based deep-learning approach. *Engineering Fracture Mechanics*, vol. 268, p. 108467, 2022.

[91] D. Kang, S.S. Benipal, D.L. Gopal, and Y.J. Cha. Hybrid Pixel-Level Concrete Crack Segmentation and Quantification Across Complex Backgrounds Using Deep Learning. *Automation in Construction*, vol. 118, p. 103291, 2020.

[92] I. Katsamenis, E. Protopapadakis, N. Bakalos, A. Varvarigos, A. Doulamis, N. Doulamis and A. Voulodimos. A Few-Shot Attention Recurrent Residual U-Net for Crack Segmentation. *Springer International Symposium on Visual Computing.*, pp. 199–209, 2023.

[93] D. Karimi, S.E. Salcudean. Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. *IEEE Trans. on medical imaging.*, vol. 19, no. 2, pp. 499–513, 2019.

[94] H. Kervadec, J. Bouchtiba, C. Desrosiers, E. Granger, J. Dolz, I.B. Ayed. Boundary loss for highly unbalanced segmentation.*International conference on medical imaging with deep learning*, pp. 285–296, 2019.

[95] S.M. Khan, S. Atamturktur, M. Chowdhury and M. Rahman. Integration of Structural Health Monitoring and Intelligent Transportation Systems for Bridge Condition Assessment: Current Status and Future Direction. *IEEE Trans. on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2107–2122, 2016.

[96] S. Khan, M. Naseer, M. Hayat, S.W. Zamir, F.S. Khan, and M. Shah. Transformers in Vision: A Survey. *ACM Computing Surveys*, 54, no. 10, pp. 1–41, 2022.

[97] I.H. Kim, H. Jeon, S.C. Baek, W.H. Hong, and H.J. Jung. Application of Crack Identification Techniques for an Aging Concrete Bridge Inspection Using an Unmanned Aerial Vehicle. *Sensors*, vol. 18, no. 6, p. 1881, 2018.

[98] B. Kim, and S. Cho. Automated Multiple Concrete Damage Detection Using Instance Segmentation Deep Learning Model. *Applied Sciences*, vol. 10, no. 22, p. 8008, 2020.

[99] B. Kim, N. Yuvaraj, K. S. Preethaa, and R. A. Pandian. Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. *Neural Computing and Applications*, vol. 33, no. 15, pp. 9289–9305, 2021.

[100] J.H. Kim ,and J. Lee. Efficient Dataset Collection for Concrete Crack Detection With Spatial-Adaptive Data Augmentation. *IEEE Access.*, vol. 11, pp 121902–121913, 2023.

[101] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.Y. Lo, P. Doll'ar. Segment Anything. *arXiv preprint arXiv:2304.02643.*

[102] K.R. Kirschke, S.A. Velinsky. Histogram-based approach for automated pavement-crack sensing. *Journal of Transportation Engineering.*, vol. 118, no. 5, pp. 700–710, 1992.

[103] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth. A Review on computer vision-Based Defect Detection and Condition Assessment of Concrete and Asphalt Civil Infrastructure. *Advanced Engineering Informatics*, Vol. 29, no. 2, pp. 196–210, 2015.

[104] J. K"onig , M.D. Jenkins, M. Mannion, P. Barrie and G. Morison. Weakly-Supervised Surface Crack Segmentation by Generating Pseudo-Labels Using Localization With a Classifier and Thresholding. *IEEE Trans. on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24083–24094, 2022

[105] A. Krizhevsky, I. Sutskever, and G.E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[106] P. Kumar, S. Batchu, S. R. Kota. Real-time Concrete Damage Detection Using Deep Learning for High Rise Structures. *IEEE Access*, vol. 9, pp. 112312–112331, 2021.

[107] J-F. Lapointe et al. AI-AR for Bridge Inspection by Drone. *Int'l Conf. on Human-Computer Interaction*, pp. 302-313, 2022.

