

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

RAPPORT TECHNIQUE FINAL
PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
DANS LE CADRE DU PROJET DE FIN D'ÉTUDES
DU BACCALAURÉAT EN GÉNIE DE LA PRODUCTION AUTOMATISÉE

CONCEPTION D'UN SYSTÈME AUTOMATISÉ D'ANALYSE DE RÉSEAU
DE VIDES D'AIR DU BÉTON

PAR
ÉRIC SURPRENANT
FÉLIX CÔTÉ

MONTRÉAL, le 20 août 2022

Remerciements

Ce travail est l'aboutissement d'un parcours universitaire de plusieurs années qui nous a permis de mettre en pratique les connaissances et expériences acquises au cours de celui-ci. Ce parcours n'aurait pu être possible sans les enseignements des différents professeurs rencontrés durant celui-ci ainsi que du soutien de la communauté de l'ÉTS. Nous tenons à remercier ces nombreuses personnes pour leur implication dans notre cheminement scolaire et professionnel.

Nous tenons à remercier spécialement Jean-Philippe Roberge, professeur de l'ÉTS, pour avoir accepté de superviser ce projet final d'étude. Son expertise dans le domaine de la robotique et de la vision artificielle nous a permis de bénéficier de conseils inestimables. Sa grande disponibilité ainsi que son souci à l'égard notre réussite furent grandement appréciés.

Ce projet n'aurait pu être possible sans la contribution de Groupe ABS. Nous les remercions d'avoir accepté d'endosser notre projet et d'avoir participé activement à celui-ci. Nous tenons à souligner spécialement la contribution d'Edith Prud'homme, Benoit St-Pierre et Philip Sawoszczuk qui ont pris le temps de partager avec nous leurs expériences sans lesquelles il aurait été impossible de maîtriser toutes les subtilités de l'essai ASTM C457.

Résumé

Dans le cadre de ce projet de fin d'étude réalisé à l'École de technologie supérieure, nous proposons un système permettant d'automatiser l'analyse du réseau de vides d'air de béton durci (essai ASTM C457). Ce système a pour objectif de réduire l'intervention humaine dans cet essai et ainsi réduire son coût et le risque d'erreur. La conception du système doit respecter les exigences de la norme ASTM C457. À cet effet, celle-ci prévoit une méthode (*Procedure C, Contrast Enhanced Method*) permettant d'automatiser la procédure. L'équipe responsable du développement de ce système est composée d'Éric Surprenant et Félix Côté (étudiant), Jean-Philippe Roberge (enseignant superviseur) et les laboratoires Groupe ABS (promoteurs du projet). La conception de ce système passera par l'essai et la comparaison de différents algorithmes d'assemblage et de segmentation d'images afin de déterminer les plus performants et précis. Pour évaluer ces algorithmes, un prototype fut assemblé à partir d'une ancienne imprimante 3D et d'un microscope digital. Ce prototype nous a permis de faire l'acquisition d'images qui ont été utilisées pour tester les algorithmes. Un *package* Python fut aussi développé pour asservir l'imprimante et le microscope. Les concepts de programmation orientée objet ont été utilisés pour sa conception afin de pouvoir l'adapter facilement à une version commerciale.

L'analyse de résultats obtenus à partir des tests de performance et de précision (similarité) nous a permis de noter plusieurs avantages et inconvénients à ces algorithmes. Malgré tout, l'algorithme d'assemblage d'images MIST et la technique de segmentation sémantique avec un modèle d'apprentissage profond se sont avérés être supérieurs en plusieurs égards. Concernant l'algorithme MIST, celui-ci a su générer des « assemblages » d'images avec peu d'artéfacts et avec un temps de traitement très court. De plus, le faible niveau de chevauchement requis entre les images a permis de réduire considérablement le nombre total d'images requis lors d'une séquence d'analyse. Cela permet alors de réduire le temps de déplacement et de traitement. Cependant, puisque l'algorithme MIST ne bénéficie pas d'une interface facilement accessible en Python, une implémentation de celui-ci à notre système devrait être fait. Ensuite, concernant le modèle de segmentation, un processus d'apprentissage supervisé avec une banque de 40 images identifiées fut utilisé pour entraîner notre modèle. De façon générale, un réseau neuronal nécessite une banque d'images beaucoup plus grande pour atteindre un niveau de confiance suffisamment élevé. Néanmoins, nous avons été en mesure d'obtenir des résultats satisfaisants considérant cette limitation. C'est en considérant la précision obtenue par ce modèle et le peu d'entraînement qui lui a été prodigué que nous avons conclu que cette technique de segmentation est celle offrant le plus de potentiel.

Table des matières

1 Définition de la problématique	2
1.1 L'essai ASTM C457	2
1.1.1 Pourquoi	2
1.1.2 Comment	3
1.2 Méthode actuellement employée par Groupe ABS	5
1.3 Problématique	7
2 Définition des objectifs	8
3 Revue de littérature	10
3.1 Systèmes d'analyse de réseau de vide d'air existants	10
3.2 Méthode (C) par contraste augmenté	11
3.3 Équipements requis pour la conception d'un système d'acquisition d'image	12
3.4 Algorithmes d'assemblage d'images	13
3.4.1 Algorithme MIST	13
3.4.2 Classe « Stitcher » de la librairie OpenCV	14
3.5 Techniques de segmentation	15
4 Méthodologie de travail	17
4.1 Étape 1 - Concevoir et assembler un prototype d'acquisition d'image	18
4.2 Étape 2 - Développer un logiciel pour contrôler le système d'acquisition	19
4.3 Étape 3 - Choisir un algorithme d'assemblage d'images	20
4.4 Étape 4 - Développer un modèle de détection des caractéristiques	21
4.4.1 Calculer la grosseur de pixel	21
4.4.2 Segmenter l'image	21
4.4.3 Évaluer les caractéristiques désirées	22
4.5 Étape 5 - Développer des méthodes de validation des résultats	23
5 Processus de conception	24
5.1 Conception du prototype	24
5.1.1 Première version	24
5.1.2 Deuxième version	25
5.2 Conception du logiciel	26

5.2.1	Fonctionnement général	26
5.2.2	Module « Microscope »	26
5.2.3	Module « Stage »	27
5.2.4	Module « Analyser »	28
5.3	Conception du modèle de détection des caractéristiques	30
5.3.1	Calculer la grosseur de pixel	30
5.3.2	Segmenter l'image	31
6	Interprétation des résultats et discussion	36
6.1	Résultats l'algorithme d'assemblage d'images	36
6.2	Résultats du modèle de détection des caractéristiques	39
6.2.1	Fonction pour calculer la grosseur de pixel	39
6.2.2	Segmentation conventionnelle	40
6.2.3	Segmentation sémantique avec « Deep Network »	43
7	Analyse financière de la solution	45

Table des figures

1.1	Vue de microscope d'un échantillon de béton fissuré	2
1.2	Diagramme vue d'ensemble de l'essai	4
1.3	Trajet de la méthode (A)	5
1.4	Installation pour effectuer l'essai ASTM C457	6
2.1	Objectifs dans une perspective de développement durable	9
3.1	Fonctionnement algorithme MIST	13
3.2	Familles de segmentations	15
4.1	Planification de projet	17
5.1	Prototype version 1	24
5.2	Prototype version 2	25
5.3	Diagramme de classe UML	29
5.4	Architecture du modèle d'apprentissage profond	35
6.1	Assemblage OpenCV	36
6.2	Assemblage MIST	36
6.3	Assemblage OpenCV agrandit	36
6.4	Assemblage MIST agrandit	36
6.5	Correspondance de descripteurs invariants OpenCV	37
6.6	Menu du plugin MIST dans le logiciel FIJI	38
6.7	Bande 0.076mm après conversion à niveau de gris	39
6.8	Bande 0.076mm après seuillage ($i < 90$)	39
6.9	Bande 0.076mm après opérations morphologiques	39
6.10	Segmentation automatique avec k-mean	40
6.11	Segmentation par seuillage avec couleur RGB	41
6.12	Segmentation par region avec algorithm de croissance par région	41
6.13	Segmentation de détection par contour avec Canny	42
6.14	Segmentation sémantique avec « Deep learning »	44

Liste des tableaux

3.1	Modèles de microscope digital	12
3.2	Modèles de plateforme XY	12
3.3	Comparaison segmentation	16
6.1	Comparaison segmentation	40
6.2	Résultat performance modèle	43
7.1	Comparaison	45
7.2	Impacts	45
7.3	Investissements	46
7.4	Rentabilité	46

Introduction

Actuellement, au Québec, un grand virage technologique s'opère dans tous les domaines de notre société. Que ce soit dans le domaine médical, agroalimentaire, militaire, construction, pour n'en nommer que quelques-uns. Ce virage est essentiel pour les entreprises québécoises afin d'augmenter leur efficacité pour rester compétitive ou encore pallier la pénurie de main-d'œuvre qui se fait grandement ressentir ces dernières années. Le gouvernement est bien au fait de cet enjeu et met en place plusieurs incitatifs pour encourager les entreprises à faire la transition vers l'industrie 4.0. Malgré tout, on remarque que certains domaines/secteurs n'ont toujours pas emboîté le pas, souvent par manque d'expertise. C'est dans ce contexte social, économique et politique que l'entreprise Groupe ABS s'est donné pour mission, en 2021, de créer un département de recherche et développement. Cette initiative a été mise en place afin de donner les moyens à l'entreprise de faire la transition vers cette nouvelle ère technologique.

En mai 2021, Groupe ABS a profité de son partenariat avec l'École de technologie supérieure (ÉTS), déjà bien établi dans le domaine du génie de la construction, pour trouver un candidat dans le domaine du génie de la production automatisé qui pourrait les aider à atteindre leurs objectifs d'innovation technologique. C'est dans ce contexte qu'Éric, l'un des deux étudiants participant à ce projet de fin d'études, a été embauché à titre de stagiaire en optimisation des opérations. Ce temps passé dans l'entreprise a permis à Éric d'observer plusieurs procédés et processus pouvant être optimisés. Parmi ceux-ci, un essai a retenu son attention et a été choisi pour en faire le sujet de ce projet de fin d'études (PFE). L'essai en question se nomme : *Détermination microscopique des caractéristiques du réseau de vides d'air du béton durci (ASTM C457)*. L'automatisation de cet essai passe par la conception d'un modèle de vision artificielle, un domaine pour lequel les deux étudiants de ce projet accordent un grand intérêt.

Outre l'intérêt porté pour ce domaine, trois facteurs rendent ce sujet d'autant plus intéressant. Premièrement, le poste qu'occupe Éric dans l'entreprise permet un accès facile aux données et statistiques relatives à cet essai. Celles-ci sont nécessaires pour le développement d'une solution optimale. Deuxièmement, l'ÉTS comporte une communauté scientifique reconnue pour leur expertise dans le domaine de la vision artificielle et dont l'équipe de ce projet peut tirer profit. Troisièmement, les deux étudiants participant à ce projet ont acquis une expérience dans le domaine de la vision artificielle à travers leur parcours universitaire à l'ÉTS. L'expérience déjà acquise dans ce domaine permet à l'équipe d'entrevoir des solutions donnant une direction plus précise à ce projet. Ceux-ci espèrent pouvoir mettre à profit les connaissances acquises durant leur parcours à l'ÉTS pour développer une ou plusieurs solutions pouvant être retenues par Groupe ABS afin d'automatiser l'essai *ASTM C457* et ainsi contribuer à l'avancement de leur mission d'optimisation des opérations.

Définition de la problématique

1.1 L'essai ASTM C457

1.1.1 Pourquoi

Cet essai permet de faire du contrôle de qualité sur les mélanges de béton avant ou après son utilisation. Concrètement, les paramètres du système de vide d'air du béton durci déterminés par cet essai sont liés à la susceptibilité de la portion de pâte de ciment du béton à être endommagée par le gel et le dégel. Le phénomène de gel/dégel s'explique ainsi : lors du durcissement du béton, des bulles d'air se forment dans la pâte. Ces bulles sont susceptibles d'accumuler de l'eau qui va prendre de l'expansion lors du gel et fissurer ainsi la pâte. Ces fissures vont, à leur tour, permettre la propagation de l'eau vers d'autres bulles d'air et ainsi amplifier graduellement la détérioration du béton lors des cycles de gel/dégel. Ces fissures se propagent facilement dans la pâte, mais peuvent difficilement fissurer une roche. La figure 1.1 illustre bien ce phénomène. Nous pouvons voir dans celle-ci que les fissures se propagent d'une bulle à l'autre tout en contournant les roches.

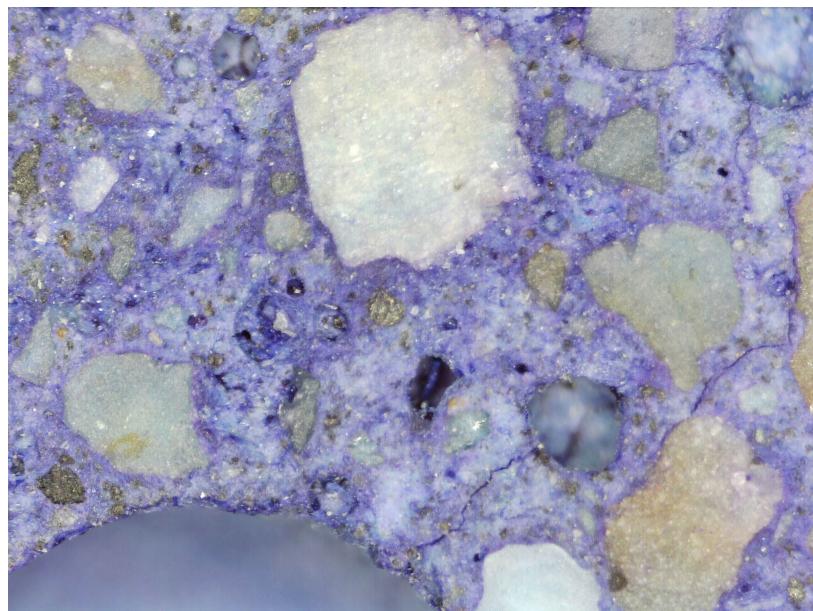


FIGURE 1.1 – Vue de microscope d'un échantillon de béton fissuré

1.1.2 Comment

Pour évaluer la résistance du béton au gel et au dégel cycliques, trois paramètres sont évalués :

- Teneur en air (%)
- Fréquence des vides (vide/cm)
- Ration pâte/air (%)

Selon la norme, trois méthodes sont possibles pour évaluer ces paramètres, la méthode par traverse linéaire (A), la méthode par point d'arrêt (B) et la méthode par contraste augmenté (C). Groupe ABS et l'industrie en général utilise la méthode (A). La méthode (C) est la dernière ayant été ajoutée à la norme et est celle permettant d'automatiser cet essai. Toutefois, avant d'entrer dans les détails de ces méthodes, il est nécessaire d'avoir une idée générale de l'essai.

Le diagramme à la page suivante donne une vue d'ensemble des processus et procédés qui concernent l'essai ASTM C457. Globalement, toutes les étapes qui précèdent la dernière (détermination des paramètres) sont effectuées afin de préparer l'échantillon à l'analyse ainsi que récupérer des données supplémentaires. Ces données sont nécessaires afin de comprendre les raisons qui ont contribué aux résultats (ex. : le client doit savoir pourquoi son béton affiche une mauvaise résistance au gel/dégel.) À noter que ce projet s'attardera uniquement à l'automatisation de la dernière étape concernant la détermination des paramètres du réseau de vide d'air. De plus, seule la méthode actuellement employée par groupe ABS (méthode A) ainsi que la méthode considérée pour l'automatisation (méthode C) seront abordées.

