



HAL
open science

segmentation appliquée à la métrologie de matériaux dans des images de microscopie électronique

Julien Baderot

► **To cite this version:**

Julien Baderot. segmentation appliquée à la métrologie de matériaux dans des images de microscopie électronique. Traitement du signal et de l'image [eess.SP]. Université Grenoble Alpes [2020-..], 2021. Français. NNT : 2021GRALT008 . tel-03256727

HAL Id: tel-03256727

<https://theses.hal.science/tel-03256727>

Submitted on 10 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

Spécialité : SIGNAL IMAGE PAROLE TELECOMS

Arrêté ministériel : 25 mai 2016

Présentée par

Julien BADEROT

Thèse dirigée par **Michel DESVIGNES**, Professeur, Université Grenoble Alpes

et codirigée par **Laurent CONDAT**, CNRS

et **Mauro DALLA MURA**, Grenoble INP

préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique**

dans l'**École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

segmentation appliquée à la métrologie de matériaux dans des images de microscopie électronique

segmentation applied to material metrology of electronic microscopy images

Thèse soutenue publiquement le **1 février 2021**,
devant le jury composé de :

Monsieur MICHEL DESVIGNES

PROFESSEUR DES UNIVERSITES, GRENOBLE INP, Directeur de thèse

Madame MARIANNE CLAUSEL

PROFESSEUR DES UNIVERSITES, UNIVERSITE DE LORRAINE, Rapporteur

Monsieur THIERRY GÉRAUD

PROFESSEUR, EPITA, Rapporteur

Monsieur OLIVIER MICHEL

PROFESSEUR DES UNIVERSITES, GRENOBLE INP, Président du jury

Monsieur JOHANN FOUCHER

INGENIEUR DOCTEUR, Pollen Metrology, Examineur

Monsieur FABRICE MERIAUDEAU

PROFESSEUR DES UNIVERSITES, UNIVERSITE DE DIJON - BOURGOGNE, Examineur



Résumé : Ces travaux de thèse s’inscrivent dans le cadre industriel de la métrologie pour le semi-conducteur. Le développement de nouvelles technologies de transistors et mémoires demande des efforts intensifs en métrologie. L’industrie actuelle répond bien à ce besoin en métrologie au stade de la production. Cependant, sur les étapes de recherche et développement, la grande majorité de la métrologie est faite manuellement et compte pour près de la moitié du temps de recherche et développement. La grande contrainte de cet aspect manuel est la faible quantité de données annotées et la variabilité des cas d’utilisation. D’un autre côté, les objets à mesurer varient peu au sein d’un même cas d’utilisation et nous avons à disposition un *design* de l’objet, qui est une version idéalisée de l’objet. Pour répondre à cette problématique, les travaux de cette thèse sont découpés en trois parties.

Tout d’abord, nous avons analysé les données à notre disposition. Les images proviennent de microscopes électroniques en transmission. Nous avons étudié le fonctionnement de ce type de microscope ainsi que les différentes façons de faire l’acquisition d’images. Cette étude a mis en évidence des modèles de bruit différents selon les paramètres d’acquisition et l’échantillon. Nous avons validé expérimentalement que le modèle de bruit est celui d’un bruit de Poisson dans nos images. De plus, cela a permis de poser les bases de notre modèle d’image pour mettre en place une méthode d’analyse. Après un état de l’art, nous avons mis en place un filtrage adapté. De plus, plusieurs modalités sont disponibles. Ces modalités mettent en évidence différentes caractéristiques de l’échantillon. Nous avons constaté une relation linéaire entre les modalités étudiées.

Dans un deuxième temps, nous avons développé une méthode de segmentation des matériaux basée sur le *design* (image schématisée représentant la configuration des matériaux dans les grandes lignes). Le but de cet algorithme est d’utiliser le *design* comme élément définissant la segmentation attendue. Nous avons utilisé l’Arbre de Formes, une structure de données qui organise les régions de l’image de manière hiérarchique avec des opérateurs morphologiques. Cette structure représente efficacement le *design* pour faire une mise en correspondance avec une image de l’objet. Cette mise en correspondance est faite de manière similaire au Graph Cut, c’est-à-dire en minimisant une énergie à travers l’Arbre de Formes de l’image. Cette minimisation s’appuie sur plusieurs caractéristiques tels que le niveau de gris et la position qui sont deux informations importantes venant du microscope et du *design*. Enfin la régularisation contraint la résolution du problème pour obtenir une segmentation cohérente avec le *design*.

Enfin, nous avons mise en place un ensemble d’algorithmes pour mesurer des objets de manière automatique. Cette mesure est définie par l’utilisateur afin de pouvoir s’adapter aux différents cas. La première étape utilise l’analyse de données précédente pour effectuer un filtrage adapté. Nous avons ensuite une étape de détection d’objets pour avoir la localisation des objets sur l’image. Pour cela nous avons développé une méthode utilisant des descripteurs de réseaux de neurones convolutifs et une fenêtre glissante. La troisième étape utilise la méthode de segmentation présentée dans ce manuscrit afin d’extraire les matériaux des objets en accord avec la définition du *design*. Enfin, nous présentons dans cette thèse deux méthodes de remplacement de mesures afin d’améliorer la précision de celles-ci. Les travaux de cette thèse ont été ou seront intégrés dans le produit de Pollen Metrology et sont utilisés au quotidien par une cinquantaine de personnes chez deux clients industriels différents.

Mots clefs : microscope électronique en transmission, arbre de formes, segmentation, détection d’objet, métrologie

Abstract : This thesis work is included in the context of the metrology of objects from the semiconductor industry. The development of new technologies of transistors and memories requires tremendous efforts for the metrology. The industry has solutions to this need for the steps of the production. However, during the research and development steps, the vast majority of the metrology is done by hand and represent half of the time of the engineers. The biggest constraint is the low quantity of annotated data and the variability of the use cases. On the other hand, the objects we want to measure have a low variability inside a use case. In addition, we have access to a design which is an ideal version of the object. To solve this challenge, this thesis is split into three parts.

First, we analyzed the available data. The images are acquired with transmission electron microscopes. We studied the working of the microscope and the different modalities of acquisition. This study highlighted different noise models depending on the sample and the acquisition parameters. We experimentally validated the Poisson noise model on our images. Based on this assumption, we created a model from our images to design an appropriate method. After a study of the state of the art, we used an adapted filter for the preprocessing. We also shown that modalities are linearly dependent. This modalities reveals different characteristics of the sample.

In a second time, we developed a segmentation method based on the design. The goal of this algorithm is to use the design as the target defining the resulting segmentation. We used the Tree of Shapes, a structure which hierarchically organize regions of the image with morphological operators. This structure represents efficiently the design to perform matching with the image of the object. This matching is done in a similar way as the Graph Cut by minimizing an energy through the Tree of Shapes of the image. The minimization uses several characteristics such as the grey level intensity and the position which are two important information coming from the microscope and the design. Finally, the regularization constraints the solving of the problem in order to obtain a resulting segmentation that is coherent with the design.

Finally, we designed a pipeline of algorithm to perform automatic measurements of objects. The measures are defined by a user to be able to adapt for all cases. The first step uses the analysis on the data done previously to perform an adapted filter. We then developed an object detection algorithm to retrieve the localization of objects in the image. This algorithm is based on convolutionnal neural networks features and sliding window. The third uses the segmentation method described previously to extract the materials from the object based on the design. Finally, we developed two methods for the replacement of measures on the detected objects to increase the precision of the measures. The work of this thesis is integrated in the product of Pollen Metrology and is used on a daily basis by around fifty persons by two industrial customers.

Keywords : transmission electron microscopy, tree of shapes, segmentation, object detection, metrology

Remerciements

Les travaux de cette thèse n'auraient pas été possibles sans l'aide et le soutien de nombreuses personnes pendant ces trois ans de découverte, de galère, de joie, de déception, de réussite, de recherche...

Je tiens tout d'abord à remercier Alice, ma future femme, pour son soutien au quotidien. Tu as à la fois réussi à me supporter du début de la thèse lorsque j'étais le plus motivé jusqu'au bout de la rédaction alors que j'étais moins disponible et agréable à vivre. Ton amour m'a aidé à tenir bon dans les moments de doute et m'a permis d'aller jusqu'au bout de cette périlleuse aventure. Je suis sûr que tu es aussi soulagée que moi que la thèse soit finie.

Ensuite je remercie Michel, Mauro et Laurent pour m'avoir guidé pendant ces trois ans, m'avoir donné tout ces conseils pour être capable de réaliser ces travaux de recherche et de manière plus générale de m'avoir appris par votre expérience le travail de la recherche. Vous avez su pointer mes manques et m'orienter pour éviter que je m'éparpille. Votre soutien a pu porter cette thèse jusqu'à sa conclusion. Grâce à vous, les travaux de cette thèse ont abouti à une conclusion utile pour l'entreprise et je l'espère le monde de la recherche. Vous avez enfin rendu ce manuscrit compréhensible aux personnes qui ne sont pas dans ma tête avec toutes vos corrections.

Enfin et pas des moindres, je remercie mes collègues avec dans le désordre : Alex, Guilhem, Johann, Sergio, Aurélien et Aurélien, Guillaume, Eric, Yann, Benoît, Thibault, Nico, Mat, Marion, Hervé, Lucie, Olivier et tout le reste de la team Pollen qui m'a vu dans le fond du bureau faire ma thèse. Vous m'avez soutenu pour les directions que devaient prendre mes travaux, la matérialisation des idées en code et enfin au quotidien autour de viennoiseries, de café et de babyfoot. La bonne ambiance qui régnait m'a beaucoup aidé à garder la motivation pour aller jusqu'au bout.

Mention spéciale pour Milord, Milka et Rune, nos trois chiens qui malgré leur incompréhension à l'égard des travaux de cette thèse ont su me détendre et m'aider à décompresser avec leur petite bouille pleine d'amour.

Pour finir, merci à tous d'avoir fait ce bout de chemin avec moi et de continuer sur ma route pour beaucoup d'entre vous, vous avez contribué à faire de moi la personne que je suis aujourd'hui.

Table des matières

1	Introduction et contexte industriel	9
1.1	Contexte industriel	9
1.2	Objectifs de la thèse	12
1.3	Contributions	17
1.4	Notations	19
2	Présentation des données	21
2.1	Microscopie électronique	21
2.1.1	Présentation des objets et leurs dimensions	21
2.1.2	Microscopie pour les structures nanoscopiques	25
2.2	Modalités Transmission Electron Microscopy (TEM)	27
2.2.1	Présentation	27
2.2.2	Exploration des modalités	32
2.3	Bruit dans les images TEM	38
2.3.1	État de l'art	38
2.3.2	Hypothèse de bruit de Poisson et validation	39
2.3.3	Choix pour notre application	45
2.4	Conclusion	46
3	Segmentation d'images basée sur les matériaux	49
3.1	Problématique	49
3.2	Arbre de formes	52
3.2.1	Présentation	52
3.2.2	Application	56
3.3	Tree Cut segmentation	64
3.3.1	Introduction	64

Table des matières

3.3.2	Formulation du problème	64
3.3.3	Approche gloutonne	65
3.3.4	Framework proposé	69
3.4	Résultats	77
3.4.1	Impact des caractéristiques	77
3.4.2	Présentation des données et méthodes	79
3.4.3	Comparaison qualitative	81
3.4.4	Comparaison quantitative	86
3.4.5	Exemple d'application sur des images d'Imagerie par Résonance Magnétique (IRM)	89
3.5	Conclusion	92
4	Application au contexte industriel	95
4.1	Pipeline pour la mesure automatique d'objets	96
4.2	Détection d'objets	100
4.2.1	Présentation du contexte	100
4.2.2	Template matching	101
4.2.3	Méthode d'apprentissage classique	104
4.2.4	Méthode d'apprentissage profond	106
4.2.5	Comparaisons des résultats	114
4.3	Mesures	121
4.4	Réalisation expérimentale du pipeline	126
4.5	Conclusion	127
5	Conclusion	129
5.1	Contributions	129
5.2	Travaux futurs	131

Table des matières

Table des matières

1 Introduction et contexte industriel

1.1 Contexte industriel

Le monde du semi-conducteur vise à produire des modules de nano-électronique de plus en plus petits afin de proposer des puces plus performantes. Ces puces, qu'elles soient des microprocesseurs, des mémoires ou des cartes graphiques, sont massivement utilisées dans tous nos appareils électroniques du quotidien comme les téléphones, les voitures, l'électroménager ou encore les objets connectés [1], [2]. Cette industrie est guidée par la loi de Moore énoncée dans [3] et [4] qui est une loi empirique de ce secteur industriel décrivant cette réduction de taille au fil des ans. Cette loi empirique a servi de fil conducteur pour l'organisme responsable de communiquer au public les spécifications des nouvelles générations de puces, l'International Roadmap for Devices and Systems [5]. Si aux prémices de cette industrie, cette loi était vérifiée, ces dernières années, la taille des modules commencent à stagner à cause des contraintes physiques et des contraintes technologiques des procédés de fabrication [6]. L'industrie s'intéresse donc à des innovations permettant d'améliorer les performances des puces sans réduire leur taille. Cette combinaison de réduction de taille et de structures innovantes augmente de manière considérable la complexité de la fabrication des puces à tous les niveaux. La conséquence directe est l'augmentation importante des coûts de recherche et développement de ces technologies, au point que le nombre d'acteurs présent sur le développement des nœuds les plus avancés se compte sur les doigts de la

1 Introduction et contexte industriel

main [7].

La fabrication de semi-conducteurs nécessite de nombreuses étapes pour aller de la conception à la production [8] (Figure 1.1). L'étape de conception définit la fonctionnalité du composant. Ensuite, il faut définir le *design* qui permet de réaliser cette fonctionnalité. Ce *design* est une esquisse de la puce avec un premier essai de formes géométriques. L'implémentation permet de convertir ce *design* dans un format compatible avec les différents logiciels de simulation et de fabrication. La simulation teste virtuellement les propriétés du composant. Ensuite viennent les étapes de définition des matériaux, de vérification de la compatibilité des matériaux et d'extraction des effets parasites de la composition choisie. Ces étapes bouclent avec la simulation afin d'obtenir les exigences de performances du *design*. Pour toutes ces étapes, il existe tout un écosystème de logiciels couvrant les besoins. Les étapes de fabrication et de tests et vérification représentent la partie recherche et développement de la production. Le but est de reproduire le plus fidèlement le *design* défini précédemment. De nombreuses itérations sont faites à nouveau avec la simulation à cause des variations entre le *design* et ce qu'il est possible de fabriquer ainsi que les variations entre la simulation et la réalité. La fabrication et le test sont les étapes plus intensives en terme de métrologie manuelle de par la grande variabilité d'images et aussi la plus coûteuse car elle utilise des analyses destructives. Enfin l'étape de production désigne la fabrication à grande échelle dans les usines du composant. Cette étape requiert aussi beaucoup de métrologie mais dans ce cas les images sont très stables et des outils automatiques sont donc mis en place pour la mesure. Dans cette étape l'intérêt est davantage dans la reproductibilité de la mesure que dans sa précision, le but étant de détecter les variations.

Les deux types de métrologie introduites précédemment poursuivent des buts différents. Pour la fabrication, l'objectif est de mesurer des grandeurs dimensionnelles ponctuelles, dans le cas où elles sont faites manuellement, ou continues sur des images provenant de microscopes afin de vérifier que ces grandeurs sont en adéquation avec le *design*. Des mesures électriques sont réalisées en complément pour s'assurer que les dimensions définies

et finalement obtenues génèrent les caractéristiques voulues. Les appareils utilisés ont une grande précision, sont lents et peuvent être destructifs. Pour des raisons de coûts, certaines mesures sont faites avec les mêmes appareils que la production. On vise la justesse de mesure, c'est-à-dire la précision de la mesure. Pour la production, on ne peut pas utiliser d'appareils destructifs et le temps devient une contrainte car un grand nombre d'échantillons doivent être mesurés sur la chaîne de production. Les besoins sont similaires car des mesures dimensionnelles et électriques sont réalisées. Cependant, on vise la robustesse de la mesure, c'est-à-dire la répétabilité. L'objectif étant de repérer les variations de dimensions pendant la production pour ajuster les paramètres de fabrication en temps réel. Étant sur des chaînes de production, les mesures sont automatiques afin de contrôler en continu.

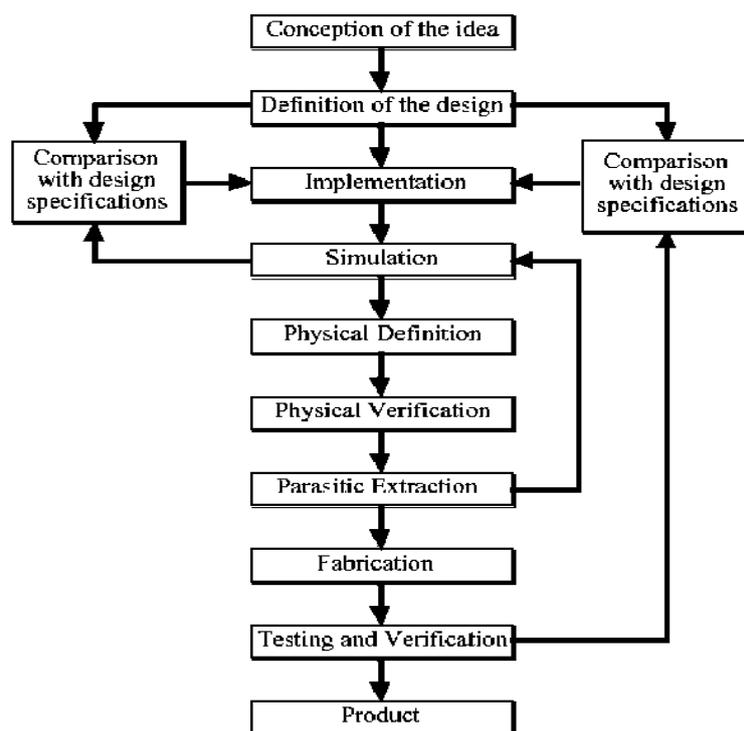


FIGURE 1.1 – Diagramme des différentes étapes de fabrication d'une puce

1.2 Objectifs de la thèse

Les travaux de cette thèse visent à réduire un poste de dépense important qui est la métrologie dimensionnelle liée aux différentes étapes de fabrication. La majeure partie de ces étapes de métrologie est réalisée manuellement et représente en moyenne la moitié du temps de travail d'un ingénieur. Elle consiste à mesurer sur des images de microscopie électronique des éléments caractéristiques des objets. Ce travail long et fastidieux peine à être automatisé dans cette industrie à cause de la variabilité des objets fabriqués. Les approches traditionnelles de traitement d'image nécessitent d'affiner les paramètres des algorithmes pour chaque cas et donc posent problème lorsque des utilisateurs non experts du domaine sont amenés à les modifier. De ce fait, nous viserons à automatiser au maximum les valeurs des paramètres des algorithmes soit en les fixant à une valeur par défaut soit en les calculant à partir des données.

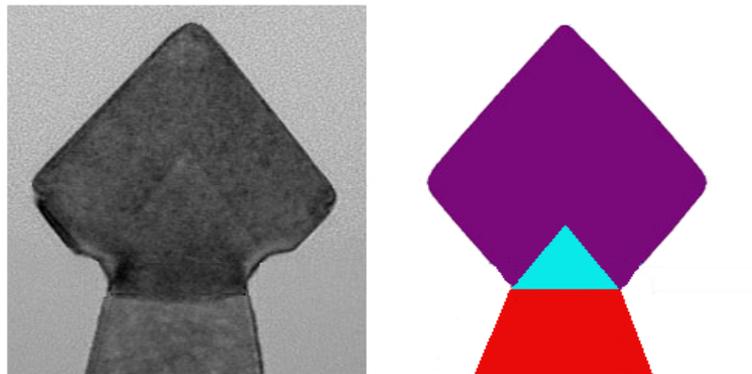
Nous considérerons deux profils d'utilisateurs. Tout d'abord, les ingénieurs procédés qui sont amenés à faire des mesures sur les objets qu'ils ont fabriqués. Leur domaine d'expertise est la gravure plasma et les dépôts de matériaux. Leurs connaissances et compétences en traitement d'image et apprentissage machine sont très disparates car certains utilisateurs développent des algorithmes de leur propre chef afin de faciliter leur travail de manière individuelle. Les métrologues sont spécialisés dans les techniques de microscopie utilisées pour obtenir des images de bonne qualité avec un processus répétable. Ils ont parfois des connaissances en traitement statistique. Leur but vis-à-vis de nos travaux de thèse est d'obtenir des mesures au moins aussi justes et plus rapidement qu'à la main. Les travaux de thèse visent aussi à être intégrés dans un logiciel, PlatypusTM, qui est le logiciel de métrologie vendu par Pollen Metrology et seront donc utilisés via cette interface.

Dans cette industrie, les objets à fabriquer sont toujours liés à un *design* de fabrication qui représente de manière simplifiée et idéale l'objet. Cette représentation sous la forme d'une image regroupe des informations de

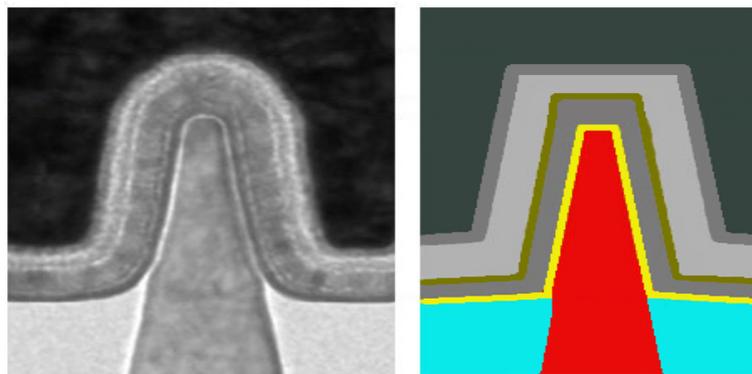
1.2 Objectifs de la thèse

matériaux, dimensionnelles et topologiques. La qualité des différents matériaux n'est pas nécessairement renseignée dans le *design* mais la distinction entre les matériaux est toujours présente. L'aspect dimensionnel est lui aussi parfois incomplet. La taille et la forme des régions en nanomètres ne sont pas strictement définies et les proportions de taille peuvent varier vis-à-vis des images réelles du fait du stade de maturité de la technologie étudiée. La complexité des objets varie, la majorité des objets sont représentés par deux à quatre matériaux mais certains d'entre eux sont composés d'une dizaine d'éléments. Enfin, ce *design* de fabrication possédé par les utilisateurs est une information cruciale pour eux. Nous en avons présenté les caractéristiques dimensionnelles mais ils comportent aussi des informations matériaux, électriques, mécaniques, de fabrication ... Cependant, le caractère confidentiel des *designs* nous les rendent inaccessibles et nous sommes amenés à les créer.

1 Introduction et contexte industriel



(a) Exemple de coupe de *fin* et son *design*



(b) Exemple de coupe de *gate* et son *design*

FIGURE 1.2 – Exemple d’images et leur *design*. Les images proviennent de [9] et les *designs* ont été réalisés par nos soins en fonction de la connaissance de cette industrie acquise au cours de la thèse.

1 Introduction et contexte industriel

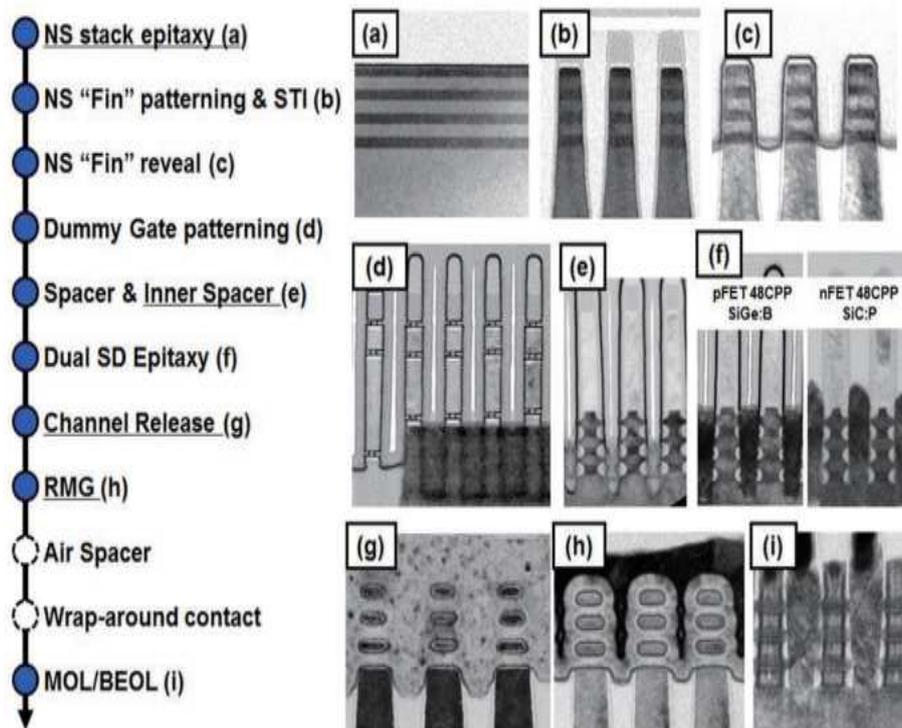


FIGURE 1.4 – Exemple de nanowire et quelques étapes de fabrication. Les images proviennent de [11].

1.3 Contributions

La première partie de ce manuscrit présente la microscopie électronique. Nous présentons différents types de structures de l'industrie du semi-conducteur. Les différents types de mesures sont définis et nous présentons les microscopes qui réalisent l'acquisition des images. Nous présentons aussi les différentes modalités d'images que les microscopes sont susceptibles de fournir. L'acquisition des différentes modalités étant simultanée, nous disposons dans ce cas d'images bi- ou tri-modales. La connaissance du fonctionnement de ces microscopes nous permet de mieux appréhender le contenu des images et d'adapter les techniques utilisées après une étude bibliographique adaptée sur les méthodes de filtrage. Nous avons aussi effectué une analyse du bruit présent sur ces images afin de pouvoir le filtrer de manière appropriée.

La deuxième partie propose une méthode de segmentation d'objets en fonction des matériaux afin d'aider à la mesure. Cette méthode de segmentation utilise le *design* de fabrication que nous appellerons par la suite *template*. Sans modéliser la chaîne d'acquisition complète, l'image est modélisée macroscopiquement par une partition en régions constantes par morceaux, dégradée par un lissage passe-bas, contaminée par des processus stochastiques ou bruit. L'information a priori du *design* est utilisée comme contrainte pour rendre la segmentation plus robuste aux différents défauts dans les matériaux ainsi qu'aux déformations possibles. De plus, le *template* est une information simple qui ne nécessite pas de compréhension des algorithmes. Le *template* peut être enrichi avec différentes informations comme le type de matériau ou la texture, qui sont à la disposition de l'utilisateur ou que l'on peut déduire des images réelles. Notre méthode de segmentation repose sur une structure de données, l'arbre de formes qui hiérarchise les formes d'une image. Cette structure est efficace pour représenter le *template* et nous permet de produire une segmentation guidée par le *template*. Nous comparons notre approche de segmentation à plusieurs approches de la littérature utilisées dans le domaine de la microscopie pour les semi-conducteurs et/ou plus génériques, y compris les

1 Introduction et contexte industriel

approches actuelles de deep-learning. Nous testons aussi cette approche de segmentation dans un autre cadre, celui de l'imagerie cérébrale à partir d'images d'IRM pour lesquelles nous pourrions définir un template idéal.

La troisième partie s'articule autour de l'application de notre algorithme de segmentation dans un cadre plus spécifique : la mesure automatique d'objets. La mesure d'objets est définie dans notre cas comme l'acquisition de grandeurs dimensionnelles, hauteur et largeur par exemple, d'objets sur une image. Pour cela nous avons mis en place un pipeline d'analyse découpé en plusieurs étapes. La première étape est une méthode de détection d'objets. Cette méthode est basée sur du template matching ou de l'apprentissage automatique afin de permettre la détection d'objets spécifique. Les approches basées sur les templates ont l'avantage de ne pas nécessiter de base d'apprentissage très grosse, mais sont limitées lorsque les objets subissent des déformations. Les techniques basées sur du machine learning classique voient leur robustesse de la détection augmenter avec la taille de la base d'apprentissage et donc du temps nécessaire pour créer ces bases qui dans notre cas sont faites par les utilisateurs. Enfin nous avons aussi exploré les méthodes d'apprentissage profond mais la taille de la base d'apprentissage devient un aspect limitant de la méthode. Une fois notre méthode de segmentation appliquée sur les objets détectés, nous pouvons procéder à la mesure. Pour cette étape, nous présenterons une méthode simple pour replacer les mesures sur les objets de la base d'apprentissage grâce à la segmentation et la base créée par l'utilisateur.

Les contributions de ce manuscrit sont les suivantes. Tout d'abord, nous avons validé de manière expérimentale et automatique que le bruit présent dans les images de microscopie électronique à transmission d'objets du semi-conducteur est un bruit de Poisson. Nous avons ensuite mis en évidence de manière expérimentale que les différentes modalités d'un microscope TEM sont liées de façon linéaire ce qui nous permet de ne pas avoir à traiter toutes les modalités sans perte d'informations. Dans un second temps, nous proposons une méthode de segmentation guidée par un template. L'arbre utilisé pour cette représentation est une structure compacte et il est possible de l'enrichir avec autant d'information a priori que néces-

saire. Enfin, cette méthode de segmentation est incluse dans un contexte industriel de mesures d'objets. Dans ce pipeline, nous proposons une méthode de détection d'objets basé sur des réseau de neurones (NN). Pour finir, nous présentons une méthode pour améliorer le remplacement des mesures grâce à une base d'apprentissage provenant de la détection d'objets afin de proposer un pipeline automatique de mesures de ces objets.

1.4 Notations

$I : X \rightarrow \mathbb{R}$: une image
$x \in X$: les coordonnées des pixels dans le domaine de l'image
$T(V, C)$: un arbre de formes
V	: l'ensemble des nœuds d'un arbre de formes
C	: l'ensemble des connections d'un arbre de formes
μ_E	: la moyenne empirique d'un échantillon E
σ_E	: l'écart type empirique d'un échantillon E

1 Introduction et contexte industriel

2 Présentation des données

Ce chapitre sera dédié à l'étude de nos données. Nous étudions des images d'objets de l'industrie du semi-conducteur qui sont donc assez différents des objets courants utilisés dans les bases de données classiques pour le traitement d'image ou l'apprentissage automatique. À notre connaissance, il n'existe pas de base de données publique d'images pour le semi-conducteur, cette industrie repose majoritairement sur le secret industriel et même des images de technologies plus vieilles sont compliquées à obtenir. Nous présentons les microscopes utilisés ainsi que leurs spécificités. Pour finir, nous discuterons du bruit caractéristique de ces images ainsi que des méthodes de filtrage adaptées.

2.1 Microscopie électronique

2.1.1 Présentation des objets et leurs dimensions

Dans l'industrie du semi-conducteur, il y a trois grandes parties dans la chaîne de fabrication : le Front End Of Line (FEOL) et le Back End Of Line (BEOL) qui forment tous deux le Front End et le Back End (Figure 2.1). La partie FEOL gère les composants de base sur une puce et elle contient les transistors, les condensateurs et les résistances. La partie BEOL s'occupe quant à elle de la connexion entre les éléments précédents afin de réaliser les opérations logiques. Ces deux parties sont appelées ensemble Front End car les procédés mis en œuvre nécessitent d'être réalisés dans une salle blanche, un environnement très contrôlé vis à vis des particules en suspension dans l'air. Enfin, le Back End teste la puce et son

2 Présentation des données

encapsulation. Dans cette thèse, nous nous intéresserons principalement à la partie FEOL, dans laquelle le nombre d'étapes de fabrication a fortement augmenté avec la réduction en taille. Les ingénieurs ont redoublé d'ingéniosité et ont mis en place des procédés qui réduisent les dimensions des transistors au prix d'une multiplication des étapes. Dans la Figure 2.2, on voit les différentes étapes de fabrication d'un *design* de transistor simple. Cette figure ne montre pas les étapes de lithographie qui sont réalisées entre chaque gravure (etch sur le graphique) et qui permettent de faire une gravure sélective. Ces étapes sont au nombre d'une cinquantaine pour les transistors proches des 30 nanomètres et jusqu'à 300 à 400 étapes pour les nœuds technologiques les plus avancés (10-3 nm). Lors de chaque étape, des ingénieurs mesurent manuellement les structures afin de s'assurer que leurs dimensions correspondent bien au *design* de simulation pour atteindre les bonnes performances électriques.

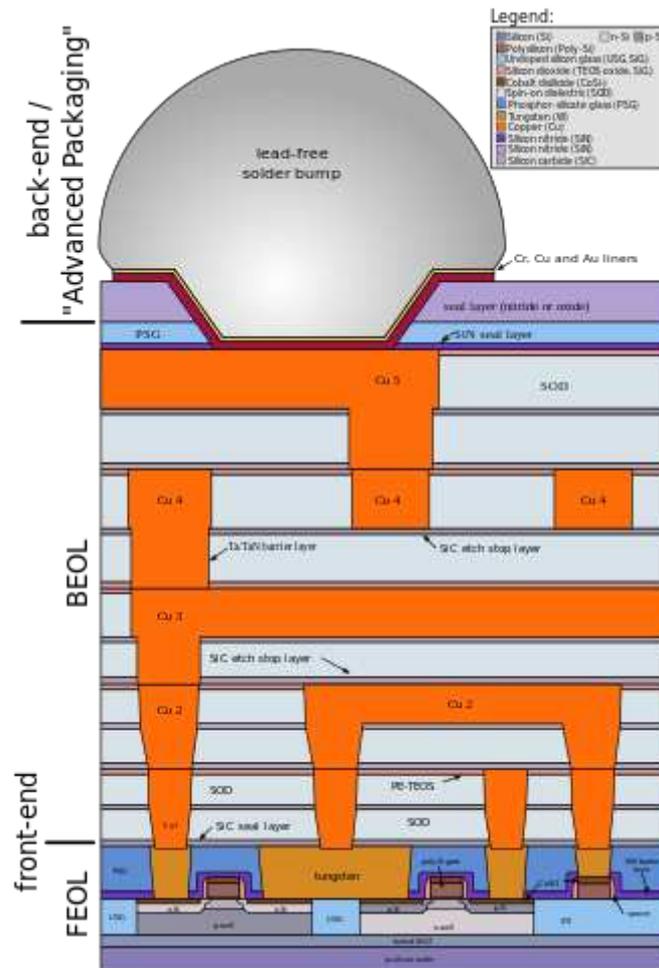


FIGURE 2.1 – Schéma de coupe d’une puce. Les images proviennent de [12].

2 Présentation des données

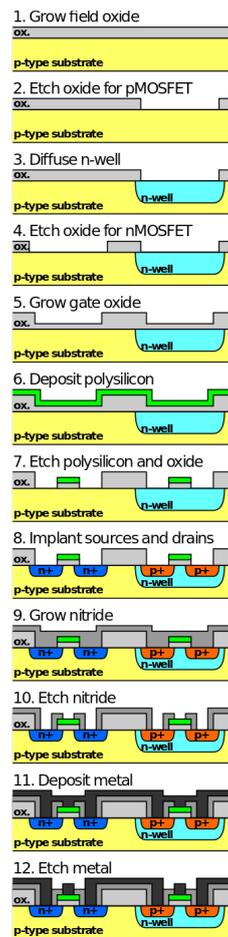


FIGURE 2.2 – Exemple des différentes étapes pour la fabrication de transistors. Les images proviennent de [12].

2.1.2 Microscopie pour les structures nanoscopiques

Dans cette thèse, nous nous intéresserons aux techniques pour imager des objets à une échelle nanométrique : Atomic Force Microscopy (AFM), Scanning Electron Microscopy (SEM) et TEM. Les autres méthodes non imagées regroupent entre autres la scattérométrie et les mesures de performances électriques.

Pour l'AFM, une pointe est déplacée en ligne le long de la surface de l'échantillon, ce qui permet de créer une carte topographique de l'échantillon. Cette technique non destructive est assez peu utilisée dans l'industrie du semi-conducteur car elle donne une carte de hauteur de l'échantillon en tout point. Si la résolution en hauteur est de l'ordre de l'angström, la résolution latérale n'est que de quelques dizaines de nanomètres, soit de l'ordre de grandeur des structures d'intérêt comme un transistor. Son mode de fonctionnement est expliqué plus en détails dans [13] ainsi que diverses applications en biologie par exemple. Cette technique peut aussi mesurer d'autres propriétés comme les propriétés nano-mécanique locale des matériaux. La faible résolution latérale reste un obstacle pour le semi-conducteur.

Le SEM est probablement la technique la plus utilisée pour la production de l'industrie du semi-conducteur. Cette technique non destructive permet d'obtenir des informations sur la topographie et sur les matériaux. Elle est utilisée à la fois sur des chaînes de production avec des microscopes capables d'imager des régions d'un wafer complet ou pendant les étapes de recherche et développement sur un échantillon plus réduit. Cependant, si la méthode permet d'obtenir plusieurs types de mesures telles que la hauteur ou la qualité des matériaux, elles sont davantage de l'ordre du qualitatif plus que du quantitatif. Un halo blanc renseigne qu'il y a un changement de hauteur mais on ne peut avoir ni sa quantité, ni son profil, ni sa position précise. L'incertitude sur la position varie grandement en fonction du type d'objets imagés de 1% jusqu'à 20% de la dimension mesurée. Le principe du microscope (Figure 2.3) consiste à envoyer des électrons avec un canon à électrons sur l'échantillon et récupérer les électrons secondaires ou

2 Présentation des données

rétro-diffusés. Pour les électrons secondaires, la quantité d'électrons produits dépend de la surface disponible sous le faisceau. Ainsi une surface perpendiculaire au faisceau produira moins d'électrons qu'une surface inclinée, ce qui donne l'information relative de topographie. Pour les électrons rétro-diffusés, la quantité dépend de la masse atomique de la matière présente sous le faisceau. On obtient donc une information de surface de l'échantillon. Il est aussi possible de faire de la microscopie SEM en vue de coupe (X-SEM), elle conserve les mêmes méthodes d'acquisition mais sur une coupe de l'échantillon. Cette technique se rapproche du TEM et devient destructive quand une coupe de l'échantillon est nécessaire.

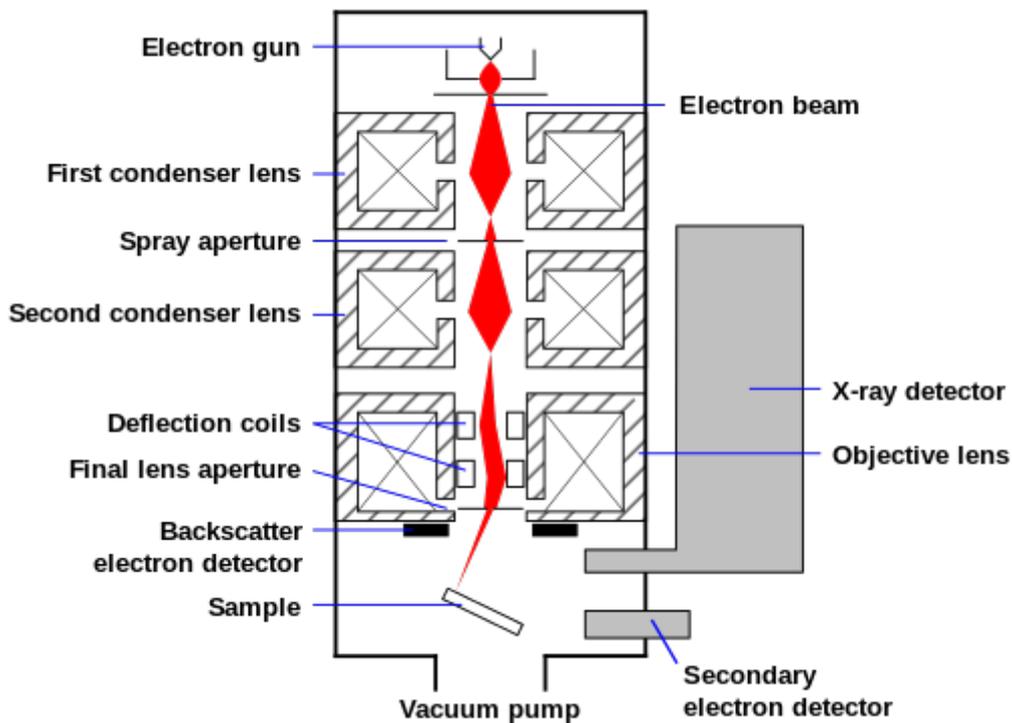


FIGURE 2.3 – Schéma d'un SEM. Les images proviennent de [14].

Enfin, le TEM fournit des informations précises pour ce qui est des matériaux et leur position. Le principe de ce microscope est similaire au précédent car on envoie toujours un faisceau d'électrons sur un échantillon. L'échantillon doit être une fine tranche de matériaux d'une épaisseur inférieure à quelques centaines de nanomètres. Les électrons passent à travers l'échantillon et sont récoltés du côté opposé au canon (Figure 2.6). L'in-

térêt de ce microscope est la résolution plus importante que le SEM. Cependant, ces microscopes sont plus coûteux que les SEM et leur utilisation nécessite des utilisateurs plus expérimentés. Elle est toujours destructive ce qui pose un problème pour faire de l'inspection sur la chaîne de production. Ces microscopes sont donc réservés aux étapes de recherche et développement. Il est important de mentionner aussi qu'il ne peut pas y avoir de vide dans les échantillons pour le TEM contrairement au SEM. De ce fait, certains matériaux présents sur les images sont utilisés pour du remplissage ou encore comme agent contrastant et ne sont donc pas présents dans le *design* de fabrication. Nous présenterons dans la suite les différents modes d'acquisition de ce microscope.

2.2 Modalités TEM

Cette section vise à présenter les modalités du TEM. Tout d'abord, nous étudierons le mode Imaging du Scanning Transmission Electron Microscopy (STEM). Nous explorerons ensuite les éventuelles relations entre ces modalités.

2.2.1 Présentation

La différence du STEM avec le TEM ou Conventional Transmission Electron Microscopy (CTEM) est que le faisceau est concentré en une fine pointe qui est balayée sur l'échantillon. La Figure 2.4 montre les deux fonctionnements. L'avantage principal est une résolution beaucoup plus élevée. Cependant, cela peut créer des artefacts sur les images lorsque le microscope est mal réglé et donc nécessite une plus grande expertise de la part de l'opérateur. Ensuite le mode Imaging est à mettre en opposition avec le mode Diffraction. Le mode Imaging vise à imager l'échantillon dans sa globalité pour récupérer une carte des matériaux (les modalités influenceront sur les informations mises en évidence). Le mode Diffraction se concentre sur une région précise de l'échantillon choisie par l'utilisateur

2 Présentation des données

afin d'observer les motifs cristallins de la zone (Figure 2.5). De plus, pour le Imaging STEM il existe plusieurs modalités qui mettent en évidence des propriétés différentes de l'échantillon. L'intérêt principal de ces modalités est qu'il est possible de les acquérir simultanément sur un échantillon dès lors que le microscope est équipé des bons capteurs (Figure 2.6). Dans notre cas, nous nous intéresserons au champ clair (Bright Field (BF)), au champ sombre (Dark Field (DF)) et au champ sombre annulaire aux grands angles (High Angle Annular Dark Field (HAADF)).

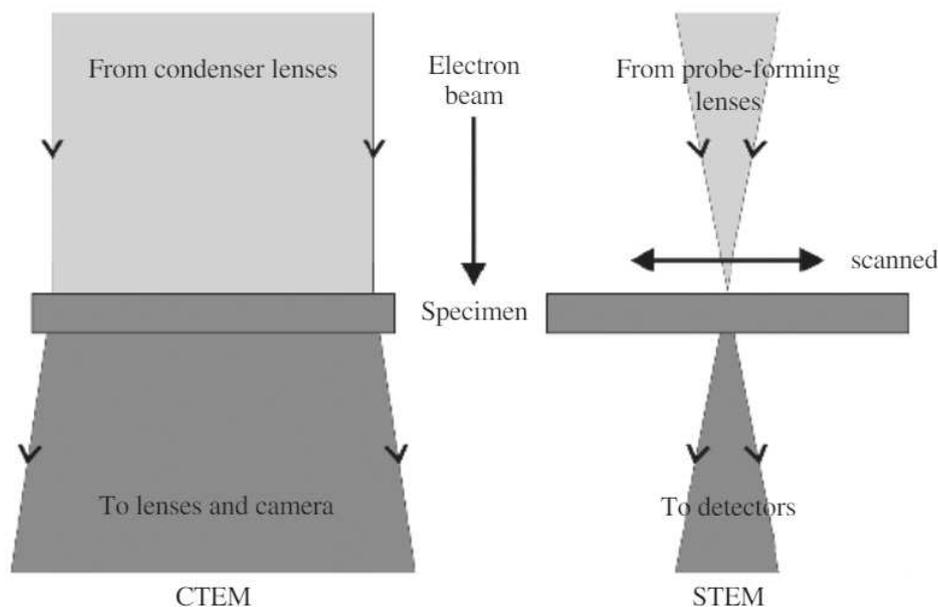


FIGURE 2.4 – Schéma des différences entre CTEM et STEM. Les images proviennent de [15].

Tout d'abord, le champ clair est l'un des plus courants parmi les images de notre base de données et principalement utilisé par les ingénieurs procédé. Cette modalité récolte les électrons qui passent directement à travers l'échantillon. Parmi les modalités, celle-ci est la plus lumineuse. Elle a l'avantage de produire des images qui contiennent beaucoup d'informations. En effet, on obtient une information de matériau car plus le numéro atomique est haut et moins les électrons peuvent passer à travers l'échantillon en ligne droite car ils sont plus facilement diffractés donc ces régions de matériaux sont plus sombres. Ensuite, on a aussi une information

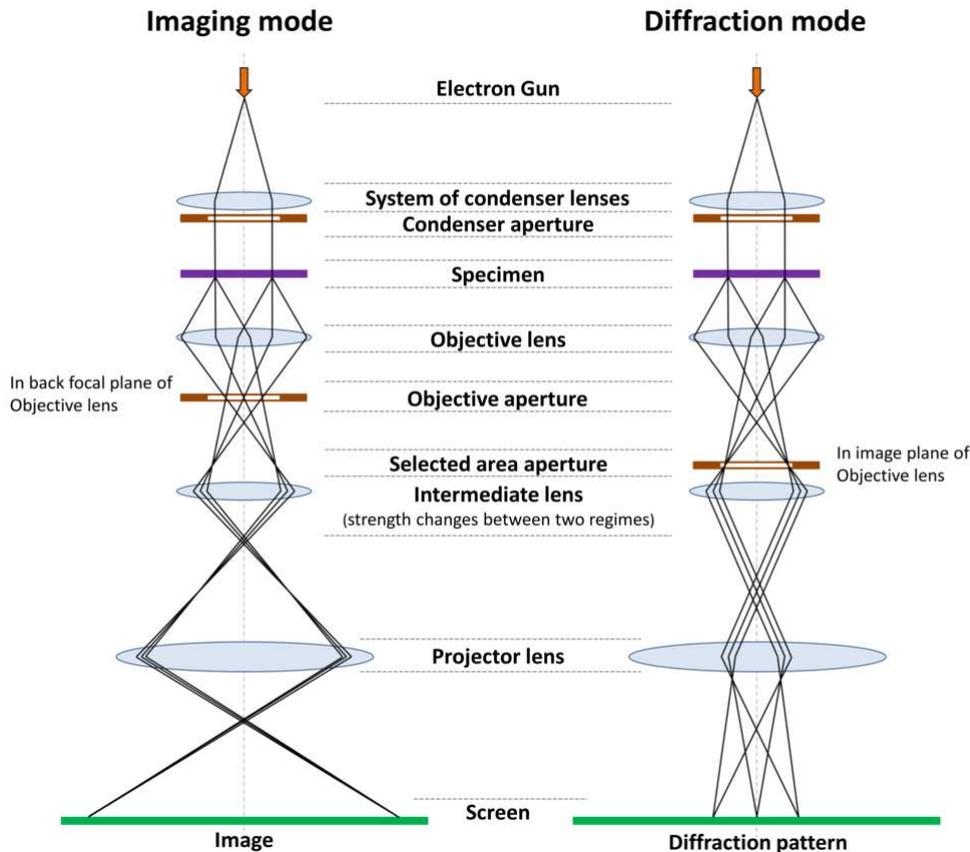
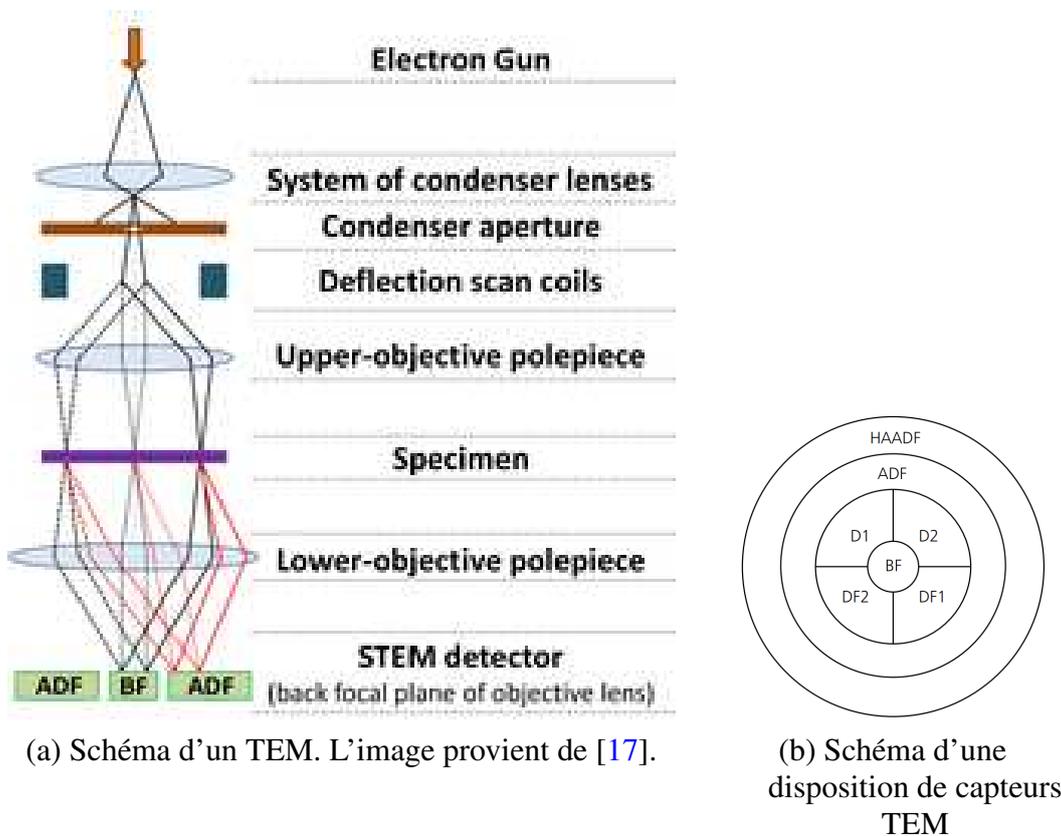


FIGURE 2.5 – Schéma des modes TEM. Les images proviennent de [16]

sur l'épaisseur de l'échantillon. Plus l'échantillon est épais, plus l'image est sombre. Cependant, pour ce point, les opérateurs pour l'industrie du semi-conducteur cherchent à préparer des échantillons qui ont une épaisseur homogène. Cette information est donc propice à créer des artefacts si l'échantillon n'est pas préparé correctement. Enfin, on obtient aussi de faibles informations sur l'orientation des zones cristallines. Comme ces informations proviennent de l'absence des électrons diffractés, le signal de cette information est plus faible que celle des matériaux ou de l'épaisseur.

Le champ sombre est la modalité la moins utilisée de notre base. Le champ sombre est découpé en plusieurs sections (Figure 2.6) selon les microscopes pour obtenir les différentes orientations cristallines. Cette modalité met en jeu les phénomènes de diffraction de Bragg afin d'avoir une

2 Présentation des données



(a) Schéma d'un TEM. L'image provient de [17].

(b) Schéma d'une disposition de capteurs TEM

FIGURE 2.6 – Illustration de l'acquisition des modalités TEM et des capteurs.

image plus ou moins lumineuse selon l'orientation de la maille cristalline locale de l'échantillon. Elle permet donc de voir si localement cette orientation varie ou si l'échantillon est amorphe. Dans le monde du semi-conducteur cette information est assez peu utilisée car les impacts de ces orientations sur les propriétés physiques des matériaux ne sont pas très bien connus. Cependant, elle peut être utile pour les étapes de dépôts de couche mince car dans ce cas, une mauvaise orientation ou une discontinuité d'orientation du matériau sur lequel est effectué le dépôt peut créer de la rugosité sur le dépôt et donc dégrader les performances de la puce. Ce genre de défaut se traduit par différents niveaux de gris dans un matériau homogène.

Enfin, la modalité HAADF est la plus utilisée par les métrologues car elle est considérée comme la plus fiable et précise pour effectuer des me-

sures sur les matériaux. Cette dernière récupère les électrons diffractés aux grands angles qui permet d'obtenir un effet appelé Z-contrast. Cet effet est expliqué par la diffraction de Rutherford. Les électrons transmis par un matériau avec un grand numéro atomique verront leur trajectoire plus fortement diffractée que dans un matériau avec un petit numéro atomique. Ainsi les zones à forte diffraction pourront être captées par le HAADF et apparaîtront comme claires et les autres seront foncées. Il existe une variante de cette modalité, le champ sombre annulaire (Annular Dark Field (ADF)) qui met en jeu les mêmes effets mais à des angles plus petits et donc qui concerne une gamme de numéros atomiques plus bas. Dans notre cas, comme nous imageons principalement du silicium et des métaux, la modalité HAADF est plus indiquée. Dans les deux modalités, le rendu est similaire si ce n'est que les matériaux seront mis en évidence différemment.

Enfin afin d'illustrer ces propos, deux exemples de différences entre les modalités sont disponibles dans les Figures 2.7 et 2.8. La première figure présente des images d'un alliage de titane et de nickel et montre le fait que uniquement certaines orientations cristallines sont visibles dans le champs sombre. La deuxième illustre un échantillon de biofilm encapsulé dans de la résine. On y observe la meilleure résolution du Scanning TEM (champs clair et champs sombre) en comparaison du TEM par une image de meilleure qualité. De plus, on constate que le champs sombre annulaire ressemble au négatif du champs clair, les informations de cristaux en moins. Dans ce cas, le matériau étant amorphe, il n'y a pas d'information de cristaux vraiment visible.

2 Présentation des données

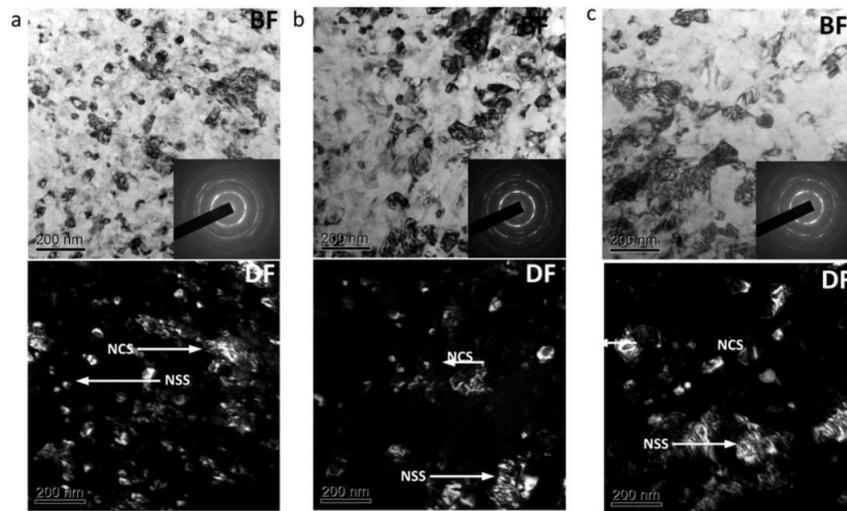


FIGURE 2.7 – Exemples de différences entre BF et DF. Les images proviennent de [18].

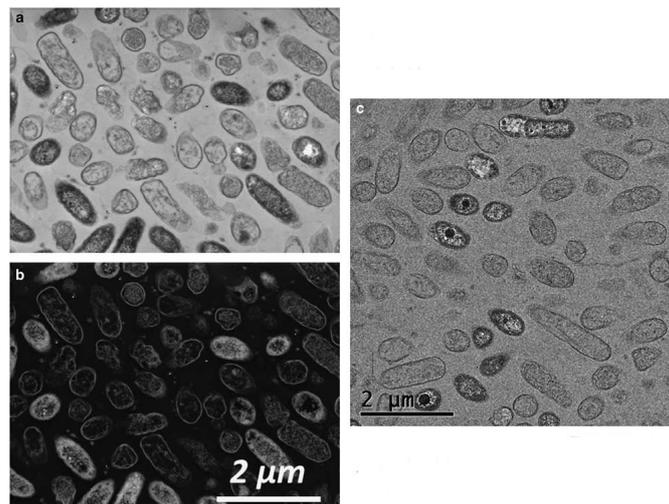


FIGURE 2.8 – Exemple de différence entre BF, ADF et CTEM. BF est en haut à gauche, ADF en bas à gauche et CTEM à droite. Les images proviennent de [19].

2.2.2 Exploration des modalités

Nous avons au démarrage de la thèse accès à un jeu de données contenant 109 triplets d'images multimodales acquises simultanément, composé du champs clair (BF), champs sombre (DF) et du champs sombre annulaire

aux grands angles (HAADF). Ce jeu provient d'un unique procédé de fabrication ce qui signifie que les objets sur les images sont semblables et le microscope d'acquisition est le même. Nous avons tout de même voulu explorer les données pour voir si les modalités sont liées. Pour des raisons de confidentialités, uniquement des portions grossies d'images pourront être montrées pour illustrer nos propos.

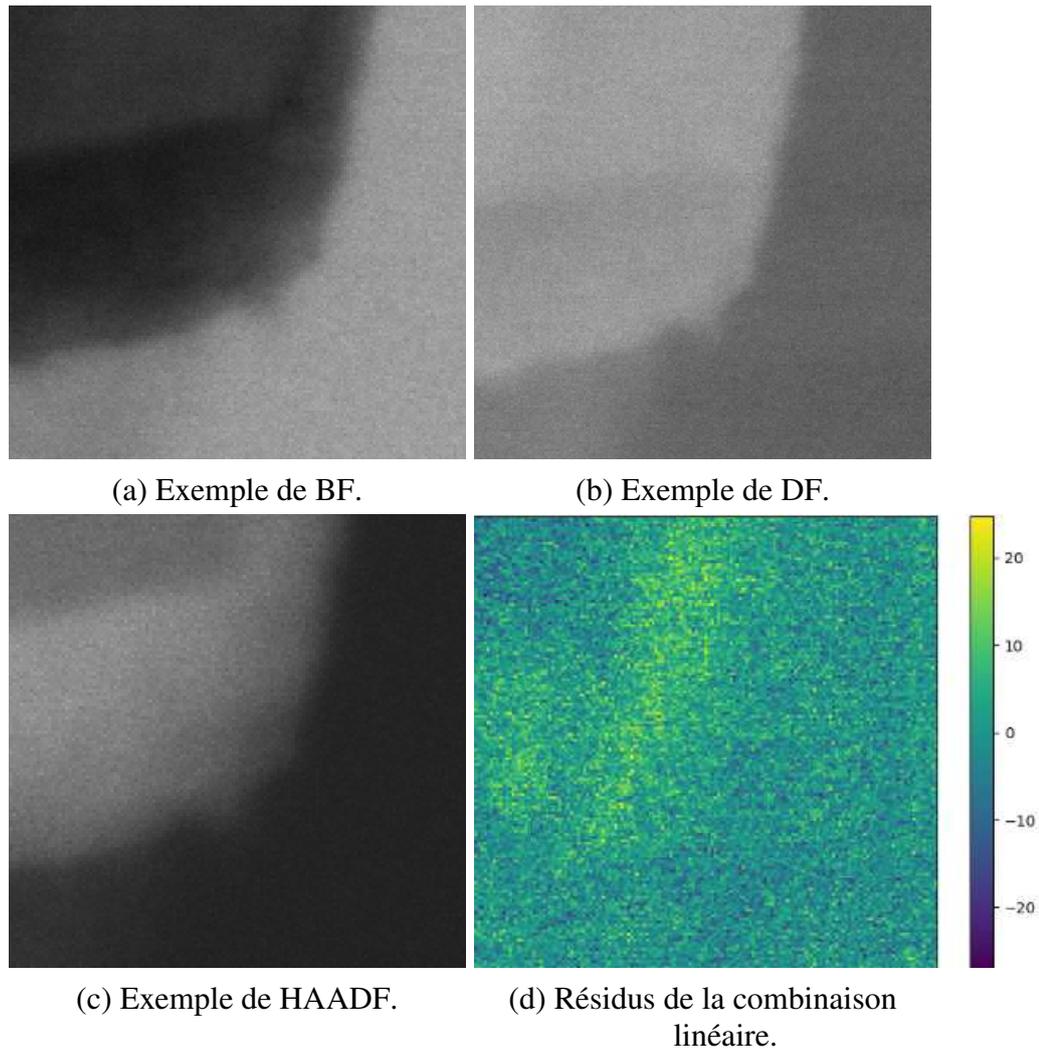


FIGURE 2.9 – Extrait d'un triplet TEM et des résidus de la combinaison linéaire.

Visuellement, il semble que les modalités sont liées par une relation linéaire (Figure 2.9). Nous proposons de poser un modèle linéaire (Équation 2.1) qui expliquerait la relation entre les modalités. Une analyse du rang

2 Présentation des données

de la matrice des modalités concaténées permettrait de répondre à cette hypothèse dans le cas idéal mais nos images étant bruitées, cette analyse ne nous permettra pas de conclure.

$$I_{BF}(x) = \alpha I_{DF}(x) + \beta I_{HAADF}(x) + \gamma, \forall x \in X \quad (2.1)$$

- I_{BF} : l'image acquise avec la modalité BF
- I_{DF} : l'image acquise avec la modalité DF
- I_{HAADF} : l'image acquise avec la modalité HAADF
- α, β, γ : les paramètres de la combinaison linéaire

Pour étudier cette hypothèse, nous avons mis en place deux méthodes. Tout d'abord une méthode qui valide que les trois modalités vérifient bien le modèle (Équation 2.1). La seconde nous permettra d'évaluer les paramètres des combinaisons afin de regarder si nous pouvons tirer quelques conclusions malgré la petite taille du jeu de données.

Le premier test est une analyse en composantes principales (ACP) du triplet. L'objectif de cet algorithme est de transformer des variables corrélées en de nouvelles variables décorrélées. Il est couramment utilisé pour réduire le nombre de variables en supprimant les informations redondantes. Dans notre cas, nous l'utilisons afin de mettre en évidence le fait que l'information d'une modalité est expliquée par les deux autres. Nous avons résumé le processus de l'ACP (Équations 2.2, 2.3, 2.4). En effectuant une ACP, nous nous attendons à constater que la valeur propre associée au troisième vecteur propre de la décomposition soit négligeable par rapport aux valeurs propres associées aux premier et deuxième vecteurs propres. La Figure 2.11 montre ce résultat sur notre jeu de données en affichant la valeur moyenne des valeurs propres de chaque triplet en échelle logarithmique avec leur barre d'erreur à 1'écart type.

Dans la deuxième approche, nous reprenons le modèle linéaire (Équation 2.1) et nous résolvons ce modèle par la méthode des moindres carrés qui minimise les résidus (Équation 2.5). Si les résidus ont une distribution gaussienne centrée en zéro alors notre hypothèse est valide et nous avons un jeu de paramètres décrivant la relation entre les modalités. Après avoir

Standardisation de chaque modalité :

$$\hat{I}_k(x) = \frac{I_k(x) - \mu_{I_k}}{\sigma_{I_k}}, \forall k \in \{BF, DF, HAADF\} \quad (2.2)$$

μ_I : la moyenne empirique de l'image I

σ_I : l'écart type empirique de l'image I

Calcul de la covariance :

$$C = \frac{1}{n-1} D^T D \quad (2.3)$$

D : la matrice des données avec les échantillons en ligne et les modalités en colonne

D^T : la transposée de D

n : le nombre d'échantillons

Calcul des valeurs et vecteurs propres :

$$V^{-1} C V = \nu \quad (2.4)$$

V : la matrice des vecteurs propres

V^{-1} : l'inverse de V

ν : la matrice diagonale contenant les valeurs propres en diagonale

FIGURE 2.10 – Description de l'ACP.

résolu le modèle, nous avons pu vérifier visuellement que la distribution des résidus est bien gaussienne sur un triplet d'images exemple (Figure 2.12). Comme la méthode précédente nous a déjà permis de valider l'hypothèse, nous n'avons pas vérifié quantitativement l'aspect gaussien de la distribution.

Nous avons exploré les méta-données des images à notre disposition afin de tenter de trouver une relation entre les paramètres d'acquisition et

2 Présentation des données

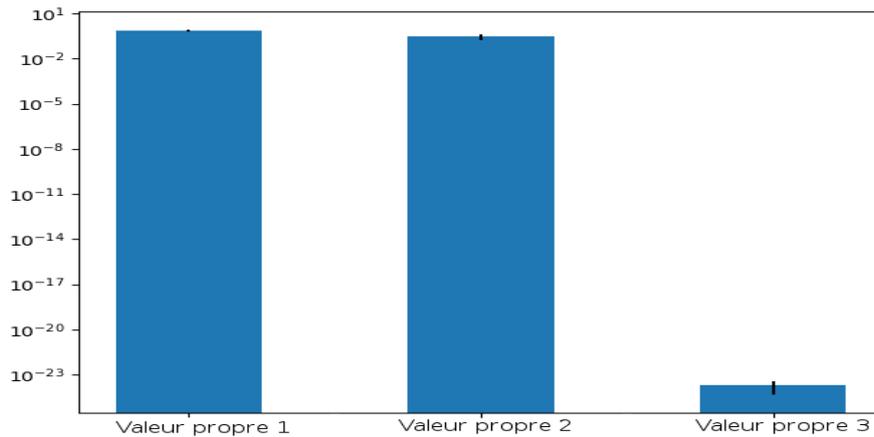


FIGURE 2.11 – Comparaison de la moyenne des valeurs propres de l’analyse en composantes principales en échelle logarithmique avec leur barre d’erreur à 1 écart type.

TABLE 2.1 – Comparaison de la moyenne des valeurs propres de l’analyse en composantes principales.

Composante	Valeur moyenne	Erreur (1 écart type)
1	7.15e-1	1.21e-1
2	2.84e-1	1.21e-1
3	2.06e-24	5.94e-24

$$S(\alpha, \beta, \gamma) = \sum_{x \in X} (I_{BF}(x) - f(x; \alpha, \beta, \gamma))^2 = \sum_{x \in X} r(x; \alpha, \beta, \gamma)^2 \quad (2.5)$$

$$f(x; \alpha, \beta, \gamma) : \alpha I_{DF}(x) + \beta I_{HAADF}(x) + \gamma$$

$r(x; \alpha, \beta, \gamma)$: les résidus du modèle

les paramètres de la combinaison linéaire mais nous n’avons pas repéré de patterns évidents (Figure 2.13). De plus la faible variabilité de notre jeu de données et le faible intérêt pour le projet de la thèse nous a poussé à ne pas approfondir la recherche, cela reste cependant, un sujet qui mériterait d’être approfondi pour le monde de la microscopie.

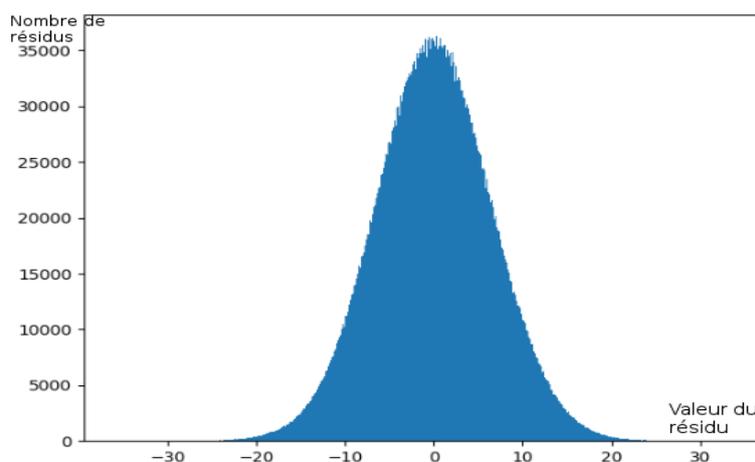


FIGURE 2.12 – Distribution des résidus sur un triplet TEM.

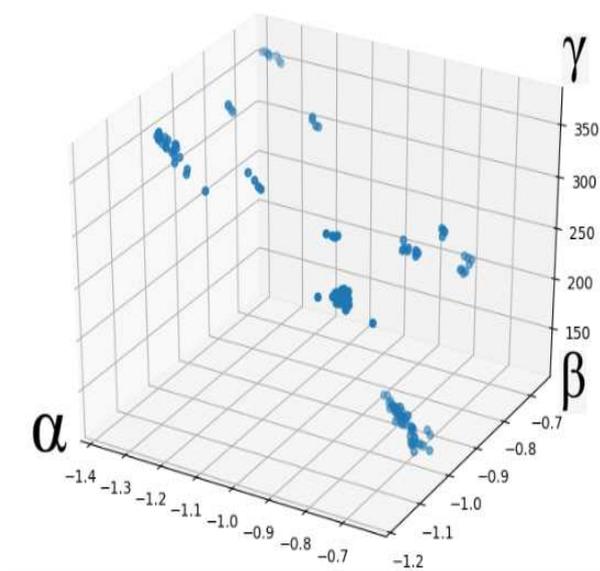


FIGURE 2.13 – Coefficients de la combinaison linéaire des triplets TEM.

À notre connaissance, la relation linéaire entre les modalités n'est pas mentionnée dans la littérature. De plus nous avons pris contact avec un opérateur TEM spécialisé dans l'analyse de matériaux cristallins qui n'avait pas connaissance de ce résultat. Physiquement, le résultat a du sens (Équation 2.6). Le faisceau transmis d'un TEM récupéré dans son intégralité produit une image blanche. Les modalités apparaissent en ne sélectionnant qu'une partie du signal transmis. Ainsi on peut s'attendre à ce que

2 Présentation des données

la somme des trois modalités soit une constante. Cependant, cette analyse concerne un unique microscope ainsi qu'un type d'échantillon, il faudrait davantage de données pour valider ce résultat. Dans le cas d'analyse d'images multimodales, ce résultat permet d'utiliser seulement deux modalités pour rassembler toutes les informations à la place de trois.

$$E_{\text{canon}} = \sum_{\text{capteur} \in \{BF, DF, ADF, HAADF\}} E_{\text{capteur}} \quad (2.6)$$

E_{canon} : représente l'énergie en sortie du canon d'électrons

E_{capteur} : représente l'énergie reçue par les différents capteurs

2.3 Bruit dans les images TEM

2.3.1 État de l'art

Dans l'état de l'art sur les images TEM, il est décrit deux modèles de bruit différents : le modèle de Poisson et un mélange de modèle de Poisson et gaussien. Après inspection, le modèle de Poisson semble plus proche de notre cas, nous étudierons cette hypothèse dans la suite. Nous avons effectué un état de l'art sur le filtrage de bruit de Poisson afin d'améliorer la qualité des images. Pour cet état de l'art, nous nous intéresserons au filtrage de bruit de Poisson tel que proposé dans le traitement d'images et non spécifiquement au TEM comme présenté précédemment. Globalement, nous avons identifié trois grandes familles de filtres : les méthodes basées sur une transformée de stabilisation de variance et un filtrage de bruit gaussien, les méthodes basées sur les ondelettes et enfin les méthodes basées sur des combinaisons de statistiques non locales.

La transformée d'Anscombe est une transformée de stabilisation de variance couramment utilisée que nous présenterons ultérieurement via une application. Elle permet de changer le bruit de Poisson de l'image en bruit

gaussien qui est plus simple à traiter. Les approches l'utilisant reposent donc sur deux éléments. Tout d'abord un filtre gaussien et deuxièmement la transformée inverse pour revenir au domaine d'origine de l'image. La transformée inverse permet de revenir à la gamme de valeurs de l'image d'origine. La méthode proposée dans [20] permet d'obtenir une transformée optimale en terme de maximum de vraisemblance.

Pour les méthodes basées sur les ondelettes, nous avons trouvé deux propositions fréquemment citées dans les comparaisons. La première, la plus récente [21], nommée Pure-Let, propose une combinaison linéaire de filtres élémentaires dont les poids sont calculés en minimisant l'estimation non biaisée de risque Poisson. L'avantage réside dans l'absence de paramètres d'entrée rendant son utilisation simple. La seconde, [22], combine les deux approches en proposant à la fois une transformée de stabilisation de variance ainsi qu'un filtrage par ondelettes pour traiter le bruit gaussien.

Parmi les méthodes non locales, l'une des plus simples et courantes est le filtre de moyenne non locale tel que proposé dans [23]. Une proposition d'amélioration de ce dernier a été faite dans [24] et suggère de faire une sélection automatique des paramètres pour minimiser l'erreur moyenne quadratique du filtrage par rapport à une image sans bruit. L'image non bruitée proviendra d'une étape de filtrage précédente et de manière itérative on converge vers un minimum.

2.3.2 Hypothèse de bruit de Poisson et validation

En connaissant le moyen d'acquisition de nos images, nous pouvons étudier le bruit et les artefacts que l'on peut obtenir sur ces images afin de les traiter au mieux. De part la construction d'un TEM, il y a deux types de bruit qui peuvent apparaître sur nos images. Le fait de compter des électrons produit du bruit de Poisson sur les images TEM [25] et [26] et un bruit gaussien est introduit par l'électronique et les lentilles électromagnétiques utilisées pour focaliser le faisceau d'électrons. On peut de manière générale décrire le bruit de ces images avec l'équation 2.8 comme décrit

2 Présentation des données

dans [27]. Dans notre cas, le bruit de Poisson est dominant. Nous nous attellerons à vérifier cette hypothèse.

Le bruit de Poisson aussi appelé bruit de grenaille est un bruit typiquement observé lors de comptage. Il est modélisé par une loi de Poisson car elle décrit les probabilités d'observer un nombre donné d'événements dans un intervalle de temps fixe (Équation 2.7). Dans ce cadre, la variance du bruit dépend du nombre moyen d'électrons qui arrivent jusqu'au capteur c'est-à-dire la valeur moyenne de l'image. Les observations dans les images suivant une distribution de Poisson (Équation 2.9), elles s'approchent localement d'une gaussienne dans les zones homogènes.

$$P(k, \lambda) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (2.7)$$

P : fonction de densité de masse

k : nombre d'événements

λ : intervalle de temps fixé

$$I(x) = S(x) + \eta_p(S(x)) + \eta_g(x) \quad (2.8)$$

$I(x)$: l'image observée

$S(x)$: le signal original

η_p : la composante Poissonienne dépendante du signal

η_g : la composante gaussienne indépendante

$$I(x) \sim P(E(x)), \forall x \in X \quad (2.9)$$

$I(x)$: l'image observée

P : distribution de Poisson

$E(x)$: les électrons reçus par le capteur

Pour valider cette hypothèse de manière expérimentale, nous avons mis au point une méthode adaptée à nos images qui sont modélisées comme constantes par morceaux. Cette proposition s'inspire des travaux de [28]

2.3 Bruit dans les images TEM

à la grande différence que l'on ne cherche pas à quantifier le bruit et que notre modèle de bruit est fixé. La première étape consiste à détecter les zones homogènes. Ensuite nous pouvons calculer les valeurs de moyenne et variance locales de ces zones. On cherchera enfin une corrélation entre ces deux grandeurs pour valider l'hypothèse du bruit de Poisson.

Pour cela, nous avons utilisé toutes les images provenant de notre base interne. Les différentes modalités sont couvertes mais il n'y a pas une répartition égale entre toutes les modalités. Nous avons à disposition 1362 images réparties en 44.9% de BF, 15.6% de DF et 39.5% de HAADF. Cependant, plusieurs artefacts peuvent altérer notre analyse (Figure 2.14). Le premier artefact provient de l'étape d'acquisition. En effet, afin d'augmenter le contraste entre deux matériaux très similaires, certains opérateurs de TEM augmentent la luminosité des images, ce qui introduit de la saturation. Nous avons donc retiré ces images de l'analyse. Le second problème provient de la qualité d'enregistrement des images. Les entreprises du semi-conducteurs sont amenées à les enregistrer au format JPEG afin de pouvoir les conserver dans leur archive. La compression JPEG modifie fortement les images. Enfin, dans les images TEM en très haute résolution, le maillage atomique des matériaux est visible. Cela se traduit par des petits cercles organisés de manière régulière. Dans ce cas, il n'y a pas de zones homogènes dans lesquelles faire notre analyse.

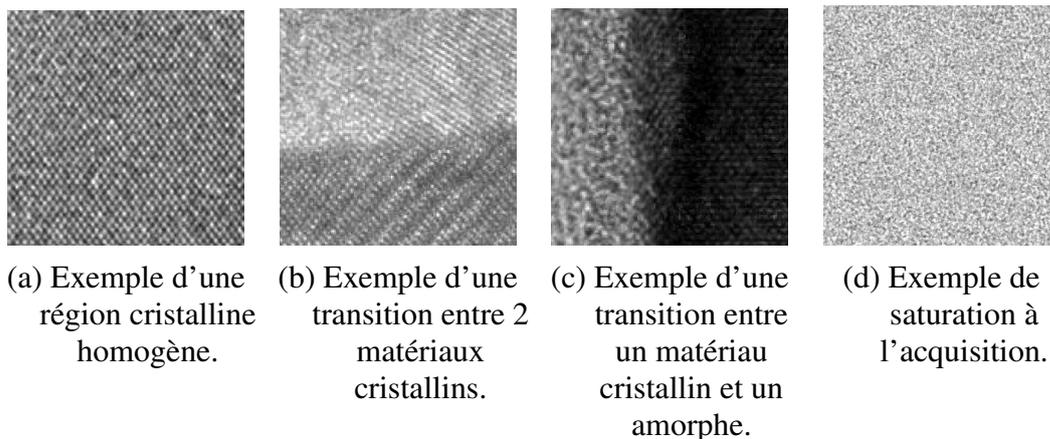


FIGURE 2.14 – Exemples d'artefacts empêchant l'analyse automatique.

2 Présentation des données

Pour détecter les régions homogènes nous avons appliqué un filtre qui consiste à retirer de l'analyse les régions qui ne sont pas localement gaussiennes par un test de normalité de D'Agostino et Pearson proposé dans [29]. Nous avons utilisé ce test pour sa simplicité et sa versatilité. Il vise à tester la symétrie et la forme de la distribution (Équation 2.10). Ces zones correspondent à des changements de matériaux flous ou à des matériaux cristallins ayant une texture.

$$K^2 = Z_1(g_1)^2 + Z_2(g_2)^2 \quad (2.10)$$

K^2 : mesure les déviations de la distribution observée avec une distribution normale
 g_1 : l'asymétrie de l'échantillon
 g_2 : le kurtosis de l'échantillon
 Z_1, Z_2 : les fonctions de transformation proposées par la méthode [29]

Pour le calcul des moyennes et variances locales et l'étape de filtrage précédente, nous devons choisir une taille de fenêtre. Elle influe sur deux facteurs. Tout d'abord, la fiabilité des calculs de moyenne et variance locales nécessitent d'avoir le voisinage le plus grand possible donc une taille de fenêtre grande. Cependant, pour constater la corrélation entre moyenne et variance, il nous faut le plus grand nombre de couples moyenne et variance. Or une grande taille de fenêtre augmente la zone filtrée par l'étape précédente et donc réduit le nombre de couples. Pour avoir un bon compromis entre les deux, nous avons sélectionné une taille de fenêtre de 13. Nous avons ensuite mesuré la corrélation entre la variance locale et la moyenne locale à l'aide du coefficient de corrélation linéaire de Pearson (Équation 2.11). L'objectif est d'obtenir un indicateur chiffré afin de valider notre hypothèse.

Notre analyse a montré que toutes les images de notre base affichent une corrélation d'au moins 0.65 entre la moyenne locale et la variance locale. Nous récapitulons les corrélations de chaque mode (Figure 2.15). On peut y voir que la majorité de nos images ont même une corrélation supérieure à 0.8. L'intérêt principal de cette approche que nous avons développée est qu'elle permet d'obtenir un score pour l'hypothèse de manière automa-

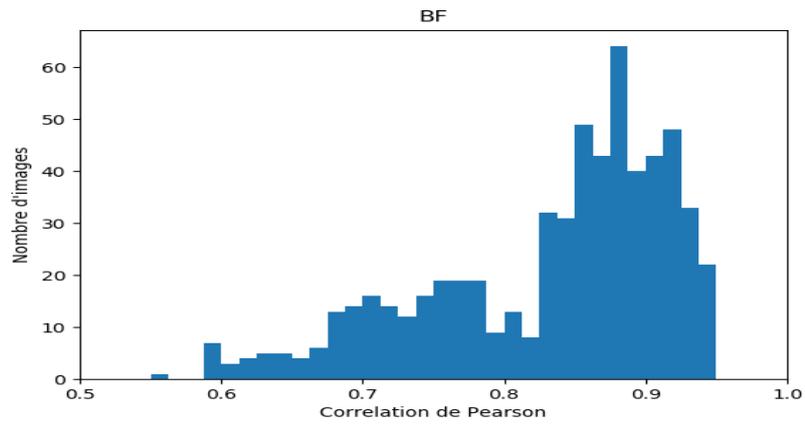
2.3 Bruit dans les images TEM

$$r = \frac{Cov(moy, var)}{\sigma_{moy}\sigma_{var}} \quad (2.11)$$

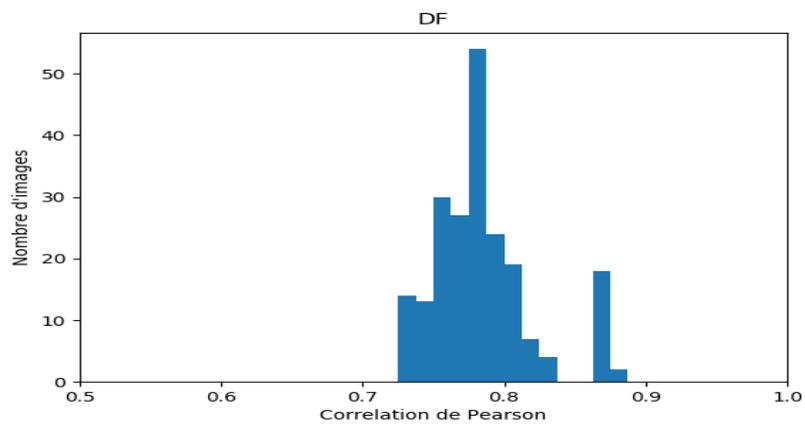
r	: le coefficient de corrélation de Pearson
Cov	: la covariance de 2 vecteurs
moy	: le vecteur de moyenne locale
$\forall x \in X; moy_x = \frac{\sum_{i \in N(x)} I(i)}{ N(x) }$: $N(x)$ étant le voisinage de x et $ N(x) $ le nombre de pixels de ce voisinage
var	: le vecteur de variance locale
$\forall x \in X; var_x = \frac{\sum_{i \in N(x)} (I(i) - moy_x)^2}{ N(x) }$: suivant les mêmes notations que moy_x

tique. Cependant, elle nécessite d'avoir des images avec des régions homogènes différentes suffisamment étendues afin d'avoir une quantité de points d'expérience statistiquement pertinente. Elle est donc plutôt bien adaptée aux images de semi-conducteurs qui contiennent plusieurs zones de matériaux. Cependant, pour d'autres types d'images TEM de composés éparses (Figures 2.7, 2.8), notre méthode n'est pas adaptée car la seule zone homogène et suffisamment étendue est le fond.

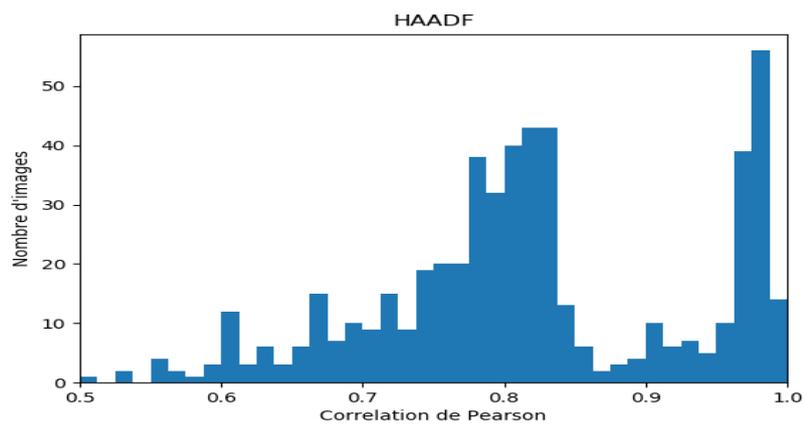
2 Présentation des données



(a) Distribution des scores de corrélation de notre base d'images BF.



(b) Distribution des scores de corrélation de notre base d'images DF.



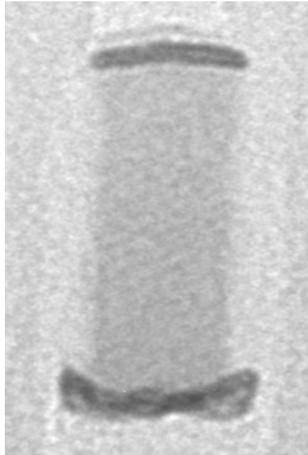
(c) Distribution des scores de corrélation de notre base d'images HAADF.

FIGURE 2.15 – Corrélation à notre hypothèse de bruit de Poisson en fonction des modalités.

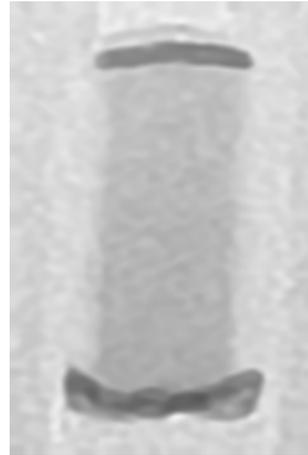
2.3.3 Choix pour notre application

Dans nos expériences, nous avons testé l'approche Pure-Let grâce à une implémentation dans ImageJ de [30], la transformée d'Anscombe proposée dans [20] que nous avons implémentée avec un filtre BM3D [31] ainsi que le filtre de moyenne non locale implémentée dans [32]. N'ayant pas d'image de référence non bruitée, nous n'avons fait qu'une analyse qualitative des résultats des filtres (Figure 2.16). À la vue de nos images qui n'ont pas un niveau de bruit très élevé mais davantage des contours flous et une texture particulière en haute résolution, les trois filtres ont des performances similaires. Pour la suite, nous avons choisi d'utiliser le filtre de moyenne non locale car il est le plus facilement implémentable dans la solution logicielle de Pollen Metrology.

2 Présentation des données



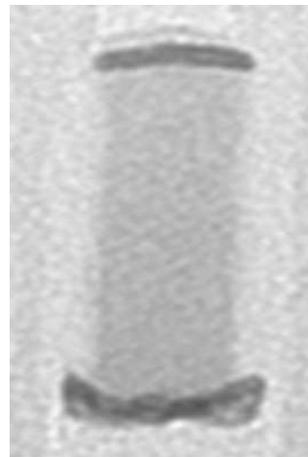
(a) Exemple d'objet par TEM.



(b) Filtrage gaussien avec la transformée d'Anscombe [20].



(c) Filtrage par moyenne non local [23].



(d) Filtrage par Pure-Let [21].

FIGURE 2.16 – Exemple d'application de filtres de l'état de l'art.

2.4 Conclusion

Dans ce chapitre, nous avons mis en évidence plusieurs éléments sur nos données. Le premier point notable est le type d'acquisition, le TEM, qui nous permet de mettre en place un modèle d'image constant par morceaux en lien avec les matériaux de l'échantillon physique. Ce modèle nous permettra de choisir une structure de données adaptée par la suite ainsi que de créer des données de tests synthétiques pour la validation de notre méthode.

2.4 Conclusion

Le deuxième point est l'analyse des différentes modalités du TEM qui peuvent être exprimées sous la forme d'une combinaison linéaire. Cette relation nous permet d'affiner l'information nécessaire dans le cadre de l'analyse multi-modale. Cette conclusion ne sera pas utilisée dans la thèse mais permet d'ouvrir des perspectives futures.

Enfin, nous avons validé l'hypothèse de bruit de Poisson sur nos images afin de pouvoir adopter une méthode de filtrage qui soit adaptée. Suite à un état de l'art, nous avons sélectionné une méthode appropriée à notre situation pour filtrer les images.

2 Présentation des données

3 Segmentation d'images basée sur les matériaux

3.1 Problématique

Pour effectuer des mesures sur nos objets et de par la méthode d'acquisition, nous avons décidé de nous tourner vers de la segmentation. Les mesures peuvent être très variées et nécessitent d'être explicitées par un utilisateur via une interface graphique, il est donc plus pratique de leur proposer des contours ou une segmentation comme support. Sur les images TEM, les matériaux sont identifiables par des niveaux de gris différents. Le but est de segmenter les différents matériaux dans l'objet. Dans cette section, un matériau du template sera défini à la fois par un niveau de gris et une position. Un template peut avoir deux régions distinctes avec des matériaux physiques identiques (du Silicium par exemple) que nous considérerons comme deux matériaux différents du template.

Dans notre contexte, les algorithmes sont initialisés par des utilisateurs, il conviendra donc que les initialisations doivent être simples et explicites. De plus, plusieurs utilisateurs peuvent être amenés à travailler sur les mêmes objets et donc souhaitent obtenir les mêmes résultats. Pour cela nous avons choisi d'utiliser le *design* qui est un élément que tous les utilisateurs partagent et qui définit l'objet en termes synthétiques.

Le *design* reprend les différents matériaux que nous considérerons comme les labels définis par l'utilisateur. Un exemple de *design* en fausses couleurs et la segmentation d'un objet sont affichés en Figure 3.1 et 3.2. Le *design*

3 Segmentation d'images basée sur les matériaux

peut être enrichi par l'utilisateur de plusieurs façons. Tout d'abord, le niveau de gris moyen attendu dans les images peut être renseigné comme les prises de vue sont faites dans des conditions similaires. De plus la segmentation résultante doit respecter les contraintes imposées par le template. Ces contraintes sont le nombre de composantes connectées, la hiérarchie des matériaux ainsi que la géométrie de ces segmentations. Dans les faits, le respect des deux premières contraintes permet d'obtenir la troisième.

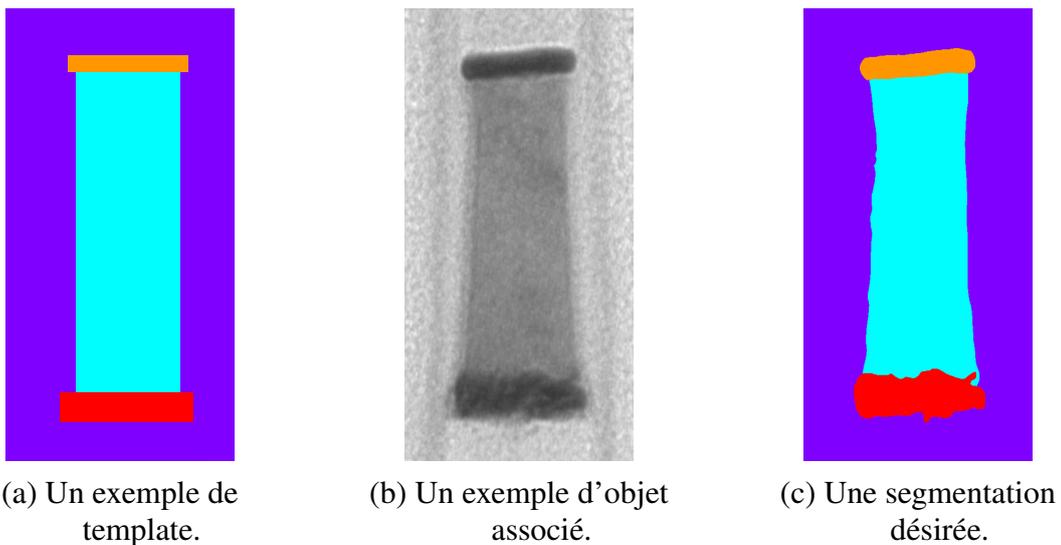


FIGURE 3.1 – Un exemple de segmentation guidée par un template.

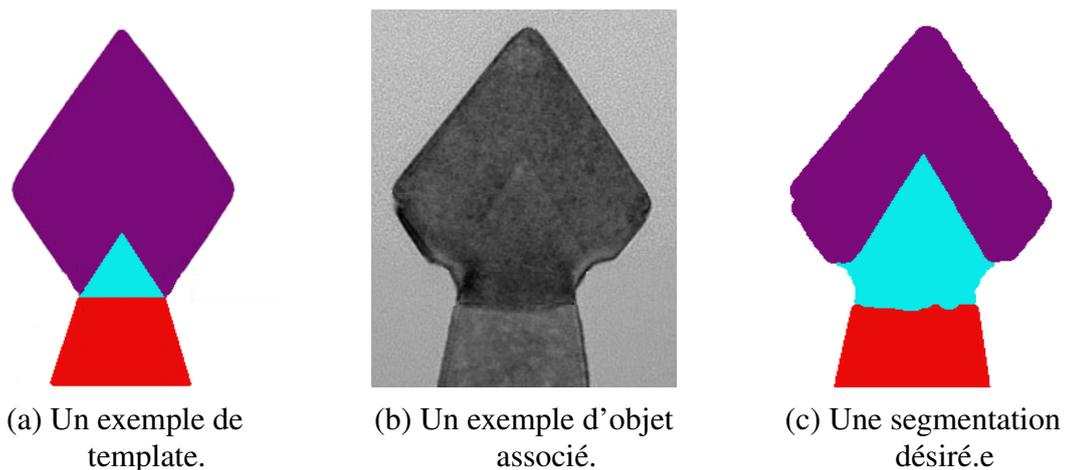


FIGURE 3.2 – Un deuxième exemple de segmentation guidée par un template.

Dans la littérature, il est proposé de nombreuses approches pour effectuer la segmentation mais à notre connaissance, aucune d'entre elles n'utilise un template ou *design* comme décrit précédemment. Dans l'univers de la microscopie TEM, les principales propositions se regroupent autour de trois grandes familles de méthodes. La première s'appuie sur des techniques basées sur des algorithmes de ligne de partage des eaux (watershed) tel que décrit dans [33]. D'autres approches comme [34] et [35] sont plus tournées vers des approches de Graph Cut. Enfin, nous avons aussi des techniques utilisant du seuillage comme [36].

On peut cependant approcher le problème avec des méthodes de template matching où le modèle, le template ou une partie d'image est cherché dans une image. Les méthodes font la comparaison pixel-à-pixel et utilisent des fonctions de coût telles que la somme des différences au carré, la somme des différences absolues ou encore la corrélation croisée normalisée comme décrit dans [37]. Ces métriques sont restreintes à la translation. Pour être robuste à la rotation, aux transformations affines, à l'occultation et aux changements d'illumination, d'autres métriques entre le template et les fenêtres candidates ont été proposées : Hamming [38] ou Hausdorff [39], M-estimateurs [40], corrélation asymétrique [41].

D'autres méthodes utilisent des points d'intérêt pour la comparaison plutôt que les pixels seuls. Des méthodes comme SIFT [42], SURF [43] ou ORB [44] sont souvent utilisées pour l'extraction des points d'intérêt avant d'exécuter la mise en correspondance avec des méthodes comme Ransac. Ces outils sont utilisés pour la reconstruction d'images panoramiques [45] et la recherche dans des bases d'images [46]. Des descripteurs globaux comme les histogrammes de descripteurs peuvent gérer des petites déformations et sont souvent utilisés pour de la recherche d'images [47] ou pour du suivi visuel [48]. Notre problématique vis-à-vis des points d'intérêts est le fait que les caractéristiques des points extraits sur le template sont différentes de ceux des points de l'image ce qui rend la mise en correspondance très compliquée.

Des versions plus robustes ont été récemment proposées dans [49] ou

3 Segmentation d'images basée sur les matériaux

encore [50]. La similarité entre le template et les fenêtres candidates est calculée en comptant les correspondances entre des paires de points qui sont mutuellement plus proches voisins dans le patch. Mais ces méthodes reposent sur le fait que le template est visuellement similaire aux images. Notre template est une version très simplifiée il est donc difficile d'appliquer ces méthodes.

D'autres approches sont proposées dans l'imagerie médicale avec la mise en correspondance d'atlas pour transposer la segmentation d'une base d'apprentissage à une nouvelle image comme décrit dans [51] et [52]. Ces approches sont principalement basées sur du recalage d'images en utilisant une segmentation comme guide. Dans [53] plusieurs de ces méthodes sont passées en revue. Une autre proposition [54] utilise des points d'intérêt pour faire le recalage. La problématique principale de ces techniques pour nous est le caractère idéal et simplifié de notre template.

Enfin, ce problème pourrait être résolu avec des approches basées sur des lignes de niveaux pour la segmentation en utilisant le template comme initialisation avec le niveau de gris du template comme valeur attendue pour la région comme proposé dans [55] et [56]. Ces techniques seront utilisées comme comparatif dans la section Résultats 3.4.

3.2 Arbre de formes

3.2.1 Présentation

De part l'aspect visuellement simple de notre template, nous avons recherché une structure de données qui nous permettrait de représenter efficacement le template afin de servir de support à la segmentation d'images. Nos templates sont composés de différents matériaux dont la nature et les caractéristiques peuvent être renseignées. La géométrie et la hiérarchie sont des éléments importants. Un graphe des régions adjacentes utilisant les nœuds pour les régions et les liens pour le caractère adjacent aurait pu

être utilisé comme représentation. Cependant, sur un graphe des régions adjacentes, la notion de régions est très différente de la notion de formes. Dans notre cas, une région sera dans la majorité du temps limitée à un pixel, ce qui produirait un graphe d'une très grande taille et contenant une information moins dense. Pour ce point, nous avons donc choisi l'Arbre de Formes (AdF) qui regroupe toutes les caractéristiques évoquées. Les travaux de [57] nous ont guidé sur cette structure en démontrant l'utilisation de l'AdF à des fins de segmentation. La géométrie et la hiérarchie sont deux informations fortes portées par le template car les objets réels fabriqués respectent ces contraintes contrairement aux aspects dimensionnels qui peuvent beaucoup varier.

L'AdF est une structure de données sous forme d'arbre qui regroupe les régions de niveaux de gris égaux et les organise en fonction de leur niveau d'inclusion dans les autres régions. Elle a l'avantage de pouvoir représenter des données multivariées [58] ou en n-D [59]. Cette approche se base sur des outils de morphologie mathématique pour la construction de l'arbre ce qui permet d'avoir une invariance au changement d'illumination. Cette propriété est intéressante dans notre cas car selon l'opérateur lors de l'acquisition, l'illumination globale de l'image peut varier.

Pour présenter la construction de l'arbre, nous considérerons une image en niveau de gris $I : X \rightarrow \mathbb{R}$ définie sur un espace à deux dimensions X et prenant ses valeurs dans un espace à une dimension \mathbb{R} ordonné avec la relation d'ordre \leq . Nous pouvons définir un opérateur H qui remplit les trous d'un ensemble de I (Équation 3.1). Une ligne de niveau de I est un vecteur de points ayant le même niveau de gris (Équation 3.2). Les formes sont définies par l'union des saturations des lignes de niveau supérieures et inférieures (Équation 3.3). Enfin, en considérant la relation d'inclusion \subseteq , la couverture (S, \subseteq) forme l'AdF T . Dans la suite, nous noterons $T = (V, C)$ avec V correspondant aux différentes formes $S(I)$ ordonnées par \subseteq et C correspondant aux différentes relations d'inclusion entre chaque élément de V .

Afin de motiver l'utilisation de l'AdF, nous avons tout d'abord construit

3 Segmentation d'images basée sur les matériaux

$$H(E) = \Omega \setminus cc(\overline{E}, \delta\Omega) \quad (3.1)$$

E : un ensemble de I
 \overline{E} : l'ensemble complémentaire de E
 $\delta\Omega$: le contour de Ω
 $cc(\overline{E}, \delta\Omega)$: les composantes connectées de \overline{E} connectée à $\delta\Omega$

$$\{x|I(x) = \lambda, \lambda \in \mathbb{R}\} \quad (3.2)$$

I : une image
 λ : une valeur de I

$$S(I) = \{H(X)|X \in \Gamma_\lambda^+\} \cup \{H(X)|X \in \Gamma_\lambda^-\} \quad (3.3)$$

Γ_λ^+ : une ligne de niveaux supérieure
 Γ_λ^- : une ligne de niveaux inférieure

et affiché plusieurs arbres afin de comprendre comment se comporte cette structure en présence de bruit et de filtres passe-bas en comparaison avec une image idéale. Un premier exemple simple avec un objet uniforme sur un fond uni est affiché en Figure 3.3. Nous avons appliqué un flou gaussien puis un bruit de Poisson sur l'image idéale pour comparer les AdF. Nous constatons sur cet exemple simple que la disposition des nœuds de l'arbre du template est similaire avec les nœuds de fond parents des nœuds de l'objet. Cependant, le bruit et le flou peuvent induire quelques incursions de nœuds objet dans les nœuds de fond à la frontière. Suite à cette vérification, nous avons poursuivi sur des exemples un peu plus proches de notre réalité.

Dans ce deuxième exemple (Figure 3.4), nous avons pris un objet réel, nous avons créé un template correspondant et nous avons généré une version simulée de la même façon que précédemment. L'objectif est de vérifier l'agencement des nœuds. Nous constatons les mêmes conclusions que précédemment. Nous pouvons noter la différence des agencements des zones

3.2 Arbre de formes

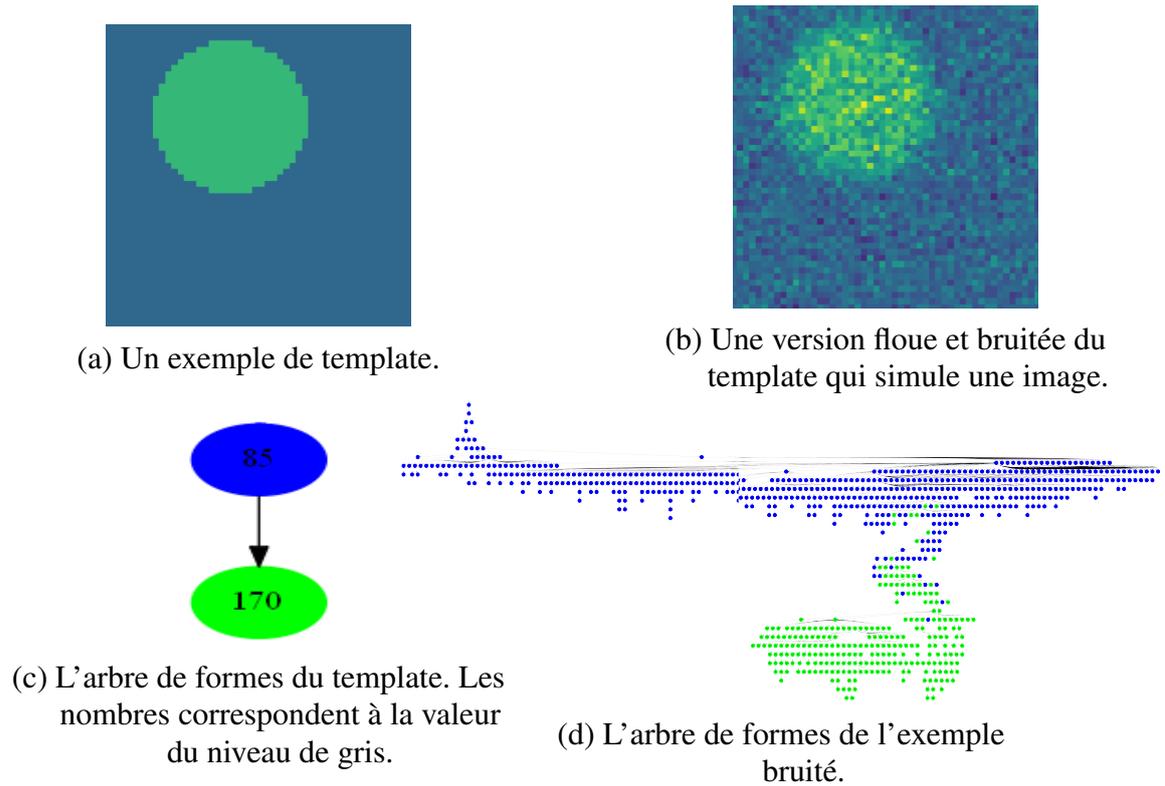


FIGURE 3.3 – Deux exemples d'arbre de formes sur des images simples.

homogènes pour deux raisons. La première est que le niveau de bruit introduit dans l'image simulée est plus important que dans l'image réelle. De plus, notre simulation ne prend en compte ni les imperfections des matériaux ni des patterns ou textures qui pourraient être présents.

3 Segmentation d'images basée sur les matériaux

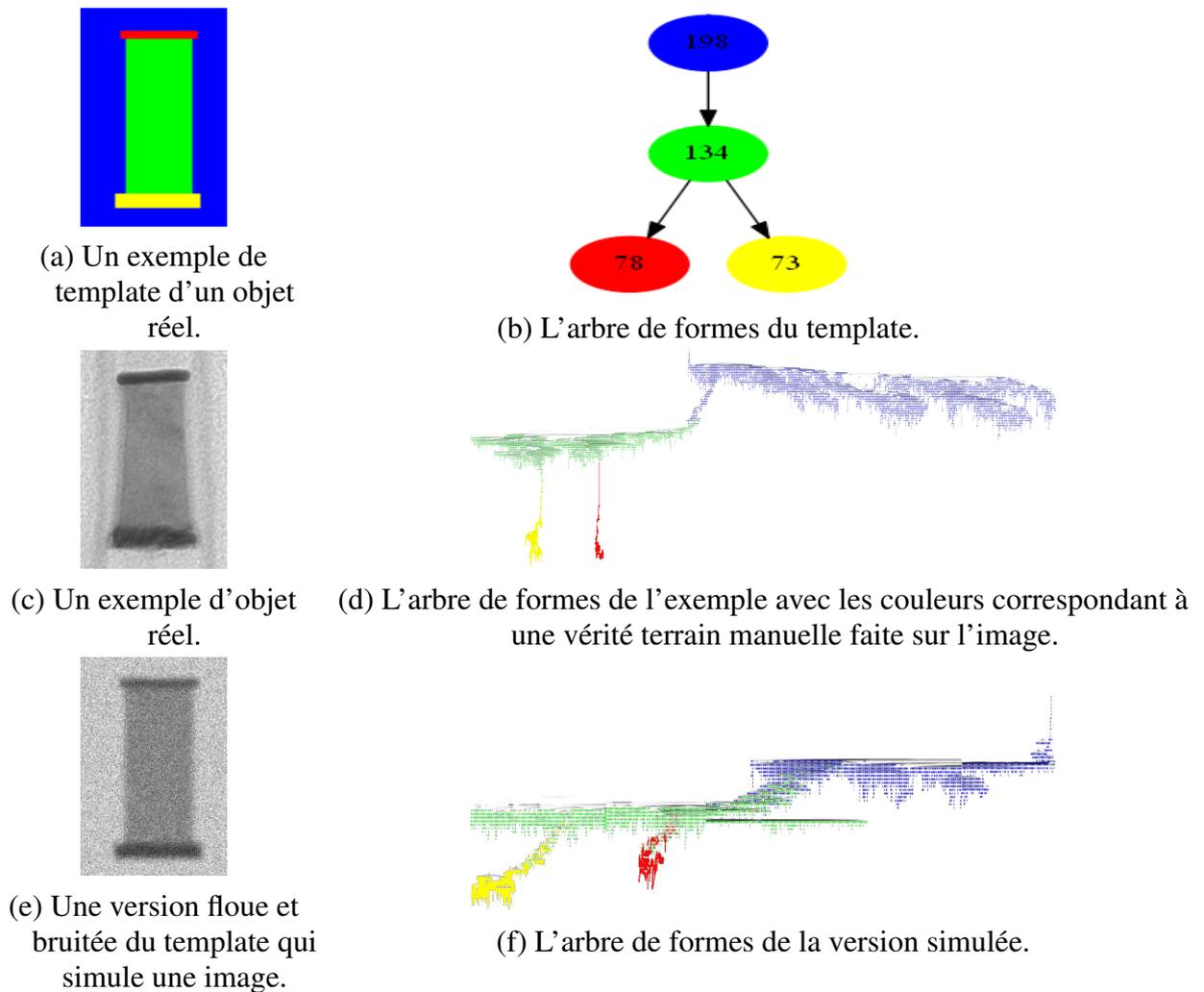


FIGURE 3.4 – Un exemple d'arbre de formes sur des images plus complexes et réelles.

3.2.2 Application

Comme les structures d'arbres de formes se correspondent entre le template et les images réelles à la fois dans la théorie et en pratique, nous avons essayé de développer une méthode de filtrage qui s'appuie sur l'AdF. En effet, le filtrage peut être considéré du point de vue de l'AdF comme un élagage en fusionnant les nœuds de bruit avec des nœuds de signal plus important. Nous avons ainsi proposé une approche permettant de filtrer le bruit de Poisson. L'idée est d'appliquer la transformée d'Anscombe (Équation

3.4) sur l'image puis de la filtrer avec un filtre de bruit gaussien enfin utiliser l'AdF comme support pour effectuer la transformée inverse. Cela nous permet de conserver au mieux les formes de notre image lors de la transformée inverse. Pour ce faire, nous calculons l'AdF de l'image filtrée mais dans le domaine d'Anscombe. Ensuite nous considérons que les formes de l'image filtrée sont celles que nous voulons garder mais elles ont la mauvaise valeur par rapport l'image d'origine. L'intérêt de cette approche est de pouvoir moyennner les valeurs de l'image d'origine bruitée en utilisant les formes comme guide. Nous pouvons ainsi espérer mieux conserver les contours et filtrer plus fortement les régions homogènes. Cela au détriment des petits éléments de détails qui ne bénéficieront pas d'une statistique suffisante pour avoir une moyenne significative. Dans notre cas, avec notre modèle constant par morceaux, ces affirmations sont acceptables.

$$A : x \rightarrow 2\sqrt{x + \frac{3}{8}} \quad (3.4)$$

Nous allons donc détailler l'approche mise en place et la comparer avec des résultats de l'état de l'art. Notre méthode se dérive des approches utilisant la transformée d'Anscombe. Le principal changement est le remplacement de la méthode de calcul de la transformée inverse comme affiché en Figure 3.5. Le principe est le suivant : nous assignons à chaque nœud de l'AdF de l'image filtrée la valeur moyenne de l'image d'origine des pixels contenu dans le nœud (Équation 3.5).

$$G_v = \frac{\sum_{x \in v_{pixels}} I(x)}{|v|}, \forall v \in T_{filter} \quad (3.5)$$

- G_v : le niveau de gris du nœud v
- v_{pixels} : la liste des pixels du nœud v
- $I(x)$: la valeur du niveau de gris dans l'image d'origine
- $|v|$: le nombre de pixels contenu dans le nœud v
- T_{filter} : l'AdF de l'image filtrée

3 Segmentation d'images basée sur les matériaux

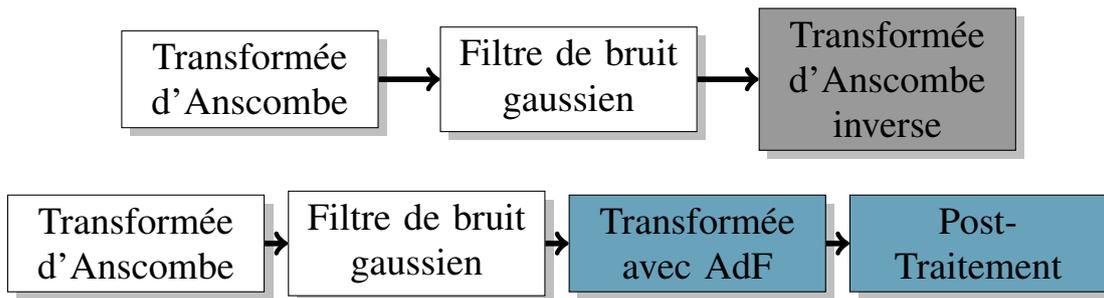


FIGURE 3.5 – Différence entre le pipeline classique et notre proposition.

De cette façon, nous conservons les formes filtrées mais en appliquant les valeurs d'origine. Cependant, nous avons constaté que les zones qui restent bruitées après le filtrage font partie de formes plus petites. Ce que nous entendons par formes bruitées sont des formes qui regroupent des pixels dont la valeur a été corrompue localement par un bruit plus fort que la moyenne ou qui n'ont pas été filtrées suffisamment. A contrario, les formes non bruitées regroupent les pixels faiblement bruités ou correctement filtrés. Cela fait que la forme en question ne récupère que des pixels bruités dans l'image d'origine et donc le filtrage est inefficace. Nous proposons un post-traitement pour améliorer cet aspect avec un filtre de grain [60] de l'AdF. Ce filtre consiste à élaguer les nœuds ayant une taille inférieure à un seuil donné en les incluant dans leur parent. Dans le domaine image, les pixels des nœuds élagués verront leur niveau de gris changer pour celle de leur parent. Ainsi les formes bruités seront incluses dans des formes plus grandes et plus stables.

N'ayant pas d'images parfaites sans bruit, nous avons dans un premier temps comparé quantitativement notre méthode à plusieurs méthodes proposées dans la Section 2.3.1 sur des images naturelles que nous avons corrompu avec du bruit de Poisson. Nous avons quantifié le filtrage avec le Peak Signal to Noise Ratio (PSNR) (Équation 3.6) qui est couramment utilisé dans les comparaisons.

En comparant les résultats entre notre méthode et certains méthodes de l'état de l'art, nous constatons que nos résultats sont inférieurs en particulier dans les zones à forts détails (Figure 3.6 et 3.7). Nous nous attendions

$$PSNR = 10 \cdot \log_{10} \left(\frac{d^2}{EQM} \right) \quad (3.6)$$

d : la dynamique de l'image ($d = 255$ pour les images codées sur 8-bits)

EQM : l'erreur quadratique moyenne

$$EQM = \frac{1}{|X|} \sum_{x \in X} (I_{ref}(x) - I_{bruit}(x))^2 \quad (3.7)$$

$|X|$: le nombre de pixels de l'image

I_{ref} : l'image non bruitée

I_{bruit} : l'image bruitée

à ce que ces zones soient problématiques mais de ce fait, la technique est globalement moins efficace car les effets dans les zones homogènes restent similaires à l'état de l'art. Nous avons essayé plusieurs ajustements afin d'améliorer les performances. Nous avons changé la méthode d'agrégation de moyenne à médiane, nous avons essayé de quantifier l'image afin de réduire le nombre de formes dans l'AdF ou encore essayer d'appliquer un traitement spécifique aux formes que nous avons identifié visuellement comme des formes de bruit.

De plus, nous avons testé la méthode sur des images de notre base (Figure 3.8 et 3.9). Globalement, le filtrage est efficace dans les zones homogènes. Cependant, nous pouvons constater que les contours très bruités comme celui de droite tendent à être effacés par le filtrage. Enfin, la méthode crée des zones de niveau de gris constant avec des formes quelconques, ce qui peut être problématique pour certains algorithmes et ressemblent à des effets de quantification.