[108] J.-F. Lapointe, H. Sekkati, M.S. Allili, L. Hebbache, D. Amirkhani, and N. Hammouche. AI-AR for Remote Visual Bridge Inspection by Drone. in *11th International*

Conference on Structural Health Monitoring of Intelligent Infrastructure (SHMII-11), pp. 462-465, 2022.

[109] J-F. Lapointe, I Kondratova . A Bridge Inspection Task Analysis. *Int'l Conf. on Human-Computer Interaction*, pp. 280-290, 2023.

[110] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature*, vol. 521, pp. 436–444, 2015.

[111] R. Li, Y. Yuan, W. Zhang, and Y. Yuan. Unified Vision-Based Methodology for Simultaneous Concrete Defect Detection and Geolocalization. *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, pp. 527–544, 2018.

[112] X. Li, W. Wang, X. Hu and, J. Yang. Selective kernel networks. *In Proceedings of the IEEE/CVF conf. on computer vision and pattern recognition*, pp. 510–519, 2019.

[113] S. Li, X. Zhao, and G. Zhou. Automatic Pixel-Level Multiple Damage Detection of Concrete Structure Using Fully Convolutional Network. *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 7, pp. 616–634, 2019.

[114] S. Li, and X. Zhao. Image-Based Concrete Crack Detection Using Convolutional Neural Network and Exhaustive Search Technique. *Advances in Civil Engineering*, vol. 2019, 2019.

[115] S. Li ,and X. Zhao. Automatic crack detection and measurement of concrete structure using convolutional encoder-decoder network. *IEEE Access.*, vol. 8, pp. 134602–134618, 2020.

[116] G. Li, J. Wan, S. He, Q. Liu ,and B. Ma. Semi-supervised semantic segmentation using adversarial learning for pavement crack detection. *IEEE Access.*, vol. 8, pp. 51446–51459, 2020.

[117] G. Li, X. Li, J. Zhou, D. Liu ,and W. Ren. Pixel-level bridge crack detection using a deep fusion about recurrent residual convolution and context encoder network. *Measurement.*, vol. 176, p.109171, 2021.

[118] Z. Li, H. Zhu ,and M. Huang. A deep learning-based fine crack segmentation network on full-scale steel bridge images with complicated backgrounds. *IEEE Access.*, vol. 9, pp. 114989–114997, 2021.

[119] T.Y.Lin, P.Goyal, R. Girshick, K. He, and P. Dollar. Focal Loss for Dense Object Detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318-327, 2020.

[120] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C.L. Zitnick. Microsoft COCO: Common Objects in Context. *European Conf. on Computer Vision*, pp. 740–755, 2018.

[121] Q. Lin, W. Li, X. Zheng, H. Fan, Z. Li. DeepCrackAT: An Effective Crack Segmentation Framework Based on Learning Multi-scale Crack Features. *Engineering Applications of Artificial Intelligence*, vol. 126, p. 106876, 2023.

[122] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg. SSD: Single Shot MultiBox Detector. *European Conf. on Computer Vision*, pp. 21–37, 2016.

[123] Y. Liu, M.M. Cheng, X. Hu, K. Wang, and X. Bai. Richer convolutional features for edge detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3000-3009, 2017.

[124] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li. Deepcrack: A Deep Hierarchical Feature Learning Architecture for Crack Segmentation. *Neurocomputing*, vol. 338, pp. 139–153, 2019.

[125] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li. Deepcrack: A Deep Hierarchical Feature Learning Architecture for Crack Segmentation. *Neurocomputing*, vol. 338, pp. 139–153, 2019.

[126] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen. Deep learning for generic object detection: A survey. *International Journal of Computer Vision*, vol. 128, pp. 261–318, Feb. 2020.

[127] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. *IEEE Int'l Conf. on Computer Vision*, pp. 9992–10002, 2021.