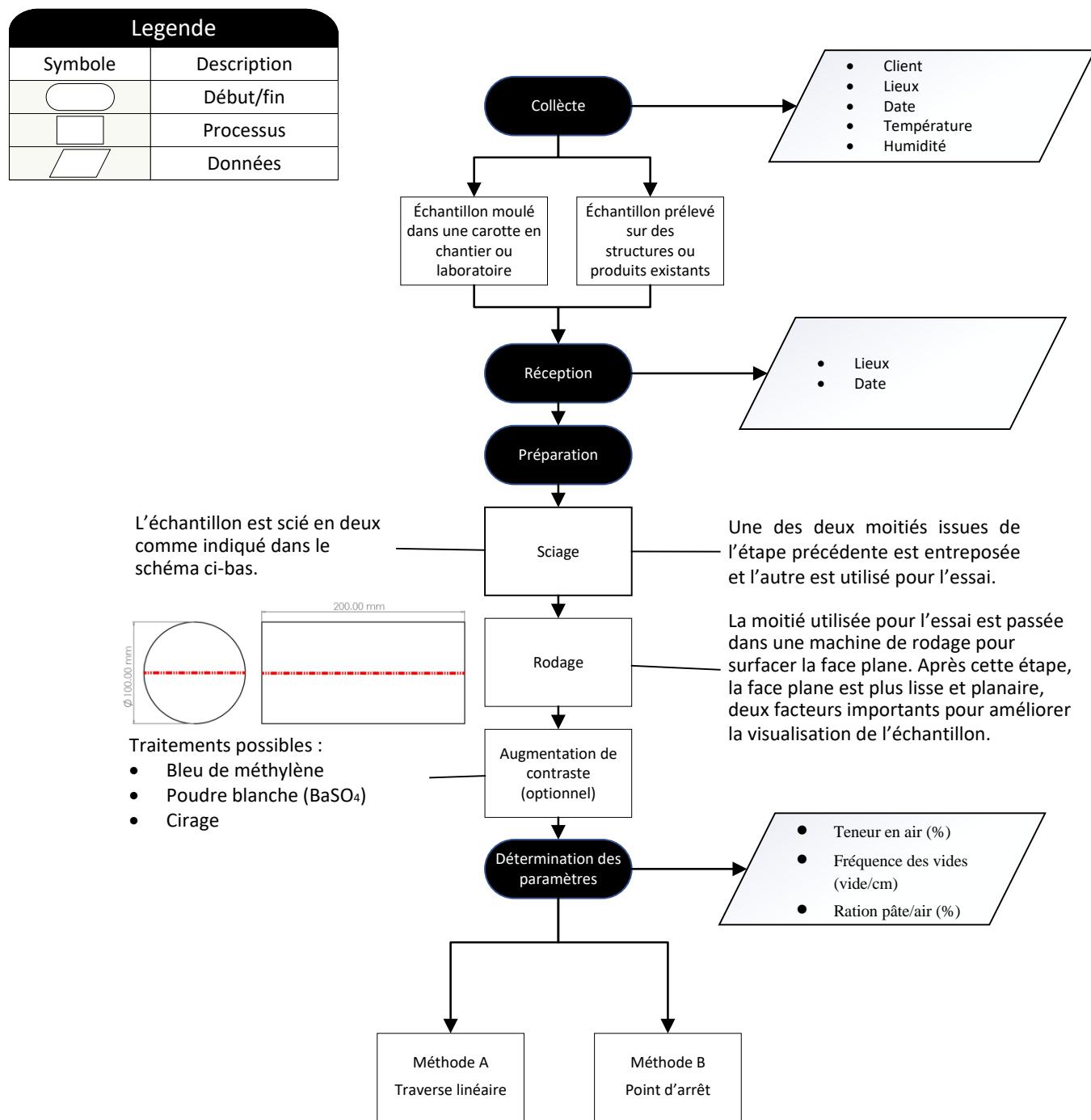


FIGURE 1.2 – Diagramme vue d'ensemble de l'essai

1.2 Méthode actuellement employée par Groupe ABS

Comme il a été mentionné précédemment, la méthode (A) est celle employée par Groupe ABS. Celle-ci consiste à effectuer des traverses linéaires dans le sens de la largeur de la surface plane à l'étude, puis répertorier les données suivantes :

- Nombre total de vides d'air intersectés
- La distance parcourue dans de l'air
- La distance parcourue dans de la pâte
- La distance totale parcourue

La Figure 1.3 représente le parcours réalisé par microscope sur l'échantillon, les lignes bleues sont ce qu'on appelle une « traverse » et les lignes vertes sont la distance séparant celles-ci.

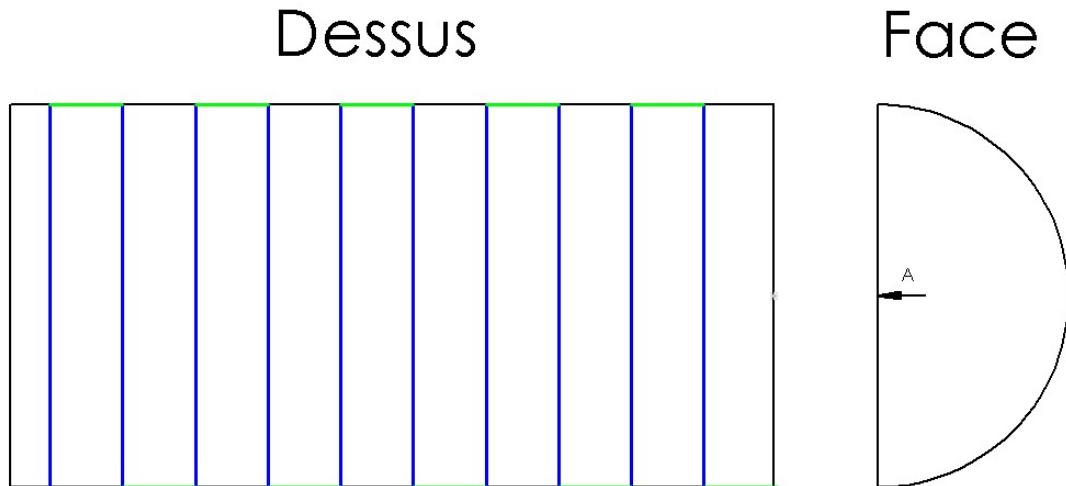


FIGURE 1.3 – Trajet de la méthode (A)

La distance totale parcourue (nombre de traverses * distance d'une traverse) est déterminée à l'aide du Tableau 2 de la norme. Le nombre de traverses est déterminé par la distance totale parcourue requise (distance totale requise divisée par la distance d'une traverse). Cela assure une prise de données réparties uniformément sur l'entièreté de la surface. Les données collectées sont ensuite utilisées pour calculer les paramètres désirés à l'aide des formules détaillées dans la norme.

Pour accomplir cette procédure, un auxiliaire de laboratoire installe l'échantillon sur une plateforme XY permettant de déplacer celui-ci sous le microscope. Le microscope est muni d'une caméra pour projeter le point de vue focal sur un écran à partir duquel l'auxiliaire peut évaluer les caractéristiques. Un compteur mécanique est utilisé pour compter les caractéristiques observées lors des traverses. Voici un exemple de ce type d'installation :

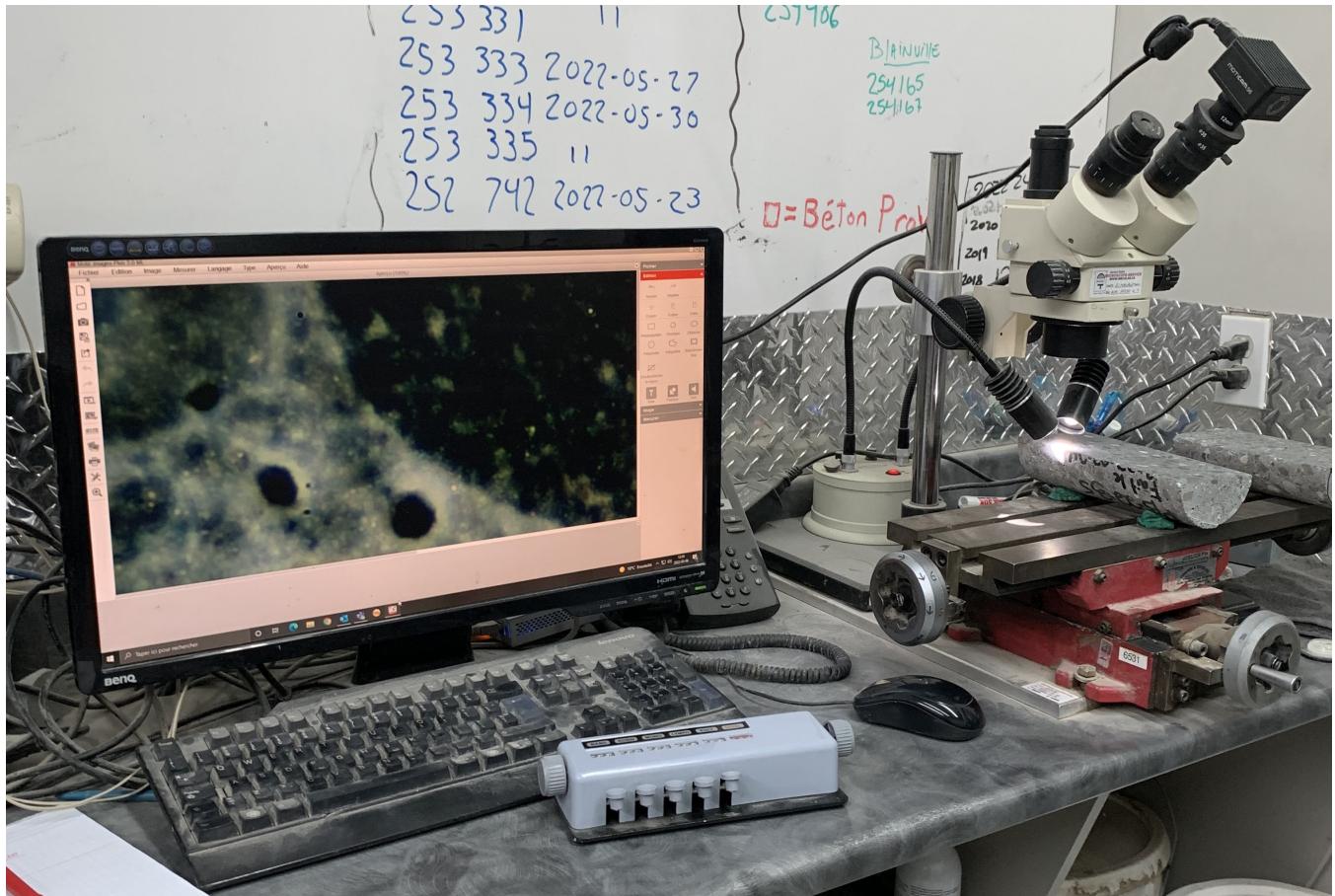


FIGURE 1.4 – Installation pour effectuer l'essai ASTM C457

1.3 Problématique

Cet essai nécessite qu'un technicien réalise chaque traverse tout en comptant les caractéristiques depuis l'écran. Cette opération prend en moyenne 3h d'observation afin d'être complétée. Pour Groupe ABS, avoir un système automatisé pour effectuer cet essai n'est plus optionnel pour trois raisons :

1. Le manque de main-d'œuvre combiné au caractère peu attrayant de cet essai (répétitif et posture statique) fait en sorte qu'il est difficile de trouver des auxiliaires prêts à le faire. Ceci limite la capacité de traitement et par conséquent augmente les délais de livraison des résultats. Pour certains clients, les délais sont critiques et cela les pousse souvent à aller vers d'autres laboratoires.
2. L'interprétation des résultats est très subjective, car la norme est vague sur certaines spécifications. Par exemple, le grossissement recommandé est entre $x50$ et $125x$. Cet intervalle assez significatif change considérablement la perception des caractéristiques observées par l'auxiliaire. De plus, il n'y a aucun barème clair pour différencier les caractéristiques à l'étude telles qu'un vide d'air versus un vide laissé par une roche. De ce fait, d'un auxiliaire à l'autre, les résultats peuvent grandement différer avec la méthode actuelle.
3. Le fait que cet essai soit long à réaliser le rend onéreux. C'est aussi ce qui rend l'automatisation de celui-ci très intéressante, car réduire le temps requis par un humain permet de réduire le coût considérablement. Sachant cela, il devient évident que si Groupe ABS n'automatise pas rapidement cet essai, un autre laboratoire va le faire et être en mesure d'offrir des tarifs beaucoup plus concurrentiels.

Étant au fait de ces dernières réalités, Groupe ABS tente, depuis plusieurs années, de trouver un système automatisé clé en main par le biais de distributeurs d'équipements de laboratoire, mais en vain. C'est pourquoi l'entreprise a maintenant besoin d'une équipe d'ingénieurs ayant une expertise dans le domaine de la vision artificielle afin de développer en collaboration une solution fonctionnelle et efficace tout en respectant les normes établies pour cet essai.

Définition des objectifs

L'entreprise Groupe ABS pourrait grandement bénéficier de l'automatisation de son essai *Determination microscopique des caractéristiques du réseau de vides d'air du béton durci (ASTM C457)*. À l'heure actuelle, cet essai nécessite qu'un auxiliaire de laboratoire compte manuellement les caractéristiques depuis la redirection graphique d'un microscope digital. Non seulement cette tâche est très monotone et grandement susceptible à l'erreur humaine, mais elle engendre aussi des frais de salaire important rendant ainsi l'essai très dispendieux pour les clients. L'objectif de notre équipe est d'automatiser à 100% l'analyse des caractéristiques et les calculs qui s'en suivent pour :

- Améliorer les conditions de travail des auxiliaires de laboratoire en éliminant cette tâche monotone et peu apprécié de leurs responsabilités ;
- Éliminer le risque d'erreur humaine, augmentant ainsi la fiabilité des résultats ;
- Réduire les coûts en salaire de cet essai et ainsi permettre à Groupe ABS d'augmenter leur marge de profits ou réduire leur prix pour aller chercher plus de clients.
- Augmenter les standards de l'industrie en développant une méthode offrant une précision supérieure, contribuant ainsi au rayonnement de Groupe ABS ;

L'automatisation de cet essai et la recherche et le développement nécessaires pour y arriver doivent néanmoins se plier à plusieurs contraintes, qui sont les suivantes :

- Le processus d'analyse automatisé doit respecter la norme ASTM-C457 ;
- Développer un prototype avec un budget de 500\$;
- Développer une solution pouvant être fabriquée avec un budget de 12 000\$;

Le développement durable occupe une place importante dans notre société de nos jours et le domaine de l'ingénierie n'y fait pas exception. Il devient donc important, lorsqu'un projet est préparé, de considérer l'impact que celui-ci aura selon les trois aspects du développement durable. Dans la figure , un diagramme de Venn a été fait pour illustrer comment les objectifs de ce projet s'intègrent dans une perspective de développement durable.

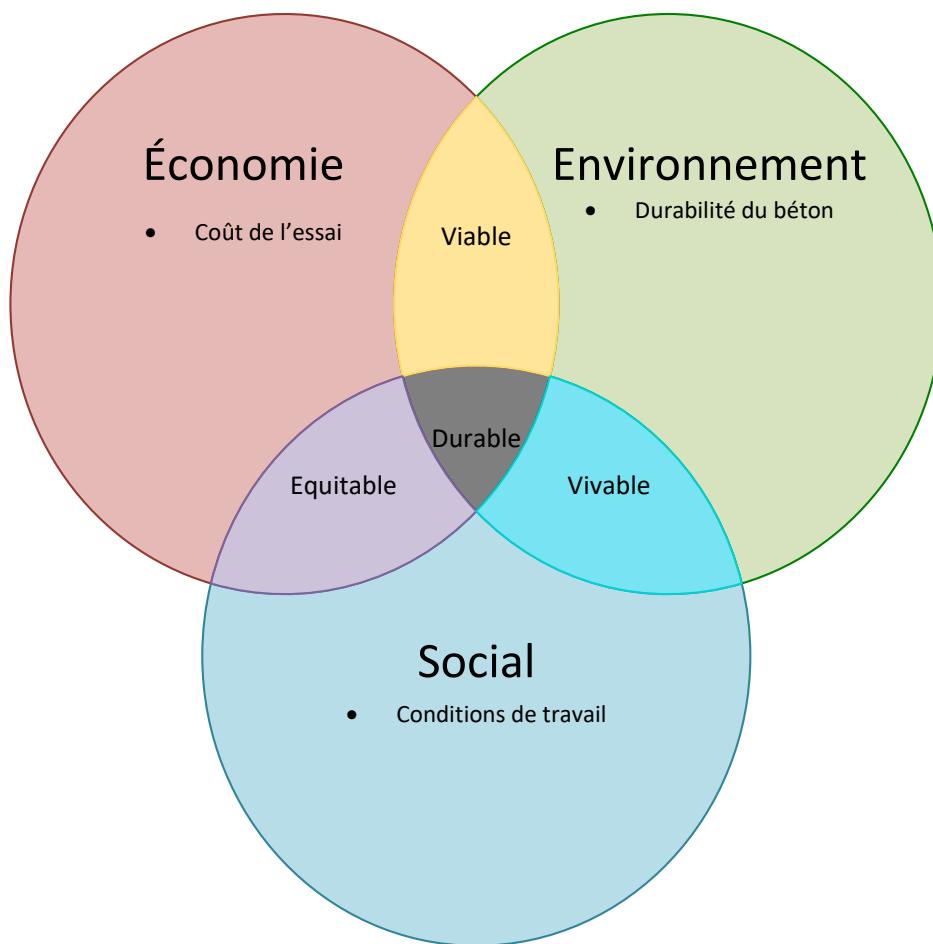


FIGURE 2.1 – Objectifs dans une perspective de développement durable

Coût de l'essai : Réduire le temps requis pour réaliser l'essai permet d'améliorer la performance économique de celui-ci.

