New strategies for the identification and enumeration of macromolecules in 3D images of cryo electron tomography

Emmanuel Moebel

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Signal, Image, Vision*

Par

Emmanuel MOEBEL

"New strategies for the identification and enumeration of macromolecules in 3D images of cryo electron tomography"

Thèse présentée et soutenue à Rennes, le 1er Février, 2019
Unité de recherche : Serpico, Inria Rennes – Bretagne Atlantique
Thèse N° : XXX

Rapporteurs avant soutenance :

Pierre Charbonnier	Directeur de Recherche, Cerema Endsum, ICube, Strasbourg
Carlos Oscar S. Sorzano	Professeur, National Center for Biotechnology (CSIC), Madrid (Espagne)

Composition du Jury :

Président :	Valérie Monbet	Professeure, Université de Rennes 1, IRMAR, Rennes
Examinateurs :	Pierre Charbonnier	Directeur de Recherche, Cerema Endsum, ICube, Strasbourg
	Carlos O. S. Sorzano	Professeur, National Center for Biotechnology (CSIC), Madrid (Espagne)
	Denis Chrétien	Directeur de Recherche, CNRS, IGDR, Rennes
	Julio Ortiz	Chercheur, Forschungszentrum Jülich GmbH, Jülich (Allemagne)
	Damien Larivière	Délégué Général, Fondation Fourmentin-Guilbert
	Thomas Walter	Enseignant-Chercheur, Center for Computational Biology (CBIO), MINES ParisTech, Paris
Dir. de thèse :	Charles Kervrann	Directeur de Recherche, Inria, Rennes
Invité :	Eric Fourmentin	Président de la Fondation Fourmentin-Guilbert

REMERCIEMENTS

J'aimerais remercier en premier lieu Charles Kervrann, qui m'a accompagné durant ces quatre années de recherche. Sa rigueur scientifique et son expérience du monde académique m'ont été d'une aide précieuse. Sa confiance et son optimisme m'ont réconforté dans les moments de doute. Ensuite, j'aimerais remercier les membres de mon jury, qui ont pris de leur temps pour évaluer mon travail. Merci à Huguette pour s'être occupée des aspects administratifs avec application, me permettant ainsi d'être plus focalisé sur ma tâche.

Thanks to W. Baumeister's team (Max Planck Institute of Biochemistry) for providing the data I needed to develop and evaluate the algorithms presented in this thesis. Thanks in particular to J. Ortiz for introducing me to cryo-ET processing, to A. Martinez for his precious advices, and to S. Pfeffer and S. Albert for sharing their macromolecule annotations with me.

Merci à mes camarades de couloir, qui ont apporté une bonne dose d'humour à mon expérience de jeune chercheur. Merci en particulier à Mayela et à Anca, à qui j'ai à de nombreuses reprises fait part de mes états d'âme.

Pour finir, j'aimerais remercier les personnes qui m'ont accompagné dans mon expérience rennaise, sans qui je n'aurais pas eu la stabilité émotionnelle nécessaire pour achever ce travail. Je remercie Maud et Lise du fond du coeur d'avoir partagé cette expérience de vie avec moi, leur amitié a été, et est toujours, mon centre de gravité. Merci à Ben, avec qui j'ai pu développer mon sens artistique, faire de la musique ensemble m'a permis de trouver un équilibre. Enfin, merci à ma famille qui a su m'insuffler la Curiosité qui m'a mené là où je suis aujourd'hui.

J'aimerais déclarer que ce chapitre de ma vie a été le plus excitant, tant d'un point de vue professionnel que personnel.

TABLE OF CONTENTS

TABLE DES FIGURES

RÉSUMÉ DE LA THÈSE

Preambule

Cette thèse a été développée dans le cadre d'une collaboration entre la Fondation Fourmentin-Guilbert et l'Institut Max Planck de biochimie (Pr Baumeister, Martinsried, Allemagne). Tout d'abord, dans le cadre du projet LifeExplorer, la Fondation Fourmentin-Guilbert a été pionnière dans la modélisation structurale et la visualisation de cellules entières.

Dans ce contexte, la création d'avatars 3D interactifs d'environnements cellulaires, reliant le niveau des atomes à celui des cellules, devrait permettre de révéler les règles régissant l'organisation spatio-temporelle du cytoplasme. Une telle approche nécessite de faire un inventaire et une cartographie de tous les composants constituant une cellule unique. La technique de choix pour une telle cartographie est la microscopie cryoélectronique appliquée sur des cellules congelées mais intactes. Depuis des années, la Fondation Fourmentin-Guilbert soutient l'Institut Max Planck de Biochimie, dirigé par Wolfgang Baumeister, dont l'équipe est capable de réaliser des cryotomographies complètes de cellules E. coli avec une résolution inégalée.

L'étape suivante, qui demeure un défi permanent, consistait à reconnaître les composantes macromoléculaires dans les tomogrammes. L'essentiel de l'effort de la communauté scientifique a porté sur l'identification des ribosomes, grâce à une méthodologie reposant sur l'analyse monoparticulaire (*single-particle analysis*) et le *template matching*, donnant des résultats impressionnants. Cependant, il est probable qu'une telle approche sera limitée aux "grands" complexes, tel que les ribosomes. Face à ce défi, la Fondation Fourmentin-Guilbert a sollicité le groupe de recherche Serpico de l'Inria, dirigé par Charles Kervrann, pour développer des méthodes alternatives de reconnaissance ayant le potentiel de contribuer à l'identification in situ des milliers de protéines encore inconnues.

Cette thèse est née de cette initiative. Ses objectifs ont été fixés de la manière suivante : développer et comparer avec des méthodes bien établies de nouvelles approches basées sur l'apprentissage profond (*deep learning*) et capables d'identifier et de compter les ribosomes "étalons" dans un tomogramme. Ces méthodes, comme alternative au *template matching*, devraient également avoir le potentiel de s'appliquer à des particules plus petites et plus difficiles à reconnaitre que les ribosomes.

Contexte

Les dernières décennies de recherche en biologie cellulaire ont révélé que les processus cellulaires sont réalisés par des groupes de macromolécules, en interaction dans un environnement complexe. Ceci est en opposition avec les modèles cellulaires plus anciens où les macromolécules étaient considérées comme des objets isolés, flottant aléatoirement dans le cytoplasme. Dorénavant, il est devenu primordial de déchiffrer les mécanismes d'interaction sous-jacents pour mieux comprendre la cellule.

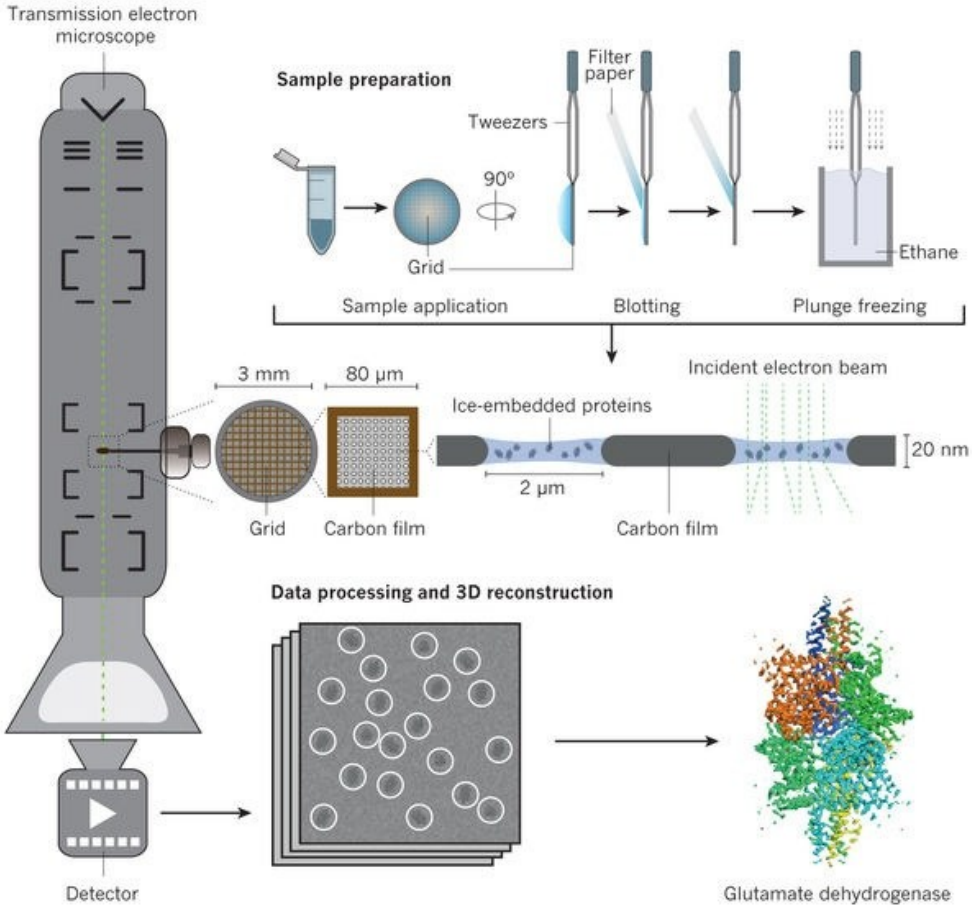

FIGURE 1 – La chaîne d'acquisition de la cryo-ET : de la préparation de l'échantillon au traitement d'image. Source : [Chung and Kim, 2017].

Pour résoudre ce problème, la cryo-tomographie électronique (cryo-ET) est une technologie capable de produire des vues 3D de grandes parties d'une cellule, tout en ayant une résolution suffisante pour localiser et identifier les macromolécules d'intérêt. La cryo-ET permet d'éviter l'utilisation de marqueurs, tels que les sondes fluorescentes utilisées en microscopie optique, qui pourraient perturber la cellule. Au préalable, les échantillons sont vitrifiés afin de préserver à la fois la distribution spatiale et la structure des macromolécules lors du processus d'acquisition des images (voir Figure 1). Cette technique a pour grand avantage de permettre l'étude de cellules dans un état proche de l'état natif, et a donc le potentiel de permettre l'établissement d'un atlas moléculaire à partir de tous les composants détectés (macromolécules, membranes). Cependant, l'analyse de ces images est difficile en raison des faibles rapports signal/bruit observés sur ces images et des artefacts d'imagerie causés par la tomographie à angle limité. L'analyse de données en cryo-ET est ainsi fortement dépendante des méthodes de traitement d'images pour interpréter le contenu des images.

Contributions

Le but de cette thèse est de proposer de nouvelles méthodes de calcul pour faciliter l'identification de macromolécules dans les images 3D de cryo-ET. A terme, nous espérons proposer des outils permettant d'étudier la distribution spatiale de différentes classes de macromolécules. Cela permettrait de mieux comprendre l'interaction entre ces classes, et ce, pour différents états fonctionnels de la cellule.

La première contribution porte sur l'amélioration de l'interprétabilité de l'image, la résolution des problèmes de faible rapports signal/bruit et l'élimination des artefacts d'imagerie. L'origine de ces artefacts réside dans un spectre de Fourier incomplet ; certains coefficients de Fourier ne sont pas mesurés lors de l'acquisition des données. Afin de réduire ces artefacts, la stratégie de la méthode proposée consiste à inférer les valeurs des coefficients manquants, en se basant sur les données disponibles observées. L'effet de ce traitement est une amélioration du signal, c'est à dire une réduction du bruit et des artefacts d'imagerie. Les applications de la méthode proposée comprennent l'amélioration de la visualisation et le prétraitement pour une analyse ultérieure (par ex. segmentation, classification). Cette méthode est inspirée de [Maggioni et al., 2013], et consiste à ajouter puis à éliminer de façon itérative un bruit artificiel à l'image. Nous ajoutons une étape à cette méthode, de façon à pouvoir l'interpreter dans un cadre de simulation de Monte-Carlo. Cela permet de justifier plus clairement la méthode avec un point de vue de statistique Bayésienne. Par ailleurs, nous améliorons également la vitesse de convergence par rapport à la version originale.

La deuxième contribution vise à localiser les macromolécules dans des cellules intactes et repose sur un apprentissage profond supervisé (*deep learning*). La méthode proposée est multi-classe, c'est à dire qu'elle est capable de localiser plusieurs types de macromolécules en un seul cycle de traitement. Elle est basée sur une segmentation de l'image grâce à un réseau neuronal convolutif 3D d'une part, et, d'autre part, permet de ce fait d'identifer également des structures telles que les membranes cellulaires. Nous ajoutons également une étape de partitionnement de données (*clustering*) afin d'analyser la segmentation. Le but de cette étape est dans un premier temps d'identifier les objets distincts, puis dans un deuxième temps de déterminer lesquels sont associés aux macromolécules d'intérêt. Dans nos expériences, nous avons obtenu un recouvrement significatif avec l'analyse faite par un expert, tandis que le faible temps de traitement (15 minutes par tomogramme) permet de déployer la méthode à une grande échelle. Nous démontrons également comment notre méthode peut être utilisée en complément de protocoles d'analyse déjà existants. Une telle analyse conjointe permet d'obtenir des détections plus complètes, mais aussi d'attribuer un niveau de confiance supplémentaire aux objets détectés.

Dans la dernière partie de la thèse, nous avons exploré différentes pistes pour de futurs travaux. Avec notre deuxième contribution, nous avons démontré l'efficacité d'une analyse basée sur l'apprentissage profond dans un contexte de données de cryo-ET. Cependant, notre méthode est supervisée et nécessite un ensemble d'apprentissage largement annoté. Ces annotations sont coûteuses à obtenir, à la fois en terme de ressources humaines que de temps de calcul. Les nouvelles pistes que nous avons exploré, et pour lesquelles nous présentons des résultats préliminaires, visent à s'affranchir de l'annotation manuelle.

Publications et communications

Posters et présentations orales :

- E. Moebel, C. Kervrann, "Denoising and missing wedge reconstruction in cryo-electron tomography". Poster presented at : *The 16th European Microscopy Congress*, 2016, Lyon
- E. Moebel, C. Kervrann, "Débruitage et correction d'artefacts en cryo-tomographie électronique". Poster presented at : *12ème Journées Imagerie Optique Non Conventionelle*, March 2017, Paris
- E. Moebel, C. Kervrann, "Exploring cellular landscapes with deep learning". Oral presentation at : *Mini-symposium on Bioimage Informatics*, June 2017, Rennes

Publications de conférence :

- E. Moebel, C. Kervrann, "A Monte Carlo framework for denoising and missing wedge reconstruction in cryo-electron tomography", *4th International Workshop on Patch-based Techniques in Medical Imaging (Patch-MI)*, September 2018

Pré-publications :

- E. Moebel, C. Kervrann, "A Monte Carlo framework for noise removal and missing wedge restoration in cryo-electron tomography", pre-print, 2018
- E. Moebel, C. Kervrann, "3D ConvNets improve macromolecule localization in 3D cellular cryo-electron tomograms", pre-print, 2018

Aperçu de la thèse

Ce document de thèse est organisé de manière suivante. Le Chapitre 1 présente le domaine de la tomographie cryo-électronique. Pour commencer, une vue d'ensemble de la modalité d'imagerie est proposée, dans laquelle la préparation de l'échantillon et la reconstruction tomographique sont décrites. Ensuite, l'état de l'art sur les méthodes d'analyse en cryo-ET est décrit.

Dans le Chapitre 2, nous décrivons notre première contribution : un cadre de Monte Carlo dédié à la restauration d'images en cryo-ET. Les problématiques du débruitage et des données manquantes sont discutées. Le cadre est validé à la fois sur des données synthétiques et expérimentales.

Le Chapitre 3 présente l'apprentissage profond (*deep learning*), en commençant par son positionnement dans un contexte plus large de l'apprentissage artificiel (*machine learning*) et en expliquant les succès récents. Les éléments essentiels sont présentés (types de couches, optimiseurs) et les concepts importants sont mis en évidence (e.g. champ réceptif). Ensuite, plusieurs problématiques courantes de la phase d'apprentissage sont discutées, telle que les classes déséquilibrées et le bruit des étiquettes. Enfin, des architectures récentes dédiées à la segmentation d'images sont décrites.

Dans le Chapitre 4, nous présentons une méthode originale de localisation de macromolécules, basée sur l'apprentissage 3D profond. Nous comparons ses performances à l'état de l'art en cryo-ET, à savoir le *template matching*. Les résultats sont présentés sur des données synthétiques et expérimentales. Nous apportons également la preuve que notre méthode a été capable de détecter des attributs d'intérêt biologique, manquées lors de l'analyse par un expert.

Au Chapitre 5, nous proposons de nouvelles pistes pour les travaux futurs. Des méthodes récentes d'apprentissage profond (e.g. réseaux adversaires génératifs, transfert de style) sont adaptées aux données 3D et appliquées à la cryo-ET. Leurs potentiels et leurs avantages sont discutés, en particulier dans le contexte de l'apprentissage non supervisé.

GENERAL INTRODUCTION

Preamble

This thesis has been developed in the frame of a joint collaboration between the Fourmentin-Guilbert Foundation and the Max Planck Institute of Biochemistry (Pr. Baumeister, Martinsried, Germany). First, through the LifeExplorer project, the Fourmentin-Guilbert Foundation has been a pioneer in approaching the structural modeling and visualization of entire cells.

In that context, it is expected from the creation of interactive 3D avatars of cellular environments, bridging from the level of atoms to the level of cells, that the rules governing the spatiotemporal organization of the cytoplasm could be revealed. Such an approach requires to make an inventory and a cartography of all the components constituting a single cell. The technique of choice for such a mapping is cryoelectron microscopy applied on frozen but intact cells. For years, the Fourmentin-Guilbert Foundation has supported the Max Planck Institute of Biochemistry, headed by Wolfgang Baumeister whose team has been capable of delivering whole cryotomograms of E. coli cells at an unprecedented resolution.

The next big step, still an ongoing challenge, was to recognize macromolecular components within the tomograms. Most of the effort of the scientific community was put on the identification of the ribosomes thanks to a methodology relying on single-particle analysis and template matching and giving impressive results. However, it is likely that such an approach will be limited to “big” complexes like the ribosomes. Facing this challenge, the Fourmentin-Guilbert Foundation has solicited the Inria research group Serpico headed by Charles Kervrann to develop alternative recognition methods having the potential to help the in-situ identification of the thousands of proteins left in the dark.

The thesis is born from this initiative. Its goals have been set up this way : develop and compare with well-established methods new approaches based on deep learning and capable of identifying and counting the “gold standard” ribosomes within a tomogram. Theses methods, as an alternative to template matching, should also have the potential to apply on particles smaller and rounder than ribosomes.

Context

The last decades of research in cell biology have revealed that cellular processes are performed by groups of interacting macromolecules in a crowded environment. This is in opposition to previous cell models where macromolecules where considered as isolated objects floating randomly in the cytoplasm. Deciphering the underlying interaction mechanisms is thus of paramount importance to gain a deeper understanding of the cell. To address this issue, cryo-electron tomography (cryo-ET) is a unique imaging technique capable of producing 3D views of large portions of a cell while having enough resolution to localize and identify macro- molecules. Cryo-ET allows avoiding the use of markers, such as fluorescent probes used in light microscopy, which could perturb the cell. Before applying cryo-ET, samples are

first vitrified in order to preserve both the spatial distribution and the structure of macromolecules in the cell during the image acquisition process. This technique enables the study of cells in a close to native state, and has thus the potential to create a molecular atlas from all the detected components (macromolecules, membranes) observed in cryo-tomograms. However, the analysis of such images is challenging due to poor signal-to-noise ratios and imaging artifacts caused by limited-angle tomography. As a consequence, cryo-ET analysis is heavily dependent on computational tools for interpreting the image contents.

Contributions

The aim of this thesis is to propose new computational methods to facilitate the identification of macromolecules in cryo-ET images. In the long term, we hope to propose tools to study the spatial distribution of different macromolecule classes. This would allow a better understanding of the interaction between these classes, for different functional states of the cell.

The first contribution is about enhancing the image interpretability, addressing the issues of low SNR and imaging artifact removal. The origin of these artifacts lies in an incomplete Fourier spectrum ; some Fourier coefficients are not measured during data acquisition. In order to reduce these artifacts, the strategy consists in inferring the values of the missing coefficients, based on the available observed data. The effect of this processing is an improvement in the signal, i.e. a reduction in noise and imaging artifacts. Applications of this iterative and stochastic method include visualization enhancement and pre-processing for further analysis (e.g. segmentation, classification). The method is inspired by [Maggioni et al., 2013], and consists in adding and then iteratively eliminating artificial noise from the image. We introduced an additional step to the method, allowing it to be interpreted in a Monte-Carlo simulation framework. This allows to justify the method more clearly from a Bayesian statistics point of view. In addition, we are also improving the speed of convergence compared to the original version.

The second contribution aims at localizing macromolecules in intact cells, and is based on supervised deep learning (DL). The DL method is multi-class, meaning that it is able to localize several types of macromolecule in a single processing round. The method is based on image segmentation using a 3D convolutional neural network, therefore it also allows structures such as cell membranes to be identified. We also add a clustering step to analyze the segmentation. The purpose of this step is first to identify the distinct objects, then to determine which ones are associated with the macromolecules of interest. In our experiments, we achieved a significant overlap with expert analysis, while the low processing time (15min per tomogram) allows the method to be deployed on a high scale.

In the last part of the thesis, we explored different options for future work. With our second contribution, we demonstrated the effectiveness of a deep learning-based analysis for cryo-ET data. However, our method is supervised and requires a widely annotated training set. These annotations are expensive, both in terms of human resources and computation time. The new approaches we have explored, and for which we present preliminary results, aim to avoid manual annotation.

Publications and communications

Poster and oral presentations :

- E. Moebel, C. Kervrann, "Denoising and missing wedge reconstruction in cryo-electron tomography". Poster presented at : *The 16th European Microscopy Congress*, 2016, Lyon
- E. Moebel, C. Kervrann, "Débruitage et correction d'artefacts en cryo-tomographie électronique". Poster presented at : *12ème Journées Imagerie Optique Non Conventionelle*, March 2017, Paris
- E. Moebel, C. Kervrann, "Exploring cellular landscapes with deep learning". Oral presentation at : *Mini-symposium on Bioimage Informatics*, June 2017, Rennes

Conference paper :

- E. Moebel, C. Kervrann, "A Monte Carlo framework for denoising and missing wedge reconstruction in cryo-electron tomography", *4th International Workshop on Patch-based Techniques in Medical Imaging (Patch-MI)*, September 2018

Pre-print papers :

- E. Moebel, C. Kervrann, "A Monte Carlo framework for denoising and missing wedge reconstruction in cryo-electron tomography", pre-print, 2018
- E. Moebel, C. Kervrann, "3D ConvNets improve macromolecule localization in 3D cellular cryo-electron tomograms", pre-print, 2018

Thesis outline

In Chapter 1, the field of cryo-electron tomography is presented. First an overview on the image modality is provided, in which sample preparation and tomographic reconstruction are described. Then the state-of-the-art on computational analysis methods for cryo-ET data is described.

In Chapter 2, we present a dedicated Monte Carlo framework for image restoration in cryo-ET. The problematics of denoising and missing data (the missing wedge) are discussed, and the framework is validated on both synthetic and experimental data.

In Chapter 3, deep learning is introduced, starting with positioning deep learning in the larger context of machine learning, and explaining the different reasons for its success. Essential building blocks are presented (e.g. layer types, optimizers) and important concepts are pointed out (e.g. receptive field). Then different problems encountered during training are discussed (e.g. unbalanced classes, label noise). Finally, state-of-the-art architectures dedicated to image segmentation are described.

In Chapter 4, we present an original macromolecule localization method, based on 3D deep learning. We compare its performance to current state-of-the-art, namely template matching. Results are presented for both synthetic and experimental data. We also provide evidence that our method found features of biological interest, missed during expert analysis.

In Chapter 5, we propose new leads for future work. Recent deep learning methods (e.g. generative adversarial networks, auto-encoders) are adapted to 3D data and applied to cryo-ET. Potential uses and benefits are discussed, in particular concerning unsupervised training.

PREMIÈRE PARTIE

Image processing for cryo electron tomography

CHAPITRE 1

AN INTRODUCTION TO CRYO-ELECTRON TOMOGRAPHY

1.1 Context and biological stakes

Cryo electron microscopy (cryo-EM) enables to visualize sub-cellular environment at nanometer scale, allowing to identify macromolecular complexes in their native state (see Figure 1.1). The field of view and the resolution are both large enough to enable a joint study of the cellular context and its structure. Therefore cryo-EM acts as a link between low resolution (e.g. light microscopy) and high resolution (e.g. X-ray crystallography) imaging techniques, unifying knowledge retrieved from several scales. For these unique characteristics and resulting new insights in the cell, the main contributors to the development of cryo-EM were recently awarded the 2017 Nobel price in chemistry. Cryo-EM can be used to generate 2D views of samples, however the most interesting feature and the focus of this thesis is the ability to generate 3D images. In this case, we are speaking of cryo-electron tomography (cryo-ET). There is however a hook to these attractive and very useful characteristics ; the noise level in these images is remarkably high, due to the low electron dose used to capture biological specimen. Nonetheless, this limitation can be overcome if identical macromolecules are present multiple times in the image. By combining a large number of these macromolecules, the resulting volume will contain more information than a single image. This strategy is exploited by two cryo-EM techniques, both relying on heavy computational processing : single particle analysis, applied to 2D images ; and subtomogram averaging, applied to 3D images. The major differences between these two techniques are as follows :

— In single particle analysis, a purified sample is being imaged, meaning that the particles of interest are isolated from their native environment. This results into images that are relatively easy to interpret, in the sense that the image contains a large number of homogeneous particles at various orientations and a uniform background. In subtomogram averaging, the sample contains a biological specimen in its native state. Therefore the image contents is quite complex, as the cell cytoplasm is crowded with a multitude of different proteins (e.g. ribosomes, proteasomes, thermosomes...) and structures (e.g. cell membranes, vesicles, microtubules...). It is therefore more difficult to identify the objects of interest.

— For both techniques, the final 3D structure is reconstructed from sub-images, each containing a different particle whose orientation is unknown. In single particle analysis, the sub-images are in 2D, and the difficulty of the reconstruction problem arises from bringing 2D images into register in a 3D space. In sub-tomogram averaging the sub-images are already in 3D (i.e. sub-tomograms),

bringing them into register should therefore be easier, however they have an anisotropic resolution (missing wedge), which greatly complicates the registration task.

Both single particle analysis and subtomogram averaging are being increasingly used in the field of structural biology, whose aim is to determine the structure of proteins in order to decode their functioning. Single particle EM is the most widely used cryo-EM technique, as can be illustrated by the proportion of structures deposited in the Electron Microscopy Data Bank (EMDB) : 76% for single particle, versus 10% for subtomogram averaging. It is however foreseeable that subtomogram averaging will be increasingly used, as relevant structural conformation of proteins may only exist in their native environment. That being said, both techniques benefit from each others advancements in imaging (e.g. phase plates, direct electron detectors...) and computational methods (e.g. template matching, classification...). A developing field that can greatly benefit from advancements in cryo-ET is visual proteomics. As opposed to stuctural biology, whose focus is protein structure, here the aim is to analyze the spatial distribution of various protein families within a cell. The ultimate goal is to build a molecular atlas of the cell [Vendeville et al., 2011], in order to understand mechanisms arising from the interaction between protein groups. Visual proteomics is still in its early stages, with only few works published on that matter [Beck et al., 2009] [Förster et al., 2010]. Cryo-ET is a well adapted imaging modality to this field, because of its ability to capture 3D views of cells in their native state with near to atomic resolution. However, due to low SNR and imaging artifacts, for now only large macromolecular complexes (e.g. ribosomes, proteasomes) and symmetrical specimens (e.g. icosahedral viruses) are identifiable in cryo-ET images. More advances in imaging procedures and in computational methods are necessary to fully benefit of the potential of cryo-ET.

In the remainder of this Chapter, the cryo-ET workflow is described (see Fig. 1.2 for an overview). Section 1.2 outlines the necessary steps to obtain the tomogram, explains the technical difficulties of the imaging modality and presents the sources of noise and artifacts. Section 1.3 provides an overview of existing computational methods to analyze the obtained tomogram, and details the sub-tomogram averaging processing chain.

1.2 Data acquisition in cryo-ET

Data acquisition requires the application of several steps as described in this Section (see Fig. 1.3).

1.2.1 Sample preparation

Freezing The sample is quickly plunged into cryogen (a liquid at $< -160\,$℃), trapping macromolecules in amorphous (or vitrified) ice, which allows to physically immobilize the dynamic machinery of the living cell. Immobilizing the sample is essential for imaging the tilt-series (e.g. 61 images are needed for a tomogram with tilt-range of $\pm 60°$ and a tilt-increment of $2°$), as it avoids blurring due to molecule movement. The rapid freezing (within microseconds) allows to avoid the forming of ice crystals which would perturb the cell structure. Therefore cryo-EM is able to preserve the specimen in a near-native, hydrated environment.

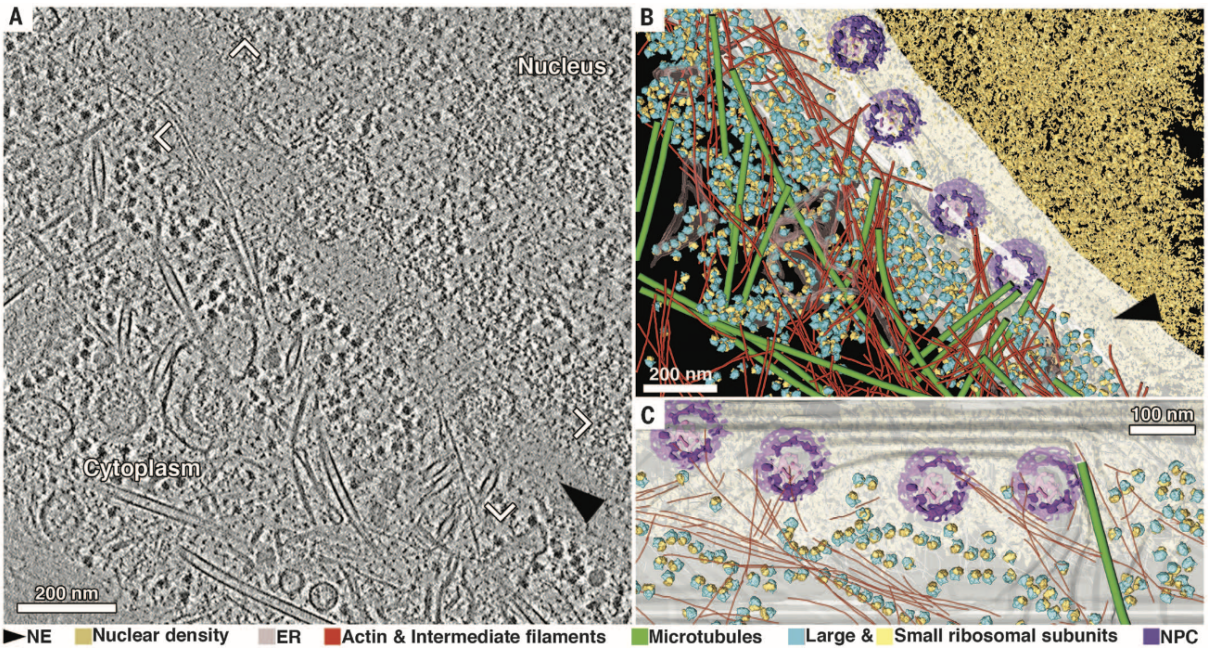

FIGURE 1.1 – Nuclear periphery of a HeLa cell revealed by cryo-ET. (A) Tomogram slice, as obtained after data acquisition and tomographic reconstruction. (B) Annotated view of the tomogram. (C) Cross-section view of the segmentation. NE : nuclear envelope ; ER : endoplasmic reticulum ; NPC : nuclear pore complex. Source : [Mahamid et al., 2016].

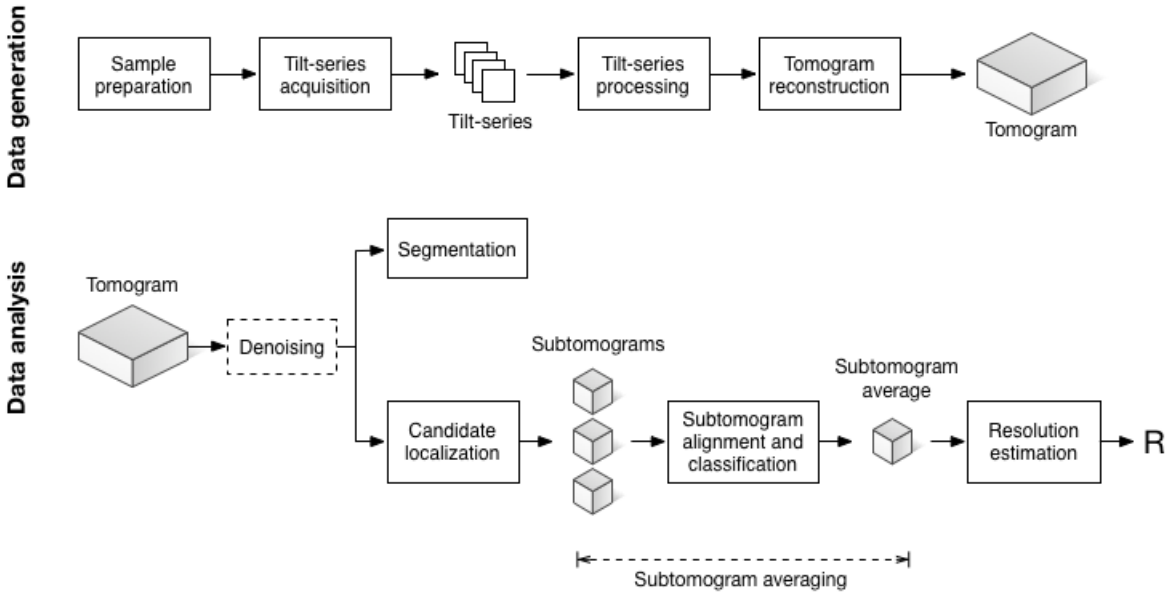

FIGURE 1.2 – Cryo electron tomography workflow.

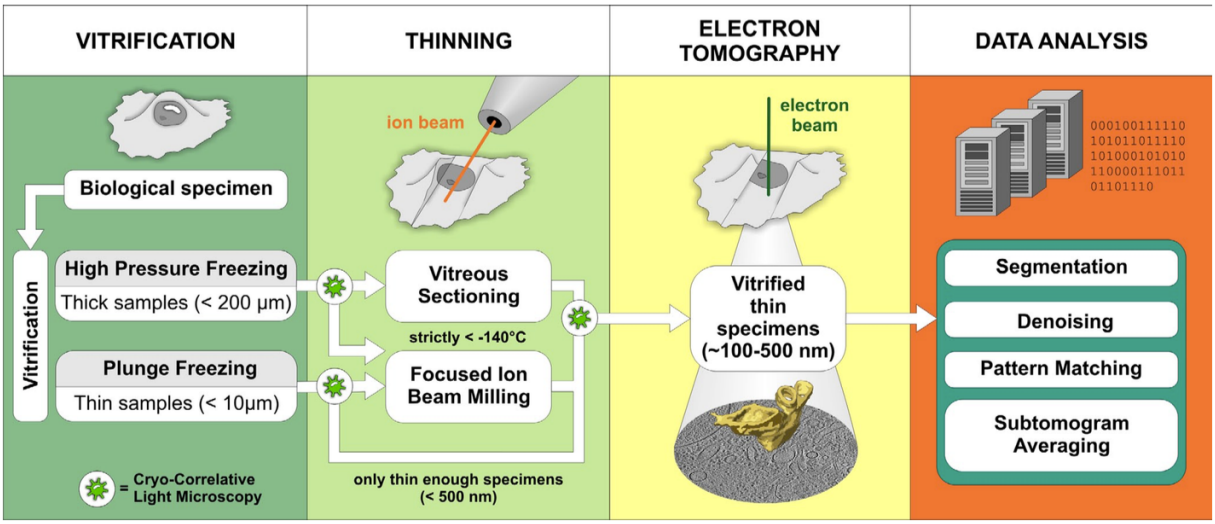

FIGURE 1.3 – Illustration of the data acquisition process. Vitrification (freezing) : samples are commonly vitrified by plunge-freezing into a cryogen at less than -160^oC. Thinning : the sample is usually too thick to allow electrons to pass through, and therefore require to be thinned. Once the sample has been prepared, it is ready for the tomographic imaging and data analysis procedures. Reproduced from [Lučić et al., 2013].

Thinning In cryo-ET, the electronic density of the sample is imaged using the highly coherent electron beam of a transmission electron microscope, operated at intermediate electron accelerating voltages (100-300kV). In this setting, the ideal sample thickness should be less than 500nm, which is usually not the case, therefore the sample needs to be thinned. A recent and efficient method for sample thinning is focused ion beam milling (FIB). A beam of gallium ions is used to cut out sample sections until desired sample thickness is reached. The precision of FIB milling is in the range of tens of nanometers, making it far more accurate than previous techniques like ultramicrotomy, where the sample is mechanically cut with a diamond knife.

1.2.2 Tilt-series acquisition

Tilting During tomographic acquisition a set of projection images, commonly called tilt-series, is acquired while rotating the sample around a single axis (see Figure 1.4). These 2D projections are then combined to produce a 3D image, using a tomographic reconstruction algorithm (see Section 1.2.4). In order to achieve precise rotation, the sample is placed in a computerized sample-holder. Several difficulties arise from this experimental setup. First, due to the slab-geometry of the sample, the apparent thickness of the sample varies with the tilt-angle. The consequence is that projection images at high tilt-angles are noisier, because less electrons are able to pass through the sample. Second, because of the mechanical restrictions of the sample holder, the tilt-range is limited (limited angle tomography) to values ranging most often from $\pm 60°$ to $\pm 70°$ (see Fig. 1.6). Therefore the acquired tilt-series is incomplete, which induces artifacts during tomographic reconstruction. The incomplete angular sampling results into a missing wedge (MW) of information in Fourier domain and geometric distortions in spatial domain. These geometric distortions include an elongation of objects along the missing orientations, as

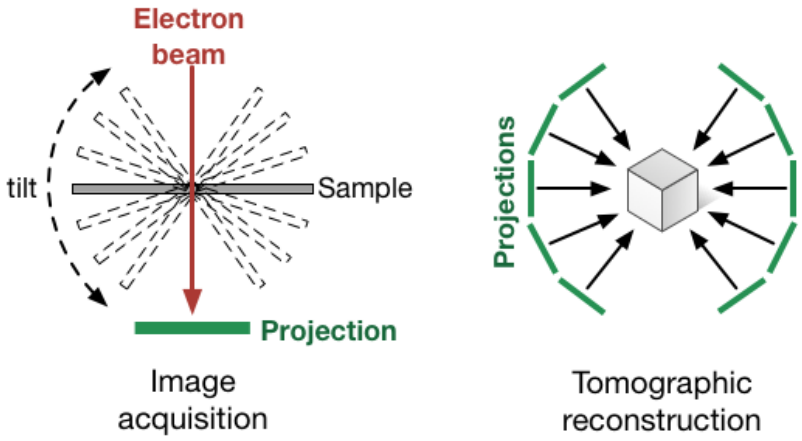

FIGURE 1.4 – Tomographic process : first, projection images are acquired while tilting the sample. These projections are then computationally assembled to produce a 3D image.

well as ray- and side-artifacts emanating from high-contrast objects (see Figure 1.5). Therefore cryo-ET images have an anisotropic resolution. It is possible to reduce the amount of missing information by rotating the sample around a second axis (dual axis tomography), but in practice this is rarely done in cryo-ET because of technical difficulty.

Acquisition Imaging biological material with an electron microscope induces several difficulties. First, image contrast depends on the difference between the atomic density of the biological sample and the embedding medium (vitrified ice). As biological material is only slightly denser than ice, the contrast is low by design. Second, when passing through the sample, the electrons transfer part of their energy to the biological material, damaging the sample. Therefore the cumulative electron dose used to acquire the entire tilt-series needs to be limited, which again considerably restricts the achievable contrast. The SNR of cryo-EM images is therefore very low, and is typically between 0.01 and 0.1, meaning that the noise variance is ten to hundred times larger than the signal variance.

1.2.3 Tilt-series processing

CTF correction The projection images are affected by the contrast transfer function (CTF) of the microscope, modeled by a damped sinusoidal function in the Fourier domain. The CTF depends on the microscope model and the imaging parameters, in particular the defocus value. Without any processing, the resolution of the images are limited to the first zero of the CTF. However, as the CTF is a deterministic perturbation, it is therefore possible to correct the images by using deconvolution techniques (see [Fernandez, 2012] for a review). The correction of the CTF is mandatory if the goal is to retrieve high resolution information as in single particle analysis and subtomogram averaging.

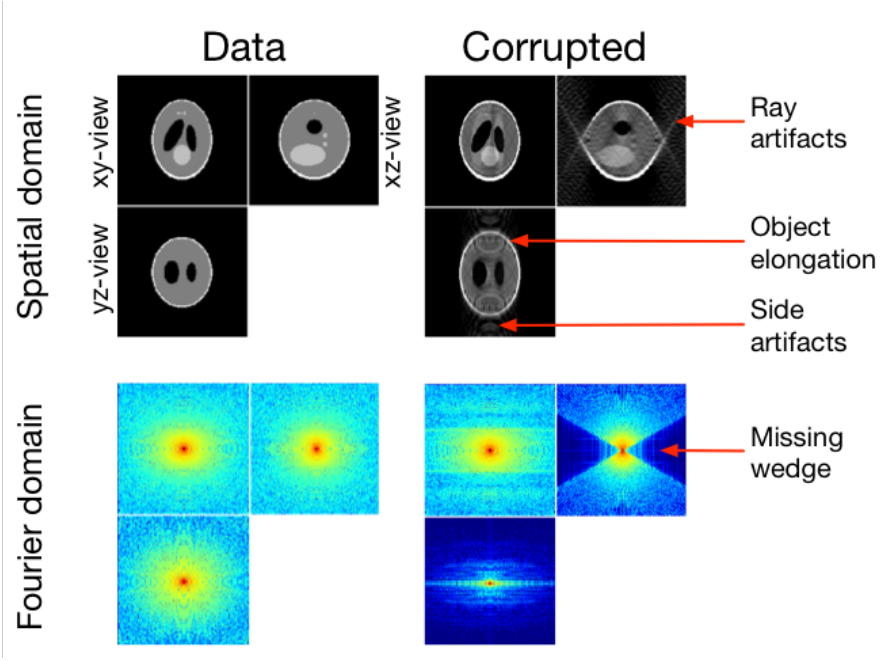

FIGURE 1.5 – The effects of the missing wedge on the Shepp-Logan phantom. On the left the uncorrupted 3D data, displayed as ortho-slices (one central slice along each dimension). On the right the data corrupted by the missing wedge. We represent the data in the spatial domain (top row) and in Fourier domain (bottom row).

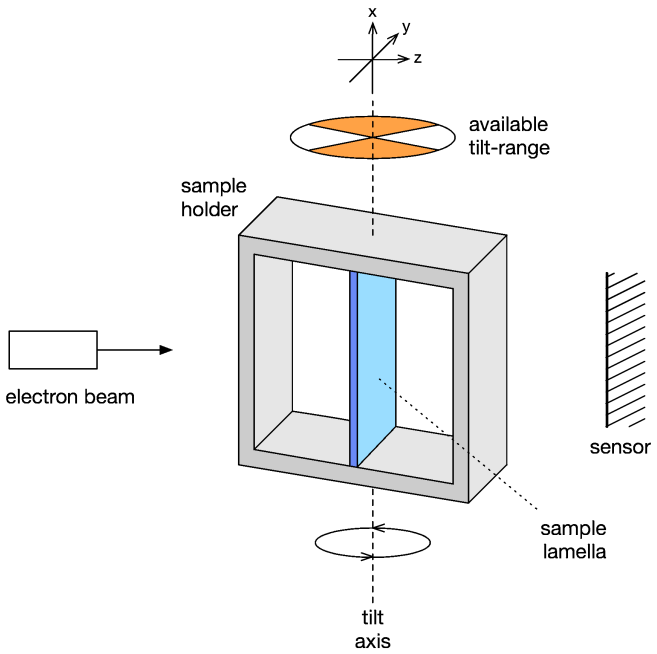

FIGURE 1.6 – Sample holder

Tilt-series alignment This step is intended to mutually set the projections to a common coordinate system, in order to correct for mechanical inaccuracies of the sample holder (e.g. imprecise tilting). Image alignment is based on the identification of common image features, which can be difficult with low contrast cryo-EM images. Therefore, a widespread strategy consists in including high contrast particles (e.g. gold) into the sample. These so-called fiducial markers are easily identified, even in very noisy images, and can thus be used to guide the alignment procedure.

1.2.4 Tomogram reconstruction

The tomographic reconstruction task aims at obtaining a 3D image from the CTF-corrected and aligned tilt-series. The mathematical foundation of this task is based on the Fourier slice theorem (also known as central section theorem), which is defined as follows.

Let $P(\theta, t)$ be the set of projections (i.e. the tilt-series, also called sinogram) of function $f(x, y)$, where θ denotes the tilt angle at which the projection is taken, and t the dimension in which the projection varies (see Fig. 1.7).

The Fourier transform of $P(\theta, t)$ is written as :

$$S(\theta, \omega) = \int P(\theta, t) e^{-i2\pi\omega t} dt.$$

In 1917, Radon showed [Radon, 1986] that this expression is equivalent to :

$$S(\theta, \omega) = \int \int f(x, y) e^{-i2\pi\omega(x\cos\theta + y\sin\theta)} dx dy,$$

which is the 2D Fourier transform $F(u, v)$ of $f(x, y)$ under the constraints $u = \omega\cos\theta$ and $v = \omega\sin\theta$. This relationship is also known as the Fourier slice theorem. The Fourier slice theorem states that a slice through the origin of $F(u, v)$ with angle θ is equal to the 1D Fourier transform $S(\theta, \omega)$ of $P(\theta, t)$. Under this perspective, tomographic reconstruction techniques essentially consist in populating an initially empty 3D Fourier space with the information contained in the projections. In this context some regions in the Fourier space are oversampled (low frequencies) and others undersampled (high frequencies, missing wedge) ; a challenge for reconstruction techniques is therefore how to compensate for this uneven sampling.

Weighted back projection

Weighted back projection (WBP) [Radermacher, 1992] is the simplest reconstruction method, and is commonly used due to its well understood artifacts and computational simplicity. This method essentially applies the Fourier slice theorem, but in the real space. Each projection is "back projected" by evenly distributing its values along its corresponding orientation. The final tomogram is the summation of all the back projections. Due to the different orientations, the distributed values will accumulate at the intersection points. These intersection points correspond to the locations where mass is found in the original sample.

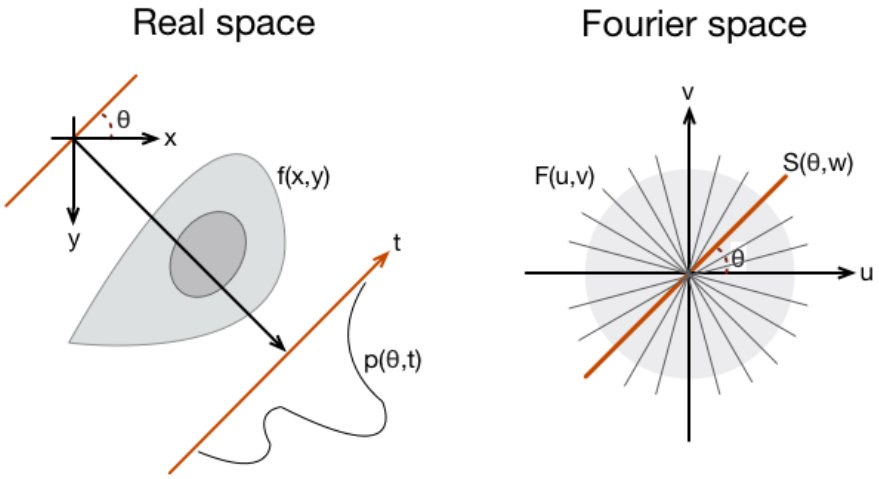

FIGURE 1.7 – Illustration of the Fourier slice theorem in 2D. On the left, a 1D projection $P(\theta, t)$ of a 2D object $f(x, y)$ is taken. The right image shows how the 1D Fourier transform $S(\theta, \omega)$ of the projection corresponds to a central slice in the 2D Fourier transform $F(u, v)$ of the object.

As explained before, the low frequencies are over-represented when applying back projection with no precaution. The resulting tomogram appears therefore blurry. To overcome this difficulty, the solution consists in ponderating the Fourier coefficients according to the amount of redundancy, hence the name "weighted" back projection.

Due to its low computational cost, WBP is very popular and used as a building block for more sophisticated iterative methods.

Iterative reconstruction methods

The common idea behind iterative reconstruction methods is to refine an initial tomogram, usually obtained via WBP, by minimizing the error between the original projections and the projections computed from the reconstructed volume. Several minimization procedures have been proposed to address this issue. Among them, a well established one is SIRT (simultaneous iterative reconstruction technique) [Gilbert, 1972], whose processing steps are as follows, given a set of projections $\{P_k\}$ and reconstructions $\{R_k\}$, with $k = 1...K$ the iterations :

1. Generate the projections P_k from the current reconstruction R_k.
2. Compute the difference D_k between the current projections P_k and original projections P_0 : $D_k = P_0 - D_k$.
3. Back project the difference D_k to get the error volume R_{error}.
4. Substract the error R_{error} from the current reconstruction R_k and repeat the process : $R_{k+1} = R_k - R_{error}$.

Iterative methods have been shown to be more robust to noise and to produce tomograms with enhanced contrast compared to WBP, and are increasingly used in the cryo-ET community [Fernandez,

2012]. However, even though SIRT exists since 1972, microscopists still frequently use WBP [Albert et al., 2017] [Bäuerlein et al., 2017] [Guo et al., 2018]. An explanation is that while with iterative methods the low frequencies are better represented, high frequencies below noise level may be lost. With WBP, even though low frequencies are less well represented and high frequencies are indistinguishable from noise, the high frequencies are still present in the tomogram [Wan and Briggs, 2016]. It is therefore possible to amplify and restore these high frequencies with subtomogram averaging. Other reasons for the wide use of WBP in cryo-ET include its lower computational cost, and the existence of well adapted post-processing tools (e.g. filtering and denoising for contrast enhancement).

1.3 Data processing and analysis in cryo-ET

1.3.1 Image denoising

Due to the low SNR of cryo electron tomograms, denoising is often applied as a pre-processing step to analysis. Denoising consists in applying a filter to reduce noise as much as possible while preserving the structural details of the image.

Denoising algorithms can be categorized in two groups : linear and non-linear methods. In linear denoising, the same filter is applied to all tomogram voxels, as is the case with low-pass filters (e.g. Gaussian filter) and average filters. These filters are efficient for reducing noise in homogeneous regions (e.g. background) but have the disadvantage of blurring structures of interest. Non-linear denoising compensate for this drawback by adapting the filter to the image contents, which makes these methods more efficient in preserving structures. Among non-linear denoising methods, two are are particularly effective in cryo-EM : anisotropic non-linear diffusion (AND) [Frangakis and Hegerl, 2001] and non-local means (NL-means) [Darbon et al., 2008].

The AND [Frangakis and Hegerl, 2001] approach is the most commonly used denoising method in cryo-EM. It is inspired from the physical diffusion process and based on Gaussian filtering, whose property is blurring. However, the Gaussian filter support is modified according to underlying structural image features, which enables to adapt the strength and the direction of the blurring (hence the term "anisotropic"). The nature of local structures is estimated by eigen-analysis, which allows to differentiate homogeneous regions from structures such as lines and planes. As such, AND blurs out homogeneous regions in isotropic manner, however when structure is detected, the blurring is applied along the structure (i.e. anisotropic), which allows to preserve edges and hence image details.

NL-means drew much attention when it was first published, because it outperformed the current the state-of-the-art in 2D image denoising [Buades et al., 2005]. Modern computers and smart coding allow the originally heavy (computationally) method to be used for a variety of applications, including cryo-EM [Darbon et al., 2008]. Unlike the linear average filter, which replaces pixels with the average value of their neighbor pixels, NL-means works patch-wise. The method relies on the self-similarity of image data. It assumes that several similar image patches exist within the image, which turns out to be a valid assumption, especially for small patch sizes. As such, a local patch is replaced by an average of similar patches, which are not necessarily in its direct neighborhood and may be located across the image, hence the term "non-local". For that mather, current state-of-the-art denoising methods [Dabov

et al., 2007] [Lebrun et al., 2013] are based on this non-local strategy (i.e. finding similar patches), but differ in the way the patch values are combined. We find that in particular [Dabov et al., 2007] achieves remarkable denoising performance in cryo-ET.

1.3.2 Segmentation

Segmentation, as it is understood in the field of cryo-ET, consists in attributing a class to each tomogram voxel (i.e. voxelwise classification). Nowadays manual segmentation is still a current method in the field, and is assisted by semi-supervised tools incorporated in dedicated cryo-ET software packages, like SPIDER [Shaikh et al., 2008] and BSOFT [Heymann and Belnap, 2007]. These tools implement basic segmentation algorithms like thresholding and the watershed transform. Although convenient for producing beautiful images for publication, these segmentations suffer from subjectivity, and most importantly are time consuming. Therefore manual segmentation can not be applied on a large scale, as would be necessary to obtain more reliable statistical results.

Numerous automatic and semi-automatic segmentation methods have been developed for cryo-ET images (see [Fernandez, 2012] for an extensive review), but few are actually employed. In general these methods tend to be ad-hoc and specific (i.e. designed for a single type of structure) like [Kervrann et al., 2014a] and [Martinez-Sanchez et al., 2014]. Here meticulous data modeling and parameter optimization allows to achieve accurate segmentation of specific structures. In [Kervrann et al., 2014a], a probabilistic model in the frame of conditional random fields, is used to segment microtubules. The final segmentation is obtained by minimizing an energy functional, using a min-cut/max-flow algorithm. In [Martinez-Sanchez et al., 2014], a tensor voting approach is used to segment membranes (see Figure 1.8), whose profiles are modeled as Gaussian functions. First and second order moments are used to detect edge- and ridge-like structures, these detections are then regularized with tensor voting, a computer vision technique for robust identification of salient features.

Recently, deep learning (DL) approaches have been proposed for tomogram segmentation [Chen et al., 2017] [Zeng et al., 2018], and given the great success of these methods in other application domains, it is very likely that they will become popular tools in cryo-ET. A great advantage is that the same deep neural network can be used to segment any kind of structure in a multi-class framework, as opposed to previous methods [Kervrann et al., 2014a] [Martinez-Sanchez et al., 2014] which are highly case-specific and mono-class. A deep neural network automatically learns a task from data (i.e. training set) ; the larger the data amount, the better it performs. The learning process can be supervised [Chen et al., 2017] or unsupervised [Zeng et al., 2018], each with its own specific characteristics. The drawback of supervised DL is the need for data annotation, the generation of which is time consuming and must be supervised by an human expert. However, this may be preferable to the tuning of several parameters that the user not fully understands. Nonetheless the advantage of supervised training is that a particular object class can be detected with high precision. On the other hand, unsupervised DL does not need annotation ; the network learns to characterize structure in data in a fully automatic manner (using so-called "auto-encoders" [Baldi, 2012]), at the expense of achieving only coarse classification [Zeng et al., 2018] (e.g. membrane-like vs globular-like structures). However unsupervised DL could be used for generic segmentation, where all structures regardless of their type need to be identified for

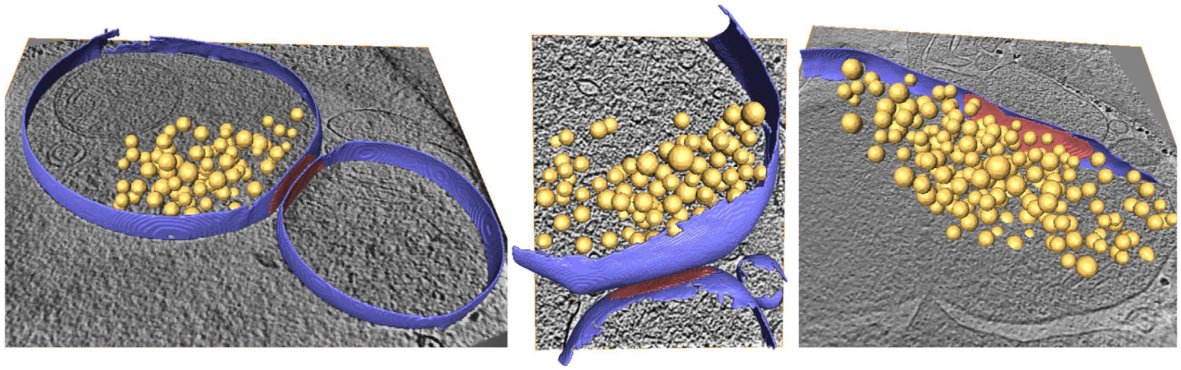

FIGURE 1.8 – 3D membrane segmentation of neuronal synapses. The segmentation tool, dedicated to membranes, is based on tensor voting. Source : [Martinez-Sanchez et al., 2014].

subsequent analysis. In this regard, an interesting non-DL approach for generic segmentation is [Zhou et al., 2018], which generates saliency maps based on unsupervised super-voxel analysis (see Section 1.3.3 for a more detailed description).

Remark : a potential use of supervised DL could be the creation of a meta-segmentation tool. Existing adhoc segmentation tools already work great for particular structures [Kervrann et al., 2014a] [Martinez-Sanchez et al., 2014]. Instead of applying each ad-hoc method individually to a tomogram to detect several classes, one could encode all these tools into a neural network which needs to be applied only once with less computing time.

1.3.3 Macromolecule localization

Template matching

Principle This technique is based on a template, a 3D image much smaller than the tomogram and containing a model of the object of interest. The template is used to explore the tomogram, to find occurrences of the target object. To this end, the template is slided along each tomogram position in a convolutive manner. At each position, a similarity score is computed between the template and the overlapped tomogram voxels. This operation results into a 3D score-map, where local maxima indicate positions of candidate objects (see Figure 1.9). Since the rotational orientation of the objects are unknown, all possible template orientations have to be considered. In the end, the outputs of the TM procedure are the positions and the orientations of candidate objects. However, this operation is very expensive computationally (20 to 30 hours on a 32 core CPU cluster), as the score has to be computed for all possible tomogram positions and template orientations. Therefore it is common to limit the search to a subregion of the tomogram, to subsample the tomogram, and to use coarse rotation increments (around $5°$ to $10°$). Consequently the positions and orientations have limited accuracy, and are usually refined using alignment procedures (see Section 1.3.4).

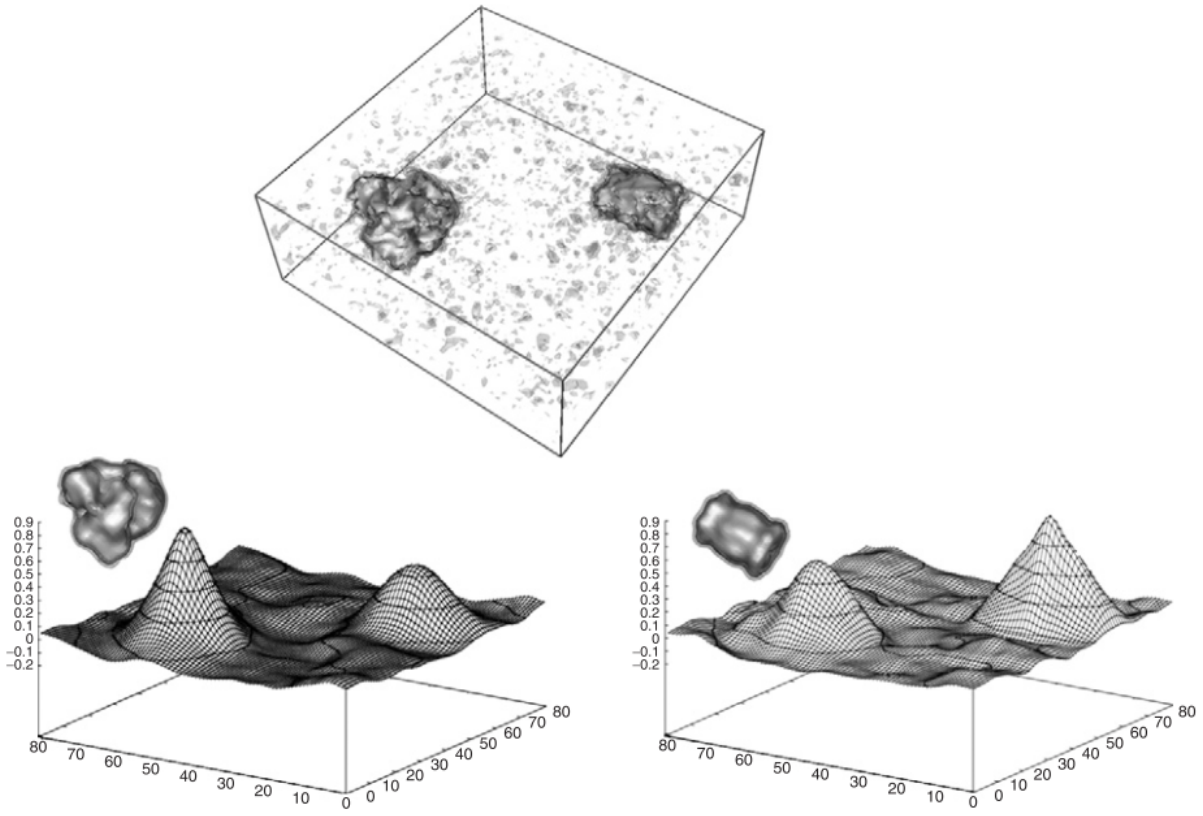

FIGURE 1.9 – Illustration of the template matching procedure. The top image represents a sample tomogram, containing two different macromolecule classes in a noisy environment. Templates for each class were used to explore the tomogram, resulting into the two scoremaps (using the cross correlation coefficient), represented as surface plots at the bottom of the figure. A local maxima indicates the potential presence of a target object. While local maxima are observed for both objects with either template, usually the score value is higher when the correct template is used. Reproduced from [Best et al., 2007].

Template generation Templates are generated from atomic-scale models of macromolecule complexes, stored in the protein data bank (PDB). These models are derived from various imaging techniques like x-ray or electron crystallography and single particle EM, and saved as PDB files. A PDB file contains the coordinates of individual atoms of the macromolecule complex. In order to transform the PDB file into a 3D image, the atoms are placed onto a grid (the image voxels) and their atomic numbers Z are summed in each grid element, resulting into a electron density map of the macromolecule. The values of the density map are proportional to the tomogram voxel values and can therefore be used as a template. Finally, the template needs to be scaled to the tomogram resolution and filtered by the tomogram CTF in order to account for the imaging process.

Alternatively, the template can be generated by hand-picking objects of interests from the tomograms and applying subtomogram averaging (see Section 1.3.4) to them. This approach is useful when a model is not available in the PDB database.

Score The similarity score used for TM in cryo-ET is the normalized cross-correlation coefficient, as defined in Equation (1.2). This score produces high values at positions of targeted macromolecules, however also at the positions of any structure with similar contrast. TM produces therefore a high number of false positives, and experts often need to refine their results using subtomogram classification (see Section 1.3.4).

A non-trivial task with TM is choosing a score threshold to select macromolecule candidates. Several strategies can be applied to that end : i/ the score distribution of local maxima should display a mode associated to high score values (see Fig. 1.10 (a)). This mode corresponds to true positives, the more the mode is pronounced, the more the score values are discriminative. Fitting a Gaussian function to this mode allows to estimate the number of true positives and therefore to choose a score threshold [Ortiz et al., 2006]. ii/ This mode is not always observable with low SNR data. Comparison with score values obtained with a mirrored template provides additional information (provided the target object has no symmetry). All hits with the mirrored template are false positives, as the contained structure has a non-native handedness. When both score distributions are compared (native and non-native handedness), a shift appears (see Fig. 1.10 (b)). The right intersection between the distributions indicates from which point the score values decline into noise correlation, and a threshold can be chosen accordingly [Förster et al., 2010]. iii/ Another possibility is to use several templates containing different subunits of the macromolecule and to colocalize the hits of obtained score-maps [Yu and Frangakis, 2011]. However the target macromolecule needs to be big enough to be able to use its separate subunits as templates.

Alternative localization methods

A number of alternative methods to TM have been published for cryo-ET, and all share the same strategy :

1. Use of a relatively simple method to localize a large number of candidates. The localization performance has therefore a high recall, but very low precision (lower than TM for instance).
2. Use of a supervised or unsupervised classification technique to filter out false positives, therefore increasing the precision.

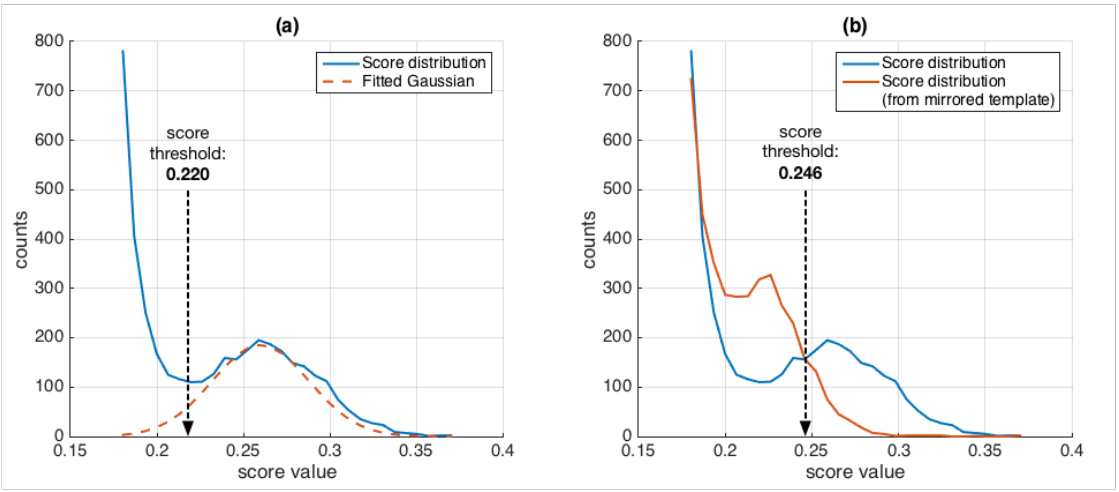

FIGURE 1.10 – Template matching (TM) score threshold. The threshold can be determined by analyzing the distribution of the local maxima of the score-map. Then, the number of true positives can be determined by either (a) fitting a Gaussian or (b) by comparing to the distribution obtained using a mirrored template.

In [Xu et al., 2011], the candidate objects are segmented using the watershed method [Beucher and Meyer, 1993]. Then each segmented object is categorized in unsupervised manner using a two stage classification approach. In the first classification stage, intended to perform a coarse object classification, the objects are clustered according to a rotation invariant feature vector, based on spherical harmonics decomposition. In the second stage, the classification is refined using a Gaussian hidden Markov random field model. This work constitutes a proof-of-principle and has only been evaluated on synthetic data.

In [Chen et al., 2014a], candidate objects are segmented using anisotropic diffusion and applying a threshold on the filtered voxel values. Similar to [Xu et al., 2011], a two stage classification step is applied : first, the negative class is filtered out by clustering (DBSCAN) the objects, using intensity histograms as features. Then, the classification is refined in a supervised manner using a sphere ring Haar descriptor as feature, and random forests as a classifier. The method is evaluated on both synthetic and experimental data, and the authors demonstrate a superior performance to TM.

In [Pei et al., 2016], the authors applied the difference-of-Gaussian (DOG) approach, commonly used in single particle cryo-EM [Voss et al., 2009]. This approach simply consists in computing the difference between two Gaussian filtered versions of the tomogram. This operation results in a DOG map, where the value peaks represent the candidate objects. The method localizes objects of a certain size, depending on the variance values chosen for the Gaussian filters. Consequently, the authors combine several DOG maps to select candidate objects of variable sizes. This work does not include a classification method to discriminate the detected objects into several categories.

Even though these methods are relevant alternatives to TM, they are not referenced in papers describing experiments in cryo-ET. This suggests that TM is still the most widely used localization method in cryo-ET.

A promising recent work having a direct application for macromolecule localization is described in

[Zhou et al., 2018]. The authors present a method for computing tomogram saliency maps. In this framework, saliency is defined as the likelihood that an image subregion stands out relative to its background. The saliency map is generated by first aggregating the tomogram voxels into super voxels (i.e. clustering). Then, the saliency of the super-voxels is obtained as follows : i/ a feature extraction is performed for each super-voxel, the feature vector consists in a combination of an intensity histogram and Gabor features ; ii/ principal component analysis (PCA) is applied on the feature vectors to obtain sparse vectors (so-called "saliency vectors"). The saliency value is then obtained by taking the mean of these sparse vectors. This work shows encouraging results on experimental data, and image regions with high saliency could be used as object candidates. Therefore the perspective of combination with unsupervised classification methods seems promising.

1.3.4 Subtomogram averaging

Subtomogram averaging is a processing chain whose objective is to combine several, noisy instances of a targeted macromolecule, in order to obtain a high-resolution density map. This processing chain is composed of two main tasks : alignment and classification. Subtomogram alignment is necessary to bring the instances into register prior to averaging. Subtomogram classification is needed for filtering out false positives and identifying target classes/sub-classes among the subtomogram set, as obtained by the localization procedure. These tasks can be performed either separately or jointly, depending on used methods. Existing methods heavily rely on a dedicated similarity score, the constrained cross-correlation coefficient (CCC), defined as follows.

Constrained cross-correlation coefficient (CCC) The correlation coefficient (CC) between volumes V_i and V_j is defined as follows :

$$CC(V_i, V_j) = \sum_{x,y,z} V_i(x,y,z) \times V_j(x,y,z),$$

where (x, y, z) denote the spatial coordinates of each voxel.

In order to adjust for contrast, the volume values need to be normalized by subtracting the mean $\bar{V}_i$ and dividing by the variance :

$$\tilde{V}_i(x,y,z) = \frac{V_i(x,y,z) - \bar{V}_i}{\sqrt{\sum_{x,y,z}(V_i(x,y,z) - \bar{V}_i)^2}}.$$

In order to account for the missing wedge (MW) in Fourier space, the CC computation is constrained to sampled Fourier coefficients (see Figure 1.11). For this purpose, let us define W_i as a binary mask in Fourier domain, having values of 1 for sampled Fourier coefficients and values of 0 otherwise (i.e. belonging to the MW). Now, it is common that the MW orientation differs between two volumes. Therefore, it is necessary to constrain the CC computation to the region Ω_{ij} not affected by the MW in both volumes. Finally, the intersection between W_i and W_j is defined as :

$$\Omega_{ij} = W_i \times W_j.$$

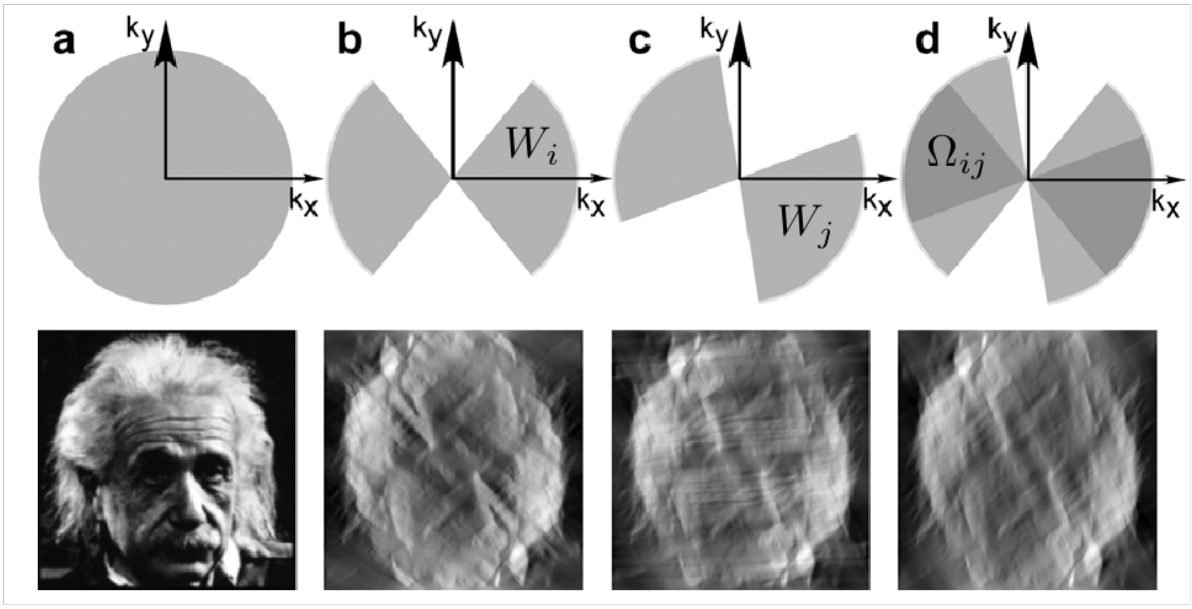

FIGURE 1.11 – Illustration of the constrained correlation measure in 2D. (a) Without a missing wedge, the Fourier spectrum is fully sampled (top row) and the image has an isotropic resolution (bottom row). (b) The image i suffers from a missing wedge : the Fourier spectrum is only partially sampled, resulting into strong deformations in real space. (c) Same as (b), however the missing wedge of image j has a different orientation. (d) The CCC measures correlation only over the Fourier region Ω_{ij} that is sampled in both images. The bottom image is the real space representation of Ω_{ij}. Reproduced from [Förster et al., 2008].

Also, in order to reduce the contribution of background noise to the CC measure, another binary mask M (e.g. spherical) is applied to the data, but in the spatial domain. The constrained volume V_i becomes :

$$V_i' = M \times \mathrm{FT}^{-1}[\Omega_{ij} \times \mathrm{FT}[\tilde{V}_i]], \tag{1.1}$$

where FT$[.]$ denotes the Fourier transform and FT$^{-1}[.]$ its inverse transform.

In the end, the constrained correlation coefficient (CCC) between volumes V_i and V_j is defined as follows :

$$CCC(V_i', V_j') = \sum_{x,y,z} V_i'(x,y,z) \times V_j'(x,y,z) \tag{1.2}$$

Subtomogram alignment

Prior to being averaged, the subtomograms need to be aligned so that the macromolecules they contain are centered and have the same orientation (see Figure 1.12). Aligning two subtomograms involves 6 parameters : i/ the translation (t_x, t_y, t_z) in each direction for centering the particle, needed to take into account the errors from the localization procedure ; ii/ the Euler angles (ϕ, ψ, θ) for characterizing the 3D rotation. This task is usually solved by maximizing the CCC (see Equation (1.2)) for all possible combinations of $(t_x, t_y, t_z, \phi, \psi, \theta)$, which involves very time consuming computations. It can be formulated

as :

$$(\hat{t_x}, \hat{t_y}, \hat{t_z}, \hat{\phi}, \hat{\psi}, \hat{\theta}) = \text{argmax}_{t_x,t_y,t_z,\phi,\psi,\theta} CCC(V_1, T_{t_x,t_y,t_z} R_{\phi,\psi,\theta} V_2),$$

where T_{t_x,t_y,t_z} is the translation operator on volume V_i and $R_{\phi,\psi,\theta}$ the rotation operator.

An exhaustive search of this 6D space is very computationally demanding if the translations and rotations are performed in real space. However, it is possible to accelerate computation for the translational search if the CCC is computed in the Fourier domain [Walz et al., 1997], by using the property that the amplitude of the Fourier domain is translation invariant. This allows to find the best translation without explicitly computing all possible configurations. This technique is known as fast translational matching (FTM). In a similar manner, the rotational search can be accelerated using the spherical Fourier transform, whose amplitude is rotation invariant. Likewise, this technique is called fast rotational matching (FRM) and has been applied in several subtomogram alignment algorithms [Bartesaghi et al., 2008] [Xu et al., 2012] [Chen et al., 2013]. Among these algorithms, [Chen et al., 2013] achieves the best performances in terms of attained resolution.

After determining the optimal alignment parameters for each subtomogram, the subtomogram average is computed by averaging the aligned subtomograms. Since the aligned subtomograms are affected by distinct missing wedges, the average has to be properly weighted to achieve uniform data distribution in the Fourier space. To do so a weighting mask is applied in the Fourier space, and is obtained by summing all the rotated wedge masks W_i.

Usually, all subtomograms are aligned to an external reference, often derived from other imaging modalities (e.g. x-ray crystallography). This way of proceeding yields two problems : i/ an external reference is not always available ; ii/ the use of an external reference might bias the alignment procedure. A typical example of reference bias is presented in [Henderson, 2013], in which an alignment procedure is applied to noise images, using an Einstein portrait as a reference. The authors show that when averaging a large enough stack of aligned noise image, the face of Einstein emerges from the noise, although the initial reference was not used for computing the average (see Figure 1.13). This poses a serious problem when the objective is to determine the structure of a macromolecule.

A possible method to alleviate the external reference bias is presented in [Chen et al., 2013]. The idea is to use as an initial reference the average of the subtomograms in random orientations, which corresponds to a "blob"-like structure (see Figure 1.14). After aligning the subtomograms to the initial reference, an updated subtomogram average is computed with obtained orientations and translations. Repeating the alignment procedure using the precedent average as a reference allows to iteratively refine the average, until converging to a high resolution macromolecule structure. Note that this method needs to repeat the whole alignment procedure for each iteration, which has only been made possible through the development of the fast matching algorithms. This method is called "reference free" alignment, although in essence the procedure still uses a reference. Rather, the name refers to the fact that no external reference is used. Fast matching algorithms are also used as part of joined alignment and classification procedures [Xu et al., 2012] [Chen et al., 2014b] (see Section 1.3.4), in which "reference-free" alignment strategies are also presented.

Alternative methods using a Bayesian framework have been developed for joined subtomogram alignment and classification [Scheres et al., 2009] [Stölken et al., 2011] and are described in Section

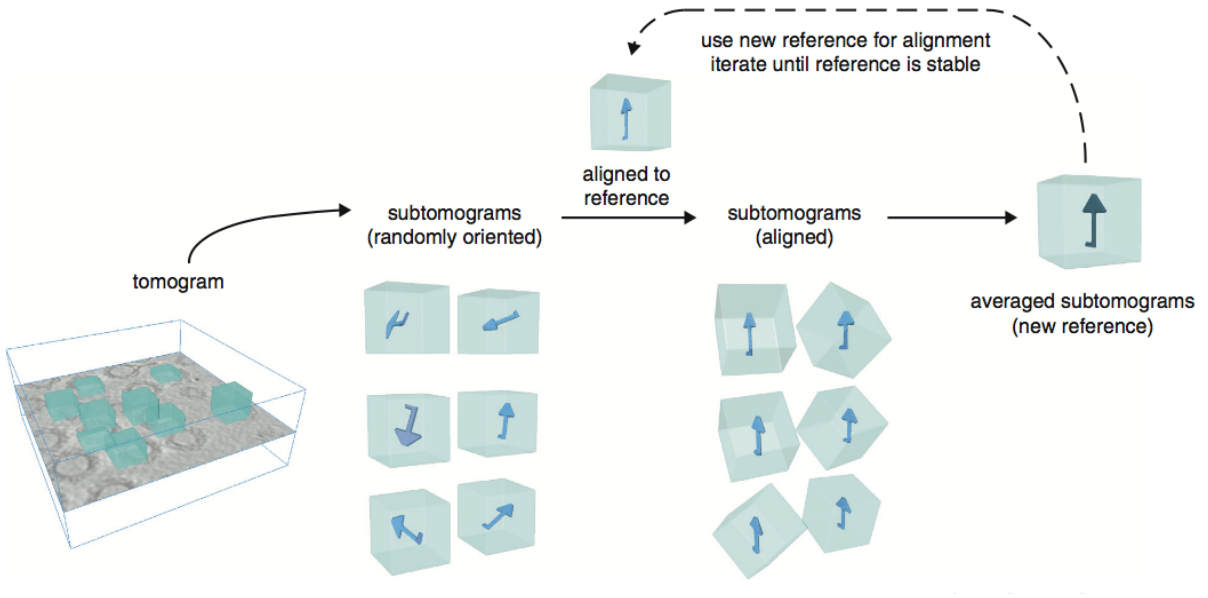

FIGURE 1.12 – Overview of the subtomogram averaging workflow. First, subtomograms are extracted from the full tomogram, each subtomogram contains a randomly oriented instance of the macromolecule of interest. Then, they are rotationally and translationnaly aligned to a common reference. The aligned subtomograms are averaged to produce a new reference. Finally, the process is iterated until aligned subtomograms converge to a stable reference. Reproduced from [Briggs, 2013].

1.3.4. It should be mentioned that "reference free" alignment is also possible with these methods, when initializing the algorithm in a similar manner than in [Chen et al., 2013].

Subtomogram classification

As for segmentation (see Section 1.3.2), classification techniques can be grouped in two categories : unsupervised and supervised techniques. Each category has certain advantages depending on the problem. Unsupervised classification is deeply needed for the identification of unknown macromolecules (or unknown structural conformations of known macromolecules). On the other hand, supervised classification is often more precise and is the ideal tool to identify known macromolecules. This is useful when a high number of macromolecule instances is needed (e.g. for subtomogram averaging) or when their spatial distribution is studied in unknown contexts (e.g. specific cell regions, specific functional states). As a counterpart, the expert has to provide manually selected (assisted by semi-automatic tools) examples of the object of interest.

Unsupervised approach For now, most effort has been put in the development of unsupervised subtomogram classification methods. All these methods have in common some kind of clustering based on an adapted similarity measure. The output is a representative (subtomogram average) for each cluster (i.e. class) ; the expert then analyzes these representatives for structural differences.

In [Förster and Hegerl, 2007], the authors proposed two non-supervised classification techniques,

FIGURE 1.13 – Illustration of the reference bias occuring with subtomogram alignment. This image is obtained after aligning 1000 images of pure white noise to a portrait of Einstein. The aligned images are then averaged : Einstein emerges from the noise, even though the portait itself has not been used for computing the average (only for alignment). Reproduced from [Shatsky et al., 2009] [Henderson, 2013].

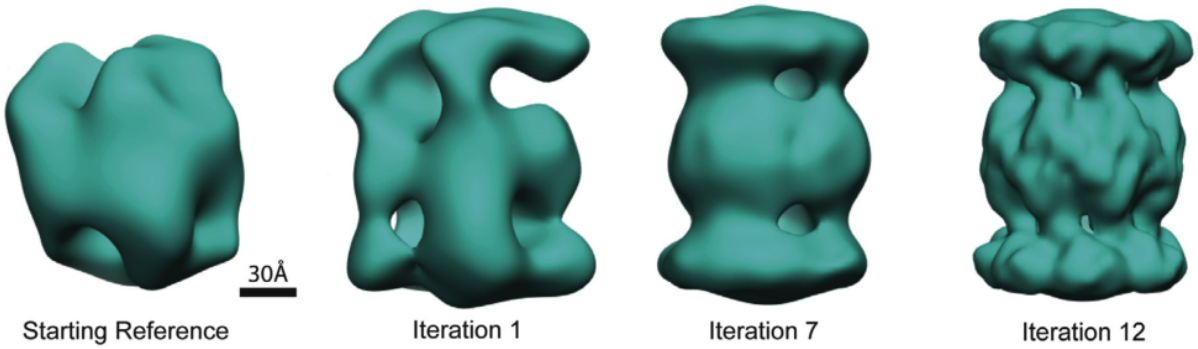

FIGURE 1.14 – Reference-free subtomogram alignment of a 20S proteasome. The procedure starts from a blob-like structure (obtained by averaging randomly rotated subtomograms) and converges to a high-resolution subtomogram average (~ 15Å). Reproduced from [Chen et al., 2013].

both based on a pairwise similarity matrix, making use of the CCC (see Eq. 1.2) as a similarity measure. The first technique consists in hierarchical clustering, for which poor performance is reported for experimental data. Therefore a more succesfull alternative is proposed, based on principal component analysis (PCA). The originality of their approach is that instead of applying the PCA on the covariance matrix, as is normally done, they apply PCA on the CCC similarity matrix (hence their method name, CPCA, for constrained PCA). This results in a set of eigenvectors, allowing a more robust representation of the data, as well as a reduced dimensionality. In the end, k-means clustering is applied to obtain the final classification.

In [Yu and Frangakis, 2011], a non-supervised classification method using self-organizing maps (SOM) is proposed, called KDSOM3D (kernel density estimator SOM). As a motivation for their work, the authors warn against using the same similarity criterion for localization and classification, as is suggested in [Förster and Hegerl, 2007] where CCC is used both in TM and CPCA. SOM is a clustering technique based on a neural network. It can be thought of being similar to k-means, in the sense that the user has to specify the number of input classes, and that a cluster representatives (called centroids in k-means and "code-vectors" for SOMs) are iteratively shifted to fit the data distribution. Data points are then classified according to the nearest code-vector. The method uses the euclidean distance, after normalizing and constraining the data as described in Equation (1.1) (i.e. "constrained L_2-norm"). The authors compare KDSOM3D with CPCA and MLTOMO (described later on) on both synthetic and experimental data. It appears that KDSOM3D is more robust to SNR and tilt-range, even though CPCA is not far behind.

In [Förster and Hegerl, 2007] and [Yu and Frangakis, 2011], subtomograms are assumed to be aligned. A limitation of these approaches is that their classification performances are strongly dependent on the quality of the alignment. Alignment errors are indeed likely to occur, because of the low SNR and the presence of structures neighboring the molecules (crowded cell environment). In order to overcome this limitation, a strategy consists in repeating the alignment and classification steps iteratively to obtain finer results, as opposed to applying each step only once as in [Förster and Hegerl, 2007] and [Yu and Frangakis, 2011]. This strategy is exploited in [Chen et al., 2014b] [Scheres et al., 2009] [Stölken et al., 2011] ; these methods propose an unified framework for subtomogram alignment and classification and are describred in the following paragraphs.

In [Chen et al., 2014b], the authors proposed a dedicated k-means algorithm, AC3D. Here, the cluster centroïds are subtomogram averages. At each iteration, the subtomograms are assigned to the closest centroid using an original similarity measure, aligned [Chen et al., 2013] and finally the subtomogram averages are updated. The proposed similarity measure, which contributes greatly to the success of AC3D, is nothing else than the CCC coefficient (Equation (1.2)) with an original way of computing the spatial mask (see M in Equation (1.1)). The spatial mask is generated automatically to focus the similarity measure where significant differences between class averages are located, which allows to capture subtle class distinctions. The method has superior performance to CPCA and MLTOMO on both synthetic and experimental data.

The above mentioned methods [Förster and Hegerl, 2007] [Yu and Frangakis, 2011] [Chen et al., 2014b] have in common the use of a clustering algorithm and a MW compensated metric. However, a disadvantage of these methods is the use of ad-hoc parameters which require careful tuning from an

expert user, possibly introducing subjectivity. In [Scheres et al., 2009] [Stölken et al., 2011], a Bayesian approach for subtomogram alignment and classification is proposed. The idea is to provide an unified and well understood statistical framework, where most parameters are learned from the data itself, allowing a more objective data analysis. A Bayesian framework consists in formulating a probabilistic data model, and fitting the model parameters to the data. The probabilistic nature of the model allows to obtain error estimates for fitted parameters. In [Stölken et al., 2011], both alignment parameters and class labels are treated as hidden variables, and expectation-maximization is used to maximize a single likelihood function. The method, called MLTOMO (maximum likelihood tomography), has been successfully applied to synthetic and experimental data. Although this approach accumulates many important advantages, like the guaranty of an unbiased optimal result (assuming enough data is available), error estimation and solving of alignment and classification in an unified framework, MLTOMO has been shown to achieve poor classification performance compared to KDSOM3D [Yu and Frangakis, 2011] and AC3D [Chen et al., 2014b], on both synthetic and experimental data. Therefore, despite its interesting statistical properties, it appears that carefully tuned ad-hoc tools still perform better. However, it is worth noting that ML is widely used in single-particle cryo-EM and implemented in a framework called RELION [Scheres, 2012]. The high popularity of RELION in single particle is most likely due to the fact that in single particle the imaged samples are simpler (i.e. homogeneous background) than in cryo-ET (i.e. crowded cell) and therefore ML approaches perform better.

Supervised approach Supervised subtomogram classification methods are for now not very common, and are still in their prototyping phase. The concept has already been demonstrated with experimental data using classification algorithms like support vector machines [Chen et al., 2012] as well as random forests [Chen et al., 2014a]. Both [Chen et al., 2012] and [Chen et al., 2014a] use rotation invariant feature vectors, and have been shown to improve class purity in a candidate subtomogram set acquired with TM. However, up to now nobody seems to use these methods, since to date no experimental cryo-ET paper cites them.

That being said, with the great success of deep learning (DL) in other imaging domains, especially for image classification [Krizhevsky et al., 2012], we anticipate an undoubted success of these methods in subtomogram classification and more generally in cryo-ET. Especially in the present context, DL is advantageous because it does not rely on a pre-defined similarity measure. All above cited unsupervised classification methods rely on the CCC coefficient (see Eq. 1.2) or some kind of variation. While the CCC proves to be usefull for several applications, this score may not always be selective enough, as can be illustrated by the high false positive rate of TM. One should not forget that a metric is a hand-crafted operation, it should be considered as an indicator (i.e. it correlates with desired features) rather than a measure of truth. DL makes it possible to get rid of the need of a pre-defined similarity measure. Instead, it learns automatically its own criteria to discriminate data features from the data itself, driven by the annotations in the training set. This allows to achieve much more discriminating power than the use of handcrafted criteria, as can be illustrated by the first DL breakthrough [Krizhevsky et al., 2012] where a convolutional neural network considerably outperformed the current state-of-the-art in image classification at that time. Investigation in that direction has already been published and demonstrates the feasibility of DL for supervised subtomogram classification on experimental data, with encouraging

recent results [Xu et al., 2017] [Che et al., 2018].

1.3.5 Quality assessment

Once the final subtomogram average is obtained, its quality is evaluated by estimating resolution. In cryo-ET, the resolution is defined as the highest frequency for which signal is discernible from noise. Without any doubt, the standard quality measure in cryo-ET today is the Fourier shell correlation (FSC) criterion.

The FSC measures the normalised cross-correlation coefficient between two volumes V_1 and V_2 as a function of frequency. The Fourier coefficients yielding the same frequency are distributed along a shell in Fourier domain, hence the name of this measure. If $F_i = \mathrm{TF}[V_i]$, the FSC is defined as :

$$\mathrm{FSC}(r_n) = \frac{\sum_{r \in r_n} F_1(r) F_2(r)^*}{\sqrt{\sum_{r \in r_n} F_1^2(r) \sum_{r \in r_n} F_2^2(r)}} \tag{1.3}$$

where r_n is a Fourier shell and "*" denotes the complex conjugation operator.

The FSC curve has generally a value close to one at low frequencies, and decreases as frequencies increase (see Fig. 1.15). At high frequencies, the correlation values originate from noise rather than signal, which is illustrated by the random fluctuations of the curve. The frequency at which the values stop originating from the signal defines the resolution. Therefore, in order to estimate the resolution of a structure, one has to assess the noise level in the image. For this purpose, several FSC threshold criteria exist in the literature : the fixed threshold values "0.5" and "0.143", and the "$\sigma-$factor" and "bit-based" threshold curves (see Fig. 1.15). The choice of the best criteria is heavily discussed [Van Heel and Schatz, 2005], and it is therefore common in structural biology papers to report several resolution values using different criteria.

The FSC measures the similarity between two 3D images. Since in structural biology the challenge is to discover new structures, there is no ground truth an therefore nothing to compare the obtained structure (e.g. subtomogram average) to. It is hence common to compute two subtomogram averages by randomly splitting the subtomogram set into two subsets. Measuring the FSC between these two averages is referred to as "gold standard" FSC.

The FSC attributes a single resolution value to an image. However, resolution may vary across the image, as is the case in particular locations where the imaged structure exhibits structural variability (i.e. blurry average, hence lower resolution). It is therefore useful to be able to determine the local resolution in the form of a resolution map. A noteworthy work for this task is [Kucukelbir et al., 2014], where a likelihood-ratio test is performed at each image location. This test measures if a local sinusoid of a certain frequency is statistically detectable above noise. The local resolution is then defined as the highest frequency at which the test passes at a given p-value.

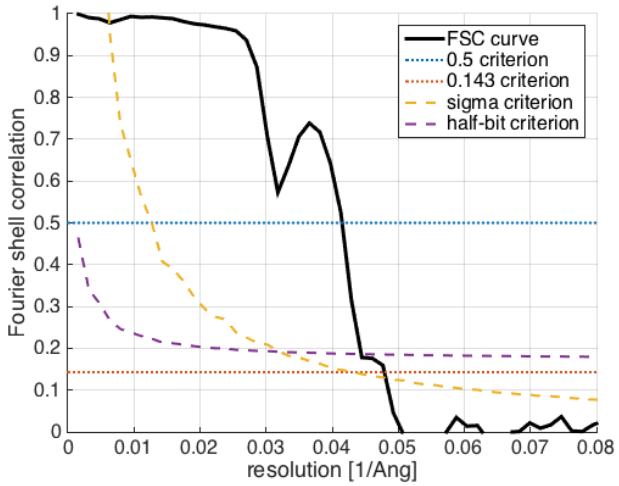

FIGURE 1.15 – Fourier shell correlation criteria. The intersection between the FSC curve and the criteria provides an estimation of obtained resolution.

CHAPITRE 2

A MONTE CARLO FRAMEWORK FOR DENOISING AND MISSING WEDGE RECONSTRUCTION IN CRYO-ELECTRON TOMOGRAPHY

We propose a statistical method to address an important issue in cryo-electron tomography image analysis : reduction of a high amount of noise and artifacts due to the presence of a missing wedge (MW) in the spectral domain. The method takes as an input a 3D tomogram derived from limited-angle tomography, and gives as an output a 3D denoised and artifact compensated volume. The artifact compensation is achieved by filling up the MW with meaningful information. To address this inverse problem, we compute a Minimum Mean Square Error (MMSE) estimator of the uncorrupted image. The underlying high-dimensional integral is computed by applying a dedicated Markov Chain Monte-Carlo (MCMC) sampling procedure based on the Metropolis-Hasting (MH) algorithm. The proposed computational method can be used to enhance visualization or as a pre-processing step for image analysis, including segmentation and classification of macromolecules. Results are presented for both synthetic data and real 3D cryo-electron images.

2.1 Introduction

Cryo-electron tomography (cryo-ET) is generally used to explore the structure of an entire cell and constitutes a rapidly growing field in biology. The particularity of cryo-ET is that it is able to produce three-dimensional views of vitrified samples at sub-nanometer resolution, which allows observing the structure of molecular complexes (e.g. ribosomes) in their physiological environment. Nevertheless, observation of highly resolved cellular mechanisms is challenging : i/ due to the low dose of electrons used to preserve specimen integrity during image acquisition, the amount of noise is very high ; ii/ due to technical limitations of the microscope, complete tilting of the sample (90^o) is impossible, resulting into a blind spot. As a consequence, projections are not available for a determined angle range, hence the term “limited angle tomography”.

The blind spot is observable in Fourier domain, where the missing projections appear as a missing wedge (MW). This separates the Fourier spectrum into two regions : the sampled region (SR) and the

unsampled regions (MW). The sharp transition between these two regions is responsible for a Gibbs-like phenomenon : ray- and side-artifacts emanate from high contrast objects (see Fig. 1.5), which can hide important structural features in the image. Another type of artifact arises from the incomplete angular sampling : objects appear elongated in the direction of the blind spot (see Fig. 1.5), in other words the data has an anisotropic resolution (e.g. linear features perpendicular to the tilt axis disappear). This elongation erases boundaries and makes it difficult to differentiate neighboring features. The quality of tomograms can be improved if sophisticated algorithms such as MBIR [Yan et al., 2019] are applied instead of conventional methods (e.g. WPB [Radermacher, 1992], SIRT [Gilbert, 1972]).

Filling up the MW with relevant data enables to potentially reduce or completely suppress these artifacts. Experimentally this can partially be achieved during data acquisition by using dual-axis tomography [Guesdon et al., 2013], where the sample is tilted with respect to the second axis. Consequently the blind spot is smaller and the MW becomes a missing pyramid, which results into a smaller missing spectrum. However dual-axis tomography is technically challenging and requires intensive post-processing in order to correct tilt and movement bias in the microscope. Another reconstruction approach consists in exploiting the symmetry of the underlying structure [Förster and Hegerl, 2007], but this can only be applicable to a limited number of biological objects (e.g. virus with either helical or icosahedral structure). Another common computational approach amounts to combining several hundred or thousands views of the same object, but with distinct blind spots. This so-called sub-tomogram averaging technique [Förster and Hegerl, 2007] is routinely used in cryo-ET, and is continuously improved for structure determination [Xu et al., 2016]. To improve sub-tomogram averaging and compensate the remaining MW artifacts, tomographic reconstruction algorithms with dedicated regularization have also been proposed in [Paavolainen et al., 2014, Leary et al., 2013].

The objective of our work is to design a statistical approach for the problem of recovering missing Fourier coefficients from a single volume in the situation where low and high frequency coefficients are missing in a specific and large region of the 3D spectrum. A simple way of handling MW artifacts is described in [Kováčik et al., 2014], where a dedicated spectral filter is used to smooth out the transition between SR and MW ; ray- and side-artifacts are reduced with this filter, but the object elongation remains in the resulting image. Inspired from [Maggioni et al., 2013], we have rather investigated MCMC methods to compute a MMSE estimator based on any non-local image denoiser to recover the missing information. We show that our Monte-Carlo sampling algorithm performs as well as the iterative method [Maggioni et al., 2013] but converges faster. Nevertheless, our concept is more general since any denoising method can be applied, included denoising algorithms dedicated to cryo-ET images [Frangakis and Hegerl, 2001, Fernandez and Li, 2003, Darbon et al., 2008, Wei and Yin, 2010, Kervrann et al., 2014b, Moreno et al., 2018]. In this Chapter, we focus on the cryo-ET restoration problem but the proposed algorithm could be potentially used to address a large range of applications including medical and seismic imaging, and other inverse scattering problems.

The remainder of the Chapter is organized as follows. In the next section, several existing methods for spectrum restoration and non-local and patch-based image denoising are reviewed. In Section 2.3, we formulate the reconstruction problem as an inverse problem. In Section 2.4, we recall the general Bayesian approach to derive a MMSE estimator. A Monte-Carlo Markov Chains method based on the Metropolis-Hastings algorithm are described to compute the underlying high-dimensional integral.

In Section 2.5, we adapt the general Bayesian framework to solve our inverse problem. An original Metropolis-Hastings procedure is presented to explore the large space of admissible solutions and to select relevant samples. Section 2.6 presents the experimental results obtained on simulated and real data. We illustrate the potential of our method with experiments on real cryo-tomogram images and we compare to other common algorithms.

2.2 Related work

We first focus on computational methods designed for spectrum restoration and Fourier coefficients recovering. Most of methods have been designed for 2D images and very few of them for 3D imaging. In general, the corruption process is supposed to be known and the artifacts observed in the input image, are due to a set of missing Fourier coefficients, well localized in the spectrum. First, several methods have been investigated to retrieve partially-missing phases of complex coefficients from modulus of coefficients in electron microscopy [Fienup, 1982] and time-frequency signal analysis [Kreme et al., 2018]. Here, we focus on another special case which consists in extrapolating the band-limited spectrum of an image up to higher frequencies. Nevertheless, these problems are generally formulated as denoising problems with specific reconstruction constraints. For instance, [Moisan, 2001] and [Guichard and Malgouyres, 1998] investigated the Total Variation (TV) minimization to extend the band-limited spectrum of an image. In [Lauze et al., 2017], the authors combine TV minimization and positivity constraints to reduce noise and artifacts, providing an inpainting-like mechanism for the sinogram missing data in limited-angle tomography (see also [Frikel and Quinto, 2013, Sentosun et al., 2017]). The common objective is to create new frequencies while preserving discontinuities and details in the restored image. Instead of explicitly imposing some regularity (e.g. Total variation, or robust regularization [Geman and Reynolds, 1992, Charbonnier et al., 1997]) on the solution, another successful restoration approach consists in exploiting the spatial redundancy of the input image. In [Chambolle and Jalalzai, 2014], a non-local method was suggested in the framework of variational methods for image reconstruction. In this approach, a patchwise similarity measure based on atoms corresponding to pseudo Gabor filters is designed to compare corrupted regions. Meanwhile, [Maggioni et al., 2013] adapted the concept of BM3D for recovering the missing spectrum applied to MRI imaging with very promising results on synthetic data. BM3D [Dabov et al., 2007] is a popular denoising algorithm which combines clustering of noisy patches, DCT-based transform and shrinkage operation to achieve the state-of-the-art results for several years. In our approach, we also focus on patch-based methods [Kervrann and Boulanger, 2008, Kindermann et al., 2005, Lou et al., 2010, Katkovnik et al., 2010, Pizarro et al., 2010, Milanfar, 2013, Sutour et al., 2014] to restore the input image corrupted by noise and non-linear transform. Indeed, it has been experimentally confirmed that the most competitive denoising methods are non-local and exploit self-similarities occurring at large distances in images, such as BM3D [Dabov et al., 2007], NL-Bayes [Lebrun et al., 2013], PLOW [Chatterjee and Milanfar, 2012], S-PLE [Wang and Morel, 2013], PEWA [Kervrann, 2014] and many other adaptative filters [Kervrann and Boulanger, 2006, Kervrann and Boulanger, 2007, Van De Ville and Kocher, 2009, Deledalle et al., 2009, Louchet and Moisan, 2011, Duval et al., 2011, Deledalle et al., 2012, Kervrann et al., 2014b, Jin et al., 2017], inspired from the seminal N(on)L(ocal)-means algorithm [Buades et al., 2005].

To complete the brief overview of non-local methods, we mention that a noisy image can also be restored from a set of noisy or "clean" patches or a learned dictionary. The statistics of a training set of image patches serve then as priors for denoising [Elad and Aharon, 2006, Mairal et al., 2009, Zoran and Weiss, 2011]. Another approach based multi-layer perceptron (MLP) exploiting a training set of noisy and noise-free patches was also able to achieve the state-of-the-art performance [Burger et al., 2012]. Very recently, [Buchholz et al., 2018] proposed to train content-aware restoration networks for denoising cryo-transmission electron microscopy data. While all these machine learning methods are attractive and powerful, computation is not always feasible in 3D because very large collection of 3D "clean" patches are required. In our study, we focus on unsupervised denoising methods since they are more flexible for real applications. They are less computationally demanding and are still competitive when compared to recent machine learning methods.

Our approach is mainly inspired from [Maggioni et al., 2013], but can use any competitive denoising methods for restoring the Fourier coefficients. The method proposed by [Maggioni et al., 2013] works by alternatively adding noise into the missing region and applying the BM4D algorithm which is the extension of BM3D [Dabov et al., 2007] to volumes. The authors interpret this iterative restoration method in the framework of compressed sensing with two information theory concepts in mind : *sparsity* of the signal in the transformed domain, and *incoherence* between the transform and the sampling matrix. Actually, BM4D does rely on a transform where the signal is sparse. Moreover, it is not clearly established that this transform is incoherent with the sampling matrix, defined by the support of the sampling region. Therefore, the proof of convergence is not clearly established, even though the authors presented convincing experimental results on synthetic images corrupted with white Gaussian noise. It remains unclear how the concept performs on experimental data and non Gaussian noise. To generalize this idea, we propose a statistical approach well-grounded in the Bayesian and MCMC framework and applied to challenging real data in cryo-ET. Our contributions are the following ones :

1. We present a MMSE estimator dedicated to the problem of MW restoration.
2. We propose an original Monte Carlo Markov Chain (MCMC) sampling procedure to efficiently compute the MMSE estimator.

2.3 Problem formulation and notation

Let us define a n-dimensional image $\boldsymbol{x} : S \subset \mathbb{Z}^3 \to \mathbb{R}$ assumed to be periodic and defined over a cubic domain $\Omega = [0, 1]^3$ and $n = |\Omega|$. The discrete Fourier transform of $\boldsymbol{x} = \{x(s), s \in S\}$ is then as follows :

$$\mathcal{F}\boldsymbol{x} : k \to \sum_{s \in S} \exp\left(-2i\pi k \cdot s\right) x(s), \tag{2.1}$$

where s is the coordinate of point in spatial domain S. In our problem, one considers a corrupted image denoted $\boldsymbol{y} = \{y(s), s \in S\}$ defined as

$$y(s) = \sum_{k \in \overline{W}} \exp\left(2i\pi k \cdot s\right) \mathcal{F}\boldsymbol{x}[k] \tag{2.2}$$

where $\overline{W}$ is the sampled spectral region (SR) where the Fourier coefficients $\mathcal{F}\boldsymbol{x}[k]$ are positive and non-zero. The region $\overline{W}$ is equivalent to the support of a binary mask $\boldsymbol{m} \in \{0,1\}^S$ such as $\boldsymbol{m}[k] = 1$ if $k \in \overline{W}$ and 0 otherwise : $\overline{W} = \text{supp}(\boldsymbol{m})$. The so-called missing wedge W is assumed to be symmetric with respect to the origin as illustrated in Fig. 1.5 (bottom right), and $S = W \cup \overline{W}$. In what follows, we assume that the clean image $\mathcal{F}\boldsymbol{x}$ is known over the region $\overline{W}$. Our objective is then to estimate $\boldsymbol{x} : S \to \mathbb{R}$ in the whole domain S such that

$$\forall k \in \overline{W}, \;\; \mathcal{F}\boldsymbol{x}[k] = \mathcal{F}\boldsymbol{y}[k] \tag{2.3}$$

and $\forall s \in S, x(s) > 0$. In other words, the set of known Fourier coefficients will be preserved by the restoration procedure. The challenge is to recover the unknown set of low, middle and high frequencies in a large region in the spectrum. This amounts to applying an interpolation operator φ_W to the spectrum of $\boldsymbol{y}$ to get an estimator $\widehat{\boldsymbol{x}}$ of $\boldsymbol{x}$:

$$\widehat{\boldsymbol{x}} = \mathcal{F}^{-1} \circ \varphi_W \circ \mathcal{F}\boldsymbol{y}. \tag{2.4}$$

In the sequel, we describe a Bayesian approach as conventionally proposed for solving such an inverse problem.

2.4 Bayesian estimator and Monte Carlo Markov Chain sampling

Solving inverse problems in image processing consists in estimating an unknown image $\boldsymbol{x} \in \mathcal{X}$ given an image $\boldsymbol{y} \in \mathcal{Y}$. Different sources of distortion may cause damages on the ideal image, including noise, blur, and projections. In the Bayesian framework, the whole information once the data have been collected, is represented by the posterior probability density function (pdf) defined via the Bayes' Theorem :

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})}{p(\boldsymbol{y})}, \tag{2.5}$$

where $p(\boldsymbol{y}|\boldsymbol{x})$ denotes the likelihood function, $p(\boldsymbol{x})$ is the prior pdf and $p(\boldsymbol{y})$ is the marginal distribution of $\boldsymbol{y}$ which is in general unknown and not computable.

2.4.1 Bayesian estimators

In this section, $\boldsymbol{x}$ and $\boldsymbol{y}$ are realizations of a random variable $\boldsymbol{X}$ (with a pdf $p(\boldsymbol{x})$) and a random realization of $\boldsymbol{Y}$ respectively. Given a cost function $C : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$, a Bayesian estimator is defined as the minimizer of expected risk $\mathbb{E}_p[C(\boldsymbol{X}, \widehat{\boldsymbol{x}}(\boldsymbol{Y}))]$ wrt the joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$ of the pair $(\boldsymbol{X}, \boldsymbol{Y})$. Several Bayesian estimators can be derived based on the choice of the cost function C.

MAP estimator The most conventional choice is $C(\boldsymbol{x}, \widehat{\boldsymbol{x}}) = 1 - \delta(\boldsymbol{x}, \widehat{\boldsymbol{x}})$ where δ is the Kronecker symbol. The corresponding Maximum A Posteriori estimator, defined as

$$\begin{aligned} \widehat{\boldsymbol{x}}_{MAP} &= \arg\max p(\boldsymbol{x}|\boldsymbol{y}) \\ &= \arg\min_{\boldsymbol{x}} \{-\log p(\boldsymbol{y}|\boldsymbol{x}) - \log p(\boldsymbol{x})\}, \end{aligned} \tag{2.6}$$

selects the most likely image $\boldsymbol{x}$, that is the solution corresponding to the mode of the posterior distribution $p(\boldsymbol{x}|\boldsymbol{y})$.

Furthermore, if we assume that the prior and likelihood functions are represented by Gibbs functions, the posterior distribution has the following form

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{Z} \exp\left(-\frac{U(\boldsymbol{x},\boldsymbol{y})}{\tau}\right) \tag{2.7}$$

where Z is normalizing factor, $U(\boldsymbol{x},\boldsymbol{y}) = D(\boldsymbol{x},\boldsymbol{y}) + \phi(\boldsymbol{x})$ is an energy functional composed of a data-fidelity term $D(\boldsymbol{x},\boldsymbol{y})$ and a prior term $\phi(\boldsymbol{x})$, and τ can be interpreted as a "temperature" or scale parameter. The prior generally encourages piecewise smoothness (TV) or sparsity of $\boldsymbol{x}$. Hence, the MAP formulation is equivalent to the popular variational problem which amounts to computing the unique image $\boldsymbol{x}$ that minimizes the following criterion :

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} D(\boldsymbol{x},\boldsymbol{y}) + \phi(\boldsymbol{x}). \tag{2.8}$$

Typically $\phi(\boldsymbol{x}) = \|\nabla \boldsymbol{x}\|_1$ ($\nabla\cdot$ is the gradient operator) is the total variation regularizer and serves to smooth the image $\boldsymbol{x}$ while preserving image discontinuities.

MMSE estimator Another well-known Bayesian estimator can be derived if we consider the quadratic cost function $C(\boldsymbol{x},\widehat{\boldsymbol{x}}) = \|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|^2$. The Minimum Mean Square Error (MMSE) estimator is the posterior expectation (or conditional mean) defined as :

$$\widehat{\boldsymbol{x}}_{MMSE} = \mathbb{E}[\boldsymbol{X}|\boldsymbol{Y} = \boldsymbol{y}] = \int_{\mathcal{X}} \boldsymbol{x}\, p(\boldsymbol{x}|\boldsymbol{y})\, d\boldsymbol{x}. \tag{2.9}$$

If the posterior is modeled as Gibbs distribution, we get :

$$\widehat{\boldsymbol{x}}_{MMSE} = \frac{\displaystyle\int_{\mathcal{X}} \exp\left(-\frac{U(\boldsymbol{x},\boldsymbol{y})}{\tau}\right)\, \boldsymbol{x}\, d\boldsymbol{x}}{\displaystyle\int_{\mathcal{X}} \exp\left(-\frac{U(\boldsymbol{x},\boldsymbol{y})}{\tau}\right)\, d\boldsymbol{x}}. \tag{2.10}$$

In our case, the MMSE estimator is typically intractable since the underlying integral involves several thousands of variables (typically n is the number of pixels in the image). The MMSE estimator cannot be computed in a closed form, and numerical approximations are typically required. In high-dimensional space, a common approach consists in approximating the integral by using Monte Carlo (MC) simulation techniques [Robert and Casella, 2004] as explained in the next section.

However, we draw the readers'attention to the fact that it has been shown that the MMSE estimator has connections with the variational optimization problem in the case of an image corrupted by white Gaussian noise :

$$\widehat{\boldsymbol{x}}_{MMSE} = \arg\min_{x} \|\boldsymbol{y} - \boldsymbol{x}\|^2 + \psi(\boldsymbol{x}), \tag{2.11}$$

where the function $\psi(\boldsymbol{x})$ can be seen as a pseudo-prior which differs from the prior distribution $p(\boldsymbol{x}) \propto \exp(-\phi(\boldsymbol{x}))$. Nevertheless, except in the case of a explicit and dedicated prior discussed in [Protter et al.,

2010, Gribonval, 2011, Louchet and Moisan, 2013, Kazerouni et al., 2013], it is not possible to derive an analytical form of $\psi(\boldsymbol{x})$ from $\phi(\boldsymbol{x})$, especially if the data-fidelity term is not quadratic. Accordingly, the most practical way to compute a MMSE estimator in the case of complex data-fidelity terms and prior terms is to applying the MCMC approach.

2.4.2 Monte-Carlo integration

Let us consider T independent and identically distributed (i.i.d.) samples $\boldsymbol{x}^{(1)}, \cdots \boldsymbol{x}^{(T)}$ drawn from a target pdf $\pi(\boldsymbol{x}) := p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})$. A consistent estimator can be computed as

$$\bar{\boldsymbol{x}}_T := \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{x}^{(t)} \xrightarrow{p} \widehat{\boldsymbol{x}}_{MMSE}, \tag{2.12}$$

i.e. the empirical mean of samples converges in probability to $\widehat{\boldsymbol{x}}_{MMSE}$ due to the weak law of large numbers. Formally, for any positive number $\epsilon > 0$, we have

$$\lim_{T \to +\infty} \mathsf{Pr}\left(|\bar{\boldsymbol{x}}_T - \widehat{\boldsymbol{x}}_{MMSE}| > \epsilon\right) = 0. \tag{2.13}$$

The Monte-Carlo estimator is unbiased $\mathbb{E}_\pi[\bar{\boldsymbol{x}}_T] = \widehat{\boldsymbol{x}}_{MMSE}$ and converges provided that the samples $\boldsymbol{x}^{(t)}$ are i.i.d.. In that case, the variance of $\bar{\boldsymbol{x}}_T$ defined as $\mathsf{Var}[\bar{\boldsymbol{x}}_T] = \upsilon^2/T$ (where $\upsilon^2 = \mathsf{Var}[\boldsymbol{X}]$) decreases with the number of samples, and $\bar{\boldsymbol{x}}_T$ is Gaussian distributed due to the central limit theorem : $\bar{\boldsymbol{x}}_T \sim \mathcal{N}(\widehat{\boldsymbol{x}}_{MMSE}, \upsilon^2/T)$ when $T \to +\infty$.

A central question is then to draw a series of i.i.d. samples. The most conventional approach in high-dimensional space is to consider Markov chain Monte Carlo (MCMC) algorithms [Gilks et al., 1995, Robert and Casella, 2004, Liang et al., 2010]. In the sequel, we focus on a few important components of the MCMC machinery and we discuss the convergence properties of the Metropolis-Hastings algorithm used in our approach to generate a Markov chain with a target stationary distribution π.

Simulating independent samples

Drawing independent samples from the target pdf $\pi(\boldsymbol{x})$ cannot be directly applied. A MCMC procedure is able to simulate an ergodic and stationary Markov chain given a target pdf $\pi(\boldsymbol{x})$ and a starting state state $\boldsymbol{x}^{(0)}$. The set of samples $\boldsymbol{x}^{(0)} \to \cdots \to \boldsymbol{x}^{(T)}$ are generally correlated samples, but it has been established that Monte-Carlo estimator is consistent as $T \to +\infty$. In [Louchet and Moisan, 2008, Louchet and Moisan, 2013], the authors also studied the behavior of the expected approximation error

$$\begin{aligned}\mathbb{E}[\|\tilde{\boldsymbol{x}}_T - \widehat{\boldsymbol{x}}_{MMSE}\|^2] = \\ \mathbb{E}[\|\tilde{\boldsymbol{x}}_T - \mathbb{E}[\tilde{\boldsymbol{x}}_T]\|^2] + \|\mathbb{E}[\tilde{\boldsymbol{x}}_T] - \widehat{\boldsymbol{x}}_{MMSE}\|^2\end{aligned} \tag{2.14}$$

decomposed into the sum of the trace of the covariance matrix (or span) and the squared bias which entails the loss of efficiency of the sampling procedure.

In practice, a MCMC method will provide better performance than another MCMC method if the

samples present less correlation. On the contrary, it is required to generate many samples to reduce the variance of the estimator.

Burn-in phase

Another consequence of the correlation is the burn-in period that the chain requires before converging to the invariant target pdf π. In general, the initial T_b samples are discarded and not included in the computation of the estimator [Robert and Casella, 2004] :

$$\tilde{\boldsymbol{x}}_T := \frac{1}{T - T_b} \sum_{t=T_b}^{T} \boldsymbol{x}^{(t)}. \tag{2.15}$$

However, the length T_b of the burn-in period cannot be easily predicted even if a few studies in the literature focused on that problem [Gelman and Rubbin, 1992, Brooks and Gelman, 1998].

Metropolis-Hastings algorithm

The most popular and widely applied MCMC algorithm is based on the Metropolis-Hastings procedure [Metropolis et al., 1953, Hastings, 1970] described below. The MH algorithm involves the definition of the proposal density $q(\mathbf{z}|\mathbf{x}), \boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$ to move from the state $\boldsymbol{x}$ to state $\boldsymbol{z}$, and the acceptance probability $0 \leq a(\boldsymbol{x}, \boldsymbol{z}) \leq 1$. The transition probability is then defined as : $p(\boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{z}|\boldsymbol{x})a(\boldsymbol{z}, \boldsymbol{x})$. In the MH procedure, a sample $\boldsymbol{z}$ is drawn from the proposal distribution and then a test is applied to accept the transition from the state $\boldsymbol{x}$ to the state $\boldsymbol{z}$ or not. If the transition is not accepted, the chain remains in the same state as before :

The Metropolis-Hastings algorithm

1. Set an initial state $\boldsymbol{x}^{(0)}$.
2. **For** $t = 1, \cdots, T$ **do**
   (a) Draw a sample $\boldsymbol{z} \sim q(\boldsymbol{x}|\boldsymbol{x}^{(t-1)})$.
   (b) Compute the acceptance probability

$$a(\boldsymbol{x}^{(t-1)}, \boldsymbol{z}) = \min\left[1, \frac{\pi(\boldsymbol{z})q(\boldsymbol{x}^{(t-1)}|\boldsymbol{z})}{\pi(\boldsymbol{x}^{(t-1)})q(\boldsymbol{z}|\boldsymbol{x}^{(t-1)})}\right].$$

   (c) Draw α from a uniform distribution : $\alpha \sim U(0, 1)$.
   (d) **If** $\alpha \leq a(\boldsymbol{x}^{(t-1)}, \boldsymbol{z})$ **then** $\boldsymbol{x}^{(t)} = \boldsymbol{z}$,
   else $\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)}$.
   end if
   end for

The MH algorithm returns a set of $T_b - T$ correlated samples if we discard the T_b first samples. Under some mild regularity conditions, it has been established that the pdf of $\mathbf{x}^{(t)}$ converges to the target pdf π when t increases [Robert and Casella, 2004]. In general, the MH algorithm satisfies the so-called detailed balance condition :

$$(\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X},\ p(\boldsymbol{x}|\boldsymbol{z})\pi(\boldsymbol{z}) = p(\boldsymbol{z}|\boldsymbol{x})\pi(\boldsymbol{x}),$$

(the chain is reversible) which is a sufficient condition to guarantee that the chain is ergodic and has π as stationary distribution [Robert and Casella, 2004]. Note that reversibility of the chain is not a necessary condition ; recent studies experimentally show that non-reversible Markov chains may provide better convergence, i.e. the number T of samples can be lower than the reversible chains [Neal, 2004, Bierkens, 2015] .

In practice, the proposal density and the acceptance probability can be modified in order to improve the performance of the algorithm, and always ensuring the ergodicity of the chain. Actually, the proposal pdf q should be chosen as close as possible to the target pdf π. In what follows, we mainly focus on the specification of the proposal densities to improve convergence.

Choices of the proposal density

There is a large flexibility in the choice of proposal function and it is a challenge to find a proposal function that is able to use the data efficiently in order to obtain satisfactory convergence. Below, we discuss four possible proposals.

- Assume that the proposal satisfies the equality $q(\boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{x}|\boldsymbol{z})$ (e.g. uniform distribution), then the acceptance probability is simplified since a sample $\boldsymbol{z}$ having a higher value $\pi(\boldsymbol{z})$ is always accepted, whereas the samples with smaller values $\pi(\boldsymbol{z}) < \pi(\boldsymbol{x})$ are accepted with a probability lower than 1.
- The proposal pdf has the following form : $q(\boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{z} - \boldsymbol{x})$. This means that the new state is explicitly randomly found in the neighborhood of the current state $\boldsymbol{x}$. This proposal is then viewed as as random walk and enables to progressively explore the large space of possible states. Nevertheless, the random walk MH algorithm (1953) tends to stay in the same state for a long period but the chain has not converged.
- When the target density is differentiable, the proposal can be generated in accordance with an approximation of the Langevin diffusion process [Girolami et al., 2011] : $z \sim \mathcal{N}(x + \frac{\delta}{2}\nabla \log \pi(x), \delta)$ for a given small value δ.
- The idea of adaptive MH algorithm consists in updating the proposal distribution by using all the information collected so far about the target distribution [Holden et al., 2009]. First, it has been suggested to model the proposal density as a Gaussian distribution centered on the current state with a covariance computed from a fixed finite number of previous states. Given the whole history $(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(t-1)})$, the new state $\boldsymbol{z}$ is obtained from $q(\boldsymbol{z}|\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(t-1)})$ assumed to be symmetric.
- If the proposal pdf $q(\boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{x})$ does not depend on the state $\boldsymbol{x}$, the acceptance probability is

defined as

$$a(\boldsymbol{x}, \boldsymbol{z}) = \min \left[1, \frac{\pi(\boldsymbol{z})q(\boldsymbol{x})}{\pi(\boldsymbol{x})q(\boldsymbol{z})}\right].$$

The independent Metropolis-Hastings algorithm is an efficient sampling algorithm only if q is reasonably close to π. An attractive property of independent proposals is their ability to make large jumps while keeping the acceptance rate high. Consequently, the autocorrelation of the chain decreases rapidly.

Note that more sophisticated proposal rules are generally recommended to address high-dimensional problems [Girolami et al., 2011]. For instance, a two-step optimization approach is typically appropriate for sampling Gaussian distributions [Papandreou and Yuille, 2010, Gilavert et al., 2015]. Another sophisticated approach is based on data augmentation and the adding of auxiliary variables to improve convergence speed if the samples are correlated [Marnissi et al., 2018].

In summary, the convergence of the chain depends on the specification of the proposal density but it is also established that the ideal proposal pdf must as close to possible to the target pdf. In that case, the MH procedure generates a sequence of states with low correlations and converges faster. In our approach described in the next section, we investigated a stochastic scheme to generate samples with low correlations in the context of MW restoration.

2.5 A MH algorithm for missing wedge restoration

Let us consider the following image model

$$\boldsymbol{y} = \mathcal{D}_W(\boldsymbol{x}), \tag{2.16}$$

where $\boldsymbol{y}, \boldsymbol{x} \in \mathbb{R}^n$, and $\mathcal{D}_W(.)$ is a degradation operator setting to zero the Fourier coefficients belonging to the MW support W assumed to be known. Our objective is to compute a MMSE estimator defined as

$$\widehat{\boldsymbol{x}}_{MMSE} = \frac{\displaystyle\int_{\mathcal{X}} p(\boldsymbol{x}|\boldsymbol{y})\boldsymbol{x}d\boldsymbol{x}}{\displaystyle\int_{\mathcal{X}} p(\boldsymbol{x}|\boldsymbol{y})d\boldsymbol{x}} = \frac{\displaystyle\int_{\mathcal{X}} p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})\boldsymbol{x}d\boldsymbol{x}}{\displaystyle\int_{\mathcal{X}} p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x}} \tag{2.17}$$

by using a dedicated MH algorithm, where $p(\boldsymbol{x})$ is the prior used to encourage the solution to be positive and piecewise smooth. In our modeling, the likelihood function is composed of two terms : i/ we impose that the Fourier coefficients of the reconstructed image are very similar to the known coefficients of the corrupted image $\boldsymbol{y}$, i.e. $|\mathcal{F}\boldsymbol{x}[k] - \mathcal{F}\boldsymbol{y}[k]| < \delta$; ii/ we consider a data fidelity term defined as the L_2 norm (alternative data fidelity terms will be considered in our experiments) between the corrupted image and the restored image degraded by the operator $\mathcal{D}_W$, i.e. :

$$D(\boldsymbol{x}, \boldsymbol{y}) = \sum_{s \in S} \left(y(s) - \sum_{k \in S} e^{2i\pi k \cdot s} \mathcal{F}\boldsymbol{x}[k] \right)^2. \tag{2.18}$$

It follows that the posterior is defined as ($\mathbb{1}_{\mathcal{A}}(\boldsymbol{z}) = 1$ if $\boldsymbol{z}$ belongs to the set $\mathcal{A}$ and zero otherwise)

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto \underbrace{\exp\left(-\frac{D(\boldsymbol{x},\boldsymbol{y})}{\beta}\right)\mathbb{1}_{\mathcal{A}_{\delta,\boldsymbol{y}}}(\boldsymbol{x})}_{likelihood}\ \underbrace{\mathbb{1}_{\mathcal{A}'_{\lambda}}(\nabla\boldsymbol{x})\,\mathbb{1}_{\mathcal{X}_+}(\boldsymbol{x})}_{prior}, \tag{2.19}$$

such that $\mathcal{A}_{\delta,\boldsymbol{y}}(\boldsymbol{x}) = \{\boldsymbol{x} \in \mathcal{X} : \forall k \in \overline{W}\ \ |\mathcal{F}\boldsymbol{x}[k] - \mathcal{F}\boldsymbol{y}[k]| < \delta\}$ and $\mathcal{A}'_{\lambda}(\nabla\boldsymbol{x}) = \{\boldsymbol{x} \in \mathcal{X} : \|\nabla\boldsymbol{x}\|_1 \leq \lambda\}$ where $\|\nabla\boldsymbol{x}\|_1 = \sum_{s\in S}|\nabla x(s)|$ is the discrete Total Variation [Rudin et al., 1992]. Here, $\mathcal{X}_+$ denotes the set of positive solutions, that is the set of images for which $x(s) \geq 0, \forall s \in S$. As mentioned in Section 2.4, β can be interpreted in (2.19) as a "temperature" parameter.

Consequently, the MMSE estimator can reformulated as

$$\widehat{\boldsymbol{x}}_{MMSE} = \frac{\displaystyle\int_{\Gamma}\exp\left(-\frac{D(\boldsymbol{x},\boldsymbol{y})}{\beta}\right)\boldsymbol{x}\,d\boldsymbol{x}}{\displaystyle\int_{\Gamma}\exp\left(-\frac{D(\boldsymbol{x},\boldsymbol{y})}{\beta}\right)d\boldsymbol{x}}, \tag{2.20}$$

where the set Γ of admissible solutions is defined as

$$\begin{aligned}\Gamma = \{\boldsymbol{x} \in \mathcal{X} : \forall k \in \overline{W}\ \ |\mathcal{F}\boldsymbol{x}[k] - \mathcal{F}\boldsymbol{y}[k]| < \delta,\\ \forall s \in S, x(s) \geq 0, \text{ and } \|\nabla\boldsymbol{x}\|_1 < \lambda\}.\end{aligned} \tag{2.21}$$

The performance of MMSE (2.20) depends on the pre-specified thresholds λ and δ. In practice, these values do not need to be accurately adjusted in practice as discussed below. Meanwhile, because of the high dimensionality of the problem, we need an efficient MH algorithm to compute MMSE.

The efficiency of a MH algorithm depends on the choice of the proposal distribution $q(.|\boldsymbol{x}^{(t-1)})$. In practice, the proposal generator leads to correlated samples $\boldsymbol{x}^{(t-1)}$ induced by the two following factors : i/ by construction, the newly proposed state $\boldsymbol{x}^{(t)} \sim q(.|\boldsymbol{x}^{(t-1)})$ is generated from the current state ; ii/ the new state $\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)}$ when the proposed move has been rejected. Note that this correlation is not known in advance but can be empirically estimated and updated from the previous samples [Holden et al., 2009]. To achieve good performance, a well-chosen proposal distribution both allows significant changes between the subsequent states with a high probability of acceptance (see Section 4.2.4). Unfortunately these requirements cannot be satisfied easily in practice. If we choose a proposal distribution with small moves, the probability of acceptance will be high, however the resulting chain will be highly correlated, as $\boldsymbol{x}^{(t)}$ changes only very slowly. In return, if we choose a proposal distribution with large moves, the probability of acceptance will be rather low. Accordingly, we investigated an original strategy to generate a sequence of moves with a probability of acceptation in the range of $a \in [0.25 - 0.6]$. In theoretical studies, it has been shown that the optimal probability of acceptance $a^\star = 0.234$ [Roberts et al., 1997] whereas $a^\star = 0.574$ in [Roberts and Rosenthal, 2001].

Given an initial state $\boldsymbol{x}^{(0)} \in \Gamma$, we explore the neighborhood of the current value set $\boldsymbol{x}^{(t)}$ of the chain. Our proposal distribution q, which enables to potentially move from $\boldsymbol{x} \in \Gamma$ to $\boldsymbol{x}' \in \Gamma$ is chosen as (also see Fig. 2.1) :

$$\boldsymbol{x}' = \mathcal{D}_{\lambda}(\mathcal{P}_W(\boldsymbol{x} + \boldsymbol{\varepsilon})), \tag{2.22}$$

where $\varepsilon \sim \mathcal{N}(0, I_{n\times n}\sigma_\varepsilon^2)$ is a white Gaussian noise, $I_{n\times n}$ is the n-dimensional identity matrix, $\mathcal{P}_W$ is a projection operator that impose that the $\forall k \in \overline{W}, \quad \mathcal{F}(\boldsymbol{x}+\varepsilon)[k] = \mathcal{F}\boldsymbol{y}[k]$ and $\mathcal{D}_\lambda$ is denoising operator that ensures that the total variation of the denoised image is lower than λ. Consequently, the distribution of increments $\boldsymbol{x}' - \boldsymbol{x}$ is not parametric due to the nonlinear operators involved in (2.22). In the sequel, we only assume that this non-parametric distribution q is approximately symmetric. Even though visualization of the empirical proposal density is not possible, we suggest that (2.22) tends to produces similar samples (denoised images) concentrated around some empirical mean belonging to Γ, with a few moves quite far away from this mean.

Our simulator can be viewed as a random walk in a high-dimensional space, where all the pixels of the images are modified at once. Note that in [Louchet and Moisan, 2008, Louchet and Moisan, 2013], only one pixel is modified at each iteration to compute the TV-LSE estimator. Our approach can be seen also a blockwise MH sampling procedure but only one block corresponding to the whole image, is considered in procedure. In our experiments, we observed a high acceptance rate, suggesting that the new sample is not far from the previous one (i.e. $\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\|$ is small). To our knowledge, this is the first time that such a proposal, is used in the context of image restoration and inverse problems. Our MH algorithm for MW restoration is then as follows (see Fig. 2.1) :

The Metropolis-Hastings algorithm for MW restoration
Set an initial state $\boldsymbol{x}^{(0)} \in \Gamma$.
For $t = 1, \cdots, T$ **do**

1. Generate a new state $\boldsymbol{z}$ with the a three-step approach :
   — **Perturbation** : $\boldsymbol{x}^{(t-1)} + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, I_{n\times n}\sigma_\varepsilon^2)$.
   — **Projection :** of $\mathcal{P}_W(\boldsymbol{x}^{(t-1)} + \varepsilon)$ onto the subspace of images having the same observed frequencies as $\boldsymbol{y}$ if $k \in \overline{W}$.
   — **Denoising** of $\mathcal{P}_W(\boldsymbol{x}^{(t-1)} + \varepsilon)$ to get an image with a small $||\nabla \boldsymbol{x}||_1$ and set $\boldsymbol{z} = \mathcal{D}_\lambda(\mathcal{P}_W(\boldsymbol{x}^{(t-1)} + \varepsilon))$.
2. Compute the acceptance probability

$$a(\boldsymbol{x}^{(t-1)}, \boldsymbol{z}) = \min\left[1, \exp\left(\frac{D(\boldsymbol{x}^{(t-1)}, \boldsymbol{y}) - D(\boldsymbol{z}, \boldsymbol{y})}{\beta}\right)\right].$$

3. Draw α from a uniform distribution : $\alpha \sim U(0, 1)$
4. **If** $\alpha \leq a(\boldsymbol{x}^{(t-1)}, \boldsymbol{z})$ **then** $\boldsymbol{x}^{(t)} = \boldsymbol{z}$
   else $\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)}$.
   end if

end for

In the end, the computational method is governed actually by three parameters : the number of iterations T, the noise variance σ_ϵ^2 and the scaling parameter β. At each iteration t, the denoising algorithm removes the perturbation noise. The parameter β affects the acceptance rate of the evaluation step. The

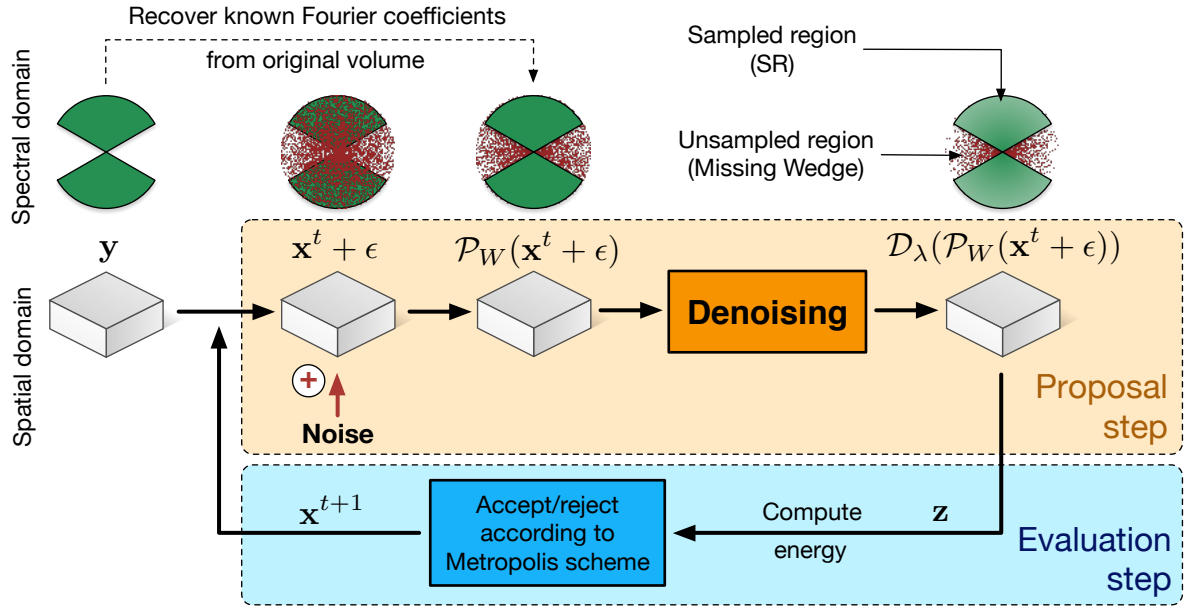

FIGURE 2.1 – The MCMC method flowchart. The 1^{st} icon row represents the data in the spectral domain, the 2^{nd} in spatial domain.

higher the value of β, the higher the acceptance rate. For a high enough β value, all proposed samples are accepted and we fall back on the original method [Maggioni et al., 2013].

Unlike [Maggioni et al., 2013], we propose a statistical physics and energy minimization framework for MW restoration. In [Maggioni et al., 2013], all candidate samples are accepted at each iteration and used to compute an aggregated estimator. It is worth noting that the standard deviation σ_ϵ decreases through iterations, and the final estimate is obtained by weighting the samples $\boldsymbol{x}^{(t)}$ with weights equal to $1/\sigma_\epsilon$ updated at each iteration. Hence, this gives more importance to the last samples. In our case, σ_ϵ is constant through iterations, and the weights results naturally from the MCMC sampling which selects the most appropriate generated samples. The most frequent accepted samples have higher weights in the computation of the Monte-Carlo estimator (2.12). Finally, we add a "periodic plus smooth image decomposition" [Moisan, 2011] operation prior to Fourier transforms, in order to reduce artifacts originating from image borders. Indeed, we noticed that a cross-structure tends to emerge in the restored MW and gets amplified through iterations. This structure is a well-known spectral artifact of the Fourier transform, resulting from the false assumption that the images are periodic signals, when in reality the images rarely have similar opposite borders. Applying this decomposition allows to reduce the cross structure and solves the problem.

2.6 Experimental results

The MW restoration method has been evaluated experimentally on synthetic noisy data by varying the parameters and the components of the MH algorithm. Furthermore, we demonstrate the potential of the

method on real cryo-ET data.

2.6.1 Results on synthetic data

In this section, we justify the choice of algorithm parameters, then evaluate robustness to noise, and finally compare our approach to a few other competitive methods. We consider an artificial dataset (Dataset A) which consists of a density map of the 20S proteasome corrupted by additive white Gaussian noise and by applying an artificial MW process, which amounts to setting to zero the Fourier coefficients within an artificial wedge shaped mask. Given the ground-truth $\boldsymbol{x}$, we use two similarity measures for quantitatively evaluating the restoration results $\hat{\boldsymbol{x}}$: the peak signal-to-noise ratio (PSNR) and the constrained correlation coefficient (CCC), defined as

$$\text{PSNR}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{\max_{s\in S} x(s)}{\frac{1}{n}\sum_{s\in S}(x(s)-\hat{x}(s))^2}, \tag{2.23}$$

$$\text{CCC}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{\sum_{k\in W} \mathcal{F}\boldsymbol{x}[k]\mathcal{F}\hat{\boldsymbol{x}}[k]^*}{\sqrt{\sum_{k\in W}\mathcal{F}\boldsymbol{x}[k]^2 \sum_{k\in W}\mathcal{F}\hat{\boldsymbol{x}}[k]^2}}. \tag{2.24}$$

The PSNR is a common score in image denoising, and is well adapted to estimate the global quality of processed images. CCC is a score used in cryo-ET for sub-tomogram averaging [Förster and Hegerl, 2007]. This score is very similar to the Pearson's correlation coefficient measured in Fourier domain. Note that only a constrained region of the Fourier domain is considered for CCC computation. In our work, we use CCC to quantify the quality of retrieved Fourier coefficients located in the region W.

Analysis of performance of denoising algorithms

First, we study the influence of several denoising algorithms embedded in our MCMC framework. We focus on three 2D methods : BM3D [Dabov et al., 2007], non-local Bayes (NL-Bayes) [Lebrun et al., 2013], total variation (TV) [Rudin et al., 1992]. Since it was not possible to extend all denoisers in 3D, we applied the method on a 2D slice of the 3D volume for assessment. As shown in Fig. 2.2, the best results have been obtained by using BM3D, both in terms of PSNR and visual quality. Nevertheless, NL-Bayes produced very similar results. TV denoising produced noticeable worse results. It turns out that the performance strongly depends on the ability of the denoising algorithm to remove the Gaussian noise. This experiment confirms that our MCMC procedure achieves better results than traditional denoising algorithms. In addition, the best results are obtained with the BM3D and NL-Bayes algorithms embedded in our MH algorithm. This is consistent with the literature in image denoising. Finally, we confirm that any image denoising algorithm (we also tested NL-means [Buades et al., 2005] PEWA [Kervrann, 2014], OWE [Jin et al., 2017]) allow us to produce a restored image with less MW artifacts. Because faster and very performant in terms of PSNR values, we used BM4D [Maggioni et al., 2013] which a 3D extension of BM3D, for processing 3D cryo-ET data.

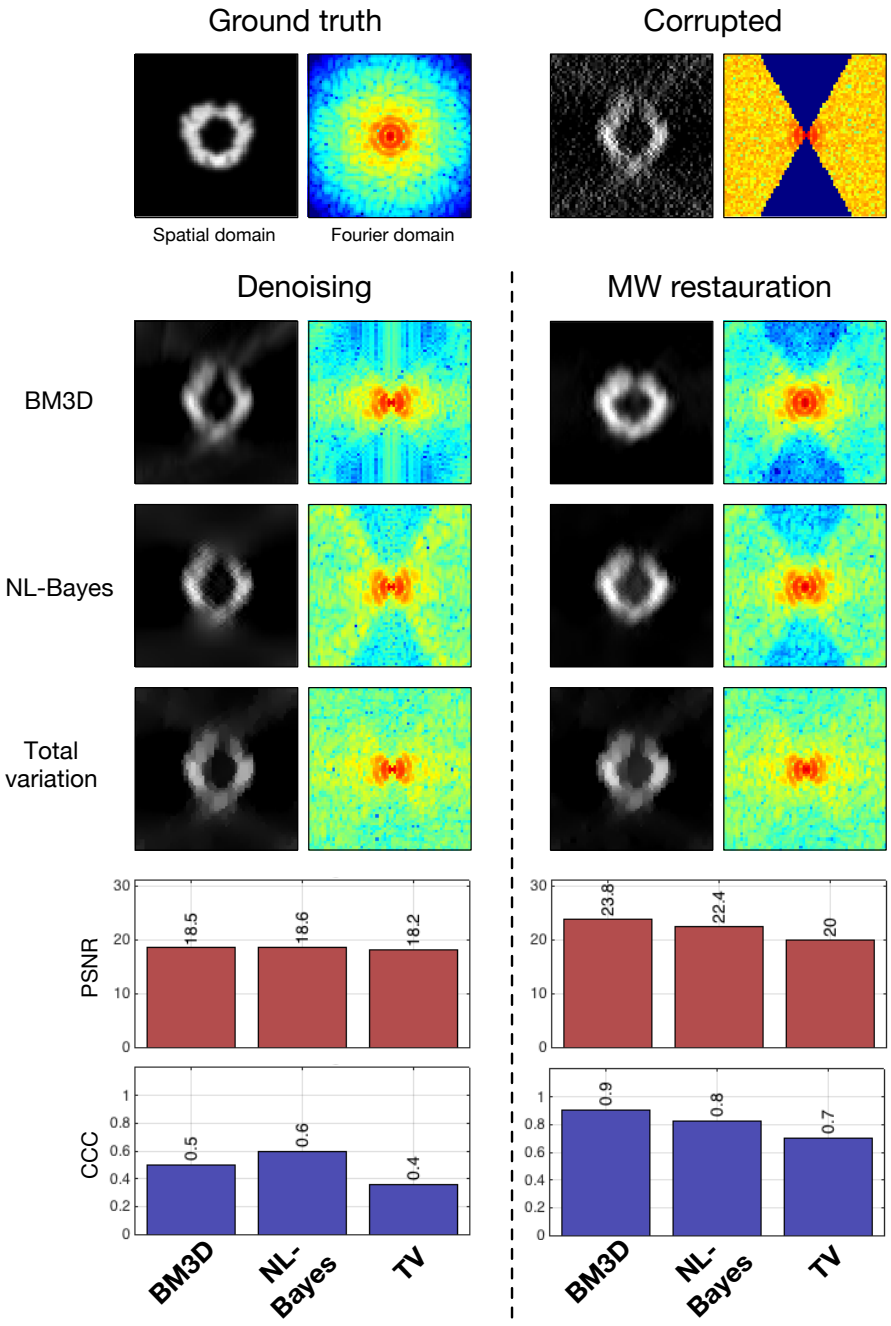

FIGURE 2.2 – Dataset A (2D) : comparison of denoising algorithms. First, we compare conventional denoisers (left column) to our MH method (right column). Second, we compare the results obtained by applying three different 2D denoising algorithms : BM3D, NL-Bayes, and ROF denoising. On the bottom, we evaluate the performance in terms of PSNR and CCC values. It turns out that our MH method performs better than conventional denoisers in all situations.

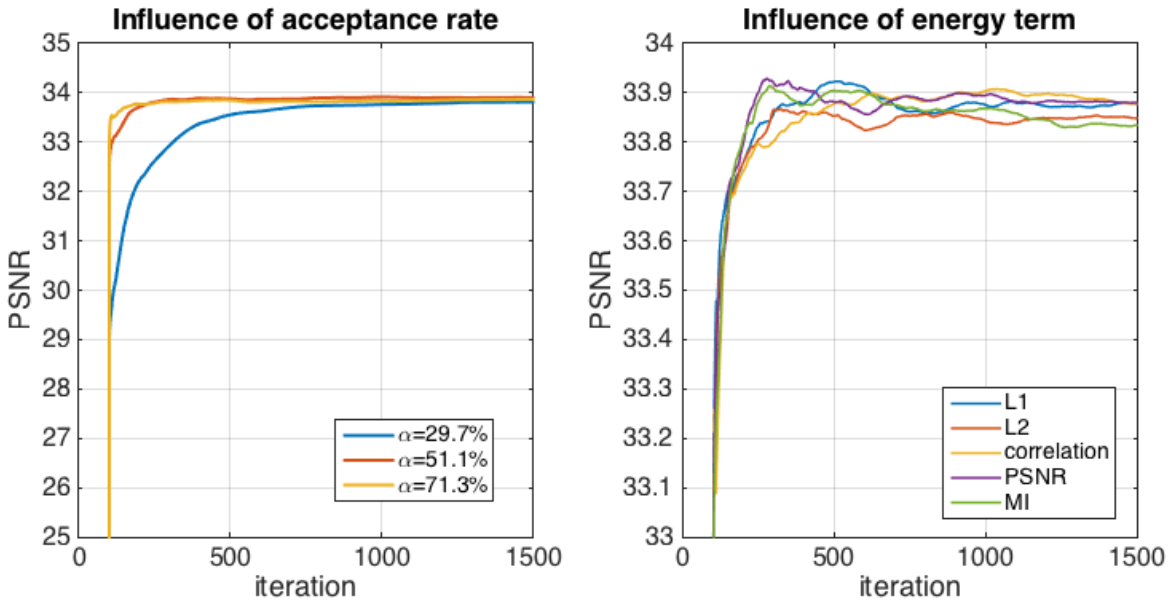

FIGURE 2.3 – Dataset A : This figure shows the impact of algorithm parameters. We illustrate these effects in terms of PSNR through iterations. We do not show images because obtained results are visually similar. On the left, the influence of parameter β controlling the acceptance rate of the MH sampling, is shown : $\beta = 1.5 \times 10^{-5}$ (blue), $\beta = 2.0 \times 10^{-5}$ (red), $\beta = 4.0 \times 10^{-5}$ (yellow). Clearly, the choice of β affects the convergence speed, however in all cases our method converges to the same result. On the right, the influence of the data fidelity term is presented. We compare the L_1 (2.25) and L_2 norms (2.26), the correlation coefficient (2.27), the PSNR (2.24), and the mutual information (2.28). The results are very similar (maximum error of 0.1 dB between restored images).

Acceptance rate of MH algorithm

In this section, we evaluate the sensitivity of the parameter β controlling the acceptance rate of the MH procedure (combined with BM4D). In Fig. 2.3 (left), it is confirmed that the acceptance rate affects the convergence speed. Nevertheless, whatever the parameter β chosen in the range $[1.5 - 4.0] \times 10^{-5}$, the algorithm provides solutions with a similar PSNR value close to 34 dB. Note that we got similar reconstructed images by uniformly aggregating all the samples or by aggregating samples with weights equal to the exponential form of the data fidelity term.

In theory, the recommended acceptation rate is about 0.234 [Breyer and Roberts, 2000] in the MH algorithm if we consider a Gaussian proposal distribution. In our case, we get faster convergence since the maximum acceptation rate is about 70%, suggesting that the set of proposed samples are relevant.

In what follows, we set $\beta = 4.0 \times 10^{-5}$ since it provides faster convergence as shown in Fig. 2.3 (left).

Data-fidelity terms

We have tested several data-fidelity terms $D(\boldsymbol{x}, \boldsymbol{y})$, corresponding to PSNR (see (2.24)), L_1 and L_2 norms defined as

$$D_{L_1}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{s \in S} |x(s) - y(s)|, \tag{2.25}$$

$$D_{L_2}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{s \in S} (x(s) - y(s))^2, \tag{2.26}$$

and the Pearson's correlation coefficient (CC) :

$$D_{\mathsf{CC}}(\boldsymbol{x}, \boldsymbol{y}) = \frac{\sum_{s \in S} (x(s) - \mu_{\boldsymbol{x}})(y(s) - \mu_{\boldsymbol{y}})}{\sqrt{\sum_{s \in S} (x(s) - \mu_{\boldsymbol{x}})^2} \sqrt{\sum_{s \in S} (y(s) - \mu_{\boldsymbol{y}})^2}}, \tag{2.27}$$

where $\mu_{\boldsymbol{x}}$ and $\mu_{\boldsymbol{y}}$ are the means of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. We have also evaluated a data-fidelity term based and the mutual information (MI) :

$$D_{\mathsf{MI}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i,j} p_{\boldsymbol{xy}}(i, j) \log \frac{p_{\boldsymbol{xy}}(i, j)}{p_{\boldsymbol{x}}(i) p_{\boldsymbol{y}}(j)}, \tag{2.28}$$

where $p_{\boldsymbol{xy}}(i, j)$ is the joint probability function of $\boldsymbol{x}$ and $\boldsymbol{y}$ with intensity bins i and j, and $p_{\boldsymbol{x}}(i)$ and $p_{\boldsymbol{y}}(j)$ denote the marginal probability distribution functions of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. In our implementation, we approximate the probability functions by histograms of pixel values.

It turns out that the resulting images are very similar for all these data-fidelity terms (see Fig. 2.3 (right)). We observed a maximum error of 0.1 db between the final restored images. In the sequel, we decided to focus on the D_{L_2} data-fidelity term to evaluate the components of the MH algorithm.

Spectral analysis of MW

The MW has different shapes if we consider different reciprocal spaces as illustrated in Fig. 2.4. In our framework, any spectral transform, provided that the transform allows us to decompose the spectral domain into connected components, that is two regions corresponding to non-zero and zero coefficients. The wavelet transform is typically not appropriate as shown in Fig. 2.4. The MW region should be as small as possible to make restoration successful. Accordingly, we investigated several spectral transforms to improve image restoration with our approach (see Fig. 2.5). The best result is achieved with the discrete fast Fourier transform (FFT), followed very closely by the discrete cosine transform (DCT). The pseudo-polar fast Fourier transform (PP-FFT), already considered in cryo-ET [Miao et al., 2005], achieves a worse result, both visually and in terms of PSNR values. Finally, it turns out that the performance of our method is not impacted by the transform type, but is actually influenced by the potential precision of the inverse transform. When evaluating the implementations of the considered three transforms, the resulting mean squared errors are in the range of 10^{-34} for DFT, 10^{-32} for DCT and 10^{-12} for PP-DFT. These errors perfectly correlate with the performance given in Fig. 2.5. Therefore, similar results could be achieved with another transform, provided that the inverse transform is precise enough.

Comparing MAP and MMSE estimators

Both the MAP and MMSE have been used considered for solving image restoration problems. For instance, in [Louchet and Moisan, 2008, Louchet and Moisan, 2013], the authors compared TV-MAP and TV-LSE and consider the TV norm as a prior to encourage piece-wise constant images. TV-MAP is more

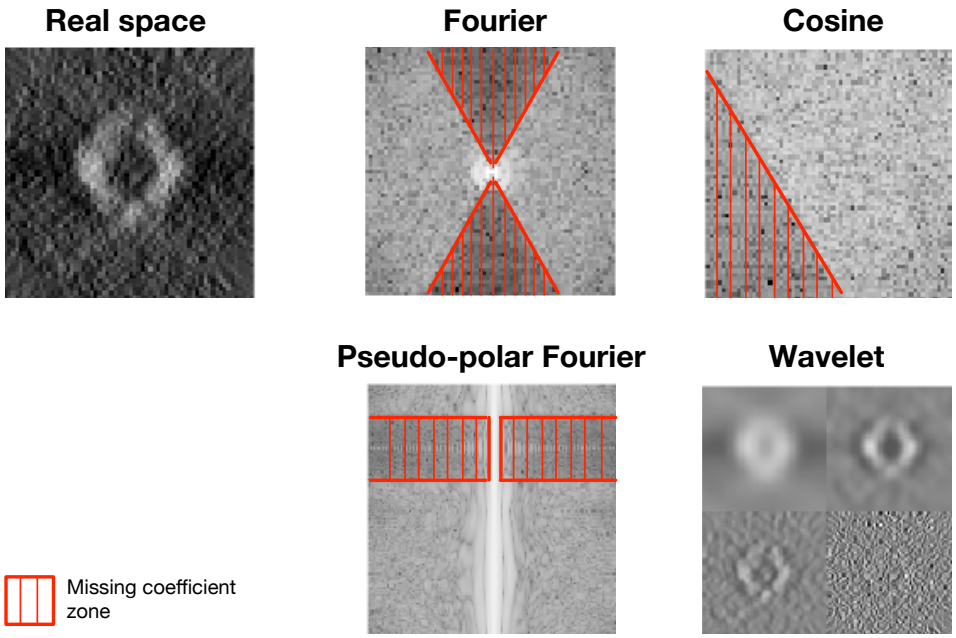

FIGURE 2.4 – The missing wedge shape (in red) for different transforms : Fourier transform, cosine transform, and pseudo-polar Fourier transform. Note that the missing wedge is not apparent in all transforms, as is illustrated with the wavelet transform.

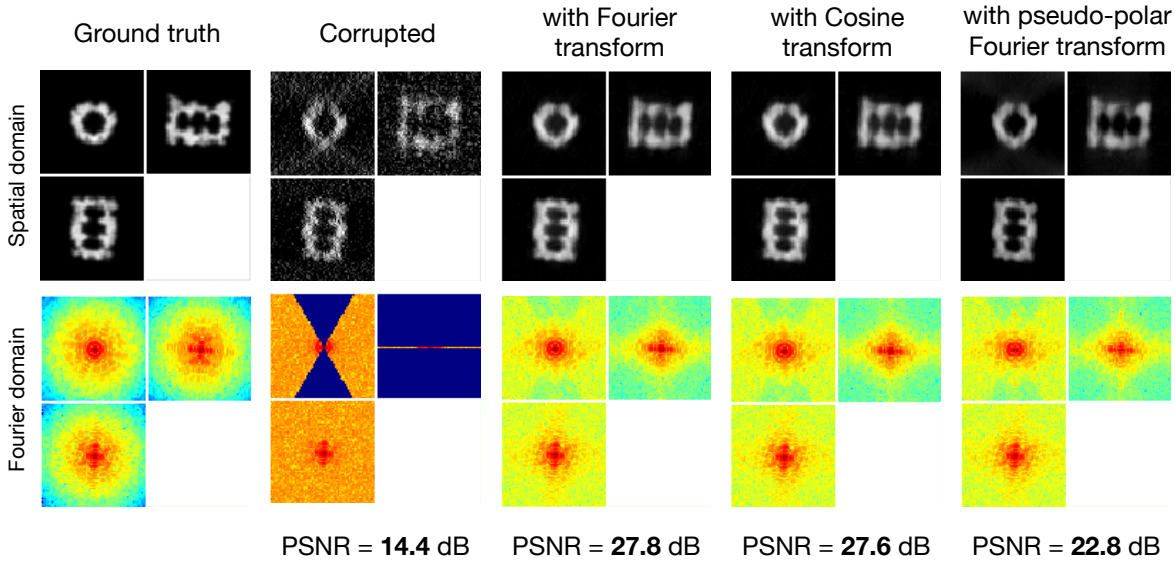

FIGURE 2.5 – Dataset A (3D) : influence of the transform type for MW restoration. From left to right : the ground truth (reference for measuring the PSNR), the corrupted image (used as input for the method), and the processed images using the Fourier transform, cosine transform and pseudo-polar Fourier transform. The best result is obtained by using the Fourier transform.

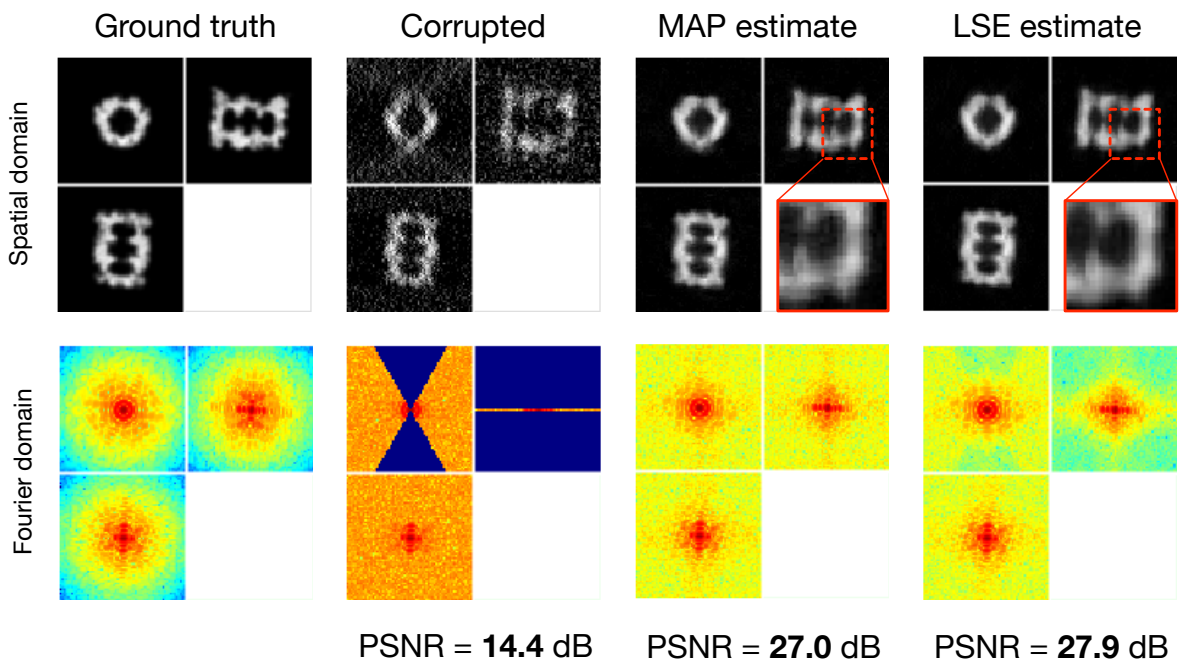

FIGURE 2.6 – Dataset A (3D) : comparison of MAP and MMSE estimators. From left to right : the ground truth (reference for PSNR), the corrupted image, the MAP and MMSE estimators computed with our MCMC procedure. As can be observed in the zoomed in regions (the red frames), the MMSE estimator is less noisy than the MAP estimate, with a higher PSNR value.

appropriate for restoring piecewise constant images (e.g. cartoons, shepp-Logan phantom), but washes out textures present in natural images and in microscopy images. To reduce stair-casing artifacts produced by TV-MAP, [Louchet and Moisan, 2013] proposed the TV-LSE estimator to favor smooth transitions between contrasted regions instead of sharp ones.

In our modeling framework, we do not directly use TV as in [Louchet and Moisan, 2008, Louchet and Moisan, 2013]. The TV constraint is actually used to a priori discard irrelevant samples during the proposal step in the MCMC sampling procedure. Nevertheless, we can compare the performance MMSE estimator to the MAP estimator corresponding to the most frequent sample in the MCMC sequence. At first glance, both estimates look visually similar, even though the MAP estimate produces more background noise (see Fig. 2.6). The difference is more noticeable in the spectral domain : the MW of the MAP estimate contains higher amplitudes in the high frequencies. However, this does not mean that the MW restoration is of better quality. Indeed, according to the PSNR values, MMSE is closer to the ground-truth than any generated sample. Therefore the higher MW amplitudes in the MAP estimate most likely carry noise rather than meaningful information.

Robustness to noise, comparison to BM4D [Maggioni et al., 2013]

From the results on dataset A (see Fig. 2.7(a)), it can be seen how well our method works in the absence of noise ($\sigma_n = 0$) : a quasi perfect image recovery has been achieved, despite the complexity of the

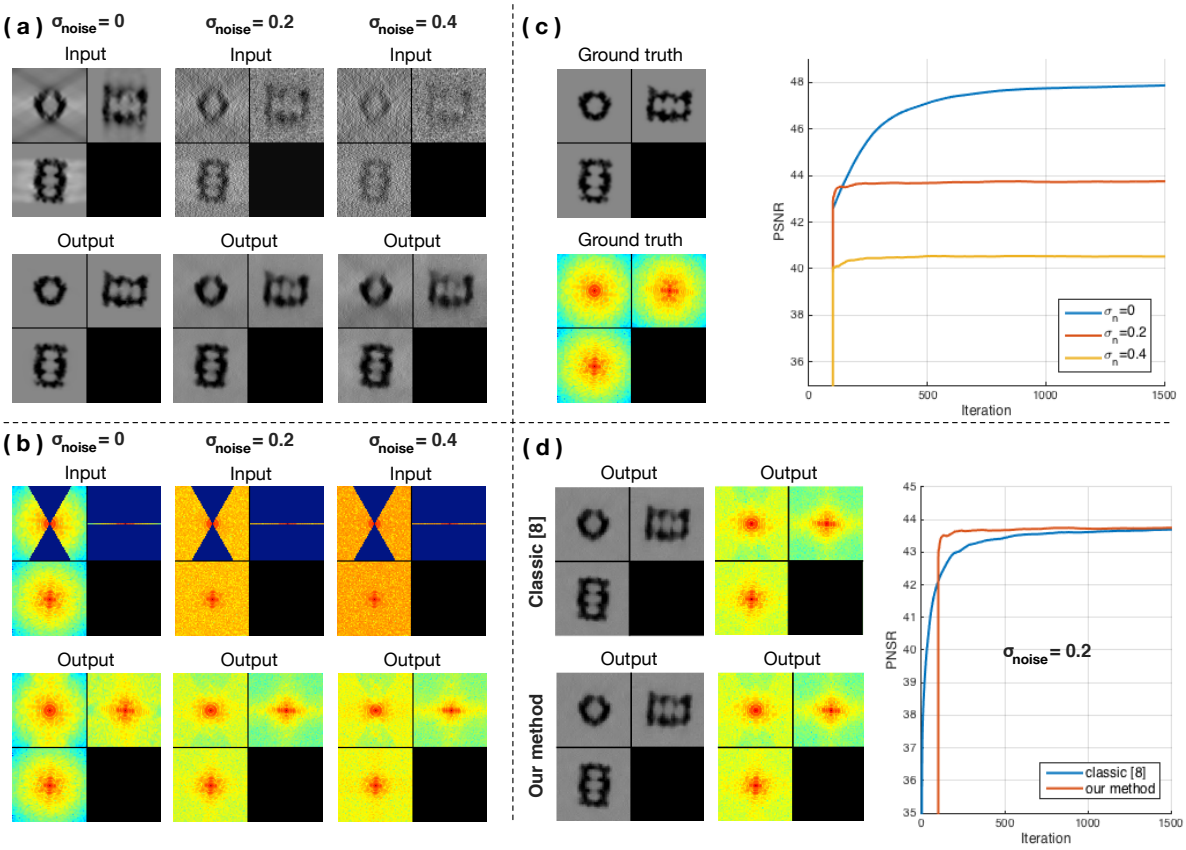

FIGURE 2.7 – Simulated data of the 20S proteasome, for varying amounts of noise (dataset A). All images depict ortho-slices of 3D volumes. The volume size is $64 \times 64 \times 64$ voxels. For (a) and (b), top row : method inputs, bottom row : method outputs. Results are shown in spatial domain (a) and spectral domain (b). In (c) can be observed the ground-truth and the increase of the PSNR values over iterations. In (d) we compare our method to the original method [Maggioni et al., 2013].

object. Increasing the amount of noise deteriorates the performance, but as it can be observed for $\sigma_n = 0.2$, the result is still satisfying. For high amounts of noise ($\sigma_n = 0.4$), the object contrast is still greatly enhanced but the MW artifacts could not be completely removed. In Fourier domain (Fig. 2.7(b)), the MW has been filled up completely when $\sigma_n = 0$, whereas for an increasing amount of noise the MW reconstruction is increasingly restrained to the low frequencies. This is because high frequency components of a signal are more affected by noise, which makes them more difficult to recover. In Fig. 2.7(c), the evolution of the PSNR over time shows that the method converges to a stable solution.

In Fig. 2.7(d), we compare our MH method to the method proposed by [Maggioni et al., 2013]. Both methods produce visually identical results in spatial domain, as well as in the spectral domain, as confirmed by the final PSNR values. However, the difference lies in the convergence speed : our method takes about half as long as the original method [Maggioni et al., 2013]. Even though the synthetic dataset A is a simplified case of data corruption in cryo-ET, it gives a good intuition of the performance of our method.

Comparison to other MW restoration methods

We compare our results to those produced by three competing methods (see Fig. 2.8) : sMAPEM [Paavolainen et al., 2014], BFLY [Kováčik et al., 2014], and a TV method with spectral constraints [Moisan, 2001], each adopting a different strategy to reduce MW artifacts. We implemented the Moisan's method as follows :

Our implementation of the [Moisan, 2001]'s method :

Set an initial state $\mathbf{x}^{(0)} \in \Gamma$.
For $t = 1, ..., T$ **do**

1. **Projection :** $\boldsymbol{z}^{(t)} = \mathcal{P}_W(\boldsymbol{x}^{(t)})$
2. **Denoising :** $\min_{\boldsymbol{x}} \|\boldsymbol{z}^{(t)} - \boldsymbol{x}\|^2 + \lambda\|\nabla \boldsymbol{x}\|_1$

end for

which amounts to alternatively minimizing TV [Rudin et al., 1992] and satisfying the spectral constraints. The sMAPEM algorithm [Paavolainen et al., 2014] is an iterative tomogram reconstruction procedure, designed to reduce MW artifacts and achieve isotropic resolution. In our experiments, we performed the comparison in 2D mainly because sMAPEM is not available in 3D. Our method, BFLY and the TV method operate on tomograms (2D in our case), while sMAPEM operates on projections (1D). These projections were obtained from the dataset A ground-truth, with a tilt-range of -60^o to 60^o and a tilt increment of 2 degrees. We then added Gaussian noise ($\sigma_n = 0.17$) to the projections. In order to make a fair comparison, the inputs should originate from these same projections. Therefore we use the weighted back-projection (WBP [Radermacher, 1992]) algorithm to produce the 2D data needed for the other methods.

As explained above, the strategy of sMAPEM differs from ours, in the sense that it takes as an input projections and gives as an output a reconstructed tomogram. One may argue that the best strategy would be the one adopted by sMAPEM, because it directly uses the projections, instead of a tomogram which is already contaminated by MW artifacts. However, as shown in Fig. 2.8, our method achieves a better PSNR value than sMAPEM. Visually both methods seem to approach isotropic resolution, however the result of our method is smoother, while the result of sMAPEM contains pixelated artifacts. The result of the Moisan's method is visually satisfying. In real space, noise appears to be removed, however the result suffers from staircasing artifacts, as well known with TV denoising. In Fourier space, the entire MW appears to be restored, as opposed to our method and sMAPEM where the restoration is constrained to lower frequencies. However according to CCC values, these restored Fourier coefficient do not correlate as well with the ground truth as our method and sMAPEM. The BFLY filter aims at reducing the MW ray-artifacts by smoothing the sharp transition between the MW and the SR. The object elongation and side artifacts however remain. Unlike other competing methods, BFLY does not recover missing Fourier coefficients, it is therefore not surprising that it has the worst performance, both visually and in terms of PSNR and CCC values.

In summary, our approach outperforms competing methods both in terms of PSNR and CCC values.

Visually, our approach produces a smoother image, while sMAPEM and the Moisan's method introduce artifacts in the result. The weakest performance is achieved by BFLY, however this was expected as the other methods have much higher complexity. That being said, BFLY is the fastest method.

2.6.2 Results on experimental data

In this section, we evaluate the performance of our MH method (combined with BM4D) on real images to confirm the results obtained on artificial images.

Experimental datasets

Three datasets (B, C and D) have been used to evaluate the performance of the method on experimental data. Dataset B is an experimental sub-tomogram containing a gold particle. Dataset C is an experimental sub-tomogram containing 80S ribosomes attached to a membrane. Dataset D is an experimental sub-tomogram depicting a region of an E. Coli bacteria, and contains unidentified macromolecules next to a membrane. Unlike data-sets B and C, the dataset D is available as single-axis and double-axis data (see Section 2.1).

Criteria and method for evaluation

The evaluation differs depending on the dataset. In dataset B, the gold particle is deformed and elongated (ellipse) due to the MW artifacts. Improving the sphericity of the object is thus a good evaluation criterion. For dataset C, we measure the similarity between the central ribosome and a reference (obtained via sub-tomogram averaging). The evaluation criterion is the Fourier shell correlation (FSC), commonly used in cryo-ET [Van Heel and Schatz, 2005]. In order to measure the quality of the recovered MW only, we also compute the FSC over the MW support ("constrained" FSC or cFSC). For dataset D, we have both single-axis and double-axis versions of the data. A double-axis volume has less missing Fourier coefficients than a single-axis volume. Therefore, when processing the single-axis volume, the additional Fourier coefficients of the double-axis volume can act as an experimental ground truth. We evaluate the result with the CCC score, as illustrated in Fig. 2.12.

Analysis of restoration results

The result on dataset B shows that noise is reduced and a significant part of the MW was recovered (see Fig. 2.9). Even though the recovery is not complete, it is enough to reduce the MW artifacts while preserving and enhancing image details. The ray and side artifacts induced by the high contrast of the gold particle are reduced and its sphericity has been improved, bringing the image closer to the expected object shape. The result on this dataset shows that the method is able to handle experimental noise in cryo-ET.

The dataset C (ribosomes, Fig. 2.10) is more challenging because the objects have a more complex shape and less contrast, i.e. the SNR is lower. Nonetheless, the method significantly enhanced the contrast and, according to the FSC criteria, the signal has indeed been improved. Although visually it

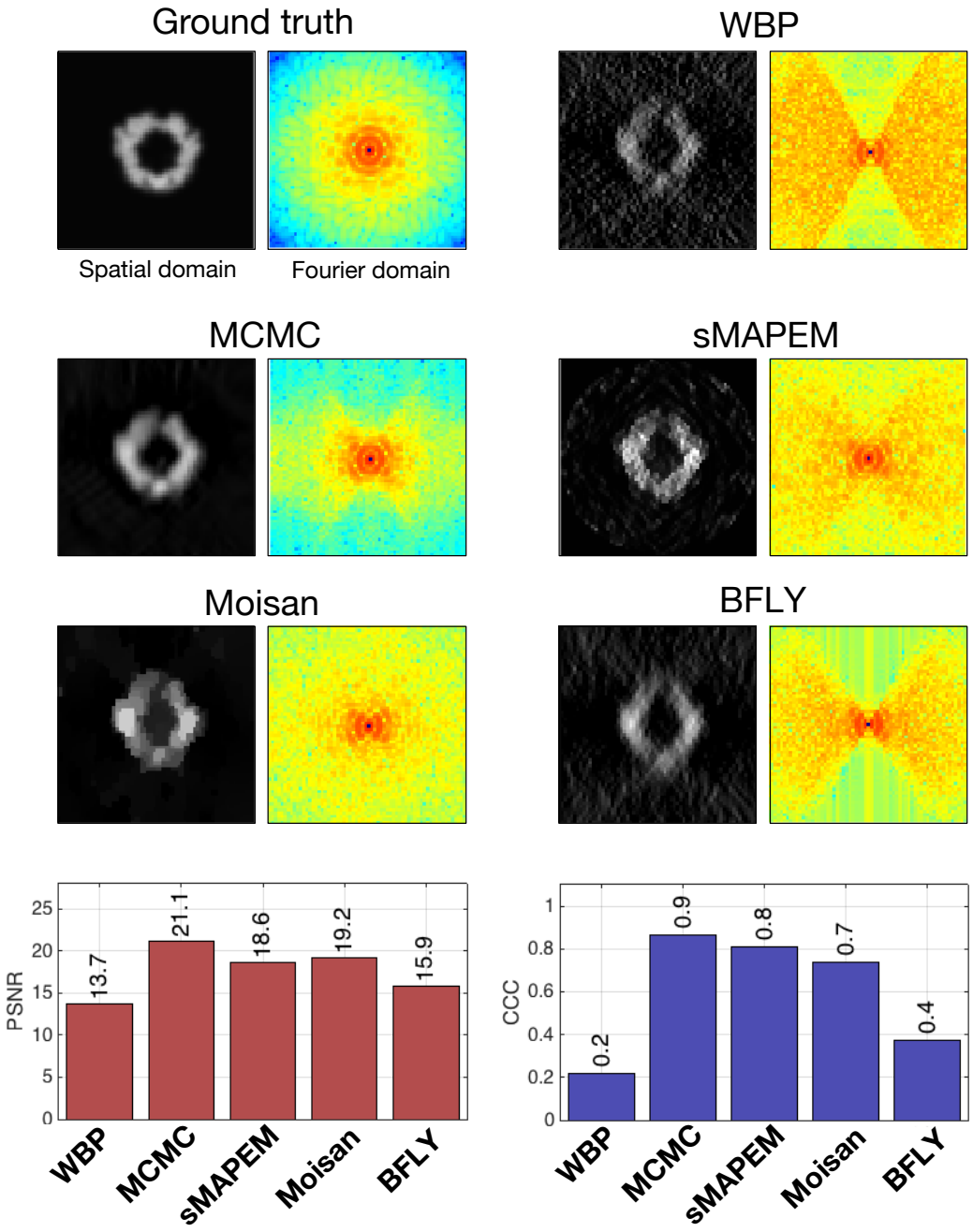

FIGURE 2.8 – Dataset A (2D) : comparing our approach to competitive methods : i/ sMAPEM, a regularized tomographic reconstruction method designed to achieve isotropic resolution ; ii/ the Moisan's method designed to extrapolate missing regions in Fourier space ; iii/ BFLY, a filter designed to reduce MW artifacts. The sMAPEM method takes projections as input, therefore we used the same projections to produce the 2D input (via WBP) for the other methods. On the bottom we display the PSNR and CCC scores obtained for all tested methods.

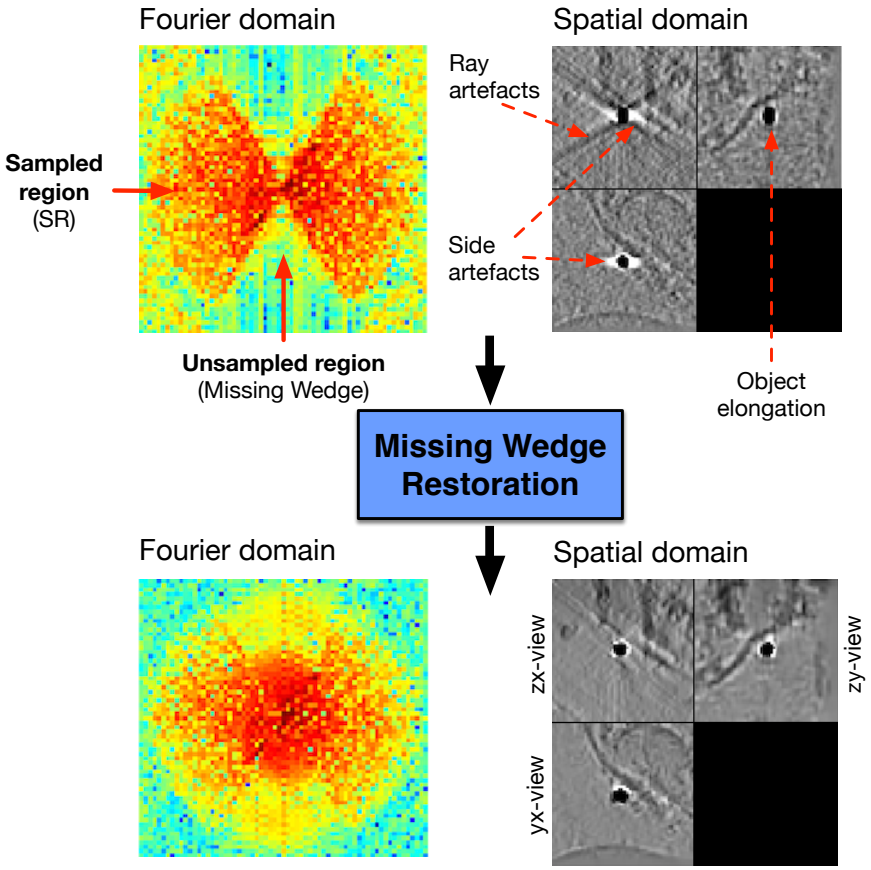

FIGURE 2.9 – Experimental sub-tomogram ($61 \times 61 \times 61$ voxels) containing a gold particle (dataset B). The top row shows the input in the spectral and spatial domains, the bottom row shows the restored image and spectrum.

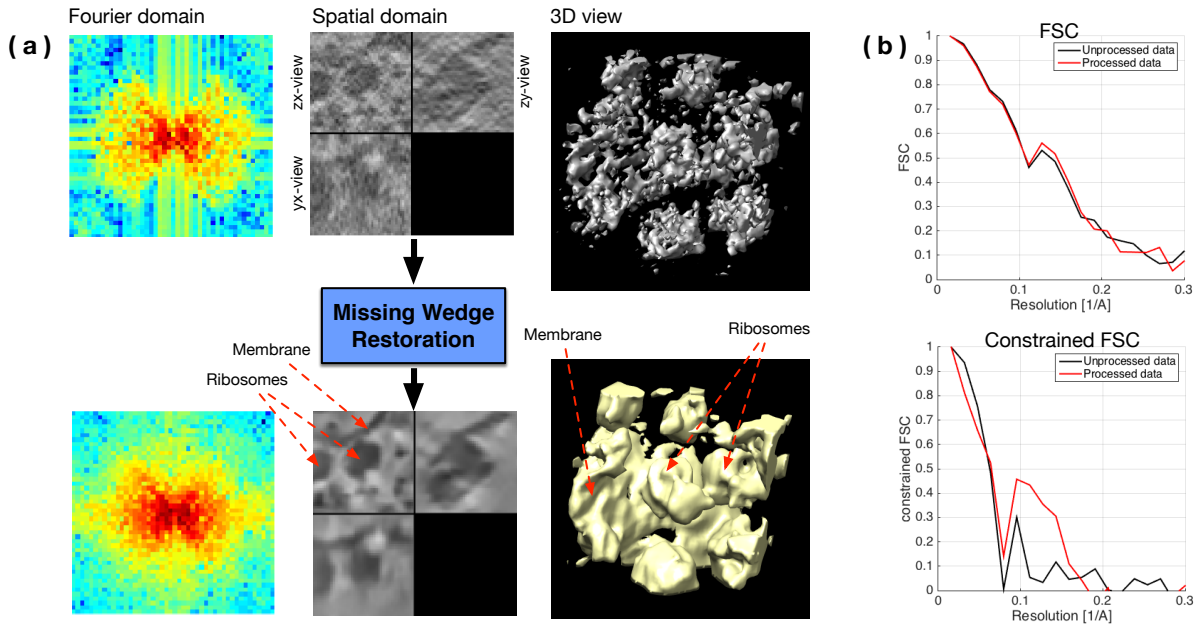

FIGURE 2.10 – Experimental sub-tomogram ($46 \times 46 \times 46$ voxels) containing ribosomes attached to a membrane (dataset C). (a) Top row : input image in spectral domain, spatial domain and 3D view of the thresholded data. Bottom row : the same representations for the output. (b) FSC and cFSC measures of the method input (in black) and output (in red). The FSC measures overall quality, while the cFSC measures quality of recovered Fourier coefficients only (i.e. MW). All measures are wrt the same reference.

is more difficult to conclude that the MW artifacts have been affected, the Fourier spectrum shows that Fourier coefficients were recovered. As expected, the amount of recovered high frequencies is less than for dataset B, because of the lower SNR. It is now necessary to provide a proof that the recovered coefficients carry a coherent signal, therefore the cFSC has been measured. The black curve in Fig. 2.10 depicts the cFSC between the unprocessed image and the reference : given that the MW contains no information, the curve represents noise correlation. Consequently, everything above the black curve is signal, which is indeed the case for the processed data (red curve in Fig. 2.10). To illustrate how the method can improve visualization, a simple thresholding has been performed on the data (3D views in Fig. 2.10). While it is difficult to distinguish objects in the unprocessed data, the shape of ribosomes become clearly visible and it can be observed how they are attached to the membrane. In addition, in order to demonstrate that our procedure is not limited to ribosomes (considered as easy targets because of their good contrast), we perform the same evaluation procedure (i.e. dataset C) on subtomograms containing proteasomes (see Fig. 2.11). In both processed proteasome sub-tomograms, the contrast has been enhanced and the FSC and cFSC curves provide proof of an improved signal.

Dataset D has a broader field of view than precedent datasets. As can be observed in Fig. 2.12, the double-axis volume has a better contrast than the single-axis volume, the reason being that it has more sampled Fourier coefficients, and therefore contains more signal and less noise. For this dataset, we processed the single-axis volume and investigated how close to the double-axis volume the result is. From Fig. 2.12, our method indeed enhanced the contrast, membrane and unknown macromolecules

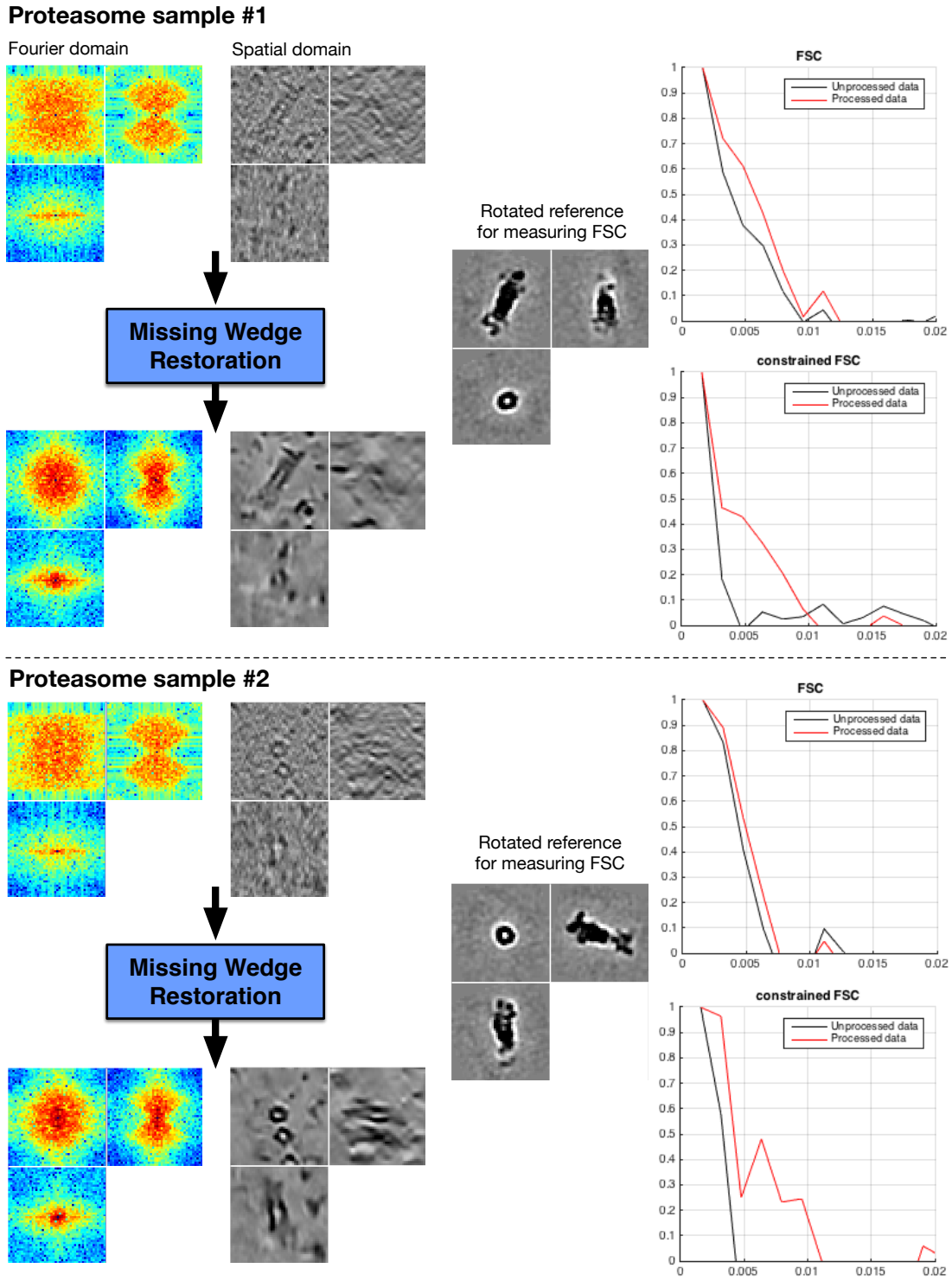

FIGURE 2.11 – Two experimental sub-tomograms ($46 \times 46 \times 46$ voxels) containing proteasomes. Data is displayed in both Fourier and spatial domains. We evaluate the result with FSC and cFSC measures of the method input (in black) and output (in red). The FSC measures overall quality, while the cFSC measures quality of recovered Fourier coefficients only (i.e. MW). The reference has been obtained via subtomogram averaging of 2949 proteasomes.

became easier to identify. When examining the data in Fourier domain, it is clear that that several Fourier coefficients have been partially recovered, and correlate better with the double-axis volume as confirmed by the CCC values (0.29 before and 0.44 after applying our MCMC method).

Software In terms of computational performance, MWR takes 5 min and 30 seconds (0.66 sec / iteration, $T = 500$ samples) on a standard volume of $64 \times 64 \times 64$ voxels on a Macbook Pro equipped with 2.9 Ghz Intel Core i7, 16 Gb of RAM and the Mac OS X v.10.12.3 operating system. The computing time increases linearly with the number of voxels. The MRW software (Matlab code) can be downloaded from the Git-Hub website : `https://gitlab.inria.fr/serpico/mwr`

2.7 Conclusion

In this Chapter, we addressed the problem of restoring an image corrupted by missing Fourier coefficients in a well-localized spectral region (missing wedge). We proposed an original Monte-Carlo method to denoise 3D cryo-ET images and compensate MW artifacts. Our algorithm cannot recover unobserved data, but it merely makes the best statistical guess of what the missing data could be, based on what has been observed. Any non-local or patch-based denoiser can be used in our Bayesian framework, and the procedure converges faster than [Maggioni et al., 2013]. Our experiments on both synthetic and experimental cryo-ET data demonstrate that even for high amounts of noise, the method is able to enhance the signal. The method performs better if the contrast of the object of interest is high, which is not always the case in cryo-ET.

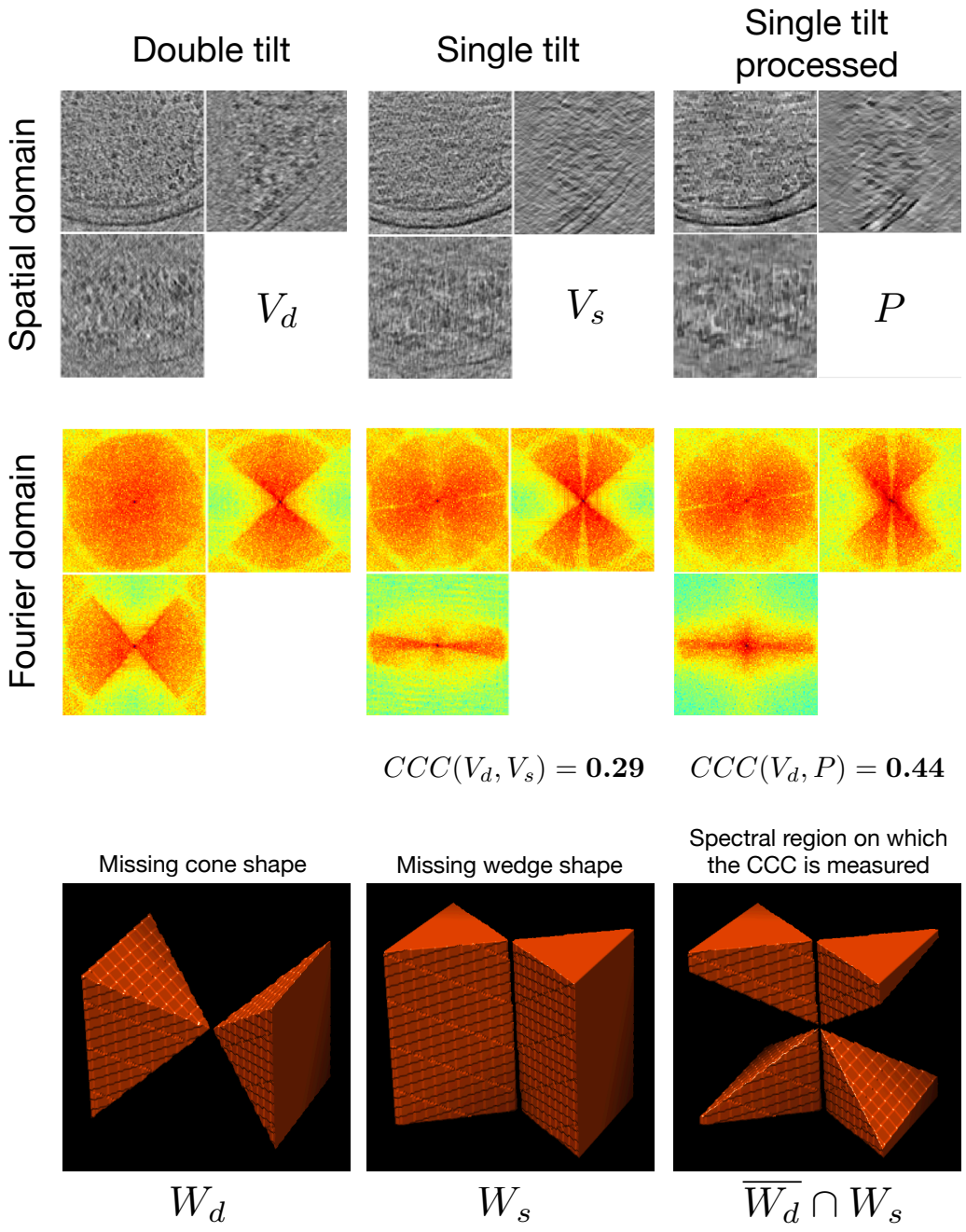

FIGURE 2.12 – Dataset D : Experimental double-axis sub-tomogram ($128 \times 128 \times 128$ voxels) containing multiple macromolecules (see black dots). We process the single-axis version of the volume (top middle) and compare to the double-axis volume (top left), which acts as a ground truth. We evaluate the processed volume (top right) by computing the CCC scores, as illustrated in the bottom right image. All volumes are displayed in spatial domain (top row) and Fourier domain (second row). The regions W_{DT} and W_{ST} (bottom row) illustrate the shape of the missing Fourier region for double-axis and single-axis data resepctively.

DEUXIÈME PARTIE

Image analysis and deep learning for macromolecule localization

CHAPITRE 3

AN INTRODUCTION TO DEEP LEARNING

3.1 Introduction to machine learning

Machine learning is a field of computer science that studies algorithms that "learn" how to execute a task by being exposed to examples, without having to explicitly program said task. What learning means for an algorithm has been well defined in [Mitchell, 1997] : "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E".

In summary, machine learning tasks can be grouped in two main categories :

— **Supervised learning** : This task consists in learning to map input data to an output *target* (also called *annotation* or *label*), given a set of input and output examples (i.e. the *training set*). Once the learning (training) is complete, the algorithm can *predict* the output for unseen inputs.

— **Unsupervised learning** : In this setting, the targets are not available and only the input examples are given. The algorithm then tries to discover structure or patterns in the data. Dimensionality reduction (e.g. principal component analysis) and clustering (e.g. k-means, mean-shift...) are well known categories of unsupervised learning.

Formally, a machine learning algorithm is a function $f : \mathcal{X} \to \mathcal{Y}$, mapping input data $x \in \mathcal{X}$ to an output space $\mathcal{Y}$. Another categorization of machine learning tasks arises when considering the nature of the output space $\mathcal{Y}$. If $\mathcal{Y}$ is defined in the continuous setting, the task is being referred to as a *regression* task, which is supervised. If $\mathcal{Y}$ is defined in the discrete setting, then it is a *classification* task, which can be supervised or unsupervised.

In order to learn a task, the distribution of the data in $\mathcal{X}$ needs to be structured, meaning that discernible distribution modes are necessary. For example in classification, it is desirable that data points belonging to different classes populate different regions in $\mathcal{X}$, so that these data points are *separable*. As such, classification (as well as regression) consists in charting $\mathcal{X}$; the algorithm then produces an output y according to which region of $\mathcal{X}$ the input x belongs to (see Fig. 3.1).

However, data is rarely separable in its raw form, or *representation*. A representation is a way to describe data, for example the common representation of a color image is the RGB format (red-green-blue). But if the task at hand is to decrease the color saturation, then a better representation would be the HSV format (hue-saturation-value). This example illustrates an important problematic in machine learning : find an adapted data representation for a given task. In other words, finding a way to transform the data into a useful representation, in which data points are separable. The information elements of a representation are known as *features*, as such a data representation can be called a *feature vector* or a

feature map.

As a result, in the field of machine learning considerable effort has been put into finding useful representations for a given data type and a given task. For instance in image analysis, performant features such as HOG [Dalal and Triggs, 2005] (histogram of gradients) and SIFT [Lowe, 2004] (scale-invariant feature transform) have been developed. However designing features manually is time consuming and a given feature does not perform well for all applications. To overcome this difficulty, a solution is to learn not only the mapping from representation X to output Y, but also the representation itself. This approach is known as *representation learning* (or feature learning). Learned representations often perform better than hand-crafted ones, and allow to reduce human intervention. Representation learning includes techniques such as principal component analysis, dictionary learning and neural networks.

Even so, the expressivity of hand-crafted or learned representations is limited. While it can be sufficient for quite complex tasks such as detecting cars and pedestrians in images, limitation start to show when the tasks involve characterizing semantic knowledge, i.e. human-level decision making, such as identifying emotions on a face or recognizing a particular accent in spoken language. Such representations are simply not powerful enough to encode all the variability of such high-level classes. *Deep learning* (DL) solves this problem by decomposing the representation : it builds *hierarchical representations*, in other words it characterizes complex high-level features (e.g. cars, persons) out of simpler low-level features (e.g. corners, contours). Deep learning is a supervised machine learning method, based on neural networks (hence the popular term "deep neural networks"). It is able to learn both a hierarchical representation and the task (e.g. regression, classification, segmentation...) in a *end-to-end* manner, i.e. in one single learning (or training) procedure. The term "deep" is a reference to the high number of layers in the neural networks, needed to achieve the hierarchical representation. The layers transform the input data by applying geometric deformation to the input representation (e.g. rotations, projections, contractions...). Applied sequentially, these layers progressively transform the data and are finally able to encode high-level concepts.

As mentioned above, deep learning is a supervised learning method, meaning that they are always targets (i.e. annotations) involved in the training procedure. However, it is able to perform *self-supervised learning*, which is a particular learning category where the data is not annotated by humans. Instead, the targets are generated from the data by using some ad-hoc heuristics. For instance, *auto-encoders* [Baldi, 2012] are a well-known DL example of self-supervised learning, for which the targets are the inputs. Here the goal is not to classify data, but to find useful and reversible representations. The fact that no humans are involved in the target generation is the reason why deep neural networks are considered to be able to achieve unsupervised learning.

In the end, the following elements are necessary to deploy a deep neural net framework :

— Input data points x.
— Targets t : examples of the expected output.
— Loss function $\mathcal{L}$: needed to measure the performance of the network and used to guide the training procedure.

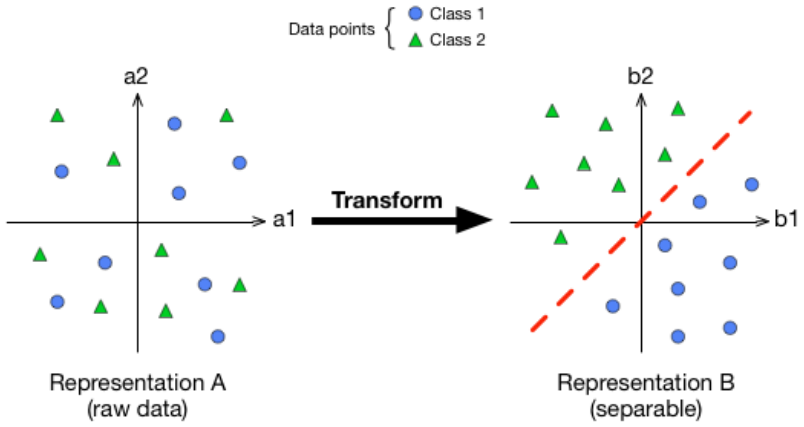

FIGURE 3.1 – Representation

3.2 Neural networks

This section describes the framework of neural networks (Section 3.2.1) and related concepts (Section 3.2.2), and explains why neural networks are so well adapted for solving any machine learning task (Section 3.2.3).

3.2.1 A brief history

In october 2012, a deep neural network algorithm won the large-scale ImageNet competition (1.2 million images, 1000 classes) and outperformed significantly the competing methods [Krizhevsky et al., 2012]. In the machine learning and computer vision fields, this milestone marks the beginning of the "deep learning revolution". Since then, DL algorithms constitute the state-of-the-art for many applications, such as image analysis [Krizhevsky et al., 2012] [Long et al., 2014] [Szegedy et al., 2013] and language processing [Hinton et al., 2012]. However, the concepts of this technology have been established a long time ago. The first functional artificial neural network with learnable weights, the perceptron, dates from 1957 [Rosenblatt, 1957] (earlier models have been published in the 40s [McCulloch and Pitts, 1943]). The backpropagation algorithm (used to train deep neural networks) has first been implemented in 1970 [Griewank, 2012], and the first time it was applied to train a convolutional neural network was in 1989 [Lecun et al., 1989]. The reason why it is only recently that DL is so successful is that we now have access to powerful computing units (GPUs) and to vast amounts of data (the internet). It was only after bringing these two essential components together that algorithmic advances were made (e.g. better activation functions, better optimization schemes), enabling current DL performance.

3.2.2 Definition

A neuron has n inputs, which are combined to produce a single value output (see Fig. 3.2 (a)). It can be expressed by a function $n : \mathbb{R}^n \to \mathbb{R}$ as follows :

$$\phi(x) = \sigma(\sum_{i=1}^{n} w_i x_i + b)$$

where the neuron is parametrized by the weights w, the bias b, and a non-linear activation function σ. The purpose of σ is to saturate the neuron output to generate non-linearities. Several types of activation functions are proposed in the literature, among which ReLU is the most commonly used (see Figure 3.3).

A set of neurons can be arranged into a layer, which maps an input vector $x \in \mathbb{R}^m$ to an output vector $y \in \mathbb{R}^n$ (see Fig. 3.2 (b)). The input x is processed by all layer neurons, and the output y consists in the concatenation of all neuron outputs. When applied in a sequential manner, the layers form a neural network, where each neuron of layer ℓ feeds its ouput to the neurons of layer $\ell + 1$. A layer can thus be represented by a non-linear function $\varphi^\ell : \mathbb{R}^{n_{\ell-1}} \to \mathbb{R}^{n_\ell}$, where $n_{\ell-1}$ is the dimension of the layer input and n_ℓ the dimension of the layer output. The most basic layer type is the *fully-connected* layer, where all neurons of layer ℓ are connected to all neurons of layer $\ell + 1$.

A network of L consecutive layers is defined by its hyper-parameters and its parameters θ. The hyper-parameters define the network *architecture*, and consist in the number of layers (the network *depth*), the number of neuron per layer (the network *width*), and how the layers are connected (the layer type). The parameters θ are given by the individual weights w and biases b of all network neurons. The network thus represents a function $N(x; \theta)$ and can be written as a composition of its layers :

$$y = N(x; \theta) = \varphi^L(x) \circ \varphi^{L-1}(x) \circ \cdots \circ \varphi^2(x) \circ \varphi^1(x)$$

where y is the network output, which is referred to as the network *prediction*.

3.2.3 Why deep neural networks ?

A powerful property of neural networks is that they are *universal function approximators*. In 1989, the author of [Cybenko, 1989] shows that given any continuous function $f(x)$ and an error ϵ, there exists a neural network $N(x)$ with one hidden layer such that $\forall x, |N(x) - f(x)| < \epsilon$. In other words, a neural network N can approximate any function f at any desired accuracy. By increasing the number of neurons in the hidden layer, the accuracy increases. The function f can be any machine learning task, like translating spoken language, tracking objects in sequences or colorizing images. Note that the universality theorem simply states that there exists a neural network for any task, however it does not provide a way of finding such a network.

In theory a network with one single hidden layer is sufficient to achieve any task, provided that the hidden layer is wide enough (i.e. has enough neurons). Thus one may ask what is the rationale behind implementing deeper networks (i.e. with more layers). Actually, the principal reason is computational efficiency. In fact, an equivalent approximation can be achieved using a deeper network that has fewer

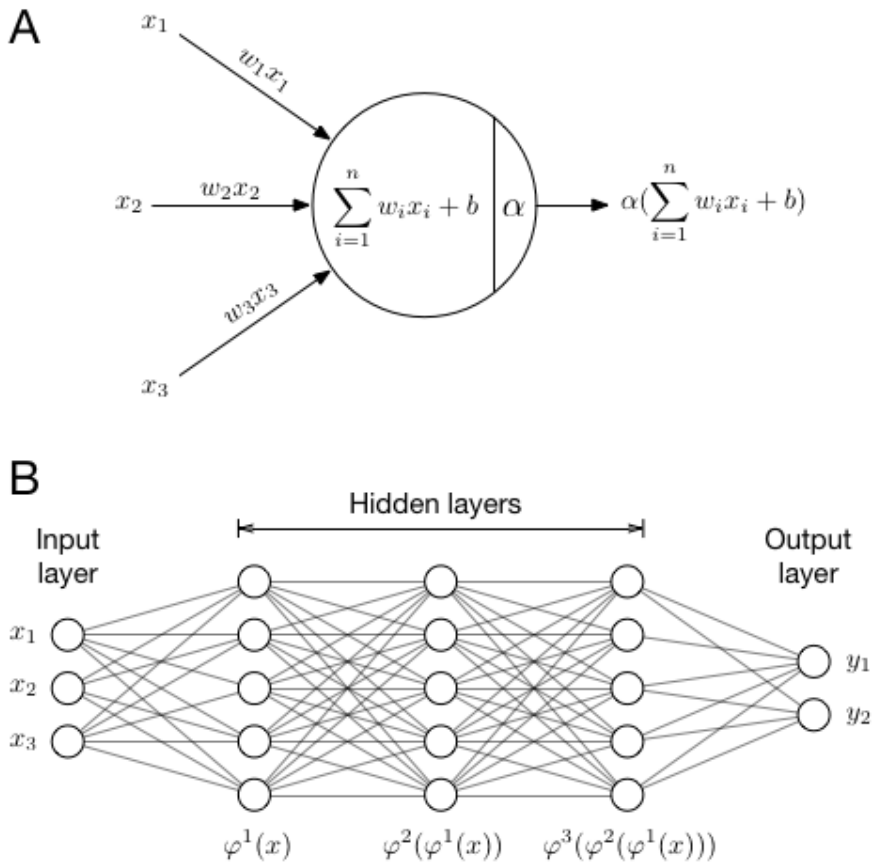

FIGURE 3.2 – (a) Artificial neuron, (b) Fully connected network

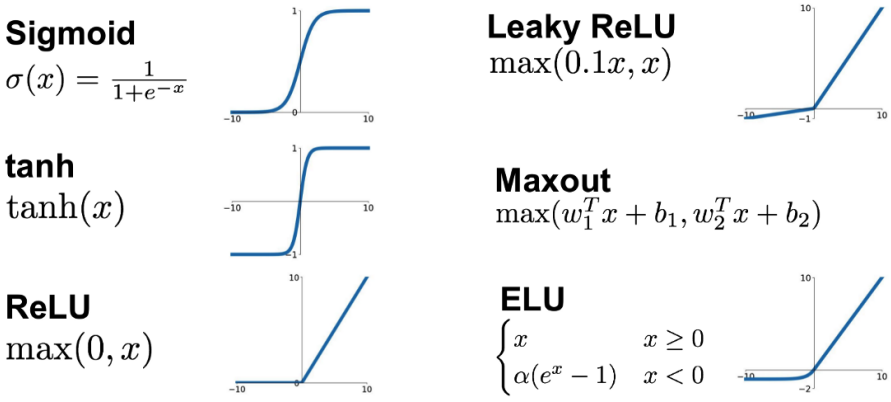

FIGURE 3.3 – Common activation functions and their graphs. The sigmoid and tanh functions are used in the output layer for classification and regression, respectively. ReLU and its derivatives are used in the hidden layers. Source : [medium.com].

total number of neurons. The optimization process has therefore less degrees of freedom, and training becomes easier and converges faster. In addition, a network with fewer neurons requires a smaller training set, which is appropriate since training sets are limited in size.

3.3 Training

This section gives an overview on how neural networks are trained. First the optimization procedure is described, then several common training issues are discussed.

3.3.1 Stochastic gradient descent

The set of parameters $\theta = \{\theta_i\}_{i=1,\dots,N}$ are optimized to minimize the objective function $\mathcal{L}$ over the training set $\mathcal{T}$. The objective function $\mathcal{L}$ is designed to measure the error between the network prediction y and the desired output t, known as *target*. In the end, the objective function depends on θ, y and t : $\mathcal{L}(\theta, y, t)$. All operations in the network, including the objective function (or loss function), are chosen to be differentiable, which means that the parameters θ can be optimized using stochastic gradient descent (SGD) [Robbins and Monro, 1951] [Goodfellow et al., 2016a]. The key idea of gradient descent is that the derivative, or gradient, of the objective $\mathcal{L}$ with respect to the network parameters θ, carries the information on how to update θ to minimize $\mathcal{L}$ (see Figure 3.4).

The SGD algorithm can be summarized as follows :

1. Draw a batch of training samples x and corresponding targets t.
2. Run the network on x to obtain predictions y.
3. Compute the loss $\mathcal{L}(\theta, y, t)$ on the batch.
4. Compute the gradient of the loss with respect to the network parameters θ (using backpropagation).
5. Update the parameters : $\theta_i \leftarrow \theta_i - \eta \frac{\delta \mathcal{L}(\theta, y, t)}{\delta \theta_i}$.
   where η is the so-called learning rate, which controls the update step size.

Data batch The stochastic term refers to the fact that training batches are drawn randomly. The batch loss is computed as the average loss over all (y, t) pairs contained in the batch. Therefore, larger batches provide more stable loss values (i.e. less variance, hence a smoother loss surface) and the training procedure converges faster. However, the final generalization ability of the network is often best when using smaller batches [Keskar et al., 2016]. Therefore the batch size needs to be tuned to find an appropriate trade-off.

Backpropagation The gradients are computed using *backpropagation* [Werbos, 1975] [Duda et al., 2000], an efficient method for computing gradients in directed graphs, such as neural networks. This

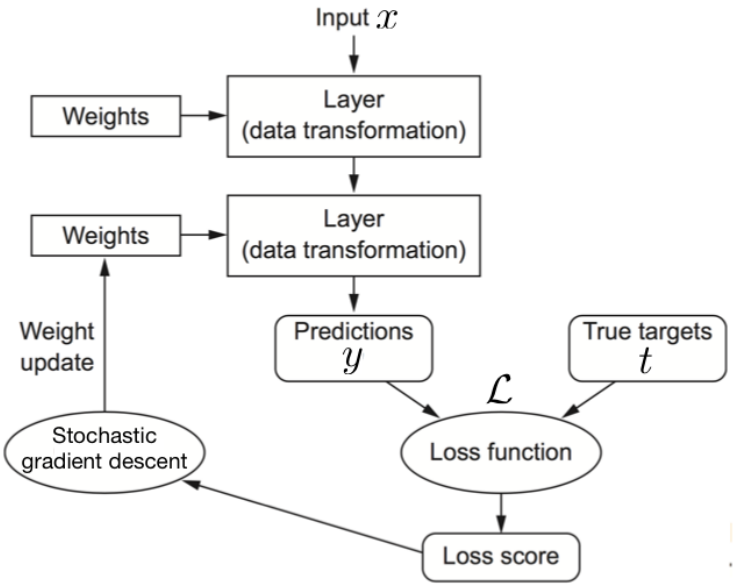

FIGURE 3.4 – Training a neural network : the loss score is used as a feedback signal to adjust the weights θ. Source : [Chollet, 2017]

method is a simple application of the chain rule for derivatives, starting from the network output and progressing backwards through the layers towards the network input. Backpropagation allows to compute all required partial derivatives in linear time with respect to the graph size.

Momentum While the basic SGD algorithm has been around for a long time [Robbins and Monro, 1951], several modern variants exist today, including Adagrad, RMSProp and Adam [Ruder, 2016]. A common extension in all these variants is the concept of *momentum* [Rumelhart et al., 1986] [Goodfellow et al., 2016a], which draws inspiration from physics. Intuitively, one can imagine the optimizer as a ball rolling down the loss surface. With momentum, the next step is computed not only according to the current slope value, but also according to the current "velocity" v. The velocity is estimated by taking into account past training iterations ; the parameters θ are updated not only according to the current gradient value, but also depend on previous updates. With momentum, the update rule becomes :

$$v \leftarrow \mu v - \eta \frac{\delta \mathcal{L}(\theta, y, t)}{\delta \theta_i},$$
$$\theta_i \leftarrow \theta_i + v.$$

The larger μ is relative to η, the more previous gradients affect the current update. Implementing the concept of momentum has two benefits : i/ convergence is faster, i.e. "the ball rolls faster down the slope" ; ii/ it reduces the risk of being trapped in local minima, i.e. "the ball does not get stuck in minor holes".

Local minima However, even when implementing momentum in SGD and its variants, there is no actual guaranty of converging to a global minimum. Also, due to the several layers of non-linearities of the network, the loss function is highly non-convex. It is therefore reasonable to wonder about the quality of these local minima. It has been suspected by many that these many local minima induce equivalent performance [Goodfellow et al., 2015]. Moreover, it has recently been mathematically proven [Kawaguchi, 2016] that, given some reasonable assumptions, local minima in deep neural networks are in fact all global minima.

3.3.2 Loss functions

Regression The most common loss function for regression is the mean squarred error (MSE, L_2 norm), defined as follows :

$$\mathcal{L}_{\text{MSE}}(y_i, t_i) = \frac{\sum_{i=1}^{n}(t_i - y_i)^2}{n}$$

with y the network prediction, t the target and n is the number of elements of the output array.

Classification In the classification regime, the targets are categorical variables known as *labels*. However, neural network outputs are numerical, therefore labels can not be used directly by the training procedure. Instead, the labels are converted into vectors containing class probabilities, ranging from 0 to 1 (known as "one-hot" encoding). The size of these vectors is equal to the number of classes that the network is expected to classify. So in the end, all tasks solved by neural network are actually regression tasks. An adapted and commonly used loss function for classification is the cross-entropy :

$$\mathcal{L}_{\text{CE}}(y_i, t_i) = -\sum_{i=1}^{n} t_i \log y_i$$

3.3.3 Data augmentation

The best way for increasing the generalization capacity of a network is to use more training data. However, in the vast majority of real-world problems, the amount of data is limited. It is therefore common to increase the training set size artificially, which is referred to as *data augmentation*. As such, transformations like translation, rotation and inversion are applied to the experimental data, in order to increase intra-class variability and to make the network invariant to these transformations. Other forms of data augmentation include histogram equalization (i.e. contrast change) or adding noise. Yet, it is important that applied transformations do not modify what characterizes the class. For example in the case of handwriting recognition, horizontal and vertical flips should be avoided, else it would not be possible to differentiate the letters 'b' and 'd' (or 'b' and 'p').

If a precise enough data model is available, it is possible to generate the training set in a completely synthetic way. Such scenarios are ideal for deep learning methods, as a virtually infinite amount of training data can be created. An example is optical flow estimation, where precise ground truth flow fields can not be obtained via manual annotation. In such scenarios, it is possible to learn from animated movies for which flow fields can be obtained automatically, the learned features are then robust enough

to be applied on real data [Mayer et al., 2018]. However, for image modalities having a complex image formation process, as is the case for cryo-ET, data models are not realistic enough an thus prediction on real data fails. Recent studies explore the possibility of learning the data model with generative adversarial networks [Mahapatra et al., 2018] [Jin et al., 2018].

3.3.4 Dealing with imbalanced classes

It is current that real data sets have an imbalanced class distribution, meaning that most data belongs to some majority classes, while the minority classes (which are often of greater interest) are under-represented. This causes the classification method to be skewed towards the over-represented classes. This is for example the case in medical imaging, where only few examples of sick subjects exist while a large number of healthy subjects is available. A number of techniques have been proposed to counter the negative effect of imbalanced classes, and an extensive review can be found in [He and Garcia, 2009]. These techniques can be grouped in two main categories. The first one is re-sampling (i.e. bootstrapping), which operates from the data perspective. It aims at balancing the class distribution by over-sampling the minority class or under-sampling the majority class, or both. While re-sampling is successful for many applications, one must be careful not to overuse it. Indeed, when over-sampling there is a risk of over-fitting, while with under-sampling the model can under-fit (because important information is missed). The second category is cost-sensitive learning, and acts from the algorithm perspective. Here, different miss-classification costs are assigned to the classes during training, which is achieved by weighting the loss-function :

$$\mathcal{L}_{\mathsf{W}}(y_c, t_c) = \sum_{c=1}^{C} w_c \mathcal{L}(y_c, t_c),$$

where c is the class index, C the number of classes, and w_c the class dependent weight. Thus, the cost of miss-classifying the under-represented classes is higher than for the over-represented classes. Here again one must be careful not to use disproportionate weight values, in fact a too large miss-classification cost can result in a high false positive rate during prediction.

3.3.5 Dealing with label noise

Experimental data sets most often contain errors in their annotations, due to the subjectivity of the human annotator. This is referred to as *label noise*, and negatively affects model training. Several strategies exist to counter its influence, the simplest one being label smoothing. Label smoothing consists in modifying the training targets to take into account the probability that a label may be incorrect. Instead of training the network with hard class probabilities of '0' and '1', these targets are replaced with relaxed values : $u/(N_{\text{class}} - 1)$ and $1 - u$, with u a small constant (encoding uncertainty), and N_{class} the number of classes. This approach is very easy to implement and is used in recent classification [Szegedy et al., 2015] and generative networks [Goodfellow, 2016]. Other strategies consists in designing corrected loss functions that are robust to *label noise* [Patrini et al., 2017]. Both label smoothing and loss correction need knowledge on the amount of label noise. While this quantity can be set empirically, it becomes

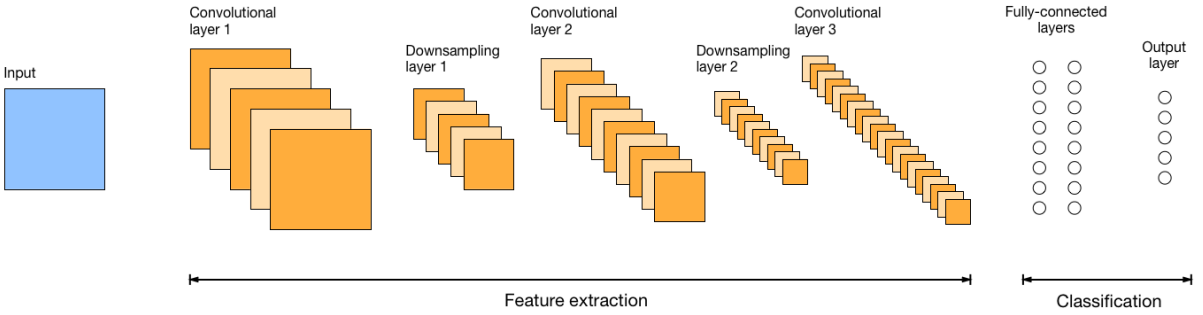

FIGURE 3.5 – Convolutional neural network for classification

less easy when the label noise is class dependent. However, it is possible to apply noise estimation techniques, as explained in [Patrini et al., 2017].

Although developing strategies to counter *label noise* is necessary, it is worth noting that deep neural networks are surprisingly robust to it. An interesting result can be found in [Rolnick et al., 2017], where the authors report that deep networks are essentially robust to an arbitrary amount of label noise, provided that the training set is large enough. They show that good performance is possible with a label accuracy of only 1% above chance. Aside from increasing the training set size, they demonstrate experimentally that increasing the training batch size and adapting the learning rate are good means to compensate *label noise*.

3.4 Convolutional neural networks

Convolutional neural networks (CNN) differentiate from ordinary neural networks in several points. While fully connected neural networks assume unordered and flattened vectors of values, CNNs make the explicit assumption that the input is a structured array, and encode this assumption into their architecture. CNNs are typically composed of several types of layers, such as convolution layers, down-sampling layers and fully-connected layers (see Fig. 3.5).

3.4.1 Convolutional layer

Convolutional layers are designed to process data arrays, whose local groups of values are highly correlated. This is the case for many natural signals : 1D arrays for temporal sequences (e.g. cardiovascular signal, language), 2D arrays for images and 3D arrays for image sequences or volumetric images. Taking into account the correlation of neighboring array elements allows to characterize local data patterns. This is performed by applying neurons in a convolutive manner, which has two important advantages. First, the connectivity of a neuron is constrained to a local region of its input array, called the *receptive field*. This limited connectivity greatly reduces the number of neuron parameters. Indeed, if fully-connected neurons were used for images, the number of connections (and hence parameters) would quickly become intractable (e.g. an image of size 256×256 pixel would need 65536 connections). Second, convo-

luting the neurons with an input array implies that the same neuron is applied to all image locations. This makes the operation location invariant, which is a useful property for image data (and other signals), where the same motif is likely to appear in any part of the image. Applying a neuron in a convolutive manner (i.e. as a sliding window) avoids using a multitude of neurons to cover the whole image area, as would have been necessary given the limited connectivity of the neurons. This allows to greatly reduce the number of network parameters, and is referred to as *parameter sharing* in the literature [Goodfellow et al., 2016b]. Since neurons are convoluted with their inputs, it is common to view them as filters, each responsible for detecting a particular pattern. Following this idea, a convolutional layer can therefore be thought of as a filter bank, whose hyper-parameters include the number of filters and the filter size.

Hierarchical representation An important concept is the combinatory power of convolutional layers. To illustrate this point, let us consider as a network input a 2D image, having two spatial dimensions h and w (height and width), as well as a third dimension c (channel). For a color image the size of the channel dimension is 3, because the image has 3 color channels : red, green and blue. When processed by a convolutional layer, the image is filtered by the layer neurons, each outputing an individual two-dimensional array, the so-called *feature maps* (i.e. the filtered images). These feature maps are stacked (i.e. concatenated) along the channel dimension c, resulting into a 3D output array, which is passed on to the next layer. Note that the channel dimension of the output array size is therefore equal to the number of neurons in the convolutional layer.

We have previously defined the receptive field of a neuron. Now, while it is true that the neuron has sparse, local connections in the spatial dimensions h and w, the neuron is always fully connected along dimension c. In other words, every filter is small spatially (along h and w), but extends through all channels of the input 3D array. This means that while filtering the input array, the neurons also make linear combinations of all input feature maps, which is a powerful property. Combining all input feature maps enables to compute more complex features from layer to layer, allowing CNNs to efficiently learn increasingly abstract concepts. This results into a hierarchical representation of data (as discussed in Section 3.1), where first layers learn how to encode simple low-level features such as edges and textures, allowing the next layers to gradually capture more complex, high-level features like shapes (e.g. circles, triangles), objects (e.g. eyes, noses) and object ensembles (e.g. faces), as illustrated in Figure 3.6.

3.4.2 Down-sampling layers

Down-sampling spatial dimension The objective of downsampling layers is to reduce the input spatial dimensions h and w by a certain factor, typically by two. This allows to save memory throughout the network layers, which generally have an increasing number of filters for deeper layers. By "coarse-graining" the feature maps, it also allows to achieve invariance to small shifts, and to obtain a more compact data representation. However, the down-side is that by reducing the resolution of feature-maps, the exact location of the features are lost. Down-sampling is typically performed using pooling layers, where adjacent patches of feature maps are summarized using the mean operation (i.e. mean-pooling) or the maximum operation (i.e. max-pooling). The most widely used pooling is *max-pooling*, because it works well in practice.

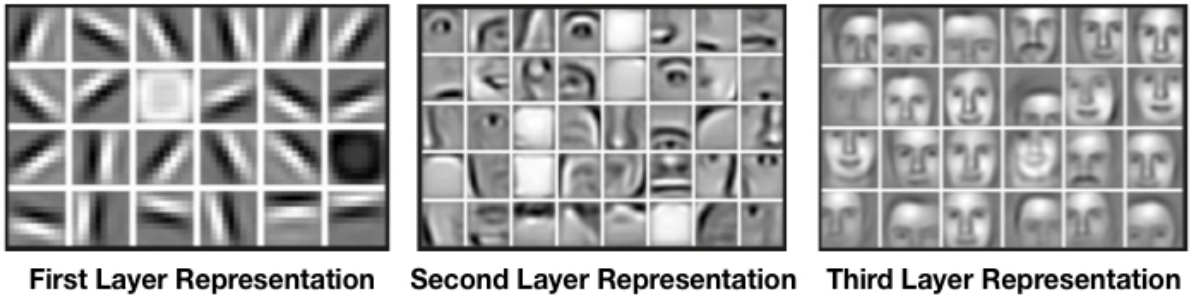

FIGURE 3.6 – Visualization of filters trained for face recognition. From layer to layer, the representation becomes more complex : first the filters encode edges (i.e. Gabor-like filters), then face parts (e.g. eyes, noses) and finally entire faces. Source : [Lee et al., 2009].

An alternative to pooling layers, in which the input is down-sampled by a fixed operation, is using a convolutional layer with a larger convolution stride. The *stride* determines by how many pixels the filters are slided along spatial dimensions (e.g., if the stride is 1, then the filters are slided 1 pixel at a time ; when the stride is 2 then the filters jump 2 pixels at a time), and counts among the hyper-parameters of convolutional layers. The idea behind this is to learn the down-sampling operation, as opposed to using an imposed operation. This also allows to learn different down-sampling strategies for each layer, tailored specifically to incoming feature maps at a specific network depth. Such layers are known as down-sampling convolution layers (or in short "down-convolution" layers), and where proposed in [Springenberg et al., 2015]. For example, to obtain a down-sampling ratio of 2, one should use a filter size of 2×2 pixels with a stride of 2. It has been shown that using down-convolution layers instead of pooling layers is beneficial for training generative models, such as generative adversarial networks (GAN) [Radford et al., 2015].

Down-sampling channel dimension Alternatively, is is also possible to reduce the input channel dimension, for similar reasons as above, namely reducing memory use and obtaining a more compact and robust data representation. As before, this task is achieved by a convolutional layer. The idea is to reduce the number of incoming feature maps by linear combination. Let us recall that filters extend through all channels of their input array. Therefore, using a convolution layer with a filter size of 1×1 ($\times c$), and a smaller number of filters than the number of incoming feature maps, allows to reduce the size of channel dimension c without changing the spatial dimensions h and w. This allows, as for down-convolutions, to learn the down-sampling operation and has first been investigated in [Lin et al., 2013]. Reducing the output channel size intentionally is also known as a "bottleneck" operation, or "bottleneck" layer [Grezl et al., 2007].

Of course, it is therefore also possible to reduce all dimensions h, w and c with a single convolution layer using above described properties.

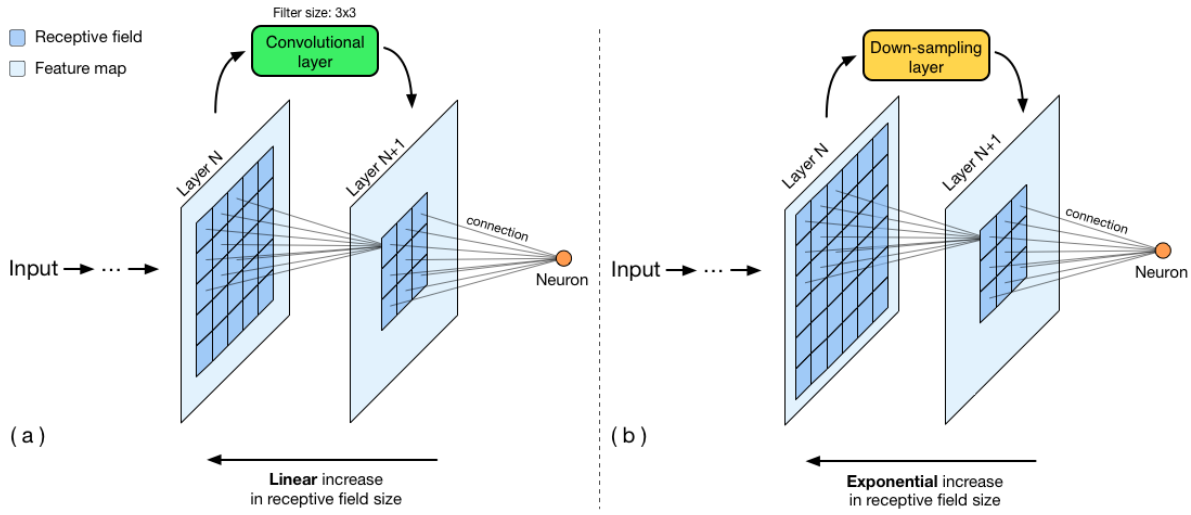

FIGURE 3.7 – Receptive field increase when applying (a) a convolutional layer and (b) a down-sampling layer.

3.4.3 Receptive field : integrating context information

The receptive field is an important concept in CNNs, responsible for integrating context information from an image. We have previously defined the receptive field (see Section 3.4.1) as being the input region a neuron is connected to. A receptive field can be calculated not only for the neuron input, but also relative to the neuron output from a precedent layer. Most often, when the term receptive field is used, what is meant is the receptive field of the network output neurons relative to the network input. Indeed, this defines the region size of the input image that influences the final decision of the network. A large receptive field allows the network to capture context information, it is therefore important to maximize its size in order to characterize the semantics of an image scene.

The receptive field can be increased by using larger filters or adding more layers to the network, as is illustrated in Fig. 3.7. Adding convolutional layers tends to increase the receptive field size linearly, while adding pooling layers increases the size exponentially. However, using large filters increases the number of parameters and when using pooling layers spatial information is lost and the feature-map size (and therefore representation size) decreases rapidly. Therefore compromises must be made to maximize the field-of-view while accounting for these factors.

3.5 CNNs for semantic segmentation

Segmentation is about assigning a class label to each image pixel (i.e. pixel-wise classification), in contrast to image classification where a single class label is attributed to the entire image. Unlike classification networks which only need to resolve *what* is in the image, segmentation networks also need to resolve *where* recognized classes are localized. In other words, the network output needs to preserve spatial resolution, which in the literature is referred to as *dense prediction*. However, locating is hampe-

red by the use of down-sampling layers, which are required to build deeper networks while being limited by memory. Indeed, by down-sampling along h and w the spatial information is lost.

First, DL segmentation approaches consisted in applying a classification network like a sliding window. The image is divided into numerous overlapping patches, which are processed individually by the network (i.e. patch classification). The reason for using patches was that classification network usually have fully-connected layers, limiting the network input to a fixed size. The disadvantage is that using a network as a sliding window demands high computational cost.

These limitations were overcome by introducing fully convolutional neural networks (FCNN) [Long et al., 2014]. In FCNNs, fully-connected layers are discarded and replaced by convolutional layers. This is based on the observation that a fully-connected layer is equivalent to a convolutional layer having a filter size equal to its input. This strategy has several advantages. First, the network input is no longer limited to a fixed size. Second, the network can output entire label arrays, instead of a single label as is the case when using fully-connected layers. This significantly speeds up the prediction process. Another contribution of [Long et al., 2014] is to propose strategies for performing dense predictions. For compensating the spatial resolution loss at the network output, an up-sampling step is applied. However, instead of using simple interpolation, the up-sampling operation is learned, in a similar way to down-convolution (see Section 3.4.2). This so-called up-sampling convolution (or up-convolution for short) is implemented by applying naive up-sampling (e.g. nearest neighbor interpolation), followed by a convolutional layer (see Fig. 3.8). The filters refine the initially coarse-grained feature maps generated by the naive interpolation. But, even with up-convolution layers, the generated feature maps are coarse, due to a lack of spatial information. Therefore the authors propose to use skip connections between feature maps of different resolutions. Combining feature maps from deeper layers (encoding the 'what') with feature maps from higher layers (encoding the 'where') enables to obtain accurate, high resolution segmentations.

The work in [Long et al., 2014] constitutes an important milestone, as most current segmentation networks adopt its concepts, i.e. discarding fully-connected layers, use of up-convolutions and skip connections. These paradigms have been further developed in architectures like the U-net [Ronneberger et al., 2015], which adopts an encoder-decoder strategy (see Fig. 3.9). The encoding layers build an efficient and compact data representation with pooling, whereas the decoding layers gradually recover object details by combining feature-maps from all available resolutions, with skip-connections. U-net has been developed to segment biomedical images, and has an important success due to its pixel-level accuracy, prediction speed and also because its architecture is easy to understand.

An alternative strategy to achieve dense prediction consists in using dilated convolutions [Yu and Koltun, 2016], also known as "a trous" convolutions. Dilated convolutions allow to encode long distance relationships (i.e. context) without loosing resolution. The dilation parameter is an additional hyper-parameter of the convolutional layer. When a filter is dilated, its kernel size and hence its receptive field is increased. However, the additional kernel elements are zero-padded, as illustrated in Fig. 3.10, therefore the number of filter parameters remains unchanged. By doing so, dilated convolutions allow an exponential increase of the field of view, without loosing resolution (as opposed to pooling layers). Hence, dense predictions can be obtained without increasing the number of parameters. However, as the size of feature maps does not decrease through the network, more memory is required. This can be problematic, kno-

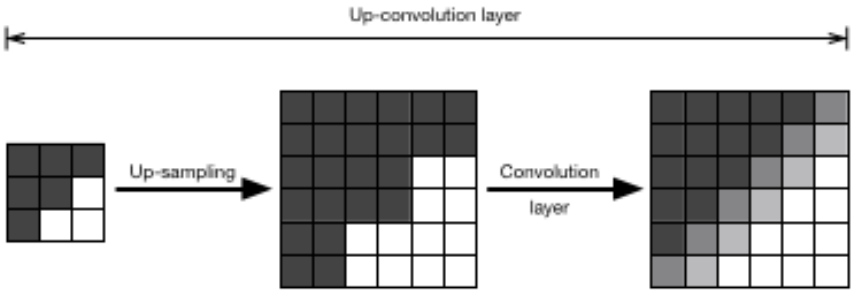

FIGURE 3.8 – The up-convolution layer is composed of a naive up-sampling step (e.g. nearest neighbor interpolation) followed by a convolution layer. After application of the multiple filters of the layer, the feature map grain becomes finer.

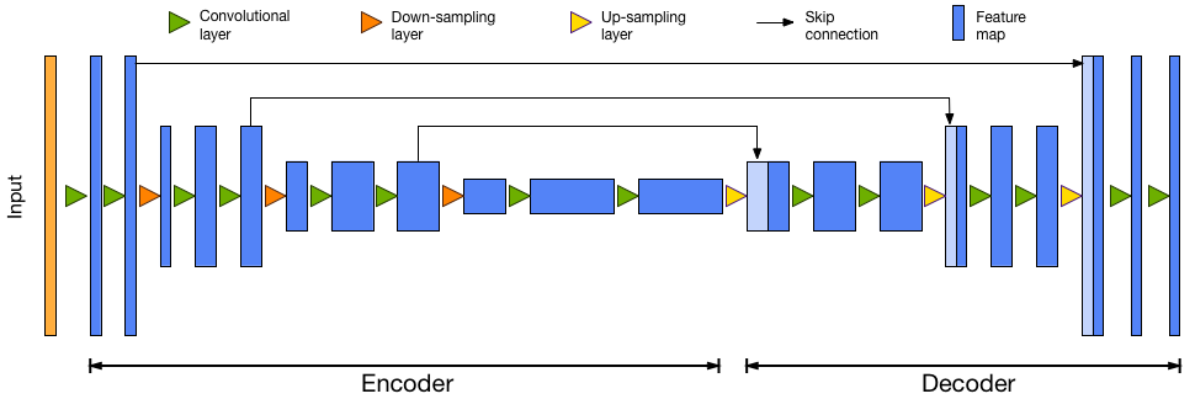

FIGURE 3.9 – A U-net inspired architecture dedicated to high-resolution segmentation [Ronneberger et al., 2015].

wing that the number of filters (and hence the number of feature maps) usually increases with network depth, to encode high-level information. The choice of a dense prediction strategy will therefore depend on available memory, the data size (e.g. 2D or 3D) and the level of abstraction needed (network depth and width).

Losses for segmentation For segmentation applications, dedicated losses have been proposed, among which the most popular is the *Dice loss*. It is derived from the Dice coefficient, which is equivalent to the F1-score. In a binary setting (i.e. 2 classes), this loss function is known to be robust to class imbalance. This is of particular interest in image segmentation, where the number of object pixels is usually low compared to background pixels (especially for 3D images). The dice loss is defined as follows :

$$L_{\mathrm{D}} = -\frac{\sum_{i=1}^{n} t_i y_i}{\sum_{i=1}^{n} t_i + y_i},$$

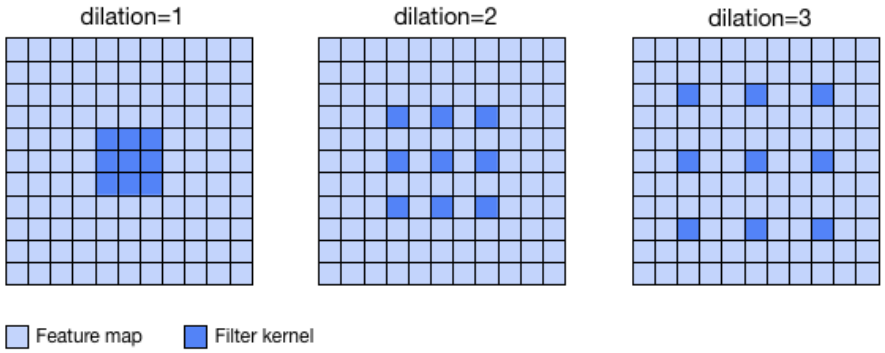

FIGURE 3.10 – Dilated convolution

with y the network prediction, t the training target and n is the number of elements of the output arrays.

In a multi-class setting, it is common to compute the loss for each class individually (i.e. one versus all) and to average the losses. In this case, the loss looses its natural robustness to class imbalance, which is why it is common to ponderate the total loss as explained in Section 3.3.4 [Sudre et al., 2017]. Other losses have been introduced for image segmentation, such as the Tversky-loss [Li et al., 2017], which is a generalized form of the dice loss. Recently, exponential forms of cross-entropy and dice losses have been proposed for 3D segmentation for highly unbalanced classes [Wong et al., 2018].

CHAPITRE 4

3D CONVNET IMPROVES MACROMOLECULE LOCALIZATION IN 3D CELLULAR CRYO-ELECTRON TOMOGRAMS : "FINDING A NEEDLE IN A HAYSTACK"

Cryo-electron tomography (cryo-ET) allows one to capture 3D images of cells in a close to native state, at sub-nanometer resolution. However, noise and artifact levels are such that heavy computational processing is needed to access the image content. In this chapter, we propose a deep learning framework to accurately and jointly localize multiple types and states of macromolecules in cellular cryo-electron tomograms. We compare this framework to the commonly-used template matching method on both synthetic and experimental data. On synthetic image data, we show that our framework is very fast and produces superior detection results. On experimental data, the detection results obtained by our method correspond to an overlap rate of 86% with the expert annotations, and comparable resolution is achieved when applying subtomogram averaging. In addition, we show that our method can be combined to template matching procedures to reliably increase the number of expected detections. In our experiments, this strategy was able to find additional 20.5% membrane-bound ribosomes that were missed or discarded during manual annotation.

4.1 Introduction

A well-established method for localizing macromolecules is template matching (TM) [Best et al., 2007], where a template containing the macromolecule of interest is used to explore a given 3D cryo-tomogram. While TM is efficient for localizing large macromoelecules such as ribosomes, it is necessary to apply several image post-processing and analysis methods to decrease the false positive (FP) rate (see Fig. 4.1). These methods include selection of regions of interest (e.g. areas in the cytoplasm), and thresholding the TM score (e.g. correlation score) values. Additional difficulties also arise when TM is used to detect and localize several types of macromolecules that are structurally similar or specific macro-

molecule states, like binding states of ribosomes (e.g. membrane-bound vs cytoplasmic ribosomes). In general, TM is applied several times to detect each subclass of interest. Unfortunately, the score values are not selective enough to allow one to perform a satisfying classification, especially when the number of considered subclasses is high. Therefore the sub-volumes containing the detected macromolecules (also named particles) of interest are manually analyzed or automatically post-processed by sophisticated classification algorithms [Förster et al., 2008]. Such complex and time-consuming processing chains are routinely applied to accurately localize macromolecules and to identify the related native structure in the cell. Note that each TM and sub-volume classification round can each take 10 to 30 hours of computation on specialized CPU clusters.

In this chapter, we propose an unified deep learning-based framework [Lecun et al., 2015] to jointly and fastly localize and classify macromolecules in cryo-ET. Deep learning (DL) is a set of machine learning techniques capable to produce state-of-the-art results in various fields (e.g. computer vision [Lecun et al., 2010], language processing [Hinton et al., 2012], super-resolution microscopy [Ouyang et al., 2018] and bioinformatics [Naylor et al., 2018]). In particular, convolutional neural networks (CNN) are able to produce impressive results in image analysis, including image classification [Krizhevsky et al., 2012], segmentation [Long et al., 2014] and object recognition [Szegedy et al., 2013]. A neural network is generally composed of successive neuron layers, each transforming incoming data and transferring it to the next layer. The neurons can be seen as small processing units capable of performing linear and non-linear operations. Each neuron is controlled by parameters which are optimized during the learning process. In the case of CNNs, the neurons are applied in a convolutive manner, which allows dealing with the information redundancy of neighboring pixels (neighboring pixels have similar values). Thus, a neuron can be thought of as a filter, and a neuron layer as a filter bank. The role of a layer is to automatically extract features from the data. Applying sequentially the layers enables to progressively compute more abstract features, which results in a hierarchical representation of the data. The underlying idea is to learn high-level features from low-level features, which allows a computer to understand complex interactions from basic patterns. The first layers typically encode basic features such as image contours/edges and textures, which allows the next layers to gradually capture more complex shapes (e.g. circles, triangles), objects (e.g. eyes, ears), object ensembles (e.g. faces) and object conditions (e.g. face gender). Those powerful data representations are learned automatically from the data, and tend to be more efficient than conventional handcrafted representations, which require human ressources and are time consuming to design.

Deep learning has been recently investigated to learn high-level generic features in cryo-electron microscopy (cryo-EM). In [Wang et al., 2016], the authors proposed first a CNN architecture to automatically detect particles in single particle cryo-EM 2D micrographs. The computational method was designed to detect a unique object class in 2D images depicting stationary noisy backgrounds. In [Chen et al., 2017], a CNN architecture was used for the first time to analyse cryo-ET data ; the authors proposed a DL framework especially dedicated to tomogram segmentation. They posed the segmentation of cryo-ET images as a set of N binary voxelwise classification problems. An ensemble of N 2D CNN (one per class) is applied slice per slice on the 3D tomograms. The modeling and computation is clearly sub-obtimal, but the proposed practical implementation (available in the EMAN2 package [Tang et al., 2007]) allows one to satistfyingly find several object classes such as cell membranes, microtubules and

ribosomes in tomograms. While the authors also show that the proposed DL framework can be used to pick up ribosomes for subtomogram averaging, an additional post-classification step is necessary to get more satisfying subtomogram averages. Unfortunately, no quantitative analysis is presented in [Chen et al., 2017] to assess the localization accuracy of detected particles and the actual resolution of macromolecule structures once subtomogram averaging is performed. Unlike [Wang et al., 2016, Chen et al., 2017], we consider a fully 3D CNN architecture in order to more accurately and reliably detect 3D particles in a native crowded cell environment as illustrated in Fig. 4.2. In addition, our network is also capable to handle multiple object subclasses at the same time. We especially demonstrate that manipulating jointly a higher number of object subclasses/classes is the key approach to improve performance of CNNs in 3D cryo-ET. Moreover, complex and time-consuming post-classification steps are no longer required to produce reliable results contrary to previous approaches [Chen et al., 2017]. Besides, while training in [Chen et al., 2017] is faster (10 min per class) than with our method (12 hours), our processing time is at least twice as fast for one class and remains constant when the number of classes increases (see Fig. 4.3). In our experiments, we compared our 3D CNN architecture to TM in order to emphasize how the ways in which they were designed and their potentials differ. Unlike TM, our CNN framework is able to localize multiple types of macromolecule at once when applied to a given cryo-tomogram. We also show quantitatively for the first time how the CNN framework, when combined to TM guided procedures, can substantially improve the localization sensitivity of the structure of interest on real cryo-ET data.

The remainder of this chapter is organized as follows. In Section 4.2, we present our deep learning framework. In Section 4.3, we explain our experimental setup focusing on ribosomes and the results. We show that DL outperforms TM on synthetic images and how TM and DL can be combined to improve the localization sensitivity in real data. In Section 4.4, we discuss the potential of DL in cryo-ET and future work.

4.2 Method

4.2.1 Localizing multiple objects with a CNN architecture

Given a training set of object classes, we propose a supervised 3D CNN-based method to classify the 3D tomogram voxels into several types of macromolecules or states of a given macromolecule. A clustering algorithm is then applied to aggregate voxels into clusters and to determine the position of particles (gravity center of clusters) in the volume (see Fig. 4.1). The detected particles are further exploited for subtomogram averaging.

Step #1 : Multiclass voxelwise classification

Our objective is to provide a classification map for which each voxel in the 3D map is assigned an object class. The convolutional neural network is usually designed to produce a single output label : it exploits global information by progressively down-sampling the image layer by layer, in order to preserve only relevant information and reduce computation. The disadvantage is that by doing so, the network

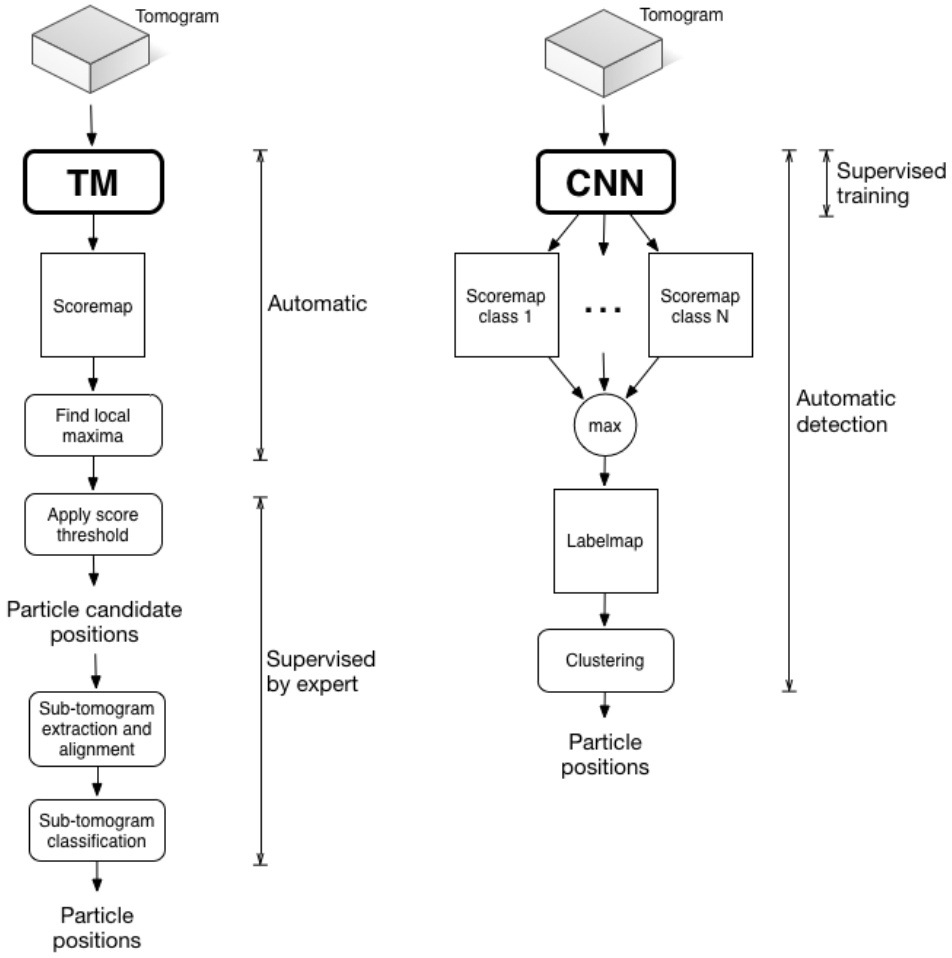

FIGURE 4.1 – Comparing workflows : on the left, the common processing chain involving TM and on the right, our DL framework. Our approach is multi-class, whereas the TM processing chain needs to be applied once for each class.

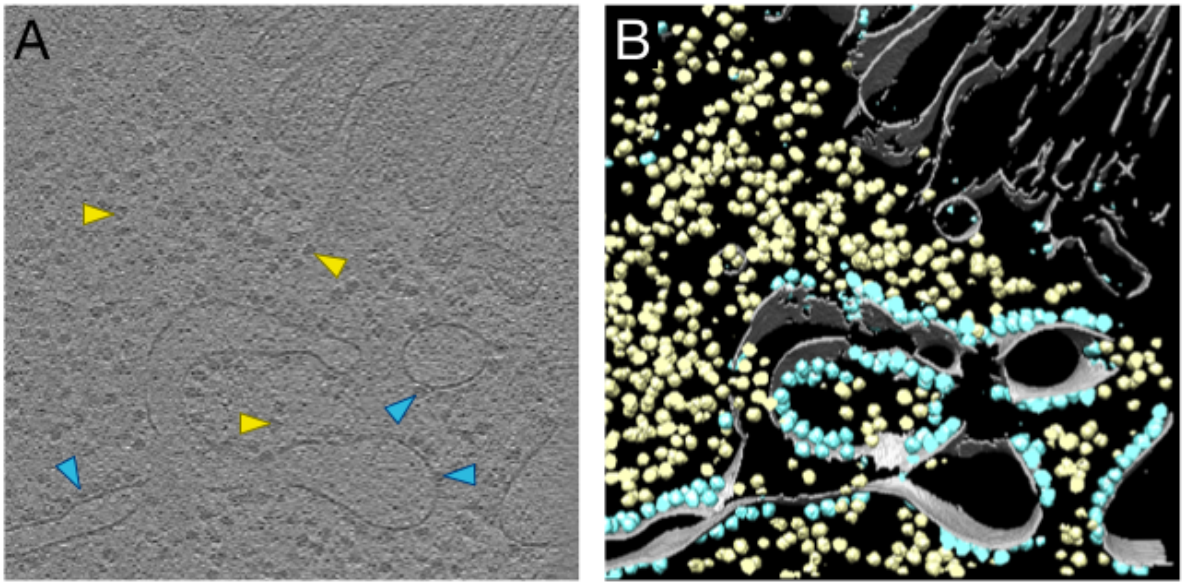

FIGURE 4.2 – Chlamydomonas cell. (A) Tomogram slice ; blue arrows indicate *mb-ribos* ; yellow arrows indicate *ct-ribos*. (B) Corresponding voxelwise classification obtained by our 3D CNN, performed for 3 classes : *mb-ribos* (blue), *ct-ribos* (yellow) and *membrane* (gray).

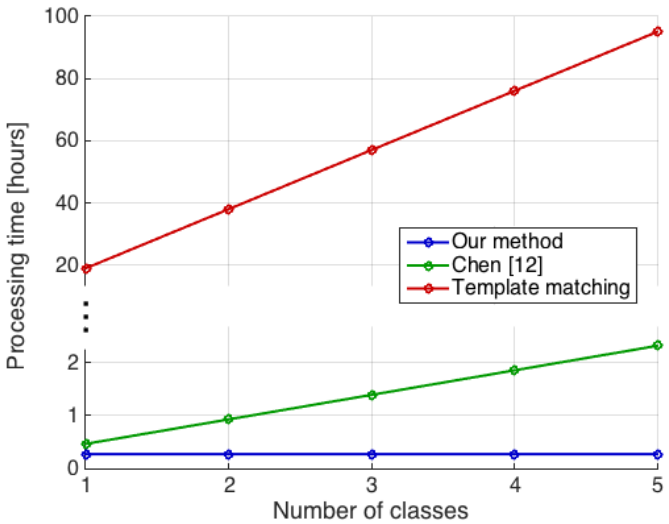

FIGURE 4.3 – Time needed to process a tomogram of size $928 \times 928 \times 464$ voxels, w.r.t. the number of classes (once training is completed). We compare our method to [Chen et al., 2017] and to template matching. We do not consider post-processing in displayed time values (i.e. without clustering step for our method, and without connected component analysis and post-classification for [Chen et al., 2017]). We use a Tesla K80 GPU for our method and a 32-core CPU cluster for template matching, while in [Chen et al., 2017] the authors used a 12-core CPU workstation.

loses the local information needed for voxelwise classification tasks. While the network is able to reliably decide if an object is present or not in an image, it is not able to accurately estimate the position of the object. As opposed to this conventional approach, fully convolutional networks (FCNN) [Long et al., 2014] overcome this difficulty by explicitly combining global and local information in order to provide high resolution label maps. FCNNs involve more interactions in the network and adapt conventional classification networks to perform more robust image voxelwise classification. This idea was exploited in [Ronneberger et al., 2015] and then adapted in [Milletari et al., 2016] for 3D image analysis. As described in [Ronneberger et al., 2015, Milletari et al., 2016], our architecture consists of a down-sampling path needed to generate global information and a up-sampling path used to generate high-resolution outputs, i.e. local information (see Fig. 4.4). Down-sampling is performed with max-pooling layers (factor 2) and up-sampling with up-convolutions [Long et al., 2014] (sometimes called "backward convolution"), which is basically a trained and non-linear up-sampling operation. Combining global and local information is performed by concatenating features at different spatial resolutions. The features are then processed with the convolutional layers of the up-sampling path. Unlike [Milletari et al., 2016], our architecture is not so "deep" since we found that using more than two down-sampling stages does not increase the classification results. Also, we used only $3 \times 3 \times 3$ filter sizes as in [Simonyan and Zisserman, 2015]. The rationale behind this choice is that two consecutive $3 \times 3 \times 3$ filters mimic a larger $5 \times 5 \times 5$ filter but with fewer parameters. Training is then faster and easier and requires less memory. An important concept in neural architectures is the receptive field of deepest neurons layers. It determines the size of the spatial context to be used to make decisions. Considering a large spatial context is essential to handle an object class involving interactions with the environment, for instance interactions with the cell membrane. It is established that adding convolutional layers after down-sampling operations is appropriate to enlarge the spatial context [Milletari et al., 2016]. Accordingly, we added two supplementary convolutional layers in the lowest stage of our architecture. To complete the description, we use rectified linear units (ReLU) [Krizhevsky et al., 2012] as activation function for every layer except the last one which uses a *soft-max* function. While ReLU is a popular choice to tackle non-linearities in the network, the *soft-max* function is mandatory in order to interpret the network outputs as probabilities for each class.

In summary, our proposed CNN architecture is capable of robustly classifying the cryo-ET tomogram into N subclasses/classes with a high accuracy. Given the voxelwise classification map, the next step consists in estimating the position of each individual object, as described in the next section.

Step #2 : Clustering for macromolecule localization

Given the multiclass voxelwise classification map and classification errors, our objective is determine the position of each particle corresponding to a state of a given macromolecule or several types of macromolecules. The voxel labels should be ideally spatially well clustered into well distinct 3D connected components, each cluster corresponding to unique object/particle. Because of noise, non-stationarities in the background, and artifacts in the tomogram, the CNN method generates isolated labels or very small groups of voxels, and groups that contain different label types. Post-processing is then necessary to assign an object class to a given position.

To address this issue, we apply a basic clustering approach. The clusters are built by aggregating

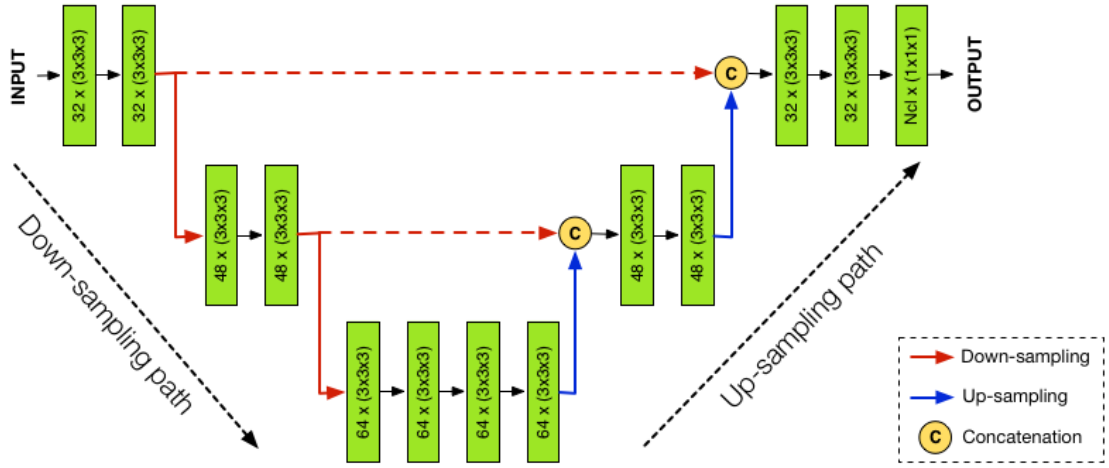

FIGURE 4.4 – CNN architecture. In green convolutional layers labeled with (#filters × (filter size)). In the last layer, Ncl stands for the number of classes.

neighboring voxels into objects and form 3D connected components. The smallest clusters are considered as false positives and are discarded. The cluster centroid is further used to estimate the object position. As the centroid is computed by uniformly averaging the coordinates of cluster voxels, we are able to numerically produce positions with sub-voxel precision. As several voxel labels can be spatially grouped in a given cluster, the most frequent label or subclass/class is assigned to the detected particle. To address the computational issues, we used the popular mean-shift clustering algorithm [Comaniciu et al., 2002]. The main advantage of mean-shift is that it is controlled by only one parameter, commonly called the bandwidth, which is directly related to the average object size. The K-means algorithm was not considered further since the number of clusters must be provided as input parameter by the user.

4.2.2 Training

In training step, the CNN is learned from pairs of tomograms and their corresponding voxelwise classification. In other words, the CNN needs every tomogram voxel annotated as member of the class of interest or as background. While voxelwise classification examples are naturally available for synthetic data, it is often not the case for experimental data. In our case, the experts accurately localized the macromolecules of interest in several training tomograms. Actually, voxelwise classification cannot be performed manually for two reasons : i/ it is time consuming to label each voxel in hundreds of objects in 3D ; ii/ the data is so noisy that the object borders are barely visible. To address this issue, we propose an original computational approach based on subtomogram averaging [Förster and Hegerl, 2007] to label voxels, only from the expert annotations corresponding to the spatial coordinates of macromolecules (see Fig. 4.5). Subtomogram averaging is a registration algorithm designed to obtain higher resolution structures by averaging thousands of aligned subvolumes containing the same structural unit. The labeled coordinates serve here as inputs to a subtomogram averaging procedure. The subtomograms around the annotated positions are extracted, aligned and finally averaged. The alignment procedure

outputs the object orientations, whereas the averaging process provides a clean and missing wedge free density of the macromolecule. From this density, it is possible to create a binary mask of the macromolecule by thresholding the averaged subtomogram. Furthermore, the resulting 3D mask is pasted into an empty volume at each labeled position with the estimated 3D orientation. The resulting volume with well delineated macromolecules is then used as a target to train the parameters of the CNN architecture. It is worth noting that annotating the macromolecule with this semi-automatic approach saves time but may introduce "label noise" in the training. Indeed, we use an average shape to label the macromolecule, and we neglect structural macromolecule variability mainly localized in the object borders. Nonetheless, it has been shown that CNNs have a natural robustness to reasonable amount of "label noise" [Rolnick et al., 2017], which is also confirmed in our experiments.

Due to memory limitations, it is not feasible to load the whole tomogram set with the corresponding targets during training. Therefore, we randomly draw smaller 3D patches around macromolecules at each training iteration. The patch size should be large enough to capture sufficient context information ; the macromolecule radius being 10 voxels, we choose a patch size of $56 \times 56 \times 56\times$ voxels. It is also common to use "data augmentation" when training a CNN ; it allows to increase the training set artificially by applying geometric transform to the training images. In our approach, we implement "data augmentation" by applying a $180°$ rotation w.r.t. the microscope tilt-axis to each training example. Nevertheless, we do not use typical mirror operations or geometric deformations because the structure of expected objects is the principal clue in the detection problem. Also, we do not use random rotations because of the well-determined orientation of missing wedge artifacts, which is preserved when applying $180°$ rotations w.r.t. the tilt-axis. In our experiments, the CNN has been computationally trained for 6000 iterations with the ADAM algorithm, chosen for its good convergence rate [Kingma and Ba, 2014], using 0.0001 as learning rate, 0.9 as exponential decay rate for the first moment estimate and 0.999 for the second moment estimate. We use categorical cross-entropy as a loss function (see Fig. 4.6). The training has been performed on a Nvidia Tesla K80 GPU and took 12 hours of computation, which is reasonable knowing that for other tasks, CNN training can last several days [Krizhevsky et al., 2012] [Simonyan and Zisserman, 2015]. In the next section, we will present the training datasets and the results of our CNN approach applied to both synthetic and real tomograms.

4.3 Results

The method has been evaluated and compared to TM on a synthetic and real datasets described below.

4.3.1 Description of data

Dataset #1 : synthetic data

The data has been generated with the AV3 toolbox [Förster and Hegerl, 2007], using atomic densities from the PDB databank [rcsb.org]. Simulation parameters include a voxel size of 13.68 Å(Angströms), a defocus of $-6\mu m$, and several values of signal-to-noise (SNR) from 0.05 to 0.10, and tilt ranges from $\pm 50°$ to $\pm 70°$, with a tilt-increment of $2°$. In this experiment, nine classes of prokaryotic macromolecules

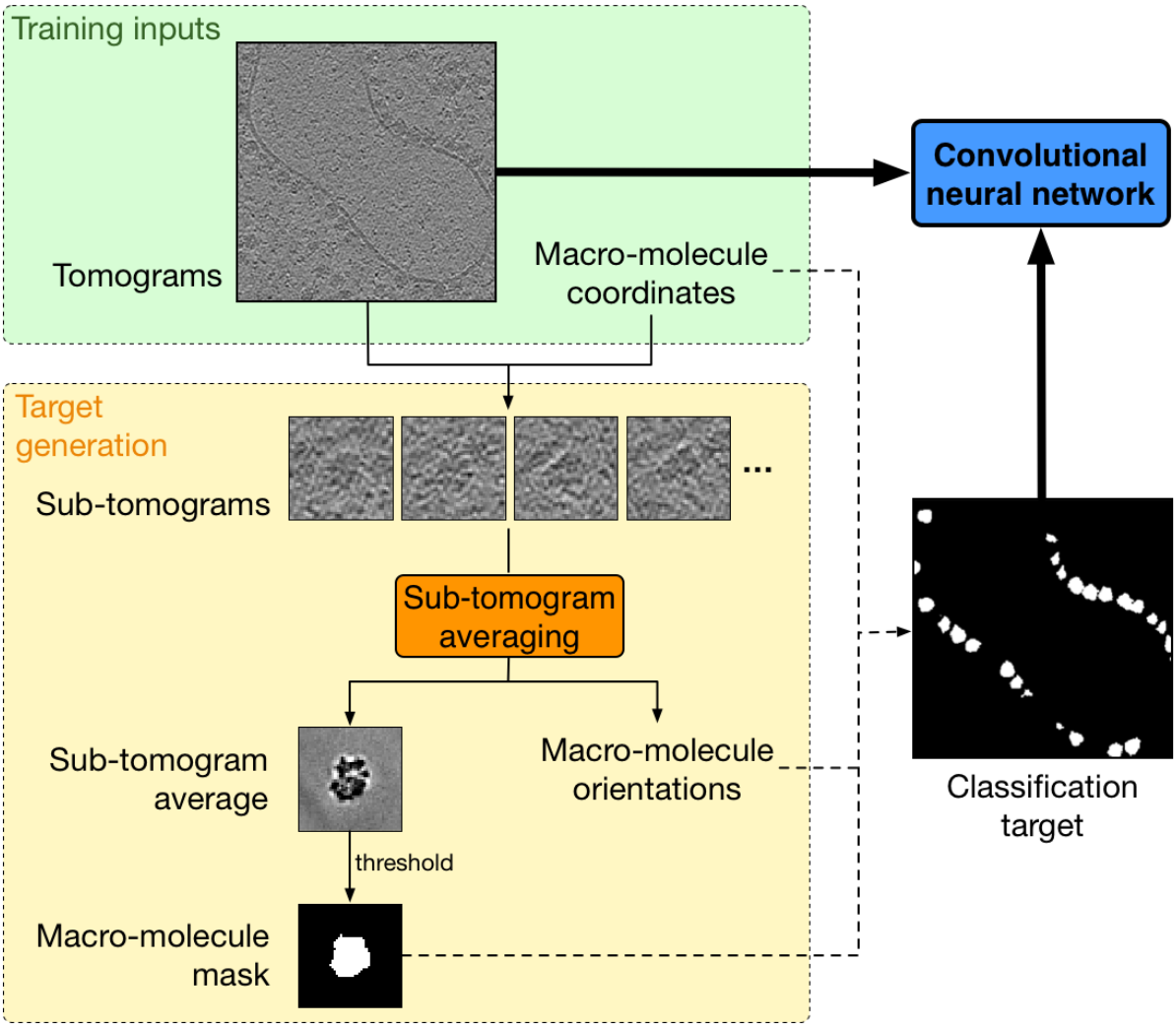

FIGURE 4.5 – CNN training : this figure illustrates how to obtain voxelwise classification examples for training, using only position annotations.

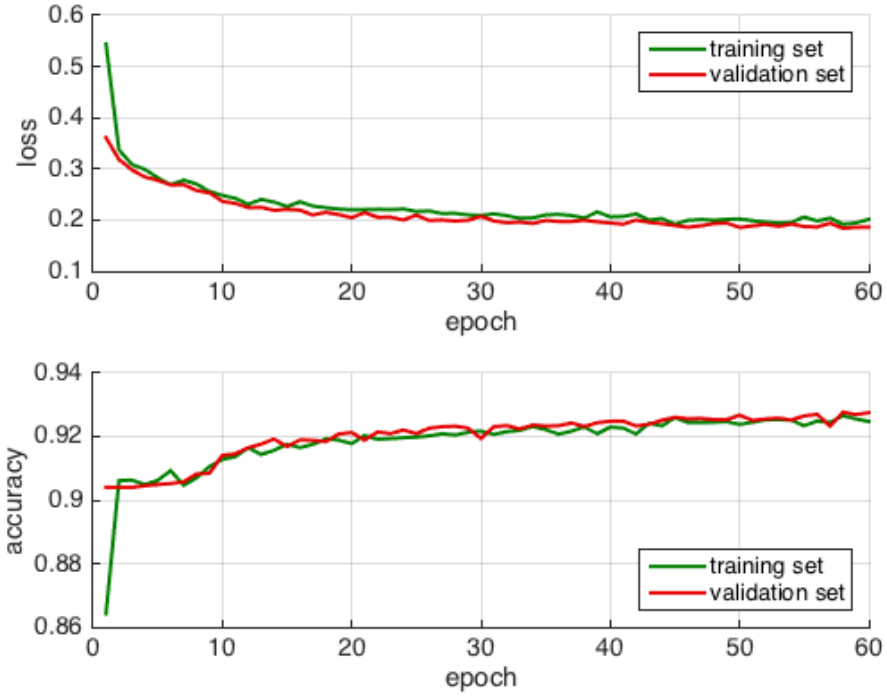

FIGURE 4.6 – Evolution of loss and accuracy during training on dataset #2. These quantities are computed for the training set, as well as for the validation set, in order to estimate the generalization capabilities of our network. The curves for both sets should overlap, else it indicates overfitting (the network memorizes train samples instead of learning discriminating features).

have been chosen to depict varying amounts of inter-class similarity (see Fig. 4.7). The GroEL and ribosome classes are significantly different in size and shape, whereas different functional states of the proteasome and GroEL are structurally similar. In addition to these well-known macromolecules, we have introduced basic objects (ellipses, spheres, discs) in order to mimic not well-defined structures found in a cell, like membrane components, small molecules, and gold particles.

The training set consists of 510 macromolecules for each class, the validation set is composed of 240 macromolecules for each class, and the test set is made of 105 macromolecules for each class. The macromolecules and basic objects have been placed at random positions and orientations in the 3D volumes in order to simulate the crowded environment of a cell.

Dataset #2 : experimental data

The second dataset is composed of 63 tomograms of Chlamydomonas Reinhardtii cells (see [Pfeffer et al., 2017] [Albert et al., 2017] for details about data acquisition) and has been annotated for membrane-bound 80S ribosomes (denoted *mb-ribo* in the following) positions by experts. To get these annotations, the experts first used TM with a template generated from the dataset, using manually selected ribosomes and subtomogram averaging. Then they refined the TM results by applying subtomogram classification (CPCA [Förster et al., 2008]) and performing careful visual inspection (see Fig. 4.1). To reduce computational cost, the tomograms were under-sampled, resulting into a tomogram size of $928 \times 928 \times 464$ voxels and a voxel size of 13.68Å. Tilt range is $\pm 60°$ with an increment of $2°$.

In the end, the dataset has been annotated with 9487 *mb-ribos*. The subtomogram average computed from the total set of *mb-ribos* is the best model we have for 80S ribosomes in Chlamydomonas Reinhardtii cells, and is therefore used as a reference for subtomogram alignment. As this dataset was originally annotated and designed to study the (*mb-ribo*) class, we used computational tools to get examples corresponding to the cytoplasmic ribosome (denoted *ct-ribos*) class and the *membrane* class. First we added the *membrane* class by employing an algorithm dedicated to membrane segmentation [Martinez-Sanchez et al., 2014]. Meanwhile, we got (*ct-ribo*) examples by applying TM and selecting the most isolated candidates, located at a distance higher than 273.6Å (i.e. the ribosome diameter) to membrane components. The motivation behind adding new classes to the available annotations was twofold : on the one hand, our motivation was to demonstrate the multiclass ability of our method on real data ; on the other hand, we noticed that the multiclass approach tends to improve the discriminating power of the network and the capacity to reliably detect *mb-ribos* in real tomograms. When trained by using only the *mb-ribo* examples, the network actually detects unwanted cytoplasmic ribosomes. By considering the *ct-ribo* class in the training, we encourage the network to better discriminate both ribosome subclasses (corresponding to binding states). Note that these annotations of membrane components and cytoplasmisc ribosomes have been obtained without the supervision of an expert. Therefore more errors are expected for these two classes when compared to the *mb-ribo* examples reliably annotated by the experts in our protocol.

Dataset #2 has been arbitrarily split into training, validation and test sets. Training and validation sets have been sampled from 55 tomograms and consist of 5971 *mb-ribos* for training and 1493 *mb-ribos* for validation. The test set is composed of 8 tomograms annotated with 1736 *mb-ribos*.

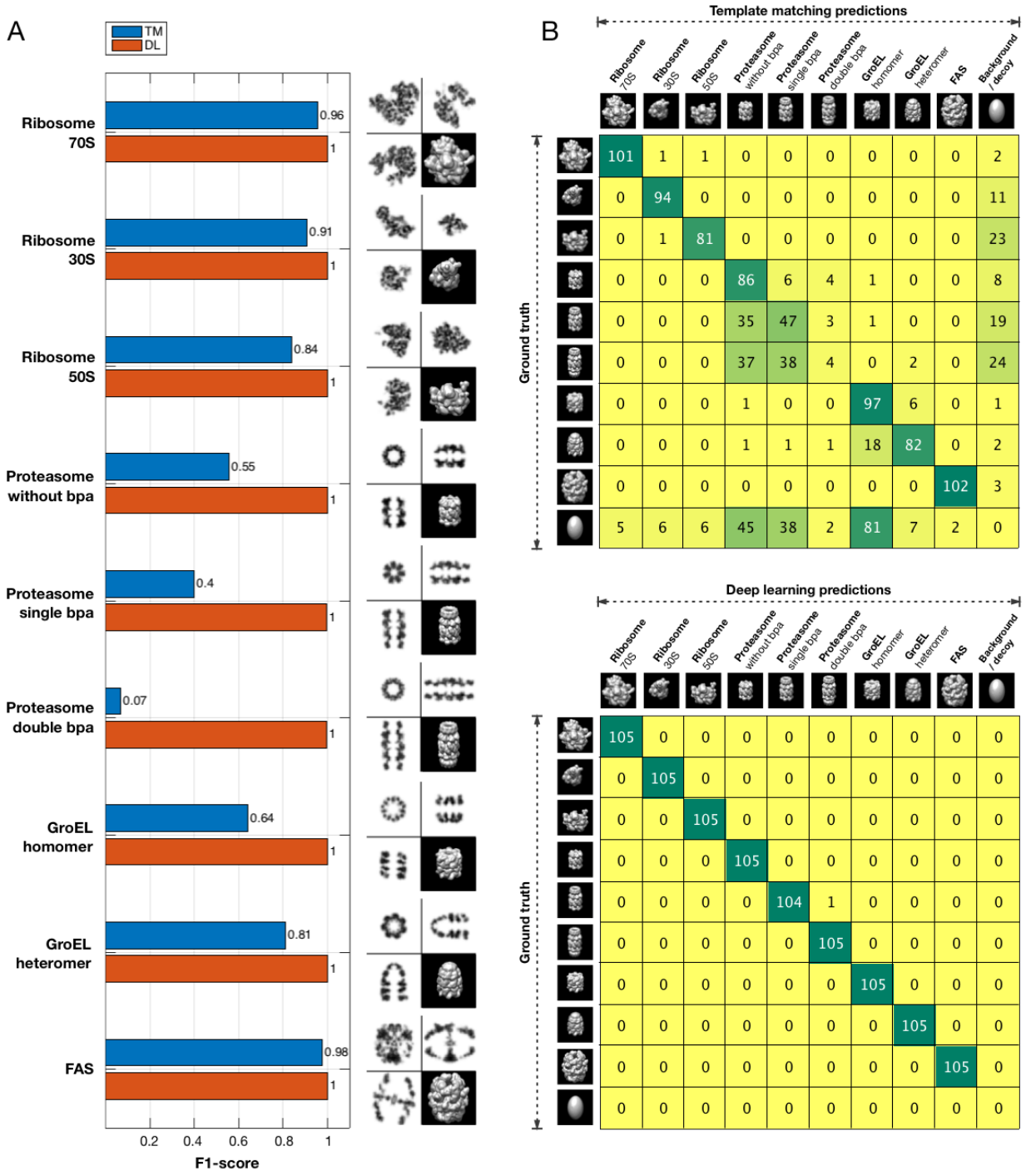

FIGURE 4.7 – Dataset #1 : comparing DL and TM performances per class. For this result we used SNR=0.1 and a tilt range of $\pm 60°$. (A) displays the achieved F1-scores and (B) are obtained confusion matrices, illustrating the miss-classifications of both methods.

4.3.2 Evaluation

The metric used to assess localization performance is the F1-score (also known as Dice coefficient), as commonly used in detection problems. The F1-score can be interpreted as a weighted average of two well-known metrics depending on the number of true positives (TP) :

— Recall (also known as sensitivity) :

$$R = \frac{\text{Number of TP}}{\text{Number of particles in the tomogram}}, \tag{4.1}$$

— Precision (also known as positive predictive value) :

$$P = \frac{\text{Number of TP}}{\text{Number of localized particles}}. \tag{4.2}$$

The F_1-score defined as follows

$$F_1 = 2\frac{RP}{R+P},$$

allows one to evaluate performance by considering a single value.

In what follows, a detected particle is considered as a TP if is closer than 136.8Å (i.e. ribosome radius) to a ground truth object.

First, we quantitatively evaluated the performance of the multiclass localization method on dataset #1. Synthetic data are very helpful to objectively study the influence of SNR and tilt-ranges on results. To perform multiclass localization with TM (which is mono-class), TM has been applied once for each class with the corresponding templates. The templates have been computed from the atomic densities used to generate the dataset #1. A threshold applied to the TM score map is used to select the N best candidates for each class. The threshold is chosen so that the overlap with the ground truth is maximized. As recommended in [Best et al., 2007], if multiple templates compete for the same position, the class with the highest score is selected.

Dataset #2 was used to evaluate the performance of our method in real conditions. In our protocol, only the *mb-ribo* annotations have been provided by experts, and are therefore considered as the ground truth. Accordingly, we only quantify results for the *mb-ribo* class, while providing visual results for the *ct-ribo* and membrane classes. Unlike [Tang et al., 2007], we also analyzed the score distributions and subtomogram averages. The resolution of obtained subtomogram averages is estimated with the commonly-used gold standard Fourier shell correlation (FSC) combined to the "0.143" threshold criterion.

4.3.3 Result analysis

Dataset #1 : synthetic data

It turns out that DL outperforms TM for each class, in terms of F_1-score (see Fig. 4.7 A). An interesting result is that the scores achieved by DL are virtually perfect. In comparison TM scores are class dependent, ranging from 0.07 to 0.98. TM achieves good scores for the biggest macromolecules (ribosome 70S and FAS), but the performance is lower if the size of the macromolecule is small. Indeed,

the confusion matrices (see Fig. 4.7 B) reveal that small targets like proteasomes and GroEL are often confused with background or decoy objects. In addition, TM has some difficulty to tackle inter-class similarity, especially for the three functional states of the proteasome (double bpa, single bpa and without bpa).

We evaluated the robustness of TM and DL methods by varying the signal-to-noise ratio (SNR) and tilt-range. Figure 4.8 plots the average F_1-score over the nine classes. As expected, we observe a performance drop for decreasing values of SNR and tilt-range. However, DL is remarkably stable and produces nearly constant scores. TM on the other hand loses 6% of F_1-score when the SNR decreases from 0.15 to 0.05, and 7% when the tilt-range decreases from $\pm 70°$ to $\pm 50°$. In all tested situations, it turns out that DL is more robust to noise and missing-wedge than TM. In summary, the results on dataset #1 prove that DL is capable to provide better results than TM as soon as DL exploits ideal ground truth during training, that is with no "label noise". These results demonstrate that DL outperforms TM on synthetic noisy and MW corrupted data.

Notice that our DL approach is able to implicitly tackle the missing wedge (MW), while TM usually considers the so-called constrained cross correlation to handle the MW information. Actually, DL is capable to manage non-linear object deformations due to MW during training, without additional prior imposed by the experts.

Dataset #2 : experimental data

Comparison of score values of TM and DL In the first part of experiments, we have carefully examined the scores produced by TM and DL methods. In Fig. 4.9, it is clear that the TM score map is much noisier than the maps generated by DL. Actually, the responses of TM based on the constrained cross correlation are very high at ribosome locations and at undesirable locations corresponding to highly contrasted structures with similar sizes (for instance see cell membrane in Fig. 4.9). TM tends to generate a lot of false positives in the cell. Consequently the experts need to apply post-processing techniques to select relevant information in order to exploit the TM results. Unlike TM, DL provides clean score maps and only depict high values in well-localized blobs. These results suggest that DL is more capable to properly learn the structure and geometry of complex macromolecules.

In order to further support this idea, we examined the distribution of local maxima values in each score map (see Fig. 4.9). A sharp mode (depicted in red in Fig. 4.9) can be regarded as an indicator to assess discrimination quality. As shown in Fig. 4.9, the mode for TM is weak, while for the DL *ct-ribo* class the mode is not very sharp either. However, for the DL *mb-ribo* class, the mode is more significant and sharp, suggesting a score less prone to ambiguity. This observation confirms the idea that the TM score is actually not very discriminating in general. More specifically, it is not a surprise if the score of the *ct-ribo* class is not as discriminating as the score of the *mb-ribo* class. This is probably related to the annotation quality used for learning, much more higher in the case of *mb-ribo* class. In our experiments, the *mb-ribos* annotations have been carefully performed by an expert.

In summary, DL produces sharper scores than TM (constrained cross-correlation), when the training data is carefully labeled (we have a better performance for *mb-ribos* than for *ct-ribos*).

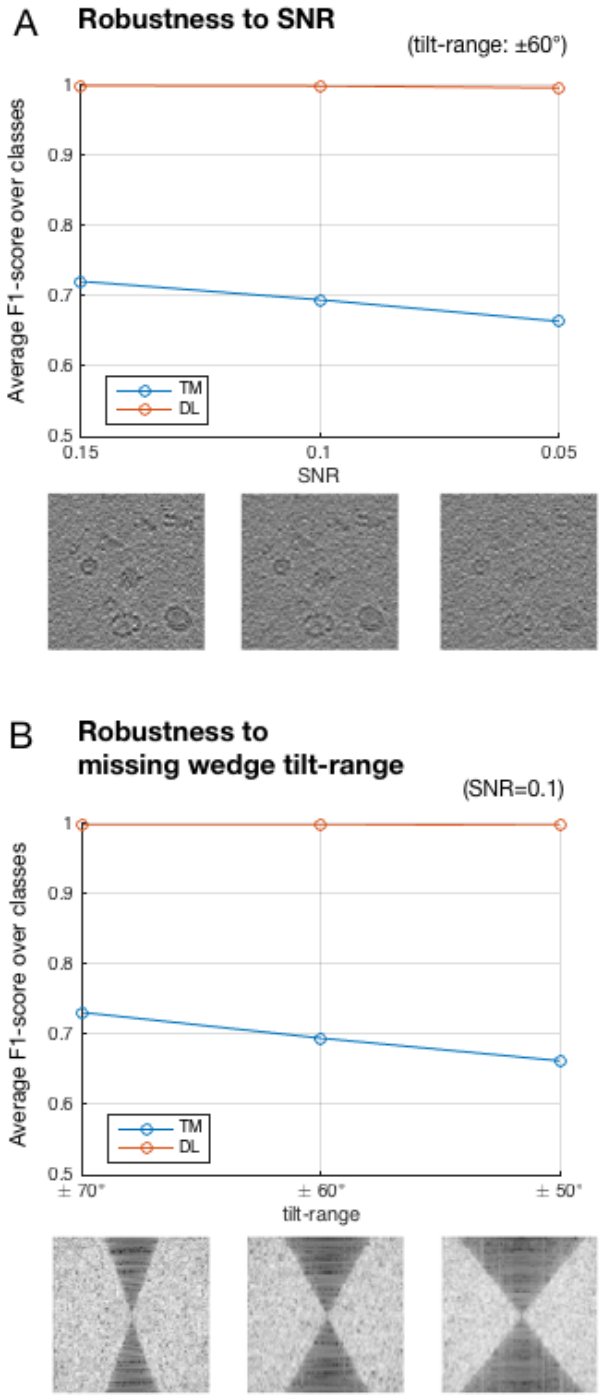

FIGURE 4.8 – Dataset #1 : average F1-score for each method, for varying SNR (A) and tilt-ranges (B). For (A), we consider a $\pm 60°$ tilt-range and SNR values 0.15, 0.10 and 0.05. For (B), we consider a SNR of 0.10 and tilt-ranges $\pm 70°$, $\pm 60°$ and $\pm 50°$. The images below the curves illustrate the effects of the varying parameters on a synthetic data sample (in image domain for (A), in spectral domain for (B)).

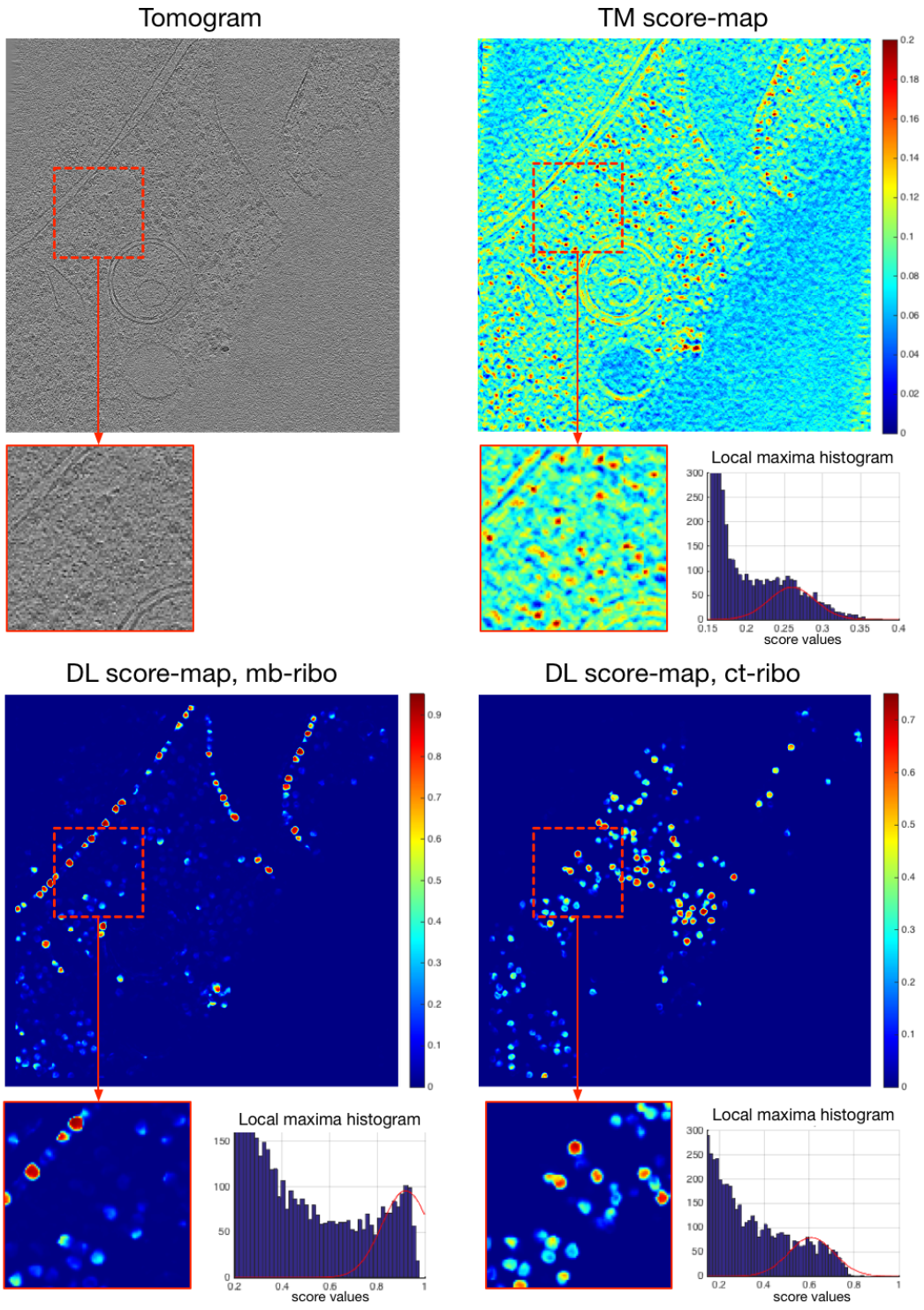

FIGURE 4.9 – Dataset #2 : comparing the score maps obtained from TM and our DL approach. On the score-maps bottom, zoomed-in windows and histograms of local maxima values.

Evaluation of voxelwise multiclass classification Figure 4.10 (A) illustrates the ability of our DL approach to recognize objects and structures in experimental tomograms. Visually, the voxelwise multiclass classification makes sense : *mb-ribos* (in blue) are primarily located against cell membranes, whereas *ct-ribos* (in yellow) occupy the remaining space. We quantitatively measured for each class the Euclidean distances between the ribosomes and the nearest membrane component (see Fig. 4.10 B). The distance histograms confirm the visual analysis : for *mb-ribos*, the distance histogram has a sharp mode at 136.8Å, which corresponds to the ribosome radius. Therefore a large majority of the voxels classified as *mb-ribos* are very close or linked to the membrane. For *ct-ribos*, we notice no sharp mode and the distance histogram looks Gaussian with a large variance. This confirms the heterogeneity of ribosomes freely floating in the cytoplasm. In conclusion, our method allows accurate multiclass object detection in cryo-ET.

Evaluation of overlap with annotations The next step of our evaluation process consists in measuring the overlap between the expert annotations and the *mb-ribos* found by DL as explained earlier. We compare the results to the TM outputs, that is before applying sophisticated post-classification methods. In Fig. 4.11, we plotted the Recall, Precision and F_1 score w.r.t. the DL and TM algorithm parameters. We focused on the thresholds used to detect objects (TM : threshold on score values ; DL : object size threshold) (see Sec. 4.2.1). We obtained a F_1 score of 0.86 for DL and a F_1 score of 0.50 for TM. These numbers illustrates the ability of our DL approach to learn and bypass the expert processing chain. Moreover, the computation time of our DL approach is very small when compared to the TM algorithm as given in Figure 4.3. Now that it has been established that DL has a better overlap with the annotations than TM, in the remaining we focus our analysis on DL detections.

We have also examined the complementarity between the two sets of *mb-ribo* macromolecules detected by the the experts (guided by TM) and the DL method. In what follows, we respectively denote S_E and S_{DL} the sets obtained by the experts and the DL method. While the overlap $S_E \cap S_{DL}$ between both sets is substantial (1516 particles), there is also a significant amount of particles belonging to $S_E \setminus S_{DL}$ (220 particles), i.e. the particles annotated by the expert but overseen by DL, and to $S_{DL} \setminus S_E$ (356 particles), i.e. particles found by DL but overseen by the expert. We can benefit from the two complementary object position estimations to improve overall validation rates. Actually, the union $S_E \cup S_{DL}$ of the two sets enables to increase the list of potential *mb-ribo* macromolecules, for which a confidence level can be assigned to each member depending on whether it belongs to $S_E \cap S_{DL}$, $S_{DL} \setminus S_E$ or $S_{DL} \setminus S_E$. Objects belonging to $S_E \cap S_{DL}$, i.e. found by both methods, are very likely to be true positives. Meanwhile the detected objects belonging to $S_E \setminus S_{DL}$ and $S_{DL} \setminus S_E$ can be labeled as “suspicious” and need more investigation. These two sets are relatively small and the experts may focus on the detected macromolecules that may correspond to rare conformations observed in the cryo-tomogram. From our analysis, it is possible to get a high overlapping rate with the expert annotations by using our DL approach, suggesting that DL is able to learn the expert analysis chain.

Analysis of subtomogram averaging results It is usually recommended in cryo-ET to analyse the 3D structure of macromolecules by using subtomogram averaging. Accordingly, we analysed the detected particles by computing subtomogram averages for each ribosome subfamily (see Fig. 4.12). In this way,

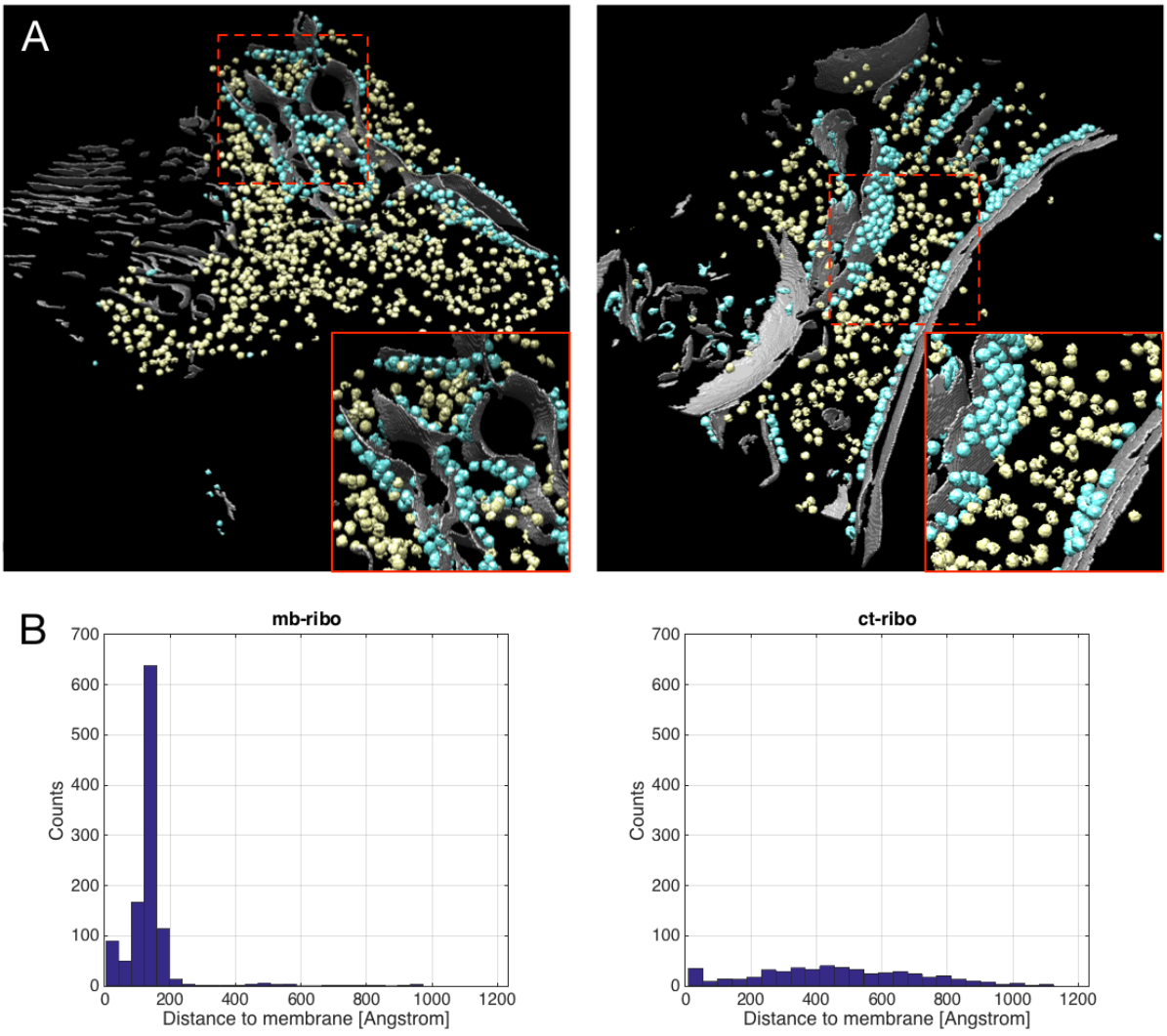

FIGURE 4.10 – (A) 3D voxelwise classification of experimental tomograms, as obtained by our CNN. The classification displays cell *membrane* (in gray), membrane-bound ribosomes (blue), and cytoplasmic ribosomes (yellow). (B) Distance to membrane histograms of detected ribosomes, on the left for *mb-ribos* and on the right for *ct-ribos*.

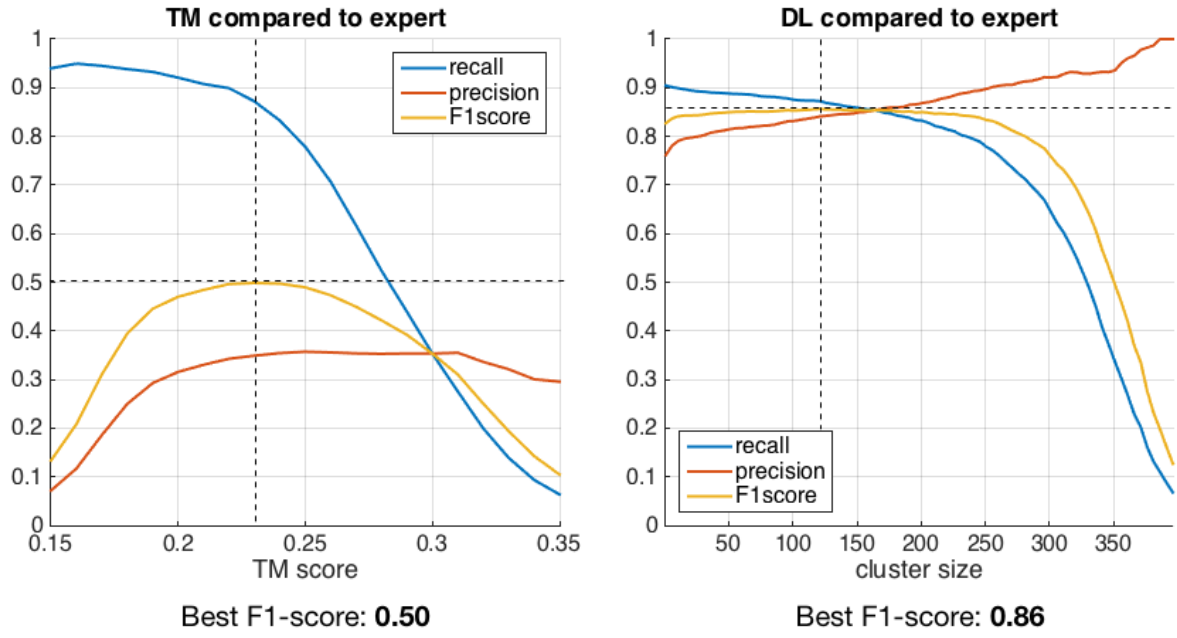

FIGURE 4.11 – Overlap with expert annotation w.r.t. method threshold parameter : on the left for TM, on the right for our DL approach.

we compare the subtomogram averages obtained from the DL detections and the expert annotations. The same process has been used to compute all averages, namely fast rotational matching [Chen et al., 2013] (see Sec. 4.3.5). The averages of *mb-ribo* are composed of two densities : the average ribosomal density and the average membrane density. The intensity level of the membrane density varies with the proportion of *mb-* and *ct-ribos* in the average ; the higher the intensity, the higher the number of *mb-ribos*.

As expected, the average membrane density computed from the set S_{DL} of *mb-ribo* particles has a high intensity, while the average membrane density computed from the set of *ct-ribo* particles detected by our DL approach, is non-existent. Consequently, our method makes little to no confusion between *mb-* and *ct-ribos*. We notice that the average ribosomal density is similar in both ribosome subfamilies, the difference lying in the neighborhood of the link with the membrane. Therefore, DL is able to efficiently represent the geometric structure of the macromolecule, and at the same time to capture the local context and interactions of the macromolecule with the environment.

Next, we computed the Fourier shell correlation (FSC) for obtained subtomogram averages (see Fig. 4.12). According to Fig. 4.12, the best resolution has been obtained with the expert average : 24Å versus 24.7Å and 32.7Å for the DL subclasses *mb-ribo* and *ct-ribo* respectively. Our method therefore allows to achieve a resolution comparable to an expert. As for the *ct-ribo* average the resolution is lower, most likely because the *ct-ribo* annotations used for learning are of lower quality (see Sec. 4.3.1). Even though resolutions for the expert and DL *mb-ribo* averages are very close, the poorer resolution values of the DL averages can be caused by two main factors :

H_1 : The averages contain potential false positives, suggesting that our method probably made a few

mistakes.

H_2 : The averages contain particles with a low quality, suggesting that our method found supplementary particles, missed or discarded during the annotation process.

In what follows, we performed further investigations to check the two hypotheses H_1 and H_2.

First, we decided to align and average the *mb-ribo* particles of the set $S_{DL} \setminus S_E$, i.e. found by DL but absent in the expert annotations (see Fig. 4.13). If no clear signature of ribosome density appears in the average, hypothesis H_1 is valid. On the contrary, if we observe a ribosome patterns in the average, we can conclude that our DL method found additional ribosome particles potentially discarded by the experts (hypothesis H_2). Nevertheless, note that the two hypotheses are not mutually exclusive.

In Fig. 4.13, we display the resulting subtomogram average denoted $\mathbf{A}_{DL}$ computed from 356 detected particles. Since it is not guaranteed that all the particles involved in the average are actual *mb-ribos*, we evaluated the difference between $\mathbf{A}_{DL}$ and the average $\mathbf{A}_{DL}^{ref}$ computed from 356 true *mb-ribos* (expert annotations) randomly picked from the the set $S_E \cap S_{DL}$. Also, in order to check if $\mathbf{A}_{DL}$ is not biased by the reference template used for subtomogram alignment (see Sec. 4.3.1) as described in [Henderson, 2013], we computed another average denoted $\mathbf{A}_{DL}^{\circ}$ from 356 subtomograms picked from random positions. We compared $\mathbf{A}_{DL}$, $\mathbf{A}_{DL}^{ref}$, and $\mathbf{A}_{DL}^{\circ}$ visually and by estimating the underlying resolution (see Fig. 4.13). We notice that $\mathbf{A}_{DL}$ has a lower resolution (36.4Å) than $\mathbf{A}_{DL}^{ref}$ (33.5Å). It means that $\mathbf{A}_{DL}$ probably contains false positives and/or very noisy instances. Nonetheless, $\mathbf{A}_{DL}$ has a higher resolution than $\mathbf{A}_{DL}^{\circ}$ (48.6Å), suggesting that the reference template bias is not significant. Moreover, a ribosome pattern visually appears in $\mathbf{A}_{DL}$. This suggests that our DL approach has actually found *mb-ribos* that have been missed during the annotation process.

To be fair, we applied a similar comparison to $S_E \setminus S_{DL}$, i.e. the set of *mb-ribos* annotated by the expert but missed by DL. As before, we obtain $\mathbf{A}_E$ from the 220 objects belonging to $S_E \setminus S_{DL}$, $\mathbf{A}_{DL}^{ref}$ by randomly sampling 220 *mb-ribos* from $S_E \cap S_{DL}$, and $\mathbf{A}_{DL}^{\circ}$ from 220 random positions. The obtained resolutions are very similar to what is achieved with DL. Here again, $\mathbf{A}_E$ has a lower resolution (37.6Å) than the reference $\mathbf{A}_E^{ref}$ (34.2Å). It is noteworthy that in both cases, the *mb-ribos* from $S_E \cap S_{DL}$ lead to a better resolution than the complement sets $S_E \setminus S_{DL}$ and $S_{DL} \setminus S_E$. This observation illustrates well what has been discussed earlier, namely that combining sets obtained from different methods allows to attribute confidence levels. The lower resolution of $\mathbf{A}_{DL}$ and $\mathbf{A}_E$ suggest that the sets $S_E \setminus S_{DL}$ and $S_{DL} \setminus S_E$ contain more heterogeneity, and thus potentially include rare conformations.

This set of results emphasizes that TM and DL can be combined to better investigate cryo-tomograms. The set of common objects found by the two methods enables to focus on detections exclusively found by the experts or by DL. In addition, we show that, while the FSC curves obtained with DL are below the curve obtained with expert annotations (see Fig. 4.12), it is risky to decide that the detected macromolecules are not ribosomes (see Fig. 4.13). The estimated resolutions are lower mainly because the particles involved in the subtomogram averaging are corrupted by noise and other sources of signal degradation. The supplementary noisy particles need to be further examined by experts since they may be valuable *mb-ribo* candidates. Finally, it appears that the number of actual *mb-ribos* is higher than expected : in our test set, we have detected $+20.5\%$ of *mb-ribos* when compared to the S_E set. In summary, DL found additional noisy *mb-ribos* that were missed or discarded during the annotation process.

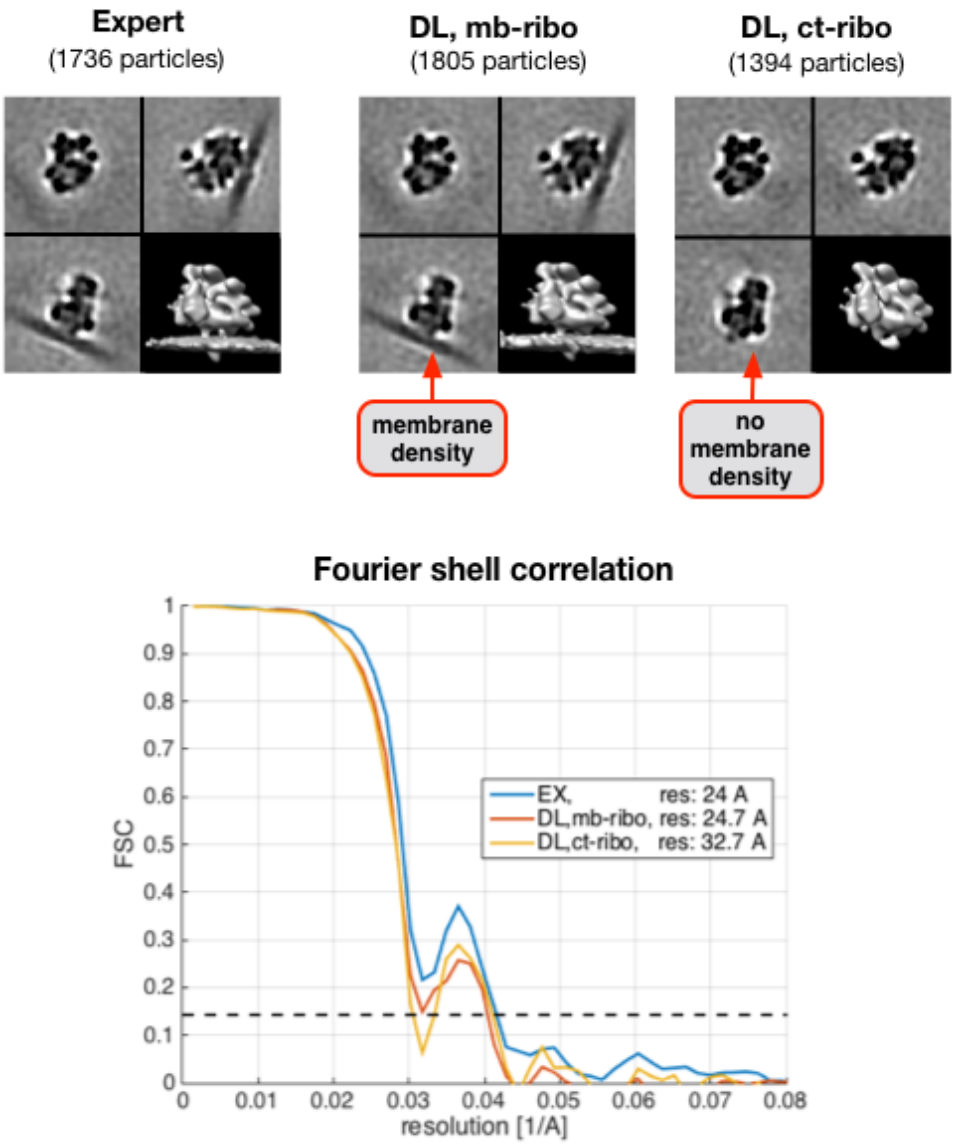

FIGURE 4.12 – On top, the subtomograms obtained from, the expert annotations, the DL detections for *mb-ribos* and *ct-ribos*, respectively. All averages have been obtained with the same alignment procedure and parameters. For visualization purpose, the averages have been low-pass filtered at 40A resolution. At the bottom, the corresponding gold-standard FSC curves with estimated resolutions.

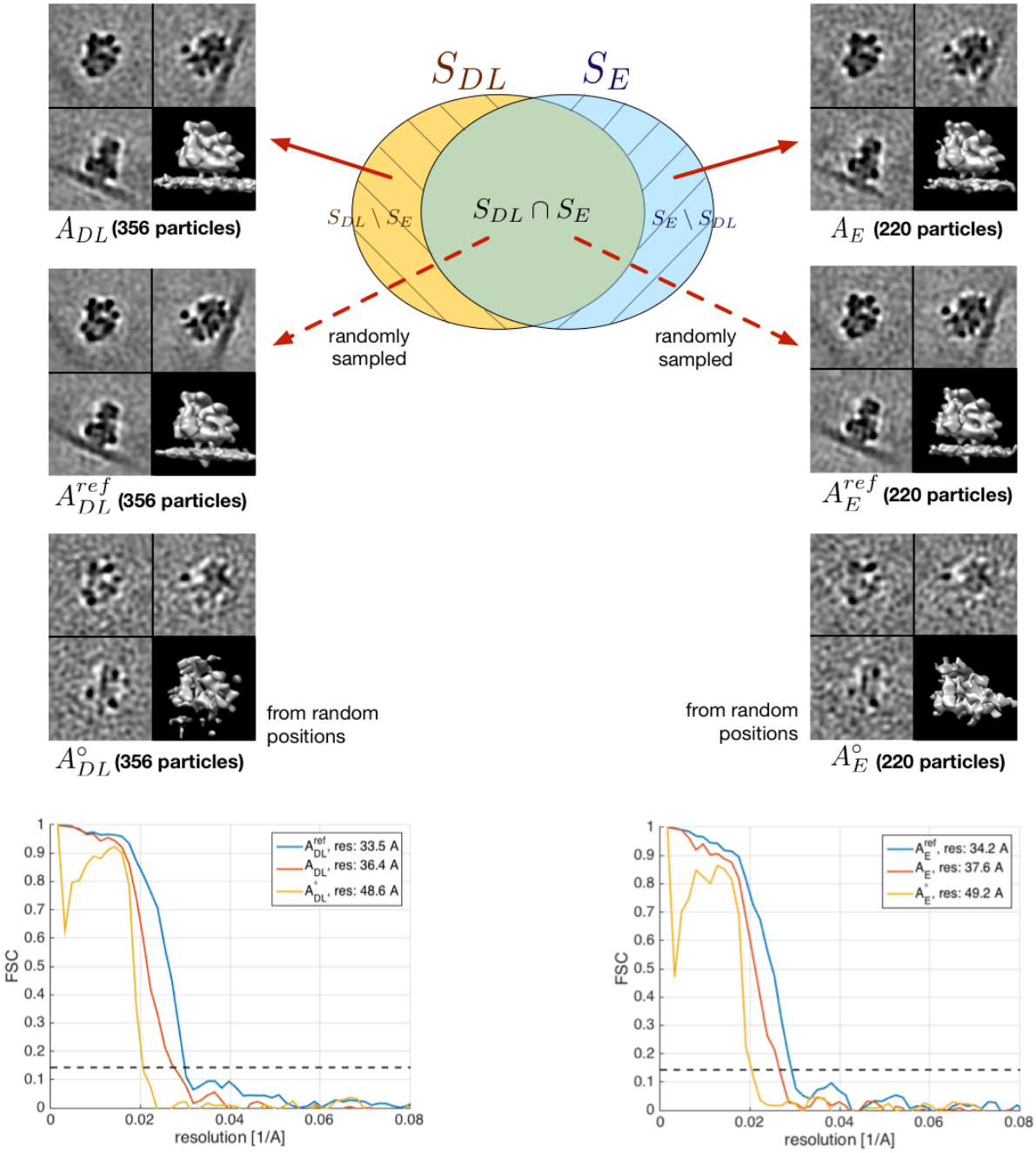

FIGURE 4.13 – Top-middle : diagram representing the overlap between *mb-ribo* sets S_{DL} and S_E. The emanating arrows represent from which sub-set the displayed subtomogram averages originate. Bottom : FSC curves for each subtomogram average.

4.3.4 Detecting proteasomes : preliminary results

In this section, we explore the ability of our method to identify proteasomes. For dataset #2, we have annotations for 2350 proteasomes, which we add to the set of annotations described in Section 4.3.1. In the end, we train our network for five classes : background, *mb-ribos*, *ct-ribos*, membrane and proteasomes. We split the proteasome set as follows : 1853 for training, 187 for validation and 310 for testing. Train and validation sets are sampled from 55 tomograms, and the test set is sampled from 3 other tomograms. The training procedure is parametrized as described in Section 4.2.2.

In this situation, we have highly imbalanced classes. First, the proteasome class is under-represented in terms of instances. In the training set, we have 5971 *mb-ribos*, but only 1853 proteasomes. Second, a proteasome has a smaller volume (2327 voxels) than a ribosome (3203 voxels). Finally, the amount of training voxels is small compared to other classes (4.10^6 voxels for proteasomes versus 19.10^6 voxels for *mb-ribos*). We therefore adapted our training procedure by including a *re-sampling* procedure (see Section 3.3.4). For each randomly sampled training batch, we make sure that it contains an equal amount of instances for each class. Without the re-sampling, the proteasome could not have been detected.

In the end, we achieved a plausible segmentation. Figure 4.14 illustrates an obtained segmentation for all five classes, and gives an insight into the spatial distribution of multiple macromolecule types. We can observe that some proteasomes are tethered around nuclear pores, and some are floating freely in the cytoplasm, which corroborates the study in [Albert et al., 2017]. In Figure 4.15, we overlay our detections with expert annotations, which reveals that a majority of the annotated proteasomes have been segmented at least partially. Also, we notice that a number of segmented proteasomes are not part of the annotations. Part of these additional detections may be false positives, but as our findings in Section 4.3.3 suggest, part of them may be proteasomes that have been missed during the annotation process.

As proteasomes are elongated objects (i.e anisotropic), unlike ribosomes which have a spherical shape (i.e. isotropic), the center position of the proteasome is not well accurately defined. Also, as sometimes only parts of proteasomes are segmented, several positions generated by our clustering step are shifted w.r.t the annotations. Therefore a finer analysis of the segmentation map is needed when studying objects with anisotropic shapes. For now, we can not provide faithful *Recall*, *Precision* and *F1-score* curves as for the *mb-ribos*, which have a more simpler shape. Nonetheless we showed that proteasomes have been identified (see Figure 4.15), and that our method can be applied to detect macromolecules of different types.

4.3.5 Implementation details of the DL (3D CNN) software

To implement our 3D CCN method, we used Keras [keras.io], an open-source toolbox written in python and using the Tensorflow framework. Our code is available on [gitlab.inria.fr/serpico/deep-finder].

As to template matching and subtomogram averaging, we used the PyTom toolbox [Hrabe et al., 2012]. We used the in Pytom implemented fast rotational matching routine [Chen et al., 2013] for subtomogram alignment. The alignment has been performed with respect to a reference template (see Section 4.3.1).

For 3D visualizations, we used Chimera [Pettersen et al., 2004].

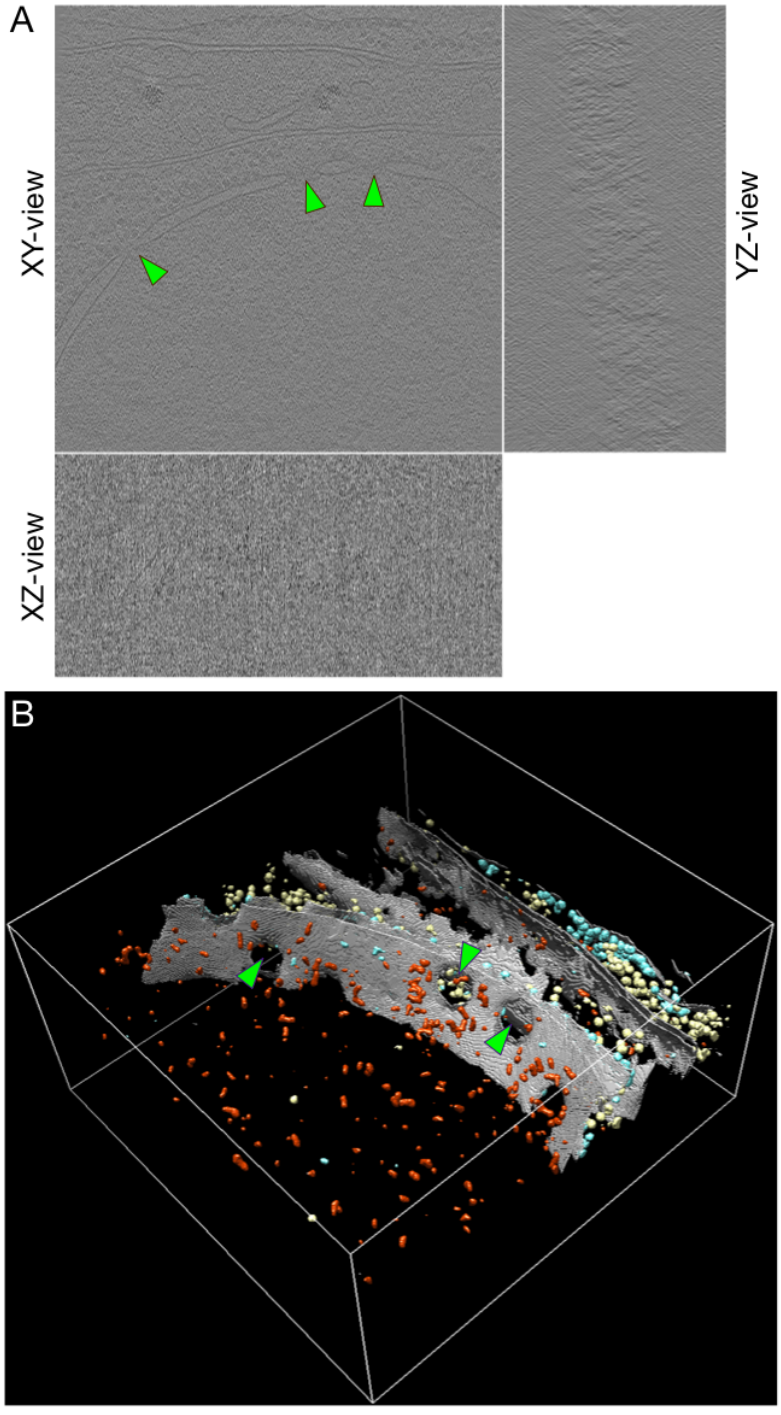

FIGURE 4.14 – (A) Experimental tomogram of a Chlamydomonas Reinhardtii cell (ortho-slices), and (B) the segmentation, as obtained by our CNN. Here we trained the CNN with an additional class : the proteasome (in red). In the end, the CNN has been trained with a total of 5 classes : membrane-bound ribosome (blue), cytoplasmic ribosome (yellow), proteasome (red) and cell membrane (gray). The green arrows indicate locations of nuclear pores, visible in both the tomogram and the segmentation.

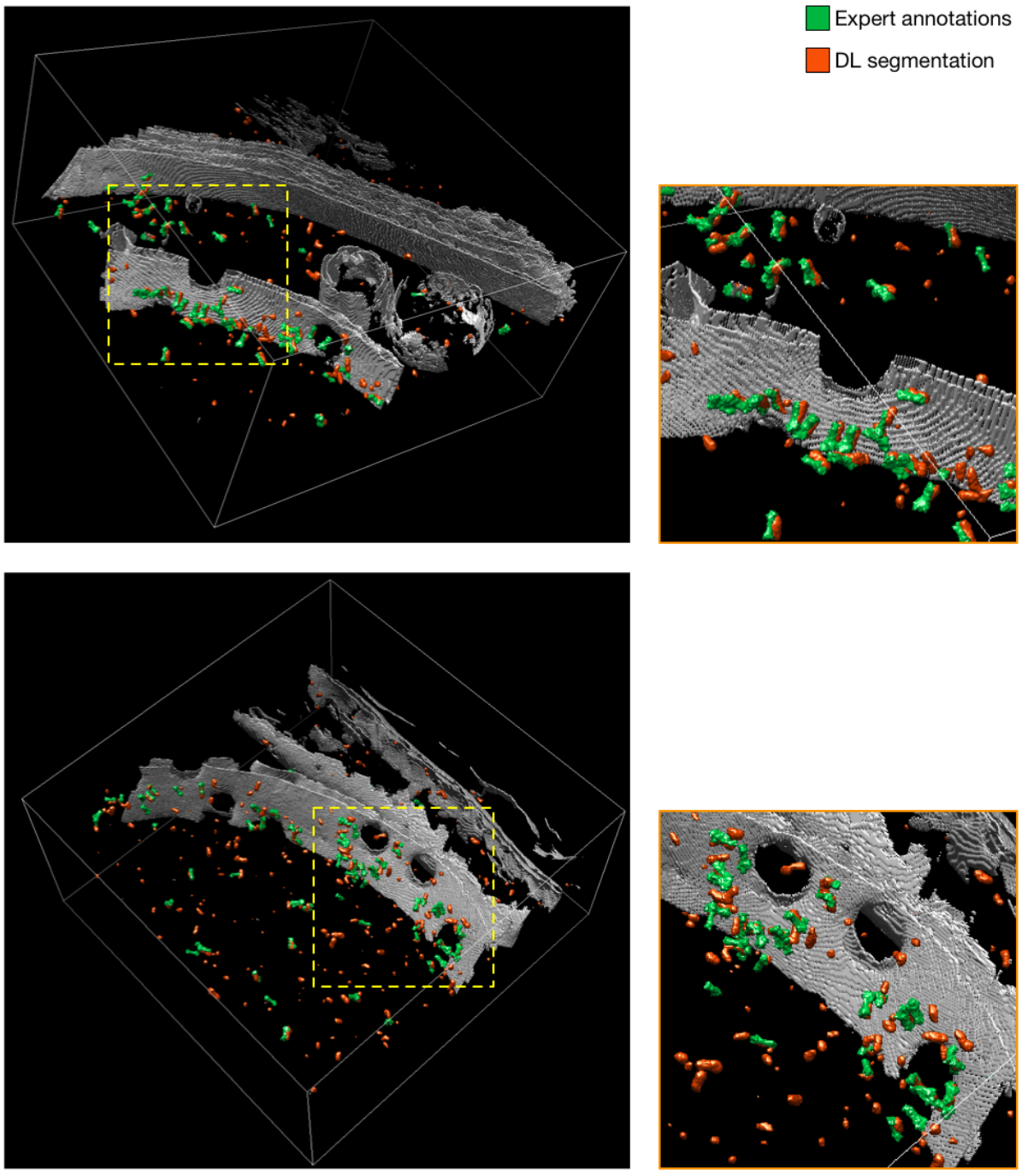

FIGURE 4.15 – Visualizing the overlap between the proteasomes detected by our CNN (in red) and the proteasomes annotated by an expert (in green). On the left an overview of the 3D segmentations, on the right a zoomed in visualization.

4.4 Discussion

We proposed a 3D CNN framework for voxelwise multiclass classification and particle localization in 3D cryo-ET. We showed on synthetic data that our DL method has superior localization performance than TM. On real data, our DL method is able to discriminate two binding states of ribosomes and to segment cell membranes. We achieved 86% of overlap with expert annotations. Also, when applying subtomogram averaging to the detections, we obtain a resolution comparable to the expert. The supplementary membrane ribosomes found by DL and missed during the annotation process, can be further inspected since the related set is small. While it is established that TM is widely applied in cryo-ET to successfully detect various macromolecular complexes, as of now it has mainly been used to detect relatively large macromolecules like ribosomes. It is worth noting that DL found ribosomes that where missed by the expert annotation process (assisted by TM), and conversely. Consequently both methods can be combined to investigate more efficiently cryo-tomograms. In particular, more effort can be made to analyze the particles detected by either DL or TM, that is when the algorithms are not in agreement. TM and DL can also be combined in a virtuous manner as follows : experts can perform a first round of detection with the routine tools, then train our method from the resulting detections ; in a second round, our DL framework can be applied to check the missed objects and increase the size of databases.

To our knowledge, TM is not usually applied to detect more than two or three classes/subclasses on the same dataset. However, cryo-electron tomograms contain much more information. They offer 3D views of the whole macromolecular environment, albeit hardly discernable from noise and artifacts. Therefore more powerful pattern recognition and machine learning techniques are needed to analyse contents and extract information. DL has been shown to be able to handle the spectacular amount of 1000 classes [Krizhevsky et al., 2012]. Also, it is invariant to diffeomorphisms [Mallat, 2016] and then able to cope with non-rigid, elastic deformations within a class. A major disadvantage of TM is that it can only handle rigid views of an object, which is problematic knowing that many proteins (e.g. proteasomes) have a high structural variability. Thus, provided that DL has been trained with enough representative examples capturing shape variability, the CNN should be able to learn different conformations of the same macromolecule.

CHAPITRE 5

GENERATING SYNTHETIC TRAINING SETS FOR DEEP LEARNING IN CRYO-ET

The purpose of this Chapter is to propose new insights for the application of deep learning methods to cryo-ET data. In the previous chapter, we have shown that convolutional neural network can be used to segment 3D cryo-ET images and to localize macromolecules with very good performance. Nevertheless, a large amount of expert annotations are required to get these results. Manually annotating data can be a very time consuming and tedious task, involving the application of template matching and subtomogram classification procedures ; each processing step being supervised by experts. Here we explore alternative strategies to overcome complex annotation process.

First, we explore the concept of generative adversarial networks [Goodfellow et al., 2014] for data augmentation. This would allow to reduce the amount of annotations needed for training a deep neural network. Furthermore, we investigate a novel strategy for generating synthetic training sets, based on artistic style transfer [Gatys et al., 2015]. Finally, we propose an original unsupervised subtomogram classification method, based on transfer learning [Yosinski et al., 2014]. The two last methods require no manual annotation. We achieve a lower level of supervision by benefiting of the knowledge stored in the PDB data bank, in the form of resolved macromolecule density maps. For all the proposed approaches, we evaluate performance on experimental data (the Chlamydomonas dataset, see Section 4.3.1).

5.1 Data augmentation with generative adversarial networks

5.1.1 An introduction to generative adversarial networks

Generative adversarial networks (GANs), originally proposed in [Goodfellow et al., 2014], provide a way of learning data distributions while requiring minor supervision. This is achieved by training simultaneously a pair of competing networks, the *generator* G and the *discriminator* D. Together, they allow to implicitly represent high-dimensional data distributions p_{data}. Once trained, the generator G provides a way of sampling from the modeled distribution p_{model}, and the discriminator allows to determine if a sample originates from p_{data}. A tool such as GANs is of particular interest, knowing that most inference problems can be addressed through estimating conditional distributions, according to Baye's theorem. Therefore GANs may be useful in a variety of applications, including image classification, image segmentation and image synthesis.

A common analogy to understand the learning process of GANs is to consider D as an *art expert*,

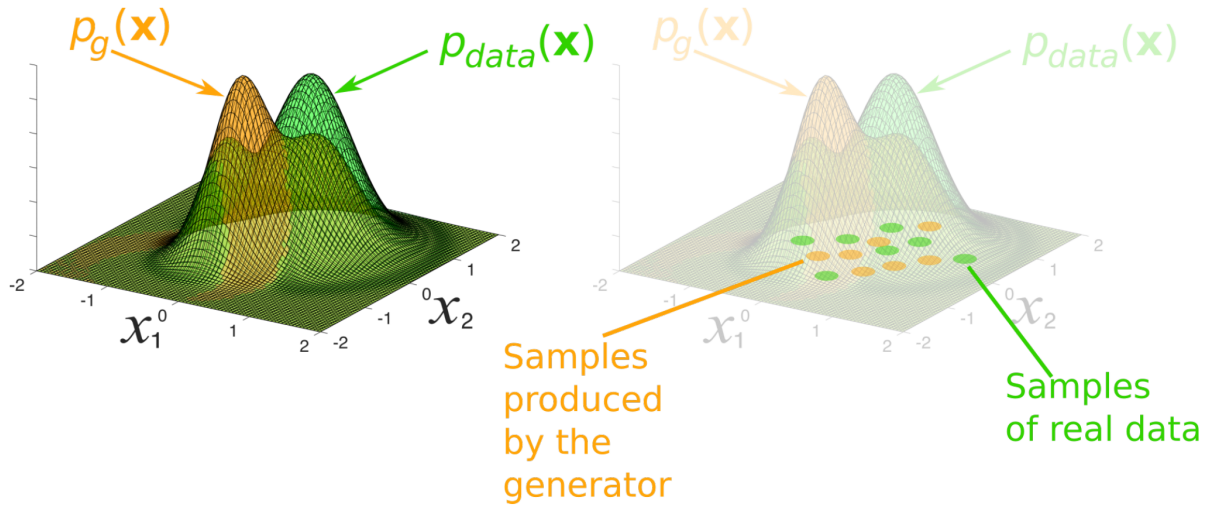

FIGURE 5.1 – During GAN training, the data distribution p_{data} is captured. The learned distribution p_g is modeled by the weights of networks G and D. On the one hand, the generator G learns to generate samples from p_g, while matching p_g to p_{data}. On the other hand, the discriminator D learns to identify if a sample originates from p_g or p_{data} (source [Creswell et al., 2017]).

and of G as a *counterfeiter*. The goal of G is to create realistic counterfeits (i.e. synthetic images), while the aim of D is to tell counterfeits apart from real images. The key point is that the generator has no access to real images, its only way of creating realistic counterfeits is to dialog with the discriminator. The error needed to train the discriminator, is provided by the ground truth of whether the images are synthetic or real. This error signal is then passed down to the generator, enabling it to learn how to generate better counterfeits. The only annotations needed in this framework is if images are real or generated (i.e. "fake"), and are therefore straightforward to obtain.

Formally, the generator is a function mapping from a random noise vector z belonging to some latent space to an image y : $G : z \rightarrow y$. The discriminator maps from image data to the probability that the image belongs to the real data distribution p_{data} : $D : x \rightarrow (0, 1)$. The images generated by G are drawn from the learned distribution p_{model}, the objective being to satisfy $p_{\text{model}} = p_{\text{data}}$. In this case, D will be *maximally confused* and predict 0.5 for all inputs.

The training procedure amounts to solving the following optimization problem :

$$\max_D \min_G \mathcal{L}_{\text{GAN}}(G, D), \tag{5.1}$$

where

$$\mathcal{L}_{\text{GAN}}(G, D) = \mathbb{E}_{p_{\text{data}}}[\log D(y)] + \mathbb{E}_{p_{\text{model}}}[\log(1 - D(G(x, z)))]. \tag{5.2}$$

The criterion is the standard cross-entropy cost for a binary classifier, where the classifier is trained on two data batches : one from the training data (i.e. label is 1) and one generated by G (i.e. label is 0).

In practice it is not easy to successfully train GANs as described above. The right balance between D and G has to be found during training, where both networks are trained sequencially. For example, if D

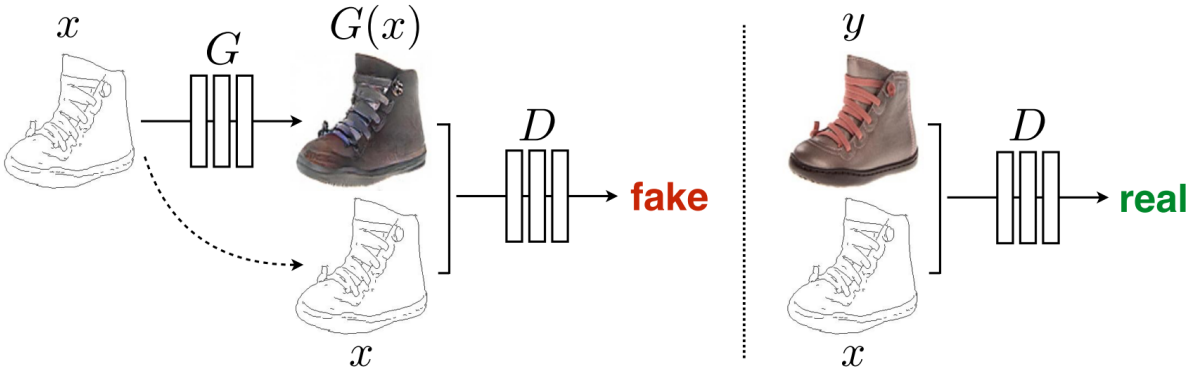

FIGURE 5.2 – Illustration of the conditional GAN training process (pix2pix framework). In this example, the cGAN is trained to map from edges to photos. The discriminator D learns to discriminate between real and fake (i.e. generated) {edge,photo} pairs, while the generator G learns to fool D. As opposed to unconditional GANs, both D and G have access to the edge map. Reproduced from [Isola et al., 2017].

gets too successful too early, the error signal vanishes and G can not learn anymore. Another common problem is known as *mode collapse* [Goodfellow, 2016], where rather than learning all of the distribution modes of p_{data}, G only learns one single mode. This results into a generator that always produces the same image, regardless of its input. A few heuristic rules have been proposed to overcome these difficulties, for instance by label smoothing or by disturbing the training of D (e.g. for each training step, train D less than G). Another way is giving the GANs more information to better guide the training. This can be achieved by giving D and/or G access to annotated data (i.e. labels). In that case, we are in the so-called *conditional GAN* (cGAN) framework [Mirza and Osindero, 2014], and the loss function becomes :

$$\mathcal{L}_{\text{cGAN}}(G, D) = \mathbb{E}_{p_{\text{data}}}[\log D(x, y)] + \mathbb{E}_{p_{\text{model}}}[\log(1 - D(G(x, z)))] \tag{5.3}$$

In this Section, we use a cGAN based on [Isola et al., 2017], the so-called pix2pix framework, presented as a general-purpose solution to image-to-image translation problems. Applications of this framework include synthesizing photos from label maps or from edge maps (see Fig. 5.2). The training objective cost is defined as :

$$\max_D \min_G = \mathcal{L}_{\text{pix2pix}}(G, D) + \lambda \mathcal{L}_{\text{L1}}(G), \tag{5.4}$$

where

$$\mathcal{L}_{\text{pix2pix}}(G, D) = \mathbb{E}_{p_{\text{data}}}[\log D(x, y)] + \mathbb{E}_{p_{\text{model}}}[\log(1 - D(G(x)))] \tag{5.5}$$

and

$$\mathcal{L}_{\text{L1}}(G) = \mathbb{E}_{x,y,z}[\| y - G(x) \|_1] \tag{5.6}$$

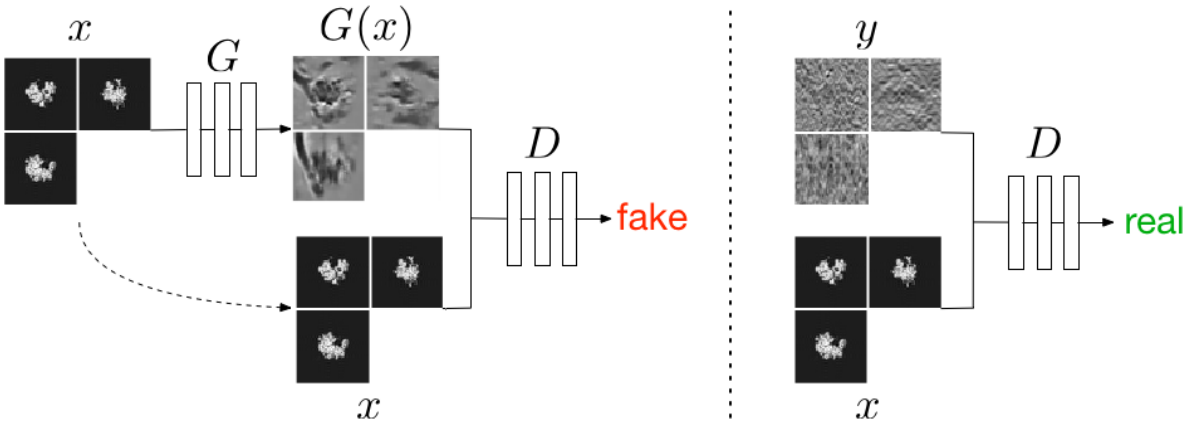

FIGURE 5.3 – Adaptation of the pix2pix framework for generating synthetic 3D subtomograms, given ribosome density maps.

5.1.2 A conditional GAN for subtomogram generation

In this Section, we explore the possibility of using GANs as a data augmentation (see Section 3.3.3) method in the framework of cryo-ET. GANs have recently be used for this purpose for medical imaging [Jin et al., 2018]. Validity of using such generated data as training sets for medical applications is still an open question, as generated image details may not have any biological reality (i.e. so-called "hallucinations").

In what follows, we use the pix2pix framework to learn a mapping between a macromolecule density map and a noisy subtomogram (see Fig. 5.3). The density map may be obtained from the Protein Data Bank (PDB). For training the cGAN, we need to provide pairs of density maps x and noisy subtomograms y. Also, the density map needs to be into register with the macromolecule contained in the subtomograms. Therefore the rotational orientation of the macromolecules is essential, which may be obtained via subtomogram alignment (see Section 1.3.4).

Once trained, G can be used to generate as many synthetic subtomograms as needed, with corresponding ground-truth (see Figure 5.4).

5.1.3 Experimental results

Training the GAN The training set is composed of 1000 {density map, subtomogram } pairs of membrane-bound 80S ribosomes. The density map has been obtained from the PDB data-bank (EMD-1780). The subtomograms are sampled from tomograms of Chlamydomonas Reinhardtii cells (see Section 4.3.1), and have a size of $64 \times 64 \times 64\times$ voxels and a voxel size of 13.68Å. The cGAN has been trained for 48k iterations and with a batch size of 24. As a generator network, we used the architecture we proposed for tomogram segmentation in Chapter 4 (see Figure 4.4). As a discriminator network, we used the architecture defined in Figure 5.5.

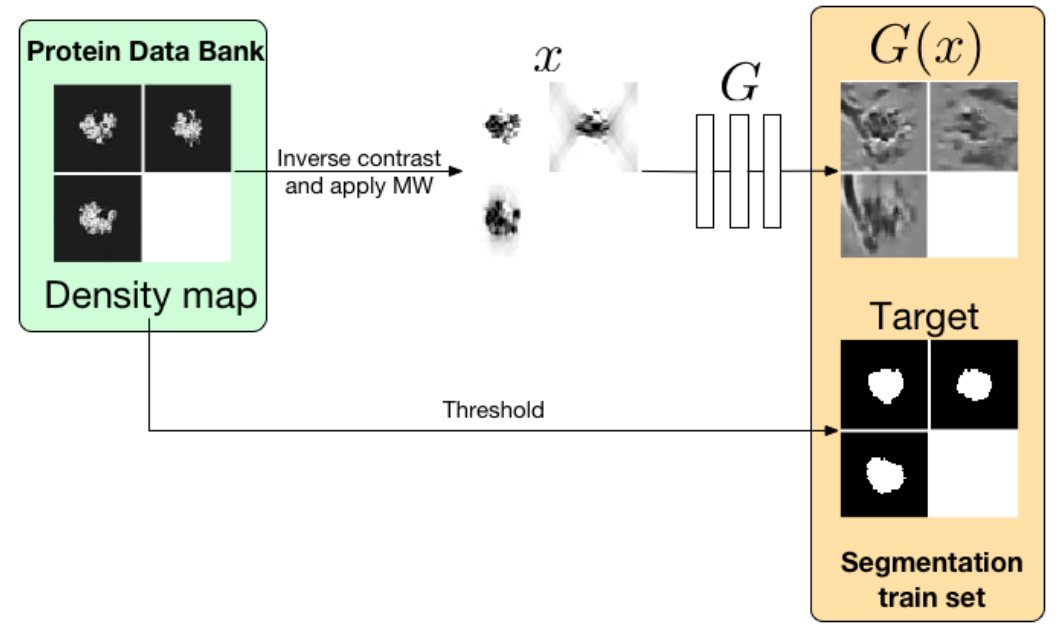

FIGURE 5.4 – Illustration of the GAN-driven data-augmentation process. The generated data is used as a training set for a segmentation task. In order to facilitate the task of the GAN, we pre-process the density maps by already inverting the contrast and applying missing wedge artifacts.

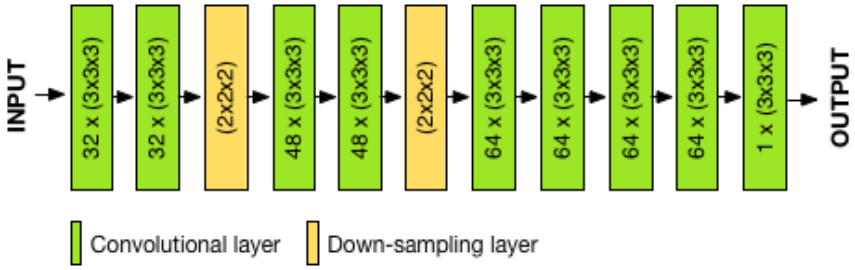

FIGURE 5.5 – Architecture of the 3D discriminator network D. It is a simplified version of the segmentation network described in Figure 4.4, also used here as generator G. In this figure, the convolutional layers are labeled with #filters $\times$ (height $\times$ width $\times$ depth), and the down-sampling layers are labeled with the down-sampling factor in each dimension (height $\times$ width $\times$ depth). All convolutional layers use ReLU as activation function, except the output layer which uses a sigmoid.

Generated subtomograms Let us first visually assess the quality of the ribosome images generated by G. Figure 5.6 displays the progress of subtomogram generation during GAN training. This Figure illustrates how the GAN progressively creates a cellular environment around the ribosome.

For the first 500 epochs, the GAN suffers from mode collapse (see Section 5.1.1) and generates constant images. Then, a ribosome density appears and the GAN starts to progressively generate a cellular environment around the ribosome. Remarkably, there is a white halo around the ribosome (see epoch 578), not present in the input of G. In experimental cryo-ET images, such a halo is an effect of the CTF (contrast transfer function). This suggests that the GAN has learned to mimic the flaws of the electron microscope. Afterwards, a membrane appears next to the ribosome (see epoch 623). Notice that the orientation of the ribosome with respect to the membrane is as found in experimental tomograms, which indicates that the GAN learned the location of the docking point of the ribosome. Finally, the GAN generates neighboring densities (see epoch 1111), which seem to correspond to surrounding macromolecules. This makes indeed sense, given that membrane-bound ribosomes are often surrounded by their peers.

Surprisingly, we notice that the only missing component is noise. The generated tomograms are indeed rather smooth compared to experimental subtomograms (see Fig. B). This confirms the findings of [Isola et al., 2017] : "we observe only minor stochasticity in the output of our nets. Designing conditional GANs that produce highly stochastic output [...] is an important question left open".

Use of GAN for analyzing real data We employed the GAN to generate a training set of 1000 $\{$subtomogram,target$\}$ pairs, as illustrated in Figure 5.4. Next we used this training set to train our segmentation network presented in Section 4.2.1. Finally, we applied the segmentation network to an experimental tomogram from the Chlamydomonas dataset (described in Section 4.3.1). The result is displayed in Figure 5.7.

At first glance, the segmentation in Figure 5.7 seems plausible : the voxels classified as ribosome are organized in clean blobs, and do overlap with contrasted objects in the data. In order to provide quantitative results, we applied the clustering step described in Section 4.2.1 to obtain the ribosome positions. We then compared obtained positions to expert annotations, as described in Section 4.3.2. According to the *Recall* and *Precision* values, the amount of false positives is higher than the amount of false negatives. In fact, for a low cluster size threshold, 80% of annotated ribosomes have been retrieved. However the Precision is low (46%), and the best achieved F1-score is only 56%. These scores are not competitive with the results presented in Section 4.3.3. However, one mentions the following points :

— The segmentation network has been trained with only two classes (ribosome and background).
— The training set is composed of synthetic datasets only. Usually when applying data augmentation, real and synthetic data are mixed.
— We do not use data-augmentation at its full potential, as the training sets is composed of only 1000 examples. In theory we could generate tens of thousands of training examples, or more.

Addressing above points should therefore improve performance. Nonetheless, we could demonstrate that GAN-driven data-augmentation is feasible in the context of cryo-ET data.

In summary, we showed that a cGAN is able to generate cellular context such as neighboring macromolecules and membranes. Moreover, the synthetic subtomograms generated by our cGAN can be

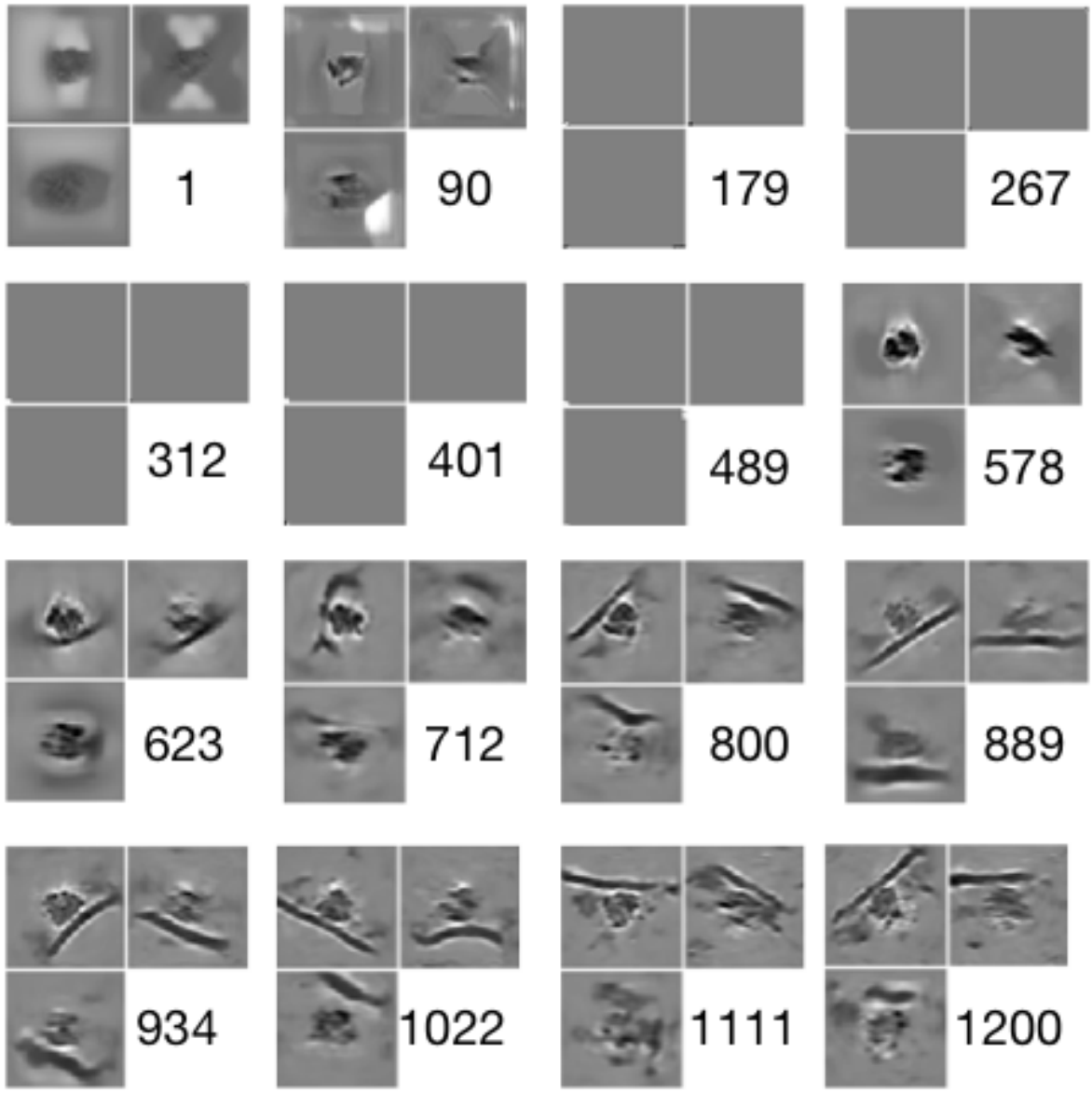

FIGURE 5.6 – Evolution of the quality of the "counterfeit" membrane-bound ribosome during training. The numbers correspond to the training epoch (1 epoch = 1000 iterations) at which the ribosome is generated. This figure illustrates how the GAN progressively generates a cellular context around the ribosome (e.g. membrane, neighboring macromolecules).

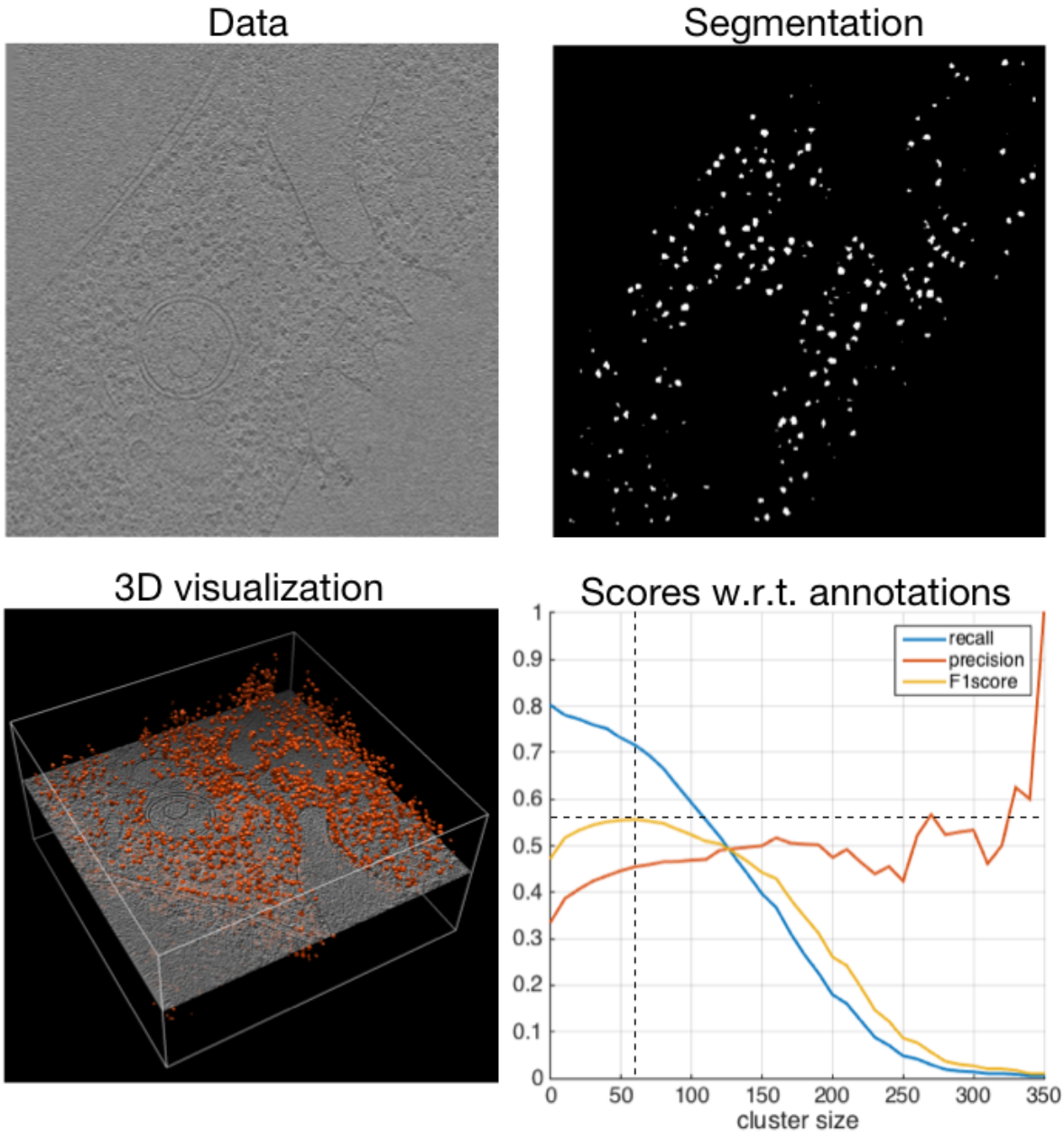

FIGURE 5.7 – Segmentation result on real data. The segmentation network has been trained on synthetic ribosomes generated by a GAN. Top row : tomogram slice and the obtained segmentation. Bottom left : 3D visualization of the segmentation, in red the segmented ribosomes. Bottom right : scores obtained when comparing to the ribosome positions annotated by an expert.

used to train a segmentation network. This segmentation network can then successfully identify ribosomes in real tomograms. Therefore, we provided the proof-of-concept that a cGAN can be used for data augmentation in the context of cryo-ET data.

However, this method still needs annotated training data to learn how to generate subtomograms. In the next Section, we present another strategy for generating synthetic data, which does not rely on annotations.

5.2 Learning noise with artistic style transfer

5.2.1 An introduction to artistic style transfer

Style transfer [Gatys et al., 2015] is an approach which exploits a pre-trained CNN to build separate representations for the *content* and the *style* of an image. The image content refers to what defines its arrangement and semantics (e.g. "a caterpillar smoking a pipe"), and is dependent on spatial information (e.g. edges). The image style is characterized by the type of textures and colors, and is invariant to position. Having separate representations allows to manipulate content and style independently, and to transfer the style of an image I_{style} to the content of an image I_{content} to produce a new image I_{combined}. For example, such a technique can be used to transfer the artistic style of a Van Gogh painting to a photo, as illustrated in Figure 5.8.

The image I_{combined} is then obtained by optimizing a loss function. In [Gatys et al., 2015], the authors define two loss functions, $\mathcal{L}_{\text{content}}(I_{\text{content}}, I_{\text{combined}})$ and $\mathcal{L}_{\text{style}}(I_{\text{style}}, I_{\text{combined}})$. Minimizing both loss functions jointly allows to produce an image combining the content of I_{content} and the style of I_{style}. Both loss functions are defined using a pre-trained neural network, such as the VGG network [Simonyan and Zisserman, 2015] trained on the ImageNet database [Deng et al., 2009] for object recognition. Such networks encode a robust high-level representation of the image. As such, the image content is well represented by the location-specific activations in the feature-maps produced by the different layers of the network. Therefore, $\mathcal{L}_{\text{content}}$ can be defined as explained below.

Formally, let the N feature maps of size M be stored in a matrix $F \in \mathbb{R}^{N \times M}$, where F_{ij} is the activation of the i^{th} filter at position j. The content loss is then defined as the squared-error between the two feature representations F and P :

$$\mathcal{L}_{\text{content}} = \frac{1}{2} \sum_{i,j} (F_{ij} - P_{ij})^2. \tag{5.7}$$

As for the image style, its representation is based on joint activation patterns, that is the correlations between the activations of different feature maps. These correlations, capturing the joint occurence of features, form the so-called Gram matrix G. This representation finds its roots in [Julesz, 1962], where the author shows that images with the same joint statistics have textures that are indistinguishable for humans. By considering the Gram matrices $G^l \in \mathbb{R}^{N_l \times N_l}$ of multiple layers l, a stationary, multi-scale representation is obtained, capturing texture information. The inner product G^l_{ij} between the vectorised

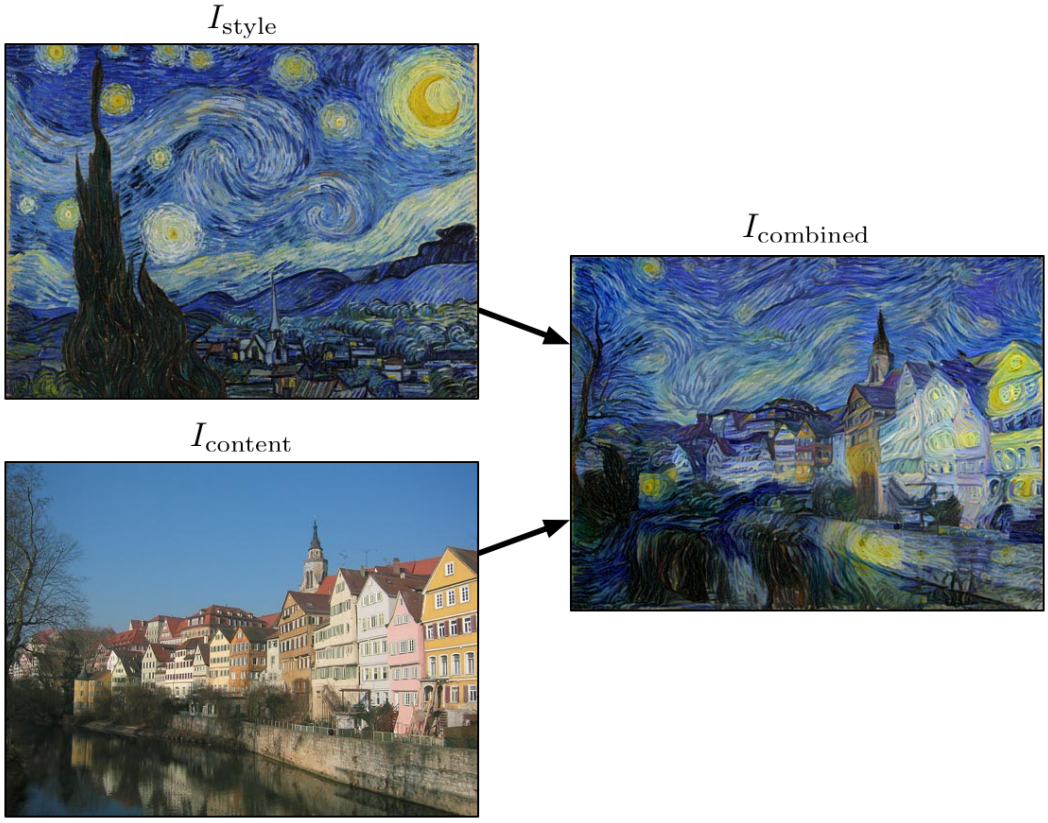

FIGURE 5.8 – Separate representations for image style and image content allow to transfer the style of a painting to the content of a photo (source : deepart.io).

feature maps i and j is defined as follows :

$$G^l_{ij} = \sum_k F^l_{ik} F^l_{jk}. \tag{5.8}$$

The contribution of layer l to the style loss $\mathcal{L}_{\text{style}}$ is the normalized squared-error between the two Gram matrices G and A :

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G^l_{ij} - A^l_{ij})^2, \tag{5.9}$$

and the total style loss is :

$$\mathcal{L}_{\text{style}} = \sum_l E_l. \tag{5.10}$$

Finally, I_{combined} is obtained by minimizing the sum of the content and style terms :

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{content}} + \lambda_2 \mathcal{L}_{\text{style}} \tag{5.11}$$

where λ_1 and λ_2 are weights used to emphasize either the content or the style, respectively. In practice, the representations for style and content are not completely independent. For example, it may happen that the style representation captures content elements. In this case some of the content of I_{style} is being transfered as texture to I_{combined}, and the resulting image becomes less perceptually meaningful. By calibrating the weights λ_1 and λ_2 this effect can be reduced.

5.2.2 Creating a plausible cryo-ET dataset

Our objective is to use *style transfer* to generate an artificial training set for a segmentation task. The idea is to project experimental noise and artifacts on images where the content is known. These content images can be arranged using macromolecule density maps from the PDB databank (see Figure 5.9). The user can choose the macromolecule classes according to what type of cell is being imaged. As the content is known, the segmentation targets are obtained automatically, and no effort is spent to produce annotations.

The key concept in this work is to consider experimental noise as an image style. We show that this noise can be captured by the style representation described in the previous section. This allows to generate realistic artificial tomograms for which the ground truth is known. We can therefore build training sets for other machine learning tasks, as large as needed.

5.2.3 Experimental results

The content images have been generated in a similar way than in Section 4.3.1, using atomic densities from the PDB databank. Five eukaryotic macromolecule classes have been chosen to constitute the content : 80S ribosome (4V88), 60S ribosome (4V88), 40S ribosome (4V88), proteasome (1RYP) and chapernonin (5GWA). In addition, we have introduced three basic object classes (spheres, ellipses and discs) in order to mimic cell content like membrane components, small molecules and particles. These densities were randomly arranged (random positions and orientations) in volumes of size $100\times100\times100\times$

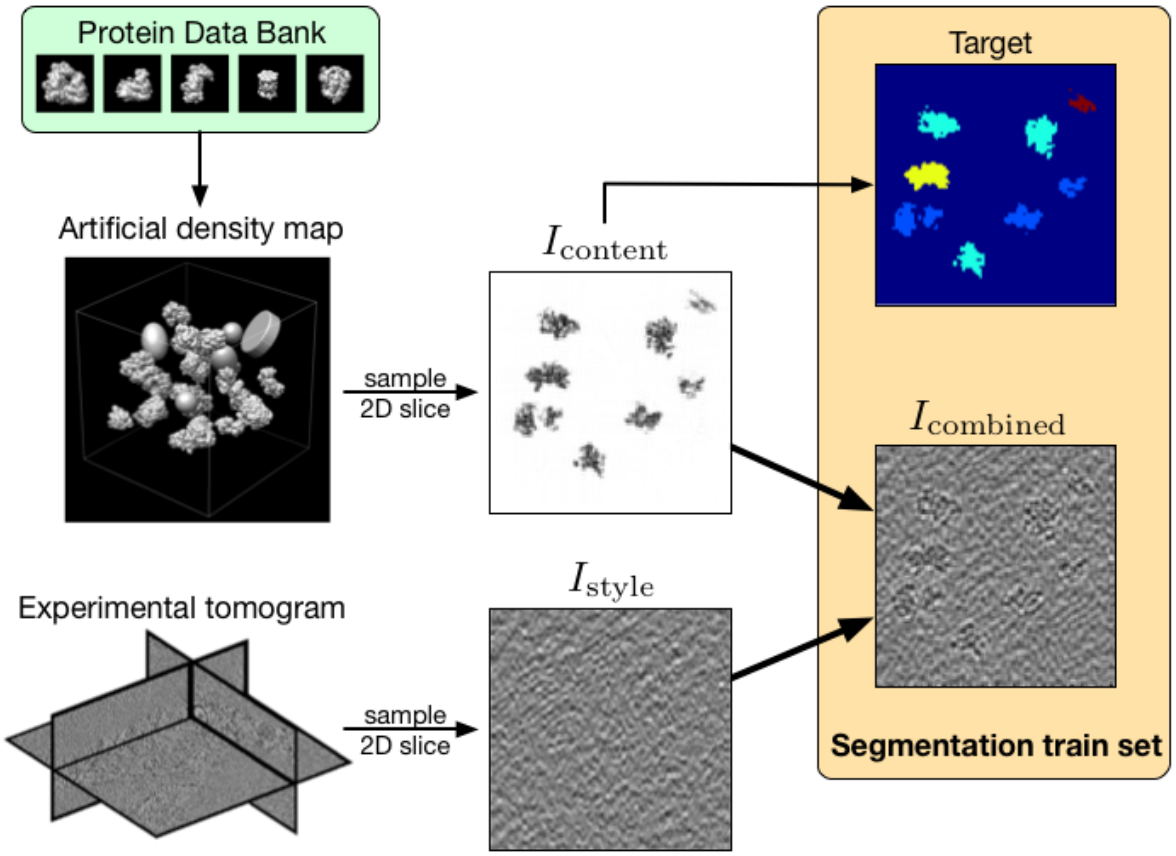

FIGURE 5.9 – Using style transfer to generate an artificial training set for segmenting cryo-ET images.

voxels in order to simulate a crowded cell environment. Training targets for the segmentation task are generated from the content images (see Fig. 5.9). In the end, the dataset is constituted of six classes : 5 macromolecule classes and the background class (the basic objects are labeled as background).

The style images are sampled from experimental tomograms. We make sure that the style images contain noise only, to avoid content elements (e.g. membrane) being captured by the style representation.

First, we tried to apply 3D style transfer. As explained in Section 5.2.1, the content and style representations are based on pre-trained networks. The quality of these representations depend on the quality of the pre-trained network, i.e. the semantic knowledge encoded in its weights. As pre-trained 3D networks are not yet in open access (such as pre-trained 2D networks), we employed our segmentation network (see Figure 4.4) trained on the Chlamydomonas dataset (described in Section 4.3.1). However, we found that in this case we could not reproduce experimental noise. We believe that the representation encoded in the 3D segmentation network is not complex enough, given that it has been trained on only 4 classes. As a comparison, in [Gatys et al., 2015] the authors use the VGG network [Simonyan and Zisserman, 2015], pre-trained on ImageNet [Deng et al., 2009] which consists of more than 10 million images and 1000 classes. Nontheless, in order to provide a proof of concept, we decided to carry out this experiment in 2D, and employ the same pre-trained network as in [Gatys et al., 2015]. The 2D images are obtained by sampling slices from the 3D volumes, as illustrated in Figure 5.9.

Generating the synthetic dataset Using the style transfer method described in previous section, we generate a training set of 1600 2D segmentation examples, and a validation set of 400 examples. The generated images are visually plausible, as can be observed in Figure 5.9. However, there is no direct way of evaluating the quality of these images. We therefore train a 2D version of our segmentation network (described in Section 4.2.1) on these images, and investigate if this network is able to successfully segment experimental data. We use as test set the 2D slices of the Chlamydomonas dataset (described in Section 4.3.1), and evaluate the segmentation for ribosomes only. The evaluation is achieved in terms of *Recall*, *Precision* and *F1-score*, by comparing the segmentations to a ground truth, obtained from expert annotations (as described in Figure 4.5).

Segmentation of real data As can be observed in Figure 5.10, the global segmentation quality is quite limited. The Precision is low (around 0.4), and the Recall is even lower (around 0.1). However, as can be seen in the figure, a few ribosomes were quite well segmented. Therefore, the synthetic training set is realistic enough to at least partially segment real ribosomes. We notice that segmentation errors are due more to false negatives than false positives. This suggests that the training is not diverse enough to represent all the variability of ribosomes as found in cryo-ET. Leads for improving this approach therefore includes generating more training data. Also, it would be helpful to have access to robust pre-trained 3D networks.

The presented method has the advantage of not requiring annotations. However, it is dependent on the availability of target macromolecules in the PDB data bank. In the next Section, we present a strategy enabling unsupervised classification of unknown macromolecules.

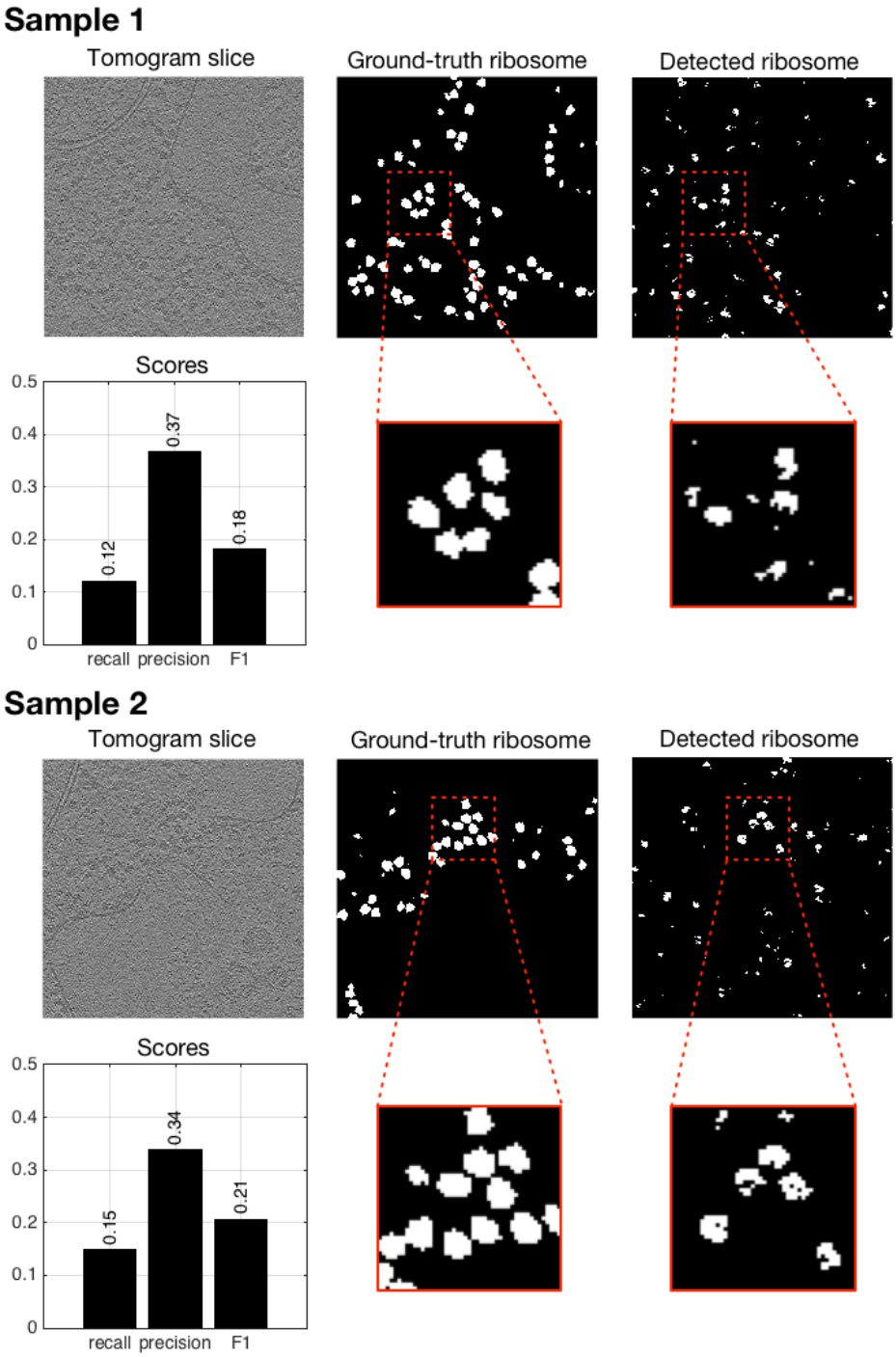

FIGURE 5.10 – After being trained on an artificial training set generated with style transfer, the segmentation network is applied on experimental data. Results are shown for two different samples.

5.3 Unsupervised classification with transfer learning

5.3.1 Transfer learning

Transfer learning [Yosinski et al., 2014] is an approach where the representation learned for a setting A is used for a setting B. For example, *style transfer* (see Section 5.2) is a special case of transfer learning. The main assumption is that the same representation may be useful for both setting. This concept is well understood in supervised image classification, and arises from the observation that the first layers of CNNs (near the input) tend to resemble Gabor filters or color blobs, regardless of the task at hand. In fact these first layers, characterizing low-level features such as edges and shapes, are relevant for categorizing dog breeds, but also for categorizing mushroom species. Various visual categories share the same low-level features, and are therefore referred to as *general* features. On the other hand, the last layers (near the output) of a classification network greatly depend on the task, and are therefore called *specific* features.

A useful application of this feature transferability is when the available training set is small and the risk of over- or underfitting exists. In this case, one can benefit from networks trained on a huge data sets (e.g. ImageNet [Deng et al., 2009]), by *freezing* the weight-values of the first layers and therefore train only the last, task-specific layers. In this way only few weights have to be trained, i.e. the training procedure has less degrees of liberty. Another possibility is *fine-tuning*, where instead of freezing transferable weights, they are used to initialize a network. The consequence is that the network parameters are closer to the target configuration (compared to simple random initialization), therefore accelerating the convergence of the training procedure.

Sometimes, the machine learning task remains the same, but the input data changes slightly. In this case, the general features are not the first layers (near the input) but the last layers (near the output). For example, a speech recognition algorithm must always output valid sentences, but its input may originate from persons with different accents (e.g. Chilean Spanish, Argentinian Spanish, Mexican Spanish...). Here, one may freeze the weights of the last layers, and train only the first layers in order to have a task-specific data pre-processing.

5.3.2 Learning a representation for 3D structures

In chapter 4 we have shown how effective supervised deep learning can be applied to analyze cryo-ET data. The disadvantage of using supervised classification is that biologist can only detect macromolecules that have been identified before. Identifying new, unknown objects however requires unsupervised classification, which is why most classification techniques in structural biology (i.e. subtomogram classification) are indeed unsupervised [Förster and Hegerl, 2007] [Yu and Frangakis, 2011] [Chen et al., 2014b]. This section is an attempt to achieve unsupervised classification of cryo-ET data, based on deep learning. More precisely, we propose a new approach to achieve unsupervised subtomogram classification (see Section 1.3.4).

Our idea is to find a representation, i.e. a feature vector, that characterizes well 3D shapes in the presence of noise and artifacts, as found in cryo-ET. This representation should also be rotation invariant, in order to avoid subtomogram alignment as in [Yu and Frangakis, 2011] and [Chen et al., 2014b]. Once

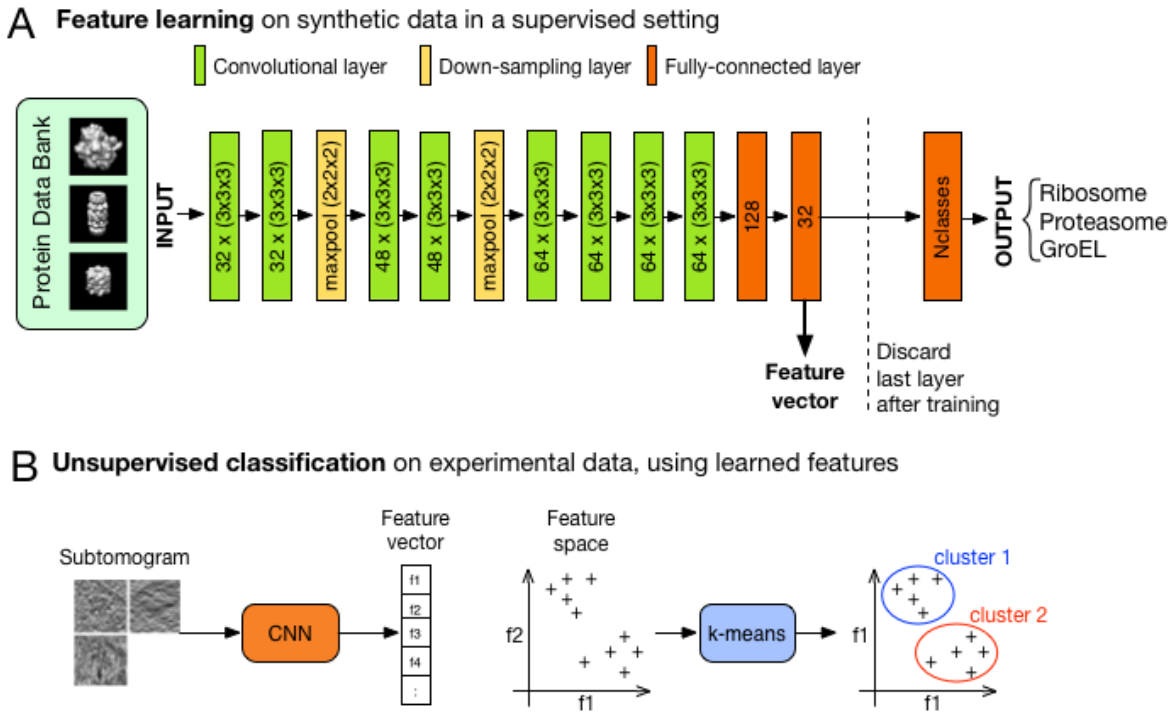

FIGURE 5.11 – Unsupervised subtomogram classification. (A) Architecture of the CNN used to learn a feature space that characterizes 3D shapes. The network is trained on a synthetic data-set composed of 10 different macromolecule classes, obtained from the PDB databank. Once trained, we discard the last layer. We now have a network that takes as an input volumes and outputs feature vectors of size 32. (B) We use this network to compute the feature vectors of experimental subtomograms. We finally achieve unsupervised classification of these subtomograms by applying k-means clustering on their feature vectors. The obtained clusters group the subtomograms by structural similarity.

such a representation exists, one can apply classic clustering algorithms (e.g. hierarchical clustering, mean-shift, k-means...) on these feature vectors to group unknown objects according to their structural similarity (see Figure 5.11 B). Finally, applying subtomogram averaging to objects belonging to the same cluster allows to reveal the high-resolution structure of unknown objects.

Instead of using hand-crafted features as in [Chen et al., 2012] and [Chen et al., 2014a], our objective is to use learned features. More specifically, we would like to determine if we can use features learned on another task, as explained in previous section (i.e. transfer learning), in order to avoid annotations. Unfortunately we can not benefit from networks trained on huge data sets like ImageNet [Deng et al., 2009], as advised in the literature [Tajbakhsh et al., 2016], since exclusively designed for 2D images. One may apply 2D networks on a 3D tomogram in a slice-by-slice manner, however here we prefer to consider 3D operations. As 3D equivalents of ImageNet do not exist yet, we will use a home-made data-set, namely the synthetic data-set used in Section 4.3.1.

5.3.3 Experimental results

Feature learning The synthetic data-set used for learning the data representation is composed of 10 macromolecule classes (see Section 4.3.1). It has been obtained by applying a tomogram simulator on density maps retrieved from the PDB data-bank. Unlike Section 4.3.1, we train a network in the classification setting (i.e. 1 label per image) instead of the segmentation setting (i.e. 1 label per voxel). Therefore we use a simplified version of our segmentation architecture, defined in Figure 5.11 A. The training set has been obtained by sampling 5616 subtomograms per class from the synthetic data-set (mixing different SNR levels). Once trained until convergence, the network achieves 100% of accuracy on a test set composed of 350 synthetic subtomograms.

Finally, the desired data representation is obtained by pruning the output layer as illustrated in Figure 5.11 A, resulting in a vector of size 32.

Classifying real data We evaluated the learned representation on an experimental subtomogram test set sampled from the Chlamydomonas dataset described in Section 4.3.1. The test set contains 100 subtomograms per class and 4 classes : background (null class), membrane-bound ribosome (*mb-ribo*), cytoplasmic ribosome (*ct-ribo*) and proteasome. The subtomograms are of size $40 \times 40 \times 40\times$ voxel and the voxel size is 13.68Å.

For a visual assesment of the learned data representation, we embeded the feature space in $\mathbb{R}^2$ using the t-SNE algorithm [van der Maaten and Hinton, 2008]. T-SNE is a non-linear dimensionality reduction technique, whose objective is to find a faithful low-dimensional representation of a high-dimensional space. As can be observed in Figure 5.12, the t-SNE visualization reveals that the data points are organized in two main blobs : one corresponding to the background and proteasome classes, and one corresponding to the *mb-* and *ct-ribo* classes. Accordingly, this feature space allows to discriminate ribosomes from other classes (i.e. background and proteasome). However, this representation is not precise enough to tell apart background from proteasomes. Proteasomes are indeed harder to find than ribosomes, because they are smaller and less dense (i.e. less contrast). Also, this representation has difficulties differenciating *mb-ribos* from *ct-ribos*. This can be explained by the fact that we trained the network to characterize 3D structures, without taking into account neighboring objects. But the context of the ribosome is precisely what differenciate *mb-ribos* from *ct-ribos*.

The conclusions drawn from the t-SNE visualization are confirmed by the clustering results. In Figure 5.11 B, we display the composition of the clusters obtained when applying the k-means algorithm. Together, the clusters 1, 3 and 4 contain 100% and 98% of the background and proteasome classes, respectively. As to cluster 2, it contains 78% and 95% of the *mb-ribo* and *ct-ribo* classes, respectively. It is also worth mentioning that cluster 1 has a tendency of grouping background samples (46%) and cluster 3 has a tendency of grouping proteasomes (51%).

These are in my opinion encouraging results. Without any annotation, this unsupervised classification approach could correctly categorize 86.5% of ribosomes, regardless of their sub-classes (i.e. membrane-bound and cytoplasmic). Also, this result has been achieved with under-sampled data (factor 4), i.e. a pixel size of 13.68Å, while most concurrent unsupervised subtomogram classification techniques use full resolution. In fact, the results presented in [Förster and Hegerl, 2007] and [Yu and Frangakis, 2011] have

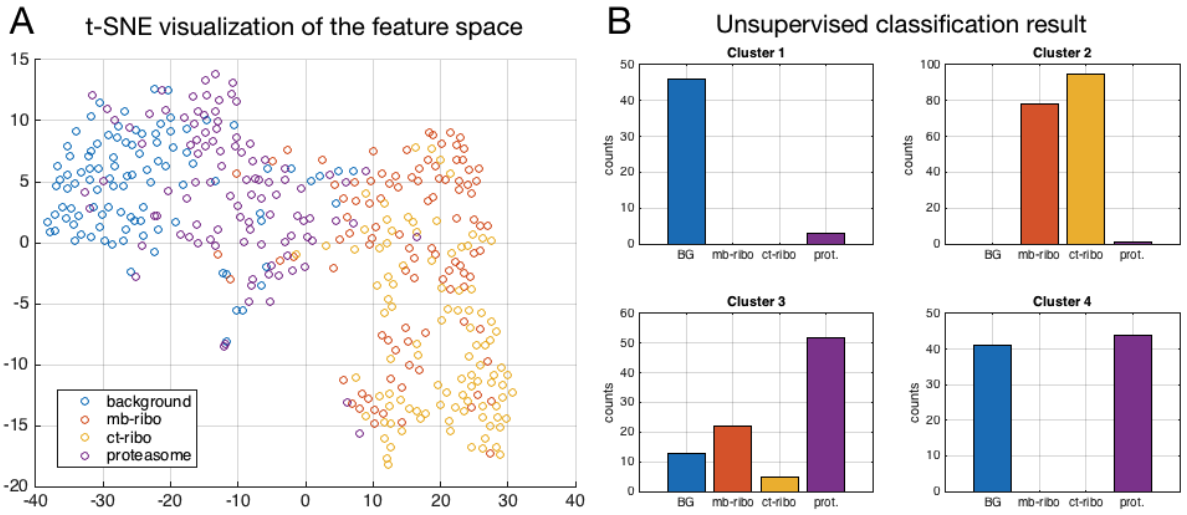

FIGURE 5.12 – Results on experimental data. (A) 2D embeding of the 32-dimensional feature space. The embeding is obtained with the t-SNE algorithm [van der Maaten and Hinton, 2008]. The dots correspond to experimental subtomograms, which contain 4 different classes of macromolecules : the null class (i.e. the background), membrane-bound ribosomes (mb-ribo), cytoplasmic ribosome (ct-ribo) and proteasome. (B) Composition of the clusters after applying the k-means algorithm on the 32-dimensional feature vectors.

been obtained with a pixel size of 6Å, and in [Chen et al., 2014a] with a pixel size of 2.62Å.

In summary, the strong points of this approach are the following ones :

— The learned feature vectors are rotation invariant.

— The method is very fast : it took only 8 seconds to classify 400 sub-tomograms of size $40 \times 40 \times 40$ on a Tesla K80 GPU.

— It allows for real unsupervised classification, and has therefore the potential to be used for discovering unknown macromocule types.

There is room for improvement :

— One could use full-resolution data in order to get a more precise structure characterization

— One could generate a much larger synthetic data-set than the one used here (10 classes), thus creating a 3D equivalent of the ImageNet dataset. Training a network to categorize thousands of macromolecule classes (as found on the PDB databank), would surely result in much more robust feature vectors. Such feature vectors may also be useful for tasks such as unsupervised segmentation.

5.4 Conclusion

In this Chapter, we have explored three strategies for reducing the amount of supervision needed to train deep neural networks in cryo-ET. These strategies are based on recent techniques that have drawn the deep learning community's attention in recent years.

In Section 5.1, we showed that data generated by a GAN may be used as a training set for identifying ribosomes in real tomograms. It was interesting to observe that the GAN could recreate cellular context, such as membranes and neighboring macromolecules. This example illustrates well the representational power of deep networks. However, we are not sure how useful such a technique would be for the user. Indeed, this framework still needs annotated data (for training the GAN). Given that the user could simply annotate more data, it remains to be determined if the efforts of training a GAN are worth it. Also, GANs may generate hallucinations, i.e. image contents that does not have biological reality. Such artifacts would likely be problematic if used as a training set.

In Section 5.2, we present a second approach for generating synthetic cryo-ET data, based on artistic *style transfer*. The difference with the precedent method is that we do not need any annotation. The strategy consists in capturing experimental noise in cryo-ET images, and projecting the noise onto images whose contents we control. From the user perspective, such an approach is more practical. First, the user could select a number of target macromolecules from the PDB databank to generate the content images. Then, he could use *style transfer* to learn a noise model adapted to his imaging parameters, and transfer the noise to the content images, therefore obtaining a plausible training set. However, such an approach has limitations. First, it depends on a pre-trained network. As pre-trained 3D networks are not common, this approach is limited to 2D up to now. Second, in order to generate training data for a given macromolecule, its density map must be available in the PDB databank. Therefore this approach can not be used to discover unknown macromolecules.

In Section 5.3, we present a novel method for achieving unsupervised subtomogram classification, based on transfer learning. As for the precedent approach, we use the PDB databank to generate a synthetic dataset. However, this time we focus on learning a representation instead of learning to identify a given macromolecule. Having a general representation for characterizing 3D shapes allows to assess the structural similarity of unknown objects. Therefore, this approach may be used to discover novel macromolecules. In our results, we obtained good accuracy for identifying ribosomes (86.5%), however the technique is currently not precise enough for proteasomes. From all three proposed strategies, I believe that this last one has the most potential. First, the performance of this approach may be easily improved. By using a more diverse synthetic training set, a more robust representation can be learned. Indeed, here we used 10 macromolecule classes, whereas the PDB databank contains thousands of them. Second, this method is the most practical from the user points of view, because it needs few manual intervention. 3D networks could be pre-trained on huge synthetic datasets, and then made available to the community. The user would simply need to download these networks and apply them as explained to analyze their data in a unsupervised way.

THESIS CONCLUSION

Contributions and discussion

In this thesis, we proposed several contributions for image restoration and macromolecule localization in cryo-ET imaging.

Image processing : Denoising and missing wedge restoration First, we proposed a stochastic method to jointly denoise and recover missing Fourier coefficients (i.e. the missing wedge) in cryo-ET images. By recovering these unobserved coefficients, missing wedge artifacts are reduced and image interpretability is improved. We embeded this method in a Monte Carlo framework, by reformulating the problem in connection with a dedicated Metropolis Hastings algorithm, therefore providing proof of convergence. Also, we compared the method to other approaches aiming to reduce missing wedge artifacts. Finally, we provided evidence that the method improves signal in experimental cryo-ET images.

However, care must be taken to ensure that the processed images are used correctly. Indeed, the restored Fourier coefficients do not represent a physical measurement, but an extrapolation based on observed data and imposed constraints. In other words, the method makes the best guess based on what is more likely. Such a processing is useful to facilitate visual inspection of an image, or as a pre-processing to computational methods that would otherwise be biased by artifacts and noise. But when it comes to measuring features of the imaged specimen (e.g. structure estimation via subtomogram averaging), the measurement should always be made on observed data only.

Image analysis : Deep learning for macromolecule localization In our second contribution, we presented an original deep learning approach to localize macromolecules in cryo-ET images. We showed that our method outperforms template matching, the current state-of-the-art in cryo-ET, for multiple imaging conditions (varying SNR and tilt-range). Also, we validated our method on experimental data and demonstrated a significant overlap with expert annotations (86%). Moreover, our method could localize macromolecules that were missed during the annotation process (+20%). The method could therefore be used as a complement to expert analysis, to search tomograms for unnoticed targets. Given the size of considered data (multiple tomograms of size $4000 \times 4000 \times 2000$ voxel) and the extremely low SNR (0.01 to 0.1), we believe that deep learning is a very well adapted tool for this field. In addition, with modern GPUs, processing time is very short compared to current methods (15min versus +30hours for 1 tomogram). There are however two open questions to be discussed :

— First, we use the segmentation task as a proxy for localization (i.e. segmenting the objects allows to localize them). However this strategy may be an overkill, because inferring the exact shape of the object is not necessary, e.g. segmenting only half of the object (i.e. with only 0.5 *recall*) is sufficient to localize it. In addition, even though we presented a procedure for generating shape

annotations, *label noise* will ultimately be introduced because of the structural variability of macromolecules. Therefore, it could be beneficial to bypass the segmentation task and to directly train a network for localization.

— Second, our method is based on supervised learning and therefore requires manual annotation, which is expensive to generate. This also means that biologists can only scan images for objects they previously identified for constituting the training set. Therefore they can not discover entirely new macromolecule classes, as they should be able to do. That being said, demonstrating the feasibility of our method in a convincing way is the first step to induce curiosity, and therefore demand, in the cryo-ET field. From there on, more investigation for unsupervised deep learning methods for cryo-ET should follow naturally.

Perspectives

In the last chapter of this thesis, I proposed new insights to reduce the amount of supervision needed for training a deep neural network for cryo-ET data. I provided proof-of-concept for three possible strategies, involving generative adversarial networks (GANs), artistic *style transfer* and *transfer learning*, respectively.

From my preliminary results with *style transfer* and *transfer learning*, I was able to highlight the importance of creating large-scale 3D datasets, similar to ImageNet. The main benefit would be to learn robust high-level representations for 3D structures, and use such representations to achieve unsupervised classification of cryo-ET data. Moreover, such representations would also be useful in other imaging domains. Currently, the medical imaging community is investigating 3D deep learning for analyzing data such as CT and MRI images. An important resource these communities lack is a benchmark to compare and optimize general purpose analysis tools for 3D images. As important amounts of annotated 3D images are rarer than 2D images, creating such a dataset represents a challenge. However, I believe that a well adapted resource for such a task is the Protein Data Bank, which contains thousands of resolved 3D structures. These structures could be used to create a diverse dataset, with various amounts and types of noise. Also, the macromolecules can be organized in classes (e.g. proteasome) as well as sub-classes (e.g. single-caped proteasome, double-caped proteasome...). Such kind of annotations is very beneficial for achieving a hierarchical data representation, encoding high-level semantic knowledge.

BIBLIOGRAPHIE

[Albert et al., 2017] Albert, S., Schaffer, M., Beck, F., Mosalaganti, S., Asano, S., Thomas, H. F., Plitzko, J. M., Beck, M., Baumeister, W., and Engel, B. D. (2017). Proteasomes tether to two distinct sites at the nuclear pore complex. In *Proc. Natl. Acad. Sci.*, volume 114, page 201716305.

[Baldi, 2012] Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. In *Work. Unsupervised Transf. Learn.*, pages 37–50.

[Bartesaghi et al., 2008] Bartesaghi, A., Sprechmann, P., Liu, J., Randall, G., Sapiro, G., and Subramaniam, S. (2008). Classification and 3D averaging with missing wedge correction in biological electron tomography. *J. Struct. Biol.*, 162(3) :436–50.

[Bäuerlein et al., 2017] Bäuerlein, F. J., Saha, I., Mishra, A., Kalemanov, M., Martínez-Sánchez, A., Klein, R., Dudanova, I., Hipp, M. S., Hartl, F. U., Baumeister, W., and Fernández-Busnadiego, R. (2017). In Situ Architecture and Cellular Interactions of PolyQ Inclusions. *Cell*, 171(1) :179–187.

[Beck et al., 2009] Beck, M., Malmström, J. A., Lange, V., Schmidt, A., Deutsch, E. W., and Aebersold, R. (2009). Visual proteomics of the human pathogen Leptospira interrogans. *Nat. Methods*, 6(11) :817–823.

[Best et al., 2007] Best, C., Nickell, S., and Baumeister, W. (2007). Localization of protein complexes by pattern recognition. *Methods Cell Biol.*, 2007(79) :615–638.

[Beucher and Meyer, 1993] Beucher, S. and Meyer, F. (1993). The morphological approach to segmentation : the watershed transformation. *Math. Morphol. Image Process.*, pages 433–481.

[Bierkens, 2015] Bierkens, J. (2015). Non-reversible Metropolis-Hastings. *Stat. Comput.*, 26(6) :1213–1228.

[Breyer and Roberts, 2000] Breyer, L. A. and Roberts, G. O. (2000). From metropolis to diffusions : Gibbs states and optimal scaling. *Stoch. Process. their Appl.*, 90(2) :181–206.

[Briggs, 2013] Briggs, J. A. G. (2013). Structural biology in situ-the potential of subtomogram averaging. *Curr. Opin. Struct. Biol.*, 23(2) :261–267.

[Brooks and Gelman, 1998] Brooks, S. P. and Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *J. Comput. Graph. Stat.*, 7(4) :434–455.

[Buades et al., 2005] Buades, A., Coll, B., and Morel, J.-m. (2005). A non-local algorithm for image denoising. *Comput. Vis. Pattern Recognit.*, 2 :60–65.

[Buchholz et al., 2018] Buchholz, T.-O., Jordan, M., Pigino, G., and Jug, F. (2018). Cryo-Care : content-aware image restoration for cryo-transmission electron microscopy data. *arXiv Prepr.*, pages 1–5.

[Burger et al., 2012] Burger, H. C., Schuler, C. J., and Harmeling, S. (2012). Image denoising : can plain neural networks compete with BM3D ?

[Chambolle and Jalalzai, 2014] Chambolle, A. and Jalalzai, K. (2014). Adapted basis for nonlocal reconstruction of missing spectrum. *SIAM J. Imaging Sci.*, 7(3) :1484–1502.

[Charbonnier et al., 1997] Charbonnier, P., Blanc-Féraud, L., Aubert, G., and Barlaud, M. (1997). Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. Image Process.*, 6(2) :298–311.

[Chatterjee and Milanfar, 2012] Chatterjee, P. and Milanfar, P. (2012). Patch-based near-optimal image denoising. *IEEE Trans. Image Process.*, 21(4) :1635–1649.

[Che et al., 2018] Che, C., Lin, R., Zeng, X., Elmaaroufi, K., Galeotti, J., and Xu, M. (2018). Improved deep learning based macromolecules structure classification from electron cryo tomograms. *Mach. Vis. Appl.*, pages 1227–1236.

[Chen et al., 2017] Chen, M., Dai, W., Sun, S. Y., Jonasch, D., He, C. Y., Schmid, M. F., Chiu, W., and Ludtke, S. J. (2017). Convolutional neural networks for automated annotation of cellular cryo-electron tomograms. *Nat. Methods*.

[Chen et al., 2014a] Chen, X., Chen, Y., Schuller, J. M., Navab, N., and Förster, F. (2014a). Automatic particle picking and multi-class classification in cryo-electron tomograms. In *Int. Symp. Biomed. Imaging*, pages 838–841.

[Chen et al., 2012] Chen, Y., Hrabe, T., Pfeffer, S., Pauly, O., Mateus, D., Navab, N., and Förster, F. (2012). Detection and identification of macromolecular complexes in cryo-electron tomograms using support vector machines. In *IEEE Int. Symp. Biomed. Imaging*, volume 1, pages 1373–1376.

[Chen et al., 2014b] Chen, Y., Pfeffer, S., Fernández, J. J., Sorzano, C. O. S., and Förster, F. (2014b). Autofocused 3D Classification of Cryoelectron Subtomograms. *Structure*, pages 1528–1537.

[Chen et al., 2013] Chen, Y., Pfeffer, S., Hrabe, T., Schuller, J. M., and Förster, F. (2013). Fast and accurate reference-free alignment of subtomograms. *J. Struct. Biol.*, 182(3) :235–45.

[Chollet, 2017] Chollet, F. (2017). *Deep learning with Python*. 1st edition.

[Chung and Kim, 2017] Chung, J.-H. and Kim, H. M. (2017). The Nobel prize in chemistry 2017 : high-resolution cryo-electron microscopy. *Appl. Microsc.*, 47(4) :218–222.

[Comaniciu et al., 2002] Comaniciu, D., Meer, P., and Member, S. (2002). Mean Shift : a robust approach toward feature space analysis. In *IEEE Trans. Pattern Anal. Mach. Intell.*, volume 24, pages 603–619.

[Creswell et al., 2017] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. (2017). Generative Adversarial Networks : An Overview. *arXiv Prepr.*, pages 1–14.

[Cybenko, 1989] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control. Signals Syst.*, 2(4) :303–314.

[Dabov et al., 2007] Dabov, K., Foi, A., and Egiazarian, K. (2007). Image denoising by sparse 3D transform-domain collaborative filtering. *IEEE Trans. Image Process.*, 16(8) :145–149.

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, volume 1, pages 886–893.

[Darbon et al., 2008] Darbon, J., Cunha, A., and Chan, T. F. (2008). Fast nonlocal filtering applied to electron cryomicroscopy. In *IEEE Int. Symp. Biomed. Imaging From Nano to Macro*, pages 1331–1334.

[Deledalle et al., 2009] Deledalle, C. A., Denis, L., and Tupin, F. (2009). Iterative weighted maximum likelihood denoising with probabilistic patch-based weights. *IEEE Trans. Image Process.*, 18(12) :2661–2672.

[Deledalle et al., 2012] Deledalle, C. A., Duval, V., and Salmon, J. (2012). Non-local methods with shape-adaptive patches (NLM-SAP). *J. Math. Imaging Vis.*, 43(2) :103–120.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet : a large-scale hierarchical image database. In *Conf. Comput. Vis. Pattern Recognit.*, pages 248–255. IEEE.

[Duda et al., 2000] Duda, R. O., Hart, P. E., and Stork, D. G. (2000). Backpropagation Algorithm (Sec. 6.3). In *Pattern Classif.*, pages 288–295. Wiley-Interscience New York, NY, USA, 2nd edition.

[Duval et al., 2011] Duval, V., Aujol, J.-F., and Gousseau, Y. (2011). A bias-variance approach for the nonlocal means. *SIAM J. Imaging Sci.*, 4(2) :760–788.

[Elad and Aharon, 2006] Elad, M. and Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. In *IEEE Trans. Image Process.*, volume 15, pages 3736–3745.

[Fernandez, 2012] Fernandez, J.-J. (2012). Computational methods for electron tomography. *Micron*, 43(10) :1010–1030.

[Fernandez and Li, 2003] Fernandez, J. J. and Li, S. (2003). An improved algorithm for anisotropic nonlinear diffusion for denoising cryo-tomograms. *J. Struct. Biol.*, 144 :152–161.

[Fienup, 1982] Fienup, J. R. (1982). Phase retrieval algorithms : a comparison. *Appl. Opt.*, 21.

[Förster et al., 2010] Förster, F., Han, B. G., and Beck, M. (2010). Visual Proteomics. *Methods Enzymol.*, 483(C) :215–243.

[Förster and Hegerl, 2007] Förster, F. and Hegerl, R. (2007). Structure determination In Situ by averaging of tomograms. In *Cell. Electron Microsc.*, volume 79, pages 741–767.

[Förster et al., 2008] Förster, F., Pruggnaller, S., Seybert, A., and Frangakis, A. S. (2008). Classification of cryo-electron sub-tomograms using constrained correlation. *J. Struct. Biol.*, 161(3) :276–286.

[Frangakis and Hegerl, 2001] Frangakis, a. S. and Hegerl, R. (2001). Noise reduction in electron tomographic reconstructions using nonlinear anisotropic diffusion. *J. Struct. Biol.*, 135 :239–250.

[Frikel and Quinto, 2013] Frikel, J. and Quinto, E. T. (2013). Characterization and reduction of artifacts in limited angle tomography. *Inverse Probl.*, 29(12).

[Gatys et al., 2015] Gatys, L. A., Ecker, A. S., and Bethge, M. (2015). A neural algorithm of artistic style. *arXiv Prepr.*, pages 1–16.

[Gelman and Rubbin, 1992] Gelman, A. and Rubbin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Stat. Sci.*, 7(4) :457–511.

[Geman and Reynolds, 1992] Geman, D. and Reynolds, G. (1992). Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3) :367–383.

[Gilavert et al., 2015] Gilavert, C., Moussaoui, S., and Idier, J. (2015). Efficient Gaussian sampling for solving large-scale inverse problems using MCMC. *IEEE Trans. Signal Process.*, 63 :70–80.

[Gilbert, 1972] Gilbert, P. (1972). Iterative methods for the three-dimensional reconstruction of an object from projections. *J. Theor. Biol.*, 36(1) :105–117.

[Gilks et al., 1995] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1995). *Markov Chain Monte Carlo in Practice : Interdisciplinary Statistics*.

[Girolami et al., 2011] Girolami, M., Calderhead, B., and Chin, S. A. (2011). Riemann manifold Langevin and Hamiltonian Monte Carlo. *J. R. Stat. Soc. Ser. B*, 73(2) :123–214.

[Goodfellow, 2016] Goodfellow, I. (2016). NIPS 2016 tutorial : Generative adversarial networks. *arXiv*.

[Goodfellow et al., 2016a] Goodfellow, I., Bengio, Y., and Courville, A. (2016a). Basic Algorithms (Sec. 8.3). In *Deep Learn.*, pages 294–300. MIT Press.

[Goodfellow et al., 2016b] Goodfellow, I., Bengio, Y., and Courville, A. (2016b). Convolutional Networks (Sec. 9.2). In *Deep Learn.*, page 335.

[Goodfellow et al., 2014] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Nets. In *Adv. Neural Inf. Process. Syst.*, pages 1–9.

[Goodfellow et al., 2015] Goodfellow, I. J., Vinyals, O., and Saxe, A. M. (2015). Qualitatively characterizing neural network optimization problems. In *Int. Conf. Learn. Represent.*, pages 1–20.

[Grezl et al., 2007] Grezl, F., Karafiat, M., Kontar, S., and Cernocky, J. (2007). Probabilistic and bottleneck features for LVCSR of meetings. In *IEEE Int. Conf. Acoust. Speech Signal Process.*, pages 757–760.

[Gribonval, 2011] Gribonval, R. (2011). Should penalized least squares regression be interpreted as maximum a posteriori estimation ? In *IEEE Trans. Signal Process.*, volume 59, pages 2405–2410.

[Griewank, 2012] Griewank, A. (2012). Who invented the reverse mode of differentiation ? *Doc. Math. · Extra Vol. ISMP*, I :389–400.

[Guesdon et al., 2013] Guesdon, A., Blestel, S., Kervrann, C., and Chrétien, D. (2013). Single versus dual-axis cryo-electron tomography of microtubules assembled in vitro : limits and perspectives. *J. Struct. Biol.*, 181(2) :169–78.

[Guichard and Malgouyres, 1998] Guichard, F. and Malgouyres, F. (1998). Total variation based interpolation. In *Eur. Signal Process. Conf.*, volume 3, pages 1741–1744.

[Guo et al., 2018] Guo, Q., Lehmer, C., Martinez-Sanchez, A., Rudack, T., Beck, F., Hartmann, H., Perez-Berlanga, M., Frottin, F., Hipp, M. S., Hartl, F. U., Edbauer, D., Baumeister, W., and Fernández-Busnadiego, R. (2018). In situ structure of neuronal C9ORF72 poly-GA aggregates reveals proteasome recruitment. *Cell*, 172(4) :696–705.

[Hastings, 1970] Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57 :97–109.

[He and Garcia, 2009] He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.*, 21(9) :1263–1284.

[Henderson, 2013] Henderson, R. (2013). Avoiding the pitfalls of single particle cryo-electron microscopy : Einstein from noise. In *Proc. Natl. Acad. Sci.*, volume 110, pages 18037–18041.

[Heymann and Belnap, 2007] Heymann, J. B. and Belnap, D. M. (2007). Bsoft : Image processing and molecular modeling for electron microscopy. *J. Struct. Biol.*, 157(1) :3–18.

[Hinton et al., 2012] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Process. Mag.*, 29(6) :82–97.

[Holden et al., 2009] Holden, L., Hauge, R., and Holden, M. (2009). Adaptive independent Metropolis-Hastings. *Ann. Appl. Probab.*, 19(1) :395–413.

[Hrabe et al., 2012] Hrabe, T., Chen, Y., Pfeffer, S., Kuhn Cuellar, L., Mangold, A.-V., and Förster, F. (2012). PyTom : A python-based toolbox for localization of macromolecules in cryo-electron tomograms and subtomogram analysis. *J. Struct. Biol.*, 178(2) :177–188.

[Isola et al., 2017] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-Image Translation with Conditional Adversarial Networks. *arXiv Prepr.*

[Jin et al., 2018] Jin, D., Xu, Z., Tang, Y., Harrison, A. P., and Mollura, D. J. (2018). CT-realistic lung nodule simulation from 3D conditional generative adversarial networks for robust lung segmentation. In *Med. Image Comput. Comput. Interv.*, pages 732–740.

[Jin et al., 2017] Jin, Q., Grama, I., Kervrann, C., and Liu, Q. (2017). Nonlocal means and optimal weights for noise removal. *SIAM J. Imaging Sci.*, 10(4) :1878–1920.

[Julesz, 1962] Julesz, B. (1962). Visual pattern discrimination. *IRE Trans. Inf. Theory*, 8(2) :84–92.

[Katkovnik et al., 2010] Katkovnik, V., Foi, A., Egiazarian, K., and Astola, J. (2010). From local kernel to nonlocal multiple-model image denoising. *Int. J. Comput. Vis.*, 86(1) :1–32.

[Kawaguchi, 2016] Kawaguchi, K. (2016). Deep learning without poor local minima. In *Conf. Neural Inf. Process. Syst.*, pages 1–9.

[Kazerouni et al., 2013] Kazerouni, A., Kamilov, U. S., Bostan, E., and Unser, M. (2013). Bayesian denoising : from MAP to MMSE using consistent cycle spinning. In *IEEE Signal Process. Lett.*, volume 20, pages 249–252.

[Kervrann, 2014] Kervrann, C. (2014). PEWA : Patch-based exponentially weighted aggregation for image denoising. In *Adv. Neural Inf. Process. Syst.*, volume 27, pages 1–9.

[Kervrann et al., 2014a] Kervrann, C., Blestel, S., and Chrétien, D. (2014a). Conditional random fields for tubulin-microtubule segmentation in cryo-electron tomography. In *IEEE Trans. Image Process.*, pages 2080–2084.

[Kervrann and Boulanger, 2006] Kervrann, C. and Boulanger, J. (2006). Optimal spatial adaptation for patch-based image denoising. In *IEEE Trans. Image Process.*, volume 15, pages 2866–2878.

[Kervrann and Boulanger, 2007] Kervrann, C. and Boulanger, J. (2007). Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal. In *Scale Sp. Var. Methods Comput. Vis.*, pages 520–532.

[Kervrann and Boulanger, 2008] Kervrann, C. and Boulanger, J. (2008). Local adaptivity to variable smoothness for exemplar-based image regularization and representation. *Int. J. Comput. Vis.*, 79(1) :45–69.

[Kervrann et al., 2014b] Kervrann, C., Roudot, P., and Waharte, F. (2014b). Approximate bayesian computation, stochastic algorithms and non-local means for complex noise models. In *IEEE Int. Conf. Image Process.*, pages 2834–2838.

[Keskar et al., 2016] Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. (2016). On Large-Batch Training for Deep Learning : Generalization Gap and Sharp Minima. *arXiv Prepr.*, pages 1–16.

[Kindermann et al., 2005] Kindermann, S., Osher, S., and Jones, P. W. (2005). Deblurring and denoising of images by nonlocal functionals. *Multiscale Model. Simul.*, 4(4) :1091–1115.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. L. (2014). Adam : a method for stochastic optimization. *arXiv Prepr.*

[Kováčik et al., 2014] Kováčik, L., Kereïche, S., Höög, J. L., Jůda, P., Matula, P., and Raška, I. (2014). A simple Fourier filter for suppression of the missing wedge ray artefacts in single-axis electron tomographic reconstructions. *J. Struct. Biol.*, 186(1) :141–52.

[Kreme et al., 2018] Kreme, A. M., Emiya, V., and Chaux, C. (2018). Phase reconstruction for time-frequency inpainting. In *Int. Conf. Latent Var. Anal. Signal Sep.*, pages 417–426.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Conf. Neural Inf. Process. Syst.*, pages 1–9.

[Kucukelbir et al., 2014] Kucukelbir, A., Sigworth, F. J., and Tagare, H. D. (2014). Quantifying the local resolution of cryo-EM density maps. *Nat. Methods*, 11(1) :63–65.

[Lauze et al., 2017] Lauze, F., Quéau, Y., and Plenge, E. (2017). Simultaneous reconstruction and segmentation of CT scans with shadowed data. In *Int. Conf. Scale Sp. Var. Methods Comput. Vis.*, pages 308–319.

[Leary et al., 2013] Leary, R., Saghi, Z., Midgley, P. A., and Holland, D. J. (2013). Compressed sensing electron tomography. *Ultramicroscopy*, 131 :70–91.

[Lebrun et al., 2013] Lebrun, M., Buades, A., and Morel, J. (2013). A nonlocal Bayesian image denoising algorithm. *SIAM J. Imaging Sci.*, 6(3) :1665–1688.

[Lecun et al., 2015] Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521 :436–444.

[Lecun et al., 1989] Lecun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Comput.*, 1 :541–551.

[Lecun et al., 2010] Lecun, Y., Kavukcuoglu, K., and Farabet, C. (2010). Convolutional networks and applications in vision. In *IEEE Int. Symp. Circuits Syst.*, pages 253–256.

[Lee et al., 2009] Lee, H., Grosse, R., Ranganath, R., and Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Int. Conf. Mach. Learn.*, pages 1–8.

[Li et al., 2017] Li, X., Zhong, A., Lin, M., Guo, N., Sun, M., Sitek, A., Ye, J., Thrall, J., and Li, Q. (2017). Tversky loss function for image segmentation using 3D fully convolutional deep networks. In *Int. Work. Mach. Learn. Med. Imaging*, volume 2, pages 379–387.

[Liang et al., 2010] Liang, F., Liu, C., and Carroll, R. J. (2010). *Advanced Markov Chain Monte Carlo Methods : Learning from Past Samples*.

[Lin et al., 2013] Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *arXiv Prepr.*, pages 1–10.

[Long et al., 2014] Long, J., Shelhamer, E., and Darrell, T. (2014). Fully convolutional networks for semantic segmentation. In *Conf. Comput. Vis. Pattern Recognit.*, pages 3431–3440.

[Lou et al., 2010] Lou, Y., Zhang, X., Osher, S., and Bertozzi, A. (2010). Image recovery via nonlocal operators. *J. Sci. Comput.*, 42(2) :185–197.

[Louchet and Moisan, 2008] Louchet, C. and Moisan, L. (2008). Total variation denoising using posterior expectation. *16th Eur. Signal Process. Conf.*, 5 :1–5.

[Louchet and Moisan, 2011] Louchet, C. and Moisan, L. (2011). Total variation as a local filter. *SIAM J. Imaging Sci.*, 4(3) :651–694.

[Louchet and Moisan, 2013] Louchet, C. and Moisan, L. (2013). Posterior expectation of the total variation model : properties and experiments. *SIAM J. Imaging Sci.*, 6(4) :2640–2684.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale invariant keypoints. *Int. J. Comput. Vis.*, 60 :91–11020042.

[Lučić et al., 2013] Lučić, V., Rigort, A., and Baumeister, W. (2013). Cryo-electron tomography : The challenge of doing structural biology in situ. *J. Cell Biol.*, 202(3) :407–419.

[Maggioni et al., 2013] Maggioni, M., Katkovnic, V., Egiazarian, K., and Foi, A. (2013). Nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE Trans. Image Process.*, 22(1) :119–133.

[Mahamid et al., 2016] Mahamid, J., Pfeffer, S., Schaffer, M., Villa, E., Danev, R., Cuellar, L. K., Förster, F., Hyman, A. A., Plitzko, J. M., and Baumeister, W. (2016). Visualizing the molecular sociology at the HeLa cell nuclear periphery. *Science (80-.).*, 351(6276) :969–972.

[Mahapatra et al., 2018] Mahapatra, D., Bozorgtabar, B., Thiran, J.-P., and Reyes, M. (2018). Efficient active learning for image classification and segmentation using a sample selection and conditional generative adversarial network. In *Med. Image Comput. Comput. Interv.*, volume 1, pages 580–588.

[Mairal et al., 2009] Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2009). Non-local sparse models for image restoration. In *IEEE Int. Conf. Comput. Vis.*, pages 2272–2279.

[Mallat, 2016] Mallat, S. (2016). Understanding deep convolutional networks. *Philos. Trans. R. Soc. London A Math. Phys. Eng. Sci.*, 374(2065).

[Marnissi et al., 2018] Marnissi, Y., Chouzenoux, E., Benazza-Benyahia, A., and Pesquet, J.-C. (2018). An auxiliary variable method for Markov chain Monte Carlo algorithms in high dimension. *Entropy*, 20.

[Martinez-Sanchez et al., 2014] Martinez-Sanchez, A., Garcia, I., Asano, S., Lucic, V., and Fernandez, J.-j. (2014). Robust membrane detection based on tensor voting for electron tomography. *J. Struct. Biol.*, 186(1) :49–61.

[Mayer et al., 2018] Mayer, N., Ilg, E., Fischer, P., Hazirbas, C., Cremers, D., Dosovitskiy, A., and Brox, T. (2018). What makes good synthetic training data for learning disparity and optical flow estimation ? *arXiv Prepr.*

[McCulloch and Pitts, 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.*, 5(4) :115–133.

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of State Calculations. *J. Chem. Phys.*, 21(6) :1087–1092.

[Miao et al., 2005] Miao, J., Förster, F., and Levi, O. (2005). Equally sloped tomography with oversampling reconstruction. *Phys. Rev. B*, 72(5) :3–6.

[Milanfar, 2013] Milanfar, P. (2013). A tour of modern image filtering. *IEEE Signal Process. Mag.*, 30(1) :106–128.

[Milletari et al., 2016] Milletari, F., Navab, N., and Ahmadi, S.-a. (2016). V-Net : fully convolutional neural networks for volumetric medical image segmentation. *arXiv Prepr.*, pages 1–11.

[Mirza and Osindero, 2014] Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv Prepr.*, pages 1–7.

[Mitchell, 1997] Mitchell, T. (1997). *Machine Learning*. McGraw-Hill Education.

[Moisan, 2011] Moisan, L. (2011). Periodic plus smooth image decomposition. *J. Math. Imaging Vis.*, 39(2) :161–179.

[Moisan, 2001] Moisan, L. C. (2001). Extrapolation de spectre et variation totale pondérée. In *18e Colloq. sur le Trait. du Signal des Images*, pages 892–895.

[Moreno et al., 2018] Moreno, J. J., Martinez-Sanchez, A., Martinez, J. A., Garzon, E. M., and Fernandez, J. J. (2018). TomoEED : fast edge-enhancing denoising of tomographic volumes. *Bioinformatics*, 34(21) :3776–3778.

[Naylor et al., 2018] Naylor, P., Lae, M., Reyal, F., and Walter, T. (2018). Segmentation of Nuclei in Histopathology Images by deep regression of the distance map. *IEEE Trans. Med. Imaging*, (3) :1–18.

[Neal, 2004] Neal, R. M. (2004). Improving asymptotic variance of MCMC estimators : non-reversible chains are bbetter. Technical Report 0406.

[Ortiz et al., 2006] Ortiz, J. O., Förster, F., Kürner, J., Linaroudis, A. a., and Baumeister, W. (2006). Mapping 70S ribosomes in intact cells by cryoelectron tomography and pattern recognition. *J. Struct. Biol.*, 156(2) :334–41.

[Ouyang et al., 2018] Ouyang, W., Aristov, A., Lelek, M., Hao, X., and Zimmer, C. (2018). Deep learning massively accelerates super-resolution localization microscopy. *Nat. Biotechnol.*, 36 :460–468.

[Paavolainen et al., 2014] Paavolainen, L., Acar, E., Tuna, U., Peltonen, S., Moriya, T., Pan, S., Marjom, V., Cheng, R. H., and Ruotsalainen, U. (2014). Compensation of missing wedge effects with sequential statistical reconstruction in electron tomography. *PLoS One*, 9(10) :1–23.

[Papandreou and Yuille, 2010] Papandreou, G. and Yuille, A. L. (2010). Gaussian sampling by local perturbations. *Adv. Neural Inf. Process. Syst.*, 23 :1–9.

[Patrini et al., 2017] Patrini, G., Rozza, A., Menon, A. K., Nock, R., and Qu, L. (2017). Making deep neural networks robust to label noise : a loss correction approach. *arXiv Prepr.*

[Pei et al., 2016] Pei, L., Xu, M., Frazier, Z., and Alber, F. (2016). Simulating cryo electron tomograms of crowded cell cytoplasm for assessment of automated particle picking. *BMC Bioinformatics*, 17(1) :1–13.

[Pettersen et al., 2004] Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., and Ferrin, T. E. (2004). UCSF Chimera - A visualization system for exploratory research and analysis. *J. Comput. Chem.*, 25(13) :1605–1612.

[Pfeffer et al., 2017] Pfeffer, S., Dudek, J., Schaffer, M., Ng, B. G., Albert, S., Plitzko, J. M., Baumeister, W., Zimmermann, R., Freeze, H. H., Engel, B. D., and Förster, F. (2017). Dissecting the molecular organization of the translocon-associated protein complex. *Nat. Commun.*, 8 :14516.

[Pizarro et al., 2010] Pizarro, L., Mrázek, P., Didas, S., Grewenig, S., and Weickert, J. (2010). Generalised nonlocal image smoothing. *Int. J. Comput. Vis.*, 90(1) :62–87.

[Protter et al., 2010] Protter, M., Yavneh, I., and Elad, M. (2010). Closed-Form MMSE estimation for signal denoising under sparse representation modeling over a unitary dictionary. *IEEE Trans. Signal Process.*, 58(7) :3471–3484.

[Radermacher, 1992] Radermacher, M. (1992). Weighted back-projection methods. In *Frank J. Electron Tomogr.*, pages 245–273.

[Radford et al., 2015] Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv Prepr.*, pages 1–16.

[Radon, 1986] Radon, J. (1986). On the determination of functions from their integral values along certain manifolds. *IEEE Trans. Med. Imaging*, 5(4) :170–176.

[Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *Ann. Math. Stat.*, 22(3) :400–407.

[Robert and Casella, 2004] Robert, C. P. and Casella, G. (2004). *Monte Carlo statistical methods*. Springer.

[Roberts et al., 1997] Roberts, G. O., Gelman, A., and Gilks, W. R. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.*, 7(1) :110–120.

[Roberts and Rosenthal, 2001] Roberts, G. O. and Rosenthal, J. S. (2001). Optimal scaling for various Metropolis-Hastings algorithms. *Stat. Sci.*, 16(4) :351–367.

[Rolnick et al., 2017] Rolnick, D., Veit, A., Belongie, S., and Shavit, N. (2017). Deep learning is robust to massive label noise. *arXiv Prepr.*

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net : Convolutional networks for biomedical image segmentation. In *Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, volume 9351, pages 234–241.

[Rosenblatt, 1957] Rosenblatt, F. (1957). The perceptron - a perceiving and recognizing automaton.

[Ruder, 2016] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv Prepr.*, pages 1–14.

[Rudin et al., 1992] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D*, 60 :259–268.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323 :533–536.

[Scheres, 2012] Scheres, S. H. (2012). RELION : Implementation of a Bayesian approach to cryo-EM structure determination. *J. Struct. Biol.*, 180(3) :519–530.

[Scheres et al., 2009] Scheres, S. H. W., Melero, R., Valle, M., and Carazo, J.-M. (2009). Averaging of electron subtomograms and random conical tilt reconstructions through likelihood optimization. *Structure*, 17(12) :1563–72.

[Sentosun et al., 2017] Sentosun, K., Lobato, I., Bladt, E., Zhang, Y., Palenstijn, W. J., Batenburg, K. J., Dyck, D. V., and Bals, S. (2017). Artifact reduction based on sinogram interpolation for the 3D reconstruction of nanoparticles using electron tomography. *Part. Part. Syst. Charact.*, 1700287 :1–8.

[Shaikh et al., 2008] Shaikh, T. R., Gao, H., Baxter, W. T., Asturias, F. J., Boisset, N., Leith, A., and Frank, J. (2008). SPIDER image processing for single-particle reconstruction of biological macromolecules from electron micrographs. *Nat. Protoc.*, 3(12) :1941–1974.

[Shatsky et al., 2009] Shatsky, M., Hall, R. J., Brenner, S. E., and Glaeser, R. M. (2009). A method for the alignment of heterogeneous macromolecules from electron microscopy. *J. Struct. Biol.*, 166(1) :67–78.

[Simonyan and Zisserman, 2015] Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *Int. Conf. Learn. Represent.*, pages 1–14.

[Springenberg et al., 2015] Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2015). Striving for simplicity : the all convolutional net. In *Int. Conf. Learn. Represent. Work.*, pages 1–14.

[Stölken et al., 2011] Stölken, M., Beck, F., Haller, T., Hegerl, R., Gutsche, I., Carazo, J. M., Baumeister, W., Scheres, S. H. W., and Nickell, S. (2011). Maximum likelihood based classification of electron tomographic data. *J. Struct. Biol.*, 173 :77–85.

[Sudre et al., 2017] Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., and Cardoso, M. J. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Int. Work. Deep Learn. Med. Image Anal.*, volume 10553, pages 240–248.

[Sutour et al., 2014] Sutour, C., Deledalle, C.-a., and Aujol, J.-f. (2014). Adaptive regularization of the NL-means : application to image and video denoising. *IEEE Trans. Image Process.*, 23(8) :3506–3521.

[Szegedy et al., 2013] Szegedy, C., Toshev, A., and Erhan, D. (2013). Deep neural networks for object detection. In *Conf. Neural Inf. Process. Syst.*, pages 1–9.

[Szegedy et al., 2015] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2015). Rethinking the Inception Architecture for Computer Vision. *arXiv Prepr.*

[Tajbakhsh et al., 2016] Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., and Liang, J. (2016). Convolutional neural networks for medical image analysis : full training or fine tuning ? *IEEE Trans. Med. Imaging*, 35(5) :1299–1312.

[Tang et al., 2007] Tang, G., Peng, L., Baldwin, P. R., Mann, D. S., Jiang, W., Rees, I., and Ludtke, S. J. (2007). EMAN2 : An extensible image processing suite for electron microscopy. *J. Struct. Biol.*, 157 :38–46.

[Van De Ville and Kocher, 2009] Van De Ville, D. and Kocher, M. (2009). SURE-based non-local means. *IEEE Signal Process. Lett.*, 16(11) :973–976.

[van der Maaten and Hinton, 2008] van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *J. Mach. Learn. Res.*, 9 :2579–2605.

[Van Heel and Schatz, 2005] Van Heel, M. and Schatz, M. (2005). Fourier shell correlation threshold criteria. *J. Struct. Biol.*, 151 :250–262.

[Vendeville et al., 2011] Vendeville, A., Larivière, D., and Fourmentin, E. (2011). An inventory of the bacterial macromolecular components and their spatial organization. *FEMS Microbiol. Rev.*, 35(2) :395–414.

[Voss et al., 2009] Voss, N. R., Yoshioka, C. K., Radermacher, M., Potter, C. S., and Carragher, B. (2009). DoG Picker and TiltPicker : software tools to facilitate particle selection in single particle electron microscopy. *J. Struct. Biol.*, 166(2) :205–213.

[Walz et al., 1997] Walz, J., Typke, D., Nitsch, M., Koster, A. J., Hegerl, R., and Baumeister, W. (1997). Electron tomography of single ice-embedded macromolecules : Three- dimensional alignment and classification. *J. Struct. Biol.*, 120(3) :387–395.

[Wan and Briggs, 2016] Wan, W. and Briggs, J. A. (2016). *Cryo-Electron Tomography and Subtomogram Averaging*, volume 579. Elsevier Inc., 1 edition.

[Wang et al., 2016] Wang, F., Gong, H., Liu, G., Li, M., Yan, C., Xia, T., Li, X., and Zeng, J. (2016). DeepPicker : A deep learning approach for fully automated particle picking in cryo-EM. *J. Struct. Biol.*, 195(3) :325–336.

[Wang and Morel, 2013] Wang, Y.-Q. and Morel, J.-M. (2013). SURE guided Gaussian mixture image denoising. *SIAM J. Imaging Sci.*, 6(2) :999–1034.

[Wei and Yin, 2010] Wei, D.-Y. and Yin, C.-C. (2010). An optimized locally adaptive non-local means denoising filter for cryo-electron microscopy data. *J. Struct. Biol.*, 172(3) :211–8.

[Werbos, 1975] Werbos, P. J. (1975). *Beyond regression : new tools for prediction and analysis in the behavioral sciences*. Harvard University.

[Wong et al., 2018] Wong, K. C. L., Moradi, M., Tang, H., and Syeda-Mahmood, T. (2018). 3D segmentation with exponential logarithmic loss for highly unbalanced object sizes. In *Med. Image Comput. Comput. Interv.*, pages 612–619.

[Xu et al., 2011] Xu, M., Beck, M., and Alber, F. (2011). Template-free detection of macromolecular complexes in cryo electron tomograms. *Bioinformatics*, 27 :69–76.

[Xu et al., 2012] Xu, M., Beck, M., and Alber, F. (2012). High-throughput subtomogram alignment and classification by Fourier space constrained fast volumetric matching. *J. Struct. Biol.*, 178(2) :152–164.

[Xu et al., 2017] Xu, M., Chai, X., Muthakana, H., Liang, X., Yang, G., Zeev-Ben-Mordehai, T., and Xing, E. P. (2017). Deep learning-based subdivision approach for large scale macromolecules structure recovery from electron cryo tomograms. *Bioinformatics*, 33(14) :i13–i22.

[Xu et al., 2016] Xu, M., Tocheva, E. I., Chang, Y.-w., Jensen, G. J., and Alber, F. (2016). De novo visual proteomics in single cells through pattern mining. *arXiv Prepr.*

[Yan et al., 2019] Yan, R., Venkatakrishnan, S. V., Liu, J., Bouman, C. A., and Jiang, W. (2019). MBIR : A cryo-ET 3D reconstruction method that effectively minimizes missing wedge artifacts and restores missing information. *J. Struct. Biol.*, 206(2) :183–192.

[Yosinski et al., 2014] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks ? In *Adv. Neural Inf. Process. Syst.*, volume 27.

[Yu and Koltun, 2016] Yu, F. and Koltun, V. (2016). Multi-Scale Context Aggregation by Dilated Convolutions. In *Int. Conf. Learn. Represent.*

[Yu and Frangakis, 2011] Yu, Z. and Frangakis, A. S. (2011). Classification of electron sub-tomograms with neural networks and its application to template-matching. *J. Struct. Biol.*, 174(3) :494–504.

[Zeng et al., 2018] Zeng, X., Leung, M. R., Zeev-Ben-Mordehai, T., and Xu, M. (2018). A convolutional autoencoder approach for mining features in cellular electron cryo-tomograms and weakly supervised coarse segmentation. *J. Struct. Biol.*, 202(2) :150–160.

[Zhou et al., 2018] Zhou, B., Guo, Q., Zeng, X., and Xu, M. (2018). Feature decomposition based saliency detection in electron cryo-tomograms. *arXiv Prepr.*

[Zoran and Weiss, 2011] Zoran, D. and Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In *IEEE Int. Conf. Comput. Vis.*, pages 479–486.

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

UNIVERSITÉ DE RENNES 1

Titre : Nouvelles stratégies pour l'identification et l'énumération de macromolécules dans des images de cryo-tomographie électronique 3D

Mot clés : traitement d'images 3D, problèmes inverses, débruitage, apprentissage automatique, microscopie

Resumé : La cryo-tomographie électronique (cryo-ET) est une technique d'imagerie capable de produire des vues 3D de spécimens biologiques. Cette technologie permet d'imager de larges portions de cellules vitrifiées à une résolution nanométrique. Elle permet de combiner plusieurs échelles de compréhension de la machinerie cellulaire, allant des interactions entre les groupes de protéines à leur structure atomique. La cryo-ET a donc le potentiel d'agir comme un lien entre l'imagerie cellulaire in vivo et les techniques atteignant la résolution atomique.

Cependant, ces images sont corrompues par un niveau de bruit élevé et d'artefacts d'imagerie. Leur interprétabilité dépend fortement des méthodes de traitement d'image. Les méthodes computationelles existantes permettent actuellement d'identifier de larges macromolécules telles que les ribosomes, mais il est avéré que ces détections sont incomplètes. De plus, ces méthodes sont limitées lorsque les objets recherchés sont de très petite taille ou présentent une plus grande variabilité structurelle.

L'objectif de cette thèse est de proposer de nouvelles méthodes d'analyse d'images, afin de permettre une identification plus robuste des macromolécules d'intérêt. Nous proposons deux méthodes computationelles pour atteindre cet objectif. La première vise à réduire le bruit et les artefacts d'imagerie, et fonctionne en ajoutant et en supprimant de façon itérative un bruit artificiel à l'image. Nous fournissons des preuves mathématiques et expérimentales de ce concept qui permet d'améliorer le signal dans les images de cryo-ET. La deuxième méthode s'appuie sur les progrès récents de l'apprentissage automatique et les méthodes convolutionelles pour améliorer la localisation des macromolécules. La méthode est basée sur un réseau neuronal convolutif. Nous montrons comment l'adapter pour obtenir des taux de détection supérieur à l'état de l'art.

Title : New strategies for the identification and enumeration of macromolecules in 3D images of cryo electron tomography

Keywords : 3D image processing, inverse problems, denoising, machine learning, microscopy

Abstract : Cryo electron tomography (cryo-ET) is an imaging technique capable of producing 3D views of biological specimens. This technology enables to capture large field of views of vitrified cells at nanometer resolution. These features allow to combine several scales of understanding of the cellular machinery, from the interactions between groups of proteins to their atomic structure. Cryo-ET therefore has the potential to act as a link between in vivo cell imaging and atomic resolution techniques.

However, cryo-ET images suffer from a high amount of noise and imaging artifacts, and the interpretability of these images heavily depends on computational image analysis methods. Existing methods allow to identify large macromolecules such as ribosomes, but there is evidence that the detections are incomplete. In addition, these methods are limited when searched objects are smaller and have more structural variability.

The purpose of this thesis is to propose new image analysis methods, in order to enable a more robust identification of macromolecules of interest. We propose two computational methods to achieve this goal. The first aims at reducing the noise and imaging artifacts, and operates by iteratively adding and removing artificial noise in the image. We provide both mathematical and experimental evidence that this concept allows to enhance signal in cryo-ET images. The second method builds on recent advances in machine learning to improve macromolecule localization. The method is based on a convolutional neural network, and we show how it can be adapted to achieve better detection rates than the current state-of- the-art.