[128] H. Liu, X. Miao, C. Mertz, C. Xu and H. Kong. CrackFormer: Transformer Network for Fine-Grained Crack Detection. *IEEE Int'l Conf. on Computer Vision*, pp. 3763-3772, 2021.

[129] S. Liu et al. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv preprint arXiv:2303.05499.* 2023.

[130] H. Liu et al., "Deep Domain Adaptation for Pavement Crack Detection," in *IEEE Trans. on Intelligent Transportation Systems,* vol. 24, no. 2, pp. 1669–1681, 2023.

[131] C. Maierhofer. "Nondestructive evaluation of concrete infrastructure with ground penetrating radar." *Journal of Materials in Civil Engineering*, vol. 15, no. 3, pp. 287-297, 2003.

[132] B. Marin, K. Brown, and M. S. Erden. Automated Masonry Crack Detection with Faster R-CNN. *IEEE Int'l Conf. on Automation Science and Engineering*, pp. 333–340, 2021.

[133] X. Meng. Concrete Crack Detection Algorithm Based on Deep Residual Neural Networks. *Scientific Programming*, vol. 2021, 2021.

[134] Q. Mei, M. Gul, and M. R. Azim. Densely Connected Deep Neural Network Considering Connectivity of Pixels for Automatic Crack Detection. *Automation in Construction*, vol. 110, p. 103018, 2020.

[135] F. Milletari, N. Navab, and S.A. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *IEEE fourth Int'l Conf. on 3D vision (3DV)*, pp. 565–571, 2016.

[136] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz and D. Terzopoulos. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 44, no. 7, pp. 3523–3542, 2022.

[137] M.A. Mohammed, Z. Han, Y. Li, Z. Al-Huda, C. Li and W. Wang. End-to-end Semi-Supervised Deep Learning Model for Surface Crack Detection of Infrastructures. *Frontiers in Materials*, vol. 9, pp. 1–19, 2022.

[138] M. Mundt, S. Majumder, S. Murali, P. Panetsos and V. Ramesh. Meta-Learning Convolutional Neural Architectures for Multi-Target Concrete Defect Classification With the COncrete DEfect BRidge IMage Dataset. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 11196–11205, 2019.

[139] H.S. Munawar, A.W.A. Hammad, A. Haddad, C.A. Pereira Soares, and T. Waller. Image-Based Crack Detection Methods: A Review. *Infrastructures*, vol. 6, no. 8, p. 115, 2021.

[140] M. Mishra, V. Jain, S.K. Singh, and D. Maity. Two-Stage Method Based on the You Only Look Once Framework and Image Segmentation for Crack Detection in Concrete Structures. *Architecture, Structures and Construction*, vol. 3, pp. 429–446, 2023.

[141] Highway Accident Report: Collapse of I-35W Highway Bridge in Minneapolis, Minnesota. *National Transportation Safety Board*, 178 pages, 2008.

[142] Y. Noh, D. Koo, Y.M. Kang, D. Park ,and D. Lee. Automatic crack detection on concrete images using segmentation via fuzzy C-means clustering. *IEEE International conference on applied system innovation (ICASI)*, pp. 877–880, 2017.

[143] T. Nishikawa, J. Yoshida, T. Sugiyama, and Y. Fujino. Concrete Crack Detection by Multiple Sequential Image Filtering. *Computer-Aided Civil and Infrastructure Engineering*, vol. 27, no. 1, pp. 29–47, 2012.

[144] Y. Nishimura, S. Takahashi, H. Mochiyama, and T. Yamaguchi. Automated Hammering Inspection System With Multi-Copter Type Mobile Robot for Concrete Structures. *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9993–10000, 2022.

[145] H. Oliveira ,and P.L. Correia. Identifying and retrieving distress images from road pavement surveys. *IEEE International Conference on Image Processing*, pp. 57–60, 2008.