Durabilité du béton : Améliorer la précision et la fiabilité de l'essai permet de prévenir plus efficacement l'usage de béton peu résistant au cycle de gel/dégel et ainsi augmenter la durabilité des infrastructures (moins de consommation de béton).

Conditions de travail : Éliminer les tâches monotones et statiques associées à cet essai améliore les conditions de travail des employés et permet de faire ainsi un développement équitable.

Revue de littérature

3.1 Systèmes d'analyse de réseau de vide d'air existants

Comme soulevé dans la définition de la problématique, Groupe ABS a tenté de trouver un système automatisé clé en main, mais cela a été sans succès. C'est pourquoi l'entreprise s'est tournée vers un développement à l'interne. Malgré tout, une recherche a été faite pour vérifier s'il n'existe bel et bien aucun système répondant à leurs besoins.

À l'heure actuelle, plusieurs systèmes d'analyse de réseau de vide d'air automatisé existent. Ces systèmes sont composés d'une plateforme motorisée XY ainsi que d'un microscope digital. Certains permettent seulement d'assister au comptage des caractéristiques tel que le HS-745T Air Voids Analysis System de Hoskin Scientifique Ltd. [12]. D'autres permettent de le faire de façon entièrement automatique à l'aide d'un logiciel spécialisé. Deux systèmes ayant ce type de logiciels ont été répertoriés dans nos recherches.

Le premier est le Linear Traverse Machine H-2964A fabriqué par Humboldt [1]. En contactant un distributeur pour avoir plus d'information (GENEQ Inc.), il a été appris que le logiciel de ce modèle est très problématique et ne permet pas de faire une analyse automatique assez précise. D'ailleurs, le distributeur ne vend plus ce produit pour ces dernières raisons.

Le deuxième modèle est le RapidAir 457 de Germann Instruments [5]. Plusieurs articles de recherche scientifique [11], [9], [10], affirment que ce système permet d'obtenir une meilleure répétabilité et reproductibilité qu'avec la méthode manuelle, et ce, en seulement 15 minutes (excluant la préparation). Toutefois, celui-ci requiert une préparation plus longue pour augmenter le contraste des vides d'air et ne permet pas de calculer le ratio de pâte automatiquement. Ce dernier point est important, car l'entièreté des essais réalisés par Groupe ABS nécessite le ratio de pâte.

Après considération, la recherche et le développement effectués dans ce projet de fin d'études pour développer un nouveau système automatisé sont encore justifiables, à condition que notre solution puisse calculer le ratio de pâte automatiquement.

3.2 Méthode (C) par contraste augmenté

Maintenant qu'il est clair qu'un nouveau système doit être conçu, il est important de vérifier s'il est possible d'automatiser l'essai tout en respectant la norme. De plus, s'il est possible de le faire, les spécifications requises doivent être établies clairement pour diriger le développement vers une solution respectant celles-ci. Comme il a été soulevé dans la description de l'essai à la première section, la méthode (C) « Procedure C - Contrast enhanced method » ayant été ajoutée dans la dernière version de la norme est celle permettant un certain degré d'automatisation. Celle-ci permet dorénavant d'utiliser des modèles de vision artificielle pour faire l'analyse de l'échantillon automatiquement. Voici plusieurs points clés à retenir de cette méthode.

- **Équipements** : Aucune contrainte n'est mentionnée sur les spécifications d'équipement devant être utilisé.
- **Préparation** : Pour faciliter la segmentation et réduire le risque d'erreur, deux produits doivent être appliqués pour augmenter le contraste :
 - ◊ De la poudre blanche faite avec du sulfate de baryum, de la wollastonite ou du titane de dioxyde avec une taille médiane de particules de $2\text{-}3 \mu\text{m}$ peut être appliquée pour remplir les vides et augmenter leur contraste ;
 - ◊ De l'encre (bleu de méthylène) peut être appliquée sur la surface pour augmenter le contraste de la pâte si le ratio de pâte est requis ;
- **Méthode d'acquisition** : Les surfaces à l'étude peuvent être acquises depuis une ou plusieurs images assemblées.
- **Méthode d'analyse** : Une fois l'image segmentée, le même principe que la méthode A doit être appliqué. Une ligne de traverse doit être tracée au centre de l'image et les paramètres requis doivent être calculés à partir de celle-ci.
- **Paramètres à évaluer** :
 - ◊ R = Dimension spatiale d'un pixel (mm)¹
 - ◊ Tt = Distance totale traversée (mm)
 - ◊ Ta = Distance traversée sur vide d'air (mm)
 - ◊ N = Nombre de vide d'air traversé par la ligne de traverse
 - ◊ Pa = Distance traversée sur pâte (mm) (OPTIONNEL)

1. Dois être fourni soit par le fabricant ou établi par l'opérateur. Si R est constant tout au long de l'acquisition, celui-ci ne doit être établi qu'une seule fois. Cependant, s'il varie (ex. : le grossissement change), il doit être établi à chaque acquisition.

3.3 Équipements requis pour la conception d'un système d'acquisition d'image

Pour la fabrication du système, deux options sont possibles. La première option serait d'acheter les pièces séparément et d'assembler le système. La deuxième option serait d'acheter un système déjà tout assemblé. Le Linear Traverse Machine H-2964A cité dans la section 3.1 pourrait être une option viable. Il ne resterait qu'à concevoir le logiciel pour faire l'analyse automatique. Si la première option était choisie, plusieurs choix d'équipements seraient possibles. Voici une liste préliminaire obtenue à la suite d'une recherche non exhaustive de différents modèles d'équipements :

TABLE 3.1 – Modèles de microscope digital

Modèle	Résolution	Fréquence	Grosseur de pixel	Port de communication
INFINITY HD [2]	1920x1080	60 fps	2.7 μm^2	USB 2.0
pco.panda 26 DS [4]	2560x2560	12 fps	2.5 μm^2	USB 3.1

TABLE 3.2 – Modèles de plateforme XY

Modèle	Déplacement max	Résolution	Vitesse max	Charge max
V-731 [7]	205mm	1 nm	200 mm/s	50 N
ONE-XY300 26 DS [3]	290mm	50 nm	200 mm/s	350 N

3.4 Algorithmes d'assemblage d'images

Dans le processus d'acquisition, il est nécessaire d'assembler (« stitcher ») les images issues d'une traverse pour ensuite faire l'analyse des caractéristiques. La raison est que d'une image à l'autre, il peut y avoir un certain chevauchement et l'assemblage permet d'éliminer celui-ci. Avec une plateforme ayant une résolution suffisamment précise, on pourrait croire que le chevauchement est assez minime pour rendre l'assemblage inutile (temps de calcul plus long). Toutefois, certaines caractéristiques sont difficilement identifiables lors de la segmentation si elles sont partiellement visibles sur une image (entre deux scans). L'assemblage des images permet de pallier à ce problème. De plus, avoir une seule image par traverse permet de faire plus facilement une validation des résultats. Pour réaliser l'assemblage d'images, plusieurs algorithmes existent déjà, voici deux des plus pertinents ayant été répertoriés.

3.4.1 Algorithme MIST

L'algorithme MIST (Microscopy Image Stitching Tool) a été développé par le National Institute of Standards and Technology (NIST) spécialement pour l'assemblage d'images en mosaïque issu de numérisations de spécimens depuis un microscope. Cet algorithme est très réputé dans la recherche biologique, notamment dans l'analyse de cultures de cellules. D'ailleurs, la revue scientifique Nature a fait un article à son sujet [8] dans laquelle il décrit son fonctionnement ainsi que ses avantages/inconvénients. Voici un schéma obtenu sur le site officiel² de l'algorithme décrivant sommairement son fonctionnement :

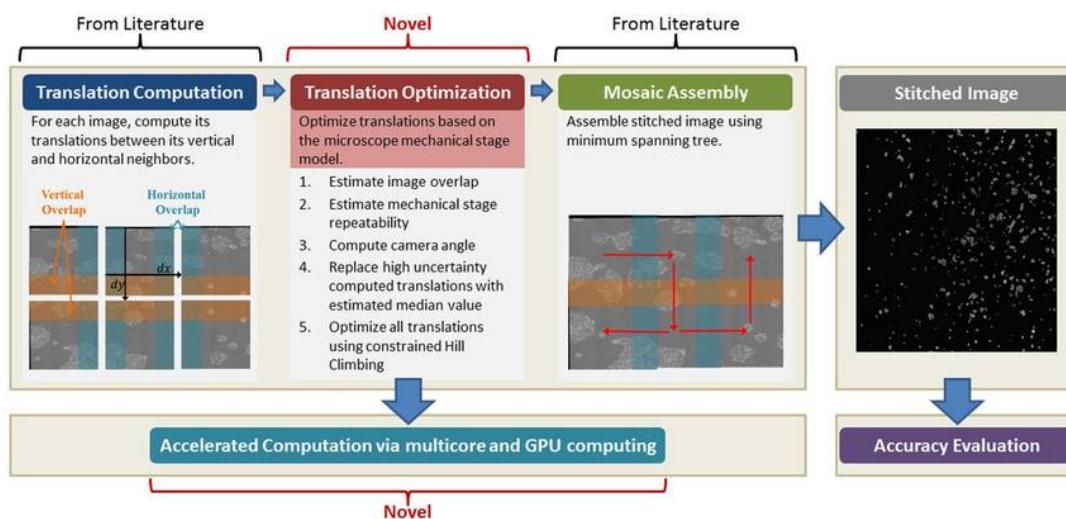


FIGURE 3.1 – Fonctionnement algorithme MIST

2. <https://pages.nist.gov/MIST/>

3.4.2 Classe « Stitcher » de la librairie OpenCV

Cette classe, implémenté en C++ et possédant un API en python (limité) n'est pas qu'un seul algorithme en soit, mais regroupe plutôt plusieurs algorithmes pour différentes applications. La classe peut être paramétrée selon les besoins de l'utilisateur, mais son fonctionnement général reste toujours le même. Celui-ci peut être décrit selon quatre étapes qui ont été bien résumées dans un article écrit par Adrian Rosebrock [13] :

- Étape 1 : Déetectez les points clés (DoG, Harris, etc.) et extraire les descripteurs invariants locaux (SIFT, SURF, etc.) des deux images d'entrée.
- Étape 2 : Faire correspondre les descripteurs entre les deux images.
- Étape 3 : Utiliser l'algorithme RANSAC pour estimer une matrice d'homographie à l'aide des vecteurs de caractéristiques appariées.
- Étape 4 : Appliquer une transformation de déformation à l'aide de la matrice d'homographie obtenue à l'étape 3.

Un des paramètres principaux est le « cv : :Stitcher : :Mode » permettant de choisir entre deux modes pour trouver les correspondances entre les caractéristiques de deux images. Le premier mode « Panorama » utilise le critère du rapport du 2e voisin le plus proche (BestOf2NearestMatcher) pour créer des images panoramiques. Le deuxième mode « Scans » utilise les transformations affines (AffineBestOf2NearestMatcher) pour assembler des scans. Pour l'application de ce projet, c'est le deuxième mode qui serait utilisé.

3.5 Techniques de segmentation

La segmentation d'images est une opération de traitement d'images qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis. Les pixels sont ainsi regroupés en régions, qui constituent un pavage ou une partition de l'image. Il peut s'agir par exemple de séparer les objets du fond.[6]. Cette opération a pour but de réduire la complexité de l'image pour faciliter l'analyse d'image qui s'en suit. Plusieurs techniques existent pour effectuer une segmentation d'images et celles-ci sont regroupées principalement en deux familles : les techniques traditionnelles et les techniques basées sur le « Deep Learning ». La revue de littérature réalisée a permis de relever plusieurs techniques qui ont été synthétisées³ dans le diagramme à la figure 3.2.

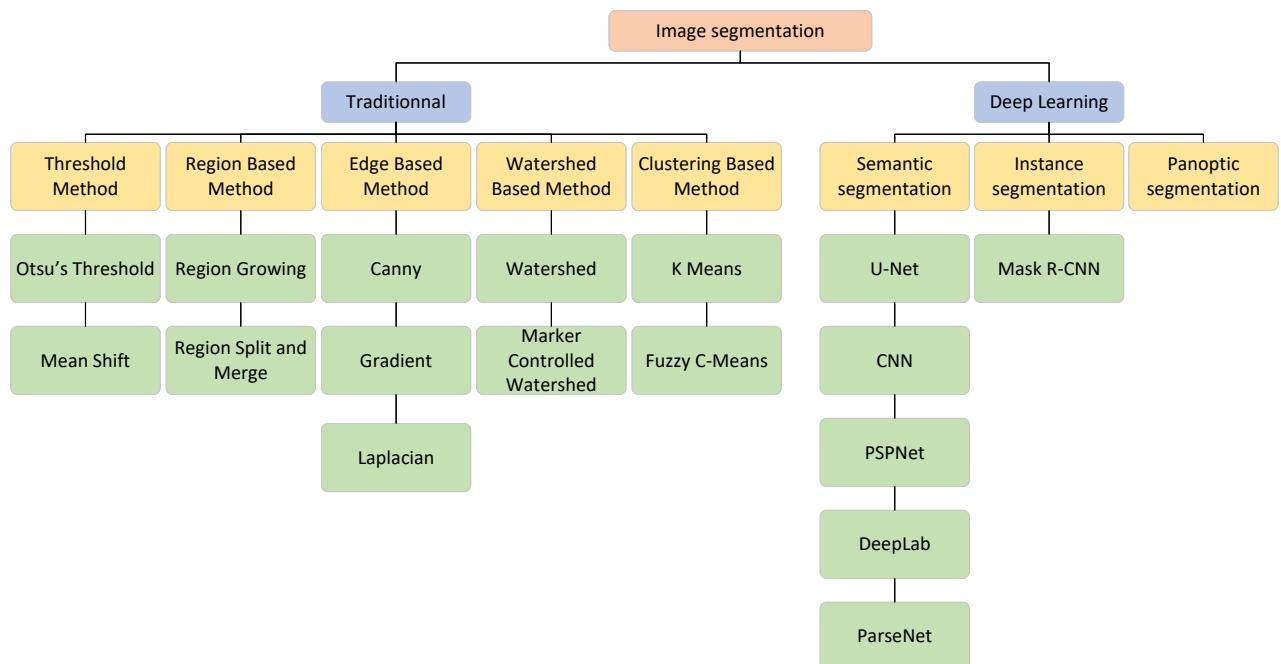


FIGURE 3.2 – Familles de segmentations

3. Les termes anglais ont été utilisés par souci de clarté.

Dans le tableau qui suit, un résumé du fonctionnement ainsi que des avantages et inconvénients a été fait pour chacune des familles d'algorithmes. Encore une fois, les termes anglais ont été favorisés par souci de clarté pour définir les noms de famille d'algorithmes.

TABLE 3.3 – Comparaison segmentation

Algorithme	Description	Avantages	Limites
Threshold Based Segmentation	C'est une opération qui vise à rassembler des pixels entre eux et les classer selon un choix de seuil préalablement déterminé.	<ul style="list-style-type: none"> 1. Très simple d'utilisation. 2. Efficace sur des histogrammes bimodaux. 	Limité lorsqu'il y a plusieurs textures et que le contraste est faible, car l'histogramme à une tendance à être plus uniforme.
Clustering Based Segmentation	La segmentation par k-mean est une méthode de segmentation vectorielle, qui vise à partitionner (n) observations en (k) classes dans lesquels chaque pixel appartient à la classe de moyenne la plus proche. L'algorithme se termine lorsque la moyenne de chaque classe cesse de varier selon un seuil prédéfini.	Avec un paramètre (k) petit, l'algorithme est relativement rapide sur de grandes images.	<ul style="list-style-type: none"> 1. La position de départ pour démarrer l'algorithme a un grand impact sur les résultats. 2. k-means est un algorithme basé sur la distance. Il n'est pas adapté au regroupement de groupes non convexes.
Edge Based Segmentation	Utilise les caractéristiques locales discontinues d'une image pour détecter les bords et ainsi définir une frontière de l'objet.	Est bon pour les images présentant un meilleur contraste entre les objets.	Ne convient pas lorsqu'il y a trop de bords dans l'image et si le contraste entre les objets est faible.
Region-Based Segmentation	Permet de séparer les objets en différentes régions en fonction d'un ou plusieurs critères d'appartenance tels que la connectivité des pixels voisins.	<ul style="list-style-type: none"> 1. Calculs simples 2. Vitesse d'exécution rapide 3. Lorsque l'objet et l'arrière-plan sont très contrastés, cette méthode est très performante. 	<ul style="list-style-type: none"> 1. Lorsqu'il n'y a pas de différence significative entre les niveaux de gris ou qu'il y a un chevauchement des valeurs des pixels en niveaux de gris, il devient très difficile d'obtenir des segments précis. 2. La position de départ de l'algorithme exerce une grande influence sur les résultats et doit être définie manuellement par l'utilisateur.
Watershed Based Method	Des minimums locaux du gradient de l'image peuvent être choisis comme marqueurs, dans ce cas une sur-segmentation est produite et une deuxième étape consiste à fusionner des régions produites à l'aide d'opération morphologique. Cet algorithme aide généralement à détecter les objets qui se touchent et se chevauchent dans l'image.	<ul style="list-style-type: none"> 1. Les limites des régions résultantes correspondent toujours aux contours qui apparaissent dans l'image comme des contours évidents d'objets. 2. Les frontières résultantes forment des régions fermées et connectées. 	Cette méthode peu parfois produit une sur-segmentation excessive. Ce qui fait apparaître de nouvelles régions indésirables.
Artificial Neural Network Based Segmentation	Basé sur l'entraînement d'un modèle avec une architecture de plusieurs couches choisi selon le cas d'utilisation.	<ul style="list-style-type: none"> 1. Une approche simple (bibliothèque de haut niveau disponible), flexible et générale. 2. Est particulièrement efficace lorsque beaucoup de paramètres sont impliqués (choix de paramètre automatique). 	<ul style="list-style-type: none"> 1. Temps d'entraînement élevé. 2. À besoin d'une grande banque d'images pré-identifiée.

Méthodologie de travail

La conception du système peut être divisé en cinq étapes principales :

1. Concevoir et assembler un prototype d'acquisition d'images ;
2. Développer un logiciel pour contrôler le système d'acquisition ;
3. Choisir un algorithme d'assemblage d'images ;
4. Développer un modèle de détection des caractéristiques ;
5. Développer des méthodes de validation des résultats ;

Pour optimiser l'avancement de ce projet, un certain ordre d'accomplissement doit être respecté. Les étapes (1) et (2) doivent être accomplies au préalable pour permettre l'avancement des trois autres, car des scans d'échantillons sont nécessaires pour commencer celles-ci. Une fois les étapes (1) et (2) accomplies, les étapes (3) et (4) peuvent être réalisées en parallèle par les deux membres de l'équipe de recherche pour gagner du temps. Finalement, l'étape (5) nécessite que toutes les autres soient accomplies. Pour chacune des étapes, plusieurs ressources et méthodologies d'expérimentation seront utilisées. Dans cette section, une sommaire de chacune de ces étapes sera présenté. Voici un diagramme illustrant la planification chronologique du projet.

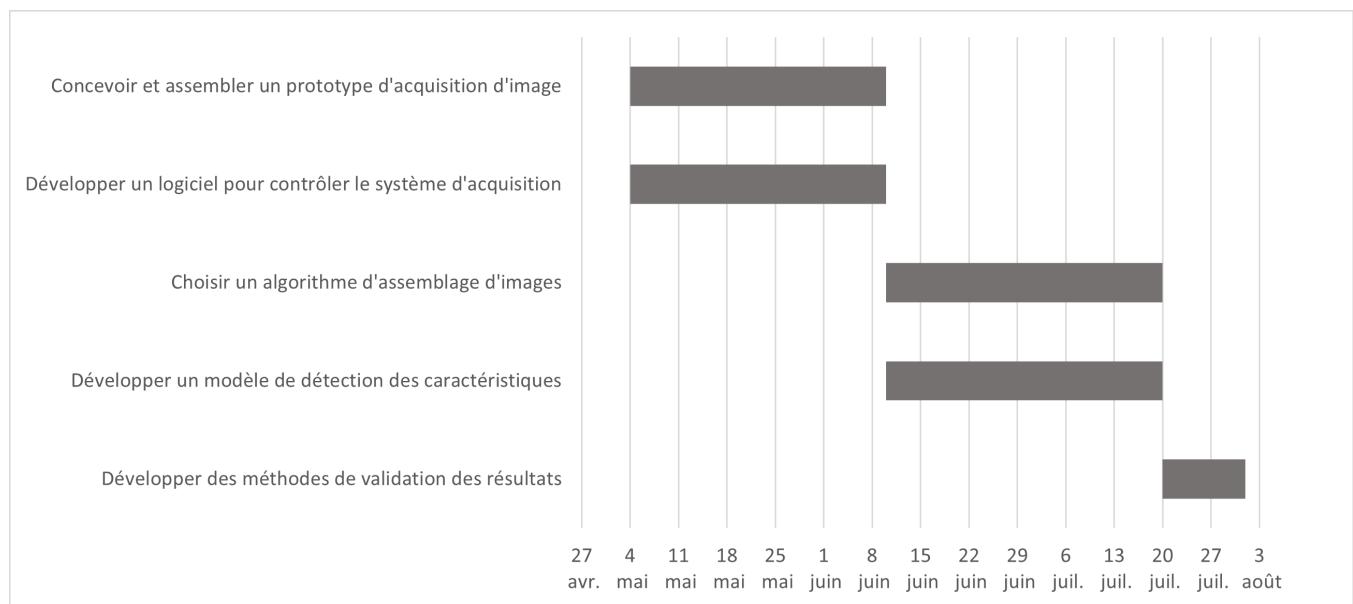
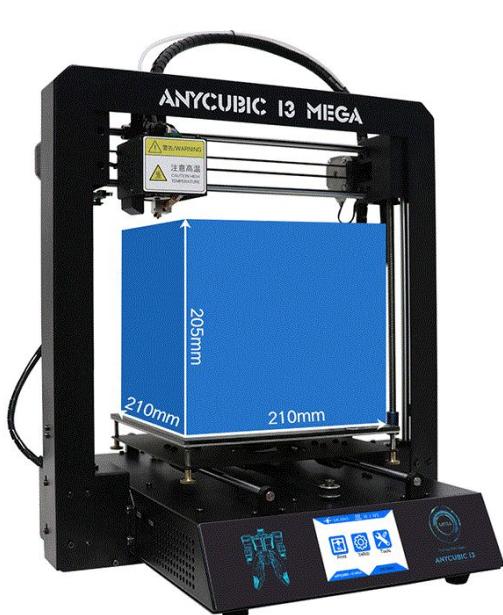


FIGURE 4.1 – Planification de projet

4.1 Étape 1 - Concevoir et assembler un prototype d'acquisition d'image

La conception du prototype sera réalisée par Éric Surprenant. L'utilisation d'une imprimante 3D appartenant à Éric sera utilisée comme plateforme XY et un microscope digital sera acheté pour être installé à la place de la buse d'extrusion. Un support de montage pour le microscope et une fixture pour mettre en place l'échantillon seront imprimés avec une imprimante 3D appartenant à Groupe ABS. Le logiciel Solidwork sera utilisé pour faire la modélisation du système. Voici une image¹ du modèle d'imprimante utilisé ainsi qu'un exemple de microscope² digital pouvant être acheté.



(a) Imprimante 3D Anycubic i3 Mega



(b) Exemple de microscope digital

1. <https://store.makerwiz.com/products/anycubic-m-i3-mega-3d-printer>

2. <https://www.amazon.ca/Skybasic-Microscope-50X-1000X-Endoscope-Smartphone/dp/B07DVFBVPF>

4.2 Étape 2 - Développer un logiciel pour contrôler le système d'acquisition

Concernant la partie « software » du système, le langage de programmation Python a été choisi pour plusieurs raisons. Premièrement, Python est un langage orienté objet, ce qui permet d'utiliser les principes d'héritage et de polymorphisme. Ces principes permettent de construire un logiciel pouvant être facilement adapté lorsque le système passera de la phase « prototypage » à la phase « production ». Ensuite, Python est un langage très populaire en vision artificielle et possède par conséquent une quantité importante de librairies utiles pour notre projet tel qu'OpenCV. Finalement, Python est un langage très facile à utiliser, car il permet de faire abstraction de mécanismes complexes à implémenter tels que la gestion de mémoire ce qui rend le développement plus rapide si on le compare au C++ par exemple.

Plusieurs librairies (*package*) Python avec des versions spécifiques seront requises pour le développement. L'interdépendance de ces librairies et la compatibilité de leur version peuvent devenir une gestion très fastidieuse lorsque plusieurs sont utilisés. L'utilitaire Conda sera utilisé pour simplifier cette tâche. Conda est un gestionnaire de librairies et d'environnement très populaire et gratuit. Celui-ci s'occupe du choix de version de *package* pour assurer une intercompatibilité en plus de créer un environnement de développement isolé et exclusif à notre projet. Finalement, Conda permet aussi de compiler notre projet en un « Conda package » permettant une portabilité qui facilite l'installation du programme et de ses dépendances.

Concernant l'environnement de développement, Visual Studio a été choisi, car il est simple, robuste et gratuit. De plus, les membres de l'équipe sont tous familiers avec celui-ci. Quant à la gestion de version, celle-ci sera assurée avec la plateforme Github qui possède une extension permettant une intégration dans Visual Studio.

4.3 Étape 3 - Choisir un algorithme d’assemblage d’images

Deux algorithmes d’assemblage d’images ont été répertoriés dans la revue de littérature. Pour pouvoir être utilisés avec le programme qui sera développé, ils doivent posséder une interface en Python. Toutefois, ce n’est pas le cas pour l’algorithme MIST. Le cadre de ce projet (temps disponible) permet seulement d’accomplir la phase de prototypage, donc pour être réaliste l’algorithme MIST sera utilisé avec le logiciel FIJI d’ImageJ sous la forme d’un *plugin* pouvant être téléchargé à partir du répertoire Github officiel³. Dans l’optique où ce projet est amené en phase de production, ceci ne représente pas une limitation, car tout algorithme peut être ré-implémenté en Python (faire un « wrapper ») si nécessaire. En ce qui concerne l’algorithme d’OpenCV, la classe « Stitcher » de l’interface Python sera simplement implémentée dans le code du logiciel. L’utilisation de ces deux algorithmes va permettre de faire un comparatif de leur performance. Pour comparer la performance, deux critères seront évalués : le temps requis pour réaliser l’assemblage et la présence d’artéfacts dans l’image résultante.

Pour évaluer le temps d’assemblage, les fonctions « tic » et « toc » seront utilisées à partir de la librairie « time » de Python pour l’algorithme d’OpenCV et les *logfiles* générés par le logiciel FIJI seront utilisés pour l’algorithme MIST. La présence d’artéfacts sera évaluée visuellement à partir des résultats.

3. <https://github.com/usnistgov/MIST>

4.4 Étape 4 - Développer un modèle de détection des caractéristiques

Le développement du modèle de détection comporte 3 étapes : (1) Calculer la grosseur de pixel, (2) Segmenter l'image, (3) Évaluer les caractéristiques désirées. Pour développer le modèle de détection, l'environnement de développement Matlab sera utilisé. Matlab possède de nombreux outils et librairies permettant de tester différentes techniques de traitement et d'analyse d'images. Toutefois, ce sont les outils de visualisation pour faciliter l'interprétation des résultats qui le rend particulièrement intéressant. La librairie « Image Processing Toolbox » sera utilisée à cet effet. Voici une description de la méthode prévue pour chacune des étapes de conception du modèle de détection.

4.4.1 Calculer la grosseur de pixel

Comme il a été établit à la section 3.2, la méthode (C) nécessite de connaître la grosseur de pixel (R) pour ensuite calculer la distance de traverse. Pour obtenir cette valeur, les microscopes commerciaux possèdent généralement des procédures de calibration automatique pour la déterminer. Toutefois, le microscope qui sera utilisé pour le prototype n'offre pas cette possibilité. Pour obtenir cette valeur, un gabarit de calibration avec des formes d'une dimension connues sera installé à une position fixe au même niveau que la surface de l'échantillon. Une procédure de calibration sera programmée pour prendre une photo de ces formes et calculer combien de pixels représentent une dimension connue. Un ratio sera calculé par la suite pour connaître la grosseur d'un pixel.

4.4.2 Segmenter l'image

La segmentation de l'image est nécessaire pour classifier les pixels appartenant aux classes suivantes : pâte, roche, vide d'air et vide de roche. Pour y parvenir, plusieurs techniques de segmentation existent. À partir de la revue de littérature qui a été faite, cinq catégories de segmentation ont été choisies pour être testées, soit :

1. Segmentation par seuillage (*Threshold Based Segmentation*)
2. Segmentation automatique (*Clustering Based Segmentation*)
3. Segmentation par détection de contours (*Edge Based Segmentation*)
4. Segmentation par région (*Region-Based Segmentation*)
5. Segmentation basée sur un réseau de neurones artificiels (*Artificial Neural Network Based Segmentation*)

4.4.3 Évaluer les caractéristiques désirées

Une fois la grosseur de pixel connue et l'image segmentée, les calculs de la méthode (A) doivent être faits pour évaluer les paramètres permettant de caractériser le béton, soit :

- n = La fréquence de vide d'air ;
- A = La teneur en air (%) ;
- p = La teneur en pâte (%) ;
- p/A = Le ratio air-pâte ;

Pour ce faire, un vecteur 1D représentant la ligne de traverse⁴ sera extrait au centre de l'image segmentée. À partir de ce vecteur, les données suivantes seront retenues :

- (P_a) = Nombre de pixels de vide d'air ;
- (P_p) = Nombre de pixels de pâte ;
- (P_t) = Nombre de pixels total du vecteur ;

De plus, une fonction sera développée pour évaluer le nombre de vides d'air (N) étant intersecté par ce vecteur. Ensuite, ces données seront utilisées pour obtenir les valeurs de calcul intermédiaires suivantes :

$$\begin{aligned} T_a &= P_a * R \\ T_p &= P_p * R \\ T_t &= P_t * R \end{aligned} \tag{4.1}$$

Finalement, à partir de ces valeurs intermédiaires, il sera possible de calculer les quatre paramètres permettant de caractériser le béton tel que :

$$\begin{aligned} n &= \frac{N}{T_t} \\ A &= \frac{T_a * 100}{T_i} \\ p &= \frac{T_p * 100}{T_t} \\ \frac{p}{A} &= \frac{T_p}{T_a} \end{aligned} \tag{4.2}$$

4. ligne bleu dans le schéma de la figure 1.3

4.5 Étape 5 - Développer des méthodes de validation des résultats

Plusieurs techniques de segmentation appartenant à une des catégories sélectionnées seront testées afin de déterminer si une, ou la combinaison de plusieurs permet d'obtenir une segmentation optimale. Pour trouver les paramètres optimaux des techniques conventionnelles, une analyse ROC sera effectuée en imbriquant plusieurs boucles itérant chacune sur un des paramètres de la fonction de segmentation. Pour ce faire, des images de référence (« groundtruth ») sont nécessaires. Celles-ci seront créées avec l'utilitaire « Image Labeler » disponible dans Matlab. Une fois que les paramètres optimaux seront trouvés pour chaque technique de segmentation, celles-ci seront comparées entre elles avec un test de similarité basé sur la méthode de Dice (« Dice score »). La méthode de Dice prend l'image binaire issue de la segmentation et la compare avec l'image binaire de référence. Étant donné que plusieurs classes seront segmentées, chacune de ces classes sera évaluée individuellement. Pour tester la segmentation sémantique, une banque de 8 images identifiées sera créée pour tester les performances du modèle qui aura été entrainé. Les démarches précédentes permettront de trouver la meilleure technique de segmentation. Une analyse d'un échantillon déjà testé par Groupe ABS sera réalisée avec la meilleure technique trouvée. Les résultats obtenus par notre système seront comparés avec ceux obtenus par Groupe ABS. De cette façon, il sera possible de déterminer si le système développé fournit des résultats acceptables.

Processus de conception

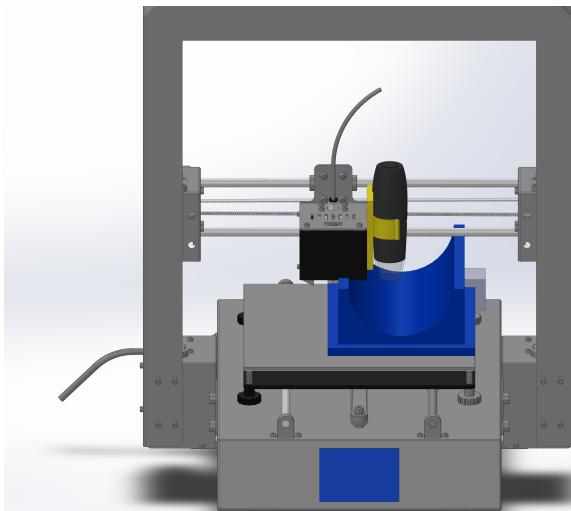
5.1 Conception du prototype

5.1.1 Première version

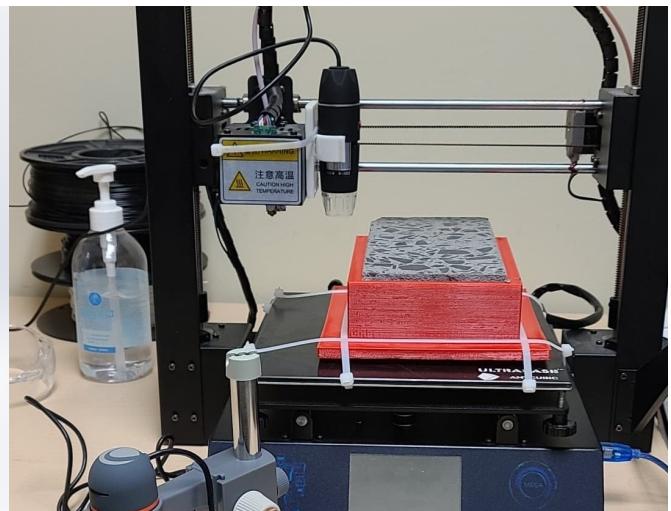
Une première version du prototype a été construite avec la configuration suivante :

- Fixture pour l'échantillon de béton conçu avec imprimante 3D (en bleu dans la Figure ??) ;
- Support pour le microscope conçu avec imprimante 3D (en jaune dans la Figure ??) ;
- Microscope digital Jiusion ;
- Imprimante 3D ANYCUBIC Mega S 3D ;

Dans cette version, l'orientation de la fixture n'était pas optimal par rapport à l'espace quelle prenait sur la plateforme. De plus, son positionnement rendait la mise à zero des axes (« homing ») très difficile. Ensuite, la résolution et le grossissement du microscope n'était pas suffisant pour avoir une bonne définition des caractéristiques. Finalement, le support de montage du microscope n'était pas très solide et lorsque celui-ci se déplaçait cela créait une vibration qui rendait l'image flou. Pour ces raisons, une deuxième version du microscope fut nécessaire.



(a) 3D



(b) Réel

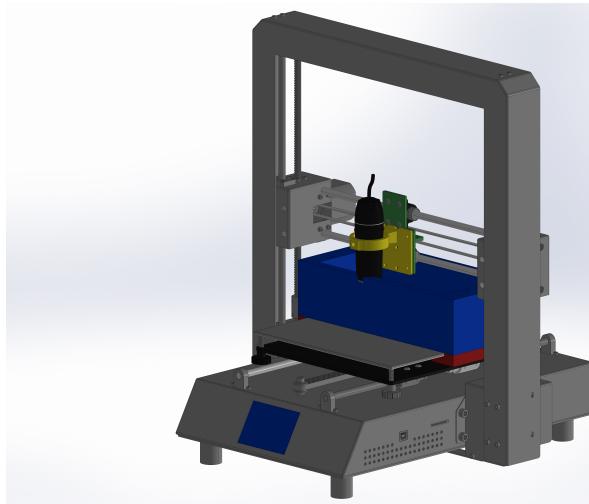
FIGURE 5.1 – Prototype version 1

5.1.2 Deuxième version

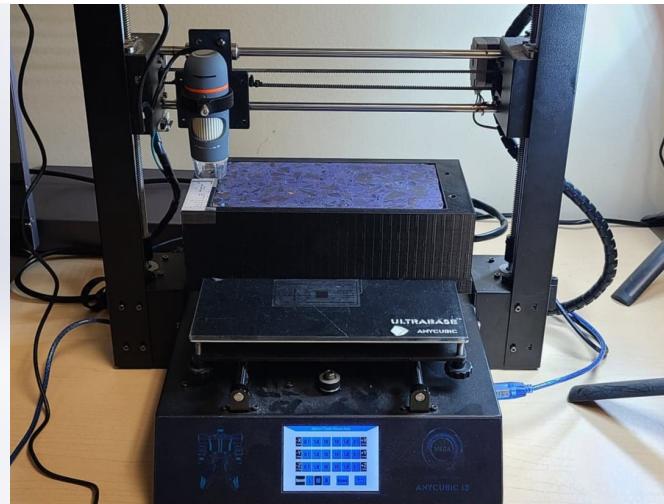
La deuxième et dernière version du prototype a permis de corriger tous les problèmes expérimentés avec la première. Voici une liste des améliorations qui ont été apportées :

- Ajout d'un nouveau modèle de microscope plus performant (Celestron 5 MP Handheld Digital Microscope Pro) ;
- Modification du design de la fixture à béton et de sa position sur la plateforme de l'imprimante 3D. Le nouveau design permet une fixation plus solide et planaire de la fixture sur la plateforme.
- La buse d'extrusion a été retirée de l'imprimante, ce qui a libéré de l'espace permettant ainsi de développer un support plus solide pour le microscope.

Voici une image de la 2ème version du prototype :



(a) 3D



(b) Réel

FIGURE 5.2 – Prototype version 2

5.2 Conception du logiciel

5.2.1 Fonctionnement général

Le programme permettant de contrôler le prototype est un « package » Python divisé en trois modules. Il y a le module « Microscope » pour contrôler le microscope, le module « Stage » pour contrôler la plateforme XY et le module « Analyser » qui utilise les deux premiers modules pour pouvoir réaliser les séquences de scan. Tel que soulevé dans la méthodologie, le principe d'héritage a été utilisé pour développer les modules afin d'être en mesure d'intégrer facilement de nouveaux équipements au système et faciliter la maintenance du code. Par exemple, une imprimante 3D est utilisée pour le prototype actuellement. Toutefois, lorsque la version « production » du système sera développée, une vraie plateforme XY sera utilisée et le code devra être adapté à celle-ci. En créant une classe abstraite (« base class ») contenant toutes les fonctions de bases, il sera beaucoup plus rapide de développer une nouvelle classe pour cette plateforme en héritant de la « base class » ainsi que d'apporter les modifications dans le code où cette classe est utilisée. La Figure 5.3 plus bas est un diagramme de classe UML représentant la structure du projet. Comme on peut le voir, deux « base class » ont été faites pour le microscope et la plateforme XY. Pour le moment, nous héritons seulement de la classe « BaseStage ». Nous n'héritons pas de la classe « BaseMicroscope », car aucune méthode spécifique au microscope que nous avons actuellement n'est requise. À noter que les objets « MoveMode », « Positioning » et « Units » sont des énumérations.

5.2.2 Module « Microscope »

Le module « microscope » est relativement simple. Le microscope digital utilise le même protocole de communication que les caméras/webcams standards. Par conséquent, il a été possible d'utiliser la classe « cv : :VideoCapture » de la librairie OpenCV pour accéder au « stream » de celle-ci. La seule particularité de ce module est que la classe abstraite « BaseMicroscope » a été développée de façon à pouvoir être utilisée facilement en mode asynchrone (« multithreading »). Cela permet de récupérer des *frames* tout en affichant le flux vidéo en continu sur un *thread* secondaire pendant que la séquence de *scan* s'effectue sur le *thread* principal.

5.2.3 Module « Stage »

Le module « Stage » est responsable du déplacement du microscope par rapport à l'échantillon. Pour asservir la plateforme XY, plusieurs fonctions et paramètres sont requis. D'abord, il y a les informations concernant le protocole de communication. La majorité des plateformes XY (incluant l'imprimante 3D) utilise une communication sérielle sur un port COM. Pour établir cette communication, la librairie « Pyserial » est utilisée et nécessite certaines informations tels le numéro de port, le débit en bauds (« baurate »), la parité, le nombre de bits, etc. Ces informations ont été ajoutées à la classe « BaseStage » avec des valeurs fréquemment utilisées par les fabricants, mais peuvent toujours être modifiées lorsqu'on hérite de la classe abstraite pour créer une classe spécifique à une plateforme XY. Concernant les propriétés, les dimensions de la plateforme ont aussi été ajoutées pour éviter d'envoyer une commande de déplacement hors limite.

Ensuite, concernant les fonctions, la première à avoir été créé est la fonction « send_command ». Cette fonction permet de formater les commandes devant être envoyées à la plateforme. Ensuite, il y a les fonctions « start » et « stop » permettant de réaliser les routines de démarrage et d'arrêt. Finalement, il y a les fonctions « home », « moveX », « moveY » et « moveZ ». Celles-ci permettent d'effectuer les déplacements et la mise à zéro des axes. Toutes ces fonctions sont spécifiques à la plateforme et par conséquent sont des méthodes abstraites devant être implémentées lorsqu'on hérite de la classe « BaseStage ». La librairie « abc » a été utilisée pour implémenter le concept de classe abstraite. Voici un extrait du code pour illustrer ce concept :

```
class BaseStage(metaclass=abc.ABCMeta):

    def __init__(self, port=1, stagezone=[0,0,0],parity=8,baudrate=250000,bits=8):
        self.port = port
        self.stagezone = stagezone
        self.baudrate = baudrate
        self.parity=parity
        self.bits=bits

    @classmethod
    @abc.abstractmethod
    def send_command(self):
        """ Implement me! """
        pass
```

5.2.4 Module « Analyser »

Le module « Analyser » est le module principal du *package*. Celui-ci contient toute la logique concernant la séquence d'acquisition et d'assemblage d'image ainsi que l'analyse qui s'en suit. Pour y arriver, le module possède une classe qui instancie un objet « BaseMicroscope » et « BaseStage » pour contrôler le microscope et la plateforme XY. Voici une liste des fonctions de cette classe et de leur utilité :

- **create_scandir** : Crée un dossier pour stocker les *scans*. Si le nom de dossier existe déjà, le terme « (X) » est ajouté automatiquement à la fin. Ex. : si le dossier « scan(1) » existe déjà, le dossier « scan(2) » est créé.
- **scan** : Exécute la séquence de *scan* pour faire l'acquisition d'images selon certains paramètres d'entrée tels que le nombre de traverses, la distance entre deux images, le nombre d'images sur une traverse, etc.
- **autofocus** : Exécute une séquence d'autofocus en ajustant la distance entre le microscope et la surface de l'échantillon. Lorsque cette fonction est appelée, celle-ci évalue le niveau de brouillage (« blur ») dans l'image en appliquant une convolution avec un filtre Laplacien et calculant la variance de l'image de sortie. Si la variance est plus élevée avec la nouvelle position du microscope, cela veut dire que l'image est plus claire. La fonction incrémente la position jusqu'à ce que la variance atteigne un maximum.
- **calibration** : Exécute une séquence qui positionne le microscope au-dessus de la zone de calibration et qui évalue la grosseur de pixels avec l'aide de forme géométrique de dimensions connues.
- **stitch** : Exécute l'assemblage d'images une fois la séquence complétée.
- **analyze** : Segmente les images assemblées de chaque traverse et calcule les propriétés du réseau de vide d'air.

La fonction « analyze » a été créée pour démontrer le fonctionnement du module, mais n'a pas été implémentée, car le développement de la technique de segmentation n'est pas complété.

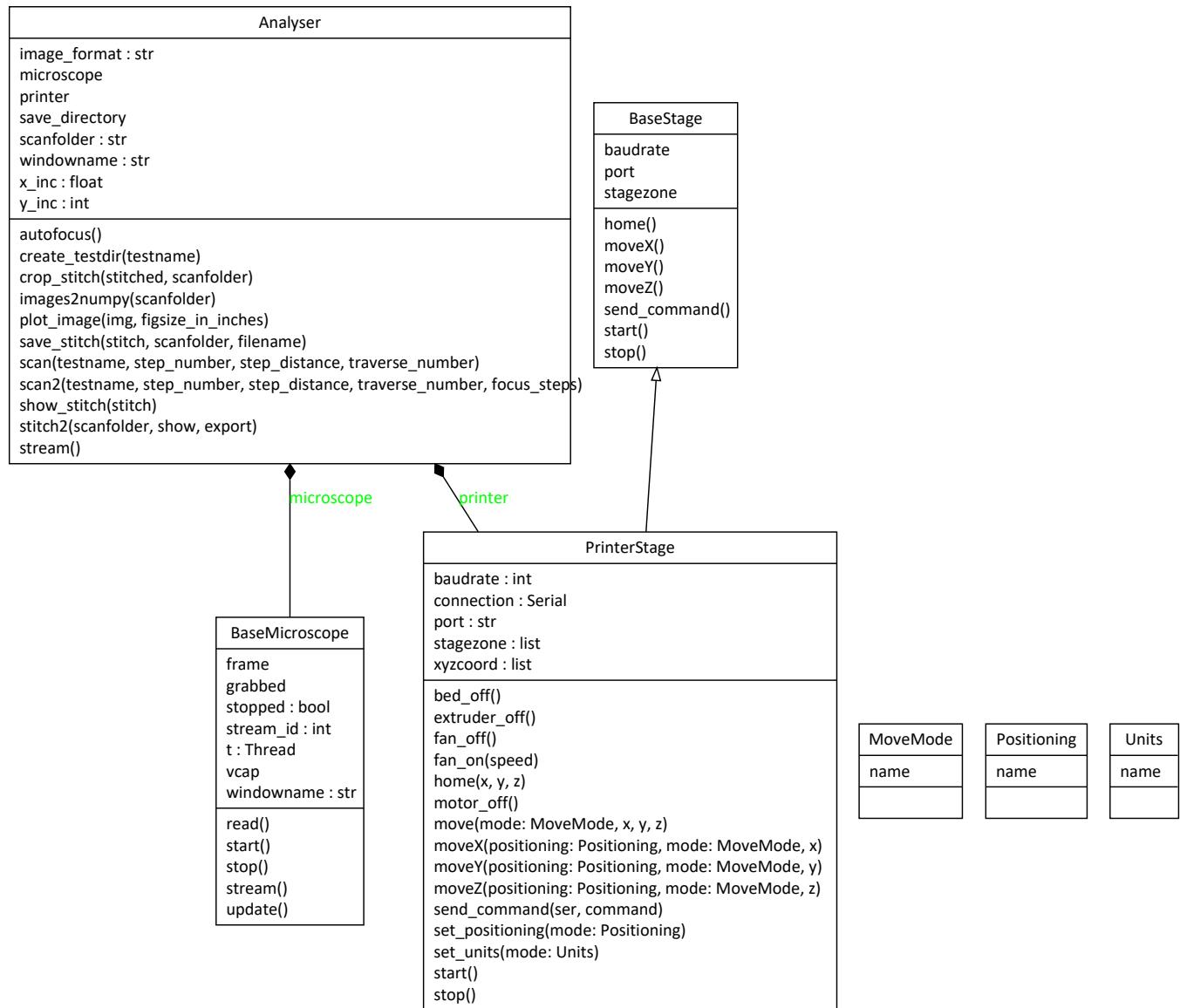


FIGURE 5.3 – Diagramme de classe UML

5.3 Conception du modèle de détection des caractéristiques

5.3.1 Calculer la grosseur de pixel

Initialement, il était prévu de faire le design du calcul de grosseur de pixel dans Matlab et ensuite implémenter la logique dans une fonction Python. Cette fonction aurait assuré le déplacement du microscope vers le gabarit de calibration pour prendre une image et ensuite les calculs et le traitement d'images développé dans Matlab aurait été appliqué pour obtenir la grosseur de pixel. Cependant, le temps limité à notre disposition a permis seulement de développer la logique dans Matlab. Celle-ci va comme suit :

1. La fonction reçoit en entrée l'image d'une ligne de 0.076mm d'épaisseur provenant du gabarit de microscope.
2. Cette image est convertie en niveau de gris et un seuillage est appliqué pour obtenir une image binaire.
3. Des opérations morphologiques d'ouverture et de fermeture sont appliquées pour corriger les artéfacts de seuillage.
4. Une séquence pour déterminer l'alignement de la caméra est exécutée afin de lever une exception si elle n'est pas alignée. La séquence va comme suit :
 - (a) Calcule la hauteur moyenne (100 pixels) aux extrémités de la bande et calcule la différence ;
 - (b) Si cette différence dépasse un certain seuil, une exception est levée ;
5. Une séquence pour déterminer l'épaisseur de la bande est exécutée. La séquence va comme suit :
 - (a) Calcul la hauteur du pixel le plus bas et du pixel le plus haut de chaque colonne ;
 - (b) Fait une moyenne des pixels les plus hauts et une moyenne des pixels les plus bas ;
 - (c) Calcul la différence entre ces deux moyennes pour obtenir l'épaisseur moyenne en pixel de la bande ;
 - (d) Divise l'épaisseur réelle en millimètre (0.076mm) par l'épaisseur moyenne calculée en pixel pour obtenir la grosseur spatiale d'un pixel ;

5.3.2 Segmenter l'image

En fonction des données qui on été receuillis dans la revue de littérature concernant les différentes techniques de segmentation , plusieurs d'entre elles ont été sélectionné pour être testé dans Matlab selon la méthodologie établit à la section 4.4.2.

« Threshold Based Segmentation »

Cette technique est considérée comme étant l'une des plus simples. Elle consiste à séparer les pixels de l'image à l'aide d'un seuil prédéterminer. Le résultat de celle-ci transforme l'image en niveau de gris en une image binaire dont les valeurs de pixels ne peuvent avoir que la valeur 1 ou 0. Cependant, un éclairage non uniforme de la scène ou un faible contraste de niveau de gris peut aboutir à une segmentation erronée. En appliquant la fonction « histogram » de l'image grisee (Figure 5.4), l'option du seuillage en ton de gris a été rejetée dû à une répartition des pixels plutôt unimodale.

De ce fait, le seuillage par couleur a été considéré. L'application « Color Thresholder » a été utilisée pour tester le seuillage sous différente codification de couleur («Color space») telle que RGB, LAB, HSV et YCbCr. La codification RGB est celle qui a semblé pouvoir le mieux séparer les classes de pixels. Une fonction a donc été créée pour trouver les meilleures valeurs de seuillage pour les trois canaux RGB à l'aide de trois boucles d'itération. Ces boucles comprenaient une segmentation de la pâte et des roches. Ensuite, à la fin de chaque itération, une analyse ROC a été effectuée à l'aide du « Grountruth » afin d'extraire la fraction des vrais positifs (TPF), c'est-à-dire le rapport du nombre de pixels correctement segmentés et la fraction des faux positifs (FPF) qui est l'idée inverse du (TPF). L'analyse des courbes ROC pour la pâte et la roche on permis de choisir les valeurs de seuillage optimal.

« Clustering Based Segmentation »

Pour tester la technique de segmentation automatique (« Clustering Based Segmentation »), l'algorithme k-mean a été utilisé. Cet algorithme peut être utilisé de deux façons. La première est d'appliquer l'algorithme avec la fonction « kmean » de Matlab sur une image convertie en niveau de gris. Cette technique s'est avérée peu concluante avec les images qui ont été testées. La deuxième méthode est d'appliquer l'algorithme sur les différents canaux de couleur d'une image à l'aide de la fonction Matlab « imsegkmeans ». Avec quelques essais, il a été rapidement observé que cette méthode est nettement supérieure à la précédente. De ce fait, une optimisation des paramètres de la fonction « imsegkmeans » a été réalisée.

Le premier paramètre, qui n'est pas directement indiqué dans la fonction, est le type de codification de couleur de l'image. Comme avec la technique de segmentation par seuillage de couleur, les codifications RGB, LAB, HSV et YCbCr ont été testées. Après quelques essais, il a été conclu que la codification LAB est celle permettant d'obtenir les meilleurs résultats. En fait, c'est en retirant le canal « L » des canaux « LAB » que les meilleurs résultats ont été obtenus.

Une image codifiée en « AB » a ensuite été utilisée dans une double boucle imbriquée pour tester les paramètres « k » (nombre de classes) et « thresh » (seuille de précision). Normalement, le paramètre « k » aurait dû être fixé à 4 pour signifier les quatre classes « roche », « pâte », « vide d'air » et « vide de roche ». Toutefois, l'algorithme fut incapable de segmenter correctement ces quatre classes. Pour cette raison, il a été décidé de tester différentes valeurs de « k » pour essayer d'obtenir la meilleure fraction de vrai positif pour les classes « roche » et « pâte ». L'analyse de courbes ROC et le test de similarité « Dice Score » ont été utilisés tels que décrits dans la méthodologie pour obtenir les meilleurs paramètres.

« Edge Based Segmentation »

La segmentation par détection de contours s'effectue en deux étapes. Premièrement, les pixels représentant les contours d'objets sont détectés en filtrant l'image. Pour ce faire, la fonction Matlab « edge » a été utilisée. Pour filtrer l'image, différentes méthodes existent et peuvent être spécifiées dans les paramètres de la fonction « edge ». La majorité des méthodes listées ci-dessous ont été testées soit :

1. approxcanny
2. canny
3. canny_old
4. log
5. prewitt
6. Roberts
7. Sobel
8. zerocross

Ces méthodes sont généralement sensibles aux variations de contraste, c'est pourquoi un filtre moyennant a été appliqué préalablement avec différentes valeurs de dimensions et de poids pour le noyau. Pour chacune de ces méthodes, des seuils peuvent être paramétrés pour optimiser les résultats, par conséquent plusieurs combinaisons de seuils ont été testées.

La deuxième étape consiste à lier les pixels entre eux pour former des objets en entier. Deux méthodes ont été testées pour lier les pixels.

1. Le traitement local où les bords voisins d'autre pixel à proximité sont reliés en utilisant le gradient et la direction de ceux-ci.
2. Le traitement global qui est réalisé à l'aide de la transformation de HOG.

Comme on peut le voir, la segmentation par détection de contours implique beaucoup de paramétrage. Il n'a pas été possible de tester toutes les méthodes possibles et d'optimiser tous les paramètres disponibles. Toutefois, la revue de littérature a permis de diriger les essais qui ont été réalisés pour se limiter seulement aux méthodes les plus adaptées au cas d'utilisation de ce projet.

« Region Based Segmentation »

Pour tester cette technique de segmentation, la méthode « Croissance de région » a été employée. Une fonction développée dans le cours GTS601 a été utilisée pour exécuter l'algorithme. Celle-ci consiste à développer des régions en incluant de manière récursive les pixels voisins qui sont similaires et connectés à un pixel de départ au hasard ou prédéterminés. La région d'intérêt « pâte » s'agrandissait par le principe suivant :

1. Un pixel aléatoire ou prédéterminé est sélectionné dans l'image.
2. Ce pixel se compare avec ces voisins connectés à l'aide de la distance euclidienne
3. $Dp = \sqrt{(R1 - R2)^2 + (G1 - G2)^2 + (B1 - B2)^2}$
4. Cette distance (Dp) permet d'inclure les pixels voisins dans la région d'intérêt si elle est plus petite ou égale au seuil d'homogénéité.
5. L'agrandissement arrête de progresser lorsque les pixels voisins sont tous au-dessus du seuil.

Donc, en gardant un seuil identique pour chaque itération. La région variait en fonction des deux paramètres suivants :

1. **La marge d'erreur (ME)** : Le niveau de tolérance entre la distance calculée entre deux pixels et le seuil d'homogénéité.
2. **La taille et la forme du masque (R)** : Il peut se dilater au fil des itérations sous différentes tailles et différentes formes. Le masque était un disque de rayon variable.

À chaque itération des deux boucles « for », le masque prenait de l'expansion à travers l'image. L'expansion variait entre un rayon de 2 à 12 et la marge d'erreur avait un intervalle de 5 à 55. Comme les précédentes, la même procédure a été effectuée pour trouver les meilleurs paramètres.

« Semantic Segmentation With Deep Learning »

Un réseau neuronal a été entraîné avec la méthode d'apprentissage supervisée (« supervised learning ») à l'aide d'une banque de 40 images où chaque pixel a été identifié selon la classe qu'il appartient. Les classes qui ont été utilisées sont « rocks », « airvoids », « rockvoids » et « paste ». Voici un diagramme de l'infrastructure du réseau neuronal qui a été conçu :

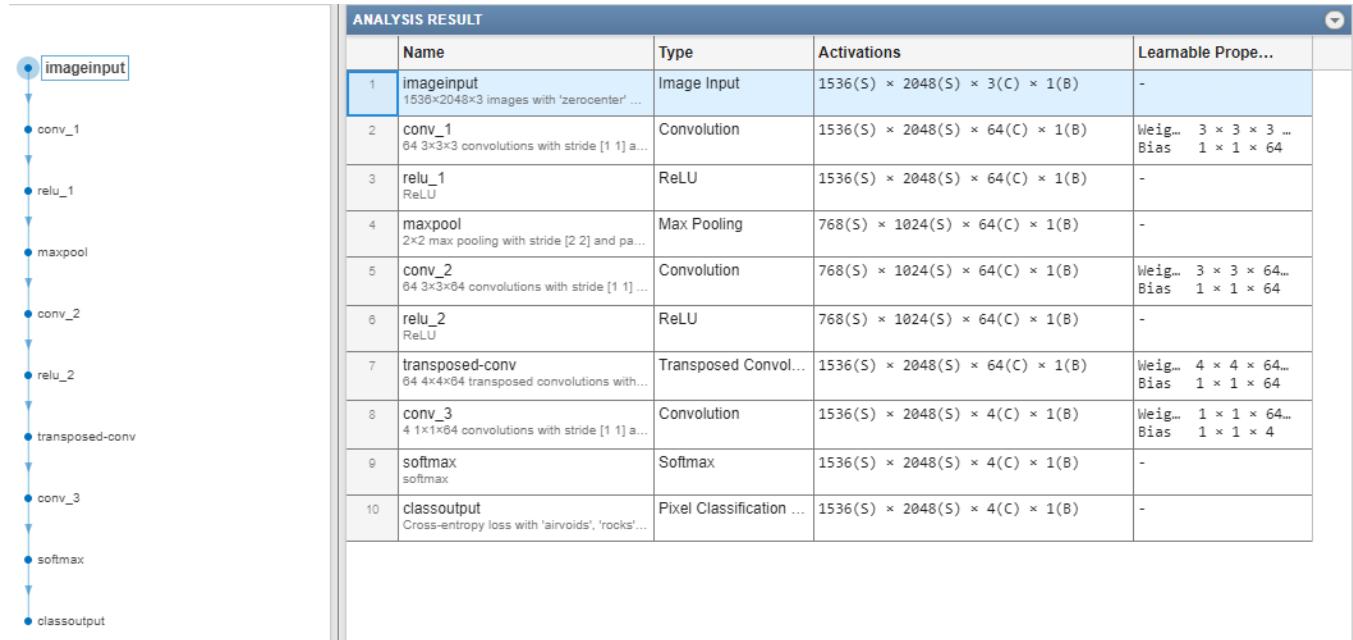


FIGURE 5.4 – Architecture du modèle d'apprentissage profond

Cette infrastructure a été prise telle quelle dans un tutoriel¹ Matlab traitant de la segmentation sémantique. Évidemment, l'infrastructure pourrait grandement bénéficier de modifications pour l'adapter à l'usage spécifique que nous voulons en faire. Toutefois, considérant le cadre de ce projet et le peu d'expérience qu'ont les membres de l'équipe dans ce domaine, l'infrastructure se limite à ce qui est fourni dans ce tutoriel. Pour tester le modèle une fois entraîné, une banque de 8 images différente de celle utilisée pour l'entraînement fut identifiée.

1. <https://www.mathworks.com/help/vision/ug/semantic-segmentation-with-deep-learning.html>

Interprétation des résultats et discussion

6.1 Résultats l'algorithme d'assemblage d'images

Pour comparer les deux algorithmes d'assemblage d'images, 10 traverses réalisé avec un chevauchement de 44% on été utilisé. Le choix d'une valeur de 44% pour le chevauchement a été fait en se basant sur le minimum requis par l'algorithme d'OpenCV pour réussir l'assemblage. L'algorithme d'OpenCV a pris en moyenne 173 secondes pour terminer l'assemblage alors que l'algorithme MIST en a pris 20. Voici un exemple d'image assemblé obtenu pour chacun des algorithmes :



FIGURE 6.1 – Assemblage OpenCV



FIGURE 6.2 – Assemblage MIST

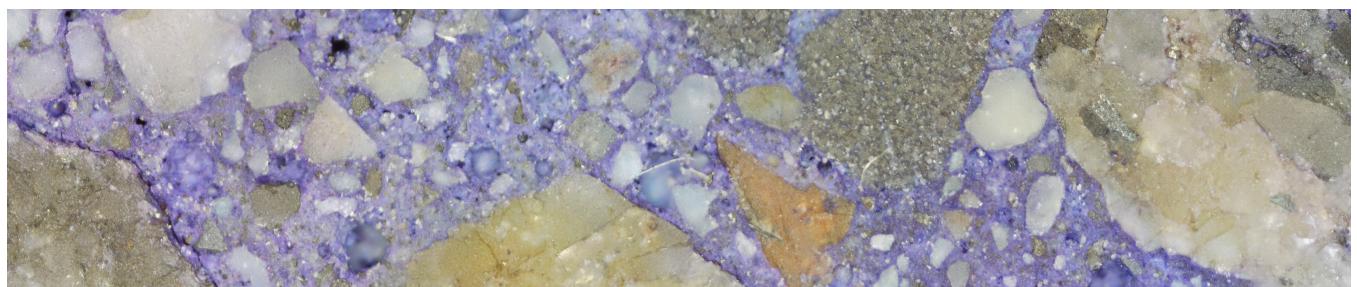


FIGURE 6.3 – Assemblage OpenCV agrandit

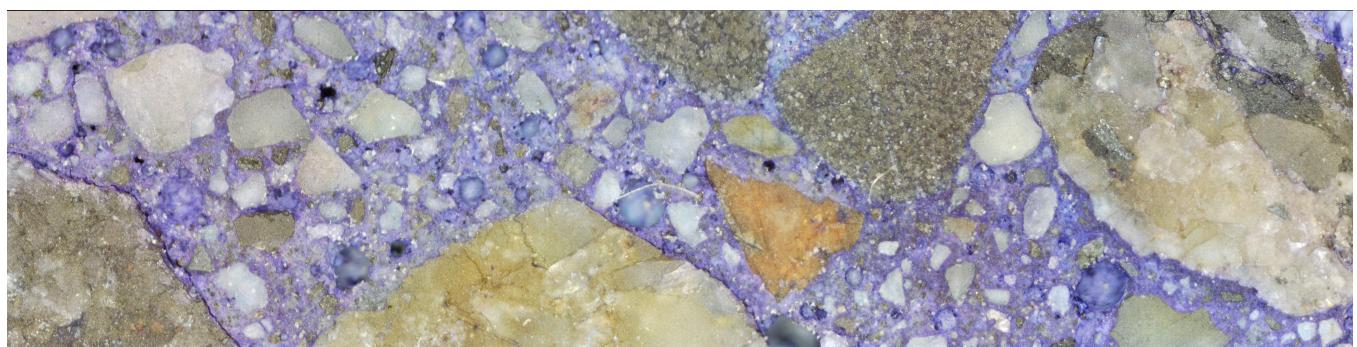


FIGURE 6.4 – Assemblage MIST agrandit

En comparant les images, on peut voir, premièrement, que celles-ci n'ont pas la même hauteur. L'option « crop » a été activée dans les paramètres de l'algorithme d'Opencv pour enlever les bandes noires issues de l'erreur d'orientation du microscope. Cela explique la différence de hauteur entre les deux images. Ensuite, en observant les parties agrandies de ces deux assemblages, on remarque que le chevauchement des images pour l'assemblage d'OpenCV n'est pas parfait. En effet, cela est subtil, mais on voit que certaines roches ne sont pas complètes. Lors des essais, il a été observé que l'algorithme d'OpenCV nécessite plus de chevauchement entre les images sinon l'assemblage échoue ou se réalise partiellement. Cela vient du fait que l'algorithme utilisé par OpenCV est entièrement basé sur la correspondance de descripteur invariant. Si le chevauchement n'est pas assez grand pour en laisser une quantité suffisante, l'algorithme ne pourra pas trouver de correspondance telle qu'illustrée à la figure 6.5.

Contrairement à l'algorithme d'OpenCV, l'algorithme MIST nécessite moins de chevauchement pour être en mesure de faire correspondre les images. Cela est possible grâce à l'estimation initiale du positionnement des images selon leur identification dans la grille de chevauchement. En effet, chacune des images possède un nom de fichier permettant de les positionner dans cette grille. Par exemple, le programme de ce projet nomme les images selon la nomenclature suivante : xc-cyrr.jpeg. Dans la figure 6.6 représentant le menu du *plugin* MIST, une grille est affichée et illustre bien ce concept. En termes de performances, l'algorithme MIST est supérieur à celui d'OpenCV pour deux raisons. Premièrement, la correspondance entre les images est plus rapide avec MIST pour le même niveau de chevauchement entre les deux algorithmes. Deuxièmement, le niveau de chevauchement minimum requis est plus petit avec MIST, ce qui permet de réduire considérablement le nombre d'images requis et ultimement réduire le temps de calcul ET de déplacement. Un test a été effectué avec un chevauchement de 20% et l'algorithme MIST a pu obtenir un temps d'assemblage de 13 secondes (7 secondes de moins qu'avec un chevauchement de 44%).

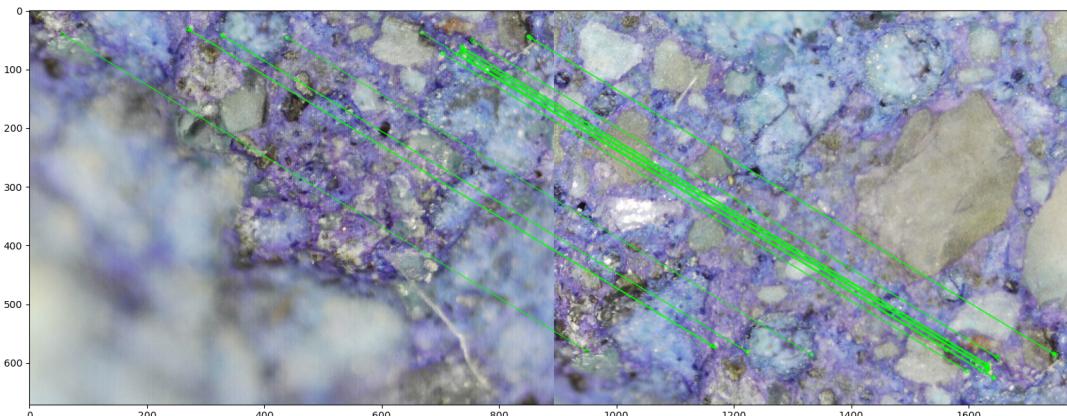


FIGURE 6.5 – Correspondance de descripteurs invariants OpenCV

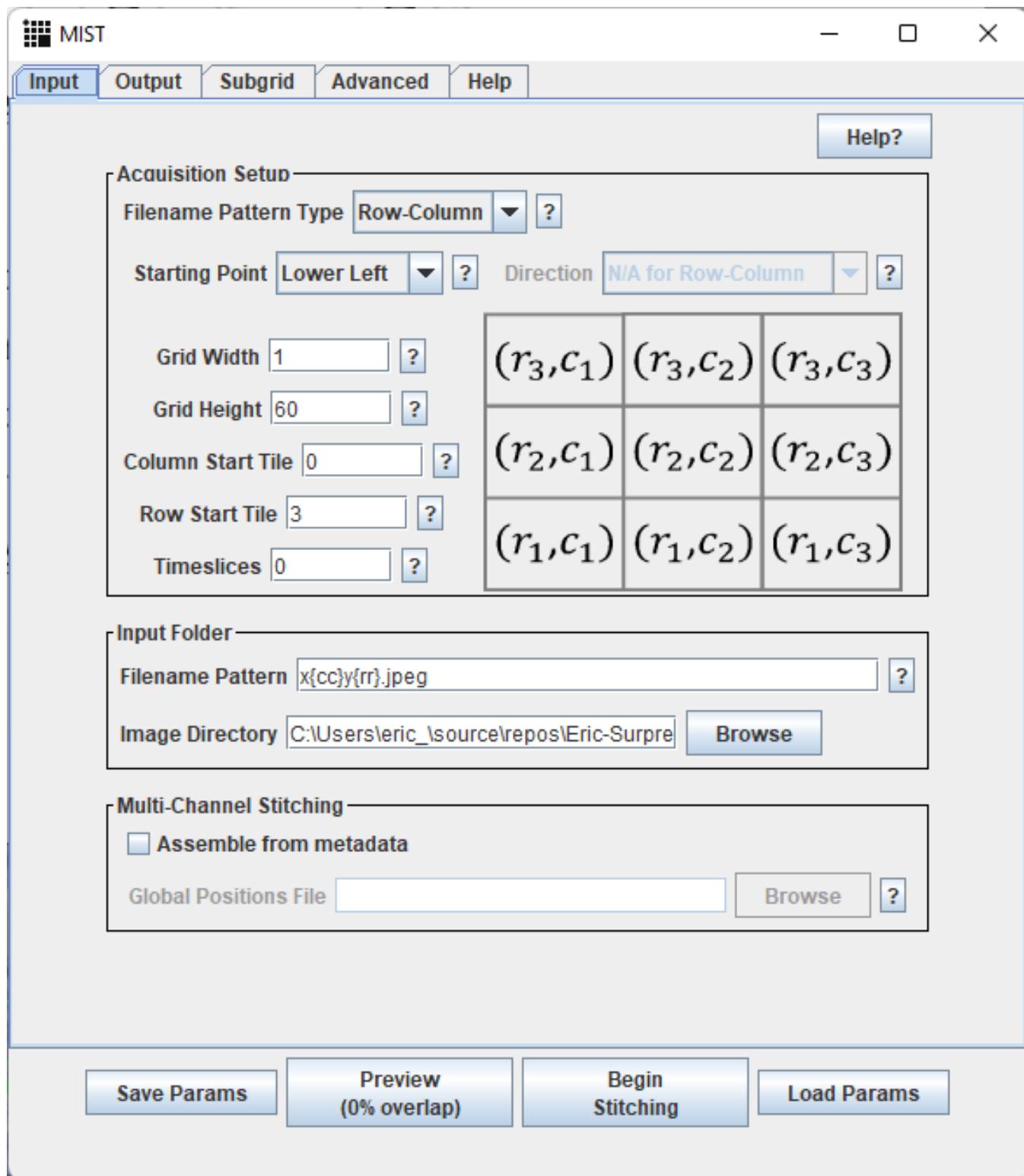


FIGURE 6.6 – Menu du plugin MIST dans le logiciel FIJI

6.2 Résultats du modèle de détection des caractéristiques

6.2.1 Fonction pour calculer la grosseur de pixel

Pour tester la fonction Matlab, une image du gabarit a été prise en se positionnant à une hauteur permettant d'obtenir un bon focus sur la bande de calibration de 0.076mm. L'image a été ensuite fournie à la fonction Matlab et a retourné une différence de hauteur de 12 pixels entre les extrémités (bon alignement) et une grosseur de pixel de 0.0008 mm. Les figures qui suivent illustrent les images obtenues à la suite de chaque traitement.

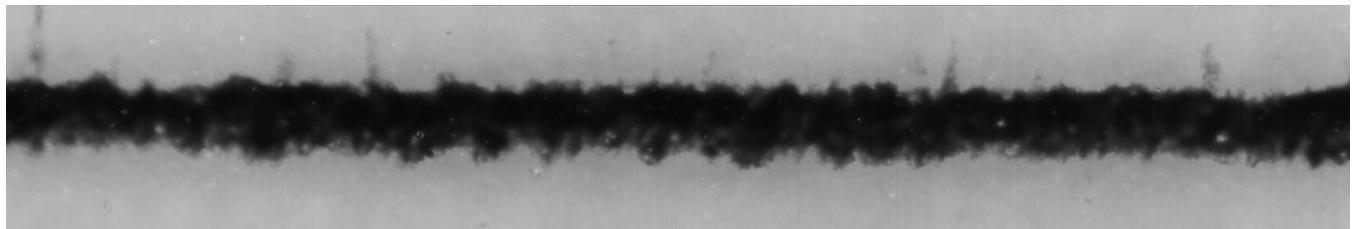


FIGURE 6.7 – Bande 0.076mm après conversion à niveau de gris

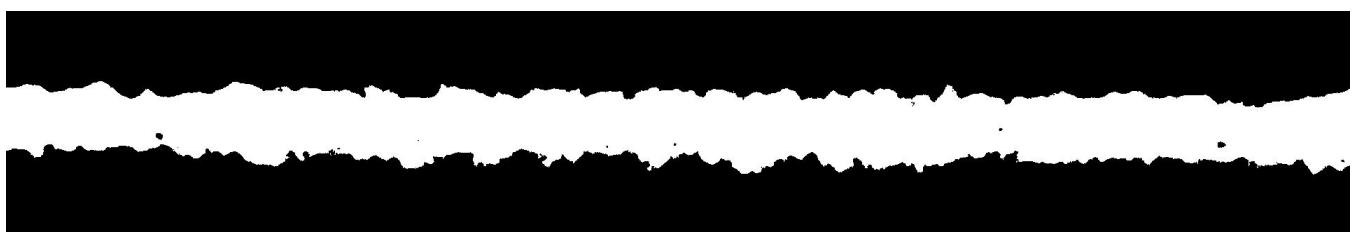


FIGURE 6.8 – Bande 0.076mm après seuillage ($i < 90$)

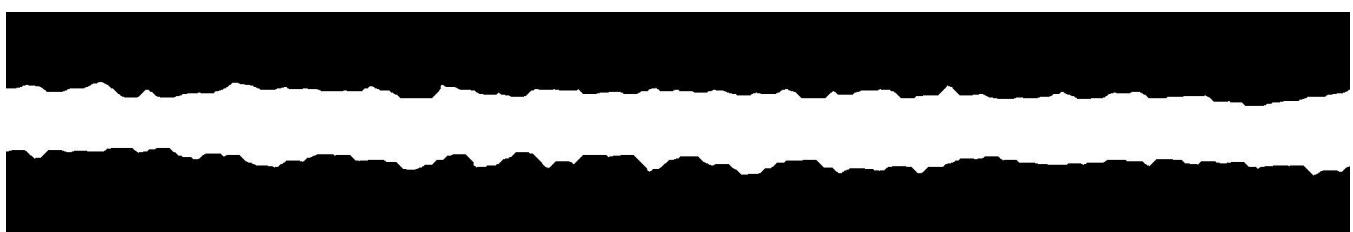


FIGURE 6.9 – Bande 0.076mm après opérations morphologiques

Pour valider la précision de la fonction, la grosseur de pixel retourné a été multipliée par la longueur en pixel du scan d'une traverse. La longueur d'une traverse complète est d'environ 100mm (diamètre du cylindre de béton). L'image de la traverse utilisée avait une longueur de 120484 pixels, ce qui a donné : $0.0008 \times 120484 = 96.3872$ mm. Considérant la tolérance du gabarit et de l'échantillon de béton, le résultat obtenu est satisfaisant.

6.2.2 Segmentation conventionnelle

Le tableau 6.1 contient les résultats de similarité obtenus avec les meilleurs paramètres selon l'analyse ROC qui a été faite pour chacune des techniques conventionnelles.

TABLE 6.1 – Comparaison segmentation

Algorithme	Similarité (%)		Temps (s/segm)	Meilleurs paramètres	
Clustering Based Segmentation	Pâte 0.8137	Roches 0.9559	1.4825	Pâte K=2 T=0.001	Roches K=2 T=0.001
Threshold Based Segmentation	Pâte 0.6154	Roches 0.9193	0.1295	Pâte RGB min=150 RGB max=255	Roches RGB min=155 RGB max=255
Region-Based Segmentation	Pâte 0.5194	Roches 0.8706	13.5922	Pâte ME=15 Rayon=2	Roches ME=55 Rayon=6
Edge Based Segmentation	Pâte 0.3471	Roches 0.4998	0.4848	Pâte M=Roberts T=0.05	Roches M=canny T=0.01

Voici quelques images de segmentations obtenues pour chacune des techniques conventionnelles.

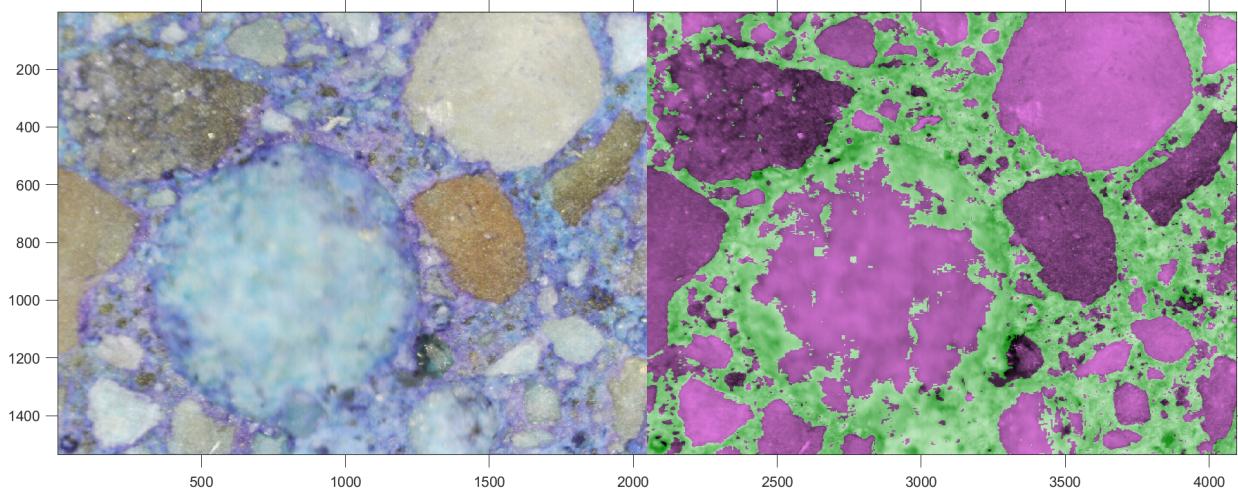


FIGURE 6.10 – Segmentation automatique avec k-mean

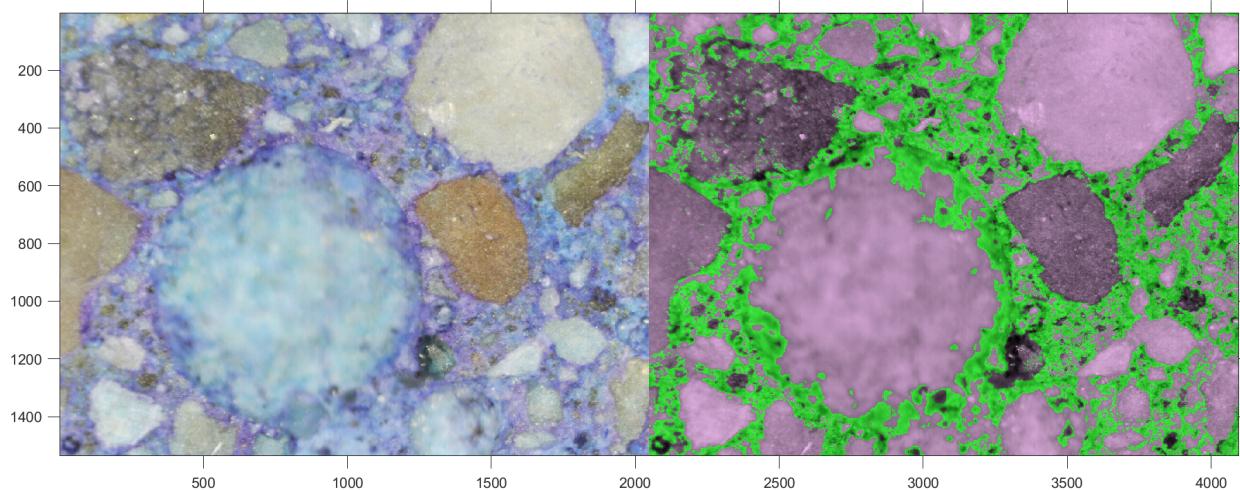


FIGURE 6.11 – Segmentation par seuillage avec couleur RGB

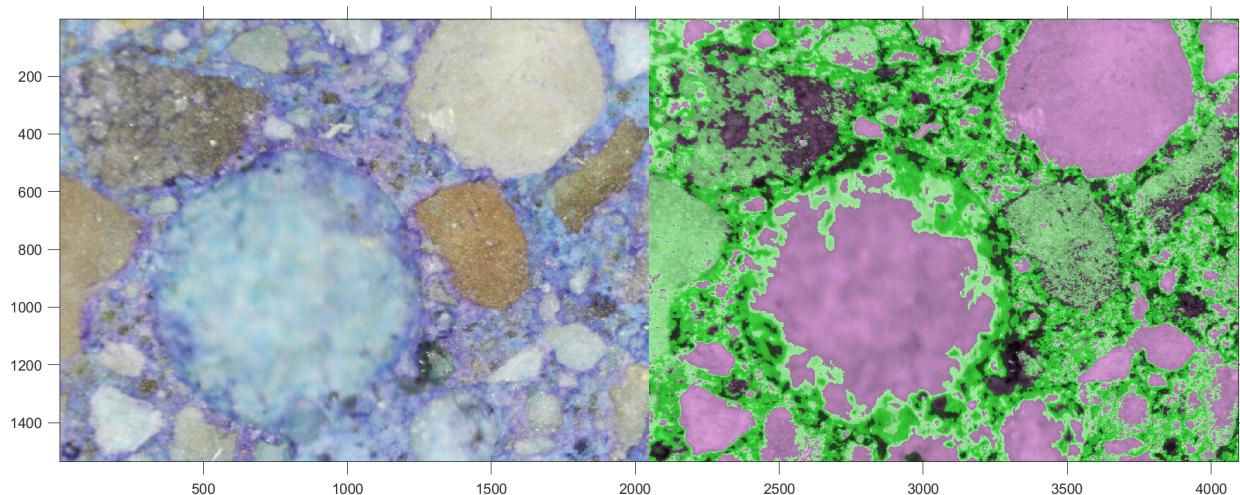


FIGURE 6.12 – Segmentation par region avec algorithm de croissance par région

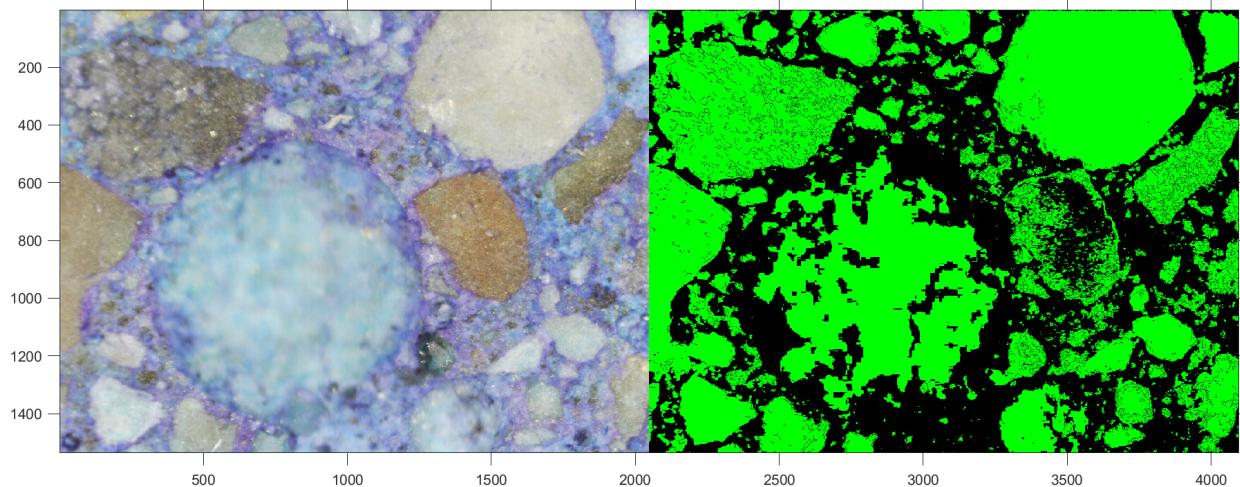


FIGURE 6.13 – Segmentation de détection par contour avec Canny

Lors du processus d'analyse et d'essai des méthodes de segmentation conventionnelle, il a été réalisé rapidement qu'aucune technique ne permettait de segmenter de façon satisfaisante les quatre classes d'images. Du moins, en une seule étape de segmentation. Pour obtenir des métriques afin de comparer avec la méthode de segmentation sémantique, seule les classes « roches » et « pâte » ont été segmentées. Ces métriques permettent de conclure que la technique de segmentation par « k-mean » est celle offrant la meilleure précision pour détecter les classes « roches » et « pâte ». Si ce projet de recherche devait se poursuivre, il serait intéressant de tester davantage de paramètres et de combiner plusieurs méthodes de segmentation ensembles. Par exemple, la méthode « k-mean » permet de faire une partition approximative des classes. À partir de cette partition, il serait possible d'appliquer une détection de contour combinée à des opérations morphologiques pour mieux délimiter les contours des bulles. Néanmoins, ce type de combinaisons pourrait impliquer une grande charge de calcul.

6.2.3 Segmentation sémantique avec « Deep Network »

Le tableau 6.2 contient les métriques retournées par la fonction « evaluateSemanticSegmentation(seg,truth) » ayant été exécutée avec un lot de 8 images « test » préalablement identifiées. Chaque métrique obtenue permet de conclure plusieurs choses.

Le « GlobalAccuracy » est le ratio de pixels bien identifiés, toutes classes confondues, sur le nombre total de pixels. Une valeur de 0.89686 a été obtenue, donc de façon générale le modèle a su bien identifier la majorité des pixels.

Le « MeanAccuracy » est la moyenne de la précision de chaque classe. Une valeur de 0.59602 a été obtenue pour cette métrique. Le fait que cette précision soit plus basse que le « GlobalAccuracy » s'explique par la différence de proportion entre les classes. En effet, les classes « roche » et « pâte » représente beaucoup de pixels dans les images et elles ont eu un bon « score » de classification donc elles ont grandement contribué au « GlobalAccuracy ». Les classes « vide d'air » et « vide de roche », quant à elle, représentent peu de pixels et ont eu de moins bons scores de classification. La moins bonne précision de classification pour ces deux dernières classes s'est mieux reflétée dans le « MeanAccuracy » du fait que le poids des classes est considéré.

Le « MeanIoU » où « IoU » signifie « Intersection over union », aussi connu sous le nom « coefficient de similarité de Jaccard » est une des métriques les plus utilisées. À la différence des deux dernières métriques, celle-ci pénalise la fraction de faux positifs. Pour chaque classe, « IoU » est le rapport entre les pixels correctement classés et le nombre total de pixels de référence et de pixels prédits. Pour cette raison, le résultat obtenu pour cette métrique (0.52023) est plus faible.

De façon similaire au « MeanAccuracy », le « WeightedIoU » est une moyenne du « IoU » de chaque classe pondérée sur le nombre de pixels que celles-ci représentent. Le fait que le score de cette métrique soit plus élevé que la précédente démontre que ce sont les classes ayant le plus de pixels qui ont le plus de faux positifs.

Finalement, un test pour déterminer la performance du modèle en termes de rapidité a aussi été effectué sur les huit images. Une moyenne de 0.522 seconde par image a été obtenue.

L'image segmentée à la figure 6.14 est le résultat d'un test effectué avec le modèle et une image différente des ensembles d'images d'entraînement et de test.

TABLE 6.2 – Résultat performance modèle

GlobalAccuracy	MeanAccuracy	MeanIoU	WeightedIoU	MeanBFScore
0.89686	0.59602	0.52023	0.82646	NaN

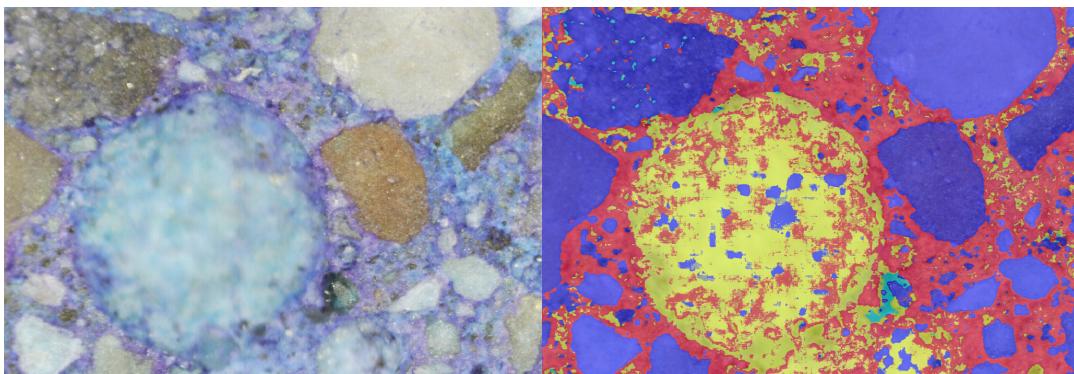


FIGURE 6.14 – Segmentation sémantique avec « Deep learning »

L'analyse de la figure 6.14 permet de conclure que le modèle a su identifier les classes de façon générale. Le modèle semble repérer les bulles d'air (jaune) et les distinguer des vides laissés par les roches (turquoise). De plus, les roches (bleu) semblent être facilement distinguées de la pâte (rouge). Cette excellente performance pour la classification des roches est attribuable au grand contraste créé par le bleu de méthylène appliqué sur l'échantillon. Néanmoins, le modèle semble avoir de la difficulté à détecter entièrement les bulles d'air (parsemé de zone de pâte et de roches à l'intérieur). Aussi, beaucoup de faux positifs sont présents dans la pâte. Cette difficulté n'est pas surprenante vu la grande ressemblance entre les bulles et la pâte en termes de couleur. En fait, la seule façon de les distinguer est par leur contour définissant une géométrie circulaire et l'effet de lissage à l'intérieur. La segmentation incomplète des bulles et les faux positifs dans la pâte sont problématiques considérant la méthode pour compter les caractéristiques qui vient ensuite. Toutefois, avec un peu plus d'entrainement et d'ajustement dans l'architecture du modèle, il est évident que celui-ci serait en mesure de faire une détection nettement supérieure des caractéristiques, ce qui le rendrait suffisamment précis pour un usage commercial.

Finalement, il est important de noter que la segmentation sémantique est la seule technique ayant permis de classifier les quatre caractéristiques dans une certaine mesure. Avec les autres techniques, il a été seulement possible de classifier la roche et la pâte. Avec plus de perfectionnement, il est fort probable que la combinaison d'une ou plusieurs de ces techniques de segmentation conventionnelle aurait pu permettre de classer les quatre caractéristiques. Toutefois, cela aurait été beaucoup plus complexe que de développer un modèle d'apprentissage comme il a été fait. En fait, ceci est justement une des raisons qui rend très populaires les réseaux de neurones convolutifs dans le domaine de classification d'images. En effet, les modèles d'apprentissage, une fois l'architecture déterminée, ne nécessitent pas de trouver les bons paramètres comme avec les méthodes de segmentation conventionnelle. C'est pourquoi cette technique de segmentation est capable de fournir de bons résultats rapidement.

Analyse financière de la solution

Dans cette section, une analyse de la rentabilité financière du système proposé a été faite dans une perspective où une version « production » serait fabriquée. Cette analyse se base sur les données du guide AFG 2022¹ et les statistiques de Groupe ABS en 2021.

TABLE 7.1 – Comparaison

	Sans système (actuel)	Avec système
Prix de l'essai AFG (2021)	323\$/essai	323\$/essai
Temps moyen requis (humain)	3.00 hrs/essai	0.25 hrs/essai
Taux horaire auxiliaire ABS (2021)	25\$/h	25\$/h
Bénéfice après salaire	248 \$/essai	317\$/essai

Dans le tableau 7.1, les taux horaires sont basés sur la moyenne salariale des auxiliaires de laboratoire. Le temps requis actuellement pour l'essai (sans système) a été évalué à 3h en faisant une moyenne des temps de plusieurs essais avec différents auxiliaires. Le temps requis *avec système* (15 minutes) a été estimé en fonction des manipulations requises pour préparer et installer l'échantillon. La préparation inclut l'application du bleu de méthylène et de la poudre blanche.

TABLE 7.2 – Impacts

Économie par essai	69\$/essai
Nombre d'essais réalisé (2021)	188 essais
Économie par année (2021)	12 925 \$

Dans le tableau 7.2, en calculant la différence de bénéfice après salaire, nous obtenons une économie par essai de 63\$ qui donne une économie par année de 11 750\$ si multiplié par le nombre d'essais réalisé en 2021 (188).

1. <https://afg.quebec/documentation/publications>

TABLE 7.3 – Investissements

Plateforme de positionnement XY	3000 \$
Système d'imagerie	5000 \$
Support, fixture et autres	1000 \$
R&D (160h à 30\$/h)	4800 \$
Total	13 800 \$

Dans le tableau 7.3, les investissements impliquent d'une part l'achat d'équipement et d'autre part le temps de R&D. La construction du système requiert principalement l'achat d'une plateforme XY et d'un microscope avec caméra digitale. Les prix prévus pour ces équipements sont basés sur la recherche qui a été faite à la section 3.3. Des supports et fixtures devront être fabriqués, mais la majorité peut être faite à l'interne avec les équipements d'usinage disponibles. Le 1000\$ prévu à cet effet inclut aussi les achats de petit montant tels que des câbles de communication, vis, etc. Le temps de R&D a été estimé en fonction du temps qui a été requis jusqu'à présent pour développer le prototype et ce qui reste à faire pour le rendre 100% fonctionnels.

TABLE 7.4 – Rentabilité

Économie	12 925/ années \$
Cout d'investissement	13 800 \$
Délais de rentabilisation	1.07 années \$

Finalement, le dernier tableau 7.4 exprime la rentabilité financière en termes de délais de récupération. Un délai d'environ une année a été calculé pour rembourser le coût d'investissement initial. Toutefois, deux autres facteurs sont à considérer.

1. Ce système va permettre de réduire les délais d'envois de résultat aux clients et ainsi augmenter le niveau de service de Groupe ABS. Par conséquent, il est fort à parier que la clientèle pour ce type d'essais augmentera, générant ainsi des revenus supplémentaires.
2. La commercialisation du système est une possibilité très intéressante, car à l'heure actuelle, très peu sinon aucun système entièrement automatisé n'existe au Québec. La commercialisation du système permettrait de faire des ventes dans des régions non desservies par Groupe ABS, ce qui engendrait des bénéfices considérables.

Conclusion

Ce projet avait pour but de concevoir un système automatisé d'analyse de réseau de vide d'air de béton durci tout en respectant la norme ASTM-C457. Le prototype qui a été fabriqué au cours de ce projet a permis de tester plusieurs techniques d'assemblage d'images et de segmentation. Toutefois, le cadre de ce projet a permis seulement de tester ces techniques de façon superficielle. Avec plus de temps, il aurait été possible d'adapter et combiner chacune des techniques de segmentation avec des étapes de pré/post traitement afin d'exploiter tout leur potentiel. Malgré tout, les résultats obtenus permettent d'arrêter notre choix sur l'algorithme MIST pour l'assemblage d'image et la segmentation sémantique par « Deep Network » pour la segmentation d'images. En effet, ces deux techniques se sont avérées nettement supérieures dans tous les critères de comparaison. Nous croyons qu'avec un peu plus de développement, ces techniques permettraient de développer un système suffisamment précis pour être utilisé dans un contexte réel. La poursuite du développement n'est pas seulement justifiée par des résultats techniques satisfaisants, mais aussi par une analyse financière affichant une très bonne rentabilité et de belles opportunités d'affaires (commercialisation du système). Finalement, en regard aux objectifs établis au début de ce projet, nous croyons que le concept du système ayant été développé permettrait de les atteindre tout en s'inscrivant dans une perspective de développement durable.

Bibliographie

- [1] H-2964a linear traverse machine. <https://www.humboldtmfg.com/linear-traverse-machine-astm-c457b.html>. [Consulté 20 août 2022].
- [2] Infinityhd. <https://www.lumenera.com/infinityhd.html>. [Consulté 20 août 2022].
- [3] Motorized xy linear stage one-xy300. <https://www.newport.com/p/ONE-XY300>. [Consulté 20 août 2022].
- [4] pco.panda 26 ds. <https://www.pco-tech.com/scientific-cameras/pcopanda-26-ds/>. [Consulté 20 août 2022].
- [5] Rapidair 457. <https://germann.org/products-by-application/air-void-system-parameters/rapidair>. [Consulté 20 août 2022].
- [6] Segmentation d'image. https://fr.wikipedia.org/wiki/Segmentation_d' image. [Consulté 20 août 2022].
- [7] V-731 high-precision xy stage. <https://www.pi-usa.us/en/products/precision-motorized-linear-stages/integrated-xy-precision-motorized-stages-planar-xy-sv-731-high-precision-xy-stage-1201908/>. [Consulté 20 août 2022].
- [8] Joe Chalfoun, Michael Majurski, Tim Blattner, Kiran Bhadriraju, Walid Keyrouz, Peter Bajesy, and Mary Brady. Mist : accurate and scalable microscopy image stitching tool with stage modeling and error minimization. *Scientific reports*, 7(1) :1–10, 2017.
- [9] Todd D Hanson et al. Evaluation of the rapidair 457 air void analyzer. Technical report, Iowa. Dept. of Transportation. Highway Division, 2012.
- [10] UH Jakobsen, C Pade, N Thaulow, D Brown, S Sahu, O Magnusson, Sandra De Buck, and Geert De Schutter. Automated air void analysis of hardened concrete ?a round robin study. *Cement and Concrete Research*, 36(8) :1444–1452, 2006.
- [11] Zhichao Liu, Will Hansen, and Bo Meng. Characterisation of air-void systems in concrete. *Magazine of Concrete Research*, 68(4) :178–186, 2016.
- [12] Hoskin Scientifique LTÉE. Hs-745t air voids analysis system. https://www.hoskin.qc.ca/catalog/index.php?main_page=product_info&cPath=2_795_1559_1942&products_id=3518. [Consulté 20 août 2022].
- [13] Adrian Rosebrock. Opencv panorama stitching. *OpenCV tutorials*, 2016.