New Approaches to Cryo-EM Image Processing

Dissertation

for the award of the degree
"Doctor rerum naturalium"
Division of Mathematics and Natural Sciences of the
Georg-August-Universität Göttingen

within the doctoral program
International Max Planck Research School for Genome Science
of the Georg-August University School of Science (GAUSS)

submitted by
Florian Alexander Jochheim
from Paderborn, Germany

Göttingen, 2022

Thesis Advisory Committee

Prof. Dr. Patrick Cramer
Department for Molecular Biology - Genome Transcription and Regulation, Max Planck Institute for Multidisciplinary Sciences Göttingen

Prof. Dr. Sarah Köster
Research Group Cellular Biophysics, Institute for X-Ray Physics, Georg-August-University Göttingen

Dr. Alex Faesen
Research Group for Biochemistry of Signal Dynamics, Max Planck Institute for Multidisciplinary Sciences Göttingen

Members of the Examination Board

Prof. Dr. Patrick Cramer (1st reviewer)
Department for Molecular Biology - Genome Transcription and Regulation, Max Planck Institute for Multidisciplinary Sciences Göttingen

Prof. Dr. Sarah Köster (2nd reviewer)
Research Group Cellular Biophysics, Institute for X-Ray Physics, Georg-August-University Göttingen

Further members of the Examination Board

Dr. Alex Faesen
Research Group for Biochemistry of Signal Dynamics, Max Planck Institute for Multidisciplinary Sciences Göttingen

Dr. Johannes Söding
Research Group Quantitative and Computational Biology, Max Planck Institute for Multidisciplinary Sciences

Prof. Dr. Helmut Grubmüller
Theoretical and Computational Biophysics, Max Planck Institute for Multidisciplinary Sciences Göttingen

Prof. Dr. Michael Altenbuchinger
Department of Medical Bioinformatics, University Medical Center Göttingen

Date of the oral examination: 26.10.2022

Acknowledgements

Foremost, I want to thank Prof. Dr. Patrick Cramer for allowing me to pursue this project. On the same line, I want to thank my Thesis Advisory Committee (TAC): Prof. Dr. Sarah Köster, as well as Dr. Alex Faesen for advising me on my journey through this project. A very special thanks also has to go to Dr. Dimitry Tegunov who has closely supervised me on cryo-EM method development, introduced me to his code base and always helped to oversee the current state of development. Furthermore, I want to thank Dr. Christian Dienemann, who took the time and introduced me to cryo-EM sample preparation and imaging. Mario Klein was always helpful in keeping my much-needed computing infrastructure running. Many thanks also go to the IMPRS-GS with Dr. Henriette Irmer and Frauke Bergmann, and to all members of my group for creating a wonderful working environment.

Thank you also to my everyday love, Annika. Together we have undergone a crazy PhD journey over the last few years. Starting as two freshly graduated students, we are now married and parents of two wonderful twin boys, and soon we shall be two Dr. rer. nat.. I am looking forward to seeing what our future holds for us, but I cannot imagine making this journey without you.

Abstract

Using cryo-electron microscopy (cryo-EM), it is possible to resolve structures of biological macromolecules by averaging numerous projections of macromolecules (particle images) frozen in vitrified ice and imaged using the electron microscope. Developments over the last years have thereby sparked a so-called "resolution revolution". Nowadays, reaching (near) atomic resolution for a large range of protein complexes has become a standard practice for structural biology and a complementing technique for X-ray crystallography. These developments entail developments in microscope hardware, such as direct electron detectors, that allow for single electron detection (counting) and therefore a much reduced radiation dose. Algorithmic developments as well as improvements in computer hardware, however, have greatly improved the *in silico* processing of samples. Especially relevant is the ability to quickly process datasets of hundreds of thousands of particles by using efficient processing algorithms as well as computation on graphic processing units (GPU). Due to the low electron dose used in cryo-EM experiments today, the signal-to-noise ratio is low and accurately reconstruction of the high resolution features entails, among other things, to average numerous particle images to increase this ratio.

In single particle cryo-EM each particle image originates from a biological copy of the macromolecule under investigation. Ideally, these copies would be exact, i.e. each macromolecule is exactly the same as all other copies. Otherwise, during reconstruction, averaging of projections will result in blurry reconstructions. In reality, however, this assumption cannot hold. Especially when active proteins or protein complexes are under investigation, it can be expected that the set of projection images originated from a homogeneous set

of macromolecules that are in different conformational states, have different occupancy or have certain regions that could undergo free movement prior to being frozen in the ice. Effort has gone into the development of classification techniques which aim to divide the dataset in such a way that within each class, the assumption of exact copies holds again. Alternatively, subsections of the structure that are in themselves rigid are refined individually. Both approaches only handle free movement suboptimally, though. Classification cannot divide a dataset such that there is no residual movement within each class and refining only sections of a structure does not give a global reconstruction.

In this work, I explore a pseudo atom based approach to reconstruction. The volume to be reconstructed is represented using a pseudo atom cloud. When two projection images originate from different macromolecules, movement of the pseudo atoms can model the difference between the two when using the projection images to update the intensities of the pseudo atoms. With this, it is possible to reconstruct a single volume from a dataset with projections originating from different macromolecule states. This approach was fully incorporated into a deep learning framework. This enables further development into an end-to-end machine learning approach.

Deep learning algorithms are thereby a promising class of algorithms for cryo-EM processing. In a separate chapter, I explored the possibility of using a generative adversarial network instead of conventional ab-initio reconstruction algorithms. In this approach, a generator learns a volume that represents the real protein in the experimental images, without seeing the experimental images directly. This approach already shows promising results. The learned volume is accurate enough that it can be used as a reference for subsequent high resolution refinement. Multiple parts can still be added and improved to further increase the usability of this approach.

In the last part of this thesis, an approach to identify dimeric particles in a homogenous dataset of monomeric and dimeric SARS-CoV-2 RdRp particles is presented. The approach was successfully applied to identify enough dimeric particles to reconstruct the dimeric state with 5.5 Å resolution and subsequently publishing it. Fitting previously

published RdRp structures allowed for a closer examination of this dimeric state. Furthermore, we hypothesize that this dimeric form might be functional and plays a role in subgenomic RNA production.

Table of Contents

List of Figures

List of Abbreviations

2D	two-dimensional
3D	three-dimensional
Adam	adaptive moment estimation
AI	artificial intelligence
ALC1	Amplified in Liver Cancer 1; a PAR-dependent nucleosome sliding enzyme
API	application programming interface
APPLE	automatic particle picking with low user effort
COVID-19	coronavirus disease 2019
CTF	contrast transfer function
CUDA	Compute Unified Device Architecture
DED	direct electron detector
EM	electron microscope
EMDB	Electron Microscopy Data Bank
EMPIAR	Electron Microscopy Public Image Archive
FSC	Fourier shell correlation
FT	Fourier transform
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
HEALPIX	Hierarchical Equal Area isoLatitude Pixelation of a sphere

IFT Inverse Fourier Transform

KL Kullback-Leibler (divergence)

MLP Multi-Layer Perceptron

NN Nearest-Neighbor

PCA Principal Component Analysis

PDB Protein Data Bank

RELION REgularised LIkelihood OptimisatioN

RELU Rectified Linear Unit

RNA Ribonucleic acid

SARS Severe acute respiratory syndrome

SGD Stochastic gradient descent

SIRT simultaneous iterations reconstruction technique

SNR signal-to-noise ratio

TFIIH Transcription factor II Human

TRS transcription regulatory site

UCSF University of California San Francisco

WGAN Wasserstein GAN

XMIPP X-windows based microscopy image processing package

1 General Introduction

1.1 Single Particle Cryo-Electron Microscopy

Using electron microscopy to determine the structure of biological macromolecules goes back to the works of Richard Henderson and Joachim Frank in the 1970s [Henderson and Unwin, 1975; Frank, 1973, 1975]. Numerous copies of the same macromolecule are frozen in amorphous ice and imaged using an electron beam. This results in 2D projections of the Coulomb potential of these macromolecules (particle images). Each individual macromolecule contributes one projection image. Only through multiple copies of the macromolecule frozen in the same sheet of ice, a distribution of different projection orientations is achieved. After data pre-processing, refinement algorithms are used to calculate the original 3D Coulomb potential [Marques et al., 2019], sometimes also referred to as a density map analogous to the result from X-ray crystallography. Processing and refinement algorithms thereby have to cope with a number of imaging artifacts, as well as unknown orientations for each particle image. Developments in the last years have sparked the so-termed "resolution revolution" [Kühlbrandt, 2014]. This term combines developments in microscope hardware such as Direct Electron Detectors (DED) but also developments in computational methods that combined enable to obtain density maps from cryo-electron microscopy (cryo-EM) data at near-atomic resolution. With modern cryo-electron microscopy methods, resolutions of 3 Å to 4 Å can be reached frequently, e.g. for [Merk et al., 2016; Nogales, 2016]. At this resolution, atomic models can be fitted to the obtained Coulomb potentials with high accuracy and cryo-EM has become a viable

method in structural biology, which had long depended on imaging crystallized proteins by X-ray diffraction for high resolution structure determination [Lyumkis, 2019].

In contrast to crystallography, however, cryo-electron microscopy does not rely on crystallizing biological macromolecules first. This is especially relevant for classes of proteins that have proven to be difficult to crystallize, e.g. membrane proteins [Cheng, 2018]. But also for proteins for which crystal structures are available, cryo-EM can provide a more heterogeneous picture of possible confirmations [Wang and Wang, 2017]. In cryo-EM, samples with numerous copies of the same macromolecule are blotted onto grids and frozen in liquid ethane to ideally obtain a thin layer of ice containing distinguishable, single macromolecule particles. Imaging of these thin layers of ice through the electron microscope yields 2D images containing projections of the Coulomb potential from multiple particles that are separated *in silico* to yield images of individual particles. These are then subjected to data processing and refinement algorithms to yield a reconstruction of the macromolecule's density map.

1.1.1 Image Formation

Biological specimens show very little amplitude contrast in an electron microscope. Amplitude contrast is caused by electrons being scattered outside the objective aperture, which is only a weak effect in thin samples [Erickson and Klug, 1971]. While the amplitude contrast is directly measurable, it would carry not enough information for high resolution structure determination in the downstream processing. The more prominent phase contrast, resulting from the electron beam passing through the Coulomb potential of the specimen, is, however, not visible on the imaging plane when performing in-focus imaging. In cryo-electron microscopy, images are therefore acquired at underfocus or, alternatively, a phase plate is introduced on the optical axis [Wang and Fan, 2019]. Both techniques have in common that they convert the phase contrast into an amplitude contrast in the imaging plane (the camera). The introduced phase shift between the scattered and unscattered beam then causes interference between, which directly alters the strength of the

measurable signal in the imaging plane. The interference does thereby alter with varying spatial frequency. This effect results in a contrast that oscillates between ± 1 following a sine function, which is termed the Contrast Transfer Function (CTF) [Frank, 1973]. The oscillation thereby depends on the defocus value. The characteristics of the CTF are the oscillation between ± 1, but even more importantly '0' crossings in between, which indicate a complete loss of signal for certain spatial frequencies. The speed of oscillation and therefore also the location of these '0' crossings is mostly determined by the spherical aberration (usually fixed for one microscope during imaging) and the defocus value. These can be altered during imaging and can vary within a single micrograph, as the specimens usually do not lie on a single, perfectly perpendicular plane with respect to the electron beam. In the frequency domain, the signal is multiplied with the CTF, while in the spatial domain its effect presents as an offset of signal side bands from their true location [Downing and Glaeser, 2008]. Commonly, the CTF in the frequency domain is expressed as a symmetrical, real-valued function, but higher order aberrations and asymmetries can be accounted for using Zernike polynomials [Zivanov et al., 2020]. With direct electron detectors employed in modern cryo-electron microscopes, the signal measured is a distribution of electron counts across the individual pixels of the detector. Mathematically, it would be ideal to have infinitely many electrons to sample the underlying true distribution (i.e. the un-obscured structural information) as well as possible. However, the high energy electrons that are used in imaging can cause radiation damage in the biological specimens when energy transfer due to inelastic scattering occurs and the transferred energy exceeds the bonding energy of individual atoms in the structure. This highly limits the appliable dosage in imaging and typically the dosage used is only in the range of $20 - 30 \mathrm{e}^{-}/\mathrm{Å}^{2}$. Even at these controlled levels of exposure, there still is an effect of energy transfer that displaces individual atoms. This effect can be compensated by a frequency dependent dampening in the frequency domain [Grant and Grigorieff, 2015]. Using such a low dosage is also the source of so-called "shot-noise", which stems from the fact that we only obtain a noisy estimate of the true underlying distribution of electron counts, and is the main source of noise in cryo-EM images. The first step in data acquisition is usually the collection of so-called "movies" which are individual frames that only contain a fraction of the

total dose information, i.e. if the desired dose is $30 e^-/Å^2$, then this could be fractionized by acquiring 30 individual images corresponding to $1 e^-/Å^2$. Subsequently, these individual frames can be averaged to obtain the micrograph images. This process and the subsequent analysis will be described in more detail in the next section.

1.1.2 Pre-Processing of Data

The first step after acquisition of movies is averaging. The shot noise can be assumed to have a mean value of 0, while the actual signal has a non-zero mean value. Therefore, averaging of N frames should increase the signal-to-noise ratio by a factor of $\sqrt{N}$. During data acquisition, however, the vitrified specimens undergo motion [Brilot et al., 2012]. This motion can be decomposed into a global motion of entire frames, caused by mechanical instabilities in the stage holding the sample, and local motion. In data processing pipelines aiming for high resolution, both kinds of motion need to be corrected for. Global motion can be accounted for by estimating a single displacement vector for each frame in a movie that minimizes the difference between individual frames [Zheng et al., 2017; Tegunov and Cramer, 2019]. For local motion, a motion model can be fitted to sub-frames, which then allows to remap each pixel prior to averaging. Motion corrected averaging combines the individual frames into the final micrograph images.

Contrast Transfer Function (CTF) parameters can be estimated directly from these micrograph images. Popular tools, such as CTFFIND [Rohou and Grigorieff, 2015] and CTER [Penczek et al., 2014], estimate defocus and astigmatism values per micrograph by fitting the CTF parameters to the power spectrum of the micrograph. Especially in tilted samples, however, local defocus values need to be estimated, as done in the preprocessing tool Warp [Tegunov and Cramer, 2019].

The input for single particle reconstruction algorithms are images that contain only single particles roughly in their center. Therefore, particles need to be located within the micrograph to extract patches containing individual ones. Due to the low signal-to-noise

ratio (SNR) in the images, this process is non-trivial and different classes of algorithms exist that try to tackle this issue. A first distinction can be made between "classical" and "machine learning" approaches. "Classical" methods can thereby be either reference (template) based or template free methods. Template based methods [Scheres, 2015; Roseman, 2004; Hoang et al., 2013; Chen and Grigorieff, 2007] thereby rely on some prior knowledge of expected particle views used for matching against the micrograph. These approaches introduce a certain bias, however. Patches of pure noise that randomly show high correlation values to the expected particle views might be extracted along with true positive matches. In a subsequent refinement, these patches would contribute to reconstructing the expected volume, as highlighted by the "Einstein from noise" example [Henderson, 2013]. Algorithms like the APPLE picker [Heimowitz et al., 2018] try to avoid this pitfall by adaptively choosing templates from the data itself. The idea is that patches of actual particle images will show high correlation with other particle patches, while patches of noise will have low correlation with other patches. This does not prevent other structure's like ice crystals from being identified as particles. Reference free methods [Voss et al., 2009; Zivanov et al., 2018] utilize image filters to identify particle locations on the micrograph, but often lack the ability to correctly filter out image artifacts [Tegunov and Cramer, 2019]. Deep Learning based methods utilize some kind of neural network to identify particle locations on the micrograph. They thereby built on the increasing success of such networks in numerous computer vision tasks in other fields. A popular example would be BoxNet, which is part of Warp [Tegunov and Cramer, 2019]. One key advantage is, that they can be trained to not only detect particles correctly, but also to ignore e.g. ice crystals or other artifacts visible in the micrographs. Once particle locations are determined, 2D patches of uniform size are extracted from the micrographs. The size of these patches is usually a user-defined parameter, which only requires a rough estimate of the expected particle size in Å and knowing the pixel size of the micrograph. The data preprocessing tool Warp, developed by Dimitry Tegunov, combines all these steps into a single automated workflow that can run in real time during data acquisition, allowing for a direct feedback to the operator of the microscope about the current quality of data.

Extracted particle images can be used as input for reconstruction algorithms that calculate estimates of the 3D density maps that gave rise to these projections. To refine the set of particles, different algorithms like 2D and 3D classification can be used to filter out false negative particle picks or images of unwanted particles. Unwanted here can mean incomplete complex assemblies, contamination or other artifacts. These algorithms are not covered here in more detail, however.

1.1.3 Refinement Algorithms

Refinement algorithms aim to calculate a 3D volume with high SNR by averaging multiple particle images [Frank, 1975]. The main assumption used here is that the dataset contains numerous particle images from different poses, which are related with the original 3D density via the Fourier slice theorem. This allows to back project particle images and average them, even from different poses. The challenge thereby is that the particle poses are unknown, posing a computational challenge to estimate the poses and the volume at the same time. Refinement algorithms in use currently [Scheres, 2012b; Punjani et al., 2017] use reference based methods to align particle images to a (coarse) density and calculating iterative updates to improve pose estimations and volume estimation. References can thereby be already known structures from closely related specimens or obtained via ab-initio reconstruction. In a forward model, the reference is projected, subjected to the CTF function and weighted by the spectral signal-to-noise ratio to filter noise. The experimental projections can then be compared with the reference projections to find either the best match or to calculate likelihoods that the reference projection's pose matches the experimental image. This way, for each experimental image, a distribution of possible poses can be obtained. The experimental images are then CTF corrected and, using the estimated poses, used in a weighted back projection paradigm to update the reference volume while also performing CTF correction. Iteratively, the updated reference volume can then be used to refine the estimated particle poses. This process can be repeated until a certain number of iterations have been performed or until the resolution does not in-

crease further. The resolution is usually estimated from the Fourier shell correlation curve calculated between two half maps that have been reconstructed from half of the available data [Henderson et al., 2012]. Between these half maps, one can assume no correlation between noise in the reconstruction and positive correlation between signal. Thus, the Fourier shell correlation (FSC) value indicates if, at a certain spatial frequency, the half maps contain mostly signal or mostly noise. Usually, the resolution estimate for these half maps is the point at which the FSC curve falls below the value 0.143 [Rosenthal and Henderson, 2003]. Parts of the reconstructed half maps usually corresponds to solvent only and therefore only contain noise. These parts are usually masked to increase FSC values, but the introduction of false correlation due to features of the mask itself need to be checked carefully [Chen et al., 2013].

1.1.4 Structural Heterogeneity

As described, cryo-EM reconstruction relies on the averaging of many images of the same structure from different view points to increase the signal-to-noise ratio, which is low in raw particle images due to the small electron dose used for imaging. This approach relies on the assumption that all 2D particle images used for the reconstruction actually originate from the same structure. This is difficult to ensure in typical cryo-EM experiments, as all particle images are from distinct biological replicates embedded in the ice. Real protein complexes can have different occupancy, different conformations, or have degrees of freedom that allow for movement of the individual domains. All these can lead to structural heterogeneity, i.e. the presence of multiple different structures as origin for the particle images in a dataset. In case of a distinct number of different structures, e.g. originating from a finite number of different conformations, classification can be used. Early examples of using this approach include [Heymann et al., 2004; Gao et al., 2004] and modern likelihood based 3D classification techniques are implemented in many processing tools [Lyumkis et al., 2013; Scheres, 2012b; Punjani et al., 2017; Grant et al., 2018]. Starting from a single 3D reference, these approaches try to classify the set of

particle images into a user defined number of different classes corresponding to different underlying 3D structures. This approach can work well when the cause for structural heterogeneity is indeed conformational changes or different occupancy, as these give rise to a finite number of structures present in a dataset. In case of movement, however, 3D classification would result in residual structural heterogeneity in each class, which leads to blurry reconstructions in the regions subjected to this movement [Nakane et al., 2018]. Different techniques have therefore been developed to deal with motion induced structural heterogeneity. One approach is to split the 3D volume into individual parts that can be reconstructed individually and describe the motion they undergo across a particle dataset using principal components. This approach is called multi-body-refinement [Nakane et al., 2018]. One can also perform focused refinement, where a mask defines a region of the density that is to be reconstructed separately. In this approach, the reference volume is multiplied with the supplied mask prior to generating the reference projections and then compared with the experimental images from which the regions outside the mask were subtracted [Bai et al., 2015]. The idea behind these two approaches is that the regions of the maps or individual bodies are rigid and can therefore be reconstructed with high accuracy. These approaches have limitations in the kind of flexibility or motion that they can compensate, and applying them requires careful tuning of the user or even a combination of multiple methods to reconstruct different regions of the same structure [Nguyen et al., 2016]. Additionally, the fact that these approaches all rely on a sort of local refinement, there are always border regions for which the refinement might be ill-defined. Only recently, deep learning based approaches have started to emerge that can generate global reconstructions and deal with structural heterogeneity. CryoDRGN uses an autoencoder like structure to encode structural heterogeneity in the latent space [Zhong et al., 2021a], while a method now implemented in cryoSPARC learns a deformation vector field for the reconstructed density [Punjani and Fleet, 2021].

1.2 Artificial Neural Networks

1.2.1 Principal Idea and Differentiable Programming

Artificial neural networks are a type of machine learning algorithm. As such, the general idea is to have an algorithm that is capable of performing a certain task by approximating a function that is not known a-priori and maps data from an input space to an output space. The core idea of these approaches is that instead of writing down an exact formula, a series of operations with different parameters is defined, of which the correct parameters to perform the desired function are not known a-priori. This series of operations is what comprises a neural network. In the simplest forms, these are so-called neurons, which accept multiple inputs that are multiplied by a weight, summed and then added to a so called "bias" to produce the output variable [Rosenblatt, 1957, 1958]. As a simplistic model of how a human brain is organized, multiple interconnected layers form so-called multi-layer perceptron networks that can approximate many mathematical functions. Next to the exact shape of the network, the values of the weight and bias parameters will determine the form of the function that maps input variables of the first layer to the output of the last layer. To obtain these parameter values, the network is "trained" to perform a specific task. For training, a set of training data is needed, i.e. a set of input data and desired output. An example in the context of cryo-EM could be images of micrographs as input and images with annotated particle locations as output. One now needs a procedure to update the parameters of the network in such a way that it best reproduces the mapping of input to desired output given in the training data. Thereby, it is desired that the network does not only perform well in recreating the mapping dictated by the training data, but also show some degree of generalization capability. That is, given some previously unseen input data that is of the same form as the training data, produce an output that is meaningful given that input. In the example of particle picking, this would mean that the network would also be able to pick particle locations in previously unseen micrograph images. To update parameters, usually gradient based methods are used [Bottou, 2010].

One of the key developments that enabled the current success of deep (i.e. comprised of multiple layers) neural networks went into efficient calculation of the gradient of the network's output w.r.t all of its parameters. It is therefore crucial, that the network's operations are all differentiable so that the combination of them, i.e. the complete network, is also differentiable. Modern neural networks are usually comprised of multiple layers and thousands of parameters to train [LeCun et al., 2015]. This makes in unfeasible to write down the function that the network performs and differentiate it analytically (even if mathematically possible). The most important development that enabled modern deep learning was therefore the development of the backpropagation algorithm [Rumelhart et al., 1986]. It works as follows: First, given an input from the training dataset, the output of the network is calculated. Then, some differentiable error function is used to calculate the difference between the network's output and the desired output associated with this training input. The differentiability of the error function can then be used to calculate its gradient w.r.t the network's output. Finally, if the individual building blocks of the network are all differentiable, one can apply the chain rule to back propagate the gradient information through the whole network. In the example of a simple neuron of which the output is a linear function of the output, the gradient information for its input is a linear function of the gradient information of its output. By iteratively propagating the gradient information through all layers of the network, a gradient for each parameter can be obtained and used to update this parameter. Bowadays, backpropagation is part of deep learning frameworks used frequently and is performed computationally efficient due to highly parallelized execution on graphics processing units (GPU)[Paszke et al., 2019; Abadi et al., 2015; Seide and Agarwal, 2016]. Stochastic approaches [Kiefer and Wolfowitz, 1952], where a random batch of training data is presented to the network at once and gradients are averaged over the whole batch, are usually employed to stabilize convergence. Convergence of neural network parameters is mathematically challenging, as the function learned by the neural network is usually highly non-convex and algorithms such as stochastic gradient descent (SGD), Adam [Kingma and Ba, 2014] or AdaGrad [Ward et al., 2020] are used to stably train the parameters.

Using the backpropagation algorithm is thereby applicable to a very broad spectrum of neural networks. The core requirement is that, for each individual operation, it is possible to carry gradient information from its output to its input, i.e. it is a differentiable operation. Some tasks in cryo-EM processing pipelines share resemblance with tasks from computer vision. The already mentioned example of particle picking is a special form of the general question of object detection, for which current network architectures reach human level accuracy [Hestness et al., 2019]. Therefore, advances in different application fields can be ported to tasks in cryo-EM processing pipelines. An example is Warp's [Tegunov and Cramer, 2019] particle picking, which is built using ResNet [He et al., 2016].

1.2.2 Generative and Discriminative Models

Generative models try to generate realistic looking data points. More precisely, given a random input z, they are tasked with generating data x_g that follows some real-world data distribution P_r. The challenge is that P_r is not known directly and can be quite complex. Many developments in this area have been in the area of natural images, an example might be the generation of images of human faces. In this case, P_r would be the distribution of all possible images of human faces, embedded in the space of all possible images. As this distribution is not known, it is also not possible to sample from it directly, hence the need to create and train an artificial generator that mimics the desired sampling. To be able to learn how to mimic data points that follow P_r, generative models usually perform some kind of training on available examples $x_r \sim P_r$. During training, their task would then be to learn an internal representation of the data points that it needs to create artificial data points x_g. To judge the quality of the generator's output, one approach is to define a second network called a discriminator. These two networks, combined to a Generative Adversarial Network (GAN), try to counteract one another [Goodfellow et al., 2014]. The generator generates data points x_g, while the discriminator is presented with real data points x_r and generated ones from the generator and trained to discriminate them. This

is often defined using the following training criterion for the generator G and discriminator D:

$$\min_G \max_D \mathbb{E}_{x \sim P_r} [\log D(x)] + \mathbb{E}_{z \sim P_z} [\log (1 - D(G(z)))] \tag{1.1}$$

The discriminator tried to maximize the distance between its output on the real data and the generated data, while the generator tries to minimize this distance. This corresponds to a 'minimax game', known from game theory [Goodfellow et al., 2014]. These kinds of networks are difficult to train, though. Theoretically, a training following Equation 1.1 is equivalent to minimizing the Kullback-Leibler divergence [Goodfellow et al., 2014]. Training such a network is challenging, though, as P_r usually has a low dimensional manifold as its support that can be disjunct from that of P_g during training [Arjovsky and Bottou, 2017]

1.2.3 Wasserstein GANs

An improvement to the training of GAN networks was the development of Wasserstein GANs, i.e. a Generative Adversarial Network that uses the Wasserstein distance as a distance measure between P_r and P_g. The advantage of using the Wasserstein distance is, that it also provides meaningful gradient information when both probability distributions do not share an overlap [Arjovsky et al., 2017]. With the use of the Kantorovich-Rubinstein duality [Villani, 2009], the Wasserstein distance can be calculated by:

$$W(P_r, P_g) = \sup \mathbb{E}_{x \sim P_r} C(x) - \mathbb{E}_{x \sim P_g} C(x). \tag{1.2}$$

These kinds of network architectures introduce an additional constraint on the discriminator, which is often called a critic C, to be 1-Lifschitz continuous. This constraint can be met with a variety of techniques. In the original WGAN publication, weight clipping was

described [Arjovsky et al., 2017], but other techniques such as gradient penalty [Gulrajani et al., 2017] or spectral normalization [Miyato et al., 2018] have since been developed.

2 Refinement in the Presence of Structural Heterogeneity

2.1 Main Idea

Single particle cryo-EM - although already developed almost 50 years ago [Henderson and Unwin, 1975; Frank, 1975; Dubochet et al., 1982] - has undergone a process termed the "resolution revolution" [Kühlbrandt, 2014] recently. Developments in software and hardware both within the microscope and for data processing have enabled near-atomic resolution for a variety of macro-molecules and complexes, and even atomic resolution for apoferritin [Yip et al., 2020; Nakane et al., 2020]. A main bottleneck that existed when this project was devised was the assumptions of processing algorithms of having a single rigid 3D electron density that is to be reconstructed from 2D projections distorted by the Contrast Transfer Function. Different gold standard software packages had limited capability to cope with this problem in samples where structural heterogeneity is present, as also described in subsection 1.1.4. A first kind of approach relies on classifying heterogeneity into a user-defined number of classes. This is ideal when the structure in the dataset actually has a finite number of discrete state. Such classification schemes are implemented in RELION's 3D classification [Scheres, 2012b] and cryoSPARC's [Punjani et al., 2017] heterogeneous refinement. To overcome this problem, local refinement of subunits that are in themselves rigid is an option. This is then called masked refinement [Bai et al., 2015]. The main disadvantage of these approaches is that only a fixed number of

different confirmations can be refined (classification) or that a fractionized version of the 3D density is obtained (local refinement) which fails to resolve the global density with high continuity. Structural heterogeneity, however, does not always represent itself as a classification problem, which only captures a subset of scenarios such as different conformations or occupancy. If the subunits in the analyzed structure have a degree of freedom, i.e. they can move against one another, then this can only be modeled with an infinite number of classes. Consequently, the resolution in such areas is lower and movement is a common source of a drop in local resolution. As an alternative, principal component-based approaches have been used to deal with structural heterogeneity by describing motion between rigid subunits that can be refined using masks [Liu and Frank, 1995; Penczek et al., 2011; Tagare et al., 2015]. Principal components cannot accurately describe all types of motion though [Sorzano and Carazo, 2021]. A different approach was employed, for example, for the pre-initiation complex with TFIIH and Mediator. Subunits were selected by minimizing inter-region normal modes and locally refined and combined using masks and pseudo atoms [Schilbach et al., 2017]. More recently, deep learning based approaches have also been explored to deal with per-particle heterogeneity [Zhong et al., 2021a; Punjani and Fleet, 2021].

With this project, we aimed to incorporate the movement into the reconstruction process by utilizing real-space refinement techniques combined with a pseudo-atom representation of the density to be refined. Pseudo atoms $\{\{x, y, z, s\}\}_{k=1...N}$ are thereby a set of N combinations of a position in 3D space $\{x, y, z\}$ and an intensity s. A mapping $f : \mathbb{R}^{4N} \rightarrow \mathbb{R}^{D^3}$ thereby allows to raster the pseudo atoms to a 3D volume with side length D. The pseudo atoms thereby serve a double function. Through the aforementioned mapping f, they form a non-Cartesian representation of the protein density. During the reconstruction process, we can update the intensities of the pseudo atoms, which in turn alters the represented density and improves on its agreement with the experimental data. Additionally, the positions encode for a current movement state of the protein. Altering the positions of the pseudo atoms should thereby model real movement, allowing to incorporate this information into a reconstruction workflow as follows: Given an input

of 2D projections $\{P(x,y)\}_i$ that each were obtained by projecting a 3D electron density $V(x,y,z)$, the following steps are performed.

1. Perform a consensus reconstruction, obtaining a consensus volume V and estimated projection angles $\{\phi,\theta,\psi\}_i$ for each projection P_i

2. Using a separate algorithm, for each image P_i, find an approximate representation of the underlying volume $\tilde{V}_i$

3. Pick one $\tilde{V}_i$ and initialize a set $A_i = \{\{x_{i,k}, y_{i,k}, z_{i,k}, s_k\}\}_{k=1...N}$ of N pseudo atoms. Thereby, the represented density should model $\tilde{V}_i$ as close as possible: $f(A) \approx \tilde{V}_i$

4. Estimate pseudo atom movements starting from A_i:
   Estimate alternate positions of the pseudo atoms such that we have for each $\tilde{V}_j$ a different set of atom positions, but with constant intensities:
   $A_j = \{\{x_{j,k}, y_{j,k}, z_{j,k}, s_k\}\}_{k=1...N}$ with $f(A_j) \approx \tilde{V}_j$

5. Iterative reconstruction, for each projection P_i:

   a) Project the pseudo atoms A_i using the orientation $\{\phi,\theta,\psi\}_i$ obtaining an image $\tilde{P}_i$

   b) Compare $\tilde{P}_i$ and P_i and calculate an update for the pseudo atom intensities

With this, pseudo atoms are "moved" to fit the correct movement state of the protein that has generated the current projection. Intensities can then be updated using information from all projections, independent of the movement state that they represent. The final result will also not be just a single volume $\tilde{V}$, but we can instead map any pseudo atom arrangement A_j to a 3D density. This then allows representing each movement state that is present in the dataset at high resolution. They can be generated using the intensities that were refined using the information from all other projections. This procedure does,

however, need multiple parts, which I addressed during my doctoral research and describe in the following. Namely:

1. A way to estimate for each projection P_i a representation of the underlying 3D density that is good enough for the initialization of the pseudo atoms

2. Define and analyze a mapping f of pseudo atoms, which can be irregularly placed in Cartesian space. This mapping should ensure that densities V and their projections P can be approximated accurately

3. Incorporation of common artifacts like the Contrast Transfer Function (CTF) into the reconstruction process

2.2 Towards Efficient and Differentiable Approach

2.2.1 Tri-linear Interpolation and Stability of Consecutive Rastering

Pseudo atoms can be modeled as a point cloud of positions and their intensities. In contrast to voxels on a Cartesian grid, pseudo atoms can thereby assume any arbitrary position in 3D space. This enables arbitrary movement of pseudo atoms to model the movement of subunits. To create a set of pseudo atoms, a 3D binary mask M of the density is used. The procedure to initialize the pseudo atoms A does then try to "fill" the mask with closely packed pseudo atoms, i.e. a hexagonal packing of spheres. Given a mask M, its side length D and the number N of pseudo atoms to initialize, the procedure is as written in algorithm 1. At the same time, however, it is desirable to have a Cartesian representation of the density represented by the pseudo atoms or its projections. For the approach here, we decided to use tri-linear interpolation to raster pseudo atom intensities back to Cartesian coordinates. For a set of pseudo atoms $A = \{\{x_k, y_k, z_k, s_k\}\}_{k=1}^{N}$ the rasterization to a 3D volume is described in algorithm 2 and the reverse procedure to update the pseudo atoms' intensities s_k given a volume V is described in algorithm 3.

Algorithm 1: Procedure to initialize a set of N pseudo atoms, given a cubic binary mask M, the number of pseudo atoms N to initialize and the sidelength D of the mask. Iteratively, the radius R is adjusted, and the mask is filled with a dense hexagonal packing of pseudo atoms until either the radius is such that exactly N atoms fit into the mask or 10 iterations have been performed. The final result is then the set of pseudo atoms saved in A and their radius R.

Data: Number of Pseudoatoms $N > 0$, cubic mask M, side length of mask D
Result: Set of pseudo atoms A, Radius r

$a \leftarrow 0$
$b \leftarrow {}^{D}/_{2}$
$N \leftarrow n$
for $it \leftarrow 0$ **to** 10 **by** 1 **do**
  $r \leftarrow \frac{a+b}{2}$
  $s_x \leftarrow 2r$, $s_y \leftarrow \sqrt{3}r$, $s_z \leftarrow {}^{2\sqrt{6}}/_{3}r$
  for $k \leftarrow 0$ **to** ${}^{D}/_{s_x}$ **do**
    for $i \leftarrow 0$ **to** ${}^{D}/_{s_y}$ **do**
      for $j \leftarrow 0$ **to** ${}^{D}/_{s_z}$ **do**
        $x \leftarrow (j + {}^{1}/_{2}((i+k) \mod 2)) \cdot s_x$
        $y \leftarrow (i + {}^{1}/_{3}(k \mod 2)) \cdot s_y$
        $z \leftarrow k \cdot s_z$
        if $M(\lfloor x \rfloor, \lfloor y \rfloor, \lfloor z \rfloor) == 1$ **then**
          $A \leftarrow A \cup \{x, y, z, 1\}$
  if $|A| == N$ **then**
    break
  else if $|A| < N$ **then**
    $b \leftarrow r$
  else
    $a \leftarrow r$

Algorithm 2: Procedure to map a set of pseudo atoms A to a volume $V(x, y, z)$ through tri-linear interpolation, i.e. the intensity of a pseudo atom is divided between the 8 adjacent voxels, weighted by the distance to each voxel.

Data: Pseudo atoms A, side length of volume D, oversampling factor α
Result: Volume V
for $x \leftarrow 1$ **to** $D \cdot \alpha$ **do**
  for $y \leftarrow 1$ **to** $D \cdot \alpha$ **do**
    for $z \leftarrow 1$ **to** $D \cdot \alpha$ **do**
      $V(x, y, z) \leftarrow 1$
for *each* $\{x, y, z, s\}$ *in* A **do**
  $x \leftarrow \alpha \cdot x$, $y \leftarrow \alpha \cdot y$, $z \leftarrow \alpha \cdot z$
  $x_0 \leftarrow \lfloor x \rfloor$, $x_1 \leftarrow \lceil x \rceil$
  $y_0 \leftarrow \lfloor y \rfloor$, $y_1 \leftarrow \lceil y \rceil$
  $z_0 \leftarrow \lfloor z \rfloor$, $z_1 \leftarrow \lceil z \rceil$
  $V(x_0, y_0, z_0) \mathrel{+}= (x_1 - x) \cdot (y_1 - y) \cdot (z_1 - z) \cdot s$
  $V(x_0, y_0, z_1) \mathrel{+}= (x_1 - x) \cdot (y_1 - y) \cdot (z - z_0) \cdot s$
  $V(x_0, y_1, z_0) \mathrel{+}= (x_1 - x) \cdot (y - y_0) \cdot (z_1 - z) \cdot s$
  $V(x_0, y_1, z_1) \mathrel{+}= (x_1 - x) \cdot (y - y_0) \cdot (z - z_0) \cdot s$
  $V(x_1, y_0, z_0) \mathrel{+}= (x - x_0) \cdot (y_1 - y) \cdot (z_1 - z) \cdot s$
  $V(x_1, y_0, z_1) \mathrel{+}= (x - x_0) \cdot (y_1 - y) \cdot (z - z_0) \cdot s$
  $V(x_1, y_1, z_0) \mathrel{+}= (x - x_0) \cdot (y - y_0) \cdot (z_1 - z) \cdot s$
  $V(x_1, y_1, z_1) \mathrel{+}= (x - x_0) \cdot (y - y_0) \cdot (z - z_0) \cdot s$
$V \leftarrow$ Scale V to D

Algorithm 3: Procedure to initialize the intensities for a set of pseudo atoms A, given a cubic volume V that needs to be modeled by the atoms.

Data: Pseudo atoms A, volume V
Result: Pseudo atoms A with updated intensities
$a \leftarrow 0$
$b \leftarrow {}^{D}/_{2}$
$N \leftarrow n$
for $k \leftarrow 1$ **to** $|A|$ **do**
  $x \leftarrow A[k].x$, $y \leftarrow A[k].y$, $z \leftarrow A[k].z$
  $x_0 \leftarrow \lfloor x \rfloor$, $x_1 \leftarrow \lceil x \rceil$
  $y_0 \leftarrow \lfloor y \rfloor$, $y_1 \leftarrow \lceil y \rceil$
  $z_0 \leftarrow \lfloor z \rfloor$, $z_1 \leftarrow \lceil z \rceil$
  $V_{00} \leftarrow V(x_0, y_0, z_0) \cdot (x_1 - x) + V(x_1, y_0, z_0) \cdot (x - x_0)$
  $V_{01} \leftarrow V(x_0, y_1, z_0) \cdot (x_1 - x) + V(x_1, y_1, z_0) \cdot (x - x_0)$
  $V_{10} \leftarrow V(x_0, y_0, z_1) \cdot (x_1 - x) + V(x_1, y_0, z_1) \cdot (x - x_0)$
  $V_{11} \leftarrow V(x_0, y_1, z_1) \cdot (x_1 - x) + V(x_1, y_1, z_1) \cdot (x - x_0)$
  $V_0 \leftarrow V_{00}(y_1 - y) + V_{01}(y - y_0)$
  $V_1 \leftarrow V_{10}(y_1 - y) + V_{11}(y - y_0)$
  $A[k].s \leftarrow V_0(z_1 - z) + V_1(z - z_0)$

2.2.2 Gradient Descent for Movement Estimation and Reconstruction

Suppose that from step 2 described on page 17 an approximate density $\tilde{V}_i$ is already known for each of our experimental images P_i. Then the set of pseudo atoms can be initialized with the procedure described in the previous section for one specific $\tilde{V}_j$. For the reconstruction procedure described below, we need to be able to express the differences between $\tilde{V}_j$ and $\tilde{V}_i$ as a change in the pseudo atom positions. The goal is therefore to move the pseudo atoms from a set $A_j = \{\{x_{j,k}, y_{j,k}, z_{j,k}, s_k\}\}_{k=1}^N$, initialized to best represent $\tilde{V}_j$, to a set of atoms $A_i = \{\{x_{i,k}, y_{i,k}, z_{i,k}, s_k\}\}_{k=1}^N$ that best represent $\tilde{V}_i$. According to algorithm 2 the value $V(x, y, z)$ of a voxel is linearly dependent on the positions of the pseudo atoms that are adjacent to it. This allows to calculate gradients and a gradient descent algorithm to update the positions of the pseudo atoms. Suppose that we have some function $f(\tilde{V}_i, \tilde{V}_j)$ that measures a goodness of fit between the two volumes. This could for example be a squared error function:

$$f(\tilde{V}_i, \tilde{V}_j) = \sum_{x,y,z} \left(\tilde{V}_i - \tilde{V}_j\right)^2 \tag{2.1}$$

Obtaining a gradient for a specific coordinate, e.g. x_k, can be done by applying the chain rule. Suppose we fix the volume $\tilde{V}_i$ and want to update $\tilde{V}_j$:

$$\frac{\partial f}{\partial x_{j,k}} = \frac{\partial f}{\partial \tilde{V}_j} \frac{\partial \tilde{V}_j}{\partial x_{j,k}} \tag{2.2}$$

The partial derivative $\partial \tilde{V}_j / \partial x_{j,k}$ can thereby be calculated as a sum of gradients obtained from all voxels that the pseudo atom k contributed intensity to in algorithm 2. The gradient information can then be used to update intensities using the Adam algorithm [Kingma and Ba, 2014], see algorithm 4.

Algorithm 4: Procedure to iteratively update the positions of a pseudo atom set A so that they best represent a volume V. In each step, a current volume V_c is calculated and compared with the reference volume V. Gradients of the squared difference w.r.t. each atom's position can be calculated as the derivative of the tri-linear interpolation (not shown). The atom positions are then updated using Adam [Kingma and Ba, 2014] for first and second moment correction. The updating process is iterated for different oversampling parameters. In this way, the atoms can first be moved on a coarse Cartesian representation of the volume and iteratively updated to higher resolution.

Data: Pseudo Atoms A, reference volume V, side length of the volume D, list of scaling factors $\{\alpha\}_{j=1}^{n}$

```
/* Adam parameters */
```

$\alpha \leftarrow 0.001$
$\beta_1 = 0.9$
$\beta_2 = 0.999$
$\epsilon = 1e-8$
for $j \leftarrow 1$ **to** n **do**
  $m_0 \leftarrow 0$
  $v_0 \leftarrow 0$
  for $i \leftarrow 1$ **to** 100 **do**
    $V_s \leftarrow$ Scale V to $\alpha \cdot D$
    $V_c \leftarrow$ rasterize with algorithm 2 using A, D, α_j)
    error $= (V_c - V)^2$
    $g \leftarrow \nabla$error `// Gradient w.r.t. atom positions`

    `/* Adam procedure */`

    $m_i \leftarrow \beta_1 \cdot m_{i-1} + (1 - \beta_1) \cdot g$
    $v_i \leftarrow \beta_2 \cdot v_{i-1} + (1 - \beta_2) \cdot (g \cdot g)$
    $\hat{m}_i \leftarrow \frac{1}{1-\beta_1^i} \cdot m_i$
    $\hat{v}_i \leftarrow \frac{1}{1-\beta_2^i} \cdot v_i$
    update $\leftarrow \alpha \cdot \hat{m}_i / (\sqrt{\hat{v}_i} + \epsilon)$; Add update to pseudo atom positions

2.2.3 Reconstruction Procedure

Given a set of pseudo atoms $A = \{x, y, z, s\}$ that represent a density in three-dimensional space, reconstruction is the task of updating the pseudo atom intensities based on experimental images containing 2D projections of an unknown structure. The reconstruction procedure will have roughly three parts:

1. A forward operator that calculates 2D projections of the pseudo atoms

2. A function that compares 2D projections of the pseudo atoms with the reference images/experimental images and defines update information for each pixel in the projections

3. A backward operator that uses the update information in each pixel to update the pseudo atom intensities

In accordance with common convention [Heymann et al., 2005], the projection is thereby parameterized by three angles $\{\phi, \theta, \psi\}$ that describe the viewing direction. The order of rotation is chosen here to be ZYZ, as is used in XMIPP [Sorzano et al., 2004] and RELION [Scheres, 2012b]. Usually, one would need to perform a line integral through a rotated 3D volume to obtain the 2D projection, but an advantage of the representation through pseudo atoms is that the line integral can be replaced with a sum over the pseudo atoms. From the three angles, we can obtain a rotation matrix:

$$R = \underbrace{\begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{Rotate } \psi \text{ around } z\text{-axis}} \cdot \underbrace{\begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}}_{\text{Rotate } \theta \text{ around } y\text{-axis}} \cdot \underbrace{\begin{pmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{Rotate } \phi \text{ around } z\text{-axis}} \tag{2.3}$$

$$= \begin{pmatrix} \mathrm{c}\,\psi \cdot \mathrm{c}\,\theta \cdot \mathrm{c}\,\phi - \mathrm{s}\,\psi \cdot \mathrm{s}\,\phi & \mathrm{c}\,\psi \cdot \mathrm{c}\,\theta \cdot \mathrm{s}\,\phi + \mathrm{s}\,\psi \cdot \mathrm{c}\,\phi & -\mathrm{c}\,\psi \cdot \mathrm{s}\,\theta \\ -\mathrm{s}\,\psi \cdot \mathrm{c}\,\theta \cdot \mathrm{c}\,\phi - \mathrm{c}\,\psi \cdot \mathrm{s}\,\phi & -\mathrm{s}\,\psi \cdot \mathrm{c}\,\theta \cdot \mathrm{s}\,\phi + \mathrm{c}\,\psi \cdot \mathrm{c}\,\phi & \mathrm{s}\,\psi \cdot \mathrm{s}\,\theta \\ \mathrm{s}\,\theta \cdot \mathrm{c}\,\phi & \mathrm{s}\,\theta \cdot \mathrm{s}\,\phi & \mathrm{c}\,\theta \end{pmatrix} \tag{2.4}$$

Here, s and c were used as shorthand for sin and cos respectively in the second line.

This rotation matrix can be used to rotate the pseudo atom clouds before projection along the z-axis. Afterwards, the projection can be simplified to rastering the pseudo atoms to a 2D Cartesian grid, ignoring the z coordinate after rotation. The resulting procedure is listed in algorithm 5.

Algorithm 5: Procedure to create a 2D projection image P, given a set of pseudo atoms A, that model a 3D density with side length D, and rotation matrix R. The rotation is applied to the pseudo atom positions before bilinear interpolation to obtain the projection image P.

Function `ProjectForward`(*Pseudo atoms A, rotation matrix R, side length of projections and volume D*)**:**

$P[1...D][1...D] \leftarrow 0$
for *each* x, y, z, s *in* A **do**
$\vec{r} \leftarrow \{x, y, z\}$
$\vec{r} \leftarrow \vec{r} - \{D/2, D/2, D/2\}$
$\vec{r} \leftarrow R \cdot \vec{r}$
$\vec{r} \leftarrow \vec{r} + \{D/2, D/2, D/2\}$
$x_0 \leftarrow \lfloor x \rfloor \; x_1 \leftarrow \lceil x \rceil$
$y_0 \leftarrow \lfloor y \rfloor \; y_1 \leftarrow \lceil y \rceil$
$P(x_0, y_0) \mathrel{+}= (x_1 - x) \cdot (y_1 - y) \cdot s$
$P(x_0, y_1) \mathrel{+}= (x_1 - x) \cdot (y - y_0) \cdot s$
$P(x_1, y_0) \mathrel{+}= (x - x_0) \cdot (y_1 - y) \cdot s$
$P(x_1, y_1) \mathrel{+}= (x - x_0) \cdot (y - y_0) \cdot s$
return P

As an additional part for the reconstruction procedure, we need a back projection operator, which can again be modeled using a linear interpolation to update the pseudo atom intensities, see algorithm 6. With the forward and backward operator defined, we can write down a simultaneous iterations reconstruction technique (SIRT) like algorithm [Sorzano et al., 2017] to iteratively update pseudo atom intensities, which are initially set to 0 (algorithm 7).

Algorithm 6: Procedure to update the intensities of the pseudo atoms A, given updates to pixel intensities in a 2D image ΔP, a rotation matrix R and volume side length D.

Function `ProjectBackward(`*Updates for each pixel in the projection ΔP, pseudo atoms A, rotation matrix R, side length of projection and volume D*`):`

for *each* x, y, z, s *in* A **do**

$\vec{r} \leftarrow \{x, y, z\}$

$\vec{r} \leftarrow \vec{r} - \{D/2, D/2, D/2\}$

$\vec{r} \leftarrow R \cdot \vec{r}$

$\vec{r} \leftarrow \vec{r} + \{D/2, D/2, D/2\}$

$x_0 \leftarrow \lfloor x \rfloor \; x_1 \leftarrow \lceil x \rceil$

$y_0 \leftarrow \lfloor y \rfloor \; y_1 \leftarrow \lceil y \rceil$

$d_0 = \Delta P(x_0, y_0)(x_1 - x) + \Delta P(x_1, y_0)(x - x_0)$

$d_1 = \Delta P(x_0, y_1)(x_1 - x) + \Delta P(x_1, y_1)(x - x_0)$

$d = d_0(y_1 - y) + d_1(y - y_0)$

$s \mathrel{+}= d$

return A

Algorithm 7: SIRT like algorithm for the refinement process. Given N pseudo atom conformations $\{A_i\}_{i=1\ldots N}$, corresponding to each experimental image $\{P_i\}_{i=1\ldots N}$, thereby the intensities of the pseudo atoms are shared between different conformations. Additionally, we know the rotation matrix R_i for each experimental image.

Data: A set of pseudo atoms for each projection $\{A_i\}$, projection images $\{P_i\}$, rotation matrices $\{R_i\}$, side length D, learning rate λ

for $i \leftarrow 1$ **to** N **do**

$\tilde{P}_i \leftarrow$ `ProjectForward` (A_i, R_i, D)

$\Delta P_i \leftarrow \lambda \cdot (\tilde{P}_i - P_i)$

for $i \leftarrow 1$ **to** N **do**

`ProjectBackward` $(\Delta P_i, A_i, R_i, D)$

2.2.4 CUDA Based Highly Efficient Implementation

NVIDIA's Compute Unified Device Architecture (commonly referred to as CUDA) is a parallel computing platform and programming model that allows to write and compile programs that can run instructions on CUDA enabled NVIDIA Graphic Processing Unit (GPU). It allows for highly parallelized and hierarchical execution. A CUDA kernel defines the operation that is to be performed by each thread. Up to 1024 threads can be part of one block, and multiple blocks can form a grid of theoretically up to $2^{16} \times 2^{16} \times 2^{16}$ individual blocks. Within the kernel, the position of the thread in the block and the block's location in the grid is accessible as a 1D, 2D or 3D index, creating an easy possibility to have a parallelization for each element in a 1D vector or multidimensional matrix. For the reconstruction procedure in algorithm 7, I implemented the ProjectForward and ProjectBackward operator using CUDA. In the forward model, when pseudo atom intensities are projected to a 2D plane, the parallelization is performed over different projections as multiple threads working on the same projection would potentially cause race conditions when different threads try to write to the same pixel. In the backward model, when updates are read from the pixels of the 2D plane and fed back to the atom intensities, the parallelization is performed across the pseudo atoms. A parallelization across projections would in this case create race conditions when different threads try to update the same pseudo atom's intensity.

2.3 Results on 20s Proteasome

I downloaded a known density of a 20S Proteasome that was published in the Electron Microscopy Database (EMDB, Lawson et al. [2016]) under the accession code EMD-9233. 20S is the 700 kDa core particle of the Proteasome and is frequently used as a model in cryo-EM method development [Punjani et al., 2017; Campbell et al., 2015; Zheng et al., 2017]. The density was reported with a resolution of 2.1 Å, a pixel size of 0.6616 Å/pixel, and a box with a side length of 400 pixels or 264.64 Å. For the tests performed during the development, the original density was down sampled to a pixel size of ~ 2 Å/pixel using Fourier cropping, resulting in a box side length of 134 pixels. A soft mask was generated by thresholding the down sampled volume at 0.01, expanding the resulting mask by 6 voxels and generating a soft edge using a raised cosine filter. The down sampled density and the generated mask were then used as references for the subsequent analyses.

2.3.1 Benchmarking Representation with Pseudo Atoms

The pseudo atom representation needs to be an accurate representation of the density. To test the achievable accuracy, we can compare the density represented by pseudo atoms with the reference, which is represented on a Cartesian grid. To this end, pseudo atoms can first be generated from the reference mask according to algorithm 1 and subsequently be updated with the correct intensities according to algorithm 3. Subsequently, the pseudo atoms can be rastered back to a Cartesian grid according to algorithm 2 and can be compared to the original density, e.g. using the Fourier shell correlation (FSC), which is a common measure for resolution estimation when comparing cryo-EM obtained densities [Henderson et al., 2012]. In Figure 2.1, the FSC values between the pseudo atom represented density and the reference are shown for different setups. Figure 2.1a shows the results when using different numbers of pseudo atoms. As is expected, a certain minimum number of pseudo atoms is needed to accurately represent a density. Here, somewhere between 200 000 and 500 000 atoms seem to be sufficient. This is on the same order of

magnitude than the 305 000 non-zero voxels present in the reference mask. Further, using an oversampling setting > 1 seems to increase the accuracy of the represented density (Figure 2.1b), and performing consecutive rastering of the form Cartesian $\rightarrow$ pseudo atoms $\rightarrow$ Cartesian does not seem to reduce the accuracy of the represented density (Figure 2.1c).

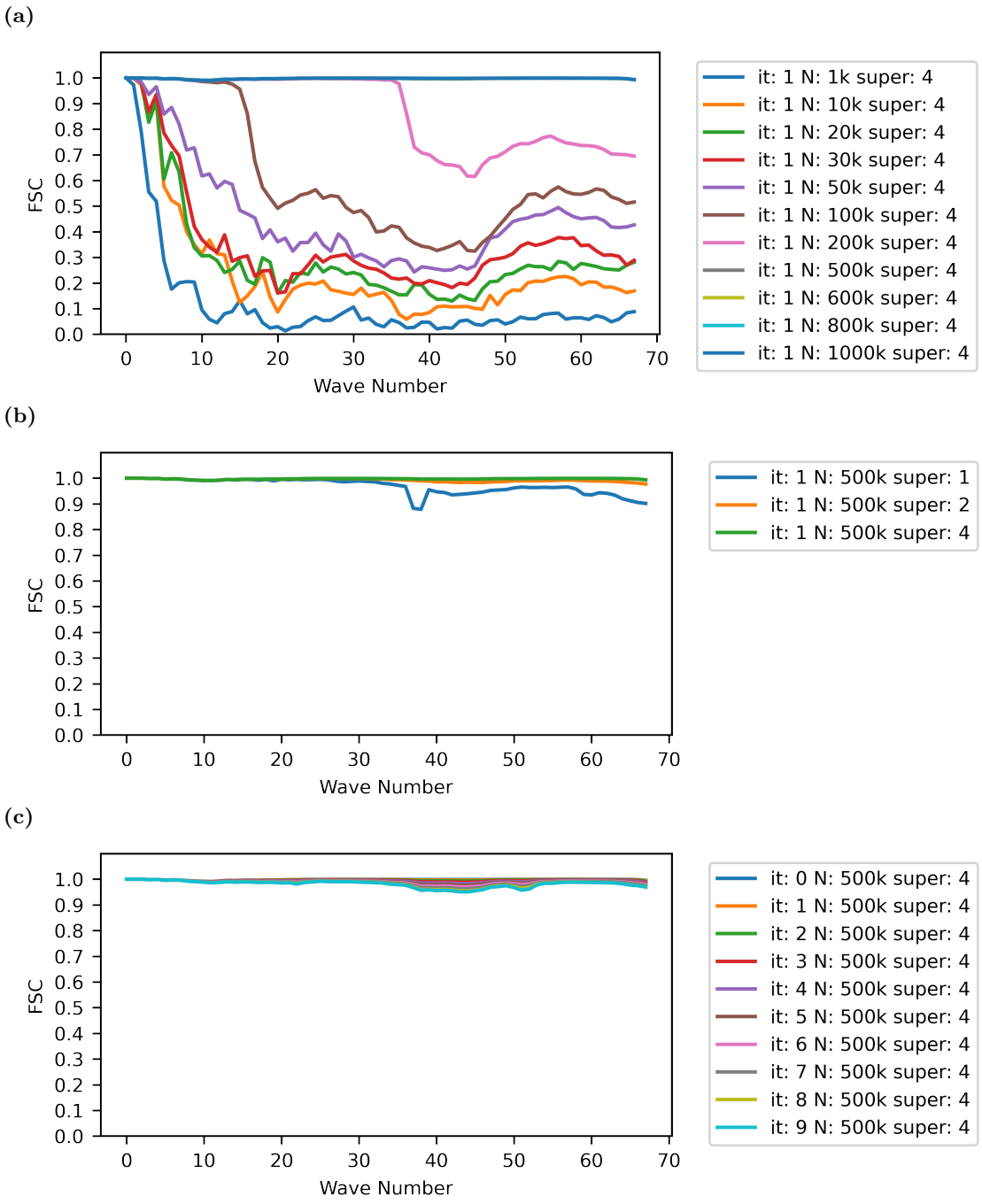

Figure 2.1: Analysis of the accuracy and stability of representing densities using pseudo atoms
To access the accuracy of the pseudoatom represenatation of a density, they can be rastered back to a Cartesian grid and then compared to the reference density using Fourier Shell correlation (FSC). Here, different representations were generated from a downsamples density of a 20S Proteasome. **a)** Representing the density using different numbers of pseudo atoms. **b)** Using different settings for the supersampling parameter. **c)** Consecutive rastering of pseudo atoms → Cartesian grid → pseudo atoms.

2.3.2 Reconstruction with Noise

A critical characteristic of cryo-EM images is the high prevalence of colored noise. The signal-to-noise ratio can thereby be as low as 0.01. This high level of noise is caused by the small number of electrons which are used for imaging - limited by radiation damage - which results in a large influence of shot-noise [Singer and Sigworth, 2020]. During the reconstruction procedure presented here, the noise does not need to be incorporated in the forward model. In algorithm 7, the experimental projections P_i will be aberrated by noise. In Fourier space, this noise can be modeled as Gaussian with a standard deviation depending on the radius (colored noise). The same noise will then be present in the calculated deviations ΔP_i. When performing the back-projection, however, the update for each pseudo atom will be a sum of all updates gathered from. By the law of large numbers, the strength of noise should decrease with an increase in the number of projections. To test this, I generated a small test dataset at different number of projections and different signal-to-noise ratios. First, I generated projections from the density of EMD-9233, which I downloaded and down-sampled to ~ 2 Å/pixel as described above. I used 3072 projection angles obtained using the HEALPIX algorithm [Gorski et al., 2005]. The projections were then generated using the `Projector` class from Warp [Tegunov and Cramer, 2019], which uses code from RELION [Scheres, 2012b]. From these noise-free ("clean") projections, I generated sets of noisy projections with different signal-to-noise ratios, resulting in sets of 3072 projections with no noise, an SNR of 1 and 0.1 respectively. I used each of these sets as input for my reconstruction procedure. I generated pseudo atoms by filling the mask of EMD-9233, as done above, and initialized their intensities to 0. The reconstruction was run for one iteration. Afterwards, I generated the 3D volume represented by the pseudo atoms. For the evaluation, I calculated the Fourier Shell Correlation with the reference density that was used to generate the projections. The result is presented in Figure 2.2. It shows the expected drop in the Fourier shell correlation, especially towards higher frequencies. The signal strength of the reference probably decreases towards higher frequencies, which would result in a spectral signal-to-noise ratio that is actually much worse towards higher frequencies than lower ones.

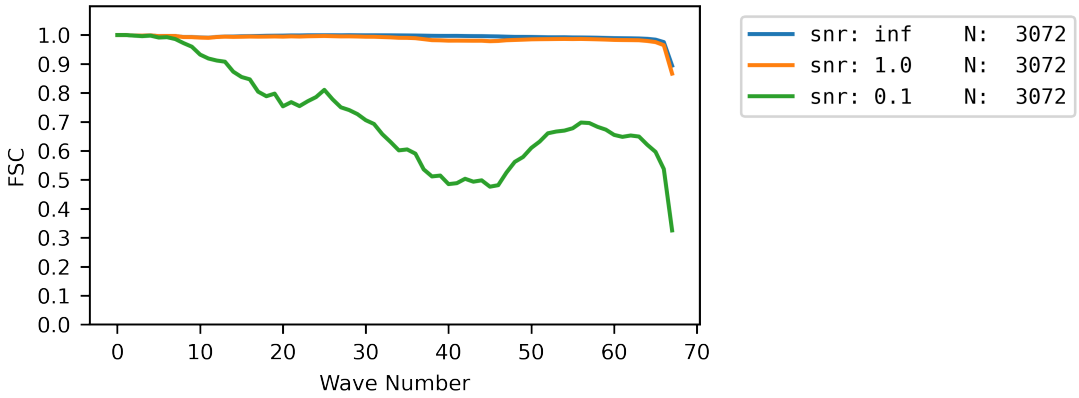

Figure 2.2: Fourier Shell Correlation between a density of 20S Proteasome reconstructed with a fixed number of projections but at different SNR and the reference density.
Starting with pseudo atoms with intensities set to 0, a single iteration of the iterative reconstruction procedure is run with different sets of 3072 projections. They deviate in the strength of added noise. After reconstruction, the density represented by the pseudo atoms was generated and compared to the reference density by calculating the Fourier Shell Correlation which is shown in this plot. SNR inf represents no added noise and shows the highest correlation, comparable to a signal-to-noise ratio of 1. At a signal-to-noise ratio of 0.1, however, the 3072 projections are not enough to average out the effect of the noise effectively, which reduces the correlation with the reference density.

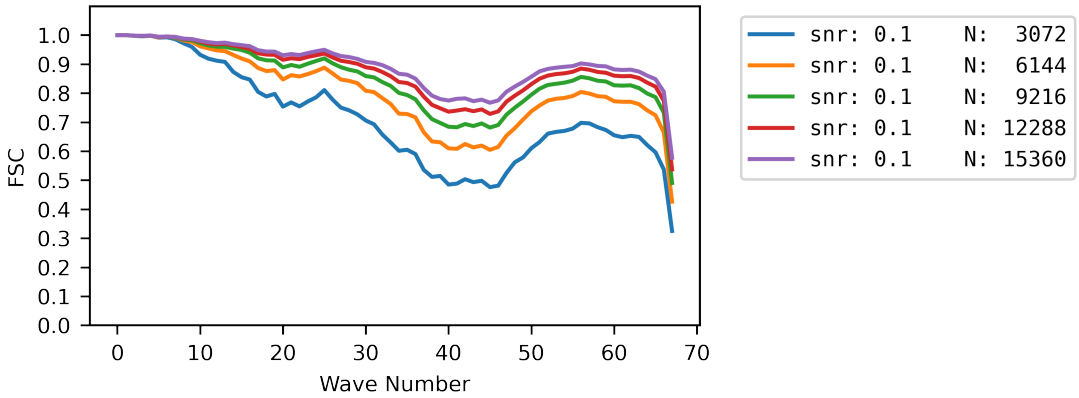

Figure 2.3: Fourier Shell Correlation between the 20S Proteasome reference density and a density reconstructed from projections with a SNR of 0.1 with an increasing number of projections.
Increasing the number of projections increases the correlation of the reconstructed density with the reference. This can be expected from the law of large number, which predicts that the noise in the different projections should decrease with $1/N$.

To test the effect of an increasing number of projections, I followed a similar scheme. For each simulated signal-to-noise ratio, I simply increased the number of projections before the noise generation and then ran the reconstruction for each resulting set. In Figure 2.3 the FSC of the density represented by the pseudo atoms after one iteration with the reference density is shown for the lowest simulated SNR of 0.1. As is expected, an increase in the number of projections does indeed result in a recovery of accuracy. The standard deviation of the noise is expected to decrease with σ^2/n. This would mean that with an ever-increasing number of projections, the noise level will decrease slower and slower, which can also be seen in Figure 2.3. Adding 3072 projections to the initial 3072 has a higher impact than increasing the number of projections from 12k to 15k.

2.3.3 Proof of Concept for Moved Reconstruction

The main advantage of the pseudo atom representation is the ability to use projections from different movement states of the same structure in one high-resolution reconstruction. To test this, I generated a test data set based on the 20S Proteasome density from the EMDB. I here used the down sampled version again.

1. Generate a reference set of pseudo atoms and update their intensities to best represent the 20S Proteasome density. This represents pseudo atom set $\tilde{A}_1$

2. Apply a non-linear displacement to the pseudo atom positions to generate a pseudo atom set $\tilde{A}_2$ that shares intensity values with $\tilde{A}_1$ but with updated positions. For each atom, the position is updated as follows:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x + 3 \cdot (z/134)^2 \\ y + 3 \cdot (z/134)^2 \\ z \end{pmatrix}$$

3. Generate projections from $\tilde{A}_1$ and $\tilde{A}_2$ using evenly distributed projection angles generated using order 4 healpix algorithm [Gorski et al., 2005]

4. Using algorithm 4 transform the set $\tilde{A}_2 \rightarrow A_1$ to best match the original 20S Proteasome density represented by $\tilde{A}_1$.

5. Using algorithm 7, use projection sets P_1 with A_1 and set P_2 with $\tilde{A}_2$ as input

In this way, the reconstruction is performed using projections from the original protein and a deformed version. $\tilde{A}_2$ represents the pseudo atom positions set to the deformed state, whereas A_1 has positions updated by algorithm 4. Using this approach, to update the shared intensities between both sets, should yield a high-resolution reconstruction of the original density. After the reconstruction, I rastered the pseudo atoms back to their Cartesian representation and calculated the Fourier shell correlation with the original density. As a comparison, I also performed a reconstruction using Fourier back projection that cannot incorporate movement information during the reconstruction. In Figure 2.4 the resulting FSC curves can be seen. While the Fourier back projection cannot reconstruct the density at high resolution, the pseudo atom-based reconstruction can incorporate the movement information and can therefore reconstruct the 20S Proteasome density accurately.

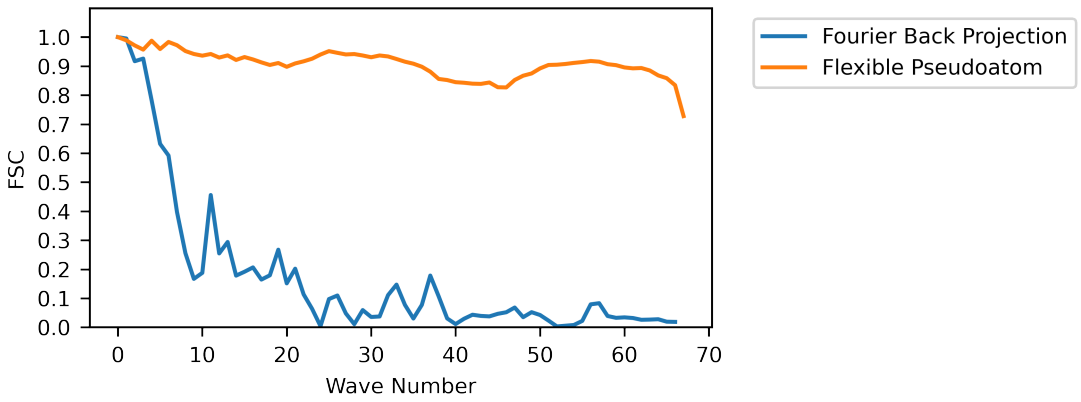

Figure 2.4: Fourier Shell Correlation between the 20S Proteasome reference density and a density reconstructed from projections obtained from the original density and one representing a moved variant.
The gold stanadard Fourier back projection algorithm is not able to correct for the fact that the projections used in this reconstruction were obtained from slightly different variants of the same density. Consequently the FSC (blue curve) quickly falls off. As an alternative, the pseudo atom based refinement using estimated atom positions for the moved density variant, can reach a high FSC with the reference density across all wave numbers.

2.4 CTF Correction During the Refinement

The signal obtained in cryo-electron microscopy is convolved with the Contrast Transfer Function (CTF), which is a result of defocus and spherical aberration [Erickson and Klug, 1970]. When trying to reconstruct the correct 3D densities from the 2D projections, we therefore need to perform CTF correction to reach high resolution [Zhang, 2016]. Fourier back projection techniques can perform CTF correction directly, as the signal is multiplied with the CTF in Fourier space. The contribution of a specific spectral component $\hat{P}(x, y)$ to the Fourier transform of the reconstructed density can be simply weighted by the value of the corresponding CTF belonging to the projection P. In the flexible refinement framework, the reconstruction is calculated in real space, where the influence of the CTF is a convolution with the original signal. While this is computationally more complex than a simple multiplication, it can be incorporated by phrasing the reconstruction problem as an optimization of a target function. This can be a squared error function between the experimental images and the projections obtained from the pseudo atoms. Given a set of pseudo atoms A, experimental projection images $\{P\}_i$, which are assigned Euler angles

$\{\alpha\}_i = \{\phi, \theta, \psi\}_i$, an estimation of the CTF for each projection $\{\mathrm{CTF}\}_i$, we can generate the corresponding projections from

$$\tilde{P}_i = \mathrm{CTF}_i \times \mathbb{P}_{\alpha_i} f(A = \{\{x, y, z, s\}\}_{k=1\ldots N}). \tag{2.5}$$

Here, $\times$ denotes a convolution, $f(A)$ is the volume represented by the pseudo atoms A and $\mathbb{P}_{\alpha_i}$ is the projection operator with Euler angles α_i. Given M experimental images, we need to update the pseudo atom intensities to minimize a target function $t(s)$, which depends on the intensities of the pseudo atoms

$$t(s) = \sum_{i=1}^{M} (\tilde{P}_i - P_i)^2 \tag{2.6}$$

$$t(s) = \min_s \sum_{i=1}^{M} (\mathrm{CTF}_i \times \mathbb{P}_{\alpha_i} f(A = \{\{x, y, z, s\}\}_{k=1\ldots N}) - P_i)^2, \tag{2.7}$$

i.e.

$$s = \arg\min_s t(s) = \arg\min_s \sum_{i=1}^{M} (\mathrm{CTF}_i \times \mathbb{P}_{\alpha_i} f(A = \{\{x, y, z, s\}\}_{k=1\ldots N}) - P_i)^2 \tag{2.8}$$

All of these operations are differentiable:

- $\mathbb{P}_{\alpha_i} f(A = \{\{x, y, z, s\}\}_{k=1\ldots N})$ is the projection of the pseudo atom represented density under the orientation α_i. It is bi-linear interpolation and therefore a simple linear dependence of the pixel values on the atom intensities s (see algorithm 5)

- The convolution is differentiable and can be calculated by multiplying the Fast-Fourier Transform of the projection with the Fourier transform of the CTF and then performing an inverse Fourier transform

By application of the chain rule, it is possible to calculate $\frac{\partial t}{\partial s}$ and use it to update the intensities. Gradient descent for a series of differentiable calculations is a frequent task in the context of training (deep) neural networks and therefore highly efficiently implemented

in deep learning frameworks. One of these frameworks is the deep learning framework PyTorch [Paszke et al., 2019]. In the next section, I will therefore describe steps taken to be able to implement the described reconstruction procedure using PyTorch.

2.5 Incorporation of Pseudo Atom Based Refinement into a Deep Learning Framework

As mentioned above, I implemented the described reconstruction procedure using PyTorch [Paszke et al., 2019]. PyTorch is a modular framework to define, train and evaluate neural networks. The definition of the network thereby means chaining a series of mathematical operations (with parameters) that produce a single output in the end. The strength of these frameworks is thereby that they automatically keep track of the necessary steps to perform a so called back-tracking to calculate the gradient of this single output with respect to the parameters. In the case of a reconstruction network, the single output is a scalar error function between experimental images and those obtained from the forward operator applied to the pseudo atom cloud. One can use the function t already given in (2.8). The parameters of the mathematical operations that need to be trained are the intensities of the pseudo atoms.

PyTorch contains the automatic differentiation package Autograd. Autograd provides automatic differentiation for scalar valued functions with arbitrarily shaped input. One needs to define the function as a series of Autograd operations, and the system is then able to automatically perform gradient calculation of the output with respect to the function's input and all its parameters. One key feature of the PyTorch and its underlying C++ API LibTorch is thereby an interface through which custom operations can be added to Autograd. More precisely: given an input ω_I that can be a single multidimensional input, or a combination of N multidimensional inputs:

$$\omega_I \in \mathbb{R}^{n_1} \cup \mathbb{R}^{n_2} \cup \cdots \cup \mathbb{R}^{n_N}$$

we want to define a function F that generates a scalar output ω_O:

$$F\colon \mathbb{R}^{n_1} \cup \mathbb{R}^{n_2} \cup \cdots \cup \mathbb{R}^{n_N} \to \mathbb{R}$$
$$\omega_I \mapsto \omega_O.$$

This function needs to be defined as a series of k individual operations f that can have parameters that I denote as θ. In the simplest form, F can be a chain of such operations:

$$\begin{aligned}
\omega_1 &= f^1_{\theta_1}(\omega_I) \\
\omega_2 &= f^2_{\theta_1}(\omega_1) = f^2_{\theta_1}\left(f^1_{\theta_1}(\omega_I)\right) \\
&\dots \\
F(\omega_I) &= f^k_{\theta_k}\left(f^{k-1}_{\theta_{k-1}}\left(\dots f^1_{\theta_1}(\omega_I)\dots\right)\right)
\end{aligned}$$

Given such a chain of operations (or a more complex combination of operations), Autograd is now able to automatically calculate all gradients, i.e. w.r.t. the input $\partial\omega_O/\partial\omega_I$ and w.r.t all function parameters $\partial\omega_O/\partial\theta_1, \dots, \partial\omega_O/\partial\theta_k$.

A custom operation can be added to Autograd by implementing two functions:

- "forward pass": given an input ω_i, define how to calculate $\omega_{i+1} = f_\theta(\omega_i)$
- "backward pass": given the gradient $\partial\omega_O/\partial\omega_{i+1} = \partial\omega_O/\partial f_\theta(\omega_i)$, apply the chain rule to obtain

$$\frac{\partial\omega_O}{\partial\omega_i} = \frac{\partial\omega_O}{\partial f_\theta(\omega_i)} \cdot \frac{\partial f_\theta(\omega i)}{\partial\omega i} \tag{2.9}$$
$$\frac{\partial\omega_O}{\partial\theta} = \frac{\partial\omega_O}{\partial f_\theta(\omega_i)} \cdot \frac{\partial f_\theta(\omega i)}{\partial\theta} \tag{2.10}$$

For the reconstruction framework presented here, the input for a batch of N experimental images consists of: N experimental images, their orientation given as Euler angles, and corresponding CTF given in Fourier space. The network then performs the following operations to obtain a scalar output:

1. Project pseudo atoms. For each of the N input items, use the corresponding Euler angles and pseudo atom coordinates, as well as the shared pseudo atom intensities to produce super sampled 2D images (4 times larger than experimental images, see subsection 2.3.1)

2. Take the Fast-Fourier Transform of the resulting 2D images

3. Crop the FT to the experimental image dimensions

4. Multiply each of the N projections in Fourier space with the corresponding CTF from the input

5. Take the Fast-Fourier Transform to obtain the 2D real-space images that now have the same dimensions as the experimental images and are convolved with the CTF

6. Calculate the mean squared error between these N images and the experimental images given in the input

The operations needed for steps 2, 4, 5 and 6 were already implemented in Autograd. I added the necessary operations for step 1 (following algorithm 5) and step 3. Furthermore, I also implemented algorithm 2 as an Autograd operation. For all cases, I used CUDA to parallelize the computation using the GPU. With all operations available, they could be chained to produce the desired output.

2.5.1 Reconstruction Test

To test the reconstruction framework in PyTorch and evaluate the CTF correction, I performed a small proof of concept analysis. I again used as reference the density of 20S Proteasome downloaded from the EMDB with accession code EMD-9233, down sampled to $\sim 2\,\text{Å}$/pixel and a box size of 134 pixels as was previously used in. A mask was also generated as described above (see section 2.3). From this reference density, I generated 3072 projections using the `Projector` class from Warp [Tegunov and Cramer, 2019]. The projections were evenly distributed on the sphere using the HEALPIX algorithm [Gorski et al., 2005]. Additionally, I sampled CTF parameters randomly from `shiny.star` provided in EMPIAR-10299 [Casañal et al., 2017] and created a separate CTF image for each projection. I convolved each reference projection with the respective CTF. The resulting projections serve as a reference for the reconstruction process.

I initialized $\sim 500\,000$ pseudo atoms as already done above and set their intensities to 0. For 128 projections at a time, I ran steps 1 through 6 (page 39), used Autograd to calculate the gradients for the pseudo atom intensities and performed stochastic gradient descent with a learning rate of $\lambda = 0.1$ to update them. This was repeated until all projections were processed exactly once. After each of such an iteration, I rastered the pseudo atom intensities to a Cartesian grid and compared the resulting volume with the reference volume previously used to generate the projections. For this comparison, I calculated the Fourier shell correlation. I ran the process for 10 iterations. The FSC values after each iteration is presented in Figure 2.5. The FSC is well above 0.5 for all wave numbers, indicating a refinement up to the Nyquist limit. Furthermore, as single iteration was sufficient, as subsequent iterations did not alter the result.

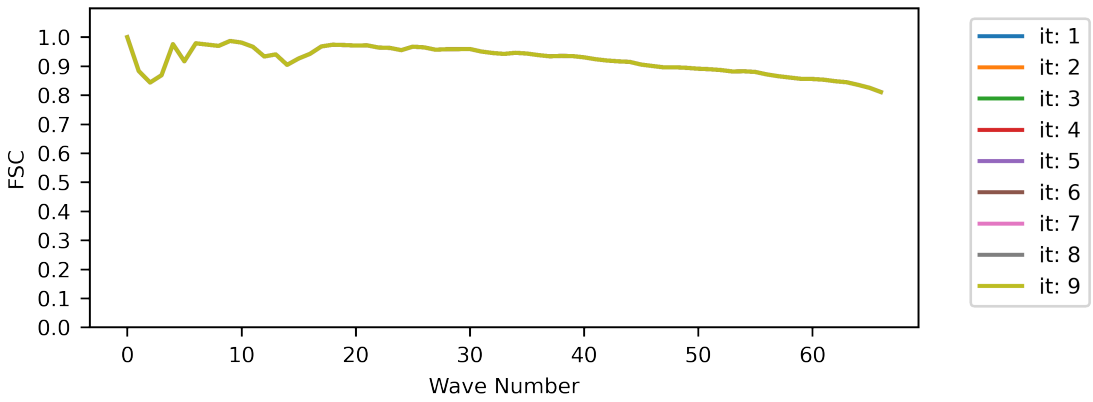

Figure 2.5: Quality of the reconstructed density using the gradient descent based approach for projections that are aberrated by the CTF function.
After processing all 3072 projections, the pseudo atom intensities were rastered to a Cartesian grid and the Fourier shell correlation with the reference density were calculated after masking with a soft mask. Curves for different iterations are all overlapping, indicating no further improvement after the first iteration.

2.6 Discussion and Outlook

Reconstruction of cryo-EM density is the calculation of a 3D electron density map from the 2D projections of particles obtained from the electron microscope. The usual formulation of the problem is that we have a set of 2D particle images P_i that all originate from exact copies of the same particle, i.e. they all represent the same density, projected from an unknown direction. Additionally, particle images are convolved with the Contrast Transfer Function (CTF) and overlaid by a high degree of noise. The Contrast Transfer Function is thereby a direct consequence of the defocus contrast imaging performed in cryo-electron microscopy. The high level of noise is a consequence of the low number of electrons used for imaging to reduce radiation damage to the sample. This results in a high level of shot noise, which is counteracted by averaging multiple movie frames and having a large number of particle images.

In this formulation, the task of reconstruction is an optimization problem to simultaneously find the unknown projection orientations for each particle image and the unknown 3D density. A gold standard algorithm that utilizes Bayesian formulation of this problem was devised by Sjors Scheres [Scheres, 2012a] and implemented in the software package

RELION [Scheres, 2012b], which provides end-to-end functionality to go from raw micrograph images to high resolution reconstructions. A major bottleneck in the above described formulation of the reconstruction procedure, however, is the assumption of exact copies of the same particles that give rise to the particle images. Alternative or incomplete assemblies of protein complexes, different states, as well as flexibility, can thereby be a source of structural heterogeneity in the dataset. Alternative assemblies and different states can thereby often be solved by performing classification of particle images, i.e. sorting them into a fixed number of classes, corresponding to unique underlying structures and performing separate reconstruction tasks that can again assume that all particle images in one class originate from the same density [Gao et al., 2004]. When dealing with structural heterogeneity originating from flexibility, however, there are no longer a finite number of different densities into which we can classify. Instead, having a free movement along one or multiple degrees of freedom will introduce an infinite amount of different densities as a source for the 2D projections. In this chapter, I explored the possibility to reconstruct a single density from a dataset that shows such structural heterogeneity. Thereby, I assumed that we have a system that can estimate for each projection image an at least low-resolution representation of the density that it originated from. I developed a pseudo atom based reconstruction framework that initializes a single set of pseudo atoms comprised of locations in 3D space and intensities, which together represent a density map. By updating the locations, the pseudo atoms can be moved along the same degree of freedom present in the biological specimen. The movement of pseudo atoms is thereby calculated by gradually updating them to fit the required coarse density representation from each projection image. All this information then enable to perform a real-space reconstruction. For each experimental image, the pseudo atoms are moved to best fit this image's 3D density, projected using the same orientation previously estimated for this experimental image and then compared to said experimental image. This gives a per-pixel error which can be back-projected to update the pseudo atom intensities. This can be performed for all experimental images and incorporates movement information. I implemented this scheme using CUDA for GPU accelerated computation of projection images and back projection of updates. I could show the correctness of the implemented operation by reconstructing

a 20S Proteasome from artificially generated projections accurately. Furthermore, I could show that in the presence of noise, the reconstructed density have lower Fourier shell correlation values. This becomes less prominent if the number of projection images is increased, as can be expected. I also showed that it is possible to correctly reconstruct a dataset with structural heterogeneity. From the 20S Proteasome density, I created an artificially moved density, used a gradient descent scheme to estimate pseudo atom coordinates and reconstructed a single density from a combined projection set from both moved and unmoved densities. This showed that it is possible to incorporate movement information into the reconstruction process by using pseudo atoms, and that high resolution reconstruction is in principle possible. Knowing that these operators work in principle for a reconstruction procedure, I made them compatible with PyTorch's Autograd framework for automatic differentiation. This enables to incorporate them into a neural network for a variety of tasks and increases their usability. In this regard, I could already show how this definition could automatically compensate for the effect of the Contrast Transfer Function (CTF) on the reconstruction process.

3 Wasserstein GAN based Ab-Initio Reconstruction

3.1 Core Idea and Related Work

In cryo-electron microscopy data processing, the task of reconstruction is the step to calculate from the 2D particle images X_{ij} a 3D representation of the corresponding macromolecule's Coulomb potential. The typical forward model how these projections are generated in the microscope is described through a linear model in Fourier space [Scheres, 2012a]:

$$X_{ij} = \mathrm{CTF}_{ij} \sum_{l=1}^{L} \mathbf{P}_{jl}^{\Phi} V_{kl} + N_{ij} \tag{3.1}$$

The components are thereby:

X_{ij}	The i-th component of the j-th experimental image
CTF_{ij}	The value of the CTF function for the component ij
$\sum_{l=1}^{L} \mathbf{P}_{jl}^{\Phi} V_{kl}$	The j-th component of the projection operator acting on a Volume V_k, i.e. this operator extracts the j-th component of a 2D slice obtained from the 3D Fourier transform of V_k with the orientation given by Φ
N_{ij}	Additive spectral noise

The experimental images are projections of the structure(s) of interest under unknown orientations. The noise level is usually significant, as the electron dose is kept low to reduce radiation damage to the sample. The combination of unknown orientations, low SNR and unknown volume make the reconstruction a computationally difficult task.

One key step is to obtain an initial model of the volume(s) from which the projections originate [Elmlund et al., 2008; Punjani et al., 2017]. The quality of this ab-initio model can thereby heavily influence subsequent refinement steps and can introduce modelling bias in case of an incorrect representation of the underlying data [Henderson, 2013]. Therefore, it is crucial to have a bias free approach to generate such ab-initio models. Over the course of the last decade, deep learning based approaches have revolutionized many fields in computer science and can reach human level accuracy on many image processing tasks like segmentation and classification [Hestness et al., 2019]. More recently, generative approaches, and more explicitly generative adversarial networks, have also reached high levels of resolution [Goodfellow et al., 2014; Gonog and Zhou, 2019; Yi et al., 2019; Arjovsky et al., 2017]. A generative network is thereby a neural network that can artificially create data points x that follow some real distribution P_r. Generative Adversarial Network, often shortened as GAN, consist of two neural networks that are competing against each other. The first is the generator, a network with a set of parameters θ that maps a random input z to a data point $x = G(z)$ that follows a certain probability distribution $x \sim P_\theta$. Its training objective will ultimately be to best mimic the real data, i.e. $P_\theta \sim P_r$. The key challenge is thereby that P_r is most likely not be known explicitly. E.g. it could just be a category of data, i.e. certain natural images like images of faces. To address this challenge, GANs train a second network called a discriminator D. This discriminator is presented with data points from the real data $x_r \sim P_r$ and those generated from the generator $x_g \sim P_\theta$. Its training objective is to best distinguish data from these two classes. The data points are not directly compared, however. Instead, the training objective can be formulated such that the discriminator is trained to produce different output for both classes. E.g. this could be a binary '1' for $x \sim P_r$ and '0' for $x \sim P_r$, or alternatively 'high' and 'small' (or negative) numbers for the two cases. It does therefore work against

the generator, as their training objectives are counteracting. G tries to achieve output that is indistinguishable from the real output $P_\theta \sim P_r$, while D tries to find a way to distinguish the two. Updates for the generator are calculated as follows: First, a data point $x = G(z) \sim P_\theta$ is generated and presented to the discriminator. If both G and D are fully differentiable, gradients can be calculated for G's parameters θ and θ can be updated such that $D(G(z))$ moves towards a 'high' output, i.e. the generator learns how to better "fool" the discriminator.

Correct training of the discriminator is therefore critical for a successful training of the generator. A special class of GANs are Wasserstein GAN networks that make use of the Wasserstein distance. In these networks the discriminator is called a Critic instead (as it can no longer perform binary classification) and we utilize the Wasserstein distance between P_r and P_θ during training:

$$W(P_r, P_\theta) = \sup_{||C||_L<1} \mathbb{E}_{x\sim P_r}[C(x)] - \mathbb{E}_{x\sim P_\theta}[C(x)] \tag{3.2}$$

The constraint $||C||_L < 1$ is that the Critic needs to be a 1-Lifschitz function mapping a data point x to the output $C(x)$ that is supposed to be large for real data points $\sim P_r$ and small for those generated from fake data points $\sim P_\theta$. The 1-Lifschitz criterion is needed to ensure that the gradient $\partial C(x)/\partial\theta$ convey useful information to actually bring P_θ closer to P_r. In this network architecture, the generator learns to produce realistic looking data, without seeing the real data directly.

In the project in this chapter, we wanted to explore the usage of a Wasserstein GAN to perform ab-initio reconstruction from a set of experimental images. According to the forward model given in Equation 3.1, each of those experimental images show a projection of the electron density of a biological structure. Each image is convolved with the CTF and aberrated by a high degree of noise. The generator will learn a volume and generating artificial data will follow the forward model:

1. The volume is projected under a random angle, creating a ‘clean‘ projection

2. Convolve with a CTF

3. Add noise

The CTF parameters can thereby be sampled from the experimental images. Theoretically, one could also try to learn a correct distribution of CTF parameters from which the generator samples, but as CTF parameters can already be estimated from the micrographs its estimation would unnecessarily complicate the training task. The generator will need to learn a volume that best represents the biological structure that was present in the sample that gave rise to the experimental images. Additionally, it will need to learn a noise model that accurately matches the noise distribution within the experimental images. With the additive noise, there is no direct way for the generator to have an incentive to encode actual structural information only in the volume and not in the noise model. However, this can be counteracted by randomly shifting the sampled noise images prior to adding them to the generated 2D projections.

Using a Wasserstein GAN for cryo-EM reconstruction was previously explored by Gupta et al. [Gupta et al., 2021]. Their approach required to semi-automatically extract “noise” patches from the same micrographs that particles were extracted from, and was only tested with symmetric structures.

3.2 Network Architecture

I implemented a Wasserstein GAN using LibTorch, the C++ API of PyTorch, version 1.7. All network parts were defined with Autograd operators, which enables automatic differentiation and subsequent learning of the network's parameters. The implemented Wasserstein GAN consists of a generator and a Critic network, which I describe in the following.

3.2.1 Generator

The main learnable parameter of the generator is a volume of size D^3, where D is the box length of the volume to be reconstructed. To create a 2D image, this volume first needs to be projected. The projection angles can either be randomly sampled from a uniform distribution on a sphere. Or, one can also empower the generator to learn an uneven distribution of viewing directions. Three random variables (sampled from a Gaussian with unit standard deviation and mean 0) can be processed by a small multi-layer-perceptron (MLP) network with a 3x3 rotation matrix as output. The task of the MLP is thereby to perform distribution transformation from the unit Gaussian distribution to the experimental viewing direction distribution. The actual projection can then be performed as follows. First, we sample a rotated volume, using the `affine_grid` and `grid_sample` which were developed in the context of spatial transformer networks [Jaderberg et al., 2015]. Subsequently, a `mean` operation along the new z axis performs the actual projection operation, creating a "clean" projection of size D^2. At this point, we could have also performed Fourier slice extraction according to the Fourier slice theorem, but is computationally not less expensive than the described method. CTF parameters are randomly sampled from the experimental images, and the "clean" projection is convolved with a CTF. Finally, random noise is generated by processing a random input through a series of MLP and deconvolutions into a noise patch of size D^2. This noise patch is circularly shifted by a random amount of pixels along the x-axis to discourage any structural information being

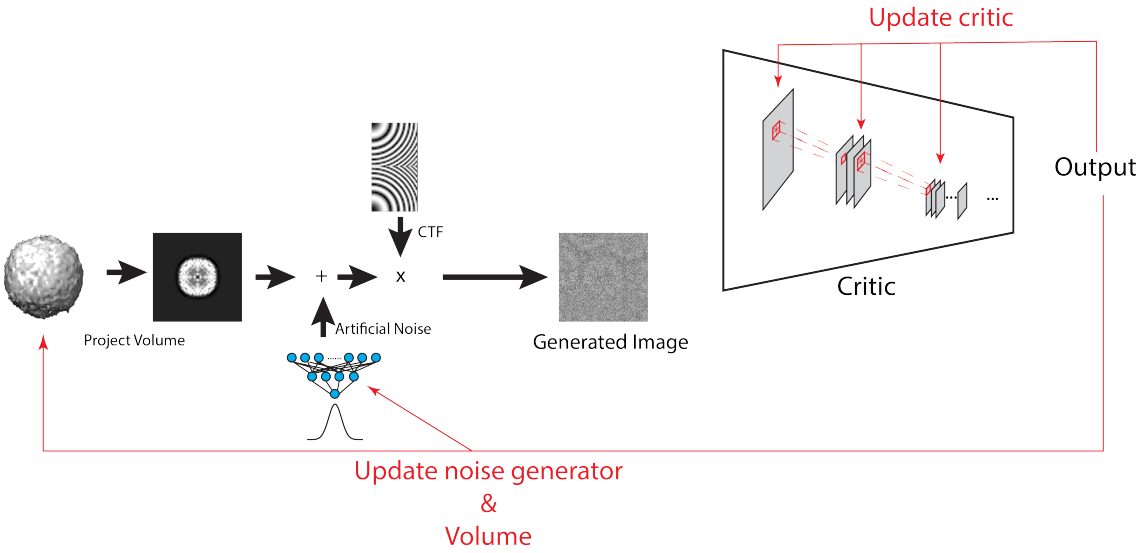

Figure 3.1: Strategy to update the generator during training.
Iteratively, the Critic and generator are updated during the training. The Critic is trained to produce high output on the real images and low output on the generated images. To update the generator, first multiple images are generated from the volume learned by the generator. These "clean" projections are then distorted by noise sampled from the noise network and convolved with the contrast transfer function (CTF). The resulting "generated image" is given to the Critic which generates a numeric output. All these operations are differentiable and back propagation is automatically performed by Autograd. In this way, gradients can be obtained for the weights in the noise generator as well as all voxels in the generator volume.

learned by the noise model. The resulting noise patch is added to the previously convolved "clean" projection. Updates to the noise generator and the volume are calculated from the Critics output on the generated images, see Figure 3.1.

3.2.2 Critic

The Critic consists of a series of convolution and downsampling operations. Each convolution block is a combination of a convolution operation, followed by max-Pooling and a leaky RELU. The convolution's kernel size is set to 3, padding and stride are set to 1. Max Pooling operates with a 2×2 kernel with a stride of 2, effectively reducing the dimensionality by a factor of 2. The Critic consists of 6 consecutive blocks with an increasing number of channels: 96, 192, 384, 768, 1536, 3072. This is followed by a fully connected layer with 10 neurons, a leaky RELU operation and a single neuron that produces the final output. All leaky RELU have a negative slope of 0.1. The main difficulty in training the

Critic in ensuring the 1-Lifschitz continuity. To ensure this, a number of different techniques have been developed, such as weight clipping, gradient clipping, gradient penalty and spectral normalization [Gulrajani et al., 2017; Miyato et al., 2018]. During training, gradient penalty was applied, i.e. the term

$$(||\nabla C(\eta * x_r + (1-\eta)x_g||_2 - 1)^2\lambda$$

is added to the Critic's loss function, which pushes it's gradient to stay close to 1. λ was set to 0.6.

3.3 Results

I processed an experimental dataset with 400k particles of a polymerase complex to obtain a test data set for the GAN based ab-initio reconstruction. All particle images had a side-length of 128 pixel and 3 Å/pixel. After 2D classification with cryoSPARC [Punjani et al., 2017], 2D classes that resembled projections of polymerase were selected, corresponding to 155 550 particles. Subsequently, ab-initio reconstruction and 3D refinement were performed to obtain a 3D reconstruction up to the Nyquist limit. The cryoSPARC reconstructed volume as well as the particle set were exported and 3D refinement was redone in RELION [Scheres, 2012b], as its output contains the parameters of the noise model fit to the data. With this processing, I now had 155 550 particle images with orientations, noise model estimated during the reconstruction and CTF parameters as well as a 3D density with 128^3 pixel at 3 Å/pixel, which I used as reference in the following.

3.3.1 Noise Free Reconstruction

I read all CTF parameters and the noise model into memory. The generator was initialized with a volume of size 64^3, with all entries set to 0. For a first test, the noise model was not

applied in the forward model. All parameters of the Critic network were initialized with PyTorch's[Paszke et al., 2019] default initialization scheme. Gradient based optimization was performed using the Adam [Kingma and Ba, 2014] optimizer with $\alpha = 0.01, \beta_1 = 0.5, \beta_2 = 0.9, \epsilon = 10^{-8}$. To generate 'real' projections, first I sampled a random orientation from the unit sphere and random CTF and noise model parameters from the experimental images. I projected the reference density to 2D images with a box size of 128 pixels, convolved them with the CTF and added noise sampled from the noise model. Finally, the resulting images were down sampled to a box size of 64 pixels, corresponding to 6 Å/pixel. The training of both networks was performed iteratively:

In each step, first the Critic was trained four times: Each time, a batch of 8 'real' images was sampled as well as a batch of 8 images from the generator. The loss to minimize for the Critic, given real images x_r and generated images x_g, was then defined as:

$$L = C(x_g) - C(x_r) + \lambda \cdot (||\nabla C(\eta * x_r + (1 - \eta)x_g||_2 - 1)^2$$

which maximizes $C(x_r)$ and the distance $C(x_r) - C(x_g)$. Additionally, it contains a gradient penalty term, which penalizes a deviation of the Critic's gradient from 1, ensuring 1-Lifschitz continuity [Gulrajani et al., 2017]. For each batch, gradients were automatically calculated by AutoGrad and updates to the Critics parameters were calculated according to Adam. After four times training of the Critic, a single iteration was performed for the generator. For this iteration, a batch of 8 images is generated from the generator, and then presented to the Critic, giving the output $C(x_g)$. The loss to minimize for the generator was then defined as

$$L = -C(x_g)$$

which tries to maximize $C(x_g)$. As for the Critic, gradients were automatically calculated by AutoGrad and updates to the generators parameters were calculated according to Adam.

In total, the network was trained for 14 000 iterations, which took $\sim$ 10 hours on an NVIDIA RTX2800TI graphics card. After training, the volume learned by the generator was extracted and compared with the original reference volume. The comparison was performed by importing both the reference and extracted volume into cryoSPARC and running a 3D alignment job. This job rotates both volumes and calculates the gold standard FSC values. These are shown in Figure 3.2. Using a cut-off of 0.5 for a resolution estimate gives a resolution of 15.8 Å for the reconstruction. The volume that the generator reconstructed had a pixel size of 6 Å, which gives a theoretical limit of 12 Å for the resolution.

3.3.2 Reconstruction in the Presence of Noise

Experimental cryo-EM particle images are aberrated by a high level of noise. To test the ability of the WGAN based approach to still be able to provide a useful ab-initio reconstruction, I expanded the test case described in the previous section. For the 'real' projections, I added Gaussian noise with standard deviation $\sigma = 1$ to the images after convolving with the CTF and prior to down sampling to 64^2 box size. As the reference volume, from which the projections are calculated, is the direct output from a RELION reconstruction, it is normalized such that the noise has $\sigma = 1$ in the experimental images from which the volume was reconstructed. Therefore, adding $\sigma = 1$ noise to the artificially created projections gives a signal-to-noise ratio that is close to the real images. The main difference is that the added noise is white, while the noise in the experimental images is usually assumed to be colored.

The generator also adds Gaussian noise to the projections after convolving with the CTF and prior to down sampling. The code was again run for 14 000 iterations. The final vol-

ume was again extracted, imported to cryoSPARC and evaluated by running an Align3D job with the reference volume. The calculated gold-standard FSC values are shown in Figure 3.3. The resolution is drastically reduced in comparison to the noise free reconstruction shown in Figure 3.2. The main property that an ab-initio reconstruction should fulfill is that it can be a good starting point for a subsequent high-resolution refinement that needs a good starting model to assign projection angles to the particle images. I used the reconstructed density as a reference for a "Refinement New" task in cryoSPARC. The 155 550 particle images from the pre-processing of the reference density were used as input for this reconstruction. This reconstruction algorithm reconstructs two independent half maps from the particles and uses the Fourier Shell correlation between the two for a resolution estimate. Usually, a cut-off of 0.143 in the FSC between both half maps is chosen to assign a resolution value to the reconstruction [Henderson et al., 2012]. The FSC curve was above this threshold for all wave numbers in this dataset, indicating the maximal possible resolution that could be reconstructed from this density (see Figure 3.4).

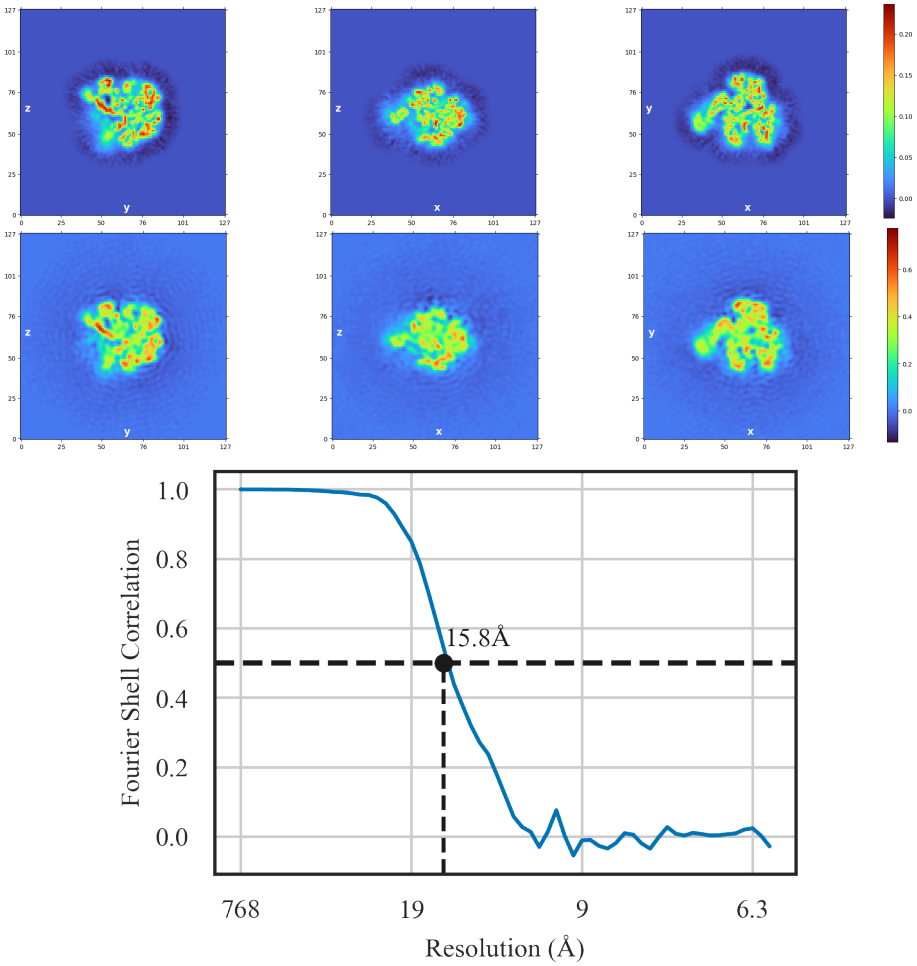

Figure 3.2: Fourier Shell Correlation between the reference map and the density map reconstructed using the WGAN approach with noise free projection images as input. The blue line represents the FSC values while the dashed line indicates the 0.5 cut-off for the resolution estimate. Here, the reolution is estimated to be 15.8 Å, which is close to the limit of 12 Å.

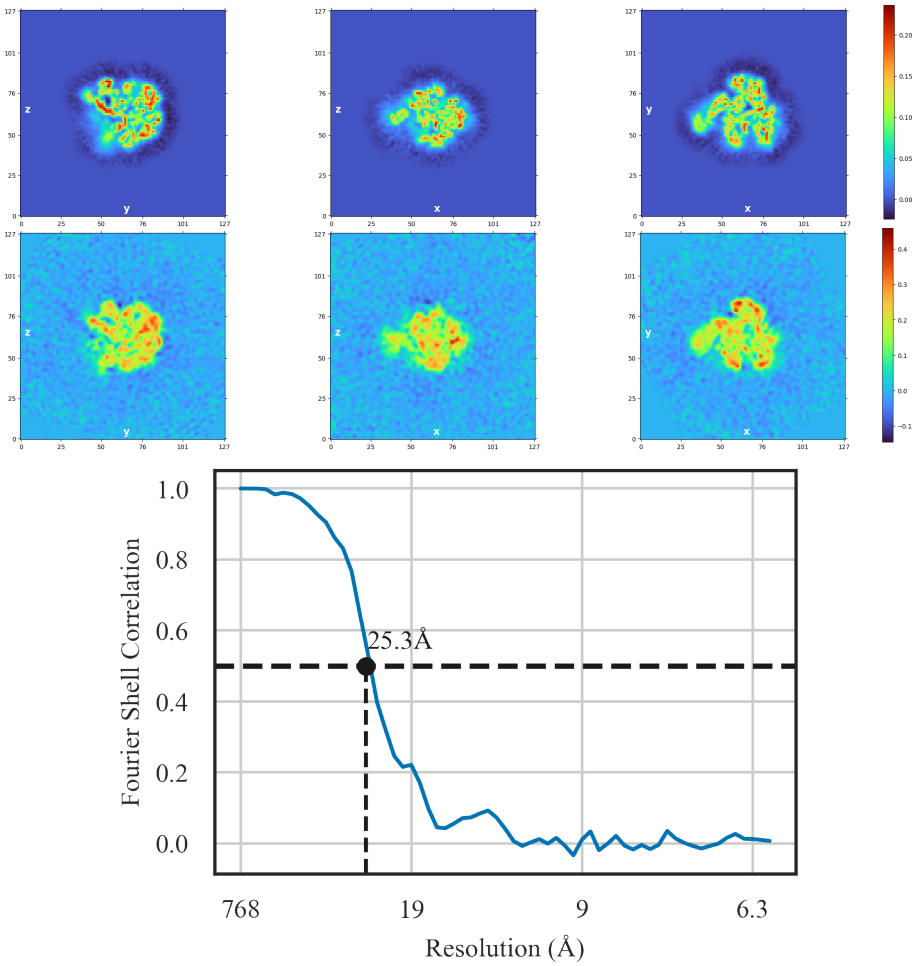

Figure 3.3: Fourier Shell Correlation between the reference map and the density map reconstructed using the WGAN approach with noisy projection images as input. The simulated projection images used as input for the WGAN reconstruction were distorted with colored noise (see main text). In this plot, the blue line represents the FSC values, while the dashed line indicates the 0.5 cut-off for the resolution estimate. Here, the resolution is estimated to be 25.3 Å.

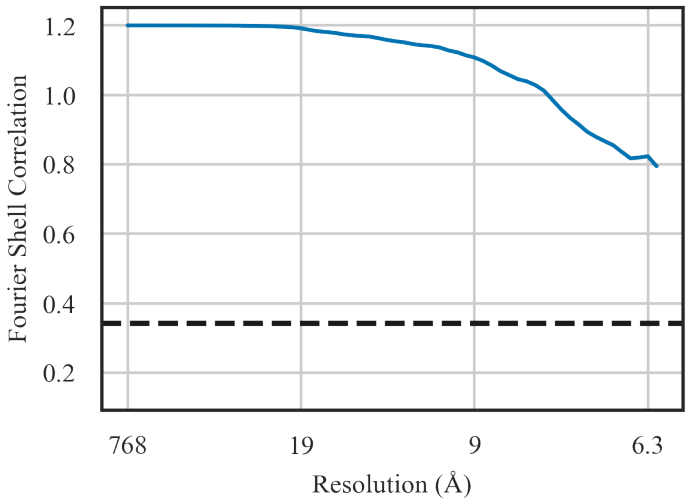

Figure 3.4: cryoSPARC reconstruction using the ab-initio reference map generated by the WGAN approach.
Unaltered result as reported by cryoSPARC. FSC values for different versions of the final reconstructed volume against the reference density map are drawn in different colors. The horizontal line indicates the 0.143 FSC cut off for the resolution estimation.

3.4 Discussion and Outlook

A priori reconstruction is a crucial step in cryo-EM method development. It is needed, as current high resolution reconstruction techniques require for an estimate of the volume to be reconstructed [Punjani et al., 2017; Scheres, 2012b]. Thereby, the volume obtained from ab-initio reconstruction is used to create artificial projections according to some forward model. These projections are then compared to experimental images to obtain estimates of possible orientations for each projection image. With this information, experimental images can then be back projected to update the current estimate of the volume to reconstruct. As such, these algorithms are sensitive to the correctness of the initial guess obtained from ab-initio refinement. Several different techniques have already been developed that perform ab-initio reconstruction [Zhong et al., 2021b; Reboul et al., 2018; Doerschuk and Johnson, 2000; Gomez-Blanco et al., 2019; Elmlund et al., 2008; Punjani et al., 2017; Levy et al., 2022]. CryoGAN is a GAN based approach that was devised for cryo-EM reconstruction [Gupta et al., 2021]. It showed good performance for a dataset of β-galactosidase, but needed to use the symmetry and a manual extraction of noise patches.

Based on this, I started to work on a Wasserstein GAN network with a similar task. The generator learns a volume from which it can generate projections using a forward model. The forward model does thereby consist of:

1. Projection of the learned volume. The pose can be randomly sampled from a uniform distribution or alternative from a learned pose distribution

2. Convolving with the CTF. The CTF parameters are thereby available from the experimental images.

3. Adding noise. The noise can thereby be generated by a small MLP network.

Images generated in this fashion are presented to the Critic, a convolutional neural network and the generator's parameters are updated to improve the output of the Critic on the generated images. The Critic is trained by presenting it with experimental images as well as artificially created once and maximizing the distance between the outputs on both. So far, I was able to implement the network and training algorithm and used a polymerase dataset to test the principal working of the approach. Thereby, a volume with a relatively large pixel size was learned. This was chosen for two reasons. Foremost, the task of ab-initio reconstruction does not require resolving high resolution features. In the refinement steps that usually follow ab-initio reconstruction, the provided reference maps are usually low pass filtered anyhow. Additionally, this choice decreased both the computational burden, but also the number of parameters to be learned. I have also tested lower pixel sizes, but saw a decrease in convergence speed as well as suboptimal performance (results not shown). Using the test dataset, a proof of concept could already be demonstrated. The generator was able to learn a low resolution representation of the true volume, which could also be used in a subsequent refinement step to reach resolution up to the Nyquist limit. At the same time, the generator was not fully able to get the full potential from the available information. In principle, the projections used for training the Critic contain information up to 12 Å, but the generator has only learned up to 15.8 Å in the noise free

case and 25.3 Å in the case with artificial noise. This might indicate that the current network design/training scheme is not ideal, yet.

At this stage, some parts could not be tested yet. The ability to learn arbitrary noise models as well as arbitrary pose distributions are two key parts that will need to be tested prior to being able to reconstruct real cryo-EM samples.

4 The Structure of a Dimeric Form of SARS-CoV-2 Polymerase

In April 2020, our group successfully solved the structure of the replicating RNA-dependent RNA polymerase of the SARS-CoV-2 [Hillen et al., 2020] which caused a global pandemic that is still ongoing at the time of writing this thesis. Based on this work, further research was able to reveal how Remdesivir interacts with the polymerase and can only cause an incomplete pausing [Kokic et al., 2021]. While working on these publications, there were hints that in some acquired datasets, dimeric structures might be present. My contribution then started by analyzing these already acquired datasets in more detail to single out the particle images that belong to dimeric conformations and subsequently perform 3D refinement to obtain the dimeric structure. With input from Dimitry Tegunov, I developed a method to identify from a set of particles those that might actually be part of a dimer. I wrote a software tool to perform this identification, re-processed the data set from [Kokic et al., 2021] to identify further dimeric particles and performed different analysis and 3D refinement to obtain the density of a dimeric form of the RdRp. Hauke Hillen and Goran Kokic then helped to fit a structure to said density and, together with Patrick Cramer and Jana Schmitzova, supported me in formulating a hypothetical model for the role of this dimeric structure in the production of subgenomic RNA. All this effort resulted in a publication:

> Jochheim, F. A., Tegunov, D., Hillen, H. S., Schmitzová, J., Kokic, G., Dienemann, C., and Cramer, P. (2021). The structure of a dimeric form of sars-cov-2 polymerase. *Communications biology*, 4(1):1–5

The following text in this chapter is taken from said publication. Formatting was edited to comply with the style of this thesis.

4.1 Abstract

The coronavirus SARS-CoV-2 uses an RNA-dependent RNA polymerase (RdRp) to replicate and transcribe its genome. Previous structures of the RdRp revealed a monomeric enzyme composed of the catalytic subunit nsp12, two copies of subunit nsp8, and one copy of subunit nsp7. Here we report an alternative, dimeric form of the enzyme and resolve its structure at 5.5 Å resolution. In this structure, the two RdRps contain only one copy of nsp8 each and dimerize via their nsp7 subunits to adopt an antiparallel arrangement. We speculate that the RdRp dimer facilitates template switching during production of sub-genomic RNAs.

4.2 Introduction

Replication and transcription of the RNA genome of the coronavirus SARS-CoV-2 rely on the viral RNA-dependent RNA polymerase (RdRp) [Hilgenfeld and Peiris, 2013; Snijder et al., 2016; Posthuma et al., 2017; Romano et al., 2020; Jiang et al., 2021]. Following the structure of the RdRp of SARS-CoV6, structures of the RdRp of SARS-CoV-2 were obtained in free form7 and as a complex with bound RNA template-product duplex [Yin et al., 2020; Wang et al., 2020; Hillen et al., 2020; Kokic et al., 2021]. These structures revealed a monomeric RdRp with a subunit stoichiometry of one copy of the catalytic subunit nsp12 [Ahn et al., 2012], two copies of the accessory subunit nsp8 [Imbert et al., 2006], and one copy of the accessory subunit nsp7 [Posthuma et al., 2017; Subissi et al., 2014]. Two studies additionally observed monomeric RdRp lacking one of the two nsp8 subunits and nsp7 [Kirchdoerfer and Ward, 2019; Wang et al., 2020]. Here we show that

the RdRp of SARS-CoV-2 can also adopt a dimeric form, with two RdRps arranged in an antiparallel fashion.

4.3 Results and Discussion

To detect possible higher-order RdRp assemblies in our cryo-EM data for RdRp-RNA complexes, we wrote a script to systematically search for dimeric particles (Methods). We calculated nearest-neighbor (NN) distances and relative orientations of neighboring RdRp enzymes. In one of our published data sets (structure 3 [Kokic et al., 2021]), we detected many particles that showed RdRps with a preferred NN distance of 80 Å and a relative orientation of 180° (Supplementary Fig. 1a, b), indicating the existence of a structurally defined RdRp dimer.

From a total of 78 787 dimeric particles, we selected 27 473 particles that showed a strong RNA signal during 2D classification. The selected particles led to a 3D reconstruction at an overall resolution of 6.1 Å (Supplementary Fig. 2b). We then fitted the obtained density with two RdRp-RNA complexes [Hillen et al., 2020], which revealed an antiparallel arrangement and RNA exiting to opposite sides of the dimeric particle. When we applied the two-fold symmetry during 3D reconstruction, the resolution increased to 5.5 Å (Supplementary Fig. 2c).

The reconstruction unambiguously showed that both RdRps lacked one copy of nsp8, and thus each enzyme was comprised of only one copy of each of the three subunits (Supplementary Fig. 3a). The lacking nsp8 is the one that interacts with nsp7 in monomeric RdRp and had been called nsp8b [Hillen et al., 2020]. The reconstruction showed poor density for the C-terminal helix of nsp7 (residues 63-73) and the sliding pole of the remaining nsp8 subunit (nsp8a, residues 6-110) (Supplementary Fig. 2b). These regions were mobile in both RdRp complexes and were removed from the model. Rigid body fitting of the known RdRp domains led to the final structure.

The structure of the antiparallel RdRp dimer showed that the two polymerases interact via their nsp7 subunits, with the nsp7 helices α1 and α3 (residues 2-20 and 44-62, respectively) contacting each other (Fig. 1). Formation of the nsp7-nsp7 dimer interface is only possible upon dissociation of nsp8b, which liberates the dimerization region of nsp7 (Supplementary Fig. 3c). To our knowledge, no similar nsp7-nsp7 interaction has been observed so far, as it differs from a previously described interaction in a nsp7-nsp8 hexadecamer structure [Zhai et al., 2005] and the nsp7-nsp7 interaction observed in a $(\text{nsp7-nsp8})_2$ heterotetramer [Zhang et al., 2021; Krichel et al., 2021].

Frequently occurring mutations in the SARS-CoV-2 genome are predicted to influence formation of the RdRp dimer. The nsp12 mutation P323L [Chand et al., 2020], coevolved with the globally dominating spike protein mutation D614G [Kannan et al., 2020], is often found in severely affected patients [Biswas and Mudi, 2020] and is predicted based on the structure to stabilize nsp12 association with nsp8a [Kannan et al., 2020; Reshamwala et al., 2021]. In contrast, the nsp7 mutation S25L is predicted to destabilize nsp7 binding to nsp8a [Reshamwala et al., 2021] and the nsp7 mutation L71F, which is associated with severe COVID-19 [Nagy et al., 2021], may destabilize binding of the nsp7 C-terminal region to nsp8b. We speculate that mutations in RdRp subunits can influence the relative stabilities of the RdRp monomer and dimer.

Neither the catalytic sites nor the RNA duplexes are involved in RdRp dimer formation. It is therefore likely that the two RdRp enzymes remain functional within the dimer structure. The two RdRp enzymes in the dimer may thus be simultaneously involved in RNA-dependent processes. Unfortunately, we could not test whether the RdRp dimer is functional because we were unable to purify it. In particular, we attempted to reconstitute the RdRp with a nsp12:nsp8:nsp7 stoichiometry of 1:1:1, but obtained preparations showed again an apparent stoichiometry of 1:2:1 that was observed in previous RdRp structures. Therefore, the functional relevance of the RdRp dimer reported here needs to be established.

We hypothesize that the RdRp dimer is involved in the production of sub-genomic RNA (sgRNA) [Sola et al., 2015; Kim et al., 2020; V'kovski et al., 2021]. In this intricate process, positive-strand genomic RNA (gRNA) is used as a template to synthesize a set of nested, negative-strand sgRNAs that are 5' and 3' coterminal with gRNA. The obtained sgRNAs are later used as templates to synthesize viral mRNAs. Production of sgRNAs involves a discontinuous step, a switch of the RdRp from an upstream to a downstream position on the gRNA template [Sawicki and Sawicki, 1998]. These positions contain transcription regulatory site (TRS) [Alonso et al., 2002; Pasternak et al., 2002; Zúñiga et al., 2004], but it is enigmatic how a single RdRp enzyme could 'jump' between these.

Our dimer structure suggests a model for sgRNA synthesis that extends a recent proposal [Chen et al., 2020; Malone et al., 2021] (Fig. 2). In the model, one RdRp of the dimer (RdRp 1) synthesizes sgRNA from the 3' end of the gRNA template until it reaches a TRS in the template body (TRS-B). Due to the lack of one nsp8 subunit, the dimeric RdRp is predicted to have lower processivity than monomeric RdRp [Hillen et al., 2020; Subissi et al., 2014] and this may facilitate TRS recognition. The viral helicase nsp13 could then cause backtracking of the RdRp [Chen et al., 2020; Malone et al., 2021]. Backtracking exposes the 3'-end of the nascent sgRNA, which is complementary to the TRS and may hybridize with another TRS located in the leader (TRS-L) at the 5'-end of the template. The resulting RNA duplex could then bind to the active center of the second RdRp (RdRp 2) to continue sgRNA synthesis.

In our model, it is not the RdRp that switches to a second RNA position, but instead the RNA switches to a second RdRp. After the switch, RdRp 1 may backtrack further, whereas RdRp 2 could move forward until it reaches the 5'-end of the template. These movements occur on the same template but in opposite directions and would be facilitated by the antiparallel arrangement of the polymerases. Superpositions show that only one copy of the template-strand engaged nsp13 [Yan et al., 2020] can be modeled on our dimer structure without clashes (Supplementary Fig. 3d). However, the interaction of this nsp13 copy to the monomeric RdRp is partially mediated by the nsp8 copy that is absent in the

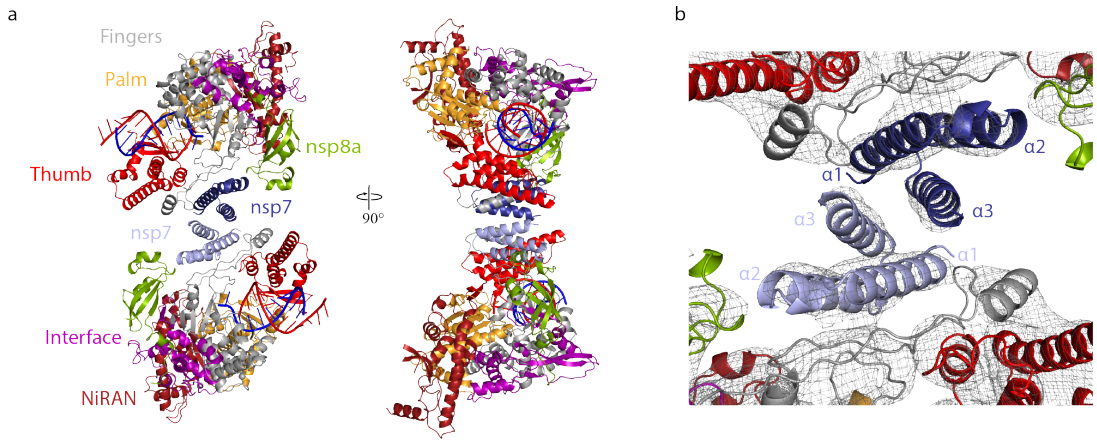

Figure 4.1: Structure of antiparallel RdRp dimer.
a Two views of a ribbon model of the antiparallel RdRp–RNA dimer. Color code for nsp7, nsp8, nsp12 domains (NiRAN, interface, fingers, palm, and thumb), RNA template (blue), and RNA product (red) is used throughout. Nsp7 subunits in the two RdRp monomers are colored slightly differently for the two monomers (dark and light blue, respectively). Views are related by a 90° rotation around the vertical axis.
b Close-up view of nsp7–nsp7 dimerization interface. View is as in the left structure of panel **a**. The final cryo-EM density is shown as a black mesh.

dimeric form. Thus, how backtracking in a dimeric complex may be facilitated remains unclear and it is possible that the second nsp13 copy that was previously not observed to be engaged with template RNA is involved in this process (Supplementary Fig. 3e).

Although the functional relevance of the RdRp dimer remains to be established, we note that RdRp dimerization and oligomerization has been reported for many other viruses including Influenza, Polio, Hepatitis C, Norovirus, and others [te Velthuis, 2014]. RdRp oligomerisation can be important for cooperative template binding [Högbom et al., 2009] and can be critical for the viral life cycle [Chang et al., 2015; Fan et al., 2019]. Future work should therefore concentrate on the preparation and functional analysis of the coronavirus RdRp dimer and possible additional higher-order structures of the enzyme.

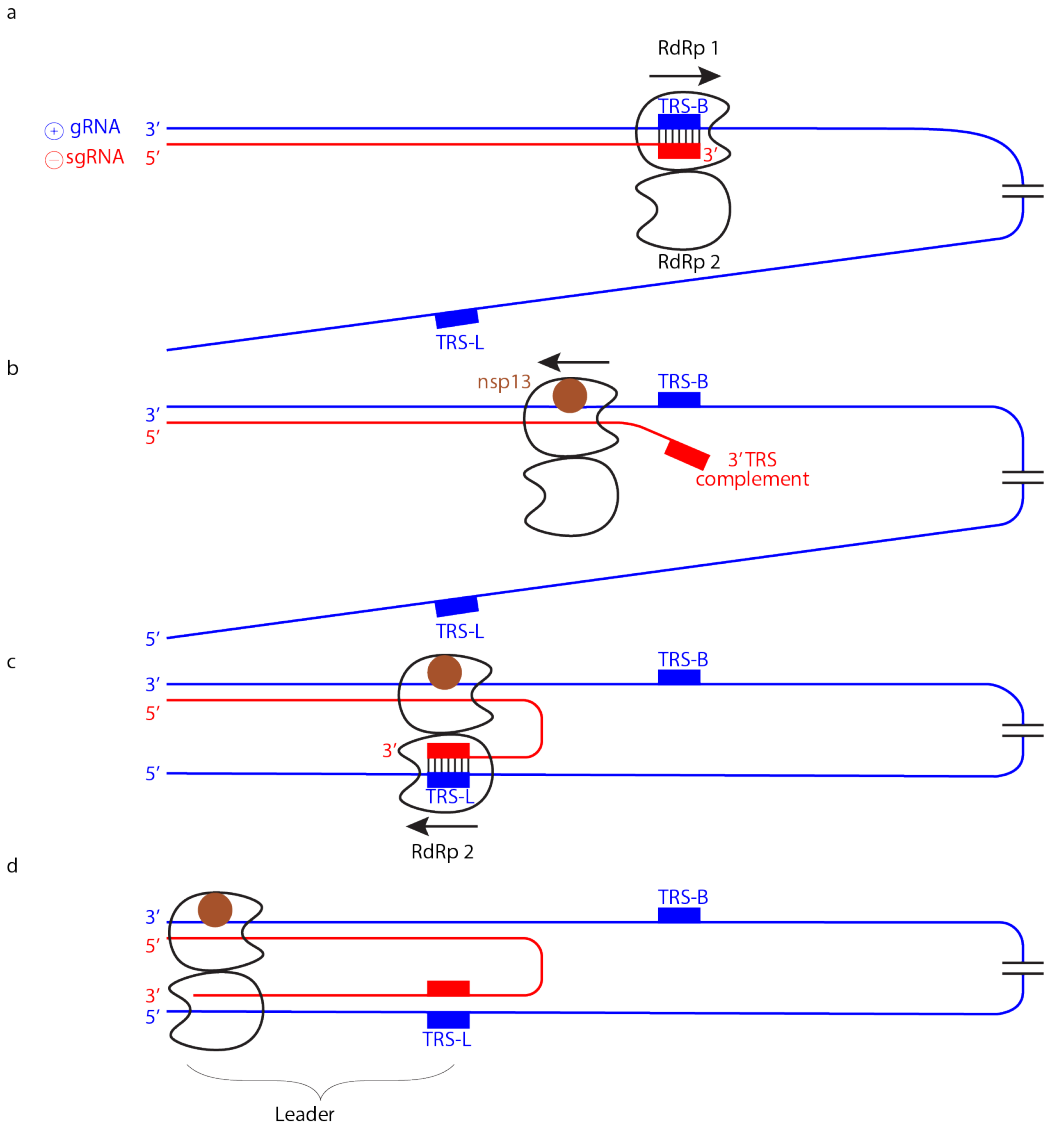

Figure 4.2: Hypothetical model of subgenomic RNA production for viral transcription.
a Genomic positive-strand ( ) RNA (gRNA) is used as a template to produce the 5' region of negative-strand ( ) sgRNA until TRS-B is reached by the RdRp monomer 1.
b Backtracking is mediated by nsp13 helicase and exposes the newly synthesized, complementary TRS sequence.
c The complementary sequence in sgRNA can pair with the downstream TRS-L in gRNA and is then loaded into RdRp monomer 2.
d RdRp 2 then completes sgRNA synthesis while RdRp 1 backtracks further.

4.4 Methods

4.4.1 Cryo-EM Sample Preparation

We reused the already processed data set 3 from our previous publication [Kokic et al., 2021]. Briefly, RNA sequence for the scaffold used was: rUrUrU rUrCrA rUrGrC rArCrU rGrCrG rUrArG rGrCrU rCrArU rArCrC rGrUrA rUrUrG rArGrA rCrCrU rUrUrU rGrGrU rCrUrC rArArU rArCrG rGrUrA and rUrGrA rGrCrC rUrArC rGrC- rA/rR -rGrUrG. RdRp-RNA complexes were formed by mixing 1.6 nmol of purified nsp12 with an equimolar amount of RNA scaffold and 4.8 nmol of each nsp8 and nsp7. The mixture was incubated for 10 min and afterwards applied to a Superdex 200 Increase 3.2/300 size exclusion chromatography column, which was equilibrated in complex buffer (20 mM Na-HEPES pH 7.4, 100 mM NaCl, 1 mM MgCl2, 1 mM TCEP) at 4 °C. Peak fractions corresponding to the RdRp-RNA complex were pooled and diluted to 2 mg/mL. Three microliters of the concentrated RdRp-RNA complex were mixed with 0.5 µL of octyl ß-d-glucopyranoside (0.003% final concentration) and applied to freshly glow discharged R 2/1 holey carbon grids (Quantifoil). The grids were blotted for 5 s using a Vitrobot MarkIV (Thermo Fischer Scientific) at 4 °C and 100% humidity and plunge frozen in liquid ethane.

4.4.2 Preprocessing of Cryo-EM Data

Data collection and preprocessing was the same as previously described [Kokic et al., 2021]. Briefly, data was collected using SerialEM [Mastronarde, 2005] on a 300 keV Titan Krios transmission electron microscope (Thermo Fischer Scientific) and a K3 direct electron detector (Gatan). Inelastically scattered electrons were filtered out prior to detection using a GIF quantum energy filter (Gatan) using a slit width of 20 eV. Images were acquired at a nominal magnification of 105 000x and a calibrated pixel size of 0.834 Å/pixel. Due to previously observed preferred orientation when imaging RdRp complexes [Hillen et al., 2020], data was collected using a 30° tilt to obtain more particle orientations. 7 043 raw

micrographs were acquired in total and preprocessed on-the-fly in Warp for automatic contrast transfer function (CTF) estimation, averaging, motion correction, and particle prediction and extraction. 2.2 million individual RdRp particles were predicted and exported by Warp and imported into cryoSPARC and subjected to a Hetero Refinement job using five 'Junk' classes and one class representing monomeric RdRp as described previously [Kokic et al., 2021]. The resulting particle set was used for a 3D homogeneous refinement to obtain refined poses and positions for each RdRp monomer.

4.4.3 Initial Detection of RdRp Dimers in Cryo-EM data

To analyze the statistical distribution of RdRp monomers in our cryo-EM data, we calculated the nearest-neighbor (NN) distances and the relative orientations for all neighboring RdRp complexes using the previously refined monomer poses. To account for the tilted data acquisition, we treated the influence on distances observed on the micrograph as a 30° rotation around the x-axis. We chose to express relative orientation through a single angle by calculating the angle of the rotation around the eigenvector of the rotation matrix. This showed that certain distances and relative orientations between two monomers were highly prevalent (Supplementary Fig. 1) and indicated the presence of dimeric particles where the two RdRps would adopt a specific distance and relative orientation with respect to each other. We then located such RdRp dimers in micrographs by identifying pairs of RdRps within a narrow range of NN distances and orientations. We observed an overrepresented RdRp distance of 80 Å and relative angle of 180°. Furthermore, we could observe that the overrepresented distance and relative orientation correlated with one another, indicating the occurrence of a defined RdRp dimer (Supplementary Fig. 1a). We initially obtained ∼31 000 dimeric particles using a distance < 90 Å and relative orientation larger than 166° as a filtering criterion.

4.4.4 Detection of Additional Dimeric Particles

Because the yield of dimeric particles depended on both halves of a dimer being first detected as monomer, we aimed to detect more RdRp monomers using two strategies. First, we carried out template-based picking and particle extraction in RELION [Zivanov et al., 2019] using the monomeric RdRp-RNA structure10 filtered to a resolution of 30 Å as a 3D reference to pick further monomers that might have been missed due to their proximity to other particles. Template-based picking of monomeric RdRp however did not introduce any model bias that could influence the dimer structure. Second, we used the previously established NN distance and relative orientation to predict for each RdRp monomer the position where its partner in a dimeric particle should be located on the micrograph. From our NN search we obtained the 3D offset that the second monomer should adopt in a dimeric particle (Supplementary Fig. 1b, c) and used this together with the monomer poses from our first refinement to predict micrograph coordinates for a potential second monomer in a dimeric arrangement. This approach does not bias the analysis towards a fixed relative orientation of NN monomers. Instead, we calculated the orientation of each monomer de novo after combining all picked monomers from Warp, template-based picking, and our prediction approach. With this procedure, patches extracted from predicted micrograph positions with pure noise will be assigned uncorrelated poses. Only NN monomers that actually form a dimer will have both the correct distance and relative orientation to be predicted as a dimer. After removing duplicates, we used the 3D classification approach described previously38 and conducted homogenous 3D refinement in cryoSPARC. Using the new monomer populations of 1.5 million particles and their refined poses, we could predict 78 787 dimer particles (i.e., $\sim 10\%$ of the monomers are predicted to be part of a dimer).

4.4.5 RdRp Dimer Reconstructions and Model Building

Final dimer coordinates were used to extract particles using a box size of 300 Å and a pixel size of 1.201 Å. 2D classification in cryoSPARC [Punjani et al., 2017] (Supplementary Fig. 2a) showed a dimeric structure that contained two RdRps in antiparallel orientation. We selected 2D classes that showed two RdRp-RNA monomers without a gap between them. After 2D classification, we used 27 473 particles for an ab initio refinement with three classes. For the subsequent 3D homogeneous refinement job in cryoSPARC, we chose a reconstruction from the ab initio refinement that had two clearly resolved RdRp monomers close to one another as a reference. A molecular model was obtained by placing two copies of RdRp-RNA complexes (PDB-7B3D, structure 3 from our previous publication) [Kokic et al., 2021], removal of mobile regions and fitting of domains as rigid bodies was done in UCSF Chimera [Pettersen et al., 2021].

4.4.6 Supplementary Figures

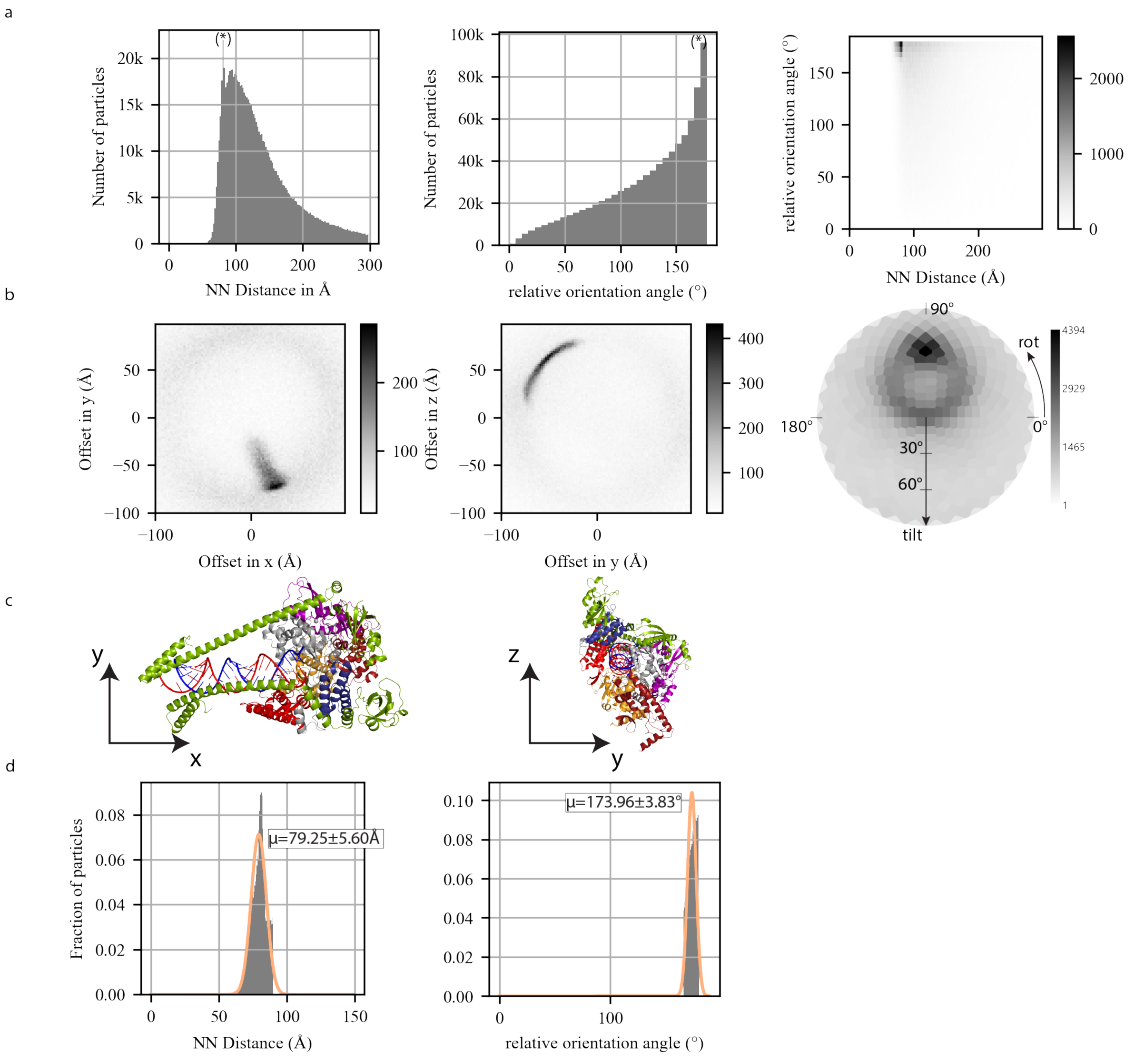

Figure 4.3: Detection of dimeric RdRp particles.
a, Detection of RdRp dimers in cryo-EM data. On the left, nearest neigbor (NN) distances for all $N = 902\,303$ Warp ([Tegunov and Cramer, 2019]) picked monomeric RdRp particles, with the most significant peak that deviates from the underlying random distribution indicated (*) at around 75 Å. In the middle plot, NN orientations are shown, expressed as a single angle of rotation around the eigenvector of the rotation matrix. A clear peak (*) is visible close to 180°. On the right, a conjoined plot of NN monomer distances and relative orientations, revealing that those nearest neighbors with a distance around 75 Å are also very likely to have relative orientation of $> 165°$. This indicates a correlation between this distance and a defined orientation within dimers.
b, Detailed analysis of our peak distances and relative orientations. In the first two panels, the elements of the vector connecting two NN monomers, expressed in the reference frame of one of the two monomers, are shown, further indicating a defined co-localization of two monomeric RdRps. In the third panel, a projection of relative orientations on a half sphere are shown. Colors indicate the number of particles in each bin.
c, Monomeric RdRp as previously reported [Hillen et al., 2020] (PDB code 6yyt) with axes indicated that were used for x,y,z offsets in **b**. Color code is the same as in Supplementary Figure 4.5.
d, NN distances and relative orientations of the $N = 31\,011$ monomers that are predicted to be part of dimers and an overlayed Gaussian fit.

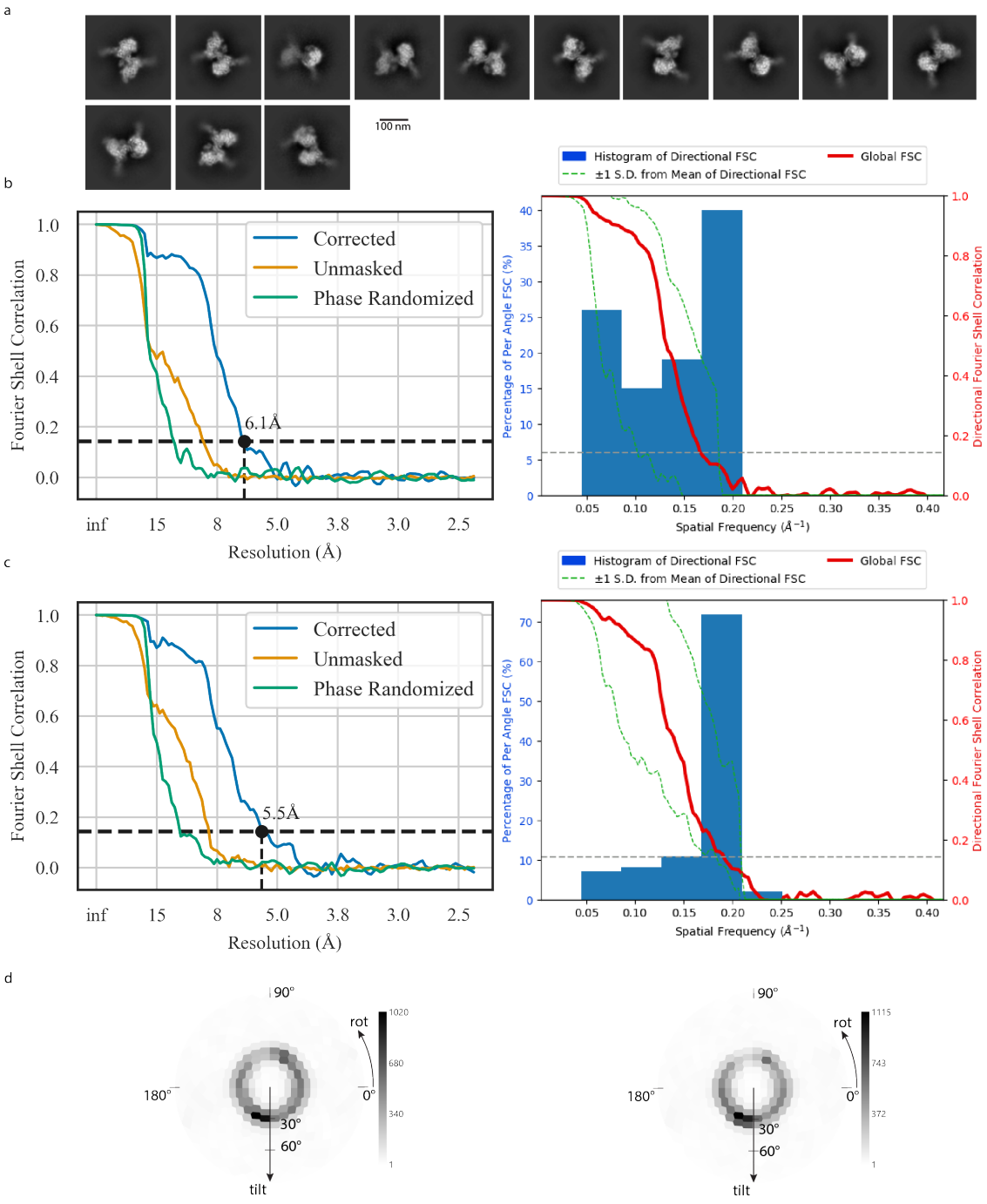

Figure 4.4: Cryo-EM processing of RdRp dimers.

Figure 4.4: Cryo-EM processing of RdRp dimers. (continued)
a, Selected 2D class averages from our predicted 78 787 dimeric particles. We chose classes with two clearly defined RdRp monomers and strong RNA signal as an indication for high alignment quality, representing 27 473 particles in total.
b, Fourier shell correlation (FSC) curve for the masked reconstruction of the antiparallel dimer, indicating an average resolution of 6.1 Å. FSC values were calculated through a RELION [Scheres, 2012b] postprocessing job to obtain values for the phase randomized curve as well. As input, the half maps and refinement mask from cryoSPARC's [Punjani et al., 2017] reconstruction were provided. The second panel contains a histogram of directional FSC values as calculated by the 3DFSC [Tan et al., 2017] server, indicating significant anisotropy with a sphericity value of 0.807.
c, With C2 symmetry applied, the average resolution of the reconstruction increased to 5.5 Å. The second panel contains a histogram of directional FSC values as calculated by the 3DFSC server, indicating significant anisotropy with a sphericity score of 0.789.
d, Orientation distribution from the reconstruction without (left) and with C2 symmetry alignment (right), showing a strong preferred orientation. Color bars indicate the number of particles in each bin.

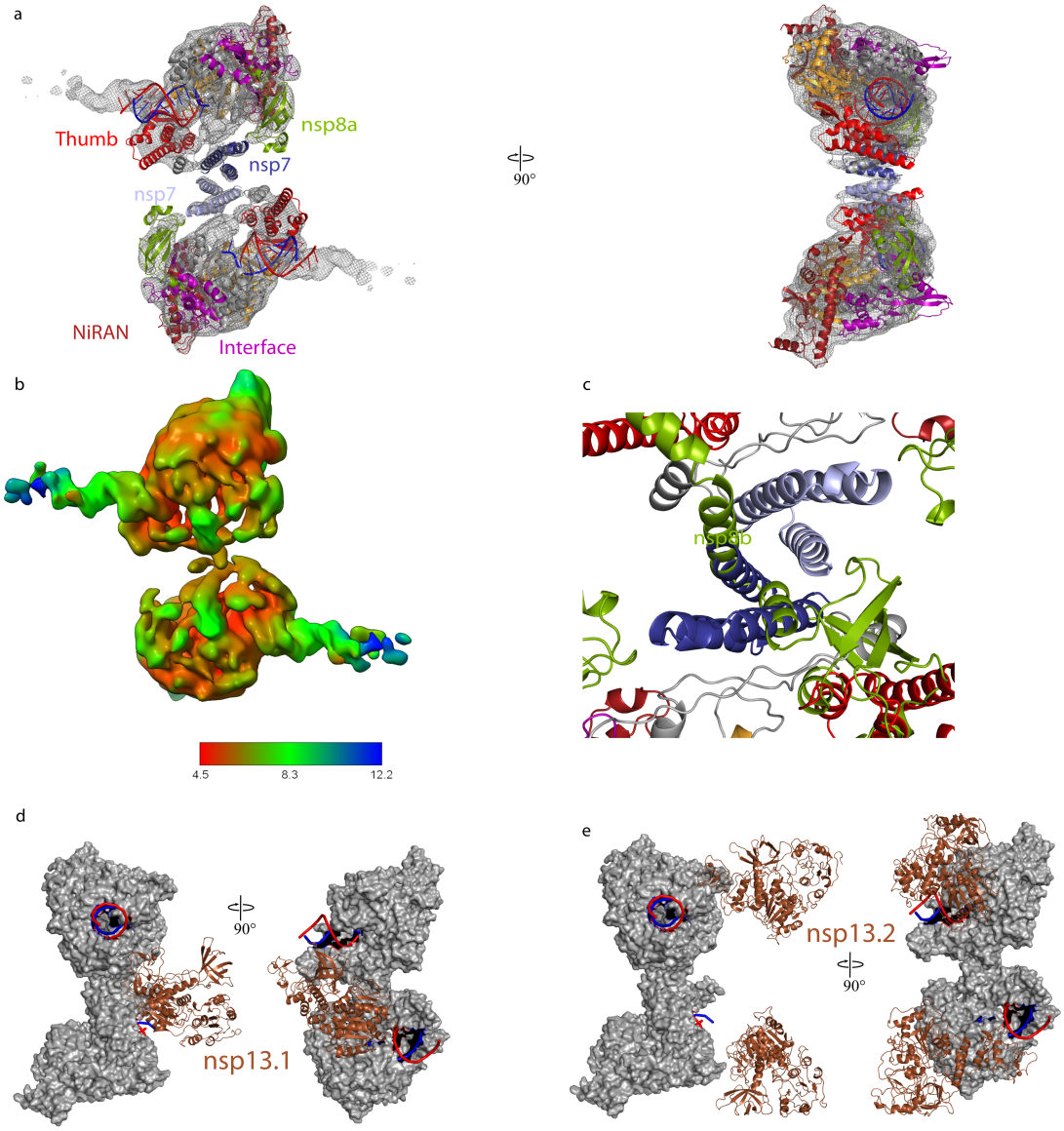

Figure 4.5: Quality of dimeric RdRp structure and structural comparisons.
a, Fit of structural model to cryo-EM density. Density was weak or lacking for the second turn of RNA and the associated nsp8a sliding pole. Views are related by a 90° rotation around the vertical axis.
b, Local resolution estimated using cryoSPARC. Resolution was weak for the second turn of RNA, which was also excluded from our model in **a**.
c, Modeling the second nsp8 copy (nsp8b) onto the top RdRp in the dimer structure resulted in clashes with the nsp7 subunit of the neighboring RdRp on the bottom.
d, Modeling of the nsp13 shows that helicase may accommodate on the RdRp dimeric structure. The nsp13 copy nsp13.1 (chain E in 7CXN [Yan et al., 2020]) can bind to one of the two monomers (binding to top monomer shown).
e, Modeling of the nsp13.2 shows that each of the two RdRp molecules may bind this copy of nsp13 (chain F in 7CXN [Yan et al., 2020]) without clashes.

5 Conclusion

Using cryo-electron microscopy (cryo-EM) to resolve structures of biological macromolecules is a rapidly evolving field, which now rivals the long-standing gold standard method of X-ray crystallography in structural biology.

To further drive this success, we require fast and automated methods for the *in silico* processing of the data. Especially the application of machine learning methods to different parts of the processing pipeline has grown in the last years. In this thesis, I introduced two methods for reconstruction. I explored a pseudo atom based approach to reconstruction in the presence of structural heterogeneity (chapter 2) and a Wasserstein GAN based ab-initio reconstruction algorithms (chapter 3). Additionally, I presented an approach to identify dimeric particles in a homogenous dataset of monomeric and dimeric SARS-CoV-2 RdRp particles (chapter 4). In the following, I discuss these three projects in the context of the current state of the fields, challenges, limitations and I also give my assessment for upcoming work related to each project.

5.1 Flexible Refinement

5.1.1 Current Status of Dealing with Structural Heterogeneity

Resolving structure(s) from datasets with structural variation has been studied for quite some time already. Thereby, this heterogeneity is often of relevance to study the activity

of the protein complex under investigation. Studying different states and the transitions between them can give insight about the function of a protein complex [Wittenborn and Marletta, 2021; Hillen et al., 2020; Kokic et al., 2021]. Furthermore, due to flexibility, it can be impossible to resolve a structure globally in high resolution [Schilbach et al., 2017]. Traditionally, classification and masked refinement were options to try and resolve structural heterogeneity through different approaches. Classification aims to divide the dataset in such a way that the different classes show little enough structural heterogeneity to allow for subsequent refinement. Masked refinement on the other hand aims to reduce the refinement task to regions of the structure that are in itself rigid enough to again allow for high resolution reconstruction. With these approaches, it is e.g. already possible to solve different states of an actively transcribing polymerase [Kokic et al., 2021; Hillen et al., 2020] or to solve the mediator complex using a combined approach of masked refinement and normal mode analysis [Schilbach et al., 2017]. Recently, the deep learning based tools CryoDRGN [Zhang et al., 2021] and 3DFlex [Punjani and Fleet, 2021] were developed that aim to solve structural heterogeneity that could not be resolved using previous methods. E.g. continuous motion cannot be classified and while it can be masked using masked refinement, this does not yield a global map, which is desirable. These tools could already be applied successfully, e.g. to reveal additional functional states of the chromatin remodeler ALC1 [Bacic et al., 2021], but proved insufficient in the processing of actomyosin-V complex [Pospich et al., 2021]. This highlights, that the development in this direction is not complete yet, and different computational methods should still be investigated to increase the number of available methods. Both of these new methods employ machine learning to tackle the problem of structural heterogeneity. The main advantage is that machine learning algorithms can learn patterns and rules in the data that may be hidden for the human observer. Machine learning approaches, e.g. for natural image classification, have already proven to be able to surpass human level accuracy [Hestness et al., 2019]. CryoDRGN and 3DFlex are just examples of transferring the predictive power of these approaches to cryo-EM image processing tasks, and one can expect that many algorithmic improvements in this field will be from applying deep learning techniques. As such, it is

important that the approach investigated in this chapter is now fully compatible with a deep learning framework.

5.1.2 Further Development

5.1.2.1 Combination of Refinement and Flexibility Estimation

To be able to move the pseudo atom clouds for each individual projection and use that information during the reconstruction, I have assumed that a rough 3D density for each projection image is already known. Dimitry Tegunov had previously worked on this problem and explored it using 3D PCA analysis [Schilbach et al., 2017] as well as an autoencoder to generate such estimates. Using an autoencoder to generate the density estimations and then using my framework to estimate pseudo atom coordinates however has an unnecessary intermediate step. One would train the network to output Cartesian densities, only to represent them as pseudo atom clouds in the subsequent step. With the reconstruction pipeline completely integrated into the deep learning framework PyTorch, however, an end-to-end approach can be developed. Currently, a combined network called TildeNet is being developed. First, it estimates pseudo atom coordinates for each projection image using an autoencoder like architecture and then utilizes my framework to perform a reconstruction using the same projection images. The agreement of two halfmaps being reconstructed separately, as well as the difference between pseudo atom generated projections and experimental images, can thereby be used as target functions for the training. This approach has similarity to cryoSPARC's flexible refinement algorithm, which was published shortly after we had devised this project [Punjani and Fleet, 2021]. However, cryoSPARC's algorithm does not employ pseudo atoms, but a deformation vector field on the Cartesian representation of the volume. As TildeNet is still under active development - mostly by Dimitry Tegunov -, no results can be shown here. This application however shows the usefulness of having implemented my operators in PyTorch, as it enables the training of larger networks that only utilize the pseudo atoms in some intermediate steps.

Using this combined approach, we hope that we will be able to reach high resolution reconstructions on datasets on which reconstruction was not possible using rigid-body approaches.

5.1.2.2 Further Benchmarking

Thus far, only a simulated dataset with two states was used to test the new reconstruction approach. Eventually moving to real datasets is, however, needed as a comparison to other published methods. Thereby, the Rag1-Rag2 dataset, [Ru et al., 2015] used in the evaluation of CryoDRGN [Zhong et al., 2021a], and the U4/U6.U5 tri-snRNP complex, used in the evaluation of 3DFlex [Punjani and Fleet, 2021], would be viable targets to quantify the performance of the approach presented here in comparison to other methods. As an intermediate step, a dataset with known ground truth could be used to judge the progress of the current development. Any arbitrary dataset from which densities were reconstructed using 3D classification could be used for this purpose. Using such a dataset as input, one could analyze if the flexible refinement algorithm is able to reach the same 3D structures obtained from the classification approach.

5.1.3 Conclusion

Prior to the start of this project, no universal tool was available to tackle structural heterogeneity when presented as flexibility that could not be classified or masked in a satisfactory way. By now, at least two methods based on deep learning have been published that aim to solve the same problem that our approach was supposed to address. To judge if working on this project further is viable, one would now carefully identify problems with the existing approaches and then test if these could be overcome with the approach developed here. There are interesting aspects to this project, however. The fact that the flexibility is parametrized by pseudo atoms already couples the modeled flexibility with the movement of physical objects, and might therefore ensure that any solutions produced

by the algorithm actually conform with what would be physically possible. In contrast to approaches like classification and masked refinement, this approach should thereby allow for a global reconstruction using all particle images.

5.1.4 Code Availability

The code of this project has been uploaded to GitHub and is freely available under https://github.com/cramerlab/FlexibleRefinement.

5.2 WGAN

5.2.1 Comparison to Other ab-initio Algorithms

In chapter 3 I explored the use of a Wasserstein GAN for the ab-initio reconstruction. Gold standard approaches as implemented in cryoSPARC [Punjani et al., 2017] and RELION [Scheres, 2012b] are always directly data driven, i.e. information from particle images is directly used to update and estimate an initial model as good as possible. The learning of the volume in the WGAN based approach only indirectly uses the information in the experimental particle images. The critic network is used to learn a representation of the real experimental data, and its learned information is then used to update the generator volume. This approach has previously been tested in the development of CryoGAN [Gupta et al., 2021, 2020] which also used a Wasserstein GAN to reconstruct 3D Coulomb potentials. CryoGAN, however, does not feature a learned distribution of the noise in the image and instead relies on extracting noise from free regions on the micrograph. This is not ideal, as there is usually no guarantee that such free regions exist. Furthermore, this approach was not tested on non-symmetrical real experimental data and probably suffers from similar issues when presented with real experimental images, as could be seen during the research on chapter 3.

5.2.2 Further Development

5.2.2.1 Regularization of Learned volume

It would be beneficial to determine why the generator was not able to already generate a volume up to the Nyquist limit; at least in the noise free case. As could be shown, this does not hinder the usability of the reconstructed volume in downstream processing. But understanding the source of this issue would give further insight into the challenges of using WGAN approaches for reconstruction purposes. As noted in [Arjovsky and Bottou, 2017], GANs are difficult to train, as the distribution the generator needs to learn is often a low-dimensional manifold. In this case, the generator has a space of 64^3 float values that it can arbitrarily change. However, it is obvious that real maps of protein structures cannot be arbitrary 3D volumes. It is difficult, if not impossible, though to write down a parametrization of protein maps that would allow for a reduction of the learnable parameters of the generator while regularizing it to be only able to learn correct protein representations. There are, however, many available examples of available density maps in public archives such as the EMDB [Lawson et al., 2016]. With this data, one can train an autoencoder network to learn a mapping from a latent space to a 3D volume [Bank et al., 2020]. In short, such a network would take a 3D volume, then pass it through a neural network called an encoder that outputs a vector in the latent space with a user defined number of variables. In a second step, this latent vector is used as input for a decoder network that would map it back to a 3D volume. The training objective would be that the volumes on both ends should be identical and that the distribution in the latent space should follow a user defined one, e.g. Gaussian. The decoder of this network could then be used in the generator of the reconstruction WGAN. Instead of learning a full volume, the generator would learn a vector in latent space and pass it through the pre-trained decoder to obtain the volume it needs to project. Training would then only update the latent vector, and needs to be regularized such that the latent vector still complies with a Gaussian distribution.

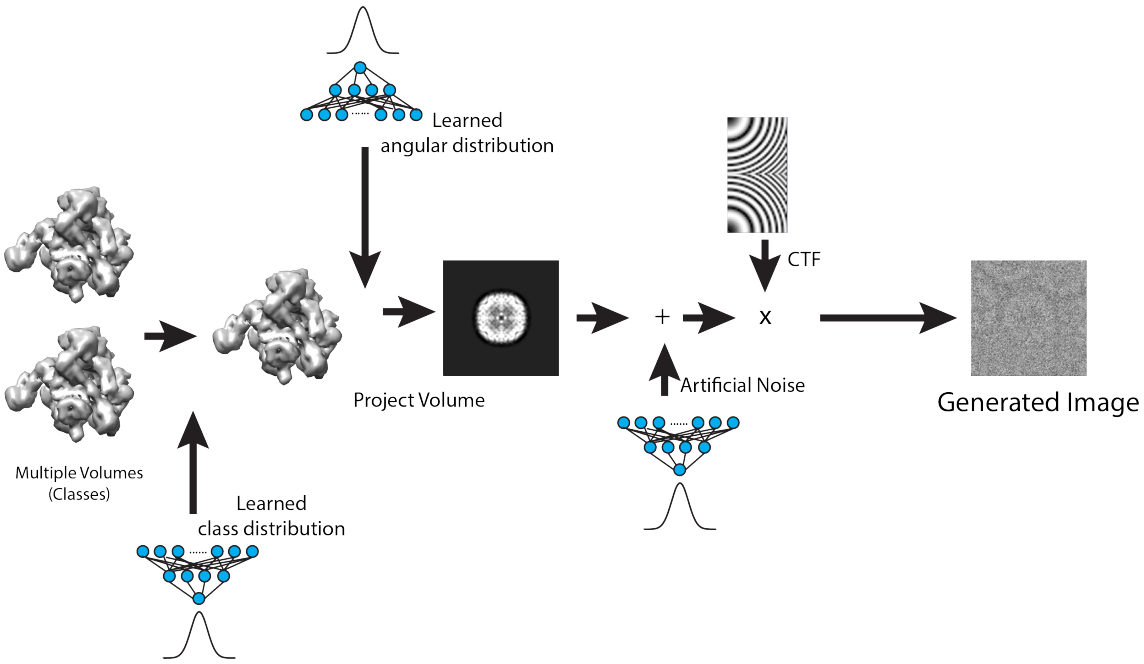

Figure 5.1: Multi-class reconstruction with a generative adversarial network. The approach introduced in chapter 3 can be expanded to be able to reconstruct multiple distinct densities (classes). These could originate from different occupancies of subunits, different states in an active complex, or from contamination. Instead of always projecting the same volume, the generator could randomly pick one of multiple volumes. In a real data set not all classes would be present in equal amounts. To be able to cope with an uneven class distribution in the source data, a value sampled from a gaussian distribution can be transformed by a small MLP network.

5.2.2.2 Additions to Fit Experimental Data

In real cryo-EM experimental data, poses are not uniformly distributed, as was assumed in the simulated datasets used in this project. Instead, there is usually some kind of bias toward a certain pose, as particles can orient relative to the air-water interface on the grid. To compensate and incorporate this into the reconstruction process would be crucial to correctly reconstruct from datasets with strong preferred orientation. Learning the correct distribution of viewing directions can be added to the generator by introducing a small MLP network that learns to map a random Gaussian input variable to a viewing direction sampled from the real distribution present in the data. A similar approach could be employed to incorporate multiple classes of volumes in the reconstruction task. Instead of a single volume, the generator could learn an ensemble of volumes, of which one is randomly picked when creating a single projection. Thereby, the distribution of classes

could again be learned by an MLP network. In total, the generator network could be adapted as seen in Figure 5.1.

5.2.3 Conclusion

The research presented here, in combination with the publications already available, [Gupta et al., 2020, 2021] hint at the potential that could be reached using a Wasserstein GAN for ab-initio reconstruction. I expect, that it is difficult for this approach to accurately reconstruct 3D maps with a high resolution *en par* with existing 3D reconstruction algorithms that require an ab-initio model. It does, however, provide an interesting approach to ab-initio reconstruction, as no prior knowledge is required. Intermediate steps, such as designing an autoencoder to parametrize possible volumes, do thereby represent interesting research directions in themselves. It will be interesting to see if such networks are possible to train, as this would further strengthen the advantages seen in the application of deep learning techniques to open questions in science. I do, however, expect that this tool will only be complementary to already existing tools and not supersede them. Due to the difficulty in training and converging a Wasserstein GAN to the optimal solution, there are probably failure cases which are difficult to predict and carefully confirming any findings done using a deep learning tool with traditional methods should always be performed.

5.2.4 Code Availability

The code of this project has been uploaded to GitHub and is freely available under https://github.com/cramerlab/WGANReconstruction.

5.3 RdRp

In the third project, we developed an approach to identify dimeric particles in a set of both monomeric and dimeric particles. We applied this approach to filter pre-existing SARS-CoV-2 RdRp datasets. With this approach, we identified and reconstructed a dimeric form of the RdRp complex and hypothesized a potential model of how it might facilitate sgRNA production.

5.3.1 Current Results on Backtracking

Central to the hypothetical model of sgRNA production is the idea that backtracking occurs when RdRp reaches the TRS site. This would then allow for the second RdRp in the dimeric complex to load the exposed TRS complement paired with the downstream TRS-L site. When working on this project, first structural insights into the RdRp-nsp13$_2$ complex had already been published [Chen et al., 2020; Yan et al., 2020, 2021]. These also suggested that nsp13 activity would be crucial for backtracking. It was however poorly understood how backtracking in activity of nsp13 is activated and controlled to allow rapid duplication of the RNA genome by RdRp in cases when backtracking is not needed. A detailed analysis of conformational changes undergone by nsp13 have revealed that backtracking can be turned on and off by the RdRp-nsp13 complex [Chen et al., 2022]. Thereby, backtracking is performed by nsp13.1, which can be bound to one of the RdRp copies in our dimeric structure, see Figure 4.5. Current knowledge of how backtracking is performed is therefor compatible with our dimeric structure, and the correctness of our model is still an open question.

5.3.2 Current Results on sgRNA production

Sub-genomic RNA production in coronaviruses, and also in SARS-CoV-2 is the result of discontinuous transcription of the full length RNA genome. The current model for

this mechanism is that two transcription regulatory sites (TRS) are paired, which then creates a location at which the RNA-dependent RNA polymerase can undergo template switching. When reaching the TRS-B site closer to the 3' end of the genome, the RdRp is then thought to template switch to the TRS-L site at the 5' end of the genome. Thereby, to my knowledge, no further insight into the exact mechanism behind the template switching is known.

Characterization of sgRNA production is thereby carried out by analyzing sequencing data from infected patients or cells [Chen et al., 2022; Parker et al., 2021; Zeng et al., 2022; Lyu et al., 2022; Parker et al., 2022]. These studies have shown that sgRNA production may play a role in gene regulation and host adaption [Lyu et al., 2022], can be used as a marker for infection severity [Chen et al., 2022] as its production shows a kinetic during infection that is yet poorly understood [Parker et al., 2021], and might provide a way to monitor upcoming variants [Parker et al., 2022]. As such, further research into the exact mechanism by which template switching during sgRNA production is performed as well as how it is regulated is an interesting field of research, as it can yield important understanding of the biology of coronaviruses, but also has clinical relevance as an additional marker for infection severity. Determining if the dimeric structure presented here plays a role in this process may thereby assist this process, and further experiments towards elucidating its role should be performed.

5.3.3 Further Experiments

5.3.3.1 Purification of RdRp Dimer

A key disadvantage in studying the presented dimeric structure in more detail so far is that it has not been purified thus far. Understanding the biochemical conditions under which the dimer forms preferably and purifying it is however needed to use it in any hypothetical essays to quantify sgRNA production. As little is known in this direction yet, experiments in this direction will need to be exploratory. The dimeric structure was reconstructed

from particles that were part of a sample of monomeric RdRp complexes. The workflow to identify dimeric particles allows to also quantify the fraction of particles that are part of a dimer. Therefore, one could start with the sample preparation parameters used to generate the sample used in this study and then alter these using cryo-EM and the dimer detection workflow to quantify the fraction of dimeric particles. This would allow identifying changes that would increase the chance to form the dimeric structure. This can then be coupled with size-exclusion chromatography to try to filter out potential dimeric structures at different steps of the sample preparation. Notably, however, size exclusion was already used to specifically filter out monomeric RdRp articles, indicating that dimers must have formed in a subsequent step of the sample preparation.

5.3.3.2 Relevance of RdRp Dimer for sgRNA production

The open question from the presented publication is the relevance of the presented dimeric structure for sgRNA production. Studying this in more detail is complicated by the fact that - to my knowledge - no essay exists to quantify sgRNA production in vitro. Knowledge of sgRNA production and kinetics stems from analyzing sequencing data from infected cells [Chen et al., 2022; Parker et al., 2021; Zeng et al., 2022; Lyu et al., 2022]. Designing an experiment that analyzes the relevance of the dimeric structure to form would also require more understanding of the conditions under which the dimer forms preferably. If, hypothetically, we had an essay to test sgRNA production in vitro, we would need mutations to nsp7 that would hinder the dimeric structure from forming while not hampering the processivity of monomeric RdRp in any way. If we have a protocol to produce the dimeric structure in a reproducible way, one could test if certain mutations to the α_1 and α_3 helix hamper the dimer formation. At the same time, one would check in an extension essay if monomeric RdRp with the same mutation is still active.

5.3.3.3 Application to Other Multi-Copy Structures

The workflow developed to identify the RdRp dimeric structures is not limited to only identify RdRp dimers. The core idea was to identify NN particles that show an over-represented relative distance and relative orientation. This approach should be viable to identify multimers of different complexes as well. Thereby, one is not only limited to dimers. In principle, one can also apply this approach to identify any kind of complex that is made up of several parts. The key requirement thereby is that the individual parts can be picked individually with high accuracy. The advantage using the pipeline developed here is thereby that it is not needed to specifically pick the multimeric structures. This can be difficult as most particle pickers will pick both pure monomeric structures, but also the multimers. Subsequent steps like 2D classification can not always be used to sort out the multimers specifically, as projections of multimers also show high correlation to projections of the monomers. I therefore suggest using the pipeline developed for the RdRp dimer for the detection of multimeric structures in future datasets. I was not able to find similar tools suited for this task.

5.3.4 Conclusion

The biological relevance of the dimeric structure that was discovered, is yet to be analyzed. For this analysis, however, the right toolkit is not available yet. One will need to solve the intermediate problems such as purifying the dimer and designing an essay for the analysis of sgRNA production first, before being able to analyze the function of this dimer in more detail. It is however crucial to understand the many aspects of SARS-CoV-2 replication in as great a detail as possible, as this will enable us to design ways to combat the ongoing pandemic through the design of custom drugs.

5.3.5 Code Availability

The code of this project has been uploaded to GitHub and is freely available under https://github.com/cramerlab/RdRp-DimerDetection.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Ahn, D.-G., Choi, J.-K., Taylor, D. R., and Oh, J.-W. (2012). Biochemical characterization of a recombinant SARS coronavirus nsp12 RNA-dependent RNA polymerase capable of copying viral RNA templates. *Arch. Virol.*, 157(11):2095–2104.

Alonso, S., Izeta, A., Sola, I., and Enjuanes, L. (2002). Transcription Regulatory Sequences and mRNA Expression Levels in the Coronavirus Transmissible Gastroenteritis Virus. *J. Virol.*, 76(3):1293–1308.

Arjovsky, M. and Bottou, L. (2017). Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

Bacic, L., Gaullier, G., Sabantsev, A., Lehmann, L. C., Brackmann, K., Dimakou, D., Halic, M., Hewitt, G., Boulton, S. J., and Deindl, S. (2021). Structure and dynamics of the chromatin remodeler alc1 bound to a parylated nucleosome. *Elife*, 10:e71420.

Bai, X.-c., Rajendra, E., Yang, G., Shi, Y., and Scheres, S. H. (2015). Sampling the conformational space of the catalytic subunit of human γ-secretase. *elife*, 4.

Bank, D., Koenigstein, N., and Giryes, R. (2020). Autoencoders. *arXiv preprint arXiv:2003.05991*.

Biswas, S. K. and Mudi, S. R. (2020). Spike protein D614G and RdRp P323L: the SARS-CoV-2 mutations associated with severity of COVID-19. *Genomics Inform.*, 18(4):e44.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer.

Brilot, A. F., Chen, J. Z., Cheng, A., Pan, J., Harrison, S. C., Potter, C. S., Carragher, B., Henderson, R., and Grigorieff, N. (2012). Beam-induced motion of vitrified specimen on holey carbon film. *Journal of structural biology*, 177(3):630–637.

Campbell, M. G., Veesler, D., Cheng, A., Potter, C. S., and Carragher, B. (2015). 2.8 å resolution reconstruction of the thermoplasma acidophilum 20s proteasome using cryo-electron microscopy. *Elife*, 4:e06380.

Casañal, A., Kumar, A., Hill, C. H., Easter, A. D., Emsley, P., Degliesposti, G., Gordiyenko, Y., Santhanam, B., Wolf, J., Wiederhold, K., et al. (2017). Architecture of eukaryotic mrna 3 -end processing machinery. *Science*, 358(6366):1056–1059.

Chand, G. B., Banerjee, A., and Azad, G. K. (2020). Identification of novel mutations in RNA-dependent RNA polymerases of SARS-CoV-2 and their implications on its protein structure. *PeerJ*, 8(7):e9492.

Chang, S., Sun, D., Liang, H., Wang, J., Li, J., Guo, L., Wang, X., Guan, C., Boruah, B. M., Yuan, L., Feng, F., Yang, M., Wang, L., Wang, Y., Wojdyla, J., Li, L., Wang, J., Wang, M., Cheng, G., Wang, H.-W. W., and Liu, Y. (2015). Cryo-EM Structure of Influenza Virus RNA Polymerase Complex at 4.3ÅResolution. *Mol. Cell*, 57(5):925–935.

Chen, J., Malone, B., Llewellyn, E., Grasso, M., Shelton, P. M., Olinares, P. D. B., Maruthi, K., Eng, E. T., Vatandaslar, H., Chait, B. T., Kapoor, T. M., Darst, S. A., and Campbell, E. A. (2020). Structural Basis for Helicase-Polymerase Coupling in the SARS-CoV-2 Replication-Transcription Complex. *Cell*, 182(6):1560–1573.e13.

Chen, J. Z. and Grigorieff, N. (2007). Signature: a single-particle selection system for molecular electron microscopy. *Journal of structural biology*, 157(1):168–173.

Chen, S., McMullan, G., Faruqi, A. R., Murshudov, G. N., Short, J. M., Scheres, S. H., and Henderson, R. (2013). High-resolution noise substitution to measure overfitting and validate resolution in 3d structure determination by single particle electron cryomicroscopy. *Ultramicroscopy*, 135:24–35.

Chen, Z., Ng, R. W. Y., Lui, G., Ling, L., Chow, C., Yeung, A. C. M., Boon, S. S., Wang, M. H., Chan, K. C. C., Chan, R. W. Y., et al. (2022). Profiling of sars-cov-2 subgenomic rnas in clinical specimens. *Microbiology Spectrum*, 10(2):e00182–22.

Cheng, Y. (2018). Membrane protein structural biology in the era of single particle cryo-em. *Current opinion in structural biology*, 52:58–63.

Doerschuk, P. C. and Johnson, J. E. (2000). Ab initio reconstruction and experimental design for cryo electron microscopy. *IEEE Transactions on Information Theory*, 46(5):1714–1729.

Downing, K. H. and Glaeser, R. M. (2008). Restoration of weak phase-contrast images recorded with a high degree of defocus: the "twin image" problem associated with ctf correction. *Ultramicroscopy*, 108(9):921–928.

Dubochet, J., Lepault, J., Freeman, R., Berriman, J. A., and Homo, J.-C. (1982). Electron microscopy of frozen water and aqueous solutions. *J. Microsc.*, 128(3):219–237.

Elmlund, H., Lundqvist, J., Al-Karadaghi, S., Hansson, M., Hebert, H., and Lindahl, M. (2008). A new cryo-em single-particle ab initio reconstruction method visualizes secondary structure elements in an atp-fueled aaa+ motor. *Journal of molecular biology*, 375(4):934–947.

Erickson, H. and Klug, A. (1971). Measurement and compensation of defocusing and aberrations by fourier processing of electron micrographs. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, 261(837):105–118.

Erickson, H. P. and Klug, A. (1970). The fourier transform of an electron micrograph: Effects of defocussing and aberrations, and implications for the use of underfocus contrast enhancement. *Berichte der Bunsengesellschaft für physikalische Chemie*, 74(11):1129–1137.

Fan, H., Walker, A. P., Carrique, L., Keown, J. R., Serna Martin, I., Karia, D., Sharps, J., Hengrung, N., Pardon, E., Steyaert, J., Grimes, J. M., and Fodor, E. (2019). Structures of influenza A virus RNA polymerase offer insight into viral genome replication. *Nature*, 573(7773):287–290.

Frank, J. (1973). The envelope of electron microscopic transfer functions for partially coherent illumination. *Optik*, 38:519–536.

Frank, J. (1975). Averaging of low exposure electron micrographs of non-periodic objects. *Ultramicroscopy*, 1(2):159–162.

Gao, H., Valle, M., Ehrenberg, M., and Frank, J. (2004). Dynamics of ef-g interaction with the ribosome explored by classification of a heterogeneous cryo-em dataset. *Journal of structural biology*, 147(3):283–290.

Gomez-Blanco, J., Kaur, S., Ortega, J., and Vargas, J. (2019). A robust approach to ab initio cryo-electron microscopy initial volume determination. *Journal of Structural Biology*, 208(3):107397.

Gonog, L. and Zhou, Y. (2019). A review: generative adversarial networks. In *2019 14th IEEE conference on industrial electronics and applications (ICIEA)*, pages 505–510. IEEE.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gorski, K. M., Hivon, E., Banday, A. J., Wandelt, B. D., Hansen, F. K., Reinecke, M., and Bartelmann, M. (2005). Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759.

Grant, T. and Grigorieff, N. (2015). Measuring the optimal exposure for single particle cryo-em using a 2.6 å reconstruction of rotavirus vp6. *elife*, 4:e06980.

Grant, T., Rohou, A., and Grigorieff, N. (2018). cistem, user-friendly software for single-particle image processing. *elife*, 7:e35383.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Gupta, H., McCann, M. T., Donati, L., and Unser, M. (2021). Cryogan: a new reconstruction paradigm for single-particle cryo-em via deep adversarial learning. *IEEE Transactions on Computational Imaging*, 7:759–774.

Gupta, H., Phan, T. H., Yoo, J., and Unser, M. (2020). Multi-cryogan: Reconstruction of continuous conformations in cryo-em using generative adversarial networks. In *European Conference on Computer Vision*, pages 429–444. Springer.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heimowitz, A., Andén, J., and Singer, A. (2018). Apple picker: Automatic particle picking, a low-effort cryo-em framework. *Journal of structural biology*, 204(2):215–227.

Henderson, R. (2013). Avoiding the pitfalls of single particle cryo-electron microscopy: Einstein from noise. *Proceedings of the National Academy of Sciences*, 110(45):18037–18041.

Henderson, R., Sali, A., Baker, M. L., Carragher, B., Devkota, B., Downing, K. H., Egelman, E. H., Feng, Z., Frank, J., Grigorieff, N., et al. (2012). Outcome of the first electron microscopy validation task force meeting. *Structure*, 20(2):205–214.

Henderson, R. and Unwin, P. N. T. (1975). Three-dimensional model of purple membrane obtained by electron microscopy. *Nature*, 257(5521):28–32.

Hestness, J., Ardalani, N., and Diamos, G. (2019). Beyond human-level accuracy: Computational challenges in deep learning. In *Proceedings of the 24th Symposium on Principles and Practice of Parallel Programming*, pages 1–14.

Heymann, J. B., Chagoyen, M., and Belnap, D. M. (2005). Common conventions for interchange and archiving of three-dimensional electron microscopy information in structural biology. *J. Struct. Biol.*, 151(2):196–207.

Heymann, J. B., Conway, J. F., and Steven, A. C. (2004). Molecular dynamics of protein complexes from four-dimensional cryo-electron microscopy. *Journal of structural biology*, 147(3):291–301.

Hilgenfeld, R. and Peiris, M. (2013). From SARS to MERS: 10 years of research on highly pathogenic human coronaviruses. *Antiviral Res.*, 100(1):286–295.

Hillen, H. S., Kokic, G., Farnung, L., Dienemann, C., Tegunov, D., and Cramer, P. (2020). Structure of replicating SARS-CoV-2 polymerase. *Nature*, 584(7819):154–156.

Hoang, T. V., Cavin, X., Schultz, P., and Ritchie, D. W. (2013). gempicker: A highly parallel gpu-accelerated particle picking tool for cryo-electron microscopy. *BMC structural biology*, 13(1):1–10.

Högbom, M., Jäger, K., Robel, I., Unge, T., and Rohayem, J. (2009). The active form of the norovirus RNA-dependent RNA polymerase is a homodimer with cooperative activity. *J. Gen. Virol.*, 90(2):281–291.

Imbert, I., Guillemot, J.-C., Bourhis, J.-M., Bussetta, C., Coutard, B., Egloff, M.-P., Ferron, F., Gorbalenya, A. E., and Canard, B. (2006). A second, non-canonical RNA-dependent RNA polymerase in SARS Coronavirus. *EMBO J.*, 25(20):4933–4942.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. (2015). Spatial transformer networks. *Advances in neural information processing systems*, 28.

Jiang, Y., Yin, W., and Xu, H. E. (2021). RNA-dependent RNA polymerase: Structure, mechanism, and drug discovery for COVID-19. *Biochem. Biophys. Res. Commun.*, 538:47–53.

Jochheim, F. A., Tegunov, D., Hillen, H. S., Schmitzová, J., Kokic, G., Dienemann, C., and Cramer, P. (2021). The structure of a dimeric form of sars-cov-2 polymerase. *Communications biology*, 4(1):1–5.

Kannan, S. R., Spratt, A. N., Quinn, T. P., Heng, X., Lorson, C. L., Sönnerborg, A., Byrareddy, S. N., and Singh, K. (2020). Infectivity of SARS-CoV-2: there Is Something More than D614G? *J. Neuroimmune Pharmacol.*, 15(4):574–577.

Kiefer, J. and Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, pages 462–466.

Kim, D., Lee, J.-Y., Yang, J.-S., Kim, J. W., Kim, V. N., and Chang, H. (2020). The Architecture of SARS-CoV-2 Transcriptome. *Cell*, 181(4):914–921.e10.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kirchdoerfer, R. N. and Ward, A. B. (2019). Structure of the SARS-CoV nsp12 polymerase bound to nsp7 and nsp8 co-factors. *Nat. Commun.*, 10(1):2342.

Kokic, G., Hillen, H. S., Tegunov, D., Dienemann, C., Seitz, F., Schmitzova, J., Farnung, L., Siewert, A., Höbartner, C., and Cramer, P. (2021). Mechanism of SARS-CoV-2 polymerase stalling by remdesivir. *Nat. Commun.*, 12(1):279.

Krichel, B., Bylapudi, G., Schmidt, C., Blanchet, C., Schubert, R., Brings, L., Koehler, M., Zenobi, R., Svergun, D., Lorenzen, K., Madhugiri, R., Ziebuhr, J., and Uetrecht, C.

(2021). Hallmarks of Alpha- and Betacoronavirus non-structural protein 7+8 complexes. *Sci. Adv.*, 7(10).

Kühlbrandt, W. (2014). The Resolution Revolution. *Science (80-.).*, 343(6178):1443–1444.

Lawson, C. L., Patwardhan, A., Baker, M. L., Hryc, C., Garcia, E. S., Hudson, B. P., Lagerstedt, I., Ludtke, S. J., Pintilie, G., Sala, R., et al. (2016). Emdatabank unified data resource for 3dem. *Nucleic acids research*, 44(D1):D396–D403.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Levy, A., Poitevin, F., Martel, J., Nashed, Y., Peck, A., Miolane, N., Ratner, D., Dunne, M., and Wetzstein, G. (2022). Cryoai: Amortized inference of poses for ab initio reconstruction of 3d molecular volumes from real cryo-em images. *arXiv preprint arXiv:2203.08138*.

Liu, W. and Frank, J. (1995). Estimation of variance distribution in three-dimensional reconstruction I Theory. *J. Opt. Soc. Am. A*, 12(12):2615.

Lyu, L., Feng, R., Zhang, M., Xie, X., Liao, Y., Zhou, Y., Guo, X., Su, B., Dorsett, Y., and Chen, L. (2022). Subgenomic rna profiling suggests novel mechanism in coronavirus gene regulation and host adaption. *Life science alliance*, 5(8).

Lyumkis, D. (2019). Challenges and opportunities in cryo-em single-particle analysis. *Journal of Biological Chemistry*, 294(13):5181–5197.

Lyumkis, D., Brilot, A. F., Theobald, D. L., and Grigorieff, N. (2013). Likelihood-based classification of cryo-em images using frealign. *Journal of structural biology*, 183(3):377–388.

Malone, B., Chen, J., Wang, Q., Llewellyn, E., Choi, Y. J., Olinares, P. D. B., Cao, X., Hernandez, C., Eng, E. T., Chait, B. T., Shaw, D. E., Landick, R., Darst, S. A., and Campbell, E. A. (2021). Structural basis for backtracking by the SARS-CoV-2 replication–transcription complex. *Proc. Natl. Acad. Sci.*, 118(19):1–23.

Marques, M. A., Purdy, M. D., and Yeager, M. (2019). Cryoem maps are full of potential. *Current Opinion in Structural Biology*, 58:214–223.

Mastronarde, D. N. (2005). Automated electron microscope tomography using robust prediction of specimen movements. *J. Struct. Biol.*, 152(1):36–51.

Merk, A., Bartesaghi, A., Banerjee, S., Falconieri, V., Rao, P., Davis, M. I., Pragani, R., Boxer, M. B., Earl, L. A., Milne, J. L., et al. (2016). Breaking cryo-em resolution barriers to facilitate drug discovery. *Cell*, 165(7):1698–1707.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*.

Nagy, Á., Pongor, S., and Győrffy, B. (2021). Different mutations in SARS-CoV-2 associate with severe and mild outcome. *Int. J. Antimicrob. Agents*, 57(2):106272.

Nakane, T., Kimanius, D., Lindahl, E., and Scheres, S. H. (2018). Characterisation of molecular motions in cryo-EM single-particle data by multi-body refinement in RELION. *Elife*, 7.

Nakane, T., Kotecha, A., Sente, A., McMullan, G., Masiulis, S., Brown, P. M., Grigoras, I. T., Malinauskaite, L., Malinauskas, T., Miehling, J., et al. (2020). Single-particle cryo-em at atomic resolution. *Nature*, 587(7832):152–156.

Nguyen, T. H. D., Galej, W. P., Bai, X.-c., Oubridge, C., Newman, A. J., Scheres, S. H., and Nagai, K. (2016). Cryo-em structure of the yeast u4/u6. u5 tri-snrnp at 3.7 å resolution. *Nature*, 530(7590):298–302.

Nogales, E. (2016). The development of cryo-em into a mainstream structural biology technique. *Nature methods*, 13(1):24–27.

Parker, M. D., Lindsey, B. B., Leary, S., Gaudieri, S., Chopra, A., Wyles, M., Angyal, A., Green, L. R., Parsons, P., Tucker, R. M., et al. (2021). Subgenomic rna identification in sars-cov-2 genomic sequencing data. *Genome research*, 31(4):645–658.

Parker, M. D., Stewart, H., Shehata, O. M., Lindsey, B. B., Shah, D. R., Hsu, S., Keeley, A. J., Partridge, D. G., Leary, S., Cope, A., et al. (2022). Altered subgenomic rna abundance provides unique insight into sars-cov-2 b. 1.1. 7/alpha variant infections. *Communications biology*, 5(1):1–10.

Pasternak, A. O., van den Born, E., Spaan, W. J., and Snijder, E. J. (2002). Sequence requirements for RNA strand transfer during nidovirus discontinuous subgenomic RNA synthesis. *EMBO J.*, 20(24):7220–7228.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala,

S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Penczek, P. A., Fang, J., Li, X., Cheng, Y., Loerke, J., and Spahn, C. M. (2014). Cter—rapid estimation of ctf parameters with error assessment. *Ultramicroscopy*, 140:9–19.

Penczek, P. A., Kimmel, M., and Spahn, C. M. (2011). Identifying conformational states of macromolecules by eigen-analysis of resampled cryo-EM images. *Structure*, 19(11):1582–1590.

Pettersen, E. F., Goddard, T. D., Huang, C. C., Meng, E. C., Couch, G. S., Croll, T. I., Morris, J. H., and Ferrin, T. E. (2021). UCSF ChimeraX: Structure visualization for researchers, educators, and developers. *Protein Sci.*, 30(1):70–82.

Pospich, S., Sweeney, H. L., Houdusse, A., and Raunser, S. (2021). High-resolution structures of the actomyosin-v complex in three nucleotide states provide insights into the force generation mechanism. *Elife*, 10:e73724.

Posthuma, C. C., te Velthuis, A. J., and Snijder, E. J. (2017). Nidovirus RNA polymerases: Complex enzymes handling exceptional RNA genomes. *Virus Res.*, 234:58–73.

Punjani, A. and Fleet, D. J. (2021). 3D Variability Analysis: Resolving continuous flexibility and discrete heterogeneity from single particle cryo-EM. *bioRxiv*, page 2020.04.08.032466.

Punjani, A., Rubinstein, J. L., Fleet, D. J., and Brubaker, M. A. (2017). cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. *Nat. Methods*, 14(3):290–296.

Reboul, C. F., Eager, M., Elmlund, D., and Elmlund, H. (2018). Single-particle cryo-em—improved ab initio 3d reconstruction with simple/prime. *Protein Science*, 27(1):51–61.

Reshamwala, S. M. S., Likhite, V., Degani, M. S., Deb, S. S., and Noronha, S. B. (2021). Mutations in SARS-CoV-2 nsp7 and nsp8 proteins and their predicted impact on replication/transcription complex structure. *J. Med. Virol.*, 93(7):4616–4619.

Rohou, A. and Grigorieff, N. (2015). Ctffind4: Fast and accurate defocus estimation from electron micrographs. *Journal of structural biology*, 192(2):216–221.

Romano, M., Ruggiero, A., Squeglia, F., Maga, G., and Berisio, R. (2020). A Structural View of SARS-CoV-2 RNA Replication Machinery: RNA Synthesis, Proofreading and Final Capping. *Cells*, 9(5):1267.

Roseman, A. (2004). Findem—a fast, efficient program for automatic selection of particles from electron micrographs. *Journal of structural biology*, 145(1-2):91–99.

Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton*. Cornell Aeronautical Laboratory.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.

Rosenthal, P. B. and Henderson, R. (2003). Optimal determination of particle orientation, absolute hand, and contrast loss in single-particle electron cryomicroscopy. *Journal of molecular biology*, 333(4):721–745.

Ru, H., Chambers, M. G., Fu, T.-M., Tong, A. B., Liao, M., and Wu, H. (2015). Molecular mechanism of v (d) j recombination from synaptic rag1-rag2 complex structures. *Cell*, 163(5):1138–1152.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Sawicki, S. G. and Sawicki, D. L. (1998). A New Model for Coronavirus Transcription. *Adv. Exp. Med. Biol.*, 440:215–219.

Scheres, S. H. (2012a). A Bayesian View on Cryo-EM Structure Determination. *J. Mol. Biol.*, 415(2):406–418.

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

Scheres, S. H. (2015). Semi-automated selection of cryo-em particles in relion-1.3. *Journal of structural biology*, 189(2):114–122.

Schilbach, S., Hantsche, M., Tegunov, D., Dienemann, C., Wigge, C., Urlaub, H., and Cramer, P. (2017). Structures of transcription pre-initiation complex with tfiih and mediator. *Nature*, 551(7679):204–209.

Seide, F. and Agarwal, A. (2016). Cntk: Microsoft's open-source deep-learning toolkit. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2135–2135.

Singer, A. and Sigworth, F. J. (2020). Computational methods for single-particle cryo-em. *arXiv preprint arXiv:2003.13828*.

Snijder, E., Decroly, E., and Ziebuhr, J. (2016). The Nonstructural Proteins Directing Coronavirus RNA Synthesis and Processing. *Adv. Virus Res.*, 96:59–126.

Sola, I., Almazán, F., Zúñiga, S., and Enjuanes, L. (2015). Continuous and Discontinuous RNA Synthesis in Coronaviruses. *Annu. Rev. Virol.*, 2(1):265–288.

Sorzano, C., Marabini, R., Velázquez-Muriel, J., Bilbao-Castro, J., Scheres, S., Carazo, J., and Pascual-Montano, A. (2004). XMIPP: a new generation of an open-source image processing package for electron microscopy. *J. Struct. Biol.*, 148(2):194–204.

Sorzano, C. O. S. and Carazo, J. M. (2021). Principal component analysis is limited to low-resolution analysis in cryoem. *Acta Crystallographica Section D: Structural Biology*, 77(6):835–839.

Sorzano, C. O. S., Vargas, J., Otón, J., de la Rosa-Trevín, J., Vilas, J., Kazemi, M., Melero, R., Del Caño, L., Cuenca, J., Conesa, P., et al. (2017). A survey of the use of iterative reconstruction algorithms in electron microscopy. *BioMed research international*, 2017.

Subissi, L., Posthuma, C. C., Collet, A., Zevenhoven-Dobbe, J. C., Gorbalenya, A. E., Decroly, E., Snijder, E. J., Canard, B., and Imbert, I. (2014). One severe acute respiratory syndrome coronavirus protein complex integrates processive RNA polymerase and exonuclease activities. *Proc. Natl. Acad. Sci.*, 111(37):E3900—-E3909.

Tagare, H. D., Kucukelbir, A., Sigworth, F. J., Wang, H., and Rao, M. (2015). Directly reconstructing principal components of heterogeneous particles from cryo-EM images. *J. Struct. Biