

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE DE LA MAÎTRISE EN
MATHÉMATIQUES ET INFORMATIQUE APPLIQUÉES

PAR
EMANUEL GUTIÉRREZ LIRA

CLASSIFICATION AUTOMATIQUE DES DIATOMÉES :
EXTRACTION ET IDENTIFICATION DES STRUCTURES
TAXONOMIQUES

OCTOBRE 2018

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

« Nous piétinerons éternellement aux frontières de l'inconnu, cherchant à comprendre ce qui restera toujours incompréhensible. Et c'est précisément cela qui fait de nous des hommes. »

Isaac Asimov

*À Gabriela Carolina,
mon épouse.*

Résumé

Les diatomées sont des algues unicellulaires de grand intérêt dans les domaines de la paléontologie, de l'écologie aquatique et de la médecine légale, entre autres. À ce jour, plus de 100 000 espèces sont connues et distribuées dans les divers écosystèmes aquatiques de la planète. Pour cette raison, il y a un grand intérêt à classifier automatiquement les images de diatomées. La procédure préliminaire d'identification automatique d'une espèce de diatomée est une étape complexe qui consiste à isoler le spécimen de l'arrière-plan de l'image. Dans ce travail, nous proposons une méthode de segmentation et une procédure d'extraction d'objets pour isoler la diatomée de l'arrière-plan. De plus, nous proposons des méthodes de détection de caractéristiques taxonomiques des diatomées tel que le raphé, les stries et les cercles concentriques. Tout d'abord, nous segmentons l'image en cherchant deux seuils optimaux dans l'histogramme des niveaux de gris qui seront déterminés à partir de la fonction de répartition de cet histogramme. Puis, nous éliminons, à l'aide d'un critère inspiré de la force d'attraction des corps, les régions pouvant ne pas appartenir à la diatomée afin de les éliminer du fond. Par la suite, nous construisons le contour de la diatomée par des raffinements successifs de l'enveloppe convexe des régions retenues. Ce contour constitue une approximation polygonale de la forme générale de la diatomée qui permet l'extraction du spécimen. Des caractéristiques taxonomiques, comme le raphé et les stries, sont trouvées en cherchant des segments continus à partir des intensités du gradient de l'image. D'autres caractéristiques, comme les cercles et les stries radiales, sont détectées à partir d'une analyse en coordonnées polaires des structures internes de la diatomée.

Abstract

The diatoms are unicellular algae of great interest in paleontology, aquatic ecology, and forensic medicine, among others. Currently, there are more than 100 000 known species distributed in aquatic ecosystems. For that reason, there is a big interest in the automatic classification of diatom images, however, the preliminary process applied to isolate the diatom from the background is a complex task. In this work, we propose a segmentation method and an object-extraction procedure to isolate the diatom from the background. We also propose a method to detect the taxonomic features of the diatoms such as the raphe, the striations and the concentric circles. First, we segment the image by searching the optimal thresholds in the histogram based on its cumulative distribution function. Then, we eliminate, under a gravitational force criteria, all regions other than those that could be part of the diatom. Afterwards, we approximate the contour of the diatom by successive refinements to the initial convex hull of all remaining regions. This contour is a polygonal approximation of the general shape of the diatom which is used to extract the specimen from the background. The taxonomic features as the raphe and the striations are found by searching continuous segments in the image gradient. Furthermore, the concentric circles and the radial striations are detected by analyzing the polar coordinates of the internal image contours or structures.

Remerciements

Mes remerciements vont premièrement à tous les membres de ma famille, spécialement à ma mère qui m'a donné son soutien et qui m'a encouragé à effectuer mes études de cycle supérieur à l'étranger. J'ai apprécié leur proximité malgré mon éloignement afin que je ne me sente pas trop loin de chez moi. Je veux aussi remercier ma fiancée Gabriela de m'avoir accompagné pendant mon séjour à l'étranger et de l'aide qu'elle m'a apportée.

Je remercie le programme Mitacs et la Fondation de l'UQTR pour le financement reçu lors de mes études de deuxième cycle en mathématiques et d'informatique à l'Université du Québec à Trois-Rivières (UQTR), sans leur contribution je n'aurais pu réaliser ce projet de recherche. Finalement, je ne peux passer sous silence les précieux conseils des professeurs Fathallah Nouboud et Alain Chalifour du département de mathématiques et d'informatique de l'UQTR et du professeur Yvon Voisin de l'Université de Bourgogne qui m'ont invité à participer à ce projet et à entreprendre un programme de maîtrise à l'Université du Québec à Trois-Rivières. Ils m'ont apporté de multiples conseils et idées pour trouver le bon chemin afin de relever les défis de cette recherche.

Merci à tous!

Table des matières

Introduction	1
1 Les diatomées	3
1.1 Anatomie	3
1.2 Aspects biologiques	5
1.3 Taxonomie des diatomées	6
1.4 Intérêts scientifiques et applications	6
1.5 Conclusion	8
2 État de l'art	9
2.1 Survol sur la classification d'images de diatomées	9
2.2 Conclusion	11
3 Traitement des images : un survol des méthodes	12
3.1 Opérateurs morphologiques	12
3.2 Filtres	16
3.3 Segmentation	20
3.4 Conclusion	24
4 Pré-traitement des images et extraction des caractéristiques	26
4.1 Pré-traitement des images	26
4.2 Extraction des caractéristiques	39
4.3 Conclusion	46

5 Résultats	47
5.1 Extraction de la diatomée	47
5.2 Identification des caractéristiques	50
5.3 Conclusion	51
Conclusion et travaux futurs	52
A Mathématiques de base	55
B Implémentation	59
B.1 Prétraitement des images	59
B.1.1 Extraction de la diatomée	64
B.1.2 Amélioration du contraste	65
B.1.3 Rotation et recadrage	65
B.2 Extraction des caractéristiques	67

Table des figures

1.1	Exemples de diatomées.	3
1.2	Structure des diatomées.	4
1.3	Classification générale des diatomées	5
3.1	Éléments structurants	13
3.2	Érosion et dilatation	15
3.3	Ouverture et fermeture	16
3.4	Filtre gaussien	17
3.5	Filtre de Canny	19
3.6	Égalisation d'histogramme	20
3.7	Histogramme équilibré	22
3.8	Otsu	22
3.9	Segmentation par K-moyennes	23
3.10	Segmentation unimodal (Paul Rosin)	24
4.1	Réduction du bruit : filtre gaussien	27
4.2	Exemples d' images de diatomées.	28
4.3	Prétraitement des images de diatomées : segmentation.	29
4.4	Prétraitement des images de diatomées : ouverture et fermeture	30
4.5	Prétraitement des images de diatomées : élimination des débris	32
4.6	Détection du contour : contour primitif	33
4.7	Détection du contour : enveloppe convexe	34
4.8	Diagramme des longueurs des segments de l'enveloppe convexe	34

4.9	Correction des segments atypiques dans l'enveloppe convexe	35
4.10	Raffinement du contour	36
4.11	Extraction de la diatomée	37
4.12	Amélioration du contraste	37
4.13	Rotation et recadrage	38
4.14	Histogramme des intensités du gradient orienté des contours de Canny	41
4.15	Détection du raphé	42
4.16	Détection des stries	42
4.17	Exemples de diatomées centrales	43
4.18	Détection des cercles et des stries d'une diatomée.	45
5.1	Résultats d'extraction des spécimens de diatomée (a)	48
5.2	Résultats d'extraction des spécimens de diatomée (b)	49
5.3	Résultats d'extraction des spécimens de diatomée (c)	50

Introduction

Les diatomées sont des algues unicellulaires que l'on retrouve dans les milieux aquatiques. Leur structure consiste en une cellule unique, le *frustule*, formée de deux valves qui s'emboîtent. Étant donné leurs propriétés biologiques, elles sont d'excellents bioindicateurs et de grand intérêt dans plusieurs domaines scientifiques. Dans le cadre de plusieurs applications scientifiques utilisant les propriétés des diatomées, il est parfois nécessaire d'identifier à l'espèce des spécimens de diatomées, et ce parmi plus de 100 000 espèces connues et distribuées dans les écosystèmes aquatiques de la planète. L'identification des diatomées, qui est faite généralement par des spécialistes à l'aide d'un microscope binoculaire, s'effectue sur la base de patterns spécifiques tel la géométrie du contour et des ornements du frustule.

Ce projet a pour but de contribuer à la classification à l'espèce des diatomées, de manière automatique ou semi automatique, dans des conditions réelles de laboratoire. L'approche courante dans un tel cas consiste à extraire les caractéristiques nécessaires à la classification tel que la géométrie globale des spécimens (contour), le *raphé*, les stries internes ou tout autre structure interne spécifique. Dans un premier temps, nous devons extraire les diatomées du fond bruité de l'image, saisie à l'aide d'un binoculaire. À cet effet, nous segmentons l'image, afin de séparer le fond de la zone d'intérêt, en déterminant automatiquement deux seuils (optimaux) de séparation des niveaux de gris (intervalle) à partir de la fonction de répartition obtenue à partir de l'histogramme des niveaux de gris dans l'image. Il résulte de cette segmentation une image binaire contenant différentes régions, dont certaines appartiennent

au domaine de la diatomée et d'autres à l'arrière plan de l'image. Pour chacune des régions segmentées, nous utilisons une valeur inspirée de la force d'attraction entre une région et celle ayant la plus grande surface, considérée comme étant la région principale appartenant à la diatomée. Ceci nous permet de déterminer si la région appartient ou non à la diatomée. Par la suite, nous construisons l'enveloppe convexe de l'ensemble des régions retenues pour reconstruire l'entièreté de la diatomée. Finalement, à partir de cette enveloppe convexe (contour initial), nous obtenons le contour final de la diatomée en effectuant des raffinements successifs.

Après cette étape d'extraction, nous obtenons une image approximative qui respecte la forme globale de la diatomée. Les travaux de [22] ont démontré qu'une classification de groupements d'espèces basée sur la géométrie du contour est prometteuse. Dans ce mémoire, nous mettons au point des méthodes d'extraction de caractéristiques supplémentaires à celles testées par [22] qui permettraient de classer à l'espèce les images de diatomées.

Dans le chapitre 1, nous décrivons brièvement les aspects biologiques relatifs aux diatomées, soit leur anatomie et l'habitat, les modes de reproduction et des applications utilisant les diatomées. Dans le chapitre 2, nous parcourons quelques travaux portant sur la classification de diatomées. Dans le chapitre 3, nous introduisons les concepts du traitement d'images qui seront utilisés tout au long de ce projet de recherche, en particulier la morphologie mathématique, la construction de filtres, la segmentation et les transformations mathématiques utilisées pour améliorer la qualité des images. Dans le chapitre 4, nous décrivons les méthodes de segmentation d'images et d'extraction d'objets développées pour extraire les diatomées de l'arrière-plan de l'image. Nous décrivons aussi quelques méthodes qui servent à extraire les caractéristiques des diatomées. Dans le chapitre 5, nous présentons les résultats obtenus et finalement, dans le dernier chapitre, nous présentons nos conclusions et les travaux futurs.

Les diatomées

1.1 Anatomie

Les diatomées, aussi appelées *Bacillariophycées*, sont des microorganismes unicellulaires photosynthétiques. Les diatomées appartiennent à la famille des *Hétérocontes* ou *Straménopiles*, qui inclut aussi les algues brunes. On les retrouve seules ou en chaînes en grand nombre dans l'eau de mer ou dans les eaux douces [12]. Des spécimens de diatomées sont illustrés à la figure 1.1.

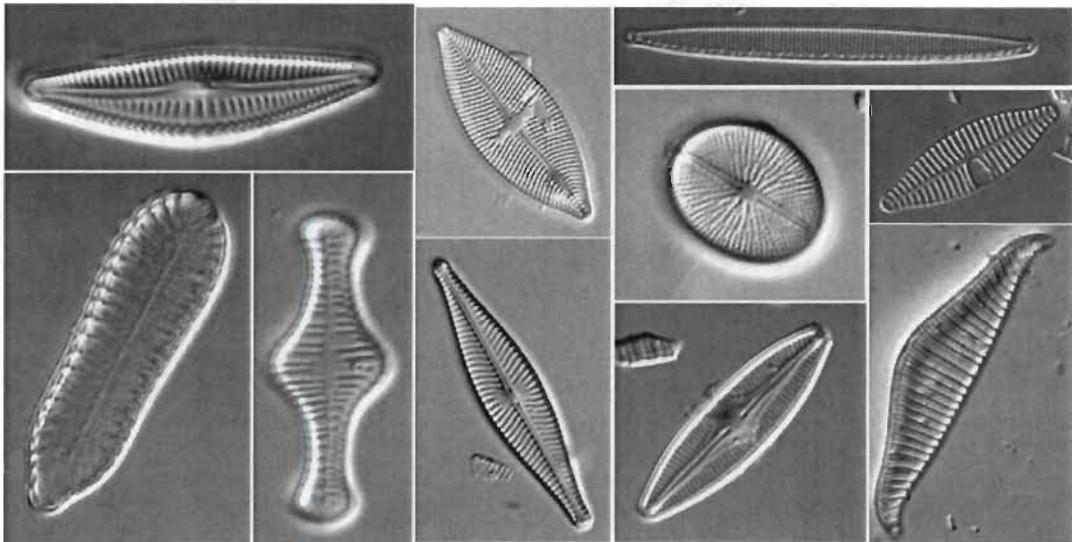


FIGURE 1.1 – Exemples de diatomées.

Structure

La taille des diatomées se situe aux environs de 20 à 200 μm , mais certaines peuvent atteindre une taille de 2 mm [14]. Les diatomées sont constituées d'une seule cellule appelée *frustule* et elles se distinguent des autres algues par une paroi rigide de silice hydratée. Cette paroi est en fait une enveloppe de cellulose constituée de deux valves qui s'emboîtent. L'*hypothèque*, la plus petite des deux valves, s'emboîte dans l'*épithèque* comme une boîte et son couvercle. La bordure verticale de l'épithèque, appelée *épicingulum*, recouvre et cache le bord de l'hypothèque, appelée *hypocingulum* (voir figure 1.2). Les valves présentent des ornements particuliers qui sont des structures à détecter dans nos images. Les motifs internes au contour et la géométrie du frustule sont les principales caractéristiques utilisées lors de la classification des espèces de diatomées.

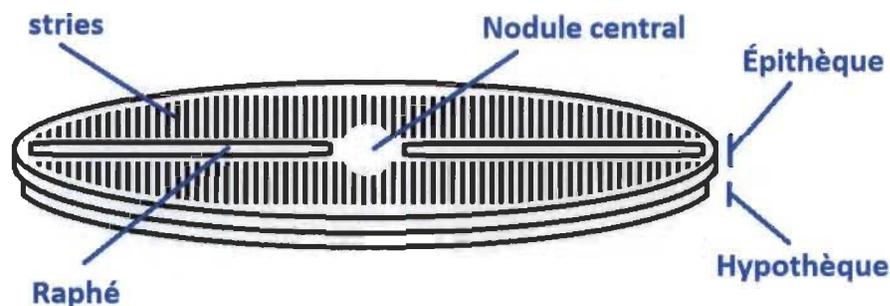


FIGURE 1.2 – Structure d'une diatomée pennale (*Naviculaceae*).

Classification

Comme mentionné ci-haut, certaines caractéristiques morphologiques des diatomées servent de paramètres lors de leur identification. La forme générale du frustule nous permet de distinguer, au départ, deux grandes catégories de diatomées : les centrales et les pennales (voir figure 1.3). Les diatomées **centrales** (ou *centriques*) présentent une symétrie radiale et elles ont un frustule circulaire portant des stries ou des alignements de punctuations qui rayonnent à partir d'un point, une aréole ou une réticulation qui n'est pas forcément située au centre de la valve. Les diatomées **pennales** présentent une symétrie bilatérale et elles ont un frustule allongé présentant des stries (ou des punctuations alignées) disposées autour d'un plan de

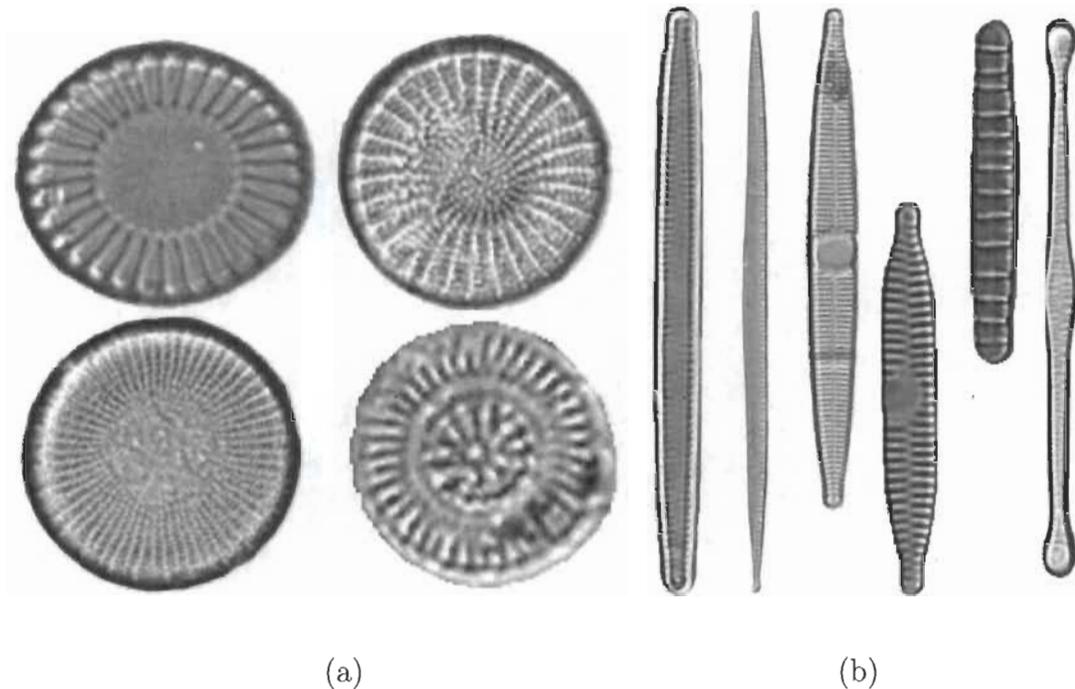


FIGURE 1.3 – (a) Exemples de diatomées centrales (b) Exemples de diatomées pennales.

symétrie bilatérale. Un grand nombre de diatomées pennales ont une « cavité » longitudinale, appelée *raphé*, laquelle est interrompue par un nodule central de silice et possédant deux autres nodules terminaux à chaque extrémité. Le raphé est un lieu d'échanges avec le milieu extérieur, dont l'excrétion d'un mucilage. La présence d'un raphé permet aussi de distinguer les espèces de diatomées. À titre d'exemple, si le raphé est atrophié, ou peu marqué, on l'appelle *pseudo-raphé*. Par ailleurs, les diatomées pennales sans raphé sont aussi appelées *araphidées* ou *cypto-raphidées*.

1.2 Aspects biologiques

Les diatomées ont une activité photosynthétique. Elles se développent principalement dans la zone euphotique (ou épipélagique) des eaux marines fraîches et froides, mais aussi dans les sols humides [7]. Leur cycle de vie est principalement diphasique, c'est-à-dire que les cellules diploïdes se multiplient par mitose pendant plusieurs mois, parfois pendant plusieurs années. Lors de la mitose, chacune des valves de la diatomée « parent » devient l'épithèque d'une nouvelle diatomée fille qui secrète l'hypothèque correspondante.

Après plusieurs divisions successives, les diatomées deviennent de plus en plus petites. Or, cette réduction de la taille atteint une limite. Lorsque les diatomées atteignent environ 30 % de leur taille initiale, elles entrent dans un état de méiose et produisent des gamètes. Lors de cette étape, la paroi cellulaire ne comporte pas de frustule siliceux. Le zygote résultant de la fusion des gamètes croît jusqu'à une taille maximale propre à l'espèce, ou de la population, avant de former un nouveau frustule. Cette reproduction sexuée, appelée *auxosporulation*, permet à deux diatomées de petites tailles de former une nouvelle diatomée d'une taille plus grande.

1.3 Taxonomie des diatomées

L'identification des diatomées est habituellement faite par des spécialistes de la biologie à l'aide d'un microscope, ou d'une loupe binoculaire, par l'observation des caractéristiques visibles du frustule. La taxonomie de ces organismes est principalement basée sur les éléments morphologiques suivants :

- **La forme générale du frustule** : géométrie et symétrie du frustule ;
- **Les stries** : forme des stries et leur densité ;
- **Le raphé** : présence ou absence, position et forme générale, des terminaisons.

Il existe d'autres caractéristiques telles que les ponctuations, les stigmas, les fibules, les côtes et les épines qui sont des paramètres servant à l'identification. Certaines de ces structures sont visibles à un grossissement de 1000x. Néanmoins, certains détails ne sont visibles qu'à l'aide d'un microscope électronique à balayage [13].

1.4 Intérêts scientifiques et applications

Les diatomées possèdent des attributs qui en font d'excellents bioindicateurs. Tout d'abord, elles sont présentes en abondance dans la plupart des écosystèmes aquatiques et humides sur la planète. Ces organismes sont de bons indicateurs environnementaux puisqu'ils sont sensibles à la qualité de l'eau et aux variations d'autres

paramètres dans leur écosystème. De plus, la structure siliceuse des diatomées est préservée, ou fossilisée, dans les sédiments aquatiques après leur mort, ce qui permet la conservation des ornements et des caractéristiques sur lesquelles s'appuie la taxonomie à des fins d'identification. Les plus vieux fossiles de diatomées mis à jour datent du début de la période du Crétacé. Ces fossiles contribuent à l'étude des changements environnementaux et géologiques sur de grandes échelles de temps [6].

L'abondance et la distribution de certaines espèces fossiles de diatomées ont permis d'effectuer des corrélations stratigraphiques entre des régions éloignées. Les diatomées sont très utilisées pour les reconstitutions paléo-environnementales et océanographiques. Les diatomées constituent une grande partie du phytoplancton des milieux aquatiques et contribuent largement à la fixation du dioxyde de carbone atmosphérique, et donc au cycle de carbone, ainsi qu'au cycle du silicium. Par ailleurs, les diatomées d'eaux douces sont de bons bio-indicateurs de la qualité des eaux, puisqu'elles colonisent les milieux humides tout au long de l'année et leurs réactions aux changements environnementaux sont connues.

Dans le domaine de la médecine légale, un diagnostic de noyade est parfois effectué grâce à l'identification des diatomées à l'intérieur de l'organisme de la personne décédée. Dans de telles circonstances, les diatomées présentes dans l'eau atteignent les poumons et envahissent le circuit sanguin. La présence de celles-ci dans les tissus humains permet d'affiner un diagnostic et de démontrer dans certains cas s'il y a eu un déplacement du corps et un contact avec un environnement humide. La *diatomite* est une roche légère, poreuse et friable formée par une grande accumulation de diatomées. Cette roche peut être exploitée par l'industrie comme abrasif léger, adjuvant de filtration, absorbant, isolant, stabilisant de la nitroglycérine dans la dynamite ou encore comme insecticide. Au Canada, les diatomées sont utilisées dans le cadre d'études paléolimnologiques lors d'études environnementales. D'autres travaux témoignent de leur importance et de leur utilisation dans des études sur le carbone organique dissous, du pH et de l'eutrophisation des rivières et des lacs [10].

1.5 Conclusion

Dans ce chapitre, nous avons introduit les caractéristiques biologiques des diatomées d'une façon générale. Cela nous a permis de connaître leur anatomie et d'autres aspects biologiques, qui seront utilisés dans ce mémoire. Nous signalé l'importance des diatomées dans plusieurs domaines scientifiques ainsi que dans l'industrie. Leur contribution dans des contextes environnementaux, en paléontologie et en médecine légale justifie l'importance de les étudier.

Dans le chapitre suivant, nous présentons un survol des travaux et des études portant sur l'identification et la classification automatiques des diatomées. Nous verrons différentes approches visant le traitement et la classification des images de diatomées. La plupart de ces travaux sont orientés vers l'analyse géométrique des formes des diatomées. Néanmoins, les structures internes sont aussi prises en considération dans certains travaux.

État de l'art

2.1 Survol sur la classification d'images de diatomées

La classification des images de diatomées a fait l'objet de multiples travaux. Un projet majeur est celui d'ADIAC (Automatic Diatom Identification And Classification). Ce projet européen a débuté en 1998 et a été financé par le programme européen MAST (Marine Science And Technology). Le but de ce projet était de développer des algorithmes de classification automatique des images de diatomées en utilisant le traitement d'images à partir de caractéristiques visuelles comme la forme du frustule et les ornementsations [11].

Dans [25], les auteurs ont utilisé les fréquences des intensités de niveaux de gris de l'image comme entrées d'un réseau neuronal dans leur modèle de classification. Par ailleurs, le contour des diatomées est une caractéristique importante dans la reconnaissance des espèces de diatomées. Wilkinson et al. [18] ont réalisé une analyse du contour en utilisant la courbure morphologique comme paramètre dans un modèle de classification non supervisé.

La classification automatique des diatomées a été réalisée en utilisant ImageJ, un logiciel de traitement d'images développé par le *National Institute of Health* [19].

Dans cette approche, la classification et le traitement des images de diatomées est faite de façon semi-automatique [4]. Dans une étape préliminaire, les auteurs ont défini une base de données des caractéristiques de diatomées échantillonnées dans les environs de Kénitra au Maroc. La base de données est organisée de la façon suivante ; les caractéristiques des diatomées sont stockées dans un vecteur indexé en fonction des espèces de diatomées possédant ces caractéristiques. Pendant la procédure d'identification, des caractéristiques comme le contour et les ornements sont extraites du spécimen à identifier et elles sont comparées aux vecteurs de la base de données. Tous les candidats n'ayant pas ces caractéristiques sont ignorés et l'analyse se poursuit avec les espèces restantes. Généralement, à la deuxième ou la troisième itération il ne reste qu'un ou deux candidats à identifier.

D'autres travaux ont utilisé des paramètres géométriques caractérisant la forme générale des diatomées et leur contour extérieur [22]. Dans ce travail, un vecteur des angles normaux au contour 8-connecté est calculé, normalisé et représenté par des descripteurs de Fourier. Ces descripteurs, en nombre limité, et des paramètres de forme, ont permis une classification satisfaisante de classes de diatomées (regroupements d'espèces de géométries similaires mais non identiques). Une autre approche de classification automatique a été réalisée par N. Claudon [5], laquelle consistait à tenir compte des structures internes des espèces à l'aide d'un vecteur de caractéristiques contenant les orientations et les fréquences des structures internes (transformée de Fourier bidimensionnelle) auquel s'ajoutent des variables booléennes (présence ou absence) d'un *raphé*, d'un *raphé centré* un d'un *nodule central*.

Dans [15], les auteurs ont utilisé un critère de similarité entre diatomées basé sur une pseudodistance prétopologique. Cette distance est obtenue par une relaxation de l'axiomatique d'une topologie. Dans [24], les auteurs ont développé un système de reconnaissance invariant à la rotation, à la position, au changement d'échelles, à la variation d'éclairage et au bruit en utilisant des masques d'anneaux binaires de Fourier.

2.2 Conclusion

Dans ce chapitre, nous avons présenté un survol des travaux sur la classification d'images de diatomées. Cela nous a permis de connaître les différentes approches qui ont été réalisées et les résultats obtenus. Nous pouvons observer que la géométrie du contour est une caractéristique importante pour différencier les espèces de diatomées. Nous observons que les structures internes ont été analysées par des méthodes utilisant la densité et la fréquence des stries. Néanmoins, ces approches n'ont pas, ou peu, abordé le prétraitement des images en vue de l'extraction des spécimens. Nous aborderons cet aspect plus spécifiquement dans ce mémoire.

Dans le chapitre suivant, nous présentons des concepts et des méthodes du traitement de l'image utilisés lors de l'extraction des diatomées dans des images bruitées.

Traitement des images : un survol des méthodes

Dans ce chapitre, nous présentons quelques concepts de base sur le traitement d'images qui sont utilisés dans ce mémoire.

3.1 Opérateurs morphologiques

La morphologie mathématique est une théorie qui a été principalement développée dans les années 60 par Georges Matheron [17] et Jean Serra [21]. Depuis, elle a été largement appliquée au traitement et à l'analyse d'images et a fait l'objet de développements théoriques importants [26]. La morphologie mathématique utilise, en l'occurrence, la théorie des ensembles pour décrire les formes, l'algèbre (définitions d'opérateurs, filtres), la topologie (amincissement, squelettisation), les probabilités, etc. Dans le traitement des images, la morphologie mathématique est utilisée pour étudier les interactions entre une image et ses caractéristiques, cela par l'utilisation d'éléments structurants et d'opérateurs morphologiques du plan discret [16].

On peut définir un ordre partiel entre deux images. Soient f et g deux images de même dimension, alors $f \leq g \iff f(x) \leq g(x), \forall x$ où x est un pixel dans une même position dans les images. Ici $f(x)$ et $g(x)$ sont les intensités de niveau de gris dans les deux images.

Un élément structurant, ou masque, est un ensemble fini du plan discret ayant une taille relativement petite par rapport à celle de l'image. Un élément structurant possède un *pixel de référence* lequel sera centré (déplacé) sur chacun des pixels de l'image. Les éléments structurants les plus utilisés sont le carré et la croix tel qu'illustré à la figure 3.1. Dans ces exemples, le pixel de référence est au centre de l'élément (en noir), mais la position du pixel de référence peut varier selon les applications. La taille d'un élément structurant est variable et dépend de la résolution désirée, ou des contraintes, lors du traitement des images dans une application donnée.

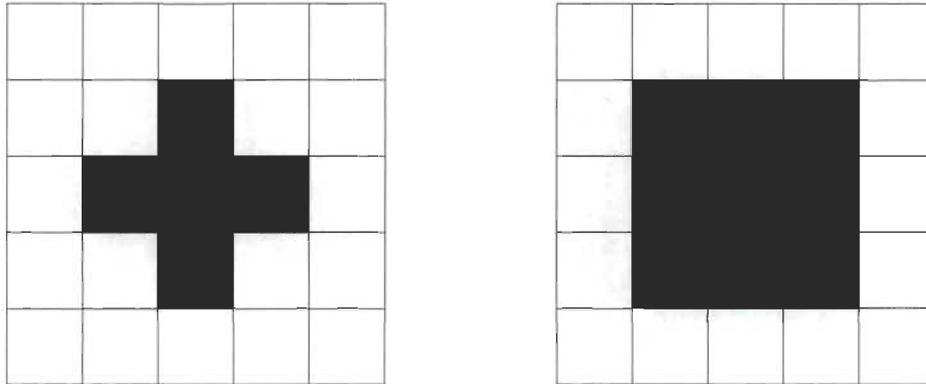


FIGURE 3.1 – Deux éléments structurants. À droite, la croix ; à gauche, le carré.

Soient $I(f)$ et $I(g)$ deux images quelconques représentées par leurs intensités de niveaux de gris $f(x)$ et $g(x)$ où x est un pixel dans une même position dans les images. On parlera des images, ou ensembles, f et g . Deux opérateurs morphologiques φ_1 et φ_2 sont dits *duaux* si l'image résultante obtenue par l'application de φ_1 à l'ensemble complément de f est égale au complément du résultat obtenu de l'application de φ_2 à f , c'est-à-dire,

$$\varphi_1(f^c) = (\varphi_2(f))^c, \forall f. \quad (3.1)$$

Un opérateur φ est dit *croissant* si pour toutes paires d'images f et g , on a ;

$$f \leq g \Rightarrow \varphi(f) \leq \varphi(g). \quad (3.2)$$

Un opérateur φ est dit *extensif* si $\varphi(f)$, l'image résultante de l'application de l'opérateur sur f , vérifie $f \leq \varphi(f), \forall f$. Sinon l'opérateur φ est dit *anti-extensif*,

c'est-à-dire si $\varphi(f) \leq f, \forall f$.

Un opérateur φ est dit *idempotent* si la condition suivante est vérifiée ;

$$\varphi(f) = \varphi(\varphi(f)), \quad \forall f. \quad (3.3)$$

Ces opérateurs sont généralement appliqués à un sous-ensemble X de pixels d'une image binaire. L'ensemble X , ou région X , est un ensemble de pixels blancs dans l'image et par conséquent X^c est un ensemble de pixels noirs.

Opérateurs d'érosion et de dilatation

Parmi les opérations morphologiques, deux sont fréquemment utilisées et fort utiles. Ce sont les opérateurs d'*érosion* et de *dilatation*. Soit X un sous-ensemble d'une image f et S un élément structurant donné, alors l'*érosion* de X par l'élément structurant S est l'ensemble ;

$$\varepsilon(f) = \{x \mid \forall s \in S, \quad x + s \in X\}. \quad (3.4)$$

L'opérateur dual de l'érosion est la *dilatation* et on obtient l'ensemble ;

$$\delta(f) = \{x + s \mid x \in X \text{ et } s \in S\}. \quad (3.5)$$

Les expressions érosion et dilatation décrivent bien les effets sur une image de ces deux opérateurs. Des exemples de ces opérations sont illustrés à la figure 3.2 avec comme élément structurant la croix (voir figure 3.1).

Étant donné que δ et ε sont des opérateurs duaux, alors $\varepsilon(f)^c = \delta(f^c)$. Par ailleurs, l'érosion n'est pas l'opérateur inverse de la dilatation et vice versa. En effet, puisque plusieurs images peuvent avoir le même ensemble construit suite à une érosion ou à une dilatation, il est impossible de retrouver les images originales après leurs transformations en l'absence de fonctions inverses.

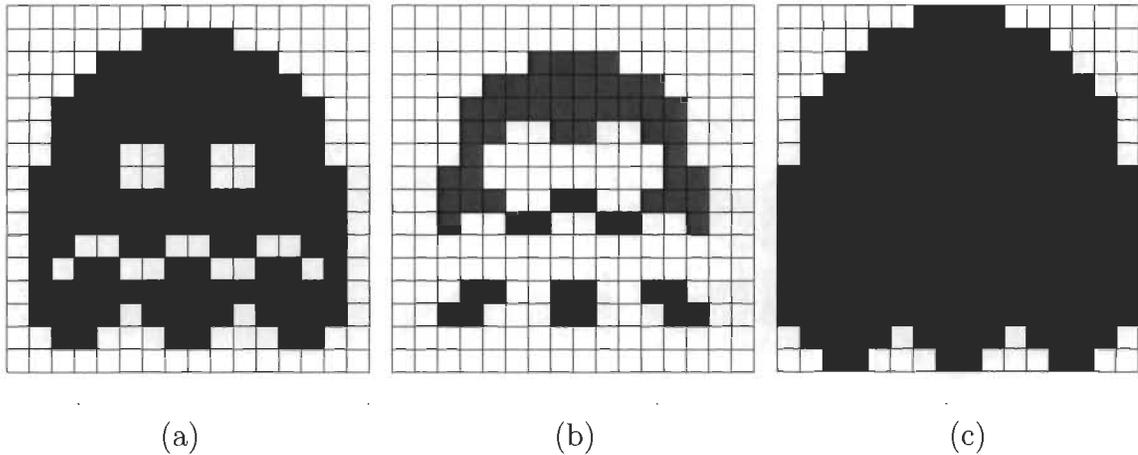


FIGURE 3.2 – Opérations morphologiques : (a) Image originale (b) Image découlant de l'érosion (c) Image découlant de la dilatation.

Ouverture et fermeture

L'érosion et la dilatation sont les opérateurs morphologiques les plus simples et si on les compose, ils peuvent produire des résultats d'une grande complexité. Leur application dans un ordre donné, définit les opérateurs morphologiques d'*ouverture* et de *fermeture*.

L'ouverture $\gamma(f)$ d'une image f est obtenue par une érosion suivie d'une dilatation, c'est-à-dire

$$\gamma(f) = \delta(\varepsilon(f)). \quad (3.6)$$

La fermeture $\phi(f)$, ou la clotûre, d'une image f est obtenue par une dilatation suivie d'une érosion, c'est-à-dire

$$\phi(f) = \varepsilon(\delta(f)). \quad (3.7)$$

Il est important, lors de l'application successive des opérateurs de fermeture et d'ouverture, d'utiliser le même élément structurant. Les opérations d'ouverture et de clôture sont d'une grande utilité en traitement de l'image. En l'occurrence, ces opérations permettent l'élimination du bruit sur les contours lors des étapes de prétraitement. L'ouverture accentue la frontière des objets tandis que la fermeture

permet la connexion de composantes non connexes. Des exemples d'utilisation de ces opérations sont illustrés à la figure 3.3 et pour lesquels l'élément structurant utilisé est la croix.

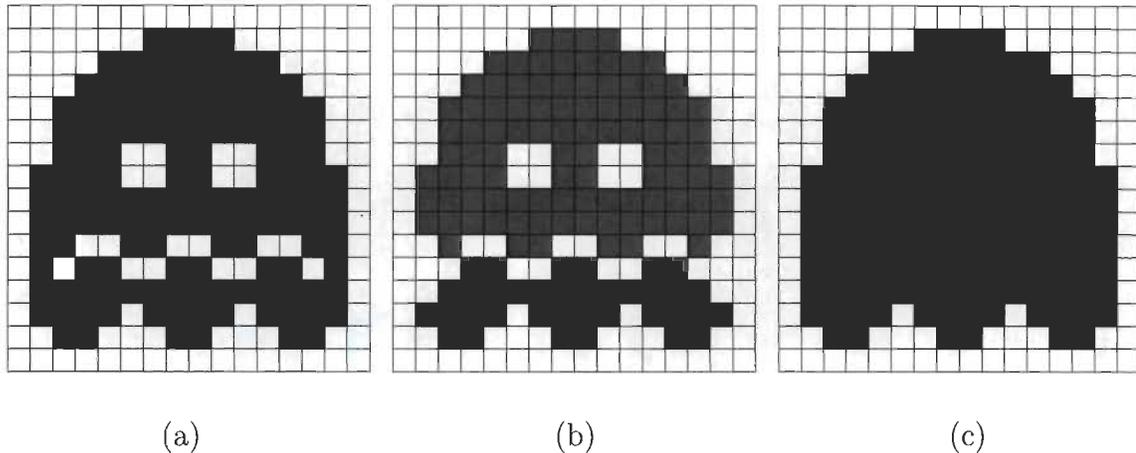


FIGURE 3.3 – Opérations morphologiques : (a) Image originale (b) Image découlant de l'ouverture (c) Image découlant de la clôture.

3.2 Filtres

Dans cette section, nous présentons une brève description des filtres utilisés dans le présent travail.

Filtre gaussien

Le filtre gaussien ou *gaussien blur* est une méthode de lissage très courante qui utilise une fonction gaussienne discrète dans le but de réduire le bruit et d'atténuer certains détails dans une image. Généralement, ce filtre est appliqué lors d'une étape de prétraitement. À la figure 3.4 nous montrons les résultats de l'application d'un filtre gaussien à une image en utilisant des fenêtres de tailles variées pour en apprécier les effets.

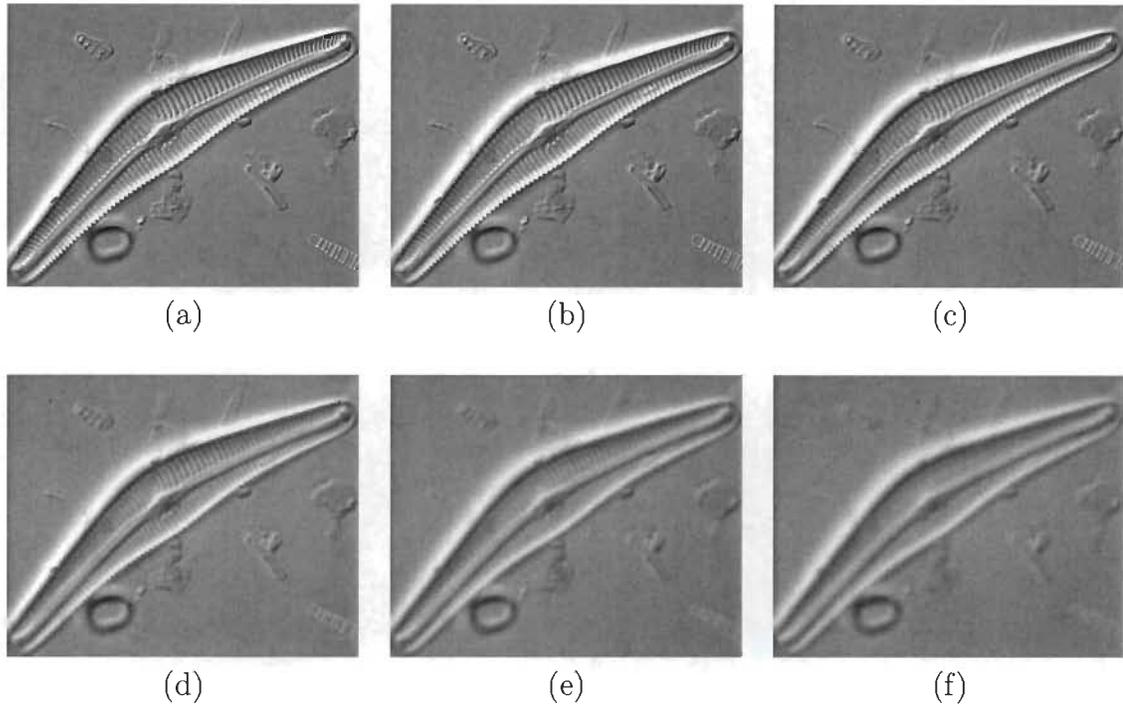


FIGURE 3.4 – (a) Image originale b) Image lissée avec une fenêtre 7x7, (c) 15x15, (d) 31x31, (e) 45x45 et (f) 57x57.

Filtre de Canny

Le filtre de Canny est utilisé comme détecteur de contours dans une image [3]. La détection des contours à l'aide du filtre de Canny s'effectue selon les 5 étapes ci-dessous.

1. **Application d'un filtre gaussien.** La détection de contours est fortement affectée par le bruit dans l'image. La première étape consiste à appliquer un filtre gaussien pour lisser l'image et éviter des faux positifs ;
2. **Gradient de l'image.** Cette méthode utilise quatre filtres afin de détecter les contours verticaux, horizontaux et diagonaux. Dans un premier temps, on calcule la dérivée première dans la direction horizontale ∇_x puis dans la direction verticale ∇_y . À partir du vecteur gradient, l'intensité et la direction de croissance maximale sont calculées ;

$$\nabla = \sqrt{\nabla_x^2 + \nabla_y^2} \quad (3.8)$$

$$\Theta = \arctan(G_y, G_x) \quad (3.9)$$

3. **Suppression du non-maximum.** La méthode de suppression du non-maximum est une méthode utilisée pour trouver les contours les plus significatifs. À partir du gradient obtenu à l'étape précédente, on supprime toutes les valeurs du gradient (en les mettant à zéro) à l'exception des maximums locaux. Cette étape permet de mettre en évidence les changements significatifs du gradient dans l'image. La méthode de suppression consiste à déplacer un masque dans l'image que l'on centre à chaque pixel. On compare la valeur au pixel central à celles dans les directions positive et négative des axes. Si la valeur du pixel est plus grande que celle des autres pixels le long de la direction considérée, la valeur est préservée, sinon la valeur est mise à zéro.
4. **Double seuillage.** La suppression du non-maximum n'est pas suffisante pour éliminer tout le bruit dans l'image du gradient. Un deuxième filtrage des intensités du gradient permet d'éliminer les faibles valeurs d'intensité et de préserver celles ayant une haute valeur. À cet effet, une borne inférieure et une borne supérieure de seuillage sont établies expérimentalement. Les pixels ayant une intensité de gradient inférieure à la borne inférieure sont éliminés, ceux ayant une valeur dans l'intervalle sont conservés et identifiés comme ayant une faible intensité et les autres sont étiquetés comme ayant une forte intensité.
5. **Détection de contours.** À cette étape on regarde quels sont les pixels qui vont former les contours. Les pixels dont l'intensité est "forte" sont considérés comme des pixels de contour. On vérifie si les pixels de faibles intensités sont connectés à un ou plusieurs pixels de forte intensité dans leur proche voisinage 8-connecté. Dans ce cas, les pixels sont considérés comme appartenant au contour.

La figure 3.5 illustre un exemple de contours obtenus avec le filtre de Canny.

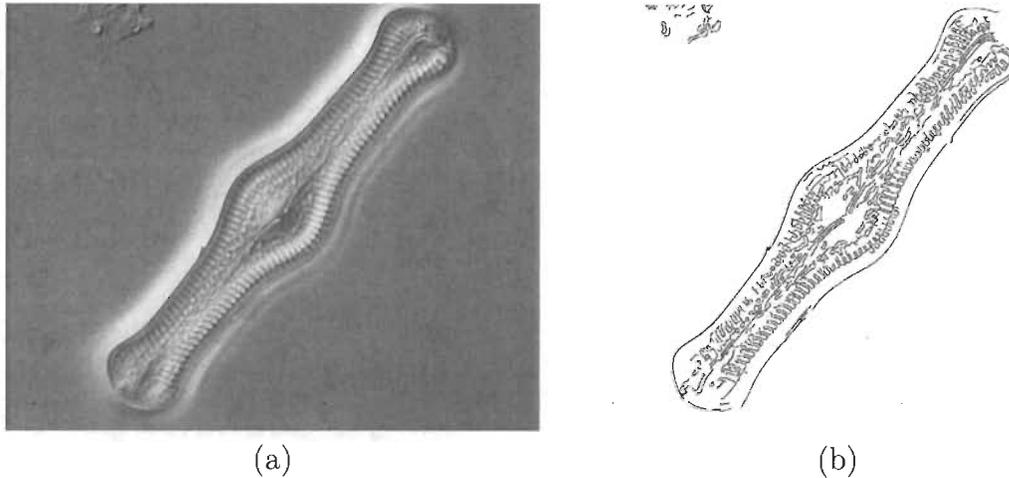


FIGURE 3.5 – (a) Image originale (b) Image des contours

Égalisation d’histogramme

L’égalisation d’histogramme est une méthode d’amélioration du contraste dans une image basée sur l’histogramme des niveaux de gris (intensités) de l’image. La méthode consiste à répartir d’une façon équilibrée les intensités sur l’ensemble des valeurs possibles en « étalant » (dilatation) l’histogramme. L’égalisation d’histogramme est utile pour ajuster le contraste des images dont les pixels sont d’intensités proches. La figure 3.6 illustre le résultat de la méthode d’égalisation d’histogramme. Nous observons que les détails du contour sont plus prononcés dans l’image après traitement. La contrepartie est l’augmentation du bruit dans certaines régions de l’image.

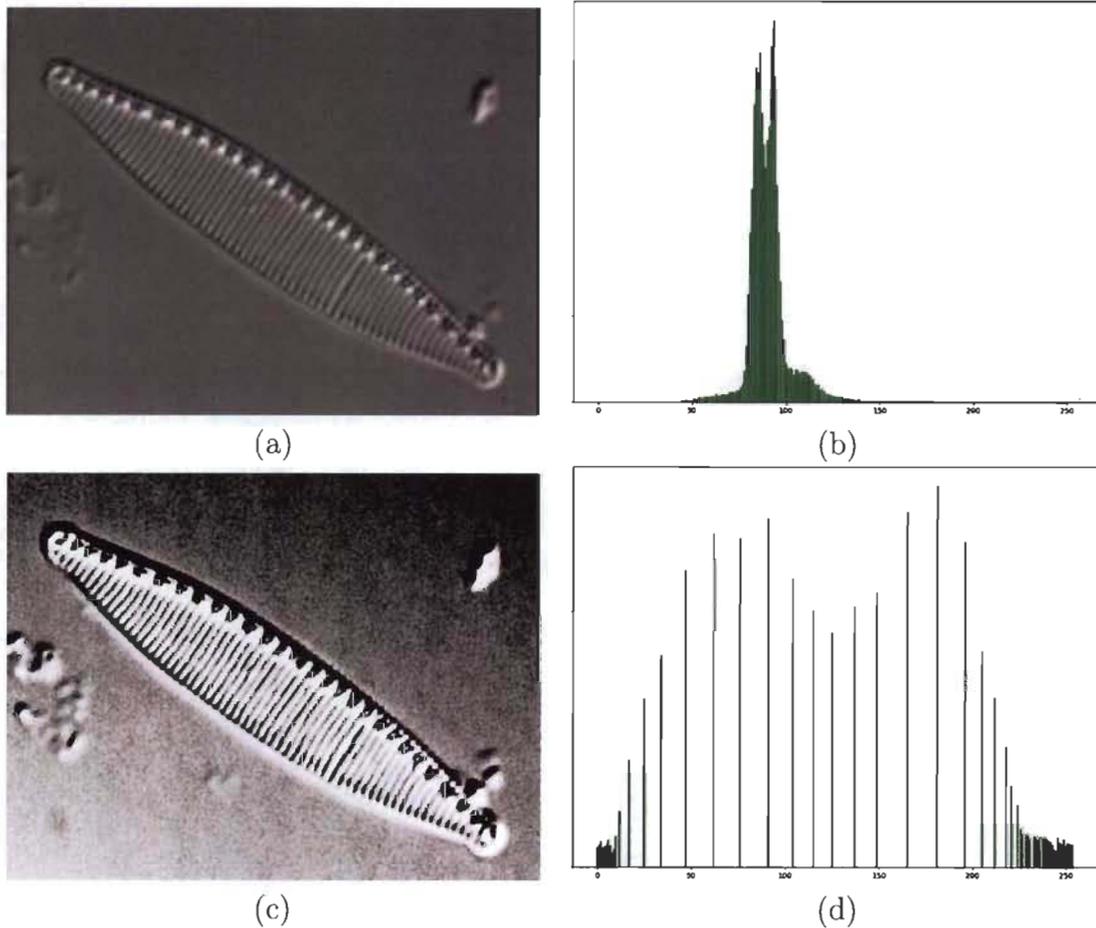


FIGURE 3.6 – (a) Image originale (b) Histogramme original (c) Résultat de l'égalisation (d) Histogramme égalisé.

3.3 Segmentation

La segmentation est une étape importante pour le traitement des images numériques. Elle peut avoir plusieurs objectifs dont l'extraction d'objets d'intérêt dans une image ou à titre de prétraitement pour favoriser l'extraction de caractéristiques, pour ne citer que celles-ci. Plusieurs méthodes de segmentation ont été développées au fil des années et certains filtres de base, comme ceux que nous avons vu dans la section précédente, sont largement utilisées pour favoriser l'étape de segmentation proprement dite.

De façon générale, une méthode de segmentation consiste à regrouper les pixels d'une image en régions suivant certains critères comme l'intensité des pixels ou les valeurs du gradient. Malheureusement, il n'existe pas de critère formel (déducible) qui permet de partager aisément, et automatiquement, les différentes régions d'une image. La majorité des méthodes existantes utilisent des hypothèses propres aux images traitées et au contexte de l'étude. Par exemple, la plupart des méthodes de segmentation supposent que l'ensemble des pixels de l'image appartient à deux classes : l'objet d'intérêt et l'arrière-plan. Dans cette section, nous présentons quelques méthodes de segmentation qui ont été testées pour segmenter les images de diatomées utilisées dans notre étude.

Histogramme équilibré

L'histogramme équilibré (*Balanced Histogram Thresholding*) est une méthode de segmentation très simple. La méthode consiste à définir un seuil comme point d'équilibre au milieu de l'histogramme des intensités de niveaux de gris et à enlever du poids aux extrémités jusqu'à ce que l'histogramme soit équilibré [1]. La procédure de segmentation se fait de façon itérative et en trois étapes :

1. Mettre le point d'équilibre au centre de l'histogramme ;
2. Calculer le poids (somme des valeurs de l'histogramme) de chaque côté du point d'équilibre ;
3. Supprimer les valeurs à l'extrémité du côté de plus grand poids (équilibrage).

Les itérations cessent lorsque les poids des deux côtés du point d'équilibre, ici le seuil, sont égaux ou lorsque les deux extrémités se rejoignent. Finalement, tous les pixels dont l'intensité est plus petite que le seuil appartiennent à une classe et le complémentaire à l'autre classe. Cette méthode est efficace lorsque l'histogramme présente deux pics dominants et distincts. À la figure 3.7 nous observons le résultat obtenu avec cette méthode sur une image de diatomée.

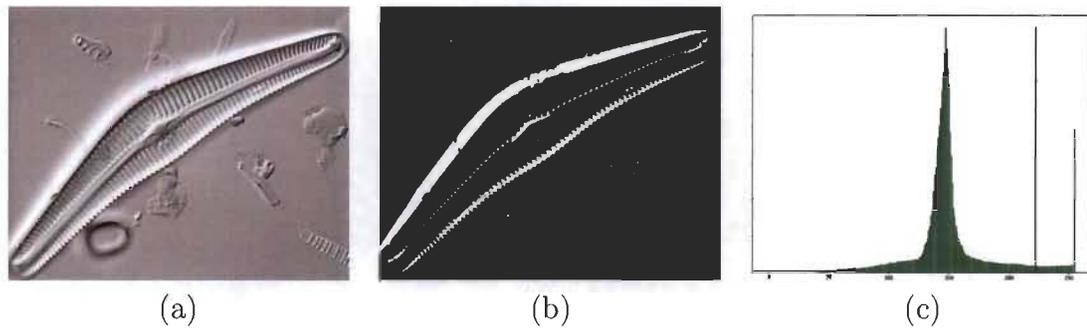


FIGURE 3.7 – (a) Image originale (b) Image segmentée (c) Histogramme de l'image.

Méthode d'Otsu

La méthode d'Otsu est basée sur l'existence de deux classes de pixels dans l'image et la variance intra-classe. Cette méthode s'avère efficace lorsque l'histogramme présente une distribution bimodale. La méthode a pour objectif la détermination d'un seuil t , idéalement optimal, en testant toutes les possibilités de seuil. Pour chaque valeur de t , la variance intra-classe est calculée. Le seuil optimal t_* est celui pour lequel la variance intra-classe est minimale. La figure 3.8 montre le résultat obtenu sur une image de diatomée.

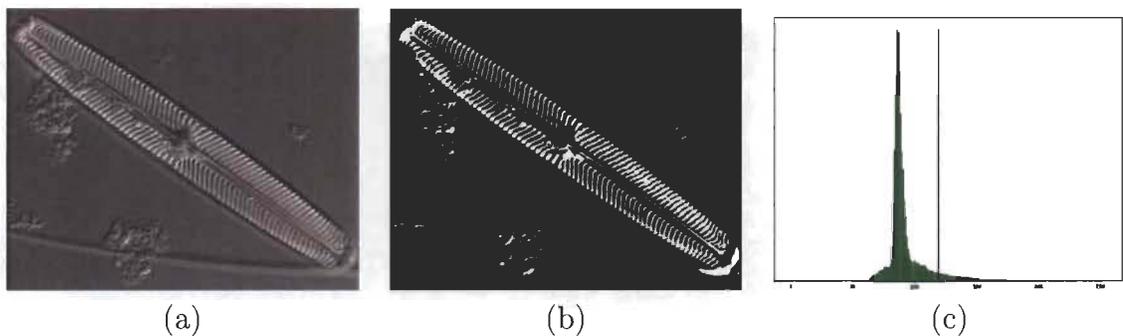


FIGURE 3.8 – (a) Image originale (b) Image segmentée (c) Histogramme de l'image.

Méthode des K-Moyennes

La méthode des k-moyennes est une méthode de *clustering* appliquée aux images pour regrouper des intensités de valeurs de pixels. À la différence des deux méthodes précédentes, la méthode des K-moyennes permet le regroupement des pixels dans

plus de deux classes. Nous présentons la méthode dans le cas de deux classes. La méthode est initialisée par le calcul de la moyenne des quatre pixels correspondant aux quatre coins de l'image, sous l'hypothèse que ces pixels appartiennent à l'arrière-plan, et le calcul de la moyenne du reste des pixels de l'image. On définit un premier seuil comme étant la moyenne arithmétique de ces deux valeurs (appelées centres). Par la suite, on met à jour successivement les centres en calculant les moyennes des pixels de chaque classe. Les itérations sont arrêtées lorsque la valeur des centres se stabilise, c'est-à-dire, que la différence entre le seuil calculé et celui de l'itération précédente est plus petite que 1 ou si un maximum d'itérations a été atteint.

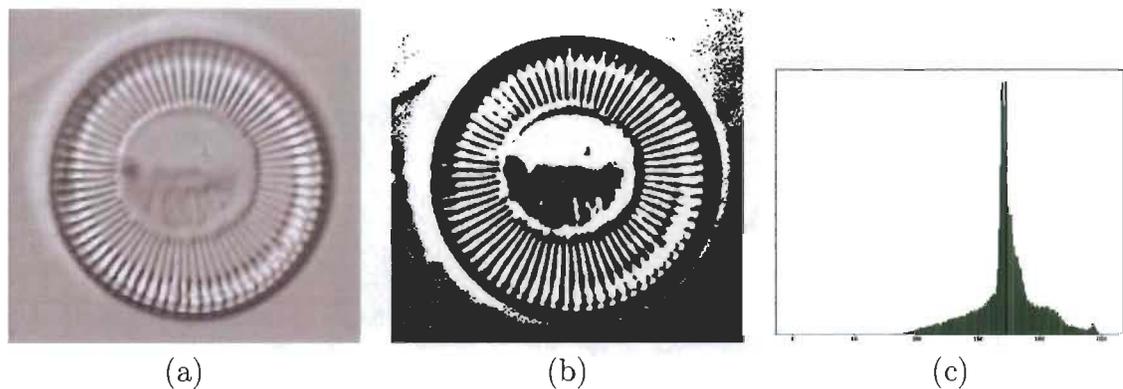


FIGURE 3.9 – (a) Image originale (b) Image segmentée (c) Histogramme de l'image.

Méthode de segmentation unimodale

Les méthodes que nous avons décrites ci-dessus sont des méthodes de segmentation basées sur l'hypothèse d'une distribution bimodale des intensités des niveaux de gris des pixels. Rosin [20] a proposé une méthode de segmentation basée sur une distribution unimodale des intensités des niveaux de gris. Dans ce cas, un seul pic dominant est présent dans l'histogramme des intensités, à gauche ou à droite de l'histogramme. La méthode consiste à calculer le seuil optimal au point d'inflexion de l'histogramme. Les intensités qui précèdent le seuil appartiendront à la première classe et le complémentaire, à la seconde classe. Afin de calculer ce seuil, nous traçons une droite reliant le point culminant du pic de l'histogramme à la première valeur non nulle à l'extrémité opposée de l'histogramme. Le seuil optimal est l'abscisse t_* tel que la distance orthogonale entre la ligne droite et la valeur de l'histogramme en

t_* est maximale (voir figure 3.10).

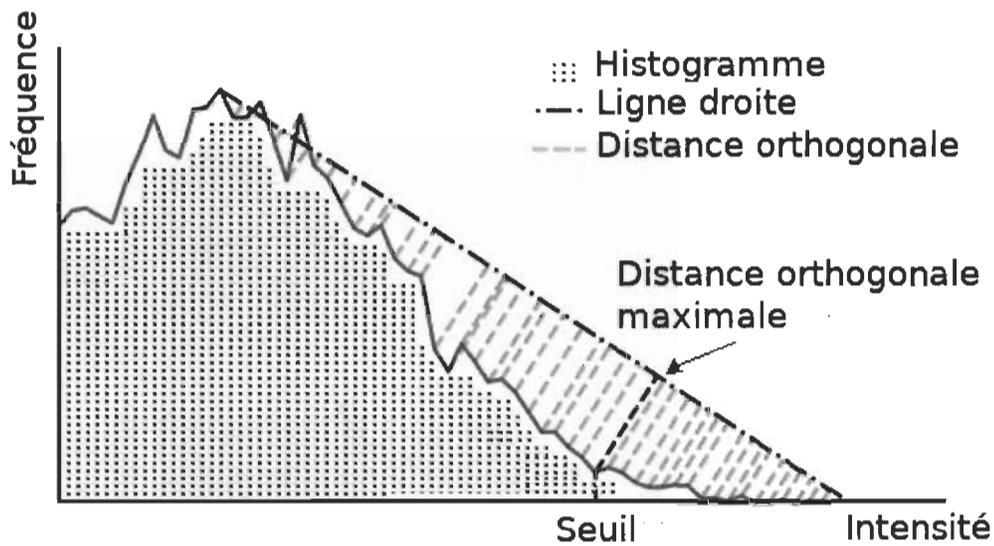


FIGURE 3.10 – Méthode de seuillage de P. Rosin pour les distributions unimodales.

Les images de diatomées traitées dans ce projet ont une distribution unimodale mais différente de celle étudiée par Rosin car l'histogramme des images présentent un pic dominant au milieu de l'histogramme. Conséquemment, nous devons chercher deux points d'inflexion, un de chaque côté du pic. Au chapitre 4, nous présentons une méthode de segmentation que nous avons développée pour les diatomées laquelle qui est une adaptation de l'approche de Rosin que nous avons appliquée à la fonction de répartition plutôt qu'à la fonction de densité (histogramme).

3.4 Conclusion

Dans ce chapitre, nous avons vu des méthodes de traitement d'images qui seront utilisées dans le chapitre suivant. Nous observons que certains paramètres comme la taille de fenêtre des filtres, ou la taille des éléments structurants dépendent des caractéristiques des images utilisées. Nous observons que la sélection d'une méthode de seuillage dépend aussi des propriétés des images. Il est nécessaire de procéder par essais en utilisant différentes méthodes et en ajustant les paramètres afin de trouver une solution adaptée à notre problématique.

Dans le chapitre suivant, nous présentons une méthode de segmentation adaptée à nos images de diatomées, ainsi qu'une méthode de détection et d'extraction pour isoler les spécimens de l'arrière-plan. Nous présentons aussi une procédure pour détecter des caractéristiques taxonomiques comme le raphé, les stries et les cercles concentriques.

Chapitre 4

Pré-traitement des images et extraction des caractéristiques

Dans ce chapitre, nous décrivons la méthodologie utilisée pour extraire des caractéristiques nécessaires à la classification automatique des images de diatomées. Ce chapitre est organisé en deux sections, le pré-traitement des images et l'extraction des caractéristiques. Dans la première section, nous présentons les méthodes de segmentation et de détection d'objets utilisées pour extraire les spécimens de l'arrière-plan des images. Dans la deuxième section nous présentons les techniques pour extraire des caractéristiques de la diatomée, en l'occurrence des structures internes des diatomées. Dans l'annexe A nous présentons les notions mathématiques à la base de notre méthode de segmentation et dans l'annexe B nous décrivons la structure du code de programmation que nous avons développé.

4.1 Pré-traitement des images

Filtrage et segmentation

Dans le cadre de ce travail, nous traitons des images planaires en niveaux de gris. Chaque image contient du bruit et des impuretés issus de la saisie. Les images de diatomées, captées à l'aide d'une caméra couplée à une loupe binoculaire, sont déposées dans des pétris contenant de l'eau naturelle en présence d'organismes comme des dé-

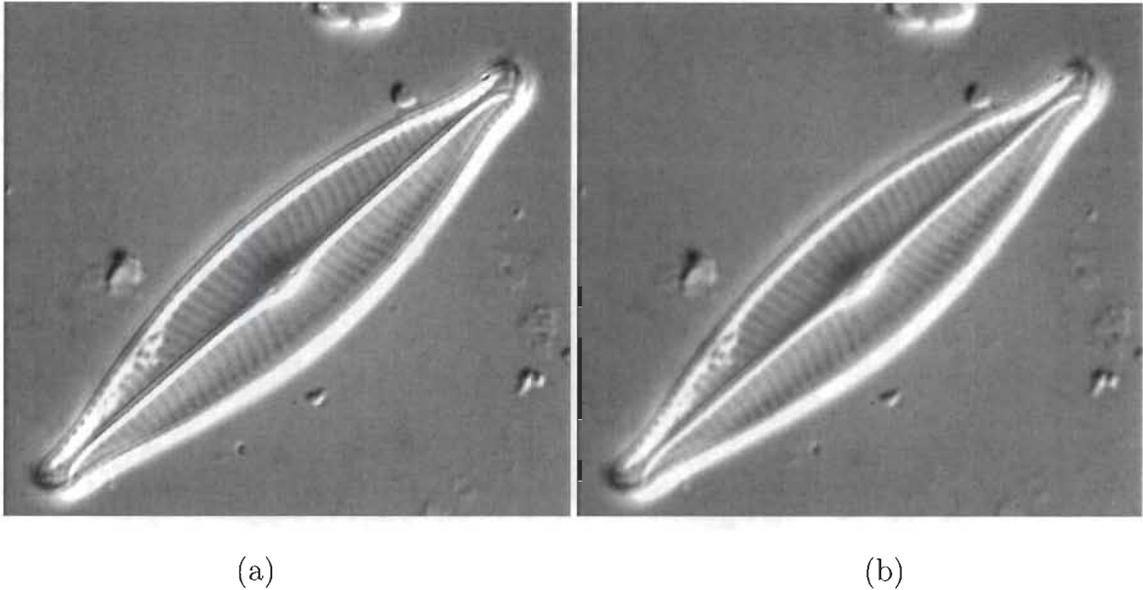


FIGURE 4.1 – (a) Image originale (b) Image lissée par filtrage gaussien

bris de diatomées et de phytoplancton, entre autres. De plus, des zones surexposées à l'éclairage artificiel apparaissent lors de la saisie et créent des zones de surbrillance. Notre objectif de départ est d'isoler une diatomée entière à des fins d'identification, et conséquemment nous devons éliminer le bruit. Une première étape consiste à éliminer une partie de ce bruit avec un filtre gaussien (section 3.2). Si la fenêtre du filtre gaussien est trop grande, nous perdons des détails dans l'image. Si la fenêtre est trop petite, nous n'éliminons pas suffisamment de bruit et d'impuretés. Après plusieurs tests, nous avons opté pour une fenêtre (7×7) . La figure 4.1 illustre le résultat obtenu.

Après le filtrage des images, nous passons à l'étape de la segmentation. Les images de diatomées ont plusieurs caractéristiques communes dont nous nous servons pour définir un seuil permettant la segmentation, c'est-à-dire, la séparation de la diatomée du fond de l'image. En effet, l'arrière-plan de l'image est d'une intensité de gris presque uniforme et toutes les régions isolées (débris) sont petites par rapport à celle de la diatomée laquelle est centrale dans l'image (voir Figure 4.2 (a) et (b)). Nous ne considérerons pas les images où les diatomées sont partiellement recouvertes par d'autres artefacts ou tronquées par le cadre de l'image, ni celles où les diatomées

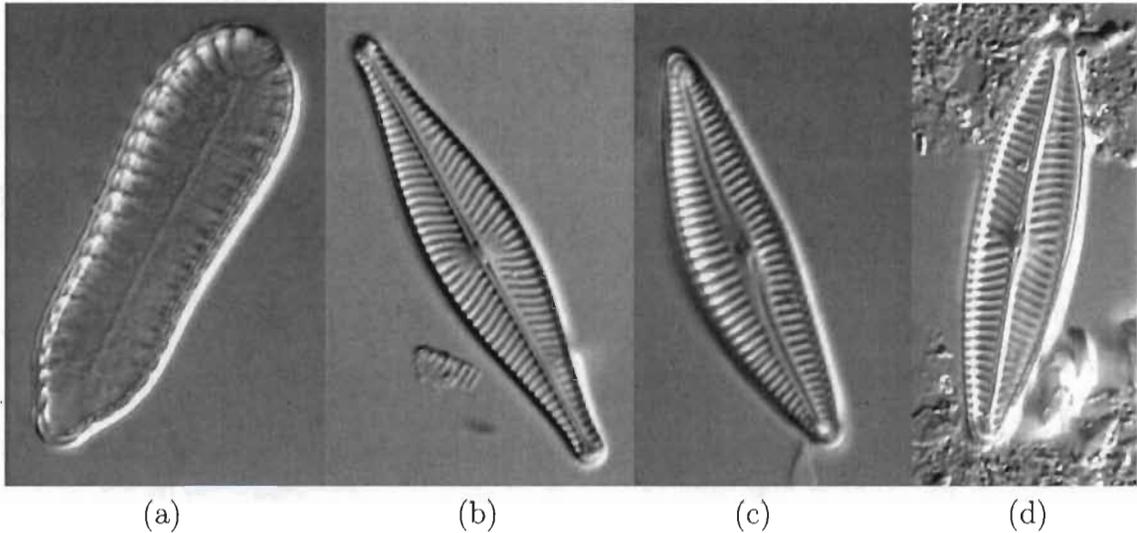


FIGURE 4.2 – (a) Diatomée avec un arrière-plan lisse (b) Diatomée avec des particules isolées (c) Diatomée en contact avec une particule (d) Diatomée avec un arrière-plan bruité

touchent un ou des artefacts ou encore si l'arrière-plan est fortement bruité par la présence de multiples débris (voir Figure 4.2 (c) et (d)). Étant donné que l'arrière-plan est l'aire la plus grande dans l'image, nous pouvons poser l'hypothèse qu'il n'y a qu'un pic dominant au centre de l'histogramme des niveaux de gris de l'image originale.

En regard de cette distribution spécifique des niveaux de gris dans nos images et de leur appartenance à une diatomée ou au fond de l'image, la méthode de segmentation développée consiste à trouver deux seuils, idéalement optimaux, de part et d'autre du pic central de la distribution des niveaux de gris de telle façon que les intensités entre ces seuils correspondent à l'arrière-plan et le reste à la diatomée. Nous pouvons tirer avantage de la forme particulière de la fonction de répartition (CDF) construite à partir d'une telle fonction de distribution possédant un pic central. En effet, en plus d'être croissante sur le domaine des niveaux de gris, celle-ci s'apparente à une fonction de type sigmoïde ou logistique. À la section 3.3, nous avons vu que la segmentation unimodale proposée par Rosin consiste à identifier un changement abrupt d'un seul côté de l'histogramme. Dans notre cas, nous sommes en présence de deux changements abrupts, ou inflexions, aux voisinages du début et de la fin du pic central. De plus, ces changements sont plus marqués sur la courbe interpolée

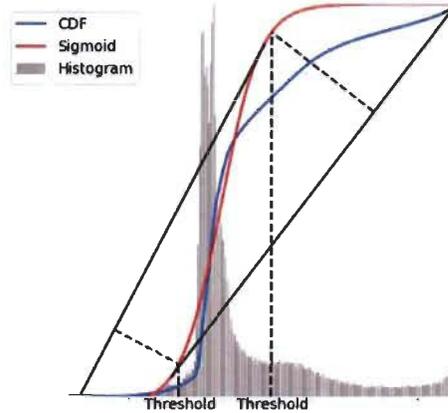


FIGURE 4.3 – Méthode de segmentation proposée.

(SCDF) de la CDF par une fonction sigmoïde. Nous appliquerons le principe de base de la méthode de Rosin sur la fonction de répartition interpolée (SCDF) pour établir les deux seuils recherchés.

Pour cela, nous traçons deux droites sécantes de chaque côté de la SCDF. La sécante à gauche a comme premier point celui correspondant à la dernière intensité de la fonction, à partir du début du domaine de définition, ayant une valeur nulle sur la SCDF. Nous notons ce point $(l_0, 0)$. Le dernier point de la sécante à droite sur la courbe correspond au point $(l_i, SCDF(l_i))$ sur la courbe pour lequel la pente $\frac{SCDF(l_i)}{l_i - l_0}$ est maximale pour $r_0 < i \leq 255$. D'une façon similaire, la sécante à droite se termine au premier point d'abscisse r_0 tel que son ordonnée vaut 1 et elle débute au point sur la courbe $(r_i, SCDF(r_i))$ tel que la valeur absolue de la pente $\frac{SCDF(r_i) - 1}{r_i - r_0}$ est maximale pour $r_0 > i \geq 0$. Ces points correspondent au point de départ et au point maximum (point final) tels que définis dans la méthode de Rosin. Par la suite, pour chacune de ces droites sécantes, nous cherchons l'abscisse x^t tel que la distance orthogonale entre le point $(x^t, SCDF(x^t))$ et la sécante soit maximale. Finalement, nous obtenons deux points x_1^t et x_2^t correspondants aux seuils optimaux pour segmenter l'image. À l'annexe A, nous présentons le calcul théorique de ces deux seuils optimaux. Si $F(i)$ est l'intensité de niveau de gris d'un pixel i quelconque, l'image segmentée F' est obtenue comme suit ;

$$F'(i) = \begin{cases} 0 & x_1^t \leq F(i) \leq x_2^t, \\ 255 & \text{sinon.} \end{cases} \quad (4.1)$$

Afin d'améliorer la segmentation, nous appliquons successivement les opérateurs morphologiques d'ouverture et de fermeture avec un élément structurant circulaire de taille trois et cinq respectivement. La figure 4.4 montre le résultat obtenu à cette étape.

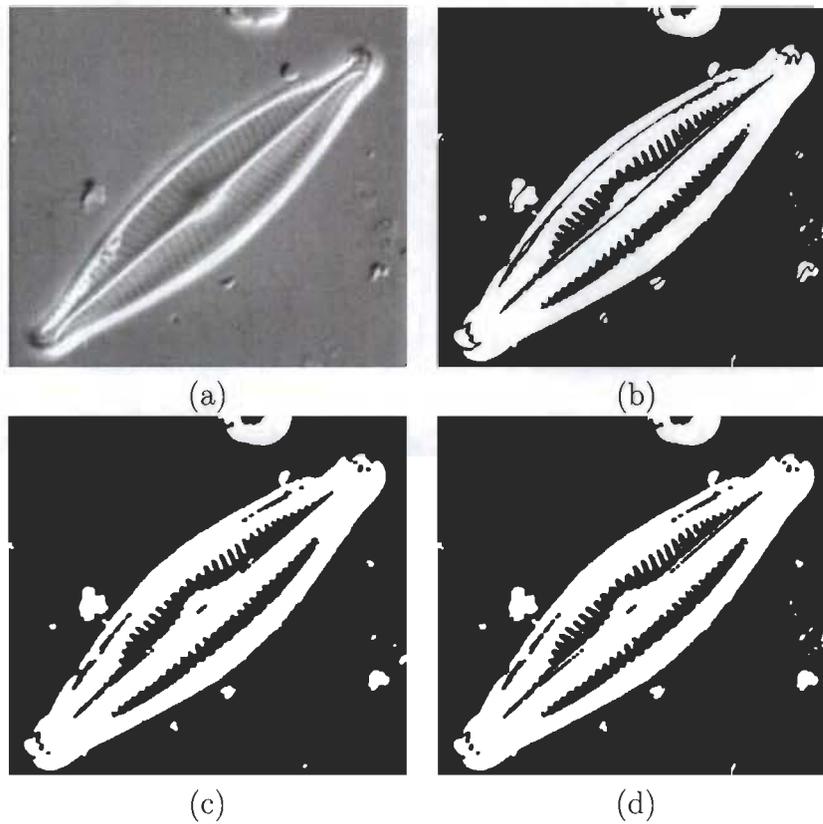


FIGURE 4.4 – (a) Image lissée. (b) Image segmentée (c) Image résultant de l'ouverture (d) Image résultant de la fermeture

Détection de la diatomée

L'image résultante (Figure 4.4 d) de l'étape précédente est binaire et contient des régions blanches avec un arrière-plan noir. Idéalement, nous devrions obtenir une seule région blanche en l'absence de bruit mais dans les images bruitées nous obtenons plusieurs régions parmi lesquelles nous devons identifier celles qui composent la diatomée.

Afin d'identifier les régions pouvant constituer des parties de la diatomée nous utiliserons la force gravitationnelle entre la région dominante et chacune des autres régions pour identifier celles pouvant appartenir à la diatomée. Les régions fortement attirées par la région principale seront considérées comme faisant partie de la diatomée. Dans un premier temps, nous étiquetons chacune des régions et en calculons la superficie, en ayant au préalable appliquée une dilatation à l'image. Cette opération de dilatation permet le regroupement de régions qui sont déjà très proches. Dans cette étape nous ne considérons pas les régions imbriquées dans une plus grande région. De cette façon, nous réduisons le nombre de composantes à traiter dans les étapes ultérieures.

Soit c_1 la plus grande région, nous normalisons la taille des régions c_i en divisant chaque aire par celle de l'enveloppe convexe de c_1 . Puis, pour chaque région c_i nous calculons la distance entre les régions c_i et c_1 comme suit ;

$$d_i = \inf\{d(x_m, y_n), x_m \in c_1, y_n \in c_i\}, i > 1. \quad (4.2)$$

Ces distances sont normalisées en divisant chacune d'elles par la moitié de la longueur de la diagonale de l'image, cette valeur étant la distance la plus longue à laquelle un objet peut être éloigné du centre de l'image. Finalement, si s_i est la taille de la région c_i et d_i est sa distance à c_1 , toutes les deux normalisées, alors nous définissons la force d'attraction f_i entre c_i et c_1 comme étant ;

$$f_i = \frac{s_i}{d_i^2} \quad (4.3)$$

La force (4.3) ne nous permet pas de juger si une région secondaire est loin ou proche de la région dominante. Selon nos tests, avec les images de diatomées, nous avons identifié un seuil qui nous permet de rejeter les débris ayant une force d'attraction $\sigma \leq 40$. Les autres régions sont considérées comme faisant partie de la diatomée (Figure 4.5).

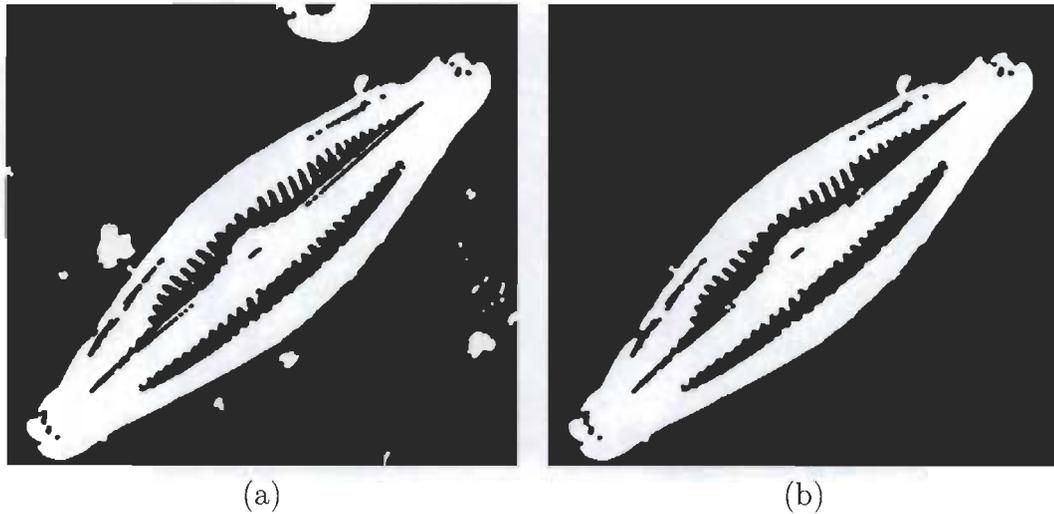


FIGURE 4.5 – (a) Image segmentée (b) Image découlant de l'élimination des débris

Après avoir identifié les régions indépendantes constituant la diatomée dans l'image, la prochaine étape consiste à regrouper ces régions et à effectuer une approximation du contour de la diatomée. Nous commençons par calculer l'enveloppe convexe de l'ensemble des régions retenues. Cette enveloppe est une première approximation du contour de la diatomée.

Pour trouver ce contour, nous identifions un ensemble ordonné γ de points, en commençant par le plus à droite et le plus haut des points sur les frontières des régions retenues. Les points de γ sont les premiers et les derniers points qui intersectent verticalement et horizontalement les régions de la diatomée. À la figure 4.6 les points de l'ensemble γ sont indiqués en vert.

Pour les diatomées ayant une forme quasi convexe (figure 4.7 (a)), cette approche donne de bons résultats et l'enveloppe convexe peut être utilisée sans modification pour extraire la diatomée de l'image. Cependant, le frustule de beaucoup d'espèces n'est pas de forme convexe (figure 4.7 (b)) et l'enveloppe convexe n'est pas suffisamment précise. Conséquemment, nous avons développée une méthode de raffinement du contour, à partir d'un contour initial, afin de construire un contour de plus en plus précis de la diatomée. Des essais ont été effectués à l'aide de différentes mé-

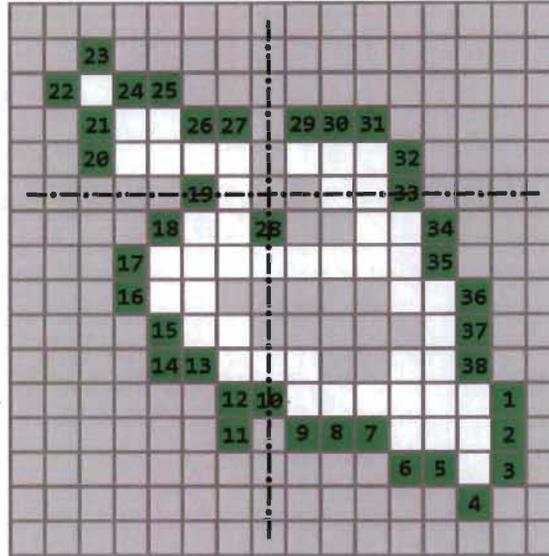


FIGURE 4.6 – Points ordonnés sur le contour primitif.

thodes classiques de contours adaptatifs mais sans succès.

L'enveloppe convexe γ_* est un sous-ensemble ordonné de points de γ dont chaque paire de points consécutifs $i, j \in \gamma_*$ forme un segment ℓ_{ij} du contour. Pour déterminer quels sont les segments ℓ_{ij} qui doivent être raffinés, nous commençons par calculer la longueur de chacun des segments, la distance euclidienne entre les points i et j , et nous identifions ceux qui ont une longueur considérée atypique. Pour cela, soit μ_* la moyenne des longueurs des segments de γ_* et soient Q_1 et Q_3 le premier et le troisième quartiles, alors une longueur d_{ij} sera dite atypique si

$$Q_1 - \frac{3IQR}{2} \geq d_{ij} \quad \text{ou} \quad d_{ij} \geq Q_3 + \frac{3IQR}{2} \quad (4.4)$$

où $IQR = Q_3 - Q_1$. Puisque nous cherchons les segments à améliorer, nous n'utilisons que l'inégalité de droite. Ces segments atypiques apparaissent à l'extérieur des moustaches dans le diagramme de boîte à moustaches au bas de la Figure 4.8.

L'étape suivante consiste à subdiviser les segments atypiques en sous-segments, d'extrémités appartenant au contour, afin de mieux épouser la forme de la diatomée. Pour chaque segment ℓ_{ij} à raffiner, nous identifions les points de γ qui se trouvent

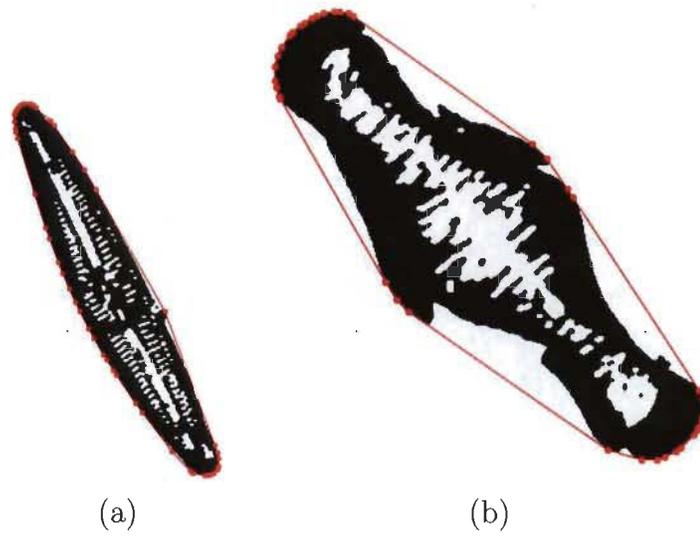


FIGURE 4.7 – Enveloppe convexe d’une diatomée : (a) de forme convexe, (b) de forme concave.

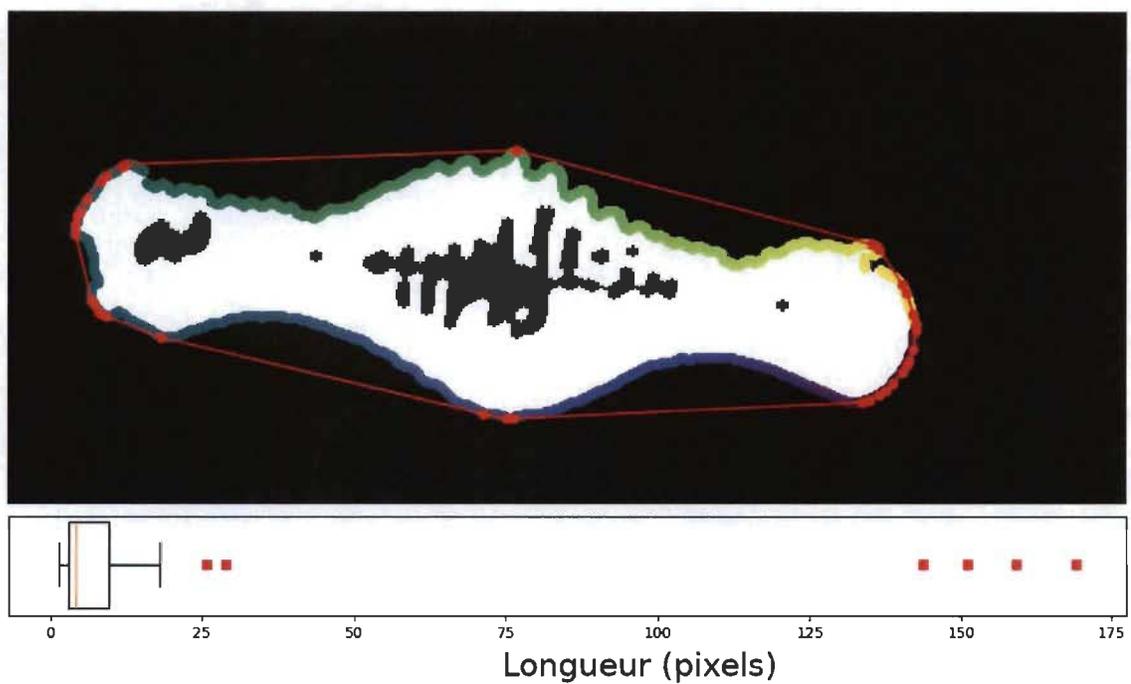


FIGURE 4.8 – Boîte à moustaches des longueurs des segments de l’enveloppe convexe d’une diatomée concave.

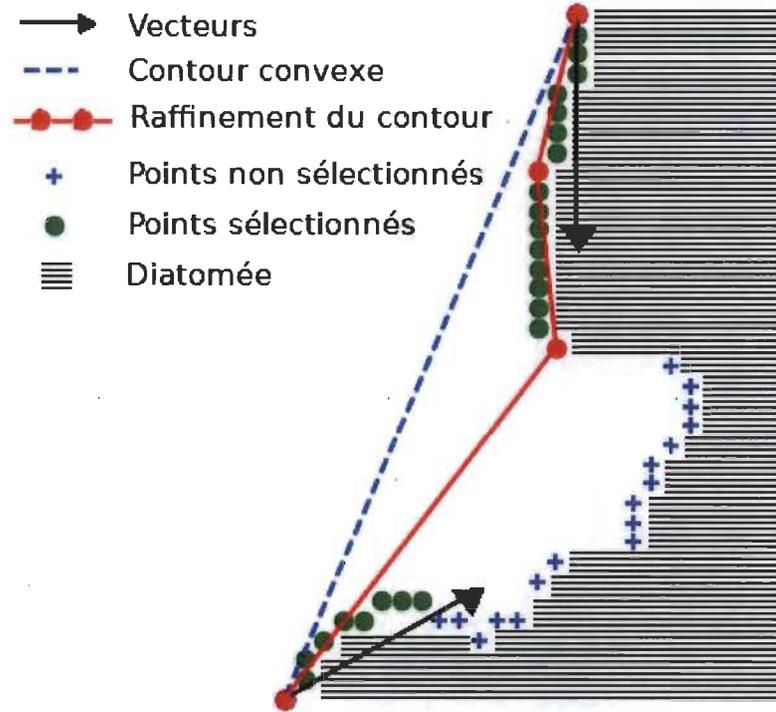


FIGURE 4.9 – Correction des segments atypiques de l'enveloppe convexe.

entre les extrémités i et j , puis nous sélectionnons parmi ces points les plus proches de ℓ_{ij} selon un critère. Soit γ_{ij} l'ensemble des points de γ entre i et j , notons γ_{1ij} la première moitié de l'ensemble des points et γ_{2ij} est la seconde. Notons c_1 et c_2 les barycentres respectifs de ces sous-ensembles et définissons les vecteurs $v_1 = (i, c_1)$ et $v_2 = (j, c_2)$. Les points considérés comme étant « proches » de ℓ_{ij} sont les points à l'intérieur du cône défini par les directions v_1 et v_2 , et borné par ℓ_{ij} (Figure 4.9). Nous utilisons le produit vectoriel pour déterminer la position relative des points de γ_{ij} par rapport aux directions v_1 et v_2 dont les extrémités sont positionnées aux extrémités i et j .

Soit $\gamma_{ij}^* = \{p_1, p_2, \dots, p_n\}$ l'ensemble ordonné des points de γ proches de γ_{ij} , nous divisons cet ensemble en k sous-ensembles de même cardinalité (à un point près) pour des valeurs successives de $k = 2, \dots$;

$$\gamma_{ij}^* = \{ \{p_1 = p_{11}, p_{12}, \dots, p_{1m}\}, \{p_{21}, p_{22}, \dots, p_{2m}\}, \dots, \{p_{k1}, p_{k2}, \dots, p_{km} = p_n\} \}.$$

Nous calculons la longueur de chacun des segments de $\gamma_{i,j}^*$ et vérifions si la lon-

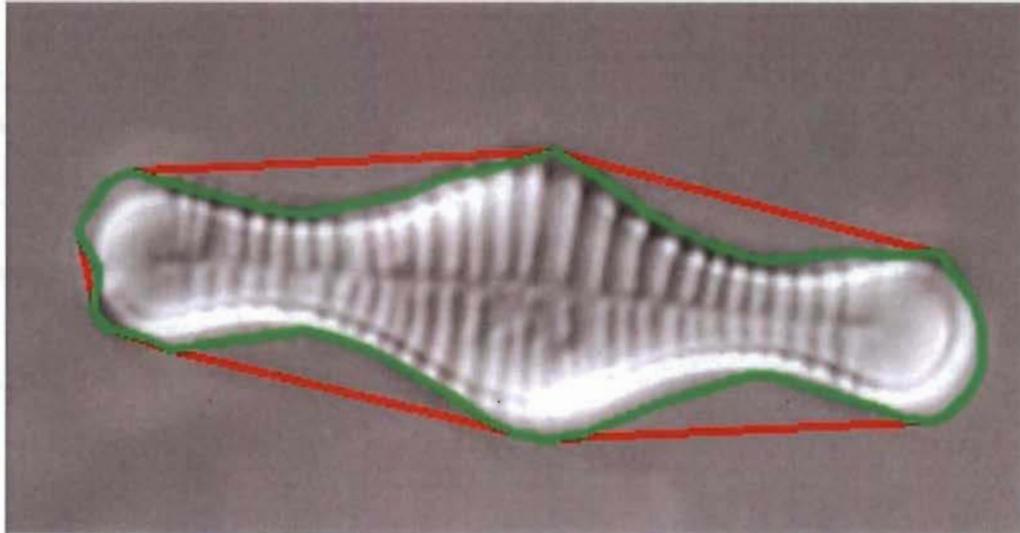


FIGURE 4.10 – Enveloppe convexe initiale en rouge et contour après raffinement en vert.

Le rayon de courbure de tous les sous-segments $[p_{i1}, p_{im}]$, pour k donné, est plus petite ou égale à la moyenne (seuil) du contour polygonal initial. Sinon, nous incrémentons la valeur de k . Les extrémités de chaque sous-ensemble de γ_{ij}^* sont ajoutées à l'enveloppe convexe pour constituer le contour final (voir Figure 4.10).

La diatomée extraite de l'arrière-plan sera l'ensemble des pixels de l'image originale qui sont sur le contour et à l'intérieur du contour (Figure 4.11).

Amélioration du contraste

Comme nous l'avons mentionné dans l'introduction de ce travail, le faible contraste des images est un facteur qui affecte la qualité des résultats de la segmentation, mais aussi l'étape d'extraction des caractéristiques. Nous améliorons le contraste des images en utilisant la méthode de normalisation d'histogramme présentée au chapitre 3. À la Figure 4.12 nous montrons le résultat obtenu suite à l'application de cette méthode. Nous remarquons que cette amélioration est obtenue après l'extraction de l'objet car la méthode aurait mis en évidence les détails de l'arrière-plan, dont celle des débris. L'amélioration du contraste favorisera l'extraction des structures internes.

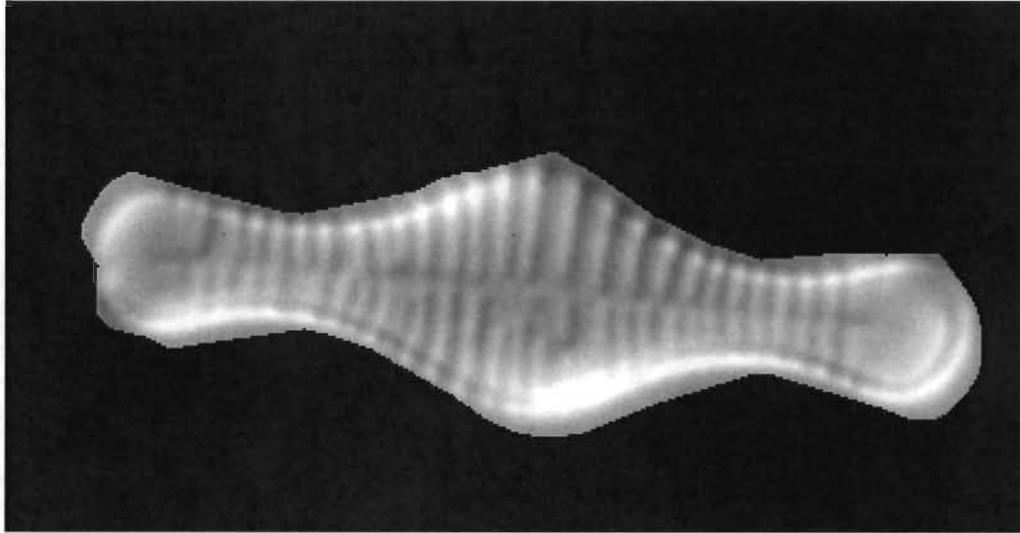


FIGURE 4.11 – Extraction de la diatomée.

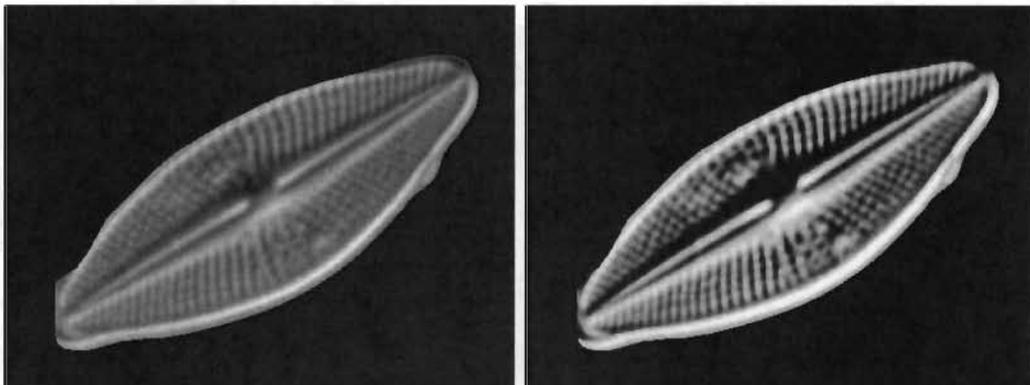


FIGURE 4.12 – Image de la diatomée extraite (gauche), rehaussement du contraste (droite).

Rotation et recadrage

Dans cette section, nous effectuons des étapes de normalisation des images afin de pouvoir les comparer lors de la classification. D'abord, nous identifions le contour 8-connecté de la diatomée. Ce contour est différent du polygone que nous avons utilisé pour extraire l'objet. Nous initialisons le contour au point de la diatomée le plus haut à droite. En parcourant l'image dans le sens horaire nous analysons les voisins du point courant, puis nous ajoutons au contour le premier point rencontré sur le contour qui est 8-connecté. Le point trouvé devient le point courant et nous répétons cette procédure jusqu'à que nous arrivions de nouveau au point de départ. Le contour aurait pu être initialisé à tout autre point référencé et construit dans le sens horaire ou anti-horaire.

Nous définissons l'axe principal de la diatomée comme étant le plus long segment reliant deux points du contour. Pour normaliser la position des diatomées, nous effectuons une rotation du spécimen de telle façon que l'axe principal soit parallèle à l'axe horizontal, puis nous recadrons l'image en laissant un espace de 5 pixels entre la diatomée et le bord de l'image (voir Figure 4.13).

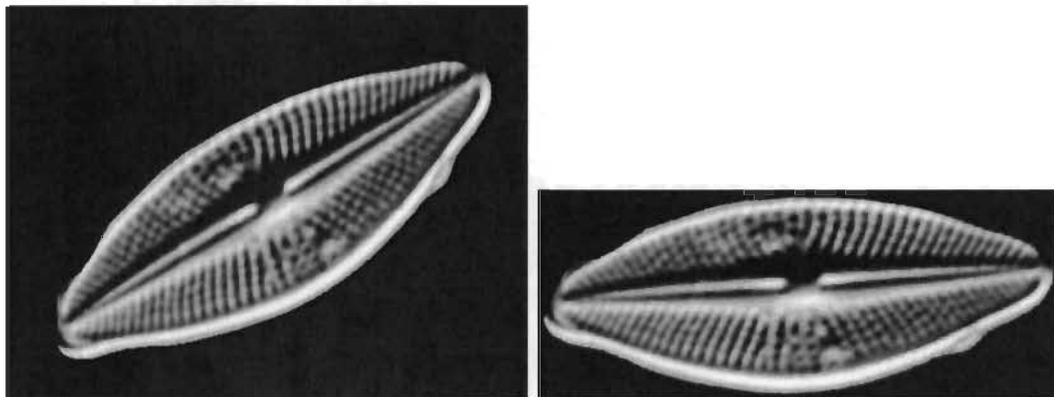


FIGURE 4.13 – Image avant recadrage (gauche), image après recadrage (droite).

4.2 Extraction des caractéristiques

Le frustule

À partir de la géométrie du frustule, il est possible d'effectuer un regroupement d'espèces de diatomées. Ce regroupement peut servir comme première différenciation des espèces selon leur forme et par la suite nous pouvons rechercher les caractéristiques internes tels que le raphé et les stries. Dans un premier temps, nous discriminons deux grandes classes de diatomées : les diatomées centrales et les diatomées pennales. Les diatomées centrales ont un frustule circulaire tandis que les diatomées pennales ont un frustule allongé, parfois asymétrique.

Le raphé

Dans les images de diatomées, le raphé apparaît comme une ligne épaisse qui va d'une extrémité à l'autre et qui passe généralement par l'axe principal. Normalement cette ligne est interrompue par un nodule central, mais elle demeure continue jusqu'à l'autre extrémité de la diatomée.

Certaines méthodes de détection de lignes, comme celle de Hough [8], utilisent le filtre de Canny pour la détection des contours mais elles ne permettent pas toujours de trouver une ligne relativement large comme celle du raphé. De plus, les irrégularités du raphé réduisent la précision de la détection même si la forme générale est une ligne droite. Nous proposons l'utilisation de l'histogramme des gradients orientés (HOG) pour généraliser la détection des contours. Pour améliorer nos résultats, nous appliquons au préalable un filtre de Sobel [23] pour accentuer les contours de l'image avant l'utilisation du filtre de Canny.

La méthode de HOG est basée sur les intensités des gradients orientés calculées sur des parties de l'image. Nous appliquons cette méthode à l'image obtenue par le filtre de Canny. Nous divisons l'image du gradient en sous-régions de même taille et pour chaque sous-région, nous calculons les intensités (normes) du gradient dans les

quatre directions principales (axes et diagonales).

Des modèles d'apprentissage automatique, par exemple [9], utilisent la méthode de HOG comme vecteur de caractéristiques pour faire de la classification. Puisque nous nous intéressons à la détection de lignes droites sur l'image des contours issue du filtre de Canny, nous proposons une approche qui nous permettra de visualiser le résultat obtenu par la méthode de HOG.

Pour le calcul des gradients orientés dans l'image, nous divisons celle-ci en sous-régions d'égales dimensions, soit des carrés de $k \times k$ pixels. Nous appelons cette partition la grille des sous-régions. Nous avons testé différentes tailles de partitionnement et avec $k = 10$ nous obtenons une représentation qui préserve adéquatement les détails des structures.

Dans chaque sous-région, nous assignons au pixel central la valeur moyenne des dérivées directionnelles selon l'axe horizontal, l'axe vertical et les deux diagonales. Chacune de ces valeurs est représentée par un segment orienté dans la direction correspondante. Le niveau de brillance d'un segment correspond à la valeur de la dérivée. Cette représentation graphique est illustrée à la figure 4.14

En présence de structures rectilignes, nous observons des segments de fortes intensités pouvant former une droite dans une direction. Si nous observons de fortes intensités dans la direction de l'axe principal de la diatomée et sur une longueur d'au moins 70% de la longueur de cet axe, nous supposons être en présence d'un raphé (Figure 4.15).

S'il existe plus d'une ligne horizontale suffisamment longue, nous choisissons la plus longue. Si aucune ligne n'a été détectée dans la direction horizontale, nous supposons que la diatomée n'a pas de raphé ou qu'il n'a pu être détecté. Certaines espèces de diatomées ont un frustule très allongé et une partie du contour peut prendre la forme d'une droite parallèle à l'axe principal, et donc il peut être inter-

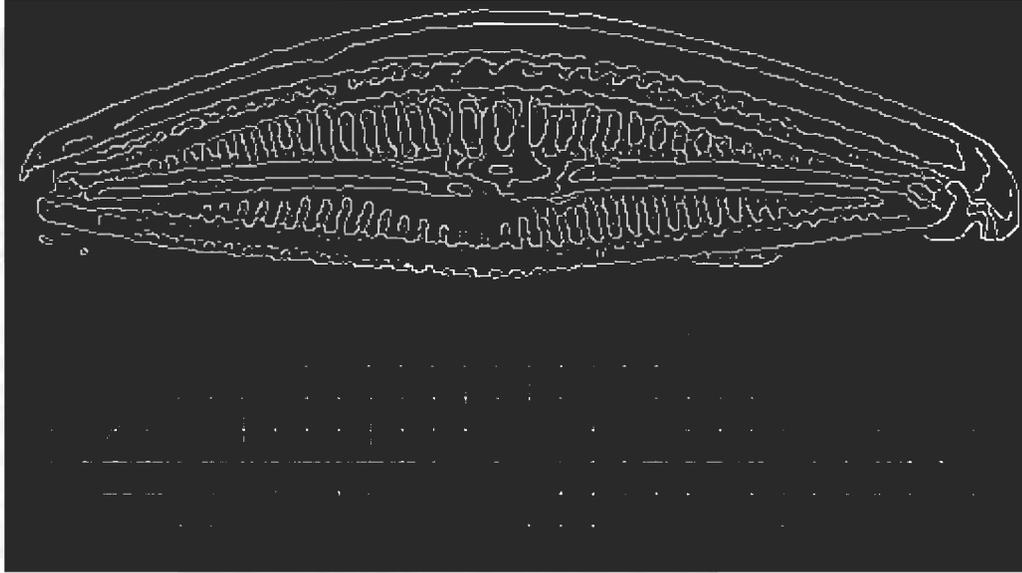


FIGURE 4.14 – Contours des intensité du gradient (haut), représentation des intensités du gradient orienté (bas).

prété comme un raphé. Pour éviter les faux positifs, nous enlevons le contour de la diatomée avant de calculer les gradients orientés. La Figure 4.14 illustre le résultat obtenu en limitant le calcul des gradients à l'intérieur du domaine.

Il se peut que le raphé soit représenté par plusieurs lignes parallèles et adjacentes. Par conséquent, deux lignes consécutives dans la grille des sous-régions peuvent représenter une même structure linéaire dans l'image. Dans ce cas, nous calculons le pourcentage obtenu lors du cumul des contribution de deux lignes adjacentes. Si la somme des lignes i et j dans la grille des sous-régions ($i < j$ avec i à une hauteur de $i \times 10$ pixels), alors on définit une ligne à mi-chemin en position $(i \times 10) + 5$, dont le pourcentage est la somme des pourcentages des lignes i et j .

Finalement, si la diatomée possède un raphé il devrait se trouver sur la ligne ayant le pourcentage le plus grand et dépassant le seuil inférieur de 70%. À la figure 4.15, nous observons la position du raphé localisé à l'aide de cette technique.

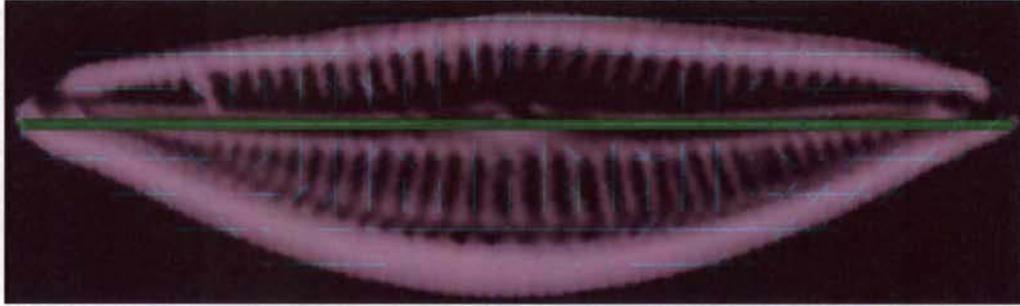


FIGURE 4.15 – Détection du raphé.

Les stries

La forme et la densité des stries varient d'une espèce à l'autre. Elles peuvent être perpendiculaires à l'axe principal ou suivre une progression d'angles par rapport à la verticale. À la différence du raphé, les stries forment des lignes plus minces, mais elles peuvent être irrégulières (angles, tailles) au sein d'un spécimen et d'une même espèce. Les méthodes de détection des lignes ne peuvent donc pas être appliquées directement.

De la même façon que pour la détection du raphé, nous proposons l'utilisation de la méthode de HOG pour trouver des lignes orthogonales ou inclinées par rapport à l'axe principal de la diatomée. Nous cherchons dans l'image les fortes intensités du gradient qui sont alignées selon certaines directions pour détecter des structures linéaires (voir Figure 4.16). Si plusieurs lignes sont trouvées, alors nous supposons que le spécimen possède des stries. Pour réduire les faux positifs, comme précédemment nous retirons de l'analyse le contour extérieur avant de calculer les gradients dans l'image.

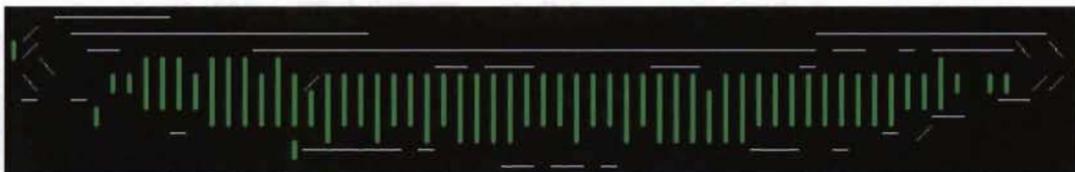


FIGURE 4.16 – Détection des stries.

Stries radiales et cercles concentriques

Comme nous l'avons mentionné précédemment, certaines espèces de diatomées possèdent un frustule circulaire. Ces espèces présentent aussi des ornements qui les différencient, comme la présence de stries radiales et de cercles concentriques (voir Figure 4.17).

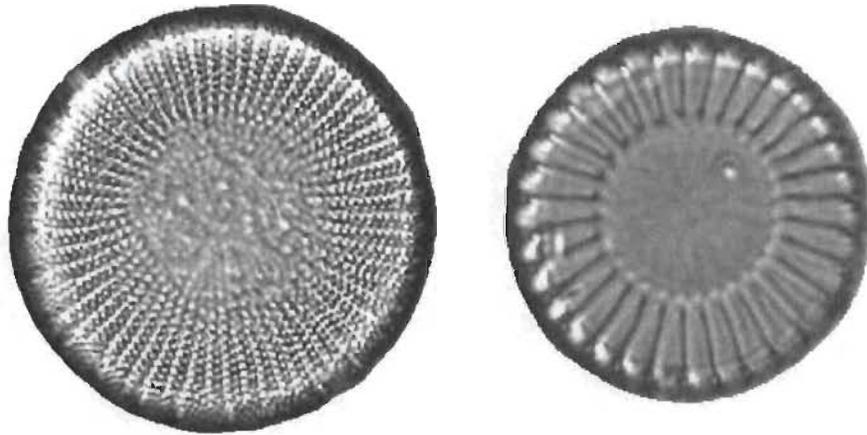


FIGURE 4.17 – Exemples de diatomées centrales.

Les cercles concentriques peuvent être délimités par une ligne ou encore par une variation circulaire de la texture de la diatomée (anneaux de texture différentes). Par ailleurs, les stries radiales se déploient à partir du centre de la diatomée à son contour extérieur et dans certains cas elles sont interrompues (voir Figure 4.17). Nous proposons une méthode visant la détection des stries et des cercles concentriques qui diffère de celle adoptée pour les diatomées allongées.

Nous commençons par trouver l'ensemble des contours de Canny dans l'image de la diatomée isolée de son fond. Pour réduire la quantité de bruit, ainsi que pour augmenter la précision de la méthode, nous filtrons l'image en éliminant les composantes n'ayant qu'un, deux ou trois pixels connexes. À partir de l'image résultante, nous trouvons l'ensemble des points $P = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ correspondant aux pixels blancs de l'image. Nous définissons le centre $c = (c_x, c_y)$ de la diatomée comme étant le barycentre des points de P et nous transformons les coordonnées

cartésiennes (x_i, y_i) des points de P en coordonnées polaires (θ_i, r_i) après avoir effectué une translation du barycentre à l'origine. La transformation en coordonnées polaires est la suivante ; pour (x, y) un point du plan cartésien on a,

$$r_i = \sqrt{a^2 + b^2}, \quad (4.5)$$

$$\theta_i = \begin{cases} \arctan\left(\frac{y}{x}\right) & x > 0 \text{ et } y \geq 0 \\ \frac{\pi}{2} & x = 0 \text{ et } y > 0 \\ \arctan\left(\frac{y}{x}\right) + \pi & 0 > x \\ \frac{3\pi}{2} & x = 0 \text{ et } 0 > y \\ \arctan\left(\frac{y}{x}\right) + 2\pi & x > 0 \text{ et } 0 > y \end{cases} \quad (4.6)$$

Notre objectif est de rechercher les points qui partagent une même valeur de θ ou de r , de telle sorte que si deux points i et j ont des valeurs proches de r ($|r_i - r_j| \leq 2$), alors ces deux points appartiennent à un même cercle dans l'image. De façon similaire, si les valeurs de θ ($|\theta_i - \theta_j| \leq 1$) sont proches alors les points sont sur un même rayon de la diatomée et pourraient appartenir à une même strie.

Pour la détection des cercles, nous parcourons, avec un pas unitaire, les valeurs de r , de la plus petite à la plus grande. Pour un rayon r fixé, nous identifions tous les points tel que la partie entière de sa valeur de θ en coordonnées polaires est égale à r . Le périmètre d'un cercle de rayon r est théoriquement égal à $2\pi r$ dans le domaine du continu ; cette valeur peut être associée, dans le cas idéal d'un point d'une unité de côté, au nombre de points formant un cercle de rayon r . Dans nos images, nous considérons que si le nombre de points dépasse le seuil de 70% du périmètre, alors ces points sont considérés comme appartenant à un même cercle de rayon r .

Après la détection des cercles dans une image de diatomée, nous identifions chacune des régions délimitées par des cercles concentriques consécutifs. Dans chacun de ces anneaux circulaires, les ornements peuvent être différentes selon l'espèce de diatomée.

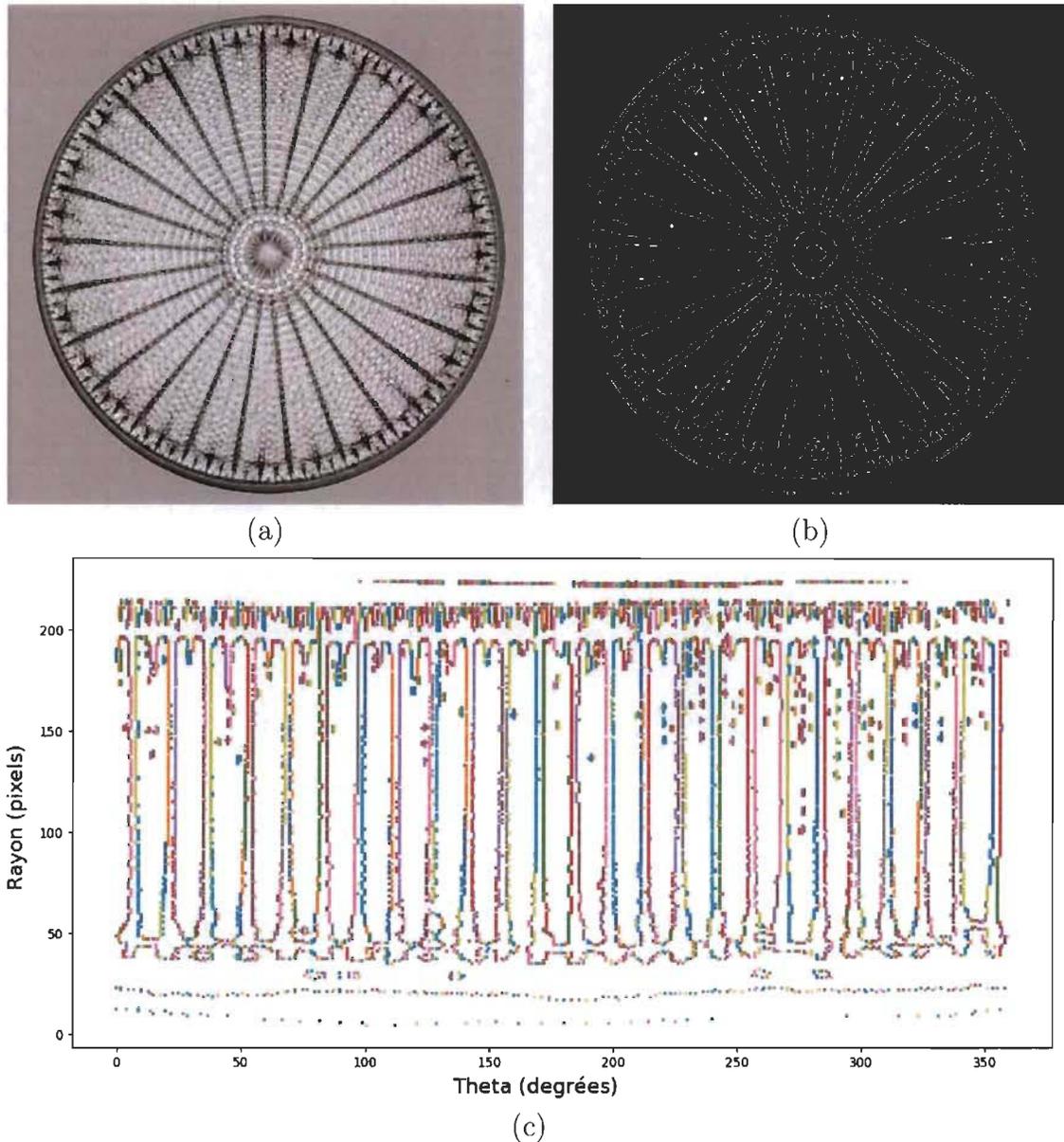


FIGURE 4.18 – Détection des cercles et des stries : (a) Image originale (b) Image des contours de Canny (c) Diagramme θ vs. r des coordonnées polaires des points dans les contours de Canny.

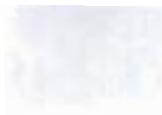
Pour chaque paire (C_i, C_j) de cercles consécutifs de rayons r_i et r_j , à partir de $r = 0$ jusqu'au rayon total de la diatomée, nous parcourons les angles $\theta = 0, 1, \dots, 359$. Pour chacune de ces valeurs d'angle θ , nous identifions l'ensemble des points (pixels blancs) ayant un rayon polaire r tel que $r_i \leq r \leq r_j$. L'ensemble de ces points pourraient appartenir à une même strie d'une longueur égale à $(r_j - r_i)$. Si le nombre de points est supérieur ou égal à 50% de la longueur $(r_j - r_i)$ alors ces points

sont considérés comme faisant partir d'une strie. À la figure 4.18 nous observons les cercles représentés par des lignes horizontales et les stries représentées par des lignes verticales dans l'espace de représentation $(\theta \times r)$ du plan des coordonnées polaires.

4.3 Conclusion

Dans ce chapitre, nous avons décrit une méthode de segmentation adaptée aux propriétés de nos images de diatomées. Nous avons décrit aussi une procédure de détection et d'extraction pour isoler les spécimens de l'arrière-plan. Nous remarquons que les méthodes ont été développées de telle sorte que leur usage soit automatique, c'est-à-dire qu'elles ne nécessitent aucun paramétrage. Nous avons présenté une méthodologie qui visait la détection de trois caractéristiques taxonomiques : le raphé, les stries et les cercles concentriques. Nous observons que la méthode de HOG est capable de détecter des zones dans le gradient de l'image qui pourraient former des lignes. Cette méthodologie d'extraction de caractéristiques taxonomiques pourrait être la base d'une analyse plus spécifique des éléments trouvés à l'intérieur des spécimens.

Dans le chapitre suivant, nous présentons les résultats suite à l'application de nos méthodes aux images de diatomées. Nous observerons que la performance des algorithmes dépend en grande partie du résultat de la segmentation et elle est affectée par les impuretés dans les images. Nous incluons dans le chapitre suivant les résultats issus de l'extraction des caractéristiques.



Chapitre 5

Résultats

Dans ce chapitre, nous présentons les résultats obtenus lors de l'application de notre approche de segmentation et d'extraction de caractéristiques à un ensemble d'images qui respectent les hypothèses énoncées au chapitre 4. Nous avons sélectionné 292 images avec un arrière-plan presque uniforme qui peut contenir de petites particules par rapport à la diatomée. Pour chacune des images, la diatomée n'est pas coupée par le cadre de l'image ni cachée par un autre objet. Nous présentons nos résultats en deux sections principales : l'extraction de la diatomée et l'identification de caractéristiques.

5.1 Extraction de la diatomée

En général, nous avons obtenu de bons résultats lors de la segmentation des images et l'extraction des spécimens. Pour 66 % des images, nous avons obtenu une seule composante connexe correspondant à la diatomée. La méthode de segmentation proposée s'avère précise car les deux seuils optimaux calculés définissent bien la frontière entre les intensités de niveaux de gris de l'arrière-plan de celles de la diatomée. La figure 5.1 illustre quelques exemples qui confirment ce résultat.

Par ailleurs, pour 22 % des spécimens nous avons obtenu plusieurs composantes connexes constituant la diatomée. Dans ce cas, le raffinement de l'enveloppe convexe a permis la fusion des régions permettant l'obtention de la forme de la diatomée et

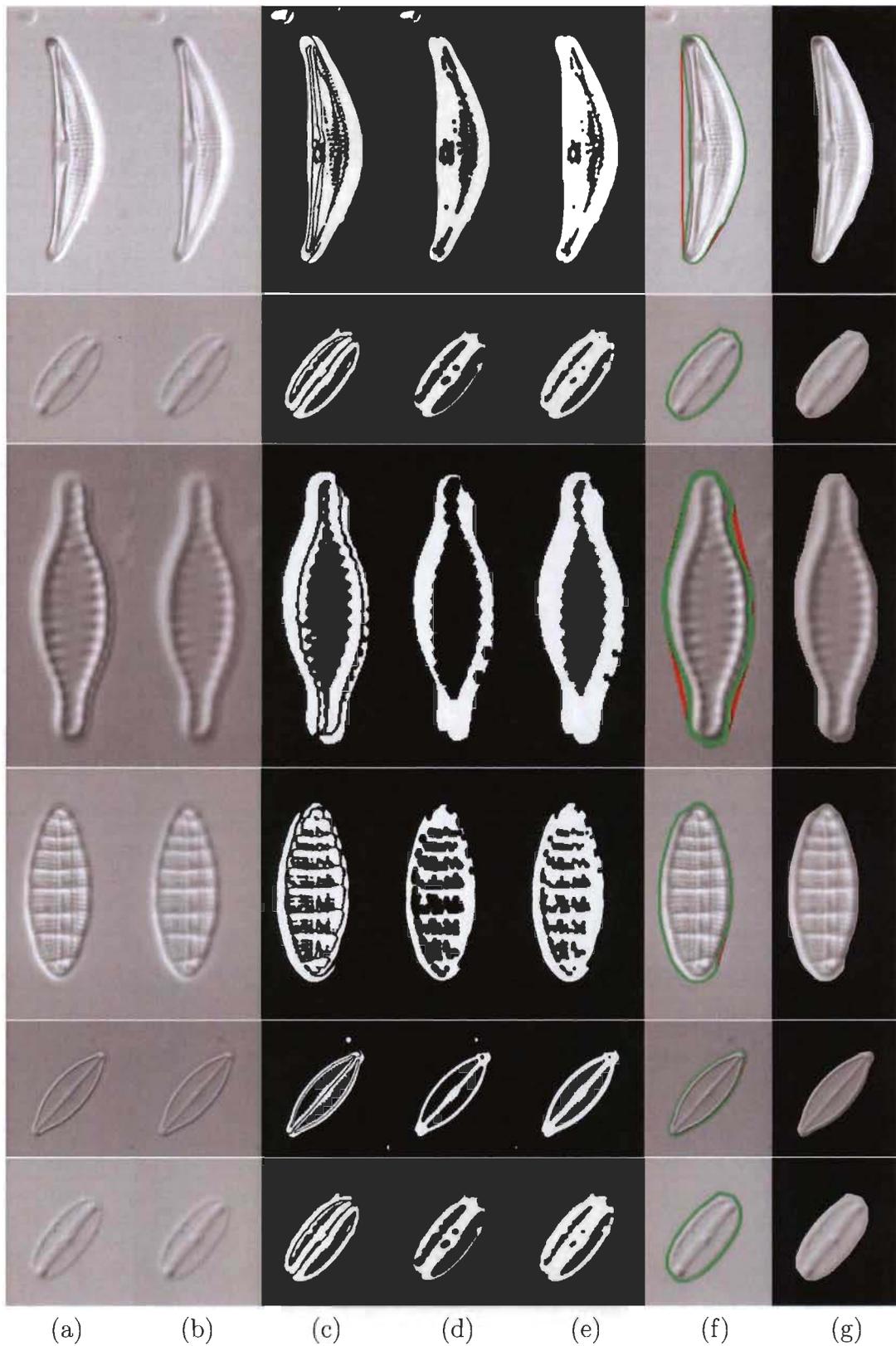


FIGURE 5.1 – Images découlant du processus de détection et d'extraction de diatomée (résultats où la forme de la diatomée est obtenue comme une seule composante connexe dans la segmentation) : (a) Image originale (b) Filtrage gaussien (c) Segmentation (d) Ouverture (e) Fermeture et nettoyage des débris (f) Détection du contour (g) Extraction.

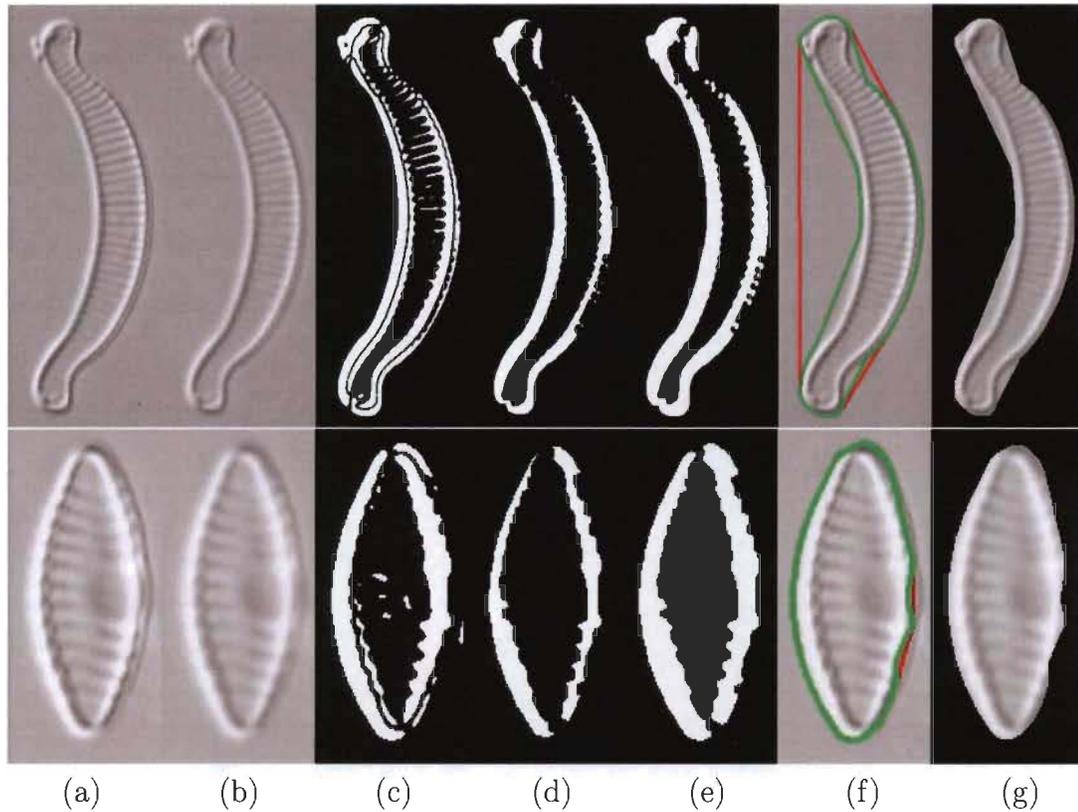


FIGURE 5.2 – Images découlant du processus de détection et d'extraction de diatomée (résultats où la forme de la diatomée est obtenue à l'aide du raffinement de l'enveloppe convexe) : (a) Image originale (b) Filtrage gaussien (c) Segmentation (d) Ouverture (e) Fermeture et nettoyage des débris (f) Détection du contour (g) Extraction.

ainsi de l'extraire de l'arrière-plan. À la figure 5.2, nous présentons des exemples.

Finalement, pour 12 % des cas, la méthode a échoué. Le principal facteur qui explique cet échec est le manque de contraste dans les images, ce qui nuit à la segmentation et par conséquent à la détection de la diatomée. Aussi, il se peut que certaines composantes appartenant à la diatomée ne soient pas retenues dans le processus car elles sont de petites surfaces ou encore trop éloignées de la composante principale. D'autres facteurs comme les reflets dus à la saturation sur le pourtour de la diatomée peuvent modifier considérablement la forme de la diatomée (voir Figure 5.3).

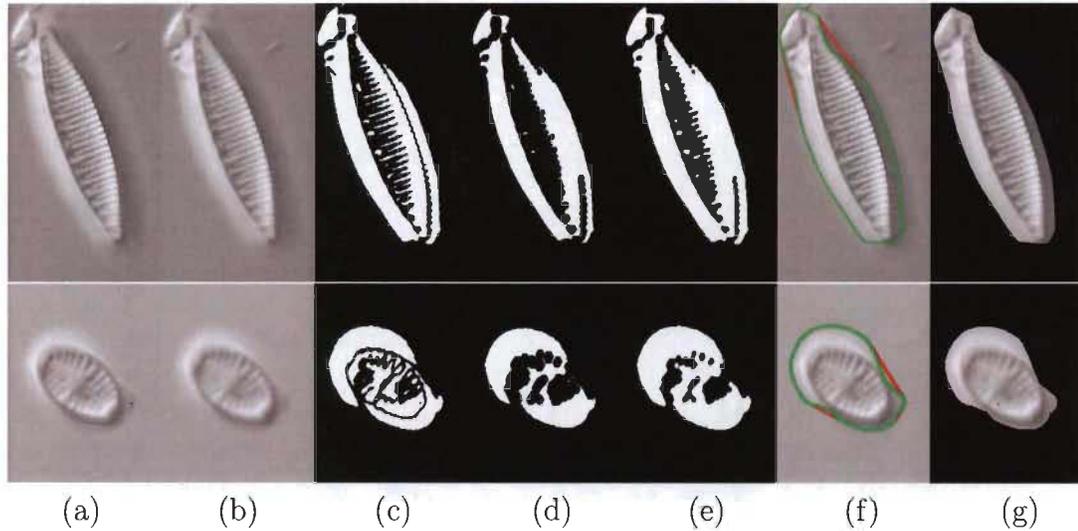


FIGURE 5.3 – Images découlant du processus de détection et d'extraction de diatomée (résultats où la procédure a échoué) : (a) Image originale avec reflets (b) Filtrage gaussien (c) Segmentation (d) Ouverture (e) Fermeture et nettoyage des débris (f) Détection du contour (g) Extraction.

5.2 Identification des caractéristiques

Dans cette section, nous montrons les résultats obtenus lors de l'extraction des caractéristiques taxonomiques tel que le raphé, les stries et les cercles concentriques. Nous avons appliqué notre approche à une base de données de 204 images de diatomées qui a été divisée en deux groupes : 190 spécimens possèdent des stries, 7 spécimens n'en possèdent pas. Parmi ces images, 145 ont un raphé visible, tandis que 52 images n'en possède pas ou il s'avère impossible de le distinguer. Pour l'étude des diatomées centrales, notre base de données se limite à 5 spécimens distincts.

Dans le tableau suivant, nous présentons la matrice de confusion obtenue lors de la détection des stries ;

	avec stries	sans stries
avec stries	190	0
sans stries	3	4

À titre d'exemple, les 190 diatomées avec stries ont été classées comme ayant des stries. Pour tous les spécimens de diatomées, avec ou sans stries, nous avons classé dans la bonne catégorie (présence ou absence de stries) nos spécimens avec un taux

de succès de 98.48 %. Tous les spécimens avec stries ont été détectés. Des tests sur un plus grand échantillonnage s'avère nécessaire avant de conclure sur les diatomées sans stries.

Dans le tableau suivant nous présentons la matrice de confusion obtenue lors des tests effectués dans le cadre de la détection du raphé ;

	avec raphé	sans raphé
raphé	135	10
sans raphé	28	24

Notre modèle de détection a permis la détection, de la présence ou de l'absence, du raphé avec un taux de succès de 80.71 %. Nous observons que la précision de cette méthode est principalement affectée par la présence d'ombres ou d'autres irrégularités lesquelles influencent l'histogramme des gradients orientés, principalement si celles-ci se produisent dans la direction de l'axe principal. De plus, les faibles contrastes dans l'image réduisent les intensités du gradient ce qui nuit, dans l'image de HOG, à la détection du raphé.

5.3 Conclusion

Dans ce chapitre nous avons présenté les résultats de notre méthode de segmentation et d'extraction de l'arrière-plan des spécimens de diatomées. Nous observons que notre méthode est précise dans le contexte des images traitées. Néanmoins, l'amélioration de la qualité des images, principalement lors de la saisie, est un aspect qui doit être amélioré. Cela nous permettra d'augmenter la performance.

Conclusion et travaux futurs

Dans ce mémoire, nous avons d'abord adapté une approche de segmentation d'images dont l'histogramme des niveaux de gris présente une distribution unimodale centrée, introduite par Rosin [20]. Cette méthode peut être utilisée pour des images autres que les diatomées. Cette méthode est robuste et s'adapte à la position du pic dominant dans l'histogramme des niveaux de gris. Si le pic est positionné vers la gauche, le seuil à gauche tend vers zéro et la segmentation est équivalente à celle proposée par Rosin, mais en utilisant les propriétés de la fonction de répartition. Si le pic des intensités est vers la droite, l'approche est symétrique.

Dans le chapitre 5, nous constatons que cette approche de segmentation produit de bons résultats, et sans surprise, la qualité des images extraites demeure affectée par le faible contraste, les reflets de lumière, ainsi que par les particules ou les débris présents dans l'arrière-plan. Aussi, nous constatons que l'utilisation de la force d'attraction entre des régions de l'image, dans le but d'éliminer les débris ou fusionner des régions appartenant à la diatomée, contribue à l'obtention d'une première approximation du domaine représentant une diatomée dans une image. Finalement, le raffinement de l'enveloppe convexe des régions fusionnées d'une diatomée, nous permet d'obtenir une représentation fidèle de celle-ci. Par ailleurs, une comparaison entre les diatomées qui seraient extraites avec notre approche et ces mêmes diatomées isolées manuellement par des spécialistes, nous permettrait de mieux apprécier la qualité des images résultantes. A priori, avec notre approche, nous pourrions anticiper des variations locales du contour mais de façon générale, les paramètres de

forme tel que la convexité, l'étirement, la circularité demeureraient globalement inchangés. Finalement, et de toute évidence, un ajustement du mode de saisie des images réduisant les effets de surbrillance, principalement sur les bords de la diatomée et au niveau des structures internes, améliorerait la qualité des résultats. Par ailleurs, l'élimination du bruit dans le fond de l'image est un défi qui demeure et nous avons résolu ce problème dans les meilleurs cas (débris de petites tailles et isolés).

À partir des images de diatomées extraites du fond, nous avons abordé le problème de la détection (présence ou absence) de structures internes : le raphé, les stries et les cercles internes des diatomées pennées. Dans un premier temps, nous avons construit l'image des gradients orientés de la diatomée et appliqué la transformée de HOG afin d'extraire les structures linéaires. Ces structures peuvent correspondre au raphé ou à des stries.

La position du raphé varie d'une espèce à l'autre. Dans notre projet, le modèle de détection suppose que le raphé est orienté selon l'axe principal de la diatomée et que sa longueur est celle de cet axe. Plusieurs tests ont démontré l'efficacité de cette approche dans le cadre de ces hypothèses. Des travaux supplémentaires et des ajustements seront nécessaires afin de prendre en compte toutes les possibilités de positionnement du raphé.

Aussi, nous observons que cette approche permet la détection des stries lorsque présentes. Par ailleurs, la méthode ne nous garantit pas que tous les segments détectés sont des stries. Afin de départager les stries des faux-positifs, un ou des critères supplémentaires doivent être élaborés. Dans l'état actuel de nos travaux, nous ne pouvons pas distinguer les différents types de stries (continues, en pointillés, etc.). En l'occurrence, l'emplacement des stries pourrait s'avérer un paramètre discriminant pour identifier leur type.

À la lumière des résultats obtenus, des travaux supplémentaires doivent être menés dans un premier temps sur l'amélioration du contraste des images, soit au niveau de la saisie ou des prétraitements. Nous avons pu constater que la transformée de HOG donne de bons résultats pour la détection des stries. Cette détection est à bonifier afin d'extraire un maximum de stries et d'en estimer la densité ainsi que leur texture. Ces paramètres de la structure interne seront couplés aux autres caractéristiques taxonomiques étudiées à ce jour et serviront à la classification à l'espèce dans une étude ultérieure.

Mathématiques de base

Dans cette annexe, nous présentons les résultats mathématiques à la base du calcul des seuils de segmentation à partir de la fonction de répartition de la fonction de densité des niveaux de gris dans une image.

Pour les images de diatomées, la fonction de répartition s'apparente à une sigmoïde, soit la fonction logistique. Le modèle général de cette fonction est donné par l'expression :

$$f_L(t) = \frac{a}{1 + be^{-ct}}, \quad (\text{A.1})$$

où a , b et c sont des constantes réelles. Le graphe de cette courbe est centré à l'origine et les extrémités sont asymptotiques à la valeur $y = a$ à droite et $y = 0$ à gauche. Après plusieurs tests sur nos images de diatomées, nous limiterons notre étude au cas où $b = 1$, $a > 0$ et $c > 0$.

Dans un premier temps, nous approchons (régression) la fonction de répartition d'une image donnée par une fonction logistique afin de trouver les valeurs de a et c correspondantes. Les calculs théoriques qui suivent sont effectués à partir du modèle obtenu. Tel que mentionné dans le chapitre 4, la méthode utilisée pour déterminer les deux seuils limites (extrémités de l'intervalle) des niveaux de gris correspondant à la diatomée et aux débris, nécessite de tracer deux droites sécantes à la fonction de répartition (CDF), dont une des extrémités est un point tangent à la sigmoïde.

Nous déduirons de façon analytique les équations permettant de trouver les points tangents de chacune de ces droites. Le calcul des deux abscisses qui correspondent aux projections de la plus grande longueur de ces droites sur la sigmoïde sont par la suite calculées (voir Figure 4.3).

A.1 Calcul des points tangents à la sigmoïde

Soient $p_m = (t_m, f_L(t_m))$ et $p_M = (t_M, f_L(t_M))$ deux points fixés aux extrémités de la courbe $f_L(t)$. Si nous supposons que la fonction logistique est symétrique par rapport à l'axe des y , le point p_m a une abscisse $t_m < 0$ et le point est situé à gauche et dans la partie inférieure de la CDF. Le point p_M a une abscisse $t_M > 0$ et le point est situé à droite et dans la partie supérieure de la CDF. La fonction de répartition est une fonction positive définie dans le domaine discret que nous centrons à l'origine.

Nous cherchons un premier point $p_1 = (t_1, f_L(t_1))$ tel que la droite passant par le point p_M est tangente à la courbe en p_1 . La valeur t_1 que nous cherchons est une solution de l'équation :

$$f'(t) = \frac{y_M - f_L(t)}{t_M - t} \iff H(t) = f'(t) - \frac{y_M - f_L(t)}{t_M - t} \quad (\text{A.2})$$

$$= \left(\frac{c}{a}\right) e^{-at} f_L^2(t) - \frac{y_M - f_L(t)}{t_M - t} \quad (\text{A.3})$$

$$= 0 \quad (\text{A.4})$$

ou de façon équivalente, t_1 est la racine négative de la fonction

$$h(t) = \left(\frac{c}{a}\right) e^{-at} f_L^2(t)(t_M - t) - y_M + f_L(t). \quad (\text{A.5})$$

Nous pouvons trouver la racine de $h(t)$ à l'aide d'une méthode numérique, comme la méthode de bisection, en tenant compte que $t_1 < 0$. De façon similaire, nous cherchons un second point $(t_2, f_L(t_2))$ tel que la droite passant par ce point et par $(t_m, f_L(t_m))$ est tangente à $f_L(t)$. Nous obtenons une condition similaire à la précé-

dente,

$$f'(t) = \frac{f_L(t) - y_m}{t - t_m} \iff H(t) = f'(t) - \frac{f_L(t) - y_m}{t - t_m} \quad (\text{A.6})$$

$$= \left(\frac{c}{a}\right) e^{-at} f_L^2(t) - \frac{f_L(t) - y_m}{t - t_m} \quad (\text{A.7})$$

$$= 0 \quad (\text{A.8})$$

ou de façon équivalente, le point t_2 est la racine négative de la fonction

$$h(t) = \left(\frac{c}{a}\right) e^{-at} f_L^2(t)(t - t_m) + y_m - f_L(t). \quad (\text{A.9})$$

Le point d'abscisse t_2 recherché est positif et nous pouvons l'approcher par une méthode numérique.

A.2 Points de la CDF à distance maximale des droites tangentes

Notons $y_1 = f_L(t_1)$, $y_2 = f_L(t_2)$, $y_m = f_L(t_m)$, $y_M = f_L(t_M)$. Notons $z_1(t)$ la droite reliant le point (t_1, y_1) au point (t_M, y_M) et $z_2(t)$ la droite reliant le point (t_m, y_m) au point (t_2, y_2) ;

$$z_1(t) = y_1 + \frac{y_M - y_1}{t_M - t_1}(t - t_1), \quad (\text{A.10})$$

$$z_2(t) = y_2 + \frac{y_2 - y_m}{t_2 - t_2}(t - t_2). \quad (\text{A.11})$$

Les deux valeurs de seuillage T_1 et T_2 recherchées correspondent aux abscisses respectives des projections orthogonales maximales de deux points de la CDF sur les droites sécantes calculées à l'étape du calcul des points tangents.

Le vecteur directeur de la droite $z_1(t)$ est $(t_M - t_1, y_M - y_1)$. Afin de trouver la valeur $T_2 > 0$, nous maximisons la hauteur du parallélogramme construit sur les côtés ;

$$\vec{u}(t) = (t - t_1, f_L(t) - y_1) \quad \text{et} \quad v = (t_M - t_1, y_M - y_1). \quad (\text{A.12})$$

La hauteur $h(t)$ est donnée par ,

$$h(t) = \frac{\|v \times \vec{u}(t)\|}{\|v\|} = \frac{(t_M - t_1)(f_L(t) - y_1) - (t - t_1)(y_M - y_1)}{\|v\|} \quad (\text{A.13})$$

Étant donné que le dénominateur est constant, nous maximisons le numérateur. Nous dérivons celui-ci par rapport à t et posons cette dérivée égale à zéro. Alors, la valeur t recherchée vérifie,

$$\frac{d}{dt}(t_M - t_1)(f_L(t) - y_1) - (t - t_1)(y_M - y_1) = 0 \quad \iff \quad (\text{A.14})$$

$$(t_M - t_1)f'_L(t) - (y_M - y_1) = 0 \quad \iff \quad (\text{A.15})$$

$$f'_L(t) = \frac{y_M - y_1}{t_m - t_1} \quad (\text{A.16})$$

Nous concluons que le point T_2 recherché correspond au point de tangence (t_2, y_2) de la CDF trouvé précédemment. Afin de trouver la valeur de T_2 , nous calculons la composante α_2 du vecteur reliant (t_1, y_1) à (t_2, y_2) le long de la direction de la droite $z_1(t)$. Nous obtenons,

$$\alpha_2 = (t_2 - t_1, y_2 - y_1) \frac{t_M - t_1, y_M - y_1}{\|(t_M - t_1, y_M - y_1)\|}. \quad (\text{A.17})$$

Finalement, nous avons,

$$T_2 = t_1 + \alpha_2 \frac{(t_M - t_1)}{\|(t_M - t_1, y_M - y_1)\|}. \quad (\text{A.18})$$

De façon similaire, nous trouvons que le second seuil recherché est :

$$T_1 = t_2 + \alpha_1 \frac{(t_m - t_2)}{\|(t_m - t_2, y_M - y_2)\|}. \quad (\text{A.19})$$

Implémentation

Nous avons développé notre projet entièrement en Python. Dans cette annexe, nous présentons les parties principales de l'implémentation des méthodes proposées dans ce travail. Nous utilisons principalement les modules `Numpy` et `OpenCV`. Tout autre module nécessaire sera mentionné dans la section concernée. Tous les modules sont chargés en utilisant la commande `import` puis le nom du module.

B.1 Prétraitement des images

Filtres et segmentation

Nous chargeons les images en utilisant la commande `imread` du module `cv2` en `OpenCV`. Le premier paramètre de cette commande correspond au nom du fichier et le deuxième est une valeur entière qui spécifie le type de sortie ; nous mettons égal à zéro ce paramètre pour définir une image en niveaux de gris. Par la suite, nous appliquons un filtre Gaussien (`GaussianBlur`) pour lisser l'image.

```
import cv2
img_ori = cv2.imread(file, 0)
img_blu = cv2.GaussianBlur(img_ori, (7,7), 0)
```

Nous définissons la fonction `Sigmoid_CDF_Threshold` qui servira à la segmentation de l'histogramme des niveaux de gris tel que décrite au Chapitre 4. Cette fonction est composée de quatre blocs : définition des variables, recherche des points initial et final des droites sécantes, recherche des seuils à gauche et à droite, et segmentation

de l'image.

```
def Sigmoid_CDF_Threshold(img_ori):
    _shape = np.shape(img_ori)
    img_thd = np.zeros(_shape)
    hist, b = np.histogram(img_ori.flatten(), 256, [0, 255])
    hist = hist / (np.max(hist) * 1.0)
    probab = hist / (np.sum(hist) * 1.0)
    cdf = Cumulative_Distribution_Function(probab)
    scdf, p = Interpolate(cdf, function="sigmoid")
    x = np.arange(len(scdf)) / 256.0

    # Find the initial points
    il0, ir0 = 0, len(scdf) - 1
    xl0, xr0 = 0, 0

    while scdf[il0] == 0:
        xl0 = x[il0]
        il0 += 1
    while scdf[ir0] == 1:
        xr0 = x[ir0]
        ir0 -= 1
    yl0, yr0 = 0, 1

    max_ml, max_mr = 0, 0
    for i in xrange(len(x)):
        ml, mr = 0, 0
        yli, xli = scdf[i], x[i]
        yri, xri = scdf[i], x[i]
        if (xli - xl0) != 0:
            ml = (yli - yl0) / (xli - xl0)
        if (xri - xr0) != 0:
            mr = (yri - yr0) / (xri - xr0)

        if max_ml < ml:
            xl1, yl1, max_ml, il1 = xli, yli, ml, i
        if max_mr < mr:
            xrl, yrl, max_mr, ir1 = xri, yri, mr, i

    # Find the left threshold
    thd_l, max_dist = 0, 0
    for i in xrange(il0, il1):
        xj = ( max_ml**2 * xl0 + x[i] + scdf[i] * max_ml ) / (max_ml**
            2 + 1)

        lxj = max_ml * (xj - xl0)
        d = np.sqrt( (x[i] - xj)**2 + (scdf[i] - lxj)**2 )
        if d > max_dist:
            _xjl, _lxjl = xj, lxj
            max_dist, thd_l = d, i

    # Find the right threshold
    thd_r, max_dist = 0, 0
    c = -max_mr * x[ir1] + scdf[ir1]
    for i in xrange(ir1, 256):
        xj = ( scdf[i]*max_mr - c*max_mr + x[i] ) / (max_mr**2 + 1)
        lxj = max_mr * (xj) + c
        d = np.sqrt( (x[i] - xj)**2 + (scdf[i] - lxj)**2 )
        if d > max_dist:
            _xjr, _lxjr = xj, lxj
            max_dist, thd_r = d, i

    # Threshold the image
    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            if thd_l <= img_ori[i, j] <= thd_r:
                img_thd[i, j] = 0
            else:
                img_thd[i, j] = 255

    return img_thd
```

Nous définissons la fonction de répartition utilisée dans *Sigmoid_CDF_Threshold* de la façon suivante ;

```
def Cumulative_Distribution_Function(prob):
    L = len(prob)
    cdf = np.zeros(L)
    cdf[0] = prob[0]
    for i in xrange(L - 1):
        cdf[i + 1] = cdf[i] + prob[i + 1]
    return cdf
```

Après la segmentation, nous appliquons successivement les opérations morphologiques d'ouverture, de fermeture et de dilatation.

```
KER_CLO = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (5, 5))
KER_OPE = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (5, 5))

img_clo = cv2.morphologyEx(img_seg, cv2.MORPH_CLOSE, KER_CLO)
img_ope = cv2.morphologyEx(img_clo, cv2.MORPH_OPEN, KER_OPE)
img_dil = cv2.morphologyEx(img_ope, cv2.MORPH_ERODE, KER_OPE)
```

Détection du contour

La fonction développée pour détecter le contour de la diatomée utilise plusieurs autres fonctions que nous avons construites. D'abord, nous décrivons la fonction *Primitive_Contour* laquelle sert à trouver l'ensemble des points du contour γ . La sortie de cette fonction est la liste des points initiaux du contour, puis les points qui interceptent la diatomée verticalement et horizontalement. Nous identifions également le point de départ du contour.

```
def Primitive_Contour(image):
    _shape = np.shape(image)

    contour = []
    for i in xrange(_shape[0]):
        lower = Lower_in_Stack(image[i, :])
        if lower != None: contour.append([i, lower])
        higher = Higher_in_Stack(image[i, :])
        if higher != None: contour.append([i, higher])

    depart = None
    for i in xrange(_shape[1]):
        lower = Lower_in_Stack(image[:, i])
        if lower != None:
            depart = [lower, i]
            contour.append([lower, i])
        higher = Higher_in_Stack(image[:, i])
        if higher != None:
            contour.append([higher, i])

    return Unique(contour), depart
```

Nous définissons la fonction *Order_Contour* qui effectue l'ordonnement, dans le sens horaire, du contour initial. Nous commençons par le point de départ et nous vérifions les points adjacents à celui-ci. Le premier point du contour initial rencontré dans le sens horaire est mis comme le suivant dans l'ordonnement. Si aucun des voisins n'est dans le contour initial, nous cherchons le point sur le contour le plus près (distance et direction) du point courant pour le mettre comme prochain point. Nous répétons cette procédure à partir du dernier point retenu et cela jusqu'à épuisement de la liste.

```
def Order_Contour(contour, depart, image):
    _shape = np.shape(image)
    indices = [[0,-1], [1,-1], [1,0], [1,1],
               [0,1], [-1,1], [-1,0], [-1,-1]]

    dic_points = {}
    for i in xrange(len(contour)):
        s = "%i, %i" % (contour[i][0], contour[i][1])
        dic_points.update({s:i})

    matrix = np.zeros((len(contour), len(contour)))
    for i in xrange(len(contour)):
        for j in xrange(i, len(contour)):
            dist = Distance(contour[i], contour[j])
            matrix[i, j], matrix[j, i] = dist, dist

    last_point = depart
    ord_con = [depart]

    last_key = "%i, %i" % (depart[0], depart[1])
    dept_key = last_key
    last_idx = dic_points[last_key]
    dic_points.pop(last_key)
    for i in xrange(1, len(contour)):
        found = False
        subwin = Sub_Window(image, (last_point[1], last_point[0]), 1)
        for idx in xrange(8):
            neighbor = [1 + indices[idx][0], 1 + indices[idx][1]]
            next_point = [last_point[0] + indices[idx][1],
                          last_point[1] + indices[idx][0]]
            key = "%i, %i" % (next_point[0], next_point[1])

            if np.shape(subwin)[0] != 3 or np.shape(subwin)[1] != 3:
                continue

            if subwin[neighbor[1]][neighbor[0]] == 255 and key in dic_points.keys():
                found = True
                last_key = key
                last_idx = dic_points[key]
                dic_points.pop(key)
                break

        if not found:
            next_id, dist_min = 0, float('inf')
            for ids in xrange(len(contour)):
                next_key = "%i, %i" % (contour[ids][0], contour[ids][1])
                if matrix[last_idx, ids] < dist_min and ids != last_idx and
                    next_key in dic_points.keys():
                    dist_min = matrix[last_idx, ids]
                    next_id = ids
```

```

        next_point = contour[next_id]
        last_key = "%i, %i" % (next_point[0], next_point[1])
        last_idx = dic_points[last_key]
        dic_points.pop(last_key)

        ord_con.append(next_point)
        last_point = next_point

ord_con = np.array(ord_con)
return ord_con

```

La fonction *ConvexHull* du module `scipy.spatial` [2] permet de trouver l'enveloppe convexe des points du contour. Cette fonction est incorporée à la procédure *Convex_Contour* pour conserver l'ordre et le format du contour en fonction de nos exigences de calcul.

```

def Convex_Contour(contour):
    hull = ConvexHull(contour)
    cvx_con = []
    for i in xrange(len(contour)):
        if i in hull.vertices: cvx_con.append(contour[i])
    cvx_con.append(cvx_con[0])
    return np.array(cvx_con)

```

Finalement, nous définissons la fonction *Contour* qui utilise les fonctions décrites ci-dessus et qui raffine le contour convexe pour obtenir la forme générale de la diatomée ou les fragments qui la composent. La fonction comporte trois parties principales : trouver le contour original et l'enveloppe convexe, détecter les segments atypiques, raffiner chacun des segments atypiques.

```

def Contour(img_ori, show=True):
    _shape = np.shape(img_ori)

    # primitive and convex contour
    contour, depart = Primitive_Contour(img_ori)
    contour = Order_Contour(contour, depart, img_ori)
    cvx_con = Convex_Contour(contour)

    distances = []
    for i in xrange(len(cvx_con) - 1):
        distances[i] = Distance(cvx_con[i], cvx_con[i + 1])
    mean_dst = np.mean(distances)
    Q1 = np.percentile(distances, 25)
    Q3 = np.percentile(distances, 75)
    Ls = Q3 + ((Q3 - Q1) * 1.5)

    atypics_lines = np.where(Ls <= distances)[0]
    ref_con = []
    for i in xrange(len(cvx_con)):
        pt_0 = cvx_con[i]
        ref_con.append(pt_0)
        if i in atypics_lines:
            pt_1 = cvx_con[i + 1]
            a = Index_in_Array(contour, pt_0)

```

```

b = Index_in_Array(contour, pt_1)
mid_pts = contour[a:b+1]

# Split points
a_side = np.array(mid_pts[:len(mid_pts) / 2])
b_side = np.array(mid_pts[len(mid_pts) / 2:])
# Find baricenter
pt_a = np.sum(a_side, axis=0) / (len(a_side))
pt_b = np.sum(b_side, axis=0) / (len(b_side))
pt_c = (pt_0 + pt_1) / 2
# Find the relative points position
sign_correction = 1
sign_a = Cross_Product(pt_0, pt_a, pt_c)
sign_b = Cross_Product(pt_1, pt_b, pt_c)
if sign_a != sign_b:
    sign_correction = -1
    sign_b *= sign_correction
# Select points on side "sign_a = sign_b"
selected = []
for pt in mid_pts:
    sign_1 = Cross_Product(pt_0, pt_a, pt)
    sign_2 = Cross_Product(pt_1, pt_b, pt) * sign_correction
    if sign_1 == 0 or sign_2 == 0:
        selected.append(pt)
    if abs(sign_1 + sign_2 + sign_a + sign_b) == 4:
        selected.append(pt)
selected = np.array(selected)

length = len(selected)
if length == 0: continue
# Refine the typic segment
factor = 2
max_distance = mean_dst + 1
ids = []
while max_distance > mean_dst:
    chunk = length / factor
    if chunk < factor or chunk == 0: break
    ids = np.arange(length / chunk + 1)
    ids = ids * chunk
    ids[-1] = length - 1

    local_distances = []
    for j in xrange(len(ids) - 1):
        aux_1 = selected[ids[j]]
        aux_2 = selected[ids[j + 1]]
        local_distance[j] = Distance(aux_1, aux_2)
    max_distance = np.max(local_distances)
    factor += 1

# Add subsegments to the final contour
for j in ids:
    ref_con.append(selected[j])

return np.array(ref_con), cvx_con

```

B.1.1 Extraction de la diatomée

Nous utilisons la procédure qui suit pour extraire la diatomée de l'arrière-plan. En considérant le contour comme un polygone, épuré à l'étape précédente, nous ne conservons que les pixels de l'image qui sont à l'intérieur de ce contour. Nous avons besoin des modules `shapely.geometry` et `shapely.geometry.polygon` pour mani-

puler les objets Point et Polygon respectivement.

```
def Cut(img_ori, contour):
    _shape = np.shape(img_ori)
    polygon = Polygon(contour)
    img_cut = np.zeros(_shape)
    img_mas = np.zeros(_shape)
    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            point = Point(i, j)
            if polygon.contains(point):
                img_mas[i, j] = 255
                img_cut[i, j] = img_ori[i, j]
    return img_cut, img_mas
```

B.1.2 Amélioration du contraste

Pour améliorer le contraste de l'image de la diatomée, lorsqu'elle a été extraite de l'arrière-plan, nous utilisons la méthode d'égalisation d'histogramme implémentée comme suit ;

```
def Histogramme_Normalization(img_ori, mask = None):
    size = np.shape(img_ori)
    if mask is None:
        img_fla = img_ori.flatten()
    else:
        img_fla = []
        for i in xrange(size[0]):
            for j in xrange(size[1]):
                if mask[i, j] != 0:
                    img_fla.append(img_ori[i, j])
    hist, bins = np.histogram(img_fla, 256, [0,256])
    hist = hist / (np.max(hist) * 1.0)
    prob = hist / (np.sum(hist) * 1.0)
    cdf = Cumulative_Distribution_Function(prob)
    img_nor = np.zeros(size)
    for i in xrange( size[0] ):
        for j in xrange( size[1] ):
            idx = int(img_ori[i,j])
            img_nor[i,j] = floor( (255) * cdf[ idx ] )
    return img_nor, hist
```

B.1.3 Rotation et recadrage

Le recadrage de l'image de la diatomée dépend de la rotation de l'image. À cet effet, nous appliquons la fonction ci-dessous.

```

def Orientate(image, mask):
    padding = 10

    _shape = np.shape(image)
    points = np.transpose(np.array(np.where(mask != 0)))

    y_min = np.min(points[:, 0])
    x_min = np.min(points[:, 1])
    y_max = np.max(points[:, 0])
    x_max = np.max(points[:, 1])

    ys_min = points[np.where(points[:, 0] == y_min)]
    xs_min = points[np.where(points[:, 1] == x_min)]
    ys_max = points[np.where(points[:, 0] == y_max)]
    xs_max = points[np.where(points[:, 1] == x_max)]

    pt_ymin = np.sum(ys_min, axis=0) / len(ys_min)
    pt_xmin = np.sum(xs_min, axis=0) / len(xs_min)
    pt_ymax = np.sum(ys_max, axis=0) / len(ys_max)
    pt_xmax = np.sum(xs_max, axis=0) / len(xs_max)

    d_ver = Distance(pt_ymin, pt_ymax)
    d_hor = Distance(pt_xmin, pt_xmax)
    if d_hor >= d_ver:
        pt_a = pt_xmin
        pt_b = pt_xmax
    else:
        pt_a = pt_ymax
        pt_b = pt_ymin

    theta = np.arctan( abs(pt_b[0] - pt_a[0]) / (1.0 * abs(pt_b[1]
        - pt_a[1])) )

    if d_hor >= d_ver:
        if pt_a[0] > pt_b[0]:
            theta *= -1
    else:
        if pt_a[1] < pt_b[1]:
            theta *= -1

    mat_rot = np.array([[np.cos(theta), -np.sin(theta)], [np.sin(
        theta), np.cos(theta)]])

    _points = np.dot(mat_rot, (points - pt_a).T).T + pt_a

    y_min = int(np.min(_points[:, 0]))
    x_min = int(np.min(_points[:, 1]))
    y_max = int(np.max(_points[:, 0]))
    x_max = int(np.max(_points[:, 1]))

    image_2 = np.zeros((y_max - y_min + padding, x_max - x_min +
        padding))

    for i in xrange(len(_points)):
        pt = _points[i]
        aux_1 = int(np rint(pt[0] - y_min)) + (padding / 2)
        aux_2 = int(np rint(pt[1] - x_min)) + (padding / 2)
        image_2[aux_1, aux_2] = image[points[i, 0], points[i, 1]]

    _shape = np.shape(image_2)
    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            win = Sub_Window(image_2, (j, i), 1)
            if image_2[i, j] == 0.0 and np.sum(win) != 0.0:
                image_2[i, j] = (np.sum(win) / 8.0)

    pts = np.array([ pt_a, pt_b ])
    _pts = (pts - pt_a).T
    _pts = np.dot(mat_rot, _pts).T
    _pts += pt_a
    _pts = _pts - (y_min, x_min) + (padding / 2.0, padding / 2.0)

    return image_2, mat_rot, _pts

```

B.2 Extraction des caractéristiques

Pour extraire les caractéristiques taxonomiques des diatomées, nous utilisons la méthode de HOG. Cette méthode est incluse dans le module `skimage.feature`. Dans la procédure suivante, nous calculons le résultat de la transformation de HOG. Lors de l'application de cette procédure, les faibles intensités sont filtrées.

```
def HOG_Descriptors(img_ori, oriens=4, pixCell=(10,10)):
    _shape = np.shape(img_ori)

    fd, img_hog = hog(img_ori,
                      orientations=oriens,
                      pixCell=pixCell,
                      cells_per_block=(1, 1))

    fd = np.reshape(fd, (len(fd) ))
    img_hog = Visualize_HOG(fd, _shape, oriens, pixCell[0])
    img_hog = (img_hog / (1.0* np.max(img_hog))) * 255.0

    img_fla = img_hog[np.nonzero(img_hog)].flatten()
    Q1, Q3 = np.percentile(img_fla, 25), np.percentile(img_fla, 75)
    Ls = Q3 + ((Q3 - Q1) * 1.5)

    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            if np.sum(img_hog[i, j]) != 0:
                if np.sum(img_hog[i, j]) < Q3:
                    img_hog[i, j] = 0

    Q1, Q3 = np.percentile(fd, 25), np.percentile(fd, 75)
    Ls = Q3 + ((Q3 - Q1) * 1.5)

    idx = 0
    h_cells = _shape[0] / pixCell[0]
    w_cells = _shape[1] / pixCell[1]
    dirs = np.zeros((orientations, h_cells, w_cells))
    for h in xrange(h_cells):
        for w in xrange(w_cells):
            for j in xrange(orientations):
                dirs[j, h, w] = fd[idx]
                idx += 1

    return dirs, img_hog
```

Avant de procéder à la détection des structures internes, nous éliminons une partie du bruit qui conduirait à la détection de faux positifs. À cet effet, nous filtrons les petites composantes obtenues suite à l'application du filtre de Canny. Pour chaque pixel dans l'image, nous prenons deux fenêtres : la première contient le pixel courant et les 8 voisins les plus proches, la seconde contient les 15 voisins les plus proches. Si le nombre de pixels blancs dans les deux fenêtres est le même, la région est isolée, donc éliminée.

```

def FilterSingletons(img_ori):
    _shape = np.shape(img_ori)

    img_fil = np.zeros(_shape)
    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            if img_ori[i, j] != 0:
                img_fil[i, j] = img_ori[i, j]
                subWin1 = np.sum(Sub_Window(img_ori, [j, i], 1))
                subWin2 = np.sum(Sub_Window(img_ori, [j, i], 2))
                if subWin1 == subWin2:
                    img_fil[i, j] = 0
                    for h in xrange(i - 1, i + 1):
                        for k in xrange(j - 1, j + 1):
                            if _shape[0]>h>0 and _shape[1]>k>0:
                                img_fil[h, k] = 0

    return img_fil

```

La procédure suivante utilise les descripteurs de HOG pour détecter le raphé. Nous éliminons le contour extérieur de la diatomée et nous filtrons les petites composantes avec la procédure précédente.

```

def Raphe_Detection(_imgCan, img_ori, _orientations=4, hog_win=8):
    imgCan = np.copy(_imgCan)
    _shape = np.shape(imgCan)

    contour = np.array(Primitive_Contour(imgCan)[0])
    max_x, min_x = np.max(contour[:, 1]), np.min(contour[:, 1])
    max_y, min_y = np.max(contour[:, 0]), np.min(contour[:, 0])
    width = max_x - min_x
    height = max_y - min_y

    imgCan = EliminateContour(imgCan, contour, 3)
    hogCan, imgHOGcur = HOG_Descriptors(imgCan,
                                       orientations=_orientations,
                                       pixCell=(hog_win, hog_win))

    imgHOGcur = FilterSingletons(Otsu(imgHOGcur)).astype(np.uint8)
    lines = cv2.HoughLinesP(imgHOGcur,
                            rho=1,
                            theta=np.pi/180,
                            threshold=int(_shape[1] * .7),
                            minLineLength=_shape[1] * .9,
                            maxLineGap=50)

    min_y, max_y = _shape[1], 0
    x1, y1, x1, x2 = 0,0,0,0
    lines = []
    dic_xs = {}

    for i in xrange(_shape[0]):
        isLine = False
        count = 0

        for j in xrange(_shape[1]):
            if imgHOGcur[i, j] != 0:
                if isLine:
                    count += 1
                else:
                    count = 1
                    isLine = True
                    x1, y1 = i, j
                    if min_y > y1:
                        min_y = y1
                else:
                    if isLine and count >= 5:

```

```

        x2, y2 = i, j
        if y2 > max_y:
            max_y = y2
        lines.append([x1, y1, x2, y2])
        if x1 not in dic_xs.keys():
            dic_xs.update({x1:0})
        dic_xs[x1] += Distance([x1, y1], [x2, y2])
        count = 0
        isLine = False

sorted_keys = sorted(dic_xs.keys())
for i in xrange(len(sorted_keys) - 1):
    k1 = sorted_keys[i]
    k2 = sorted_keys[i+1]
    if abs(k1 - k2) - hog_win == 0:
        dic_xs.update({(k1+k2)/2:0})
sorted_keys = sorted(dic_xs.keys())

longest = {}
for i in xrange(len(sorted_keys)):
    k = sorted_keys[i]
    value = dic_xs[k] / width

    if k not in longest.keys():
        longest.update({k:value})
    else:
        longest[k] += value

    if i-1 > 0:
        key_1 = int(k - (hog_win / 2))
        k_aux = sorted_keys[i-1]
        if 2 > abs(k_aux - key_1):
            if k_aux not in longest.keys():
                longest.update({k_aux:0})
            if value > .35:
                longest[k_aux] += value
    if len(dic_xs) > i+1 :
        key_1 = int(k + (hog_win / 2))
        k_aux = sorted_keys[i+1]
        if 2 > abs(k_aux - key_1):
            if k_aux not in longest.keys():
                longest.update({k_aux:0})
            if value > .35:
                longest[k_aux] += value

imgHOGcur = cv2.cvtColor(imgHOGcur.astype(np.uint8),
                        cv2.COLOR_GRAY2RGB)

raphe = []
lengths = np.array([[k, longest[k]] for k in longest.keys()])
if len(lengths) != 0:
    if np.max(lengths[:,1]) >= 0.7:
        idx = np.argmax(lengths[:,1])
        x = lengths[idx, 0]
        x1, y1, x2, y2 = int(x), int(min_y), int(x), int(max_y)
        raphe = np.array([[x1, y1], [x2, y2]])
        imgHOGcur = DrawContour(imgHOGcur, raphe, (0, 255, 0))
    else:
        raphe = []

return imgHOGcur, raphe

```

Nous avons développé une procédure similaire à la précédente pour la détection des stries, mais dans ce cas, nous prenons les directions du gradient qui sont dans la direction orthogonale à l'axe principal de la diatomée.

```

def Stries_Detection(_imgCan, img_ori, hog_win = 10):
    imgCan = np.copy(_imgCan)
    _shape = np.shape(imgCan)

    contour = np.array(Primitive_Contour(imgCan)[0])
    imgCan = EliminateContour(imgCan, contour, 3)

    hog_can, imgHOGcur = HOG_Descriptors(imgCan,
                                         orientations=4,
                                         pixCell=(hog_win, hog_win))

    imgHOGcur = FilterSingletons(Otsu(imgHOGcur)).astype(np.uint8)

    x1, y1, x1, x2 = 0,0,0,0
    lines = []
    for i in xrange(_shape[1]):
        isLine = False
        count = 0

        for j in xrange(_shape[0]):
            if imgHOGcur[j, i] != 0:
                if isLine:
                    count += 1
                else:
                    count = 1
                    isLine = True
                    x1, y1 = i, j
            else:
                if isLine and count >= hog_win-2:
                    x2, y2 = i, j
                    lines.append([x1, y1, x2, y2])
                    count = 0
                    isLine = False

    imgHOGcur = cv2.cvtColor(imgHOGcur.astype(np.uint8),
                             cv2.COLOR_GRAY2RGB)

    for line in lines:
        x1, y1, x2, y2 = line[0], line[1], line[2], line[3]
        strie = np.array([[y1, x1], [y2, x2]])
        imgHOGcur = DrawContour(imgHOGcur, strie, (0, 255, 0))

    return imgHOGcur, lines

```

La procédure pour détecter les cercles concentriques et les stries radiales, est décrite ci-dessous.

```

def FindCircles(img_can, img_ori):
    _shape = np.shape(img_ori)
    img_ori = cv2.cvtColor(img_ori.astype(np.uint8),
                           cv2.COLOR_GRAY2RGB)
    img_can1 = FilterSingletons(img_can)

    points = []
    for i in xrange(_shape[0]):
        for j in xrange(_shape[1]):
            if img_can1[i, j] != 0:
                points.append([i, j])
    points = np.array(points)
    c = (int(np.mean(points[:, 0])), int(np.mean(points[:, 1])))

    polar_pts = []
    for pt in points:
        r = Distance(c, pt)
        if r > 0:
            theta = np.degrees(Cartesian_Polar(pt - c))
            polar_pts.append([int(round(theta)), int(round(r))])
    polar_pts = np.array(polar_pts)

```

```

min_r = int(np.min(polar_pts[:, 1]))
max_r = int(np.max(polar_pts[:, 1]))
min_a = int(np.min(polar_pts[:, 0]))
max_a = int(np.max(polar_pts[:, 0]))

circles = []
last_r, cont = -99, 0
for radius in xrange(min_r, max_r + 1):
    perimeter = 2 * np.pi * radius
    circle = []

    idx = []
    for i in xrange(len(polar_pts)):
        pt = polar_pts[i]
        if radius == pt[1]:
            circle.append(pt)
            idx.append(i)

    if len(circle) > 2:
        circle = np.array(circle)

    if len(circle) / perimeter >= .30:
        circle = np.array(circle)
        if radius - last_r < 2:
            circles[cont-1] = np.concatenate((circles[cont-1], circle),
                                              axis=0)
        else:
            circles.append(circle)
            cont += 1
            last_r = radius

    polar_pts = np.delete(polar_pts, idx, 0)

radius = []
for circle in circles:
    mean = int(round(np.mean(circle[:, 1])))
    radius.append(mean)
    circle[:, 1] = mean

stries = []
cont = 0
for i in xrange(len(radius) - 1):
    r1, r2 = radius[i], radius[i+1]

    idx = np.where(polar_pts[:, 1] < r2)[0]
    points = polar_pts[idx]
    polar_pts = np.delete(polar_pts, idx, 0)

    last_a = -99
    for angle in xrange(min_a, max_a + 1):
        strie = []

        for pt in points:
            if angle == pt[0]: strie.append(pt)

        if len(strie) > 2:
            strie = np.array(strie)

        if len(strie) / (1.0 * (r2 - r1)) >= .50:
            strie = np.array(strie)
            if angle - last_a < 2:
                stries[cont-1] = np.concatenate((stries[cont-1], strie),
                                                axis=0)
            else:
                stries.append(strie)
                cont += 1
                last_a = angle

_stries = []
for strie in stries:
    atypics = Atypicals(strie[:, 1])
    strie = np.delete(strie, atypics, 0)

```

```
    if len(strie):
        mean = int(round(np.mean(strie[:, 0])))
        strie[:, 0] = mean
    _stries = np.array(_stries)

    for r in radius:
        cv2.circle(img_ori, c, r, (0, 255, 0), 2)

    for strie in _stries:
        theta, r1, r2 = np.radians(strie[0]), strie[1], strie[2]
        x1 = int(r1*np.cos(theta) + c[1])
        y1 = int(r1*np.sin(theta) + c[0])
        x2 = int(r2*np.cos(theta) + c[1])
        y2 = int(r2*np.sin(theta) + c[0])
        strie = np.array([[y1, x1], [y2, x2]])
        img_ori = DrawContour(img_ori, strie, (0, 255, 0))

    return img_ori
```

Bibliographie

- [1] ANJOS, A., AND SHAHBAZKIA, H. Bi-level image thresholding - a fast method. *BIOSIGNALS 2* (2008), 70–76.
- [2] BARBER, C. B., DOBKIN, D. P., AND HUHDANPAA, H. The quickhull algorithm for convex hulls. *ACM TRANSACTIONS ON MATHEMATICAL SOFTWARE 22*, 4 (1996), 469–483.
- [3] CANNY, J. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence* (1986), 679–698.
- [4] CHAHBOUNE, N., MEHDI, M., AND DOUIRA, A. Automatic classification of diatoms of merja fouarate. *bioRxiv* (2017).
- [5] CLAUDON, N. *Classification automatique des diatomées : Une approche par les motifs des structures internes. Mémoire de maîtrise de l'Université du Québec à Trois-Rivières*. 2007.
- [6] DU BUF, H., AND BAYER, M. M. *Automatic diatom identification*. 2002.
- [7] DU BUF, H., ET AL. *Diatom Identification : A double challenge called ADIAC. In proceedings of the 10th Int. Conf. on image Analysis*. 1999.
- [8] DUDA, R. O., AND HART, P. E. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM 15*, 1 (Jan. 1972), 11–15.
- [9] F. SUARD, A. RAKOTOMAMONJY, A. B. A. B. Pedestrian detection using infrared images and histograms of oriented gradients. *Intelligent Vehicles Symposium* (2006), 13–15.

- [10] GRENIER, M. *Détermination des communautés physobentiques de référence et des objectifs de restauration des rivières du Québec. Mémoire de maîtrise de l'Université du Québec à Trois-Rivières.* 2005.
- [11] H. DU BUF, M. BAYER, S. D., AND HEAD, R. Diatom identification : a double challenge called adiac. *Proceedings 10th International Conference on Image Analysis and Processing* (September 1999).
- [12] HSIUNG, D. *Les diatomées.* 2002.
- [13] LAVOIE, I., ET AL. *Guide d'identification des diatomées des rivières de l'Est du Canada.* 2008.
- [14] LONDON, U. C. An Insight into Micropalaeontology. www.ucl.ac.uk/GeolSci/micropal/, 2008. [Online ; accessed 2018].
- [15] M. EL YASSA, D. MAMMASS, F. N., AND CHALIFOUR, A. Classification de diatomées par une pseudo-distance prétopologique. *Sciences of Electronic Technologies of Information and Telecommunications* (March 2005).
- [16] MAINTZ, T. *Digital and Medical Image Processing.* 2005, ch. 6.
- [17] MATHERON, G. *Éléments pour une théorie des milieux poreux.* 1967.
- [18] M.H.F. WILKINSON, J.B.T.M. ROERDINK, S. D., AND BAYER, M. Diatom contour analysis using morphological curvature scale spaces. *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (September 2000).
- [19] R., W. Nih image to imagej : 25 years of image analysis.
- [20] ROSIN, P. *Measuring sigmoidality, Pattern Recognition*, vol. 37. 2004.
- [21] SERRA, J. *Image Analysis and Mathematical Morphology*, vol. 1. 1982.
- [22] SIDDOUR, A. *Classification automatique des diatomées : Une approche basée sur le contour et la géométrie. Mémoire de maîtrise de l'Université du Québec à Trois-Rivières.* 2007.
- [23] SOBEL, I. An isotropic 3x3 image gradient operator.
- [24] SOLÍS-VENTURA, A., ÁLVAREZ BORREGO, J., AND SOLORZA, S. Image analysis using one binary ring mask invariant to rotation and scale. In *Latin*

- America Optics and Photonics Conference* (2014), Optical Society of America, p. LF1D.4.
- [25] T. ALVAREZ, Y. MARTIN, S. P. F. S. F. T. S. G. J. A., AND VEGA, P. Classification of microorganisms using image processing techniques. *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)* (October 2001).
- [26] VACHIER, C. *Extraction de caractéristiques, segmentation d'image et morphologie mathématique. thèse doctoral de l'École Nationale Supérieure des Mines de Paris.* 1995.