Finalement, nous n'avons pas continué à essayer d'améliorer la méthode car les différentes propositions que nous avons semblaient davantage être des petits correctifs que de réelles améliorations. Nous pensons que notre proposition ne fera pas mieux que les méthodes déjà proposées dans l'état de l'art malgré le caractère intéressant et original de la méthode.

3 Segmentation d'images basée sur les matériaux



(a) L'image bruitée.
PSNR=27.12dB

(b) Filtrage avec Anscombe.
PSNR=32.95dB



(c) Filtrage avec les moyennes non
locales. PSNR=31.06dB

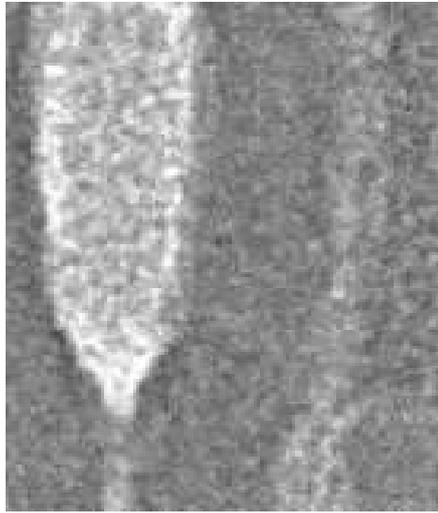
(d) Filtrage avec notre méthode.
PSNR=32.61dB

FIGURE 3.6 – Un exemple d'image naturelle que nous avons faiblement bruitée avec un comparatif de différentes méthodes. L'image provient de [61].

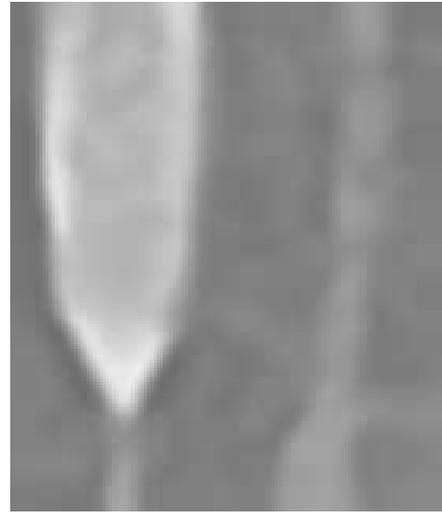


FIGURE 3.7 – Un exemple d'image naturelle que nous avons fortement bruitée avec un comparatif de différentes méthodes. L'image provient de [61].

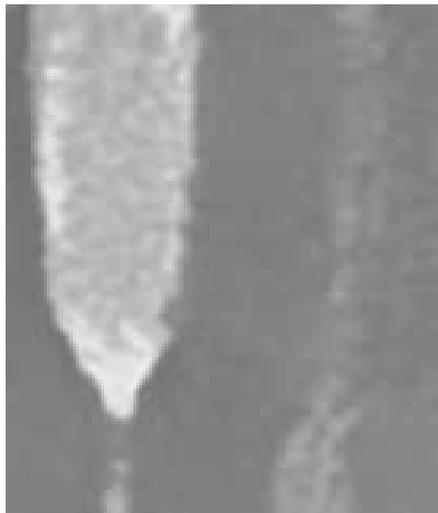
3 Segmentation d'images basée sur les matériaux



(a) L'image bruitée.



(b) Filtrage avec Anscombe.

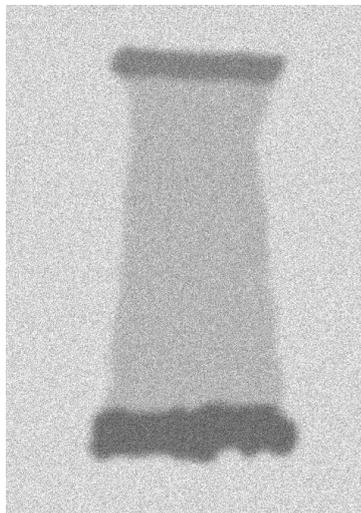


(c) Filtrage avec les moyennes non locales.



(d) Filtrage avec notre méthode à base d'AdF.

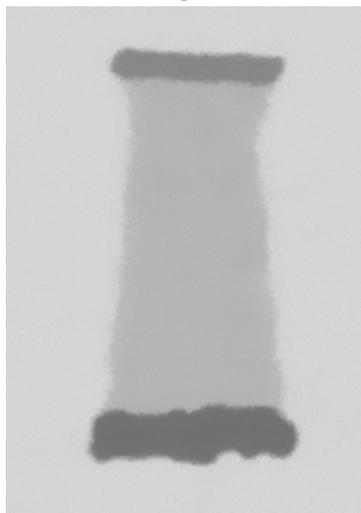
FIGURE 3.8 – Un exemple d'image de semi-conducteur avec un comparatif de différentes méthodes.



(a) L'image bruitée.



(b) Filtrage avec Anscombe.



(c) Filtrage avec les moyennes non locales.



(d) Filtrage avec notre méthode à base d'AdF.

FIGURE 3.9 – Un exemple d'image simulée par notre modèle constant par morceaux avec un comparatif de différentes méthodes.

3.3 Tree Cut segmentation

3.3.1 Introduction

Notre objectif de segmentation est d'obtenir un résultat qui respecte certaines contraintes imposées par le template. La contrainte principale est que le nombre de labels proposé par la méthode doit être le même que le nombre de matériaux. La seconde contrainte est plus souple et concerne les dimensions de différentes régions de la segmentation qui sont similaires à celles du template. Enfin, la géométrie est la dernière contrainte forte de notre segmentation car les objets fabriqués conservent la géométrie du template.

Pour parvenir à cet objectif, nous avons montré empiriquement que l'AdF était une structure de données adaptée à cette problématique. Cependant, il existe de nombreuses possibilités pour utiliser cette structure à des fins de segmentation. Dans un premier temps, nous avons envisagé une approche gloutonne du problème. Cette approche consiste à sélectionner le nœud de l'AdF de l'image le plus ressemblant avec chaque nœud de l'AdF du template. Dans un deuxième temps, nous avons considéré une approche basée sur une minimisation d'énergie nous permettant de mettre en place plus facilement des mécanismes de régularisation.

3.3.2 Formulation du problème

En utilisant l'AdF pour représenter le template et l'image, nous pouvons approcher ce problème en considérant une partition de l'AdF de l'image en un nombre de régions égal au nombre de régions ou nœuds du template. Ce problème de partition est présenté dans l'équation 3.8.

Cependant, en définissant une partition de cette façon, nous n'imposons aucune contraintes sur les coupes. Nous souhaitons associer ces partitions avec les matériaux définis par le template. Pour cela, nous pouvons considérer la relation 3.9 qui permet de minimiser l'attribution des partitions au

$$\begin{aligned}
 V_{img} &= V_{img_1} \cup V_{img_2} \cup \dots \cup V_{img_{\#V_{template}}} \\
 \text{avec} & \\
 V_{img_1} \cap V_{img_2} \cap \dots \cap V_{img_{\#V_{template}}} &= \emptyset
 \end{aligned}
 \tag{3.8}$$

$\#V_{template}$: le nombre de nœuds dans l'AdF du template

matériau le plus similaire.

$$D(V_{img}) = \sum_{k \in T_{img}} \min_{\forall v \in V_{template}} 1 - \text{Sim}(v, V_k)
 \tag{3.9}$$

$\text{Sim}(v, V_k)$: une mesure de similarité comprise entre 0 et 1

Dans la littérature, ce type de problème peut être approché par de la mise en correspondance de graphes. L'objectif est de lier les nœuds de l'AdF du template aux nœuds de l'AdF de l'image. Les approches [62] et [63] proposent des solutions qui visent à faire une mise en correspondance un à un des nœuds d'un graphe vers l'autre. De façon plus souple, la mise en correspondance d'arbre inexact permet d'élargir le problème en liant un nœud vers plusieurs comme dans [64] et [65]. Les ordres de grandeur du nombre de nœuds à mettre en correspondance restent tout de même différents, la mise en correspondance inexacte prévoit de lier un nœud à quelques autres alors que dans notre cas, nous visons un rapport de un nœud vers des milliers.

3.3.3 Approche gloutonne

Le nombre de nœuds d'un AdF dans nos portions d'images qui représentent des objets sont de l'ordre de la dizaine de milliers de nœuds. Dans ce contexte, la complexité du calcul d'une partition sur un tel arbre devient limitante en terme de temps de calcul. Il est cependant possible de

3 Segmentation d'images basée sur les matériaux

créer une partition de l'arbre en utilisant une approche gloutonne. Cette motivation provient du fait que l'AdF est une structure orientée ainsi il est possible de penser à une approche gloutonne pour faire une partition avec la contrainte de similarité.

Le template et l'image ayant des nombres de nœuds différents de plusieurs ordres de grandeur, cette mise en correspondance est faite à un nœud du template vers plusieurs nœuds de l'image. Une approche gloutonne peut donc se traduire par une approche itérative qui cherche à assigner à un nœud du template un ensemble connecté (dans le cas d'une partition) de nœuds de l'image. La complexité de ce problème peut être importante car nous voulons trouver l'ensemble de nœuds qui a les caractéristiques les plus proches du nœud du template. Par caractéristique, nous entendons la valeur moyenne de niveau de gris ou l'espace recouvert dans l'image, par exemple. Il faudrait calculer pour un nœud de l'image les caractéristiques de tous les ensembles que nous pouvons faire avec ces descendants. Cela donne un nombre de caractéristiques par nœud égal à son nombre de descendants (Équation 3.10) en considérant, comme caractéristique, la moyenne des niveaux de gris. De ce fait, pour chaque nœud du template, nous devons faire une recherche parmi N possibilités, N étant le nombre de nœuds de l'image.

$$\begin{aligned} & \text{minimiser } \sum_{i \in T_{\text{template}}} \left\| G_{V_i} - \frac{\sum_{j \in T_{\text{image}}} G_{V_j} x_{i,j}}{\sum_{j \in T_{\text{image}}} x_{i,j}} \right\|^2 \\ & \text{avec } x_{i,j} \in \{0, 1\} \text{ and } \forall j \sum_{i \in T_{\text{template}}} x_{i,j} = 1 \end{aligned} \quad (3.10)$$

T_{template} : l'AdF du template

T_{image} : l'AdF de l'image

G_V : le niveau de gris du nœud V

$x_{i,j}$: un index permettant d'assurer que chaque nœud de l'image est associé à un seul nœud du template

3.3 Tree Cut segmentation

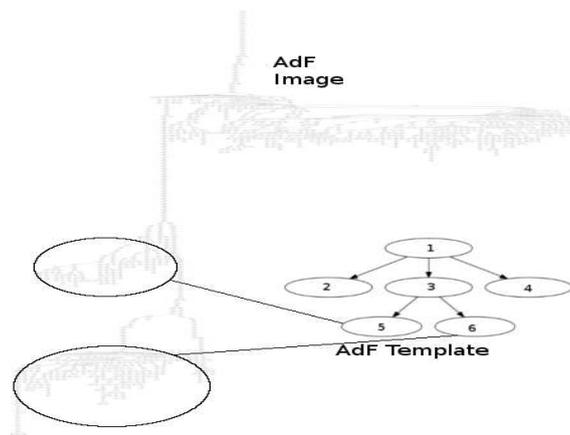
Nous avons pris le parti de faire une mise en correspondance en commençant par les parties les plus profondes de l'arbre (Figure 3.10). Ainsi pour calculer efficacement des attributs sur des groupes de nœuds, nous avons choisi de considérer le nœud courant et ses descendants. Grâce à ce choix, en réalisant une mise en correspondance en parcourant les deux AdF par les nœuds les plus profonds, nous pouvons ainsi assigner un nœud du template à un groupement de nœuds de l'image. Une fois un minimum trouvé, nous pouvons simplifier l'AdF de l'image afin de passer au nœud suivant du template pour la mise en correspondance.

Avec la simplification proposée, nous pouvons faire la mise en correspondance en cherchant seulement dans N possibilités pour chaque nœud de l'AdF du template.

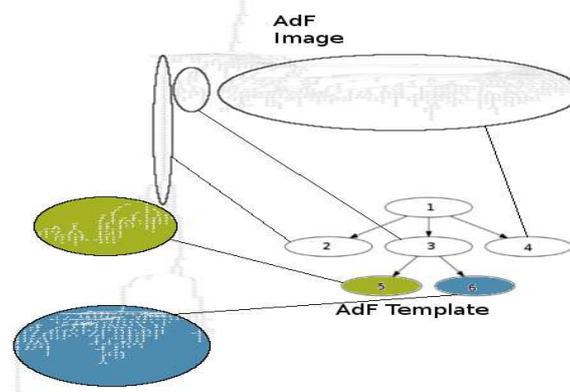
Dans l'exemple fourni (Figure 3.11), nous pouvons constater que les résultats sont plutôt satisfaisants sur un exemple simple. Il y a tout de même quelques défauts créant des trous dans certains matériaux qui nécessiterait une boucle de rétroaction. De plus, l'approche est déjà assez coûteuse en ne considérant que le niveau de gris malgré la complexité faible. Le temps de calcul est de l'ordre de 10 à 30 secondes selon la taille de l'AdF de l'image. L'ajout de la position comme caractéristique rendrait l'approche vraiment trop lente. En effet, le niveau de gris est une information rapide à agréger, le calcul d'une moyenne étant possible de manière itérative et à comparer par avec une différence. Pour la position, l'agrégation demandera soit plus de temps ou de mémoire car on agrège plus d'une valeur pour chaque nœud. La comparaison demande le parcours de grandes listes de points pour chaque nœud de l'AdF du template. Cependant, ce premier essai nous indique que la piste est potentiellement intéressante.

Comme la majorité des algorithmes gloutons, nous n'avons pas de garanties de trouver un minimum global ni de connaître la marge d'erreur entre la solution proposée et la solution globale. De plus, de par la construction itérative, nous savons que les erreurs sont propagées. Pour corriger cela, il faudrait mettre en place une rétroaction permettant de modifier les mises en correspondance précédentes. La mise en place d'une rétroaction

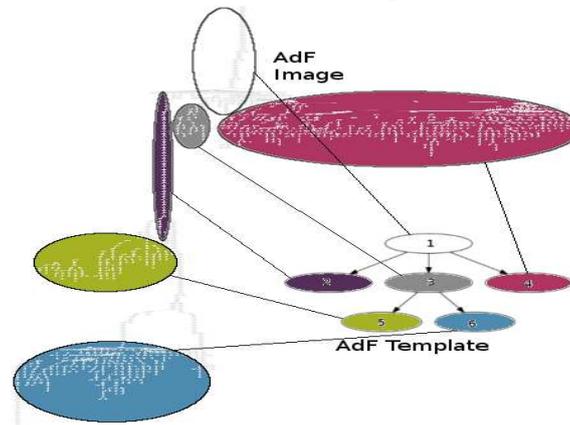
3 Segmentation d'images basée sur les matériaux



(a) Première étape de la mise en correspondance.



(b) Deuxième étape.



(c) Dernière étape.

FIGURE 3.10 – Démonstration étape par étape de l'approche gloutonne.

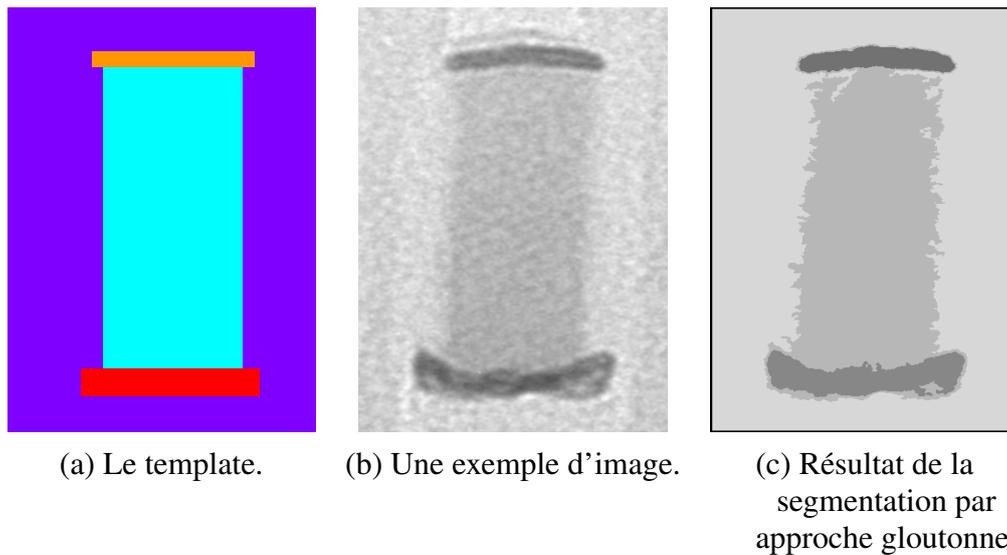


FIGURE 3.11 – Démonstration étape par étape de l'approche gloutonne.

revient à chercher $v - 1$ coupes dans l'AdF de l'image, v étant le nombre de nœuds dans l'AdF du template. Avec une approche de force brute, la complexité d'une telle approche est en $O(n^{v-1})$, ce qui est trop coûteux.

Pour ces raisons, nous avons choisi de mettre en place un algorithme différent.

3.3.4 Framework proposé

Nous avons vu que le problème de partition était compliqué à résoudre à cause de la taille des arbres en jeu. Nous avons choisi de relaxer la problématique et d'imposer une régularisation pour approcher la partition. La partition est une réponse idéale à notre problème mais dans les faits, des défauts dans la partition sont acceptables tant que les frontières entre les zones sont correctes. Nous pouvons ainsi poser une énergie globale sur l'AdF et la minimiser. Nous avons donc deux termes : un pour la fidélité aux données et un autre pour la régularisation. Nous minimisons l'équation suivante 3.11.

Le terme de fidélité mesure les différences entre le template et l'image

3 Segmentation d'images basée sur les matériaux

$$E(V) = \sum_{\substack{i \in T_{template} \\ j \in T_{img}}} D_i(V_j) + \sum_{\substack{i \in T_{img} \\ j \in N_i}} R_{ij}(V_i, V_j) \quad (3.11)$$

$D_i(V_j)$: le terme de fidélité aux données du nœud V_j appartenant au nœud de l'AdF V_i
 R_{ij} : le terme de régularisation entre les nœuds i et j
 N_i : le voisinage du premier ordre du nœud i , cela correspond aux nœuds *parent* ou *enfant* de i

pour que la segmentation proposée corresponde au modèle. La régularisation nous permet d'approximer le problème de partition. Cette formulation est très proche de la formulation du Graph Cut [66] avec un terme de fidélité qui lie les nœuds du graphe à un label et un terme de régularisation qui lie les nœuds entre eux en fonction de la structure du graphe.

La formulation d'origine du Graph Cut est une proposition de segmentation binaire utilisant un graphe comme support. L'idée est de considérer deux labels (Figure 3.12), le fond et l'objet, et de créer un graphe qui représente l'image et relie les deux labels. Dans le Graph Cut, la structure de graphe est relativement simple, nous reprenons la structure de l'image et nous considérons chaque pixel comme un nœud relié à ces pixels voisins. Ensuite, il faut établir les poids des connexions entre un label et un pixel et entre deux pixels. Les formulations sont assez libres selon les besoins. La connexion label/pixel représente la similarité entre le pixel et le label. La connexion pixel/pixel représente un terme de régularisation qui permet d'imposer des contraintes sur la segmentation finale.

Les avantages de cette formulation sont les suivants. Nous pouvons calculer la minimisation de l'équation 3.11 comme décrit dans [67] si nos termes respectent certaines conditions avec des garanties sur les minima locaux dans le cas n-aire. De plus, le terme de fidélité peut être enrichi avec d'autres informations différentes. Par exemple, nous considérerons le niveau de gris ou la position des pixels dans nos expériences mais nous pourrions aussi utiliser la texture ou des descripteurs de formes ou étendre

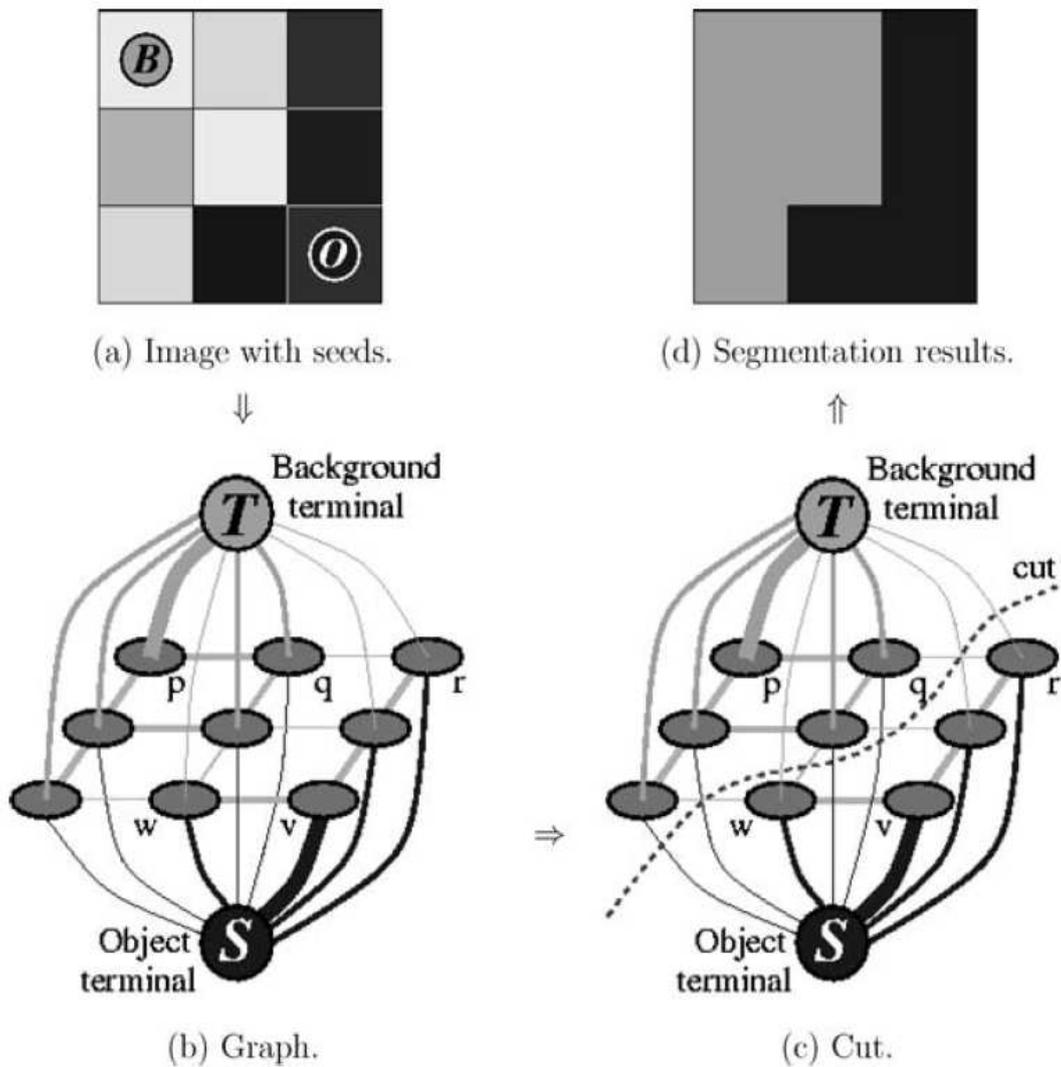


FIGURE 3.12 – Illustration de la méthode de segmentation Graph Cut dans le cas binaire. Le visuel provient de [66].

la formulation à des images multi-modales.

Nous avons mis en place un framework qui permet d'utiliser un template pour produire une segmentation sur une image réelle. Ce framework utilise l'AdF comme support pour à la fois représenter le template de manière efficace et aider à la segmentation.

Pour le terme de fidélité, le niveau de gris est le premier élément qui

3 Segmentation d'images basée sur les matériaux

```
Pour chaque label  $\alpha$  :  
   $\alpha$  Expansion move :  
    Calcul de l'énergie de chaque nœud pour passer  
    au label  $\alpha$   
    Changement de label à  $\alpha$  si l'énergie est plus  
    basse  
  Stop si convergence
```

FIGURE 3.13 – Description de l'algorithme d' α -expansion.

nous semble pertinent. Il est en effet caractéristique dans les images TEM et chaque matériau physique correspond à un niveau de gris. Il peut être facilement renseigné par un utilisateur. La précision de cette information a priori n'est pas importante. Cependant, seul, il montre des limites dans le cas d'images avec des matériaux similaires avec des niveaux de gris proches et/ou dans le cas où deux régions de même matériau sont disjointes.

La position relative est donc un bon complément au niveau de gris vis-à-vis des remarques précédentes. Cette information est disponible directement via l'image du template. Elle permet de lever les ambiguïtés dans les situations problématiques pour le niveau de gris. Cependant, les formes du template sont très idéalisées et donc l'utilisation de la position relative devrait dégrader légèrement les performances de l'algorithme à l'interface de deux matériaux.

Pour la régularisation, nous avons tout d'abord regardé quelles sont les formulations employées dans le Graph Cut. La régularisation de Potts est celle qui ressort le plus. Elle permet d'homogénéiser les régions en pénalisant les attributions de labels qui sont différentes de l'attribution faite aux voisins (Équation 3.12). De cette façon, nous approchons le problème de la partition en créant des régions dans l'arbre qui sont largement connectées. Elle ne prend cependant pas en compte la structure de l'AdF du template.

Pour ce faire, nous proposons d'utiliser un terme de régularisation qui va

$$R = 1 - \delta(V_i, V_n) \quad (3.12)$$

R : le terme de régularisation d'un nœud

δ : le symbole de Kronecker $\delta(i, j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$

V_i : le nœud considéré

V_n : un nœud voisin (parent ou enfant) de V_i

pénaliser les attributions de labels en fonction de la structure de l'AdF du template et ainsi favoriser les labellisations qui essaient de recréer un AdF similaire à celui du template. Enfin, quelques étapes de post-traitement seront nécessaires afin de satisfaire les contraintes de notre problème (Sous Section 3.3.2).

Terme de fidélité

Comme présenté précédemment, nous avons choisi d'utiliser deux termes complémentaires. Pour le terme qui concerne le niveau de gris, nous avons utilisé la norme L2 (Équation 3.13). Ce choix a été fait de par la simplicité de l'implémentation malgré le fait que la norme L2 ne soit pas particulièrement adaptée à la distribution de Poisson. Cependant, notre modèle étant constant par morceaux et le nombre de niveau de gris différent étant faible, nous approximons le bruit de Poisson par un bruit gaussien de manière locale comme présenté en section 2.3.

$$D_{V_{ig}}(V_j) = \|G_{V_i} - G_{V_j}\|_2 \quad (3.13)$$

G_{V_x} : le niveau de gris du nœud V_x

Nous avons ajouté un terme lié à la position des matériaux dans le template vis-à-vis de l'image. Le but est d'avoir un terme qui maximise le recoupement entre la segmentation proposée et le template. De cette fa-

3 Segmentation d'images basée sur les matériaux

çon, en combinaison avec le terme précédent, notre minimisation d'énergie cherche à proposer une segmentation qui est proche du template à la fois spatialement et sur les niveaux de gris. Pour cela, nous comptons le nombre de pixels du nœud de l'image qui ne se recoupent pas avec les pixels du nœud du template mis en correspondance (Équation 3.14). Nous avons normalisé ce terme afin que chaque nœud ait une valeur cohérente avec les autres car les nœuds ont des tailles différentes.

$$D_{V_i p}(V_j) = \frac{|P_{V_j} \setminus (L_j \cap P_{V_j})|}{|P_{V_j}|} \quad (3.14)$$

P_{V_x} : la liste des pixels de V_x

$|P_{V_x}|$: le nombre de pixels de P_{V_x}

L_x : la liste des pixels du nœud du template ayant le même label que V_x

Afin de combiner le deux termes, nous avons ajouté un terme de pondération afin de pouvoir contrôler l'impact d'un paramètre par rapport à l'autre (Équation 3.15). Ce terme a vocation à être modifié selon les cas d'utilisation. Pour les cas où les matériaux sont fortement déformés, nous voulons prendre en compte davantage le niveau de gris. Par contre dans des cas fortement bruités, la position sera plus utile.

$$D_{V_i}(V_j) = \alpha D_{V_i g}(V_j) + \beta D_{V_i p}(V_j) \quad (3.15)$$

Pour des raisons techniques, nous avons mis un terme de pondération de chaque terme de fidélité car pour mettre en œuvre les algorithmes de minimisation de façon efficace, les poids appliqués à l'AdF doivent être des entiers naturels. Ainsi, comme nous avons $D_{V_i g}$ qui est compris entre 0 et 255 et $D_{V_i p}$ compris entre 0 et 1, nous avons multiplié ce dernier par 255 afin d'avoir des dynamiques similaires. Ensuite, α et β doivent être des entiers naturels aussi pour satisfaire la condition énumérée. Nous verrons que la terme de régularisation obéit à ces contraintes par construction.

Terme de régularisation

Nous avons présenté le terme de régularisation de Potts (Équation 3.12) qui est classiquement utilisé dans la formulation du Graph Cut. Dans notre cas, nous proposons d'utiliser un terme différent qui pénalise en fonction de la structure de l'AdF du template. Cependant, afin d'avoir des garanties de convergence pour l'algorithme d'alpha-expansion (Figure 3.13), notre terme de régularisation doit satisfaire les conditions d'une distance (Figure 3.14) ou au moins d'une semimétrie.

Soit f une fonction positive de \mathbb{R}^2 vers \mathbb{R} . f est une distance si :

$$f(x, y) = 0 \Leftrightarrow x = y \quad (3.16)$$

$$f(x, y) = f(y, x) \quad (3.17)$$

$$f(x, y) \leq f(x, z) + f(z, y) \quad (3.18)$$

FIGURE 3.14 – Conditions pour qu'une fonction soit une distance.

Le but de notre régularisation est de pénaliser davantage les nœuds qui n'ont pas le même label que leurs voisins de façon proportionnelle à leur différence de label dans l'AdF du template. Considérons un AdF de l'image avec deux nœuds et un AdF du template avec trois nœuds (Figure 3.15). En assignant le label A au nœud feuille de l'AdF de l'image, il ne sera pas pénalisé car son parent a le même label. Dans la régularisation de Potts, il sera pénalisé de la même manière si nous lui assignons le label B ou C. Nous proposons de pénaliser davantage si nous assignons le label C par rapport à B car dans l'AdF du template, le nœud C est visuellement plus distant que le nœud B. Cela traduit le fait qu'il est peu probable que dans notre segmentation, un nœud avec le label A soit voisin d'un nœud avec un label C. Il convient donc de définir cette notion de distance.

Il existe beaucoup de façons de définir une distance entre des nœuds dans

3 Segmentation d'images basée sur les matériaux

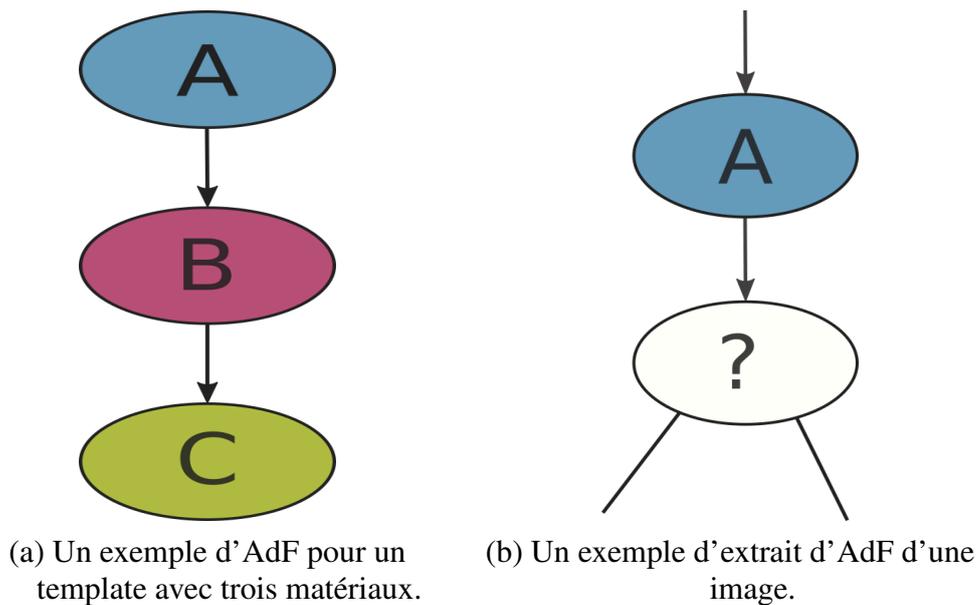


FIGURE 3.15 – Exemple de calcul pour la régularisation.

un arbre. Dans notre cas, nous voulons compter la distance entre les nœuds quelque soit la direction. De plus, le but est de quantifier la proximité des régions dans l'AdF, nous avons donc choisi le plus court chemin comme métrique. Nous pouvons vérifier aisément que cette formulation est une distance.

$$R_{ij}(V_i, V_j) = \min_{\forall C \in T_{design}} \sum_{i=L_{V_i}}^{L_{V_j}} f(C_{i,i+1}) \quad (3.19)$$

C : un chemin $C = (V_1, V_2, \dots, V_n)$ avec $C_1 = L_{V_i}$ et $C_2 = L_{V_j}$

$\forall C \in T_{design}, f(C_{i,i+1}) = 1$: les poids que nous assignons aux connexions entre les nœuds de l'AdF du template

Ce framework s'inspire du Graph Cut pour proposer une segmentation basée sur un template qui a l'avantage d'être efficace car l'AdF est moins dense que le graphe du Graph Cut et adapté à l'utilisation d'un template

comme ceux qui sont à notre disposition.

Pour finir, nous avons mis au point un méthode de post-traitement pour affiner le résultat et le rendre plus cohérent à notre définition de partition. La minimisation ne permet pas d'obtenir un nombre de régions finales en accord avec le nombre de matériaux. Nous avons donc choisi de garder pour chaque matériau la forme la plus grande en aire dans la segmentation finale. Ainsi, chaque autre forme voit son label changer pour le label de son ascendant d'une des formes les plus grandes. Ce post-traitement améliore légèrement les résultats et mériterait d'être amélioré pour mieux tenir compte des poids relatifs des formes les plus grandes. En effet, avec cette méthode, si un matériau est séparé en deux formes dans la segmentation, nous perdons la moitié du résultat à cause de ce traitement.

3.4 Résultats

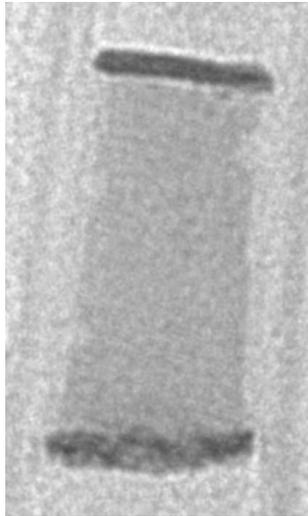
Dans cette section, nous montrons l'impact des différents paramètres de notre méthode et nous comparerons celle-ci avec des méthodes de l'état de l'art. Nous ferons aussi des comparaisons sur les différents paramètres. Enfin, nous discuterons des résultats sur deux jeux de données.

3.4.1 Impact des caractéristiques

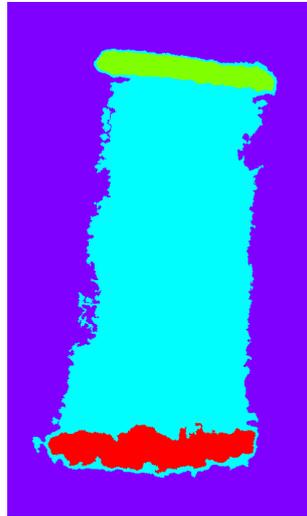
Nous avons mis en évidence trois caractéristiques dans notre méthode : le niveau de gris, la position et la régularisation (Figure 3.16). Le niveau de gris permet de respecter le niveau global de l'intensité attendu. Lorsque nous surestimons cette caractéristique, les formes bruitées sont mal segmentées et des inversions de matériaux similaires apparaissent. La position différencie les objets lorsque les matériaux sont similaires. Cependant, en forçant cette caractéristique, nous obtenons une segmentation qui s'approche du template et qui respecte moins les caractéristiques propre à l'image. Enfin, la régularisation homogénéise les régions forçant

3 Segmentation d'images basée sur les matériaux

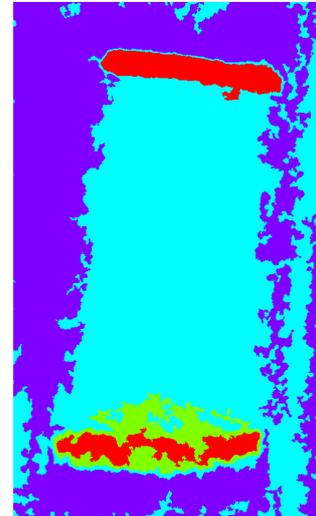
les formes adjacentes à avoir des labels similaires.



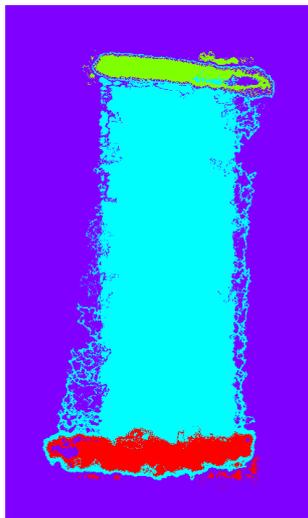
(a) Exemple de mémoire.



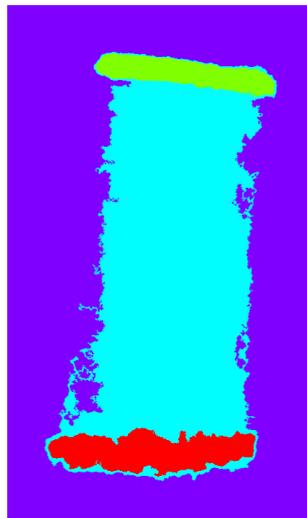
(b) Le meilleur jeu de paramètres.



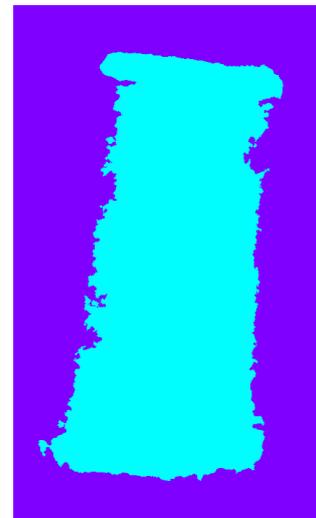
(c) Jeu de paramètres avec le niveau de gris fortement augmenté.



(d) Jeu de paramètres avec la position fortement augmentée.



(e) Jeu de paramètres avec la régularisation faible.



(f) Jeu de paramètres avec la régularisation forte.

FIGURE 3.16 – Exemples de variations de segmentation en fonction des paramètres.

3.4.2 Présentation des données et méthodes

Le premier jeu de données représente un élément présent dans la fabrication de mémoires (Figure 3.17). Il contient trois matériaux plus le fond. La particularité de ce jeu est que deux matériaux ont des niveaux de gris similaires et que le troisième matériau peut présenter des écarts de niveau de gris similaire aux autres matériaux. Ces écarts correspondent cependant au même matériau.

Le deuxième jeu de données est une étape de fabrication de transistor Fin Field-Effect Transistor (FinFET) et représente la partie *fin*. Les images à notre disposition étant confidentielles, l'objet affiché n'est pas exactement celui de notre base mais il est similaire (Figure 3.17d). La particularité de ces objets est la présence de régions longues et fines qui sont compliquées à segmenter comme la bande claire entre la bande noire et l'élément central gris foncé de l'exemple. De plus, dans notre jeu de données, il a un matériau qui est inclus dans un second matériau de niveau de gris similaire.

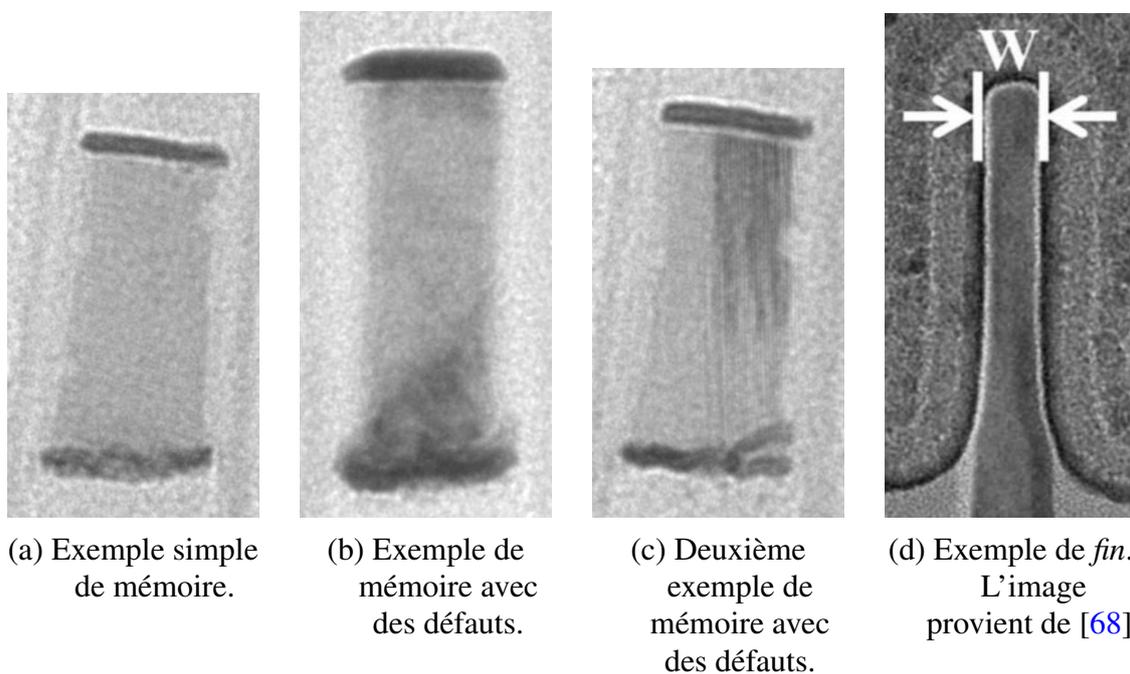


FIGURE 3.17 – Exemples des objets de nos jeux de données.

Nous avons mis en place quatre méthodes de segmentation pour la com-

3 Segmentation d'images basée sur les matériaux

paraison. Nous avons repris des méthodes proposées dans l'état de l'art de la segmentation d'images TEM.

La première technique mise en place est le Graph Cut [66]. Le template est utilisé pour faire une première segmentation du graphe et nous optimisons ensuite cette initialisation avec l'algorithme de Graph Cut. Nous avons cependant apporté une amélioration en ajoutant la position comme critère supplémentaire dans le terme d'attache aux données. Ce terme permet d'améliorer la distinction entre deux matériaux de niveau de gris proche.

La seconde méthode est le watershed. Cette approche nécessite une initialisation avec des graines. Pour que l'approche reste proche de la philosophie de notre cas d'utilisation, les graines doivent être placées sur le template. Nous avons réalisé cette étape manuellement mais un érodé ultime sur chaque matériau pourrait les extraire de façon automatique. Ensuite, nous avons utilisé la carte de gradient comme support pour l'algorithme.

La troisième méthode ne provient pas de l'état de l'art de la segmentation TEM. C'est une méthode de segmentation basées sur les lignes de niveaux, Chan Vese multiphase [55]. Pour cette méthode, le paramètre d'entrée est le nombre de matériaux dans le template.

Enfin, la dernière méthode est basée sur le deep learning. Pour cette approche, nous avons choisi la problématique de segmentation sémantique. Dans ce problème, chaque pixel de l'image est assigné à un label. Ainsi, nous considérons chaque matériau du template comme un label. Nous avons utilisé le NN U-net [69]. Le principe de ce NN est d'extraire des descripteurs à plusieurs échelles pour ensuite construire une carte de segmentation à chaque échelle en utilisant à la fois les descripteurs de l'échelle courante et la segmentation de l'échelle précédente. L'intérêt principal pour notre problématique est la faible quantité de données utilisée pour l'entraînement.

3.4.3 Comparaison qualitative

Avec les deux exemples (Figures 3.18 et 3.19), nous illustrons que les résultats de la méthode proposée sont meilleurs. Les résultats de Chan-Vese multiphase confond les matériaux orange (2) et rouge (3) qui sont très similaires en terme de niveau de gris. Pour cette méthode, le niveau de gris est un des facteurs principaux de discrimination. Pour le watershed, les matériaux orange et rouge sont plutôt bien segmentés car ils sont bien contrastés. Par contre, le matériau bleu (1) est moins bien segmenté car moins contrasté par rapport au fond. Enfin pour le Graph Cut, la version classique, les matériaux orange et rouge sont confondus comme pour Chan-Vese multiphase. Avec la position en plus, les résultats s'améliorent car le template permet d'augmenter la discrimination entre les deux matériaux similaires.

Enfin, pour l'approche deep learning, nous avons intégré les résultats à part car ils sont clairement sous-optimaux (Figure 3.20). La faute revient à notre jeu de données. Il est à la fois trop petit et trop difficile. En effet, sur une aussi faible quantité de données, il est difficile pour un NN d'apprendre la distinction entre les zones que nous avons défini car c'est principalement le niveau de gris et la position relative qui entrent en jeu. Nous avons inclus cette méthode afin d'essayer une méthode parmi la littérature récente du deep learning même si notre problématique n'est pas très adaptée à l'utilisation de ces techniques en terme de jeu de données labellisées.

Notre méthode propose des résultats plus robustes. Les matériaux orange et rouge sont toujours distingués. Le fond est segmenté de façon homogène. Le principal défaut est l'effet de halo que nous pouvons percevoir autour des matériaux orange et rouge. Il est particulièrement marqué sur l'exemple difficile (Figure 3.19). Cet effet provient du fait que l'AdF regroupe les formes en fonction de leur niveau de gris. Ainsi, quand des contours sont flous, le dégradé de niveau de gris rassemble des pixels dans la même forme qui est plus proche du matériel bleu.

3 Segmentation d'images basée sur les matériaux

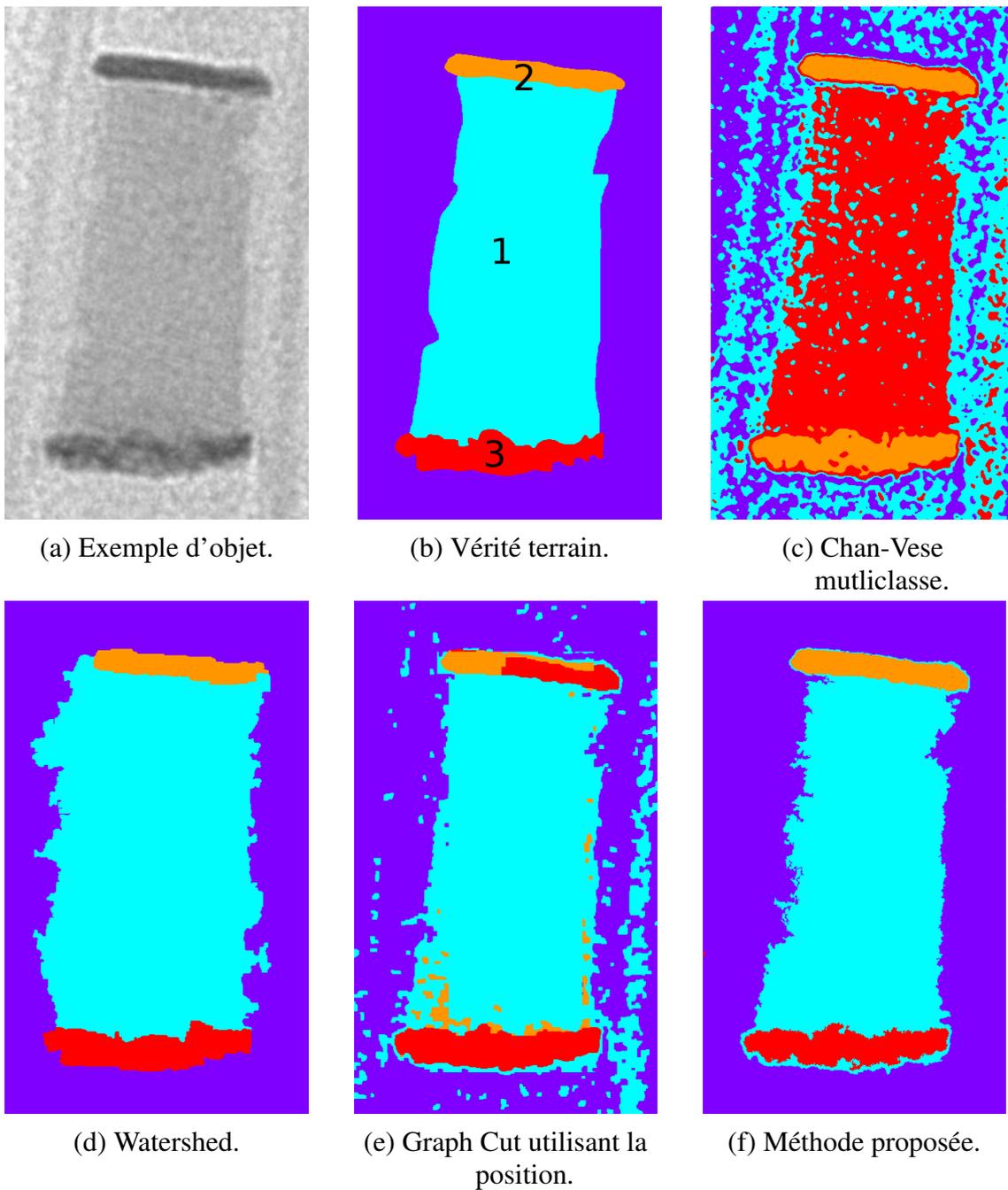


FIGURE 3.18 – Comparaison des résultats sur un exemple simple du jeu de données 1.

Pour le jeu de données 2, les deux difficultés sont que un matériau présente des régions longues et fines et que deux matériaux ont des niveaux de gris similaires. Pour les régions longues et fines, le watershed est mis

en difficulté car la graine que nous avons initialisée est trop loin du bout de cette région et donc un label plus proche finit par déborder dessus. Notre approche fonctionne mieux grâce aux lignes de niveaux qui permettent ainsi de lier les pixels dans la même forme même si la région est longue et fine. Pour les niveaux de gris similaires, Chan-Vese multiphase et le Graph Cut montrent les mêmes problèmes que sur le jeu de données 1. Pour finir, les défauts de notre approche sur le jeu de données 1 sont les mêmes sur le jeu de données 2.

3 Segmentation d'images basée sur les matériaux

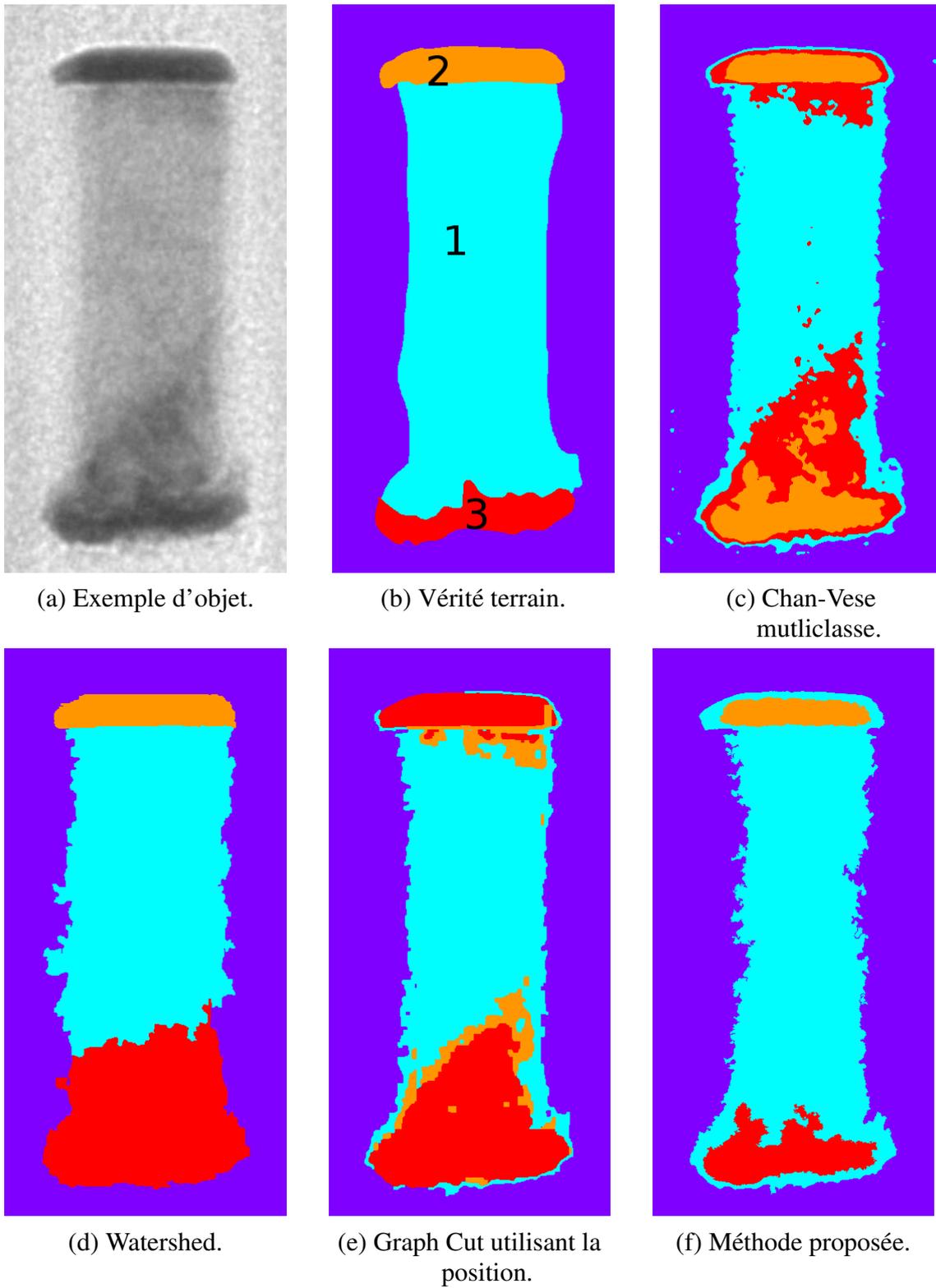


FIGURE 3.19 – Comparaison des résultats sur un exemple difficile du jeu de données 1.

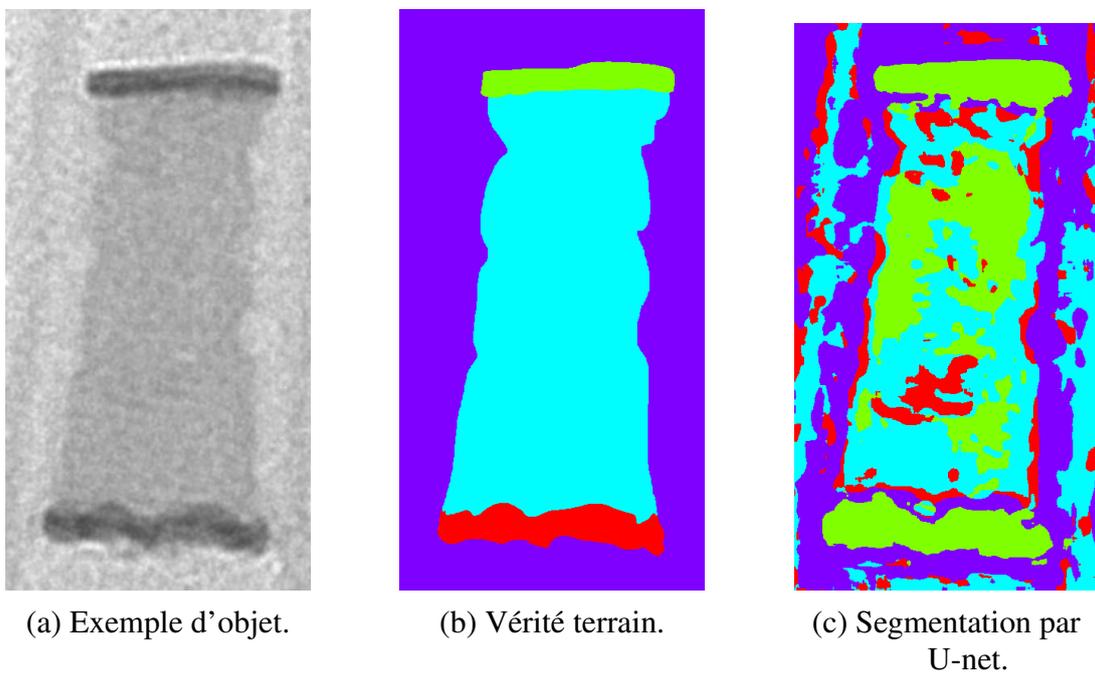


FIGURE 3.20 – Présentation des résultats de U-net.

3.4.4 Comparaison quantitative

Pour comparer les méthodes quantitativement, nous avons utilisé le coefficient de Dice (Équation 3.20). Nous avons dans un premier temps utilisé notre modèle d'image constant par morceaux (Section 2.3). Nous avons pris les segmentations vérité terrain de notre jeu de données pour générer des images synthétiques. Nous leur avons appliqué un filtre gaussien pour simuler le filtrage passe-bas du système de lentilles magnétiques puis un bruit de Poisson avec une intensité variable (Équation 3.21) [70] afin de pouvoir comparer les méthodes en fonction du niveau de bruit de l'image. Dans cette fonction de bruit, plus le paramètre est faible et plus le bruit est élevé.

$$D = \frac{2|X \cap Y|}{|X| + |Y|} \quad (3.20)$$

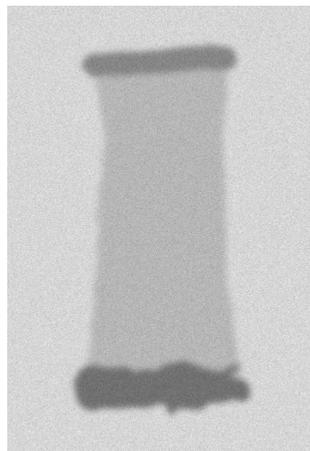
X, Y : des ensembles de pixels
 $|X|$: le nombre d'éléments de X

$$Noisy(x) \sim \frac{P(s \times I(x))}{s}, \forall x \in X \quad (3.21)$$

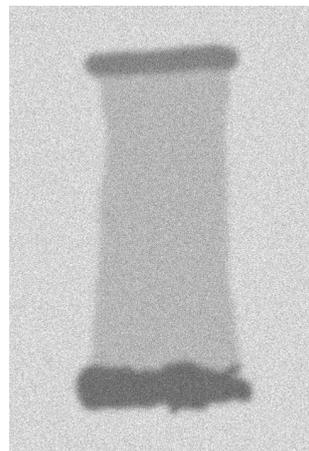
$Noisy$: l'image bruitée
 $P(\lambda)$: une loi de Poisson de paramètre λ (Équation 2.7)
 $\frac{1}{s}$: le niveau de bruit

Sur ces images simulées (Figure 3.21), nous avons appliqué notre algorithme de segmentation. Nous pouvons observer un effet de halo sur les images faiblement et modérément bruitées. Cela est dû au fait que le filtre passe bas floute les bords et qu'en terme d'AdF, cela regroupe les pixels dans une même forme. Lorsque le niveau de bruit augmente, il y a davantage de débordements de la segmentation jusqu'à atteindre un niveau où ces sur-segmentations deviennent problématiques. Les zones occupées par

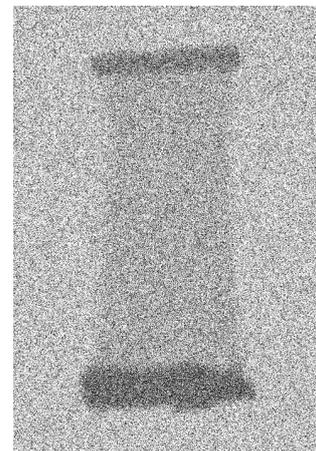
les segmentations ne sont plus stables. Dans l'exemple le plus bruité (Figure 3.21f), le matériau central est largement sur-segmenté en dépit des autres. Dans d'autres exemples de niveaux de bruit similaires, le matériau central est sous-segmenté au profit des autres.



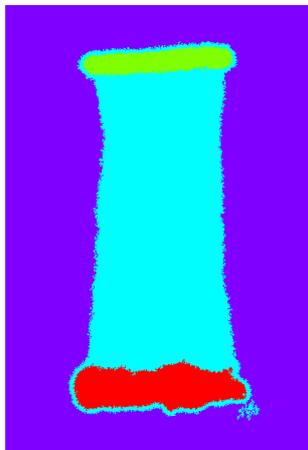
(a) Simulation avec un paramètre de bruit de 0.05.



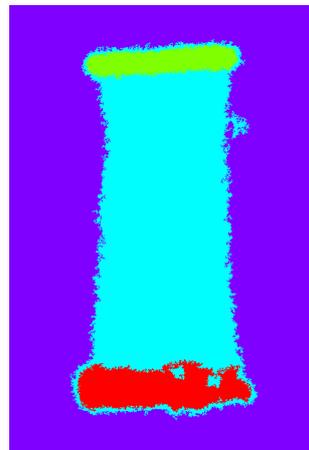
(b) Simulation avec un paramètre de bruit de 0.1.



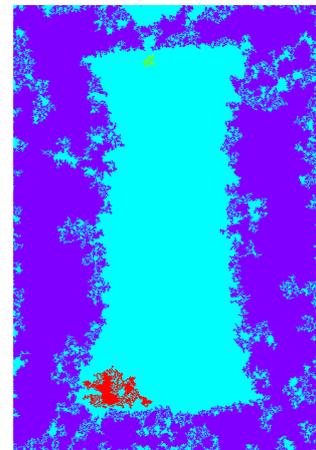
(c) Simulation avec un paramètre de bruit de 0.5.



(d) Résultat du tree cut sur l'image synthétique (a).



(e) Résultat du tree cut sur l'image synthétique (b).



(f) Résultat du tree cut sur l'image synthétique (c).

FIGURE 3.21 – Illustration des résultats sur des images simulées.

En effectuant cette analyse sur le jeu complet avec des niveaux de bruit variables (Figure 3.22), nous pouvons suivre l'évolution des performances des différentes méthodes. Nous constatons que le Tree Cut et le watershed

3 Segmentation d'images basée sur les matériaux

sont similaires en terme de performance sur les premiers niveaux de bruit et sont au-dessus des deux autres méthodes. Cependant, les performances du watershed décrochent rapidement. Notre méthode conserve des performances importantes malgré le niveau de bruit qui augmente en comparaison des autres méthodes.

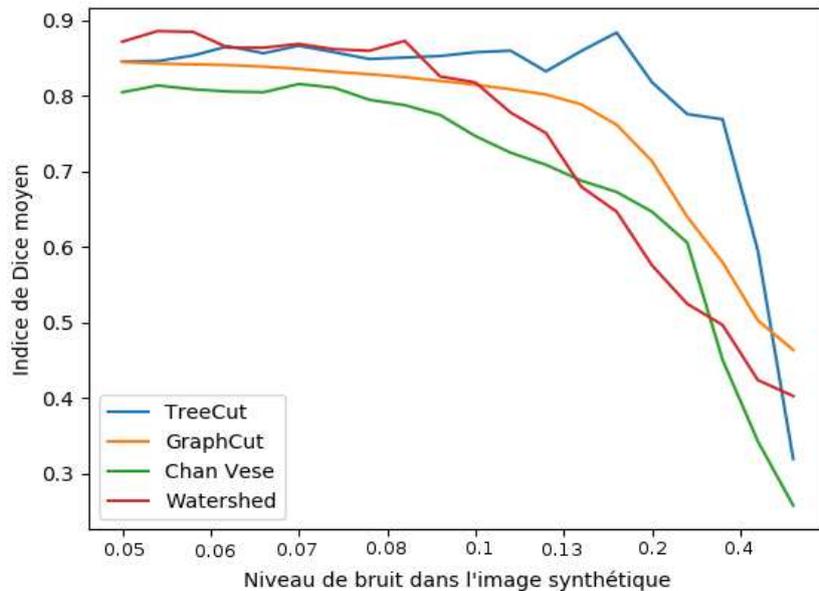


FIGURE 3.22 – Analyse des performances sur des images synthétiques avec des niveaux de bruits variables.

Pour le jeu de donnée 1, les résultats quantitatifs sont cohérents avec les observations (Table 3.1). Le fond est bien segmenté par l'ensemble des méthodes. Le matériau bleu est un peu moins bien segmenté à cause des halos mentionnés ci-dessus. Enfin, le matériau rouge est moins bien segmenté que le orange car dans la base, le rouge est le plus concerné des deux par les défauts que nous pouvons observer dans l'exemple difficile (Figure 3.19).

Pour le jeu de données 2, les observations sont similaires au jeu de données 1. Dans ce cas, le matériau (1) est long et fin ce qui pose problème aux méthodes de comparaison. De plus, les matériaux (1) et (2) ont des niveaux de gris similaires. Nous constatons cela avec des performances plus faibles sur les matériaux (1) et (2) alors que le matériau (3) est correcte-

TABLE 3.1 – Comparaison des résultats sur le jeu de données 1 avec le coefficient de Dice.

Jeu de données 1	Chan-Vese Multiphase	Watershed	GC Position	Tree cut
Arrière Plan	0.86	0.91	0.80	0.95
Matériau 1 (bleu)	0.52	0.82	0.59	0.90
Matériau 2 (orange)	0.36	0.71	0.48	0.79
Matériau 3 (rouge)	0.23	0.73	0.12	0.68
Mean	0.44	0.79	0.52	0.83

ment segmenté.

TABLE 3.2 – Comparaison des résultats sur le jeu de données 2 avec le coefficient de Dice.

Jeu de données 2	Chan-Vese Multiphase	Watershed	GC Position	Tree cut
Arrière Plan	0.95	0.91	0.93	0.93
Matériau 1	0.74	0.24	0.71	0.73
Matériau 2	0.26	0.51	0.29	0.64
Matériau 3	0.77	0.86	0.89	0.93
Mean	0.68	0.63	0.70	0.81

3.4.5 Exemple d'application sur des images d'IRM

Notre approche de segmentation a été développée pour la segmentation de matériaux dans des images de semi-conducteur. Nous proposons de montrer l'application de cette méthode dans le cadre de la segmentation d'images d'IRM. En effet, sur les images d'IRM le postulat de départ qui est qu'une label correspond à un niveau de gris est respecté. Ainsi la

3 Segmentation d'images basée sur les matériaux

matière blanche, la matière grise et le liquide céphalorachidien ont des couleurs différentes dans l'image IRM lorsque l'on choisit la modalité T1.

Pour des raisons pratiques, nous avons récupéré les images d'IRM grâce aux travaux de simulation de BrainWeb [71] [72] [73] [74] [75]. Notre procédé consiste à prendre la segmentation d'un patient comme template et de l'utiliser pour segmenter l'image d'un autre patient (Figure 3.23). Nous avons pris le cas d'un patient différent sans pathologie et le cas d'un patient atteint de sclérose en plaques.

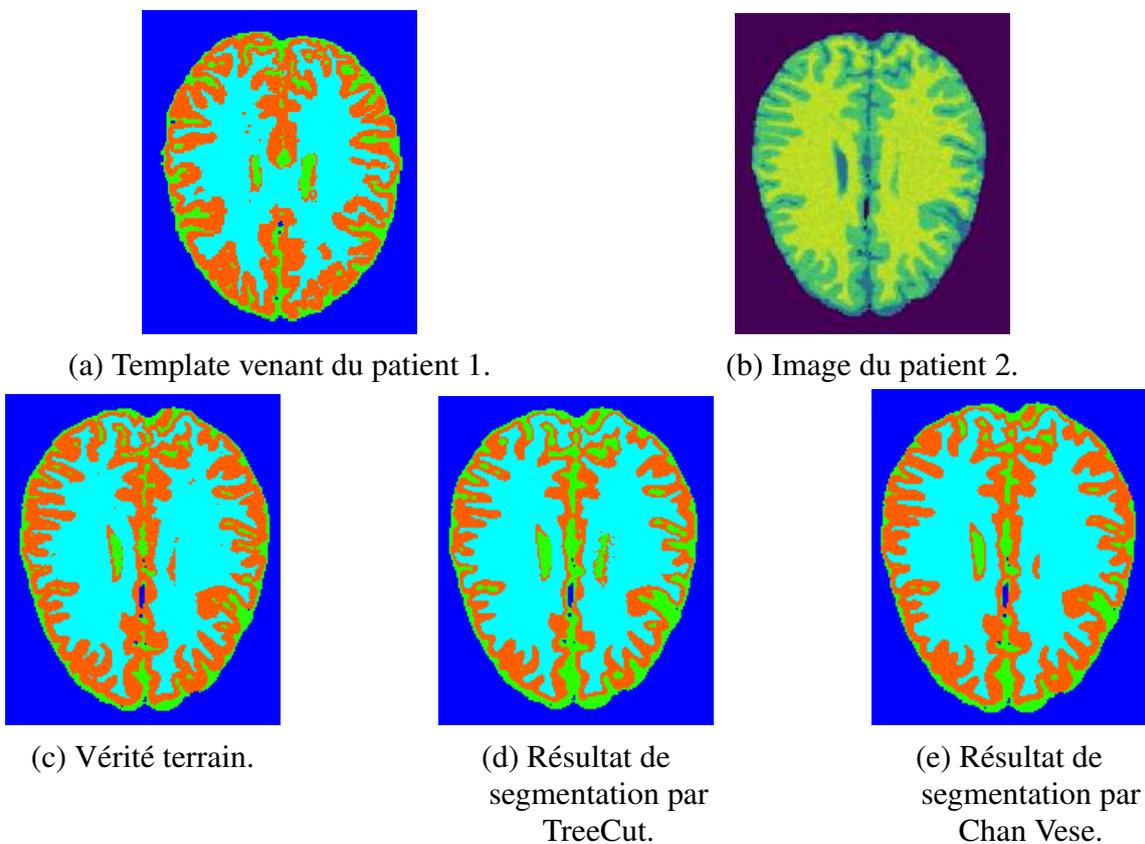


FIGURE 3.23 – Présentation d'exemples de segmentations sur des images d'un patient sans pathologie d'IRM.

Nous avons utilisé les mêmes paramètres que pour les expériences précédentes. Sur les exemples 3.23 3.24, nous pouvons voir les différences entre le TreeCut et Chan Vese. Globalement, le TreeCut sur-segmente le liquide céphalorachidien (vert). Cependant, la géométrie est mieux resti-

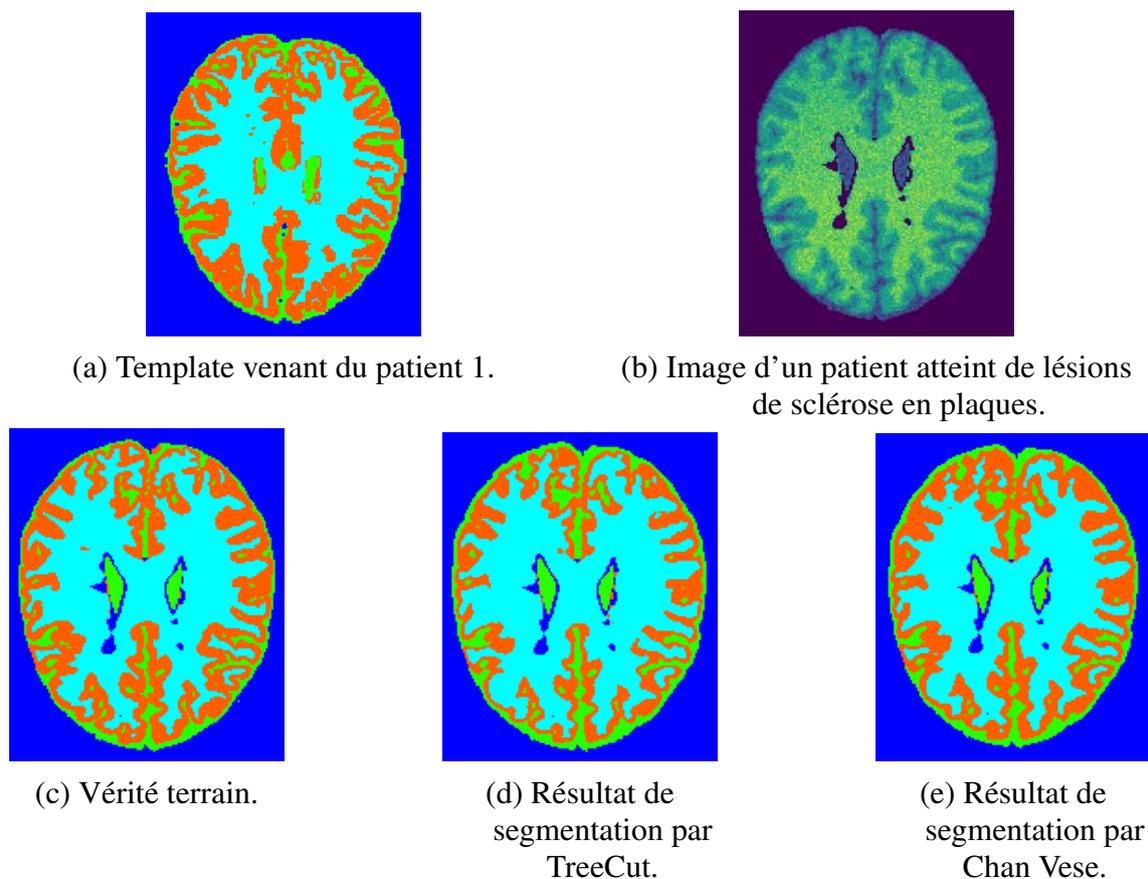


FIGURE 3.24 – Présentation d'exemples de segmentations sur des images d'un patient atteint de lésions de sclérose en plaques d'IRM.

tuée. Quantitativement les résultats, la méthode de Chan-Vese multiphase utilisée (Section 3.4) donne des résultats légèrement supérieurs (Table 3.3).

La principale explication à la faiblesse des résultats est le fait que l'un des présupposés de notre méthode n'est pas respecté. Dans notre cas d'application, le template possède moins d'une dizaine de matériaux donc de nœuds. Dans ce cadre d'images IRM, le template a de nombreuses régions distinctes et donc l'AdF du template a 80 nœuds. De ce fait avec notre méthode, la segmentation résultante cherche à proposer 80 régions distinctes.

Même si les résultats chiffrés sont moins bons qu'avec une autre méthode de notre comparaison, les résultats sur cette application montrent que notre méthode est possiblement adaptable à d'autres applications. De

3 Segmentation d'images basée sur les matériaux

TABLE 3.3 – Comparaison des résultats avec le coefficient de Dice

	Chan-Vese Multiphase Patient 2	Tree cut Patient 2	Chan-Vese Multiphase Lésions	Tree cut Lésions
Liquide céphalorachidien	0.820	0.762	0.843	0.809
Matière grise	0.882	0.841	0.873	0.816
Matière blanche	0.947	0.940	0.945	0.931
Moyenne	0.883	0.847	0.887	0.852

plus, avec l'AdF, il est possible d'effectuer la mise en correspondance en 3D [59] ainsi que de fusionner les modalités T1, T2 et PD [58]. Dans le contexte de la segmentation en 3D, notre approche sera même certainement meilleure car le fait de travailler sur une coupe augmente le nombre de nœuds de façon importante sur le template comparativement au nombre de pixels. Dans le cas 3D, il est probable que le nombre de nœuds soit similaire mais le nombre de voxels sera lui beaucoup plus important.

3.5 Conclusion

Notre problématique de segmentation de matériaux vise à utiliser un template pour faire une segmentation et une mise en correspondance des matériaux (régions) du template sur une image réelle. Le template étant une version idéalisée de l'image, nous avons mis en place une structure de données adaptée pour le représenter. L'AdF permet d'avoir une structure qui représente de manière hiérarchique les différents matériaux de notre template. Nous avons vérifié que l'image réelle et le template ont des structures similaires. Ainsi nous avons développé une méthode de segmentation utilisant l'AdF et le template pour produire une segmentation avec une mise en correspondance. Cette méthode s'inspire du Graph Cut pour minimiser le coût de la segmentation. Grâce à cette méthode, nous pouvons aussi

3.5 Conclusion

étendre les caractéristiques prises en compte lors de la minimisation. Nous avons comparé cette méthode avec des méthodes de l'état de l'art sur des images simulées et deux jeux de données réelles. Enfin, nous avons montré une utilisation possible à la segmentation d'images IRM.

3 Segmentation d'images basée sur les matériaux

4 Application au contexte industriel

Notre application est l'utilisation des algorithmes présentés dans ce manuscrit pour l'analyse d'images TEM de Pollen Metrology. Dans la pratique, le logiciel met à disposition différents outils pour automatiser les mesures des utilisateurs. Dans le cadre des travaux de cette thèse, l'utilisateur possède un jeu d'images qu'il souhaite mesurer ainsi que le *design* de l'objet fabriqué (Figure 4.1). Sur le *design*, il définit les mesures qu'il souhaite réaliser. Dans un deuxième temps, l'utilisateur doit extraire de ses images quelques exemples de l'objet réel défini par le *design*. Dans la suite, nous ferons la distinction entre trois cas d'utilisation différents en fonction de la quantité de données annotées par l'utilisateur.

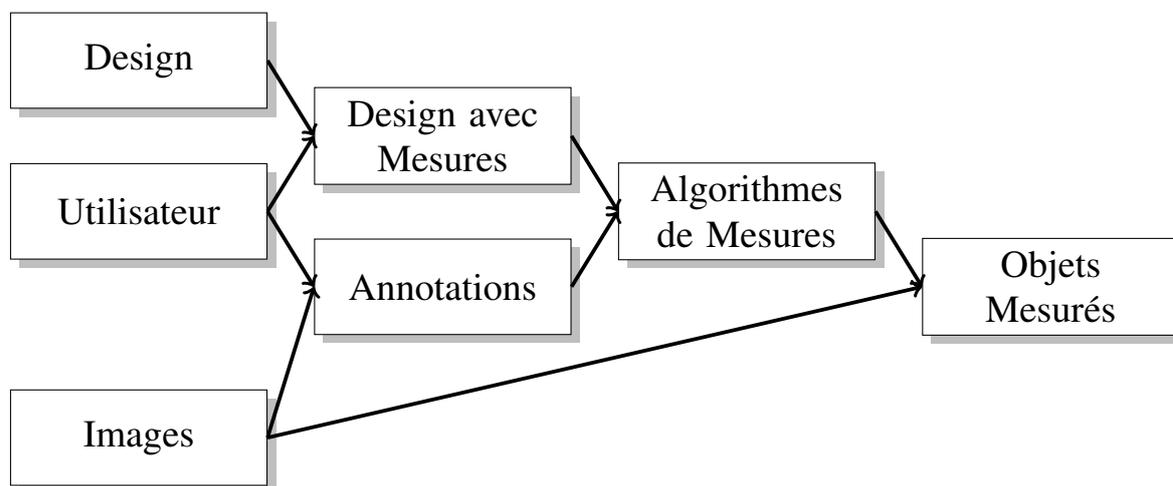


FIGURE 4.1 – Procédure suivie par les utilisateurs pour faire la mesure d'objets.

Ce chapitre présentera tout d'abord le pipeline mis en place avec une

description des différentes parties ainsi que les contributions de cette thèse. Il se concentrera ensuite sur l'aspect détection d'objets qui représente une partie importante du pipeline pour permettre la sélection des objets des images pour la segmentation. Puis la troisième partie se concentrera sur la segmentation afin d'en extraire les mesures voulues.

4.1 Pipeline pour la mesure automatique d'objets

Les principales difficultés pour établir un pipeline d'algorithmes pour la mesure automatique d'objets de semi-conducteurs sont la variabilité des cas d'utilisation ainsi que la faible quantité de données labellisées à disposition. Ce pipeline vise à être découpé en un maximum de parties indépendantes qui réalisent des tâches réduites afin d'être en mesure de modifier chaque partie de manière indépendante pour conserver une flexibilité. Enfin, le pipeline présenté a été développé dans le cadre de l'analyse d'images TEM mais vise à être utilisé de manière plus large pour d'autres types d'images (SEM, AFM, image de chimie fine, ...) en dehors du cadre des travaux de cette thèse.

Le pipeline se décompose en 4 étapes (Figure 4.2).

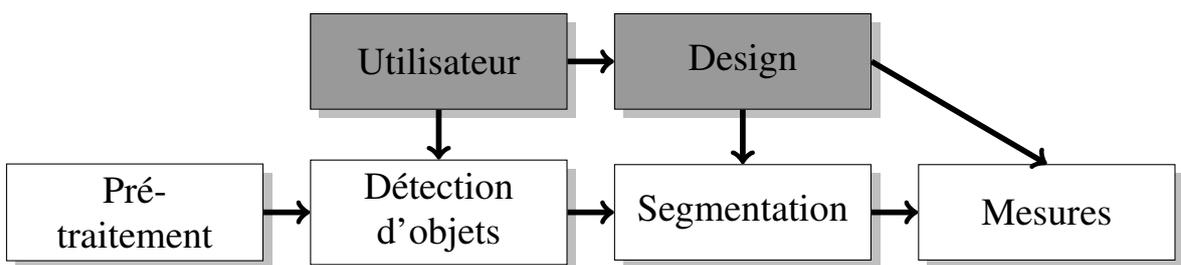


FIGURE 4.2 – Pipeline d'algorithmes pour la mesure automatique d'objets. Les 4 étapes sont représentées en blanc.

La première étape correspond à du pré-traitement et généralement à du filtrage. Pour cette étape, la contribution principale de la thèse est l'apport

4.1 Pipeline pour la mesure automatique d'objets

de la connaissance des caractéristiques du bruit présent dans la microscopie TEM (Section 2.3). Nous utilisons un filtre adapté au bruit de Poisson. Le but général de cette étape du pipeline est d'améliorer la qualité des images. Dans l'utilisation générale du logiciel, il est mis à disposition d'autres filtres pour l'utilisateur pour des usages sur des images provenant de différentes microscopies ne rentrant pas dans le cadre des travaux de cette thèse. De plus, l'utilisateur exige de pouvoir ajuster les paramètres de ce filtre avec un retour visuel. Nous proposons tout de même une évaluation automatique des paramètres.

La deuxième étape du pipeline correspond à la détection d'objets. Les images que nous avons à analyser contiennent plusieurs éléments à mesurer (Figure 4.3). Les objets sont assez semblables sur une même image mais peuvent varier de manière plus importante au fil des images. À la vue de la variabilité des différents objets, nous avons choisi d'imposer à l'utilisateur de définir ces objets à détecter. Pour ce faire nous avons deux familles d'algorithmes : le template matching et les méthodes par apprentissage. Nous avons combiné ces méthodes pour proposer trois scénarios d'application différents (Figure 4.4).

Le premier, l'utilisateur a peu d'images à analyser (moins de 10) et ne souhaite pas faire beaucoup d'annotations. Dans ce cas, nous proposons d'utiliser des outils de template matching. L'utilisateur sélectionne un exemple et peut avoir des détections d'objets sur ces images. Si le détecteur ne fonctionne plus parce que ses images varient trop, il sélectionne un nouvel objet pour créer un autre détecteur. Le deuxième scénario, l'utilisateur a plus d'images à analyser et est prêt à en annoter un plus grand nombre. Dans ce cas, nous utilisons un détecteur d'objets par apprentissage. Par la suite l'utilisateur est susceptible d'ajouter plus d'exemples (manuellement ou via le détecteur) à sa base pour ré-entraîner le modèle et améliorer la robustesse. Enfin, la dernière approche est une combinaison des deux précédentes. L'utilisateur ne va sélectionner que quelques objets et utiliser les détections du template matching pour créer sa base d'apprentissage pour la détection par apprentissage.

4 Application au contexte industriel

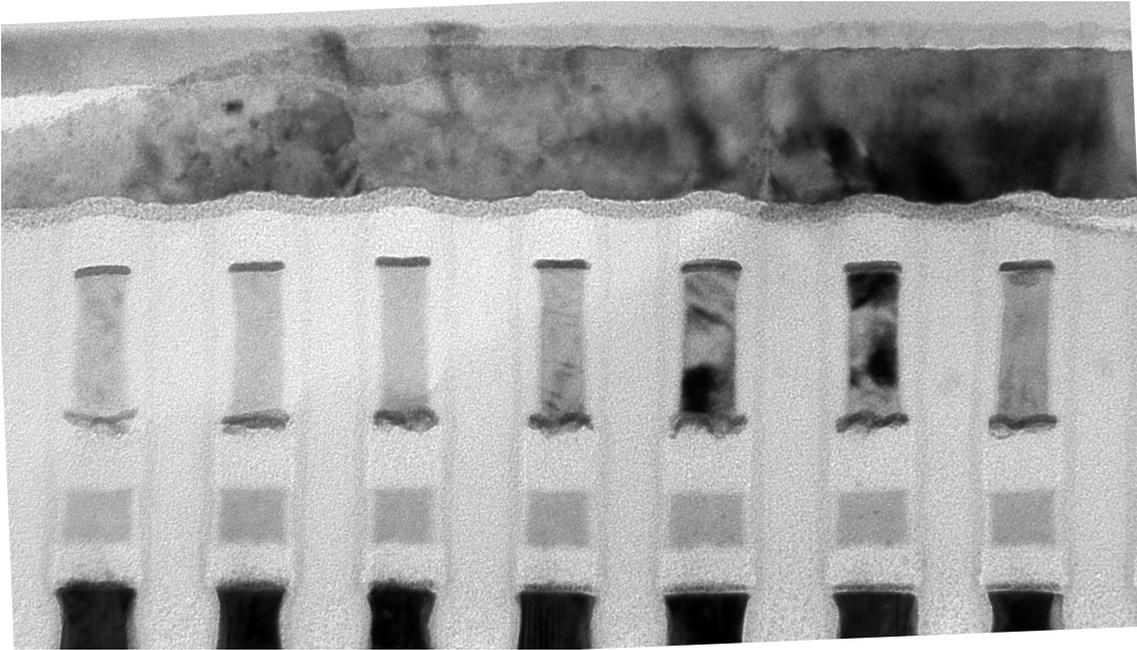


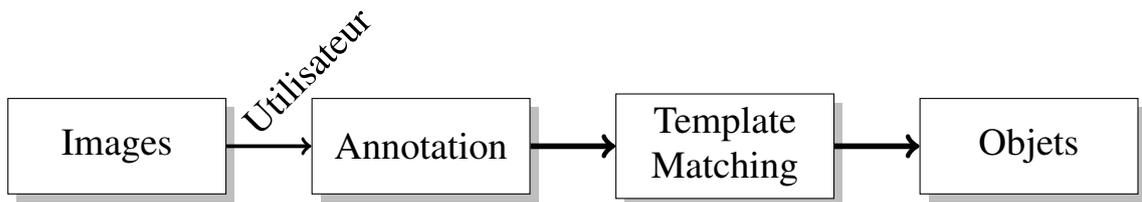
FIGURE 4.3 – Une image d'exemple contenant plusieurs objets à mesurer. Cette image provient de [76].

La troisième étape est une étape de segmentation. Dans notre cas, la segmentation s'intéresse aux différents matériaux. Le but de cette étape est d'extraire des informations de contours ou de régions qui sont cohérentes avec les mesures de l'utilisateur. Cette thèse apporte évidemment la contribution de la segmentation de matériaux des différents objets par TreeCut (Section 3.3).

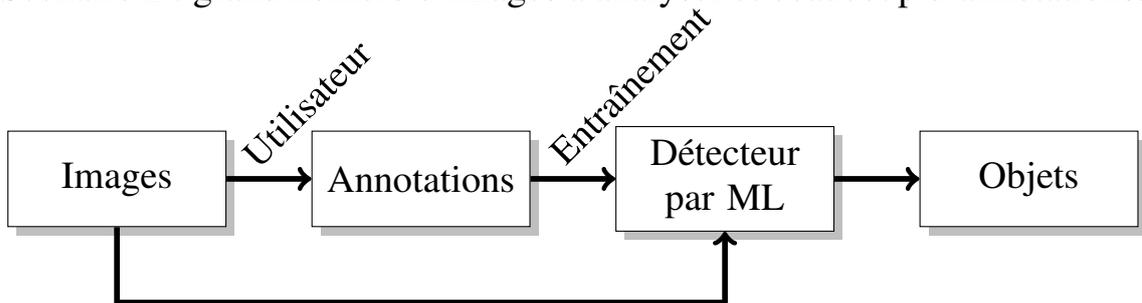
Enfin la dernière étape correspond à la mesure finale des objets. La difficulté est la diversité des mesures que veulent effectuer les utilisateurs. Dans un premier temps nous avons d'abord considéré des mesures dimensionnelles ponctuelles (Figure 4.5). Parmi les mesures utiles, nous avons des métriques de cristallographie, des informations topologiques, des aires, périmètres et des descripteurs de formes. Cette thèse propose d'utiliser les outils de segmentation proposés en section 3.3 ainsi que la base créée par l'utilisateur pour améliorer les mesures. Dans le logiciel de Pollen, sont disponibles plusieurs outils pour faire des mesures, ils sont cependant initialisés sur l'image par l'utilisateur. Le but de cette partie est d'améliorer

4.1 Pipeline pour la mesure automatique d'objets

Scénario 1 : petit nombre d'images à analyser et peu d'annotations.



Scénario 2 : grand nombre d'images à analyser et beaucoup d'annotations.



Scénario 3 : grand nombre d'images à analyser et beaucoup d'annotations.

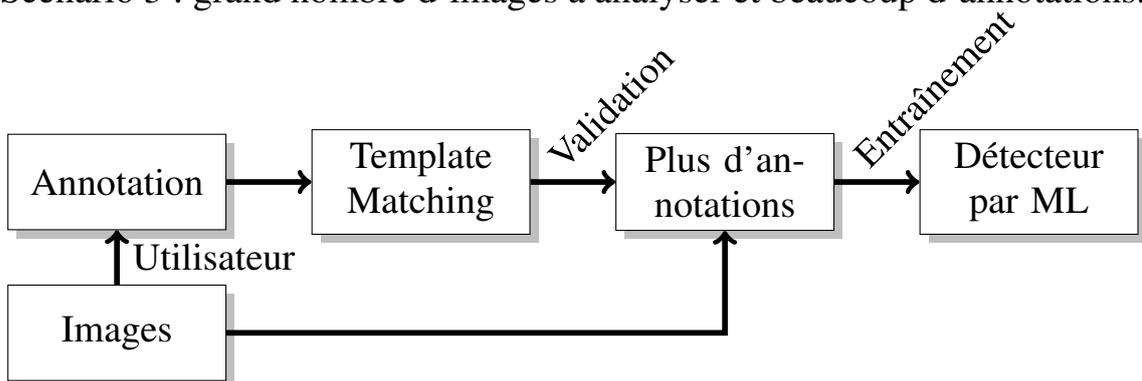
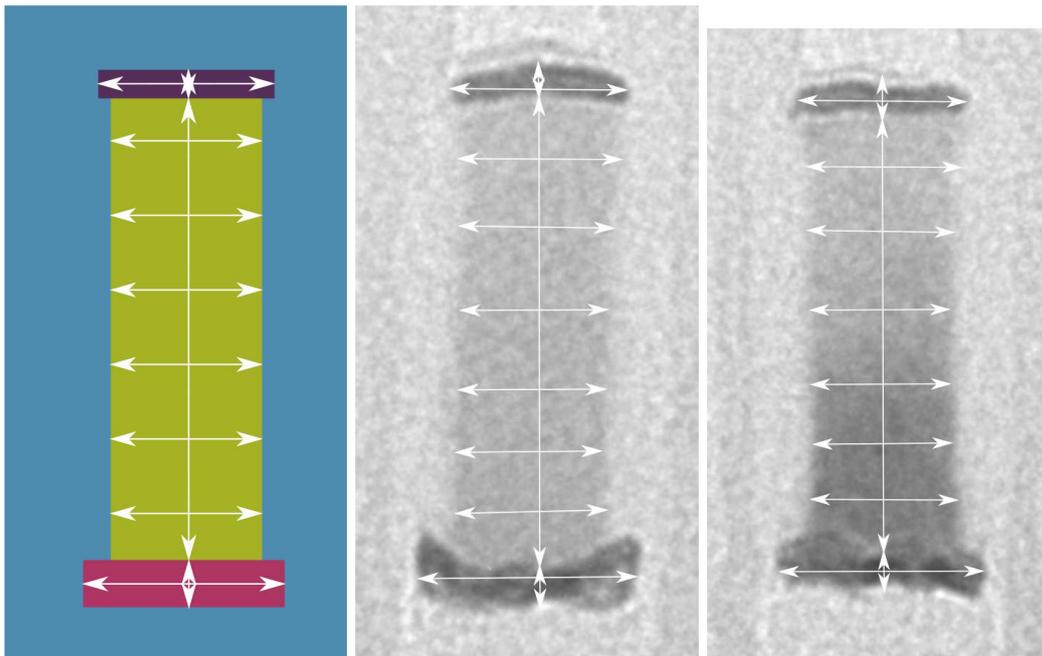


FIGURE 4.4 – Les différents modes pour la détection d'objets.

le placement initial de ces outils sur les objets trouvés automatiquement.

4 Application au contexte industriel



(a) Un exemple de *design*. (b) Un exemple d'objet. (c) Un autre exemple d'objet.

FIGURE 4.5 – Un *design* et deux exemples d'objet avec leur mesure.

4.2 Détection d'objets

4.2.1 Présentation du contexte

Dans le pipeline précédemment présenté, une des étapes importante est la détection d'objets. Elle permet de localiser de manière automatique les objets dans l'image afin de pouvoir faire les mesures dimensionnelles. Nous ne pouvons pas définir a priori de modèle du fait de la variabilité des objets. Nous nous sommes donc orientés vers des méthodes par apprentissage afin de pouvoir adapter la détection aux objets.

Notre principale contrainte est la taille de la base de données. Selon les cas, nous avons à notre disposition de un à une centaine d'objets disponibles, tout n'étant pas annoté.

Tout d'abord, nous étudierons des méthodes de template matching qui permettent de faire la détection des objets avec un seul exemple lorsque

la variabilité des objets sur l'image est faible. Typiquement, nous pouvons avec cette méthode détecter les objets présents dans une même image.

4.2.2 Template matching

L'idée est de sélectionner une région d'une image et de retrouver des instances similaires dans l'image (Figure 4.6). La translation est la transformée principale que nous voulons retrouver, la rotation et la mise à l'échelle ne seront donc pas traités ici. Il existe deux grandes familles de méthodes de template matching.

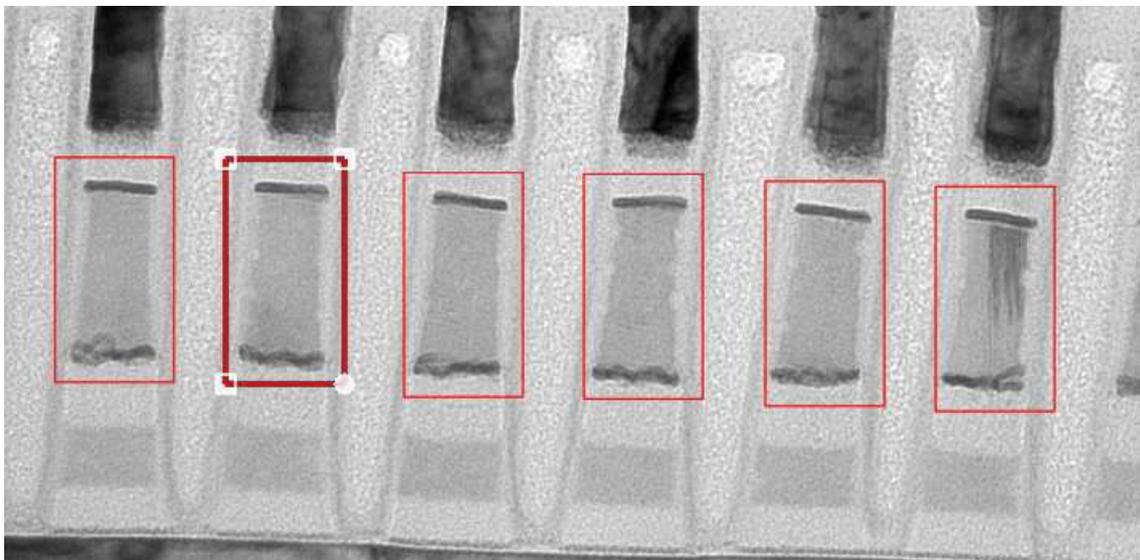


FIGURE 4.6 – Exemple de template matching. La boîte en trait plus épais est celle qui est utilisée comme template.

La première famille se base sur des points d'intérêts. Ils sont extraits à la fois sur le template et sur l'image. Leur mise en correspondance est ensuite réalisée [44] [54] [49]. Ces méthodes gèrent de façon efficace la transformée affine. Dans notre cas, nous avons fait le choix d'écarter ces méthodes car les points d'intérêts extraits ne sont pas suffisamment robustes dans un cadre générique.

La deuxième catégorie se base sur l'image. Elles reposent sur le parcours de l'image cible et la comparaison de chaque partie de l'image avec le tem-

4 Application au contexte industriel

plate pour en extraire un score. Ces approches ont été étendues à la transformée affine. L'une des plus simple est sans doute la corrélation de phase [77]. Mais plusieurs approches plus récentes [78] [79] proposent d'étendre la gamme de transformation ainsi que la robustesse. Nous nous sommes intéressés aux techniques basées sur la corrélation de phase (Équation 4.1) et sur la corrélation croisée normalisée (Équation 4.2) car elles sont simples à implémenter pour la translation.

$$R = F^{-1} \left(\frac{F(I) \circ F(T_e)^*}{|F(I) \circ F(T_e)^*|} \right) \quad (4.1)$$

- R : la corrélation de phase
- $F(I)$: la transformée de Fourier de l'image I
- $F^{-1}(I)$: la transformée de Fourier inverse
- X^* : le conjugué complexe d'une matrice X
- $|X|$: le module de X élément par élément
- \circ : le produit de Hadamard, qui est le produit élément par élément des matrices
- I : l'image
- T_e : le template dont les bordures ont été étendues avec des zéros jusqu'à la taille de l'image

$$R(x, y) = \frac{\sum_{x_T, y_T} (T(x_T, y_T) \cdot I(x + x_T, y + y_T))}{\sqrt{\sum_{x_T, y_T} T(x_T, y_T)^2 \cdot \sum_{x_T, y_T} I(x + x_T, y + y_T)^2}} \quad (4.2)$$

- R : la corrélation croisée normalisée
- T : le template
- I : l'image
- x_T, y_T : les coordonnées du template
- x, y : les coordonnées de l'image

À partir d'un exemple donné par l'utilisateur, l'algorithme que nous avons mis en place (Figure 4.7) extrait tous les objets présents dans une

même image. Une étape de maximum local sélectionne localement les instances sur une même image sans superposition des objets.

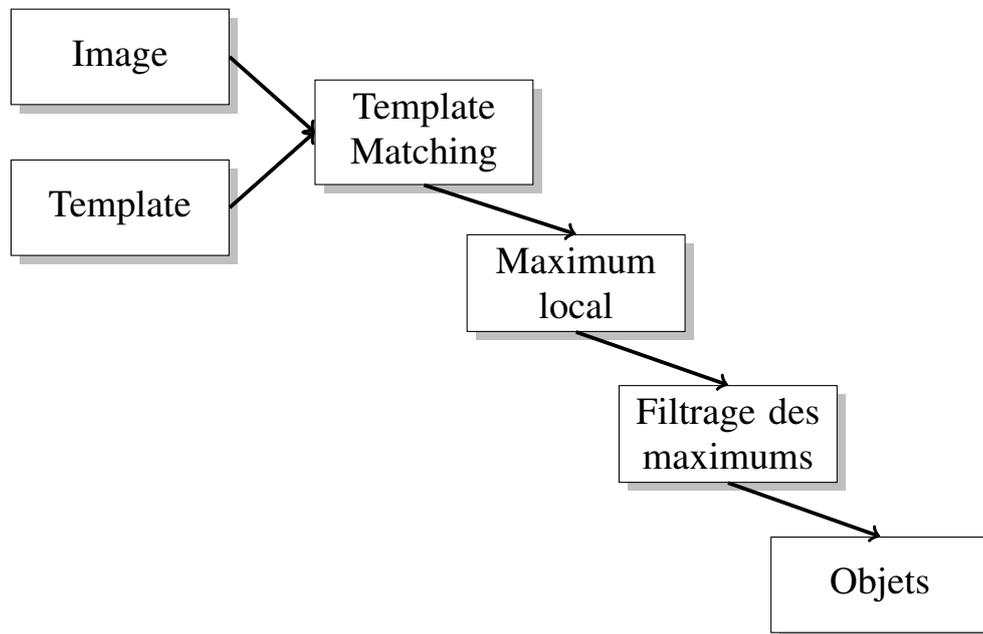


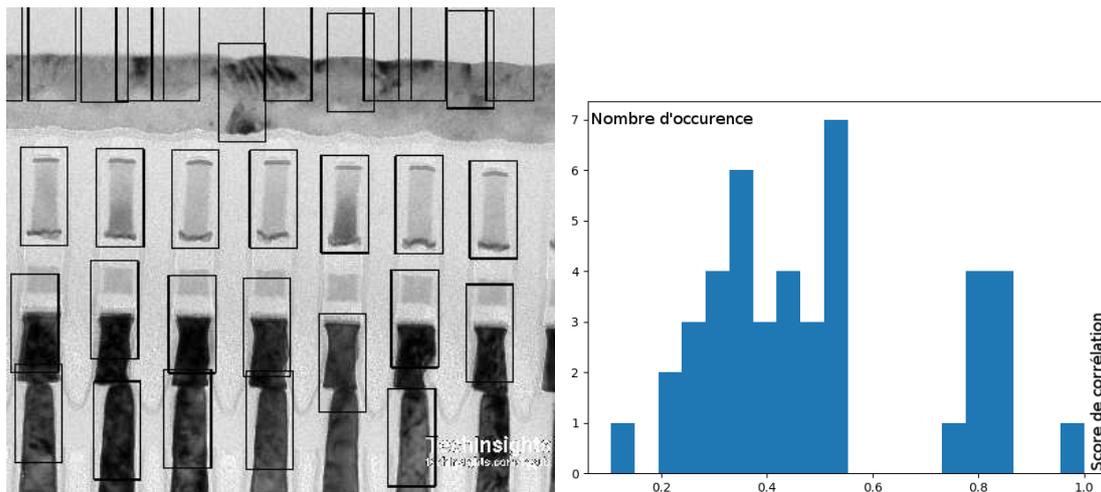
FIGURE 4.7 – Présentation de l’algorithme de template matching

Le maximum local est une recherche de maximum dans un voisinage défini par la taille du template. Cette recherche peut être longue car la complexité de l’approche naïve est en $O(n \times m)$ avec n la taille de l’image et m la taille du template. Pour accélérer la procédure, nous avons utilisé une procédure similaire à [80], utilisant un histogramme local récurrent, pour réduire la complexité à $O(n)$.

Afin d’éliminer les faux positifs de la détection produits par le maximum local, nous avons une étape de filtrage en fonction du score. La distribution des scores est multimodale avec le mode supérieur contenant nos objets (Figure 4.8). Nous avons donc approché le problème par partitionnement (clustering), avec le nombre de modes inconnu. Avec un clustering comme Mean Shift [81], nous pouvons séparer les différents modes. De cette façon, les objets recherchés font partis du cluster avec le score moyen le plus élevé, les autres clusters ne correspondant pas à nos objets.

Cette méthode sera appelé Template Matching (TM) dans la suite.

4 Application au contexte industriel



(a) Exemple de détections en sortie de (b) Histogramme des scores de corrélation. maximum local.

FIGURE 4.8 – Exemple des détections successives sur plusieurs tailles d'images.

4.2.3 Méthode d'apprentissage classique

Pour améliorer l'étape de détection, nous avons étudié les méthodes d'apprentissage. Nos objets ne subissent une transformation qu'en translation, des méthodes à base de fenêtres glissantes nous paraissent adaptées. Nous nous sommes tout d'abord intéressés aux classifieurs en cascades issus de la détection de visages [82]. Expérimentalement, l'entraînement d'un classifieur en cascades s'est avéré compliqué à ajuster pour obtenir des performances satisfaisantes. Dans une idée similaire, [83] propose la détection de personnes dans des images naturelles (Figure 4.9). Ce choix nous est apparu comme plus pertinent car le Support Vector Machine (SVM) est un classifieur fort et nous avons expérimentalement entraîné un modèle avec facilité.

La méthode propose d'extraire des descripteurs basés sur l'orientation des gradients, les descripteurs Histogram of Oriented Gradients (HOG). Sur une zone locale, l'histogramme de l'orientation des gradients est calculée. Cet histogramme, une fois normalisé, sert de descripteur pour la zone. En regroupant plusieurs zones, nous obtenons un vecteur de descripteurs

pour la zone et donc potentiellement pour un objet.

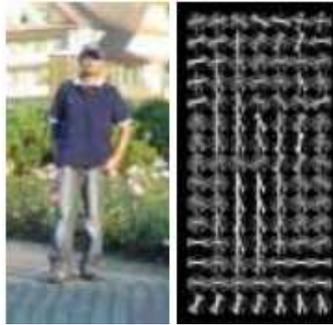


FIGURE 4.9 – Exemple d'HOG sur une image. Le visuel provient de [83].

Les descripteurs HOG sont extraits sur l'image. Ainsi, l'image est découpée en blocs de descripteurs. Ensuite nous utilisons une fenêtre glissante pour parcourir les objets potentiels de l'image. Chaque position est classifiée par un SVM comme étant un objet ou du fond. Une étape de Non Maximum Suppression (NMS) est enfin mise en place pour discriminer les propositions qui se superposent (Figure 4.10).

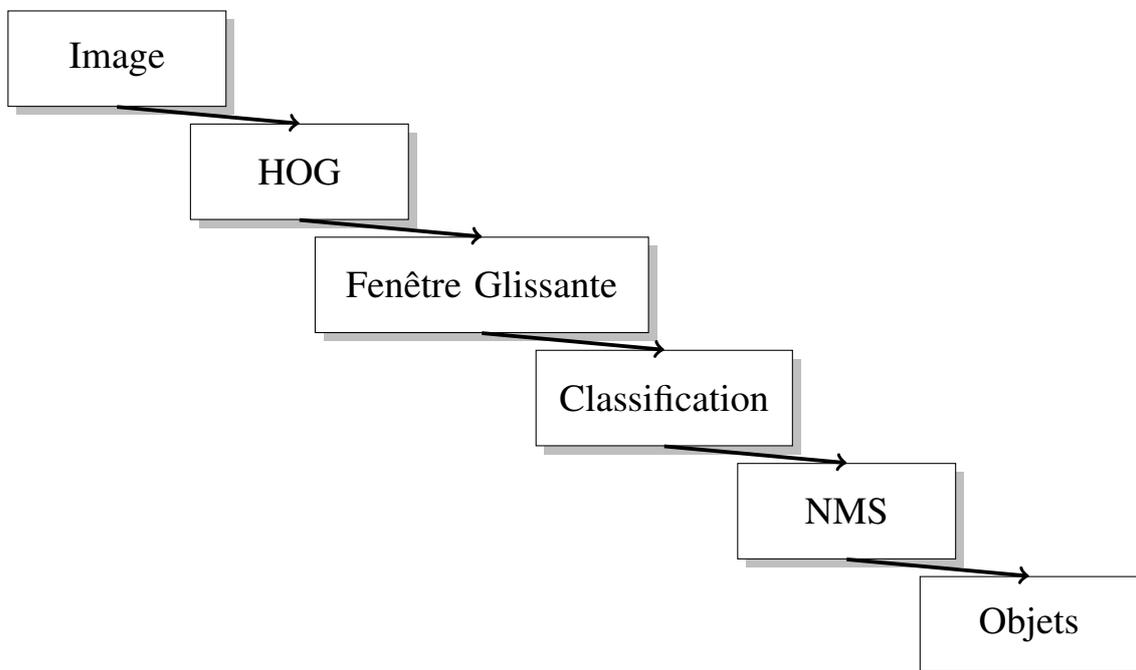


FIGURE 4.10 – Présentation de l'algorithme de détection d'objets avec le HOG.

4 Application au contexte industriel

Pour les descripteurs HOG, nous avons utilisé les paramètres proposés dans [83]. Nous avons fait varier les paramètres autour de ce point de référence et les résultats varient peu (moins de 5% sur la précision et le rappel). Pour le classifieur, nous avons utilisé un SVM linéaire car nous avons peu de données d'entraînement (environ 10 objets pour une centaine d'exemples de fond). Avec si peu de données, il est préférable de conserver un modèle simple pour limiter le sur-apprentissage.

La procédure pour le NMS est assez simple. L'idée est de supprimer les nombreuses boîtes détectées autour d'un seul objet. Pour ce faire, nous sélectionnons la détection avec le plus haut niveau de confiance puis nous supprimons toutes les détections qui se superposent à cette première. Dans notre cas, le critère de superposition est l'intersection sur l'union entre la boîte avec le plus haut niveau de confiance et les autres. Nous répétons cette opération jusqu'à l'épuisement des propositions de l'algorithme de détection. Finalement, à chaque opération, la détection la plus confiante est conservée comme proposition finale.

Les résultats seront discutés plus loin en comparaison des autres méthodes proposées. Cette méthode sera nommée HOG dans la suite.

4.2.4 Méthode d'apprentissage profond

État de l'art

Nous avons étudié plusieurs méthodes de l'état de l'art sur la détection d'objet par apprentissage profond. La première, le Region with CNN features (R-CNN) [84], est assez similaire à la méthode que nous avons utilisé dans l'apprentissage classique. Un classifieur est entraîné à discriminer des objets du fond avec des descripteurs Convolutionnal Neural Network (CNN). Le principal défaut est que les objets que l'on veut détecter doivent suivre certains aprioris forts. Nous détaillons plus loin l'approche.

La deuxième méthode fait partie de la famille des algorithmes entraînant un module de proposition de régions [85]. Ces approches utilisent

une base de CNN qui sert à extraire des descripteurs. Puis sur cette base, deux modules de NN sont branchés, le premier fait de la proposition de régions d'intérêt et la deuxième de la classification. L'intérêt principal est que la forme des objets que l'on veut détecter est plus souple que précédemment car elle est apprise. Cependant, la quantité de données requises pour l'apprentissage est beaucoup plus importante.

La troisième méthode fait partie des méthodes basées sur la classification de régions de l'image [86]. Ces méthodes cherchent à découper l'image en différentes régions qui sont responsables de la détection d'objets. Ensuite, une étape de régression est mise en place afin d'affiner la position de l'objet. Ici, aussi l'a priori de formes n'est pas très fort et dépend grandement des données d'entraînement. La quantité de données pour l'apprentissage reste lui aussi conséquent.

Enfin, dans la littérature, une autre catégorie de méthode est présente, celle du One Shot ou Few Shot Learning [87]. Cette problématique repose sur le fait d'avoir une base d'apprentissage conséquente pour l'entraînement avec une grande variété de classes. Une fois entraîné, le modèle est évalué en présentant quelques exemples d'une nouvelle classe inconnue puis doit être capable de retrouver cette nouvelle classe sur des images de test. Ainsi de cette façon, le modèle se comporterait de façon similaire à notre proposition de template matching. Cependant, la robustesse des résultats de l'état de l'art reste en dessous de nos exigences pour le moment et la quantité de données totale à notre disposition pour entraîner un tel modèle est insuffisante en comparaison des bases utilisées dans l'état de l'art.

Pour ces raisons, nous avons choisi d'étudier plus en détails la première méthode, le R-CNN, puis avons décidé de développer notre propre méthode.

R-CNN

L'algorithme de R-CNN (Figure 4.11) est composé de trois étapes : la proposition de régions, le calcul de descripteurs et enfin la classification. La méthode nous paraît donc en adéquation avec notre problématique, l'apprentissage étant au final réduit à de la classification de la même façon que dans la méthode classique (Sous Section 4.2.3). L'utilisation de CNN pré-entraîné pour faire de la classification montre de belles performances sans modification avec un classifieur SVM [88].

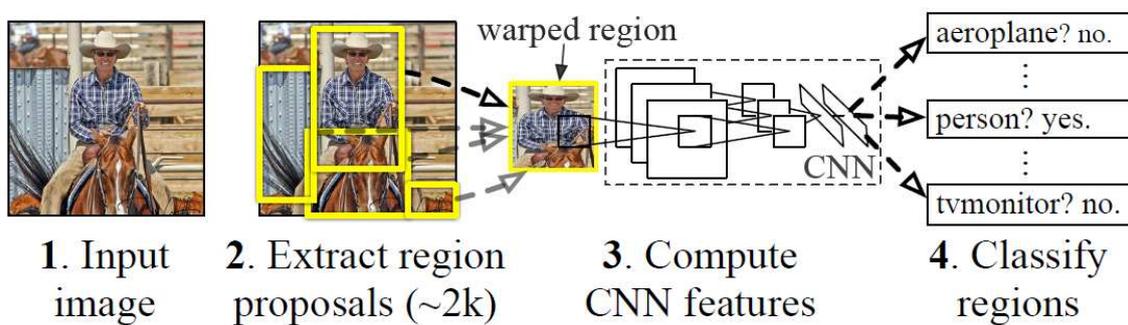


FIGURE 4.11 – Description de l'algorithme R-CNN. L'image provient de [84].

De part la construction de l'algorithme, nous avons choisi un CNN de la littérature [89] pré-entraîné sur ImageNet [90] pour calculer les descripteurs ainsi qu'un SVM linéaire pour le classifieur. Il reste donc la partie proposition de régions. La proposition de régions consiste en un algorithme, qui, à partir d'une image propose des régions rectangulaires contenant potentiellement un objet. La définition d'un objet dans ce cas est vague mais vise principalement les objets de la vie courante. R-CNN propose d'utiliser Selective Search [91]. Cette méthode repose sur une sur-segmentation de l'image pour ensuite fusionner les régions de manière itérative selon un critère (Équation 4.3). Un objet est donc une région de l'image qui a un niveau de gris homogène, une texture homogène, qui rassemble des petites régions entre elles ainsi qu'une forme globalement convexe. Cette définition n'est pas compatible avec la définition de tous nos objets (classe "fin" Figure 4.17). Une autre proposition, Edge Boxes

[92] propose une méthode similaire en travaillant sur les contours extraits pour les organiser afin de former des régions rectangulaires. Les a priori de formes ne sont toujours pas en accord avec la totalité de nos objets. Le R-CNN sera utilisé en comparaison d'autres méthodes sur des objets appropriés pour comparer des méthodes d'apprentissage classiques avec des méthodes d'apprentissage profond.

$$s(r_1, r_2) = s_{couleur}(r_1, r_2) + s_{texture}(r_1, r_2) + s_{taille}(r_1, r_2) + s_{remplissage}(r_1, r_2) \quad (4.3)$$

- $s(r_1, r_2)$: indice de similarité entre deux régions
- $s_{couleur}$: similarité de niveau de gris (intersection des histogrammes de niveaux de gris)
- $s_{texture}$: similarité de texture (intersection des histogrammes des orientations de SIFT)
- s_{taille} : similarité de taille (favorise la fusion de régions de tailles similaires et en priorité les petites)
- $s_{remplissage}$: similarité pour le remplissage (favorise la fusion des régions incluses dans une autre)

Sliding CNN¹

Dérivée des méthodes précédentes, nous avons pris le parti de développer une autre approche. Dans un premier temps, nous avons implémenté le premier algorithme en remplaçant le HOG par un CNN pré-entraîné (Figure 4.12). Cette approche naïve est particulièrement lente car les descripteurs de CNN sont extraits sur chaque vignette. Nous pourrions aussi calculer les descripteurs avant de faire la fenêtre glissante mais il nous faut calculer la carte de descripteurs de nombreuses fois pour conserver une précision au pixel. En effet, la carte des descripteurs réduit la taille par un facteur 8 ou 16 selon les architectures de CNN.

1. Ces travaux sont le fruit d'une collaboration avec Aurélien Laviron, Ingénieur Recherche chez Pollen Metrology.

4 Application au contexte industriel

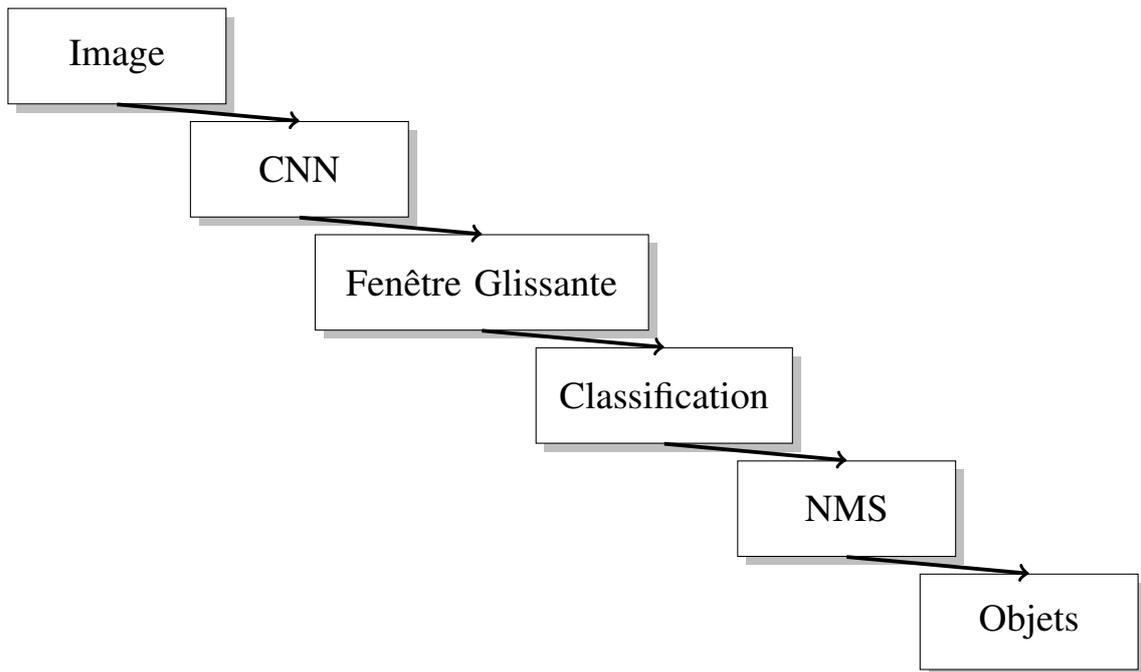


FIGURE 4.12 – Présentation de l'algorithme "Naïve CNN" pour la détection d'objets.

Notre motivation à conserver une approche par fenêtre glissante est la taille de notre base d'apprentissage qui est en moyenne de quelques dizaines d'objets. Nous avons constaté que l'approche classique permettait d'avoir suffisamment de robustesse dans notre contexte. Nous cherchons donc à reproduire le comportement avec des descripteurs CNN de manière efficace.

L'inconvénient de l'approche naïve est le calcul des descripteurs sur chaque fenêtre, ce qui est très coûteux. Il existe dans la littérature le Region Of Interest (ROI) Pooling aussi appelé ROI Align [93] [94]. Le principe du *Pooling* est de réduire la taille de l'image en entrée de la couche en effectuant une opération locale de réduction (ici, une interpolation bilinéaire) sur une fenêtre généralement de 2×2 . Une couche dans un NN est une étape de calcul. Ces étapes peuvent prendre plusieurs formes comme des convolutions, des combinaisons linéaires ou encore l'application de fonctions. L'idée est de calculer les descripteurs sur l'image globale puis d'interpoler les valeurs en fonction de la région que l'on veut extraire. Nous calculons

ensuite une seule carte de descripteurs et la fenêtre glissante est utilisée sur cette unique carte.

L'approche reste limitée car la précision du placement varie beaucoup selon le facteur de réduction du NN. En effet, en fonction du nombre de pooling dans le NN, la carte de descripteurs a une taille plus petite que l'image d'origine. Ainsi un vecteur de descripteurs à une position donnée nous renseigne sur une portion plus large de l'image. De plus, le temps d'exécution est assez long car la plupart de nos clients n'ont pas accès à un Graphical Processing Unit (GPU). Nous avons donc conçu un algorithme utilisant des détections à plusieurs tailles d'images afin d'accélérer le processus. Cette approche s'inspire des approches de Cascade classifier [82]. L'idée est de prendre un CNN pré-entraîné pour extraire les descripteurs. Nous ajoutons ensuite deux couches de pooling pour réduire la taille par quatre afin d'obtenir deux cartes supplémentaires.

Avec cet algorithme (Figure 4.13), que nous appellerons Sliding CNN, nous effectuons une première détection sur une carte à l'échelle 1/32 (CNN + Max Pooling \times 2). Ainsi, la position d'un objet proposé à une erreur maximale de 16 pixels. L'avantage est la vitesse, nous pouvons parcourir l'image 16 fois plus rapidement que sur une carte à l'échelle 1/8 (CNN uniquement). Cette première étape met en évidence les zones contenant des objets avec une précision grossière (Figure 4.14). Nous effectuons ensuite une recherche sur l'échelle 1/16 (CNN + Max Pooling \times 1) en ne parcourant que les régions classées comme contenant un objet dans l'étape précédente. Nous répétons l'opération une troisième fois pour avoir la détection finale à l'échelle 1/8.

Finalement, pour l'architecture, il nous reste à choisir le CNN pour l'extraction des descripteurs. Pour cette partie, le choix est large. Nous avons à notre disposition VGG [95], qui est parmi les plus anciens et propose une architecture très simple. Le plus utilisé de la littérature est certainement ResNet [96] qui propose une architecture très profonde. Une amélioration de ce dernier, DenseNet [97], propose une architecture moins profonde mais qui reprend les blocs de bases de ResNet. Enfin, nous avons aussi identi-

4 Application au contexte industriel

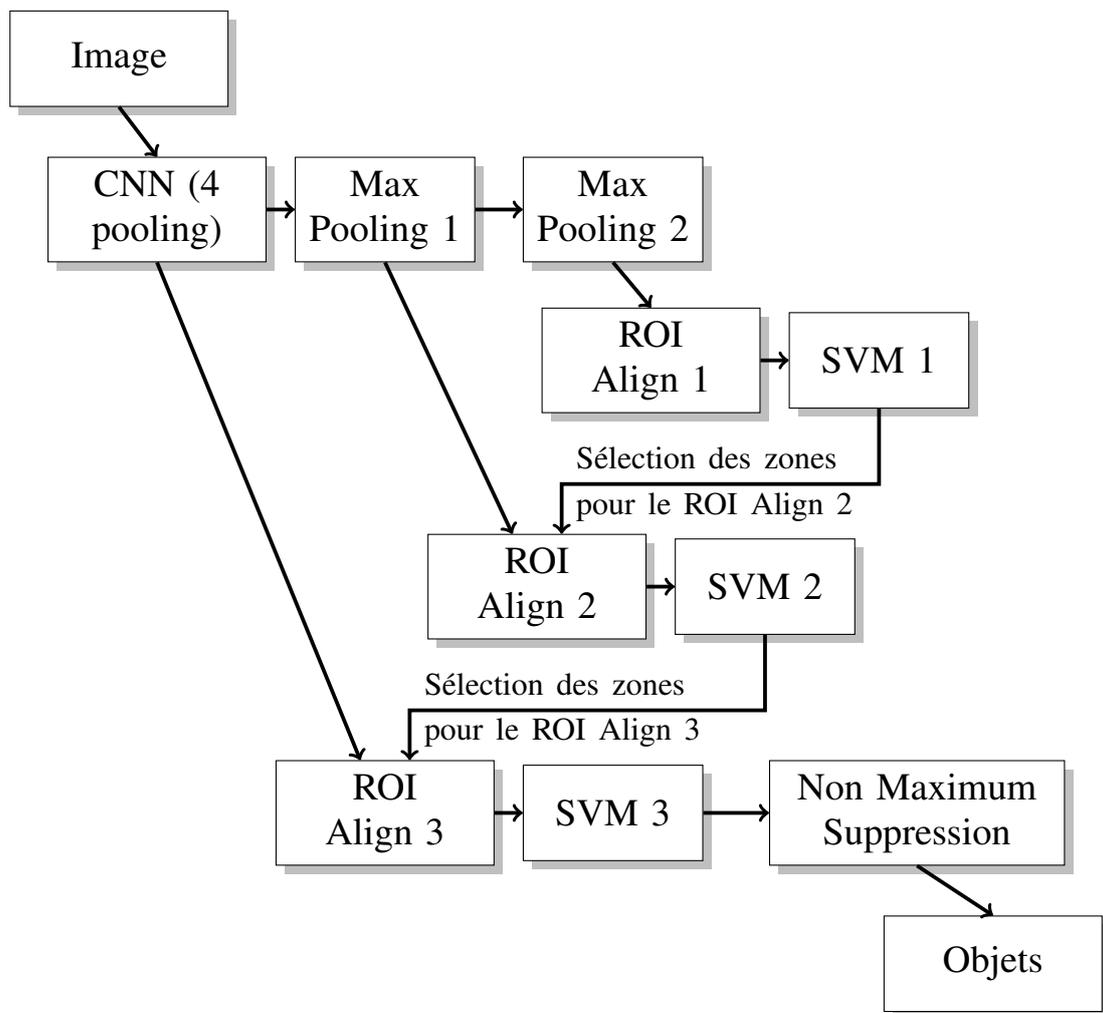
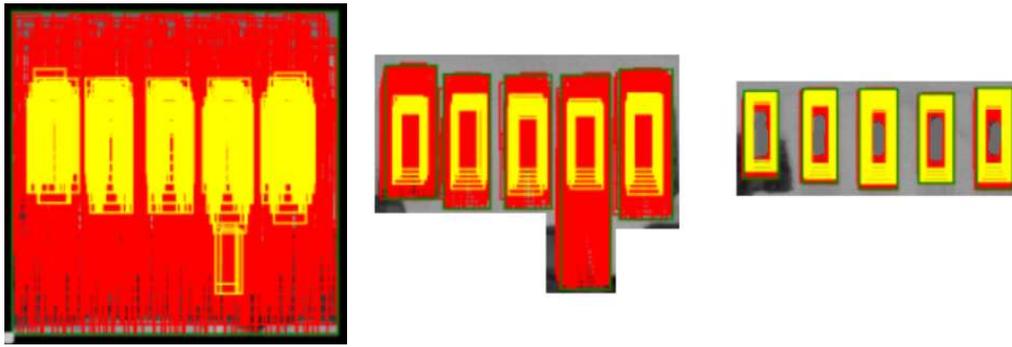


FIGURE 4.13 – Présentation de Sliding CNN, une méthode en cascades de détection d'objets avec un CNN.



(a) Première détection sur une carte de descripteurs à l'échelle 1/32. (b) Deuxième détection à l'échelle 1/16 sur une zone réduite par l'étape précédente. (c) Détection finale à l'échelle 1/8 sur une zone de nouveau réduite.

FIGURE 4.14 – Exemple des détections successives sur plusieurs tailles d'images.

fié SqueezeNet [98] qui tire sa particularité d'être très petit et léger pour être utilisé sur des supports nomades (téléphone, électronique embarquée). Les performances des quatre réseaux étant similaires dans notre méthode, le facteur primordial est donc le temps d'exécution sur CPU. Dans cette optique, SqueezeNet est le plus rapide de part sa petite taille. Il est en moyenne cinq fois plus rapide (1 seconde pour l'inférence pour 5 à 8 sur ResNet et DenseNet).

Enfin pour affiner les descripteurs à notre cas d'utilisation, nous avons pris un réseau pré-entraîné sur ImageNet [90]. Ce CNN est entraîné sur des images naturelles en RGB qui sont très différentes de nos images en niveau de gris à l'échelle nanométrique. Couramment, ces réseaux pré-entraînés sont ensuite ajustés sur les données utilisées. Dans notre cas, nous avons ajusté le CNN en posant un problème de classification similaire à ImageNet mais avec nos catégories. Nous distinguerons par la suite ces deux réseaux par Sliding CNN ImageNet pour la version pré-entraînée sur ImageNet et Sliding CNN FineTuned pour la version ré-entraînée sur nos images.

L'apport de cette méthode par rapport à la méthode classique sera discuté dans la suite.

4.2.5 Comparaisons des résultats

Jeux de données

Pour la comparaison des méthodes nous avons mis en place deux jeux de données différents. Le jeu de données 1 comporte 115 images d'un même cas d'utilisation comportant 475 objets d'une même classe ("gate") à détecter. La variabilité inter-image est importante et est due aux variations de procédés de fabrication. Les images sont confidentielles. L'objet ressemble à un "U" à l'envers gris clair sur fond noir avec une barre verticale blanche à l'intérieur. La grande difficulté provient de la barre verticale à hauteur très variable ainsi que le faible contraste du "U" sur plusieurs images. La variabilité intra-image est faible du fait que les objets d'une même image proviennent d'un échantillon physique fabriqué. Des images différentes signifient des échantillons différents et donc des procédés de fabrication différents. J'ai réalisé cette labellisation pendant la première année de thèse. Les données de ce jeu sont devenues indisponibles pour Pollen Metrology dans le cours de la deuxième année pour des causes d'accord de confidentialité. Pour cette raison, nous avons annoté un deuxième jeu de données.

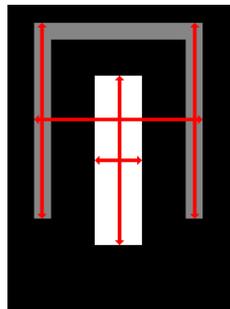


FIGURE 4.15 – Présentation du *design* du jeu de données 1 ainsi que les mesures à réaliser.

Le jeu de données 2 contient 12 classes réparties en 2210 objets. Il a été annoté de manière conjointe avec Aurélien. La répartition des objets n'est pas homogène (Figure 4.16). Les cinq classes les plus représentées contiennent 85% des objets de même que la classe la plus représentée contient un quart des objets. La variabilité inter-image est plus importante

dans ce jeu de données car nous avons inclus des images provenant de cas d'utilisations différents. La variabilité intra-image reste similaire au cas précédent. Les données sont aussi confidentielles mais des exemples représentatifs sont montrés (Figure 4.17) pour permettre de se rendre compte de la variabilité intra-classe. Dans ce jeu de données, nous avons ajouté des images de chimie fine car cela représente un secteur d'activité potentiel pour Pollen Metrology. L'approche globale reste la même et cet ajout ne perturbe pas notre test. Dans ce cadre avec plusieurs classes, nous entraînerons un détecteur par classe mais nous comparerons les métriques de tous les détecteurs ensemble.

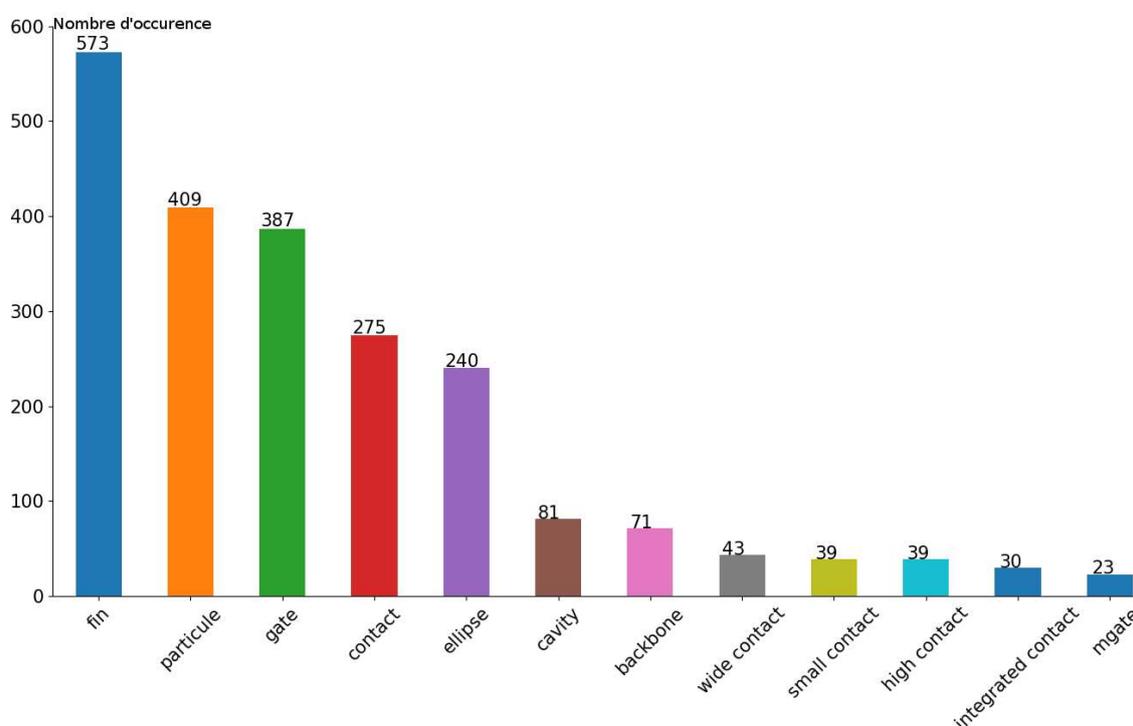


FIGURE 4.16 – Répartition des classes du jeu de données 2 pour la détection d'objets.

Pour les comparatifs, nous avons mis en place deux expériences différentes. La première vise à comparer les méthodes de template matching avec les méthodes d'apprentissage classique et à base d'apprentissage profond de la littérature. Pour cette étude, nous avons utilisé le premier jeu de données et fait varier le nombre d'exemples pour l'entraînement. La

4 Application au contexte industriel

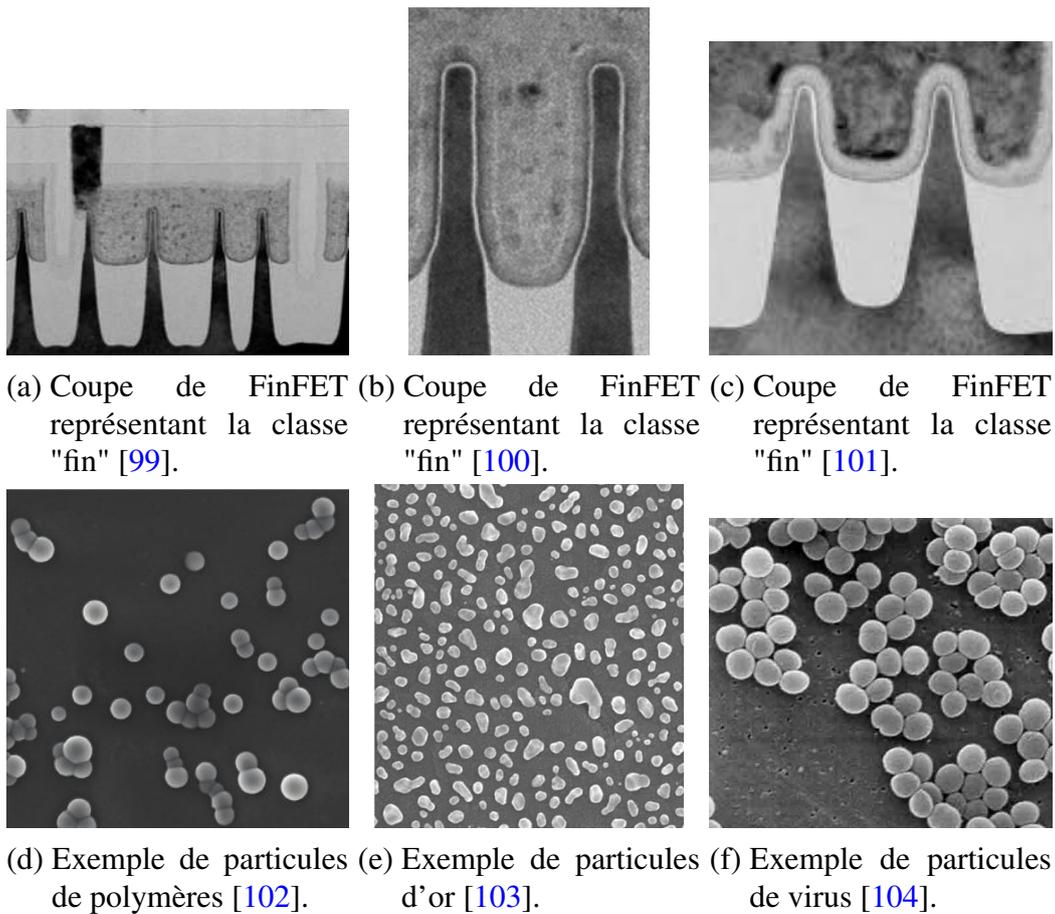


FIGURE 4.17 – Exemple d'images des classes "fin" et "particule". Ces images ne sont pas les images de notre jeu de données mais des images représentatives pour l'illustration.

deuxième expérience vise à comparer les deux méthodes d'apprentissage par fenêtre glissante.

Il est important de noter que malgré le nombre d'exemples disponibles dans nos jeux de données, les cas d'utilisations classiques ont beaucoup moins d'exemples. Dans nos comparaisons, nous utiliserons donc de préférence un nombre d'objets pour l'apprentissage assez faible pour reproduire cet aspect.

Comparaison des méthodes de l'état de l'art

Notre première comparaison concerne les méthodes de l'état de l'art. Au démarrage de cette thèse, nous avons étudié les méthodes de détection. À ce moment, nous n'avons accès qu'au jeu de données 1.

TABLE 4.1 – Comparaison des résultats en utilisant les objets d'une image pour l'apprentissage (4 objets)

	Précision	Rappel	Indice de Jaccard
TM	20.5%	35.1%	0.792
HOG	59.9%	69.7%	0.829
R-CNN	95.4%	38.4%	0.884

TABLE 4.2 – Comparaison des résultats en utilisant 20 objets pour l'apprentissage

	Précision	Rappel	Indice de Jaccard
HOG	68.7%	82.1%	0.834
R-CNN	92.1%	71.1%	0.887

TABLE 4.3 – Comparaison des résultats en utilisant 300 objets pour l'apprentissage

	Précision	Rappel	Indice de Jaccard
HOG	49.9%	99.3%	0.851
R-CNN	88.5%	93.0%	0.890

Pour cette première expérience, la précision et le rappel ont été mesurés en considérant comme positif les propositions qui ont un indice de Jaccard (Équation 4.4) avec la vérité terrain d'au moins 0.7. L'indice de Jaccard est la moyenne des indices de Jaccard des objets correctement détectés.

Pour les détecteurs comparés, le Template Matching (TM) correspond à la description de la Sous Section 4.2.2. Nous n'utilisons qu'un seul exemple d'apprentissage car les exemples positifs sont pris sur une même image

4 Application au contexte industriel

$$D = \frac{|X \cap Y|}{|X \cup Y|} \quad (4.4)$$

X, Y : des ensembles de pixels
 $|X|$: le nombre d'élément de X

et très similaires, de plus cela correspond à l'utilisation réelle. Le HOG correspond à l'implémentation proposée dans la Sous Section 4.2.3. Enfin nous avons reproduit R-CNN. Nous avons changé le CNN utilisé pour l'extraction des descripteurs par un NN postérieur (Xception [89]) à la publication originale.

Nous constatons tout d'abord que, comme pressenti, les résultats du Template Matching ne sont pas très robustes aux variations inter-images. Ensuite, les performances des deux autres modèles augmentent avec le nombre d'exemples d'apprentissage. Globalement, le R-CNN est plus performant que le HOG. Il est intéressant de noter que le R-CNN a une précision plus élevée que son rappel. À partir de ces comparaisons, nous pouvons dire que le rappel faible n'est pas causé par le Selective Search car lorsque le nombre d'exemples d'apprentissage est élevé, le rappel est élevé. Or, cette étape n'est pas apprise. Enfin, le HOG a son rappel qui augmente avec le nombre d'exemples au prix de la précision. Le placement global des propositions des modèles augmentent le nombre d'exemples mais reste supérieur avec le R-CNN.

Suite à cette expérience, les trois algorithmes ont été implémentés dans le produit de Pollen Metrology car ils répondent chacun à des besoins différents. Le Template Matching répond au besoin de la petite base d'apprentissage, le R-CNN propose des performances élevées et enfin, le HOG permet de détecter des objets que le R-CNN ne peut pas détecter à cause de l'algorithme de proposition de régions.

À force d'utilisation dans le logiciel par les utilisateurs, nous avons constaté que le R-CNN est peu utilisé. En effet, la méthode est plus longue que les autres pour la détection et il est difficile pour les utilisateurs de sa-

voir à l'avance quels objets ne peuvent pas être détectés par le R-CNN. Le HOG est la méthode principale employée par les utilisateurs. Cependant, comme discuté dans les résultats, le rappel pose problème pour les utilisateurs lorsque la variabilité des objets augmente. Nous avons donc décidé lors de la troisième année de développer une nouvelle méthode, le Sliding CNN.

Comparaison de Sliding CNN

Pour la deuxième expérience, nous voulons cette fois évaluer l'apport du Sliding CNN en comparaison du HOG sur une plus grande variété de cas. Pour cette expérience, nous avons conservé la même définition de la précision et du rappel que pour la première expérience. Nous avons substitué l'indice de Jaccard par la Mean Average Precision (mAP) qui a plus de sens pour évaluer un modèle sur plusieurs classes et est couramment utilisée pour comparer les modèles de détection d'objets. La mAP est la moyenne des Average Precision (AP). Le calcul des AP correspond à l'aire sous la courbe de la courbe de précision et rappel (Figure 4.18).

Enfin pour l'entraînement, nous avons fait une validation croisée en 5 blocs en conservant 30% des données pour l'entraînement et 70% pour le test. Le but étant d'être dans une situation proche de notre cas d'utilisation avec peu de données d'entraînement. Nous avons ensuite moyenné les résultats. Nous avons entraîné un modèle différent pour chaque classe d'objet présenté puis agrégé les résultats.

Les résultats nous permettent de dire que sur le jeu utilisé, notre approche de Sliding CNN apporte une amélioration importante pour la détection d'objets. Le ré-entraînement avec nos données sur une tâche de classification permet également d'améliorer les performances du détecteur avec des descripteurs mieux adaptés. En particulier, la précision qui augmente de presque 15 dixièmes soit une amélioration de 20% de la précision. Vis-à-vis du rappel, le Sliding CNN améliore les performances de 35 dixièmes soit une amélioration de 55%. Pour situer les résultats en terme

4 Application au contexte industriel

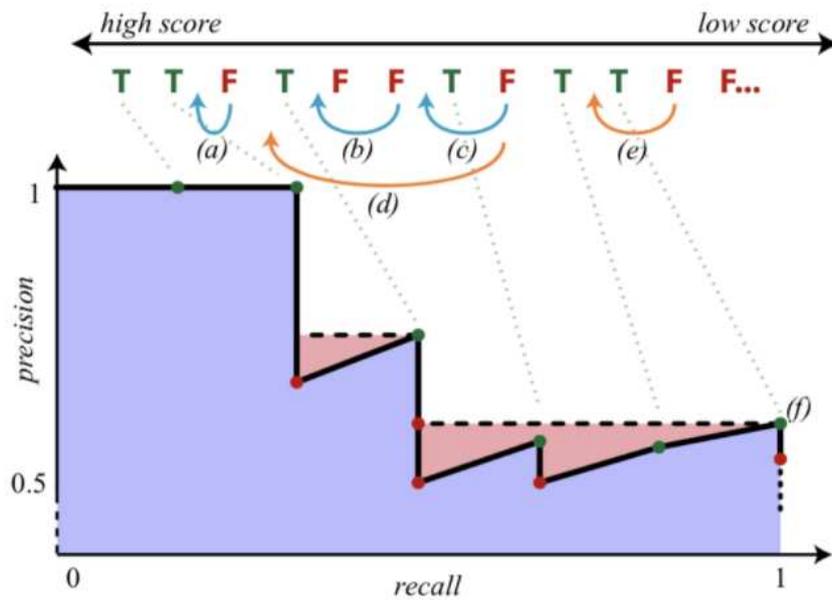


FIGURE 4.18 – Schéma explicatif du calcul de l'AP. L'AP correspond à l'aire de la région bleu et rouge. Le schéma provient de [105]

TABLE 4.4 – Comparaison des méthodes par fenêtre glissante sur le jeu de données 2

	Précision	Rappel	mAP
HOG	77.2%	61.3%	0.51
Sliding CNN ImageNet	70.5%	92.2%	0.64
Sliding CNN FineTuned	84.4%	96.0%	0.74

de mAP, sur ImageNet les résultats de l'état de l'art sont aux alentours de 0.75 et sur Pascal VOC [106] aux alentours de 0.80. La méthode du Sliding CNN propose donc des performances qui sont dans la gamme de valeur de l'état de l'art, même s'il reste compliqué de comparer ces métriques sur des jeux de données si différents.

Enfin, en terme de comparaison de temps, le temps d'entraînement n'est pas particulièrement important dans notre application. Cependant, notre

méthode entraîne uniquement un SVM donc la procédure est rapide comme le nombre de données est faible. Nos approches CNN sont plus longues que le HOG car il est plus long de passer tout les exemples dans le CNN. Sur le temps de détection, la version naïve était très longue car nous n'utilisons pas un CNN compact. De plus, pour obtenir une bonne précision, il nous fallait calculer plusieurs cartes de descripteurs à cause de la réduction de taille. Notre version Sliding CNN est plus rapide que le HOG (Table 4.5). Les versions ImageNet et FineTuned du Sliding CNN ont la même vitesse d'exécution de part le fait que la seule différence entre les méthodes sont les valeurs des poids du CNN. De plus, la confiance du SVM étant plus élevée, nous avons moins de boîtes à traiter dans le NMS.

TABLE 4.5 – Comparaison des méthodes par fenêtre glissante sur le jeu de données 2.

	HOG	Naïve CNN	Sliding CNN
Temps d'entraînement	1 sec	1-12 sec	1-12 sec
Temps de détection	1-15 sec	>180 sec	1-3 sec

En conclusion, notre proposition de Sliding CNN améliore grandement les performances de la détection d'objets dans notre cadre. Les limitations techniques de vitesse de la version naïve nous ont poussé à développer une accélération pour obtenir des temps raisonnable. Actuellement, la méthode n'a pas encore été implémentée dans le produit de Pollen Metrology mais l'implémentation du Sliding CNN est planifiée en accord avec les ressources de l'entreprise.

4.3 Mesures

L'objectif final de notre application est d'obtenir des mesures sur les différents objets de l'image. Nous venons de présenter des façons de définir un objet de manière visuelle et de le retrouver dans des images de manière

4 Application au contexte industriel

automatique. Il ne nous reste plus qu'à être capable de mesurer les éléments. Pour ce faire, il existe deux types d'outils qui ont été développés à Pollen Metrology. Le premier est un outil de contours actifs que j'ai personnellement développé antérieurement aux travaux de thèse. Il est basé sur le Gradient Vector Flow (GVF) [107] et nous proposons de résoudre l'optimisation de manière itérative. À partir de ce contour, nous pouvons extraire des dimensions ponctuelles. Le second outil vise à obtenir directement ces dimensions à partir de l'image. Nous venons localement faire un ajustement d'une fonction de transition sur les niveaux de gris. Dans les deux cas, l'algorithme fonctionne bien lorsque l'initialisation est proche (moins de 5 à 10 pixels) du résultat attendu. L'objectif de cette partie est de proposer un moyen d'améliorer le placement de ces outils sur les objets trouvés.

La problématique de base est la suivante : nous avons à notre disposition une base d'objets dont les mesures sont correctes, et nous voulons mesurer les nouveaux objets détectés sans intervention humaine (Figure 4.19). Ainsi, il nous faut initialiser automatiquement les mesures au plus proche de la mesure finale. Dans la suite, nous appellerons localisation de mesures le fait de positionner les mesures depuis le *design* vers les objets trouvés.

Notre première façon de replacer les mesures est une méthode naïve. Nous redimensionnons le *design* à la taille de l'objet détecté, de même pour les mesures sur le *design*. Ce changement de taille nous permet de poser les mesures sur l'objet. Cette approche reste performante tant que les objets sont similaires au *design*. Cependant, lorsque les objets subissent des déformations à cause des procédés de fabrication, cette méthode devient imprécise. Nous n'avons pas utilisé cette méthode de part son manque de robustesse.

Dans un premier temps, nous avons développé une méthode se basant sur les descripteurs HOG. En effet, l'utilisateur renseigne une base d'objets pour l'étape de détection dont les mesures ont été validées et donc à la bonne position. Ainsi, nous pouvons nous en servir à la place du *design* des objets ajoutés par l'utilisateur étant plus ressemblant aux objets que nous

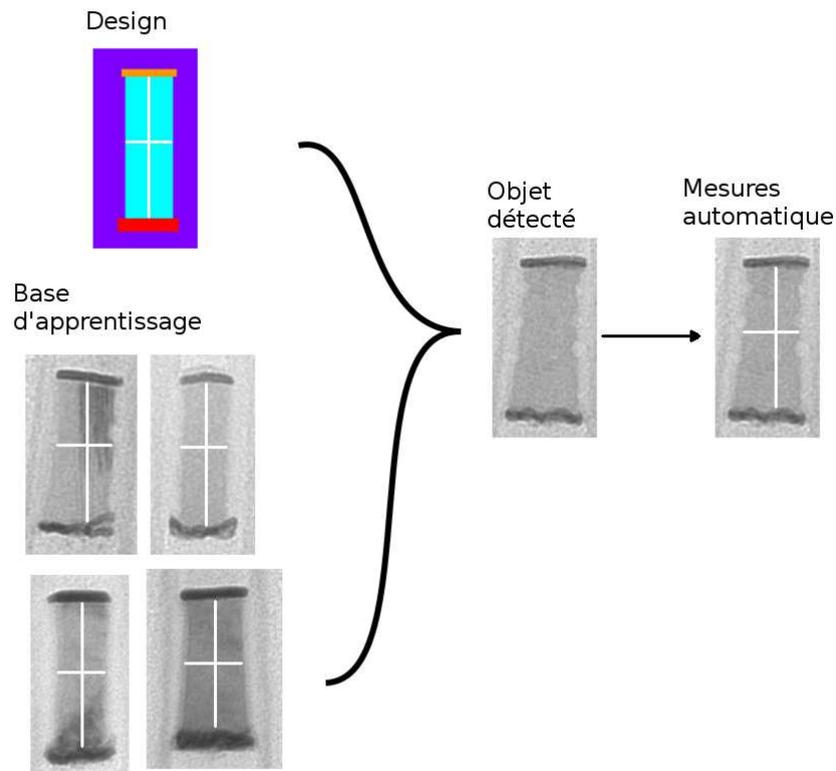


FIGURE 4.19 – Exemple de situation avec une base d'objets mesurés disponible et le résultat attendu.

allons détecter. Pour chaque objet détecté, nous cherchons quel est l'objet le plus proche dans la base d'apprentissage de l'étape de détection d'objets. Notre critère de vraisemblance est la norme L2 entre les descripteurs HOG des objets. Une fois le plus ressemblant trouvé, nous effectuons les mêmes ajustements que la méthode naïve.

Nous avons amélioré cette approche en ne considérant pas uniquement le plus ressemblant mais plusieurs. L'objectif est de prendre les n plus ressemblant et de faire la moyenne des mesures initiales, afin d'affiner le placement. Dans la pratique, nous avons pris $n = 3$. Nous avons observé une amélioration de l'ordre de 5% sur la précision de la localisation. Dans les comparaisons, nous nommerons cette méthode "Path Matcher".

Dans un second temps, avec l'utilisation du TreeCut, une segmentation

4 Application au contexte industriel

de l'objet est disponible. Nous pouvons donc utiliser de manière plus efficace le *design* sur lequel l'utilisateur a défini ces mesures. De cette façon, nous voulons reporter des mesures du *design* à la segmentation de l'objet. Le *design* et la segmentation permettent de définir des contours autour des régions et nous voulons faire leur mise en correspondance. Nous proposons de chercher le point de contour de la segmentation qui est le plus proche de la proposition du Patch Matcher. Nous nommerons cette approche "TreeCut Matcher Hybrid". La différence entre les deux approches est présentée en Figure 4.20.

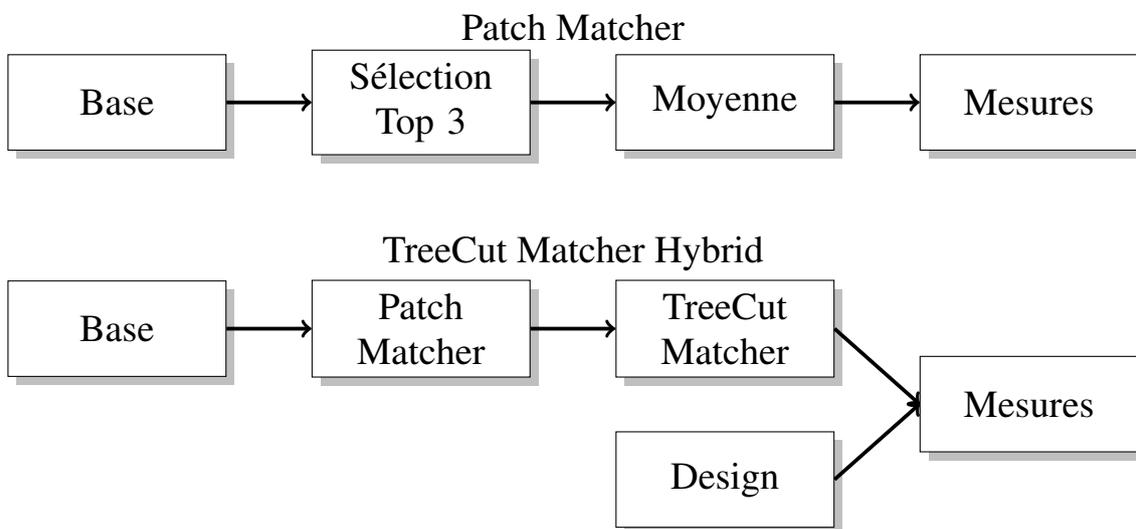


FIGURE 4.20 – Présentation des différences entre Patch Matcher et TreeCut Matcher Hybrid.

Enfin, pour la comparaison nous avons annoté six cas d'utilisations différents avec une dizaine d'objets par cas, ainsi que cinq mesures par objets provenant du jeu de données 2. Nous avons mis en place deux métriques différentes. La première est la norme L2 moyenne en pixels entre la mesure replacée et la vérité terrain. Cette métrique nous permet d'avoir une mesure de la précision de la méthode. La deuxième est le ratio de réussite de mesures sur le cas. Elle comptabilise le nombre de mesures qui sont à moins de 10 pixels de la vérité terrain sur le nombre de mesures total. Cette métrique est plus représentative des performances pour l'utilisation car elle correspond à son efficacité sur la localisation avec les algorithmes

de mesures actuels.

TABLE 4.6 – Comparaison des méthodes de localisation de mesures sur différentes classes d’objets provenant du jeu de données 2.

		Cavity	Memory	Back bone	Fin	Gate	Contact
Patch Matcher	Distance Moyenne	26px	41px	13px	30px	4px	31px
	Ratio Réussite	7.4%	4.1%	56.7%	18.6%	98.7%	26.7%
TreeCut Matcher Hybrid	Distance Moyenne	48px	23px	8px	8px	5px	19px
	Ratio Réussite	19.4%	32.2%	73.6%	71.8%	95.1%	30.0%

Nous constatons que les performances sont très dispersées selon les cas. Cela est principalement dû aux variations que subissent les objets ainsi que leur taille. Pour la variation, nous ne pouvons pas la connaître à l’avance. Elle dépend du niveau de maturité du procédé ainsi que de la complexité de l’objet. Pour la taille, les grands objets sont plus compliqués à mesurer car une petite variation relative représente une variation importante en terme de nombre de pixels. Dans le cas des *Cavity*, le Patch Matcher a une distance moyenne plus faible du fait que les cas d’erreur du TreeCut Matcher sont très éloignés de la vérité terrain, mais en moyenne les mesures sont plus souvent dans notre critère de 10 pixels. Pour le cas des *Gate*, le Patch Matcher affiche de peu des meilleurs résultats car ces objets ont une petite taille et la vérité terrain est souvent proche de l’initialisation. Cependant, le TreeCut a tendance à proposer des contours arbitraires selon le niveau de flou, ce qui réduit un peu ses performances.

Lorsque l’on parle de métrologie, les incertitudes de mesures sont une information importante pour mesurer la fiabilité de ces dernières. Les outils de mesures de Pollen renvoient une forme d’incertitude sur le placement de la transition de matériaux en mesurant la dispersion de l’ajustement de la

4 Application au contexte industriel

fonction de transition. Cependant, nous n'avons pas fait d'étude des incertitudes de cet ajustement vis-à-vis de nos données ou du modèle d'images que nous avons mise en place. Dans le quotidien de l'utilisation de ces outils, le point le plus important pour les utilisateurs n'est pas tant la précision de la mesure obtenue mais sa répétabilité. La variabilité entre les utilisateurs sur la position de la transition nous a poussé à leur permettre de l'ajuster visuellement, pour que cette décision se répercute de manière fiable sur toutes les mesures suivantes. Ce facteur est un point crucial pour pouvoir suivre l'évolution de leur expérience afin d'obtenir un procédé mature.

Dans l'ensemble, la localisation des mesures et la mesure elle-même sont deux sujets qui mériteraient des approfondissements pour une utilisation automatique. Dans la pratique, les utilisateurs vérifient les objets mesurés, corrigent les mauvaises mesures manuellement ou retirent l'objet en question de l'analyse.

4.4 Réalisation expérimentale du pipeline

Pour le pré-traitement, nous avons utilisé un filtre de Moyennes Non Locales. La paramétrisation des deux fenêtres de calcul des moyennes a été faite manuellement en utilisant une taille de 7 pour les moyennes locales et de 21 pour la moyenne des moyennes locales. Le paramètre de force du filtre a été fixé par la variance locale du patch de taille 21×21 le plus lumineux de l'image. En effet, avec le modèle de bruit de Poisson, cette zone est la plus bruitée.

Pour le template matching, nous avons testé deux méthodes, la corrélation de phase que nous avons réalisé et la corrélation croisée normalisée proposée dans OpenCV [108]. Les résultats étant très similaires, nous avons choisi d'utiliser la corrélation croisée normalisée d'OpenCV.

Pour la détection d'objets par HOG, nous avons utilisé les paramètres suivants :

- Taille de bloc : 16
- Taille de cellule : 8
- Nombre de classes dans l’histogramme : 16

Ces paramètres correspondent à une taille pour laquelle les éléments discriminants des objets sont encore visibles. Cependant, lorsque les objets sont grands (plus de 500px de large ou de haut) le vecteur de descripteurs de l’objet a une taille importante ce qui peut réduire les performances du SVM. Dans la pratique, l’aspect dense des images contenant des grands objets nous aide sur la partie classification. Pour le classifieur SVM, le terme de pénalisation a été fixé à 0.0001 pour correspondre à un SVM à marge souple.

4.5 Conclusion

Nous avons présenté dans ce chapitre la chaîne de traitement général que nous avons mis en place pour réaliser la mesure automatique d’objets à partir d’un *design*. La contribution principale de ce chapitre est la nouvelle méthode de détection d’objets que nous avons développée à partir de CNN et fenêtre glissante. Elle permet en effet d’obtenir des résultats plus robustes que les méthodes classiques avec moins d’objets d’apprentissage que les méthodes d’apprentissage profond. Notre deuxième contribution est la localisation de mesures à l’aide du TreeCut. Cette méthode a été présentée dans un contexte simpliste de mesures dimensionnelles ponctuelles.

4 Application au contexte industriel

5 Conclusion

Cette thèse s'inscrit dans un cadre industriel de la métrologie appliquée à des objets manufacturés du type semi-conducteur. Le but industriel est de proposer une suite d'algorithmes permettant la mise en place de mesures automatiques sur des images de microscopie électronique. Sur l'aspect méthodologique, cette thèse contribue à une meilleure connaissance des microscopes électroniques en transmission. De plus, nous avons développé une méthode de segmentation basée sur un *design*. Enfin, nous proposons l'assemblage de ces algorithmes pour la mesure automatique.

5.1 Contributions

Nous avons dans un premier temps analysé les images à notre disposition. Elles proviennent de microscopes électroniques en transmission et peuvent être multi-modales. Les différentes modalités mettent en évidence des caractéristiques particulières de l'échantillon imagé. Nous avons mis en évidence une relation linéaire entre les modalités. Nous pouvons affirmer que l'utilisation de deux modalités au lieu de trois est suffisant sans perte d'informations. Dans un deuxième temps, nous avons vérifié l'hypothèse de bruit de ces images. Nous avons expérimentalement confirmé que le bruit des images de microscopie électronique en transmission présente les caractéristiques d'un bruit de Poisson. Nous avons réalisé un état de l'art succinct pour traiter les images de manière adéquate. De plus, en connaissant le fonctionnement du microscope, nous avons mis en place un modèle d'images afin d'évaluer notre méthode de segmentation en fonction de différents niveaux de bruit.

5 Conclusion

Dans un second temps, nous avons développé une méthode de segmentation de matériaux, le TreeCut. Cet algorithme repose sur la particularité du microscope électronique en transmission qui image les matériaux avec des niveaux de gris différents. La méthode utilise un *design* qui est une version simplifiée et idéale de l'objet. Pour permettre une représentation efficace du *design*, nous avons utilisé l'Arbre de Formes. Cette structure de données nous permet d'organiser simplement le *design* pour effectuer une segmentation de l'image. Le principe est de mettre en correspondance l'Arbre de Formes de l'image avec celui du *design*. En utilisant cette structure, nous avons mis en place une minimisation d'énergie pour réaliser la segmentation. Grâce à cette minimisation, nous pouvons donner plus ou moins d'importance à différents attributs de l'image tel que la position ou le niveau de gris. Enfin, un paramètre de régularisation nous permet de contraindre la minimisation pour respecter la forme du *design*. Afin de montrer la versatilité de l'approche, nous avons démontré l'application de notre méthode sur des images d'Imagerie à Raisonance Magnétique sans adaptation ainsi que comparer la méthode proposée avec des approches de l'état de l'art.

Pour finir, nous avons présenté l'articulation de travaux précédents dans une suite globale d'algorithmes permettant la mesure d'objets définis par un utilisateur. Pour ce faire, nous avons développé une méthode de détection d'objets plus robuste que les existantes capable de gérer une petite base d'apprentissage. Cette approche utilise les réseaux de neurones convolutifs pour extraire les descripteurs permettant de discriminer les objets du fond. Nous avons proposé une architecture nous permettant d'utiliser des méthodes de fenêtres glissantes de manière efficace. En utilisant notre approche de segmentation, nous avons proposé une méthode permettant de replacer des mesures. Le remplacement des mesures correspond à la transposition des mesures des objets de la base d'apprentissage et du *design* sur les objets trouvés par la méthode de détection. Cette dernière se base sur la méthode de segmentation développée précédemment. Nous avons comparé cette méthode à celle précédemment développé chez Pollen Metrology. Cet ensemble d'algorithmes a été intégré dans le produit de Pollen Metrology.

5.2 Travaux futurs

Sur l'aspect microscopie, des travaux additionnels pourraient être menés afin d'étudier la relation entre les modalités. En effet, nous avons montré la combinaison linéaire mais nous ne pouvons pas déduire les coefficients sans avoir les trois modalités à disposition. Ces coefficients dépendent certainement de l'échantillon ainsi que des paramètres d'acquisition. Cependant, des travaux complémentaires sont nécessaires avec un acteur de la microscopie électronique. Avec de meilleures connaissances sur le fonctionnement du microscope, nous pourrions aussi améliorer le modèle d'images afin de générer des données synthétiques pour l'apprentissage ou les tests. Cependant, la microscopie électronique en transmission met en application des phénomènes physiques complexes à modéliser ce qui rend la tâche particulièrement difficile.

Pour la partie segmentation, la première amélioration potentielle de notre méthode dans le cas de la microscopie électronique en transmission est l'utilisation de la proposition de structure d'Arbre de Formes multivariée [58]. Cette méthode permettrait d'utiliser de manière efficace les différentes modalités en une seule représentation. La deuxième piste d'amélioration serait de modifier les différents termes de la minimisation. En effet, des descripteurs de textures seraient particulièrement adaptés pour les images haute résolution dans lesquels nous observons la structure atomique de l'échantillon. Enfin, notre minimisation actuelle repose sur le Graph Cut et ne permet donc pas de respecter la contrainte de partition que nous souhaitons afin d'être au plus près du *design*. De plus, pour améliorer l'automatisation, les différents facteurs de la minimisation pourraient être évalués de manière automatique en fonction du *design*. Nous avons dans un premier temps fixé ces valeurs en fonction de nos cas d'utilisation, mais nous avons constaté que certaines caractéristiques du *design* influent grandement sur les valeurs optimales.

Enfin, sur l'aspect de la mesure automatique, plusieurs pistes d'amélioration sont envisageables. En premier lieu, la détection d'objet pourrait

5 Conclusion

être améliorée à l'aide de méthodes d'apprentissage profond. Parmi ces dernières, un champs d'étude sur le transfert d'apprentissage ou l'adaptation de domaine permettrait de tirer profit d'une large base d'apprentissage afin d'utiliser moins d'exemples pour les détections d'objets différents. Chez Pollen Metrology, nous avons au fil du temps construit une base de quelques milliers d'images ce qui reste encore très peu pour mettre en place de telle approches. Cependant, un acteur du semi-conducteur serait en mesure d'avoir suffisamment de données. Sur la partie remplacement de mesures, notre méthode actuelle ne permet pas de faire une mise en correspondance efficace depuis le *design* vers la segmentation. Des méthodes appropriées pourraient être utilisées. Des améliorations de la segmentation seront directement répercutées par des améliorations sur le remplacement. De plus, ayant accès à une carte de segmentation qui correspond au *design*, il est possible d'étendre l'approche à des mesures plus complexes telles que des mesures dimensionnelles continues, des descripteurs de formes ou encore des analyses de texture et d'homogénéité du matériau. Enfin, il serait bénéfique d'être capable d'évaluer la qualité de la segmentation, car dans certains cas, la méthode de segmentation propose des résultats hautement dégradés et la méthode utilisée précédemment chez Pollen Metrology permet d'obtenir de meilleurs résultats.

Bibliographie

- [1] “Nanomaterials Market Size, Share | Industry Forecast 2014-2022.” [Online]. Available : <https://www.alliedmarketresearch.com/nano-materials-market>
- [2] T. M. Research, “Advanced Materials Market Estimated to Reach US\$ 102.48 Bn by 2024; Global Industry Analysis, Size, Share, Growth, Trends, and Forecast 2016 - 2024 - Transparency Market Research,” 2016. [Online]. Available : <https://www.prnewswire.com/news-releases/advanced-materials-market-estimated-to-reach-us-10248-bn-by-2024-global-industry-analysis-size-share-growth-trends-and-forecast-2016---2024---transparency-market-research-608036376.html>
- [3] G. Moore, “Moore’s Law,” 1965. [Online]. Available : <http://archive.wikiwix.com/cache/?url=http%3A%2F%2Fdownload.intel.com%2Fresearch%2Fsilicon%2Fmoorepaper.pdf>
- [4] ———, “Gordon_Moore_Speech.pdf,” 1975. [Online]. Available : http://www.eng.auburn.edu/~agrawvd/COURSE/E7770_Spr07/READ/Gordon_Moore_1975_Speech.pdf
- [5] “About - IEEE International Roadmap for Devices and Systems™.” [Online]. Available : <https://irds.ieee.org/>
- [6] “Big Trouble At 3nm,” Jun. 2018. [Online]. Available : [\url{https://semiengineering.com/big-trouble-at-3nm/}](https://semiengineering.com/big-trouble-at-3nm/)

Bibliographie

- [7] “5nm Vs. 3nm,” Jun. 2019. [Online]. Available : <https://semiengineering.com/5nm-vs-3nm/>
- [8] P. Allen and D. Holberg, “Cmos analog circuit design (allen,holberg-2002).pdf,” 2002. [Online]. Available : https://doc.xdevs.com/docs/_Books/ASIC_Design/cmos%20analog%20circuit%20design%20%28allen%2Cholberg-2002%29.pdf
- [9] C. Auth, C. Allen, A. Blattner, D. Bergstrom, M. Brazier, M. Bost, M. Buehler, V. Chikarmane, T. Ghani, T. Glassman, R. Grover, W. Han, D. Hanken, M. Hattendorf, P. Hentges, R. Heussner, J. Hicks, D. Ingerly, P. Jain, S. Jaloviar, R. James, D. Jones, J. Jopling, S. Joshi, C. Kenyon, H. Liu, R. McFadden, B. McIntyre, J. Neiryneck, C. Parker, L. Pipes, I. Post, S. Pradhan, M. Prince, S. Ramey, T. Reynolds, J. Roesler, J. Sandford, J. Seiple, P. Smith, C. Thomas, D. Towner, T. Troeger, C. Weber, P. Yashar, K. Zawadzki, and K. Mistry, “A 22nm high performance and low-power CMOS technology featuring fully-depleted tri-gate transistors, self-aligned contacts and high density MIM capacitors,” in *2012 Symposium on VLSI Technology (VLSIT)*, Jun. 2012, pp. 131–132.
- [10] R. Muthinti, N. Loubet, R. Chao, J. Ott, M. Guillorn, N. Felix, J. Gaudiello, P. Lund, A. Cepler, M. Sendelbach, O. Cohen, S. Wolfling, C. Bozdog, and M. Klare, “Advanced in-line optical metrology of sub-10nm structures for gate all around devices (GAA),” in *Metrology, Inspection, and Process Control for Microlithography XXX*, vol. 9778. International Society for Optics and Photonics, Mar. 2016, p. 977810. [Online]. Available : <http://www.spiedigitallibrary.org/conference-proceedings-of-spie/9778/977810/Advanced-in-line-optical-metrology-of-sub-10nm-structures-for/10.1117/12.2220379.short>
- [11] N. Loubet, T. Hook, P. Montanini, C.-W. Yeung, S. Kanakasabapathy, M. Guillom, T. Yamashita, J. Zhang, X. Miao, J. Wang, A. Young, R. Chao, M. Kang, Z. Liu, S. Fan, B. Hamieh, S. Sieg,

- Y. Mignot, W. Xu, S.-C. Seo, J. Yoo, S. Mochizuki, M. Sankarapandian, O. Kwon, A. Carr, A. Greene, Y. Park, J. Frougier, R. Galatage, R. Bao, J. Shearer, R. Conti, H. Song, D. Lee, D. Kong, Y. Xu, A. Arceo, Z. Bi, P. Xu, R. Muthinti, J. Li, R. Wong, D. Brown, P. Oldiges, R. Robison, J. Arnold, N. Felix, S. Skordas, J. Gaudiello, T. Standaert, H. Jagannathan, D. Corliss, M.-H. Na, A. Knorr, T. Wu, D. Gupta, S. Lian, R. Divakaruni, T. Gow, C. Labelle, S. Lee, V. Paruchuri, H. Bu, and M. Khare, “Stacked nanosheet gate-all-around transistor to enable scaling beyond FinFET,” in *2017 Symposium on VLSI Technology*, Jun. 2017, pp. T230–T231.
- [12] “Front end of line,” *Wikipedia*, May 2019. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Front_end_of_line&oldid=897424226
- [13] M. Baykara and U. Schwarz, “Atomic Force Microscopy : Methods and Applications,” in *Encyclopedia of Spectroscopy and Spectrometry*, Feb. 2016.
- [14] “Scanning electron microscope,” *Wikipedia*, Mar. 2020. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Scanning_electron_microscope&oldid=945029768
- [15] P. Goodhew, “General Introduction to Transmission Electron Microscopy (TEM),” Jul. 2011, pp. 1–19.
- [16] “Transmission electron microscopy,” *Wikipedia*, Mar. 2020. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Transmission_electron_microscopy&oldid=946411540
- [17] “Scanning transmission electron microscopy,” *Wikipedia*, Mar. 2020. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Scanning_transmission_electron_microscopy&oldid=947065910

Bibliographie

- [18] K. Polyakova, E. Ryklina, and S. Prokoshkin, “Thermomechanical response of Titanium Nickelide on austenite grain/subgrain size,” *Materials Today : Proceedings*, vol. 4, pp. 4836–4840, Dec. 2017.
- [19] G. Wille, J. Hellal, P. Ollivier, A. Richard, A. Burel, L. Jolly, M. Crampon, and C. Michel, “Cryo-Scanning Electron Microscopy (SEM) and Scanning Transmission Electron Microscopy (STEM)-in-SEM for Bio- and Organo-Mineral Interface Characterization in the Environment,” *Microscopy and Microanalysis*, vol. 23, no. 6, pp. 1159–1172, Dec. 2017. [Online]. Available : <https://www.cambridge.org/core/journals/microscopy-and-microanalysis/article/cryoscanning-electron-microscopy-sem-and-scanning-transmission-electron-microscopy-steminsem-for-bio-and-organomineral-interface-characterization-in-the-environment/5CF8BAC8EA4BA8AC73D27FF31F27E09D/core-reader>
- [20] M. Makitalo and A. Foi, “Optimal Inversion of the Anscombe Transformation in Low-Count Poisson Image Denoising,” *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 99–109, Jan. 2011. [Online]. Available : <http://ieeexplore.ieee.org/document/5504216/>
- [21] F. Luisier, C. Vonesch, T. Blu, and M. Unser, “Fast interscale wavelet denoising of Poisson-corrupted images,” *Signal Processing*, vol. 90, no. 2, pp. 415–427, Feb. 2010. [Online]. Available : <http://www.sciencedirect.com/science/article/pii/S0165168409003016>
- [22] B. Zhang, J. M. Fadili, and J.-L. Starck, “Wavelets, Ridgelets, and Curvelets for Poisson Noise Removal,” *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1093–1108, Jul. 2008.
- [23] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 2, Jun. 2005, pp. 60–65 vol. 2.

- [24] C.-A. Deledalle, F. Tupin, and L. Denis, “Poisson NL means : Unsupervised non local means for Poisson noise,” in *2010 IEEE International Conference on Image Processing*. Hong Kong, Hong Kong : IEEE, Sep. 2010, pp. 801–804. [Online]. Available : <http://ieeexplore.ieee.org/document/5653394/>
- [25] R. F. Loane, E. J. Kirkland, and J. Silcox, “Visibility of single heavy atoms on thin crystalline silicon in simulated annular dark-field STEM images,” *Acta Crystallographica Section A Foundations of Crystallography*, vol. 44, no. 6, pp. 912–927, Nov. 1988. [Online]. Available : <http://scripts.iucr.org/cgi-bin/paper?S0108767388006403>
- [26] T. Seki, Y. Ikuhara, and N. Shibata, “Theoretical framework of statistical noise in scanning transmission electron microscopy,” *Ultramicroscopy*, vol. 193, pp. 118–125, Oct. 2018. [Online]. Available : <http://www.sciencedirect.com/science/article/pii/S0304399118300603>
- [27] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, “Practical Poissonian-Gaussian Noise Modeling and Fitting for Single-Image Raw-Data,” *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008. [Online]. Available : <http://ieeexplore.ieee.org/document/4623175/>
- [28] C. Sutour, C.-A. Deledalle, and J.-F. Aujol, “Estimation du niveau de bruit par la détection non paramétrique de zones homogènes,” in *Gretsi*, Lyon, France, Sep. 2015. [Online]. Available : <https://hal.archives-ouvertes.fr/hal-01241209>
- [29] R. D’Agostino and E. S. Pearson, “Tests for Departure from Normality. Empirical Results for the Distributions of b_2 and $\sqrt{b_1}$,” *Biometrika*, vol. 60, no. 3, pp. 613–622, 1973. [Online]. Available : <https://www.jstor.org/stable/2335012>

Bibliographie

- [30] “BIG > PureDenoise.” [Online]. Available : <http://bigwww.epfl.ch/algorithms/denoise/>
- [31] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “BM3D Image Denoising with Shape-Adaptive Principal Component Analysis,” p. 7.
- [32] “Denoising — OpenCV 2.4.13.7 documentation.” [Online]. Available : <https://docs.opencv.org/2.4/modules/photo/doc/denoising.html>
- [33] K. Stepka, M. Mavska, J. J. Pálenik, V. Pospíchalová, A. Kotrbová, L. Ilkovic, D. Klemová, A. Hampl, V. Bryja, and P. Matula, “Automatic Detection and Segmentation of Exosomes in Transmission Electron Microscopy,” in *ECCV Workshops*, 2016.
- [34] N. Vu and B. S. Manjunath, “Graph cut segmentation of neuronal structures from transmission electron micrographs,” in *2008 15th IEEE International Conference on Image Processing*. IEEE, 2008, pp. 725–728.
- [35] Y. Li, E. Roth, V. Agrawal, A. Eshein, J. Fredrick, L. Almasalha, A. Shim, R. Bleher, V. P. Dravid, and V. Backman, “Quantifying Three-dimensional Chromatin Organization Utilizing Scanning Transmission Electron Microscopy : ChromSTEM,” *bioRxiv*, 2019.
- [36] W. Yin, R. Huang, R. Qi, and C. Duan, “Extraction of structural and chemical information from high angle annular dark-field image by an improved peaks finding method,” *Microscopy Research and Technique*, vol. 79, no. 9, pp. 820–826, 2016. [Online]. Available : <http://onlinelibrary.wiley.com/doi/abs/10.1002/jemt.22704>
- [37] W. Ouyang, F. Tombari, S. Mattoccia, L. Di Stefano, and W. Cham, “Performance Evaluation of Full Search Equivalent Pattern Mat-

- ching Algorithms,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 127–143, 2012.
- [38] O. Pele and M. Werman, “Robust Real-Time Pattern Matching Using Bayesian Sequential Hypothesis Testing,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1427–1443, 2008.
- [39] D. Huttenlocher, G. Klanderman, and W. Rucklidge, “Comparing images using the Hausdorff distance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850–863, 1993.
- [40] A. Sibiryakov, “Fast and high-performance template matching method,” in *CVPR 2011*, 2011, pp. 1417–1424.
- [41] E. Elboher and M. Werman, “Asymmetric Correlation : A Noise Robust Similarity Measure for Template Matching,” *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3062–3073, 2013.
- [42] D. Lowe, “Object recognition from local scale-invariant features,” in *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Kerkyra, Greece : IEEE, 1999, pp. 1150–1157 vol.2. [Online]. Available : <http://ieeexplore.ieee.org/document/790410/>
- [43] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF : Speeded Up Robust Features,” in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg : Springer Berlin Heidelberg, 2006, vol. 3951, pp. 404–417. [Online]. Available : http://link.springer.com/10.1007/11744023_32
- [44] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB : An efficient alternative to SIFT or SURF,” in *2011 International Conference on Computer Vision*, 2011, pp. 2564–2571.

Bibliographie

- [45] M. Brown and D. G. Lowe, “Automatic Panoramic Image Stitching using Invariant Features,” *International Journal of Computer Vision*, vol. 74, no. 1, pp. 59–73, 2007. [Online]. Available : <https://doi.org/10.1007/s11263-006-0002-3>
- [46] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2007, pp. 1–8.
- [47] F. Perronnin and C. Dance, “Fisher Kernels on Visual Vocabularies for Image Categorization,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. Minneapolis, MN, USA : IEEE, Jun. 2007, pp. 1–8. [Online]. Available : <http://ieeexplore.ieee.org/document/4270291/>
- [48] S. He, Q. Yang, R. W. Lau, J. Wang, and M.-H. Yang, “Visual Tracking via Locality Sensitive Histograms,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*. Portland, OR, USA : IEEE, Jun. 2013, pp. 2427–2434. [Online]. Available : <http://ieeexplore.ieee.org/document/6619158/>
- [49] S. Oron, T. Dekel, T. Xue, W. T. Freeman, and S. Avidan, “Best-Buddies Similarity—Robust Template Matching Using Mutual Nearest Neighbors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 1799–1813, 2018.
- [50] I. Talmi, R. Mechrez, and L. Zelnik-Manor, “Template Matching With Deformable Diversity Similarity,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 175–183. [Online]. Available : http://openaccess.thecvf.com/content_cvpr_2017/html/Talmi_Template_Matching_With_CVPR_2017_paper.html
- [51] S. Klein, U. A. van der Heide, I. M. Lips, M. van Vulpen, M. Staring, and J. P. W. Pluim, “Automatic segmentation of the

- prostate in 3D MR images by atlas matching using localized mutual information,” *Medical Physics*, vol. 35, no. 4, pp. 1407–1417, 2008. [Online]. Available : <http://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.2842076>
- [52] S. Martin, V. Daanen, and J. Troccaz, “Atlas-based prostate segmentation using an hybrid registration,” *International Journal of Computer Assisted Radiology and Surgery*, vol. 3, no. 6, pp. 485–492, 2008. [Online]. Available : <http://link.springer.com/10.1007/s11548-008-0247-0>
- [53] A. Sotiras, C. Davatzikos, and N. Paragios, “Deformable Medical Image Registration : A Survey,” *IEEE Transactions on Medical Imaging*, vol. 32, no. 7, pp. 1153–1190, 2013. [Online]. Available : <http://ieeexplore.ieee.org/document/6522524/>
- [54] E. Karami, S. Prasad, and M. Shehata, “Image Matching Using SIFT, SURF, BRIEF and ORB : Performance Comparison for Distorted Images,” *arXiv :1710.02726 [cs]*, 2017. [Online]. Available : <http://arxiv.org/abs/1710.02726>
- [55] L. He and S. Osher, “A fast multiphase level set algorithm for solving the Chan-Vese model,” *PAMM*, vol. 7, no. 1, pp. 1 041 911–1 041 912, 2007. [Online]. Available : <https://onlinelibrary.wiley.com/doi/abs/10.1002/pamm.200700659>
- [56] S. Gao and T. Bui, “Image segmentation and selective smoothing by using Mumford-Shah model,” *Image Processing, IEEE Transactions on*, vol. 14, pp. 1537–1549, 2005.
- [57] Y. Xu, E. Carlinet, T. Geraud, and L. Najman, “Hierarchical Segmentation Using Tree-Based Shape Spaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 457–469, Mar. 2017. [Online]. Available : <http://ieeexplore.ieee.org/document/7452658/>

Bibliographie

- [58] E. Carlinet, “A Tree of shapes for multivariate images.”
- [59] T. Géraud, E. Carlinet, S. Crozet, and L. Najman, “A quasi-linear algorithm to compute the tree of shapes of n-D images,” in *International Symposium on Mathematical Morphology*, ser. Lecture Notes in Computer Science, C. L. L. Hendriks, G. Borgefors, and R. Strand, Eds., vol. 7883. Uppsala, Sweden : Springer, May 2013, pp. 97–108. [Online]. Available : <https://hal.archives-ouvertes.fr/hal-00798620>
- [60] V. Caselles and P. Monasse, “Grain Filters,” *Journal of Mathematical Imaging and Vision*, vol. 17, no. 3, pp. 249–270, Nov. 2002. [Online]. Available : <https://doi.org/10.1023/A:1020715626538>
- [61] “Module : Data — skimage v0.18.dev0 docs.” [Online]. Available : <https://scikit-image.org/docs/dev/api/skimage.data.html?highlight=astro#skimage.data.astronaut>
- [62] E. Kazemi, S. H. Hassani, and M. Grossglauser, “Growing a Graph Matching from a Handful of Seeds,” *Proc. VLDB Endow.*, vol. 8, no. 10, pp. 1010–1021, 2015. [Online]. Available : <https://doi.org/10.14778/2794367.2794371>
- [63] Z. Zhai, M. Staring, H. Ota, and B. Stoel, “Pulmonary Vessel Tree Matching for Quantifying Changes in Vascular Morphology : 21st International Conference, Granada, Spain, September 16–20, 2018, Proceedings, Part II,” 2018, pp. 517–524.
- [64] K. Riesen, M. Ferrer, and H. Bunke, “Approximate Graph Edit Distance in Quadratic Time,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1–1, 2014. [Online]. Available : <https://ieeexplore.ieee.org/document/7264987/>
- [65] M. F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, and S. Dickinson, “Object Recognition as Many-to-Many Feature Matching,” *International Journal of Computer Vision*, vol. 69,

- no. 2, pp. 203–222, 2006. [Online]. Available : <http://link.springer.com/10.1007/s11263-006-6993-y>
- [66] Y. Boykov and G. Funka-Lea, “Graph Cuts and Efficient N-D Image Segmentation,” *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006. [Online]. Available : <http://link.springer.com/10.1007/s11263-006-7934-5>
- [67] Y. Boykov and V. Kolmogorov, “An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision,” vol. 26, no. 9, p. 34, 2004.
- [68] X. He, J. Fronheiser, P. Zhao, Z. Hu, S. Uppal, X. Wu, Y. Hu, R. Sporer, L. Qin, R. Krishnan, E. M. Bazizi, R. Carter, K. Tabakman, A. K. Jha, H. Yu, O. Hu, D. Choi, J. G. Lee, S. B. Samavedam, and D. K. Sohn, “Impact of aggressive fin width scaling on FinFET device characteristics,” in *2017 IEEE International Electron Devices Meeting (IEDM)*, Dec. 2017, pp. 20.2.1–20.2.4.
- [69] O. Ronneberger, P. Fischer, and T. Brox, “U-Net : Convolutional Networks for Biomedical Image Segmentation,” *arXiv :1505.04597 [cs]*, May 2015. [Online]. Available : <http://arxiv.org/abs/1505.04597>
- [70] A. Buades, B. Coll, and J.-M. Morel, “Non-Local Means Denoising,” *Image Processing On Line*, vol. 1, pp. 208–212, Sep. 2011. [Online]. Available : http://www.ipol.im/pub/art/2011/bcm_nlm/
- [71] “BrainWeb : Simulated Brain Database.” [Online]. Available : <https://brainweb.bic.mni.mcgill.ca/>
- [72] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, G. B. Pike, and A. C. Evans, “BrainWeb : Online Interface to a 3D MRI Simulated Brain Database,” *NeuroImage*, vol. 5, p. 425, 1997.

Bibliographie

- [73] R.-S. Kwan, A. Evans, and G. Pike, “MRI simulation-based evaluation of image-processing and classification methods,” *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1085–1097, Nov. 1999.
- [74] R. K. S. Kwan, A. C. Evans, and G. B. Pike, “An extensible MRI simulator for post-processing evaluation,” in *Visualization in Biomedical Computing*, ser. Lecture Notes in Computer Science, K. H. Höhne and R. Kikinis, Eds. Berlin, Heidelberg : Springer, 1996, pp. 135–140.
- [75] D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes, and A. C. Evans, “Design and construction of a realistic digital brain phantom,” *IEEE transactions on medical imaging*, vol. 17, no. 3, pp. 463–468, Jun. 1998.
- [76] “TechInsights Inc. - Semiconductor Analysis & IP Services.” [Online]. Available : <https://www.techinsights.com/>
- [77] C. D. KUGLIN, “The Phase Correlation Image Alignment Method,” *Proc. Int. Conference Cybernetics Society*, pp. 163–165, 1975. [Online]. Available : <https://ci.nii.ac.jp/naid/80013746774/>
- [78] Z. Yang, “Fast Template Matching Based on Normalized Cross Correlation with Centroid Bounding,” in *2010 International Conference on Measuring Technology and Mechatronics Automation*, vol. 2, Mar. 2010, pp. 224–227.
- [79] G. Wolberg and S. Zokai, “Robust image registration using log-polar transform,” in *Proceedings 2000 International Conference on Image Processing (Cat. No.00CH37101)*, vol. 1, Sep. 2000, pp. 493–496 vol.1.
- [80] P. Dokládál and E. Dokládálová, “Computationally efficient, one-pass algorithm for morphological filters,” *Journal of Visual Communication and Image Representation*, vol. 22, no. 5, pp.

- 411–420, Jul. 2011. [Online]. Available : <https://linkinghub.elsevier.com/retrieve/pii/S1047320311000393>
- [81] Y. Cheng, “Mean shift, mode seeking, and clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, Aug. 1995.
- [82] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, Dec. 2001, pp. I–I.
- [83] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1. San Diego, CA, USA : IEEE, 2005, pp. 886–893. [Online]. Available : <http://ieeexplore.ieee.org/document/1467360/>
- [84] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *arXiv :1311.2524 [cs]*, Oct. 2014. [Online]. Available : <http://arxiv.org/abs/1311.2524>
- [85] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks,” *arXiv :1506.01497 [cs]*, Jan. 2016. [Online]. Available : <http://arxiv.org/abs/1506.01497>
- [86] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once : Unified, Real-Time Object Detection,” *arXiv :1506.02640 [cs]*, May 2016. [Online]. Available : <http://arxiv.org/abs/1506.02640>
- [87] X. Wang, T. E. Huang, T. Darrell, J. E. Gonzalez, and F. Yu, “Frustratingly Simple Few-Shot Object Detection,”

Bibliographie

- arXiv* :2003.06957 [cs], Mar. 2020. [Online]. Available : <http://arxiv.org/abs/2003.06957>
- [88] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN Features off-the-shelf : An Astounding Baseline for Recognition,” *arXiv* :1403.6382 [cs], May 2014. [Online]. Available : <http://arxiv.org/abs/1403.6382>
- [89] F. Chollet, “Xception : Deep Learning with Depthwise Separable Convolutions,” *arXiv* :1610.02357 [cs], Apr. 2017. [Online]. Available : <http://arxiv.org/abs/1610.02357>
- [90] F.-F. Li, “Crowdsourcing, benchmarking & other cool things,” p. 64, 2010.
- [91] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, “Selective Search for Object Recognition,” *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, Sep. 2013. [Online]. Available : <http://link.springer.com/10.1007/s11263-013-0620-5>
- [92] C. L. Zitnick and P. Dollár, “Edge Boxes : Locating Object Proposals from Edges,” in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham : Springer International Publishing, 2014, vol. 8693, pp. 391–405. [Online]. Available : http://link.springer.com/10.1007/978-3-319-10602-1_26
- [93] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition,” *arXiv* :1406.4729 [cs], vol. 8691, pp. 346–361, 2014. [Online]. Available : <http://arxiv.org/abs/1406.4729>
- [94] R. Girshick, “Fast R-CNN,” *arXiv* :1504.08083 [cs], Sep. 2015. [Online]. Available : <http://arxiv.org/abs/1504.08083>

- [95] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *arXiv :1409.1556 [cs]*, Apr. 2015. [Online]. Available : <http://arxiv.org/abs/1409.1556>
- [96] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *arXiv :1512.03385 [cs]*, Dec. 2015. [Online]. Available : <http://arxiv.org/abs/1512.03385>
- [97] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” *arXiv :1608.06993 [cs]*, Jan. 2018. [Online]. Available : <http://arxiv.org/abs/1608.06993>
- [98] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet : AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size,” *arXiv :1602.07360 [cs]*, Nov. 2016. [Online]. Available : <http://arxiv.org/abs/1602.07360>
- [99] B.-H. Huang and S.-H. Chang, “10 nm vs 10 nm technology node comparison,” p. 9.
- [100] “IEDM 2017 : GlobalFoundries 7nm process ; Cobalt, EUV,” Dec. 2017. [Online]. Available : <https://fuse.wikichip.org/news/641/iedm-2017-globalfoundries-7nm-process-cobalt-euv/>
- [101] “Field Emission Transmission Electron Microscope HF-3300 Hitachi High Technologies | Quantum Design.” [Online]. Available : <https://qd-europe.com/fr/fr/produit/field-emission-transmission-electron-microscope-hf-3300/>
- [102] I. Ortega-Oller, M. Padiál-Molina, P. Galindo-Moreno, F. O’Valle, A. Belén, and J. Peula-García, “Bone regeneration from PLGA micro-nanoparticles,” *BioMed Research International*, vol. 2015, Oct. 2015.
- [103] F. Dou, C. Small, F. Provencher, J. Ferreira, X. Wang, E. Rezasoltani, H. Liu, and H. Liu, “Particle plasmon-induced charge trapping

Bibliographie

- at heterointerfaces in PCDTBT :PC 70 BM blends,” *Journal of Polymer Science Part B : Polymer Physics*, vol. 55, Mar. 2017.
- [104] “Nanotechnology Now - Press Release : NanoSight nanoparticle detection technique contributes to virus production to counter MRSA.” [Online]. Available : http://www.nanotech-now.com/news.cgi?story_id=31605
- [105] “Metrics - Average Precision and mAP.” [Online]. Available : <https://innerpeace-wu.github.io/2018/03/22/Metrics-Average-Precision/>
- [106] “The PASCAL Visual Object Classes Challenge 2012 (VOC2012).” [Online]. Available : <http://host.robots.ox.ac.uk/pascal/VOC/voc2012/>
- [107] C. Xu and J. Prince, “Gradient vector flow : A new external force for snakes,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 1997, pp. 66–71.
- [108] “Object Detection — OpenCV 2.4.13.7 documentation.” [Online]. Available : https://docs.opencv.org/2.4/modules/imgproc/doc/object_detection.html?highlight=matchtemplate#matchtemplate