[146] Ontario Ministry of Transportation. Ontario Structure Inspection Manual (OSIM), 2008. [Online] Available: `https://www.bv.transports.gouv.qc.ca/mono/1162522.pdf`

[147] Ontario Structure Inspection Manual (OSIM). ISBN 0-7794-0431-9, 2008 (http://www.bv.transports.gouv.qc.ca/mono/1162522.pdf).

[148] T. Omar, and M. Nehdi. Condition Assessment of Reinforced Concrete Bridges: Current Practice and Research Challenges. *Infrastructures*, vol. 3, p. 36, 2018.

[149] M. Pak, and S. Kim. Crack Detection Using Fully Convolutional Network in Wall-climbing Robot. *Advances in Computer Science and Ubiquitous Computing: CSA-CUTE 2019*, vol. 126, pp. 267-272, 2021.

[150] M. Pâques, D. Law-Hine, O.A. Hamedane, G. Magnaval, N. Allezard. Automatic Multi-label Classification of Bridge Components and Defects Based on Inspection Photographs. *ce/papers*, vol. 6, no. 5, pp. 1080–65, 2023.

[151] P. Prasanna, K.J. Dana, N. Gucunski, B.B. Basily, H.M. La, R.S. Lim, H. Parvardeh. Automated crack detection on concrete bridges. *IEEE Transactions on automation science and engineering.*, vol. 13, no. 2, pp. 591–599, 2014.

[152] L. Pei, Z. Sun, L. Xiao, W. Li, J. Sun and H. Zhang. Virtual Generation of Pavement Crack Images based on Improved Deep Convolutional GAN. *Engineering Applications of Artificial Intelligence*, vol. 104, p. 104376, 2021.

[153] M. Petrou, J. Kittler, K.Y. Song. Automatic surface crack detection on textured materials. *Journal of materials processing technology.*, vol. 56, no. 1-4, pp. 158–167, 1996.

[154] Z. Qi, D. Liu, J. Zhang, J. Chen. Micro-concrete Crack Detection of Underwater Structures Based on Convolutional Neural Network. *Machine Vision and Applications*, vol. 33, no. 5, p. 74, 2022.

[155] Z. Qu, J. Mei, L. Liu ,and D.Y. Zhou. Crack detection of concrete pavement with cross-entropy loss function and improved VGG16 network model. *Ieee Access.*, vol. 8, pp. 54564–54573, 2020.

[156] Z. Qu, W. Chen, S.Y. Wang, T.M. Yi, L. Liu. A crack detection algorithm for concrete pavement based on attention mechanism and multi-features fusion. *IEEE Transactions on Intelligent Transportation Systems.* 2021 Aug 30;23(8):11710-9.

[157] Z. Qu, C.Y. Wang, S.Y. Wang, F.R. Ju. A method of hierarchical feature fusion and connected attention architecture for pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems.*, vol. 23, no. 9, pp. 16038–16047, 2022.

[158] R.-S. Rajadurai and S.-T. Kang. Automated Vision-based Crack Detection on Concrete Surfaces Using Deep Learning. *Applied Sciences*, vol. 11, no. 11, p. 5229, 2021.

[159] S. Ren, K. He, R. Girshick, J. Sun. Faster RCNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems*, pp. 91–9928, 2015.

[160] Y. Ren, J. Huang, Z. Hong, W. Lu, J. Yin, L. Zou, and X. Shen. Image-Based Concrete Crack Detection in Tunnels Using Deep Fully Convolutional Networks. *Construction and Building Materials*, vol. 234, p. 117367, 2020.

[161] O. Ronneberger, P. Fischer, and T. Brox. 2015. U-net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

[162] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

[163] J. Redmon, and A. Farhadi. YOLO9000: Better, Faster, stronger. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 7263–7271, 2017.

[164] J. Redmon, A. Farhadi. YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*, 2018 .

[165] J. L. Rose. "Ultrasonic guided waves in structural health monitoring." *Key Engineering Materials*, vol. 270, pp. 14-21, 2004.

[166] S.S. Salehi, D. Erdogmus, A. Gholipour. Tversky loss function for image segmentation using 3D fully convolutional deep networks. *Springer International workshop on machine learning in medical imaging*,pp. 379–387, 2017.

[167] E.A. Shamsabadi, C. Xu, A.S. Rao, T. Nguyen, T. Ngo, D. Dias-da-Costa. Vision Transformer-Based Autonomous Crack Detection on Asphalt and Concrete Surfaces. *Automation in Construction*, vol. 140, p. 104316, 2022.

[168] N.L. Shao, F. Zhu, and X. Li. Transfer Learning for Visual Categorization: A Survey. *IEEE Trans. on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 1019–1034, 2015.

[169] F. Shao, L. Chen, J. Shao, W. Ji, S. Xiao, L. Ye, Y. Zhuang and J. Xiao. Deep Learning for Weakly-Supervised Object Detection and Localization: A Survey. *Neurocomputing*, vol. 496, pp. 192–207, 2022.

[170] H. Sharma and A.S. Jalal. A Survey of Methods, Datasets and Evaluation metrics for Visual Question Answering. *Image and Vision Computing*, vol. 116, p. 104327, 2021.

[171] E. Shelhamer, J. Long, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 39, no. 94, pp 640–651, 2017.

[172] Y. Shi, L. Cui Z. Qi, F. Meng, Z. Chen. Automatic Road Crack Detection Using Random Structured Forests. *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–45, 2016.

[173] P. Shi, S. Shao, X. Fan, Z. Zhou, Y. Xin. MCL-CrackNet: A Concrete Crack Segmentation Network Using Multi-level Contrastive Learning. *IEEE Transactions on Instrumentation and Measurement.*, vol. 72,2023.

[174] S. Shim, J. Kim, G.C. Cho, S.W. Lee. Multiscale and adversarial learning-based semi-supervised semantic segmentation approach for crack detection in concrete structures. *IEEE Access*, vol. 8, pp. 170939-50, 2020.

[175] H.K. Shin, Y.H. Ahn, S.H. Lee, and H.Y. Kim. Automatic Concrete Damage Recognition Using Multi-Level Attention Convolutional Neural Network. *Materials*, vol. 13, no. 23, p. 5549, 2020.

[176] K. Simonyan, and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Int'l Conf. on Learning Representations*, pp. 1-14, 2015.

[177] S. Skansi. Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence. *Springer*, 2018.

[178] C. Song, L. Wu, Z. Chen, H. Zhou, P. Lin, S. Cheng, and Z. Wu. Pixel-Level Crack Detection in Images Using Segnet. *Int'l Conf. on Multi-disciplinary Trends in Artificial Intelligence*, pp. 247–254, 2019.

[179] C. Su, and W. Wang. Concrete Cracks Detection using Convolutional Neural Network Based on Transfer Learning. *Mathematical Problems in Engineering*, vol. 2020, Article ID 7240129, 2020.

[180] P. Subirats, J. Dumoulin, V. Legeay, D. Barba. Automation of pavement surface crack detection using the continuous wavelet transform. *IEEE International Conference on Image Processing.*, pp. 3037–3040, 2006.

[181] Y. Sun, Y. Yang, G. Yao, F. Wei, and M. Wong. Autonomous Crack and Bughole Detection for Concrete Surface Image Based on Deep Learning. *IEEE Access*, Vol. 9, pp. 85709–85720, 2021.

[182] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

[183] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *AAAI Conf. on Artificial Intelligence*, pp. 4278–4284, 2017.

[184] R. Szeliski. Computer Vision: Algorithms and Applications, *Springer*, 2nd Ed, 2022.

[185] M. Tan, and Q.V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Int'l Conf. on Machine Learning*, pp. 6105–6114, 2019.

[186] M. Tan, R Pang and Q.V. Le. Efficientdet: Scalable and efficient object detection. *IEEE Conf. on Computer Vision and Pattern Recognition.* pp. 10781–10790, 2020.

[187] S. Teng, Z. Liu, G. Chen, and L. Cheng. Concrete Crack Detection Based on Well-known Feature Extractor Model and the Yolo v2 Network. *Applied Sciences*, vol. 11, no. 2, p. 813, 2021.

[188] T. T. Teoh. *Convolutional Neural Networks for Medical Applications*. SpringerBriefs in Computer Science, Springer Singapore, 1st ed., 2023. DOI: 10.1007/978-981-19-8814-1.

[189] J.M. Tomczak. *Deep Generative Modeling*. Springer, 2022.

[190] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training Data-Efficient Image Transformers & Distillation Through Attention. *Int'l Conf. on Machine Learning*, pp. 10347–10357, 2021.

[191] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers and A.W.M. Smeulders. Selective Search for Object Recognition. *Int'l J. of Computer Vision*, vol. 104, pp. 154–171, 2013.

[192] J.M.J. Valanarasu and V.M. Patel. UNeXt: MLP-based rapid medical image segmentation network. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 23-33, Springer Nature Switzerland, Cham, September 2022.

[193] A. Vaswani. "Attention is all you need." *Advances in neural information processing systems*, Vol. 30, p. I, 2017.

[194] S. Wakata, N. Hosoya, N. Hasegawa, and M. Nishikino. Defect Detection of Concrete in Infrastructure based on Rayleigh Wave Propagation Penetrated by Laser-Induced Plasma Shock Waves. *Int'l J. of Mechanical Sciences*, Vol. 218, p. 107039, 2022.

[195] X. Wang, R. Girshick, A. Gupta and, K. He. Non-local neural networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7794–7803, 2018.

[196] H. Wan, L. Gao, Z. Yuan, H. Qu, Q. Sun, H. Cheng, R. Wang. A Novel Transformer Model for Surface Damage Detection and Cognition of Concrete Bridges. *Expert Systems with Applications*, Vol. 213, p. 119019, 2023.

[197] W. Wang, C. Su. Convolutional Neural Network-based Pavement Crack Segmentation Using Pyramid Attention Network. *IEEE Access*, vol. 8, p. 206548–206558, 2020.

[198] W. Wang, E. Xie, X. Li, D-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. *IEEE Int'l Conf. on Computer Vision*, pp. 568–578, 2021.

[199] C.Y. Wang, A. Bochkovskiy, and H-Y. M. Liao. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-The-Art for Real-Time Object Detectors. *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 7464-7475, 2023.

[200] J.J. Wang, Y.F. Liu, X. Nie and Y.L. Mo. Deep Convolutional Neural Networks for Semantic Segmentation of Cracks *Structural Control Health Monitoring*, vol. 29, p. e2850, 2022.

[201] W. Wang, C. Su, and D. Fu. Automatic Detection of Defects in Concrete Structures Based on Deep Learning. *Structures*, vol. 43, pp. 192–199, 2022.

[202] Y. Wang, Z. He, X. Zeng, J. Zeng, Z. Cen, L. Qiu, X. Xu, and Q. Zhuo. GGMNet: Pavement-Crack Detection Based on Global Context Awareness and Multi-Scale Fusion. *Remote Sensing*, vol. 16, no. 10, p. 1797, 2024.

[203] S. Woo, J. Park, J.Y. Lee, and I.S. Kweon. Cbam: Convolutional block attention module. *In Proceedings of the European conf. on computer vision (ECCV)*, pp. 3–19, 2018.

[204] P. Wu, A. Liu, J. Fu, X. Ye, and Y. Zhao. Autonomous Surface Crack Identification of Concrete Structures Based on an Improved One-stage Object Detection Algorithm. *Engineering structures*, vol. 1, no. 272, p. 114962, 2022.

[205] C. Xiang, V. J.L. Gan, J. Guo and L. Deng. Semi-Supervised Learning Framework for Crack Segmentation Based on Contrastive Learning and Cross-Pseudo Supervision. *Measurement*,vol. 217,p. 113091, 2023.

[206] X. Xia et al., GAN-based anomaly detection: A review. *Neurocomputing*, vol. 493, pp. 497–535, 2022.

[207] S. Xie and Z. Tu. Holistically-nested edge detection. *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1395-1403, 2015.

[208] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J.M. Alvarez, and P. Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, vol. 34, pp. 12077-12090, 2021.

[209] H. Xu, X. Su, Y. Wang, H. Cai, K. Cui, and X. Chen. Automatic Bridge Crack Detection Using a Convolutional Neural Network. *Applied Sciences*, vol. 9, no. 14, p. 2867, 2019.

[210] B. Xu and C. Liu. Pavement Crack Detection Algorithm Based on Generative Adversarial Network and CNN Under Small Samples. *Measurement*, vol. 196, p. 111219, 2022.

[211] S. Xu, et al. PP-YOLOE: An Evolved Version of YOLO. *arXiv preprint arXiv:2203.16250*, 2022.

[212] D.P. Yadav, S. Chauhan, B. Kada and A. Kumar. Spatial Attention-Based Dual Stream Transformer for Concrete Defect Identification. *Measurement*, vol. 218, p. 113137, 2023.

[213] T. Yamaguchi, S. Nakamura, R. Saegusa, S. Hashimoto. Image-based crack detection for real concrete surfaces. *IEEJ Transactions on Electrical and Electronic Engineering.*, vol. 3, no. 1, pp. 128–135, 2008.

[214] T. Yamaguchi, S. Hashimoto. Fast crack detection method for large-size concrete surface images using percolation-based image processing. *Machine Vision and Applications.*, vol. 21, pp. 797–809, 2010.

[215] L. Yang, B. Li, W. Li, Z. Liu, G. Yang, and J. Xiao. Deep Concrete Inspection Using Unmanned Aerial Vehicle Towards CSSC Database. *IEEE Int'l Conf. on Intelligent Robots and Systems*, pp. 24–28, 2017.

[216] X. Yang, H. Li, Y. Yu, X. Luo, T. Huang, and X. Yang. Automatic Pixel-level Crack Detection and Measurement Using Fully Convolutional Network. *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 12, pp. 1090–1109, 2018.

[217] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, H. Ling. Feature pyramid and hierarchical boosting network for pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1525–35, 2019.

[218] Q. Yang, W. Shi, J. Chen, W. Lin. Deep Convolution Neural Network-based Transfer Learning Method for Civil Infrastructure Crack Detection.*Automation in Construction*, vol. 116, p. 103199, 2020.

[219] J. Yang, H. Li, J. Zou, S. Jiang, R. Li ,and X. Liu. Concrete crack segmentation based on UAV-enabled edge computing. *Neurocomputing.*, vol. 485, pp. 233–241, 2022.

[220] X. Yang , Z. Song , I. King and Z. Xu. A Survey on Deep Semi-Supervised Learning. *IEEE Trans. on Knowledge and Data Engineering*, vol. 35, no. 9, pp. 8934–8954, 2023.

[221] D. Yapi, M.S. Allili and N. Baaziz. Automatic Fabric Defect Detection Using Learning-Based Local Textural Distributions in the Contourlet Domain. *IEEE Trans Automatioin Science and Engineering,* vol. 15, no. 3, pp. 1014-1026, 2018.

[222] G. Yao, F. Wei, Y. Yang, Y. Sun, . Deep-Learning-Based Bughole Detection for Concrete Surface Image. *Advances in Civil Engineering*, Article ID 8582963, 2019.

[223] Z. Yao, J. Xu, S. Hou and M.C. Chuah. Cracknex: a few-shot low-light crack segmentation model based on retinex theory for uav inspections. *arXiv preprint arXiv:2403.03063.*2024.

[224] X.W. Ye, T. Jin, Z. Li, S. Ma, Y. Ding, and Y. Ou. Structural Crack Detection from Benchmark Data Sets Using Pruned Fully Convolutional Networks. *Journal of Structural Engineering*, vol. 147, no. 11, p. 04721008, 2021.

[225] L. Zhang, F. Yang, Y.D. Zhang, and Y.J. Zhu. Road Crack Detection Using a Deep Convolutional Neural Network. *IEEE Int'l Conf. on Image Processing*, pp. 3708–3712, 2016.

[226] Z. Zhang, M. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 2018.

[227] C. Zhang, C.C. Chang, and M. Jamshidi. Concrete Bridge Surface Damage Detection Using a Single-Stage Detector. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 35, no. 4, pp. 389–409, 2020.

[228] X. Zhang, D. Rajan, and B. Story. Concrete crack detection using context-aware deep semantic segmentation network. *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 11, pp. 951–971, 2019.

[229] Y. Zhang, J. Wu, Q. Li, X. Zhao and M. Tan. Beyond Crack: Fine-Grained Pavement Defect Segmentation Using Three-Stream Neural Networks. *IEEE Trans. On Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14820–14831, 2022.

[230] K. Zhang, Y. Zhang and H.-D. Cheng. "CrackGAN: Pavement Crack Detection Using Partially Accurate Ground Truths Based on Generative Adversarial Learning. *IEEE Trans. on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 1306-1319, 2021.

[231] J. Zhang, S. Qian, C. Tan. Automated Bridge Surface Crack Detection and Segmentation Using Computer Vision-based Deep Learning Model. *Engineering Applications of Artificial Intelligence*, vol. 115, p. 105225, 2022.

[232] C. Zhang, M.M. Karim, and R. Qin. A Multitask Deep Learning Model for Parsing Bridge Elements and Segmenting Defect in Bridge Inspection Images. *Transportation Research Record*, vol. 2677, no. 7, pp. 693–704, 2023.

[233] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2881-2890, 2017.

[234] J. Zhong, J. Huyan, W. Zhang, H. Cheng, J. Zhang, Z. Tong, X. Jiang, B. Huang. A Deeper Generative Adversarial Network for Grooved Cement Concrete Pavement Crack Detection. *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105808, 2023.

[235] J. Zhu, C. Zhang, H. Qi, and Z. Lu. Vision-Based Defects Detection for Bridges Using Transfer Learning and Convolutional Neural Networks. *Structure and Infrastructure Engineering*, vol. 16, no. 7, pp. 1037–1049, 2020.

[236] W. Zhu, H. Zhang, J. Eastwood, X. Qi, J. Jia and Y. Cao. Concrete Crack Detection Using Lightweight Attention Feature Fusion Single Shot Multibox Detector. *Knowledge-Based Systems*, Volume 261, p. 110216, 2023.

[237] Q. Zou, Y. Cao, Q. Li, Q. Mao, S. Wang. CrackTree: Automatic crack detection from pavement images. *Pattern Recognition Letters.*, vol. 33, no. 3, pp. 227–238, 2012.

[238] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang. Deepcrack: Learning Hierarchical Convolutional Features for Crack Detection. *IEEE Trans. on Image Processing*, vol. 28, no. 3, pp. 1498–1512, 2019.

[239] D. Zou, M. Zhang, Z. Bai, T. Liu, A. Zhou, X. Wang, W. Cui, and S. Zhang. Multicategory Damage Detection and Safety Assessment of Post-earthquake Reinforced Concrete Structures Using Deep Learning. *Computer-Aided Civil and Infrastructure Engineering*, vol. 37, no. 9, pp. 1188–1204, 2022.

[240] H. Zoubir, M. Rguig, M. El Aroussi, A. Chehri, R. Saadane, and G. Jeon. Concrete Bridge Defects Identification and Localization Based on Classification Deep Convolutional Neural Networks and Transfer Learning. *Remote Sensing*, vol. 14, p. 4882, 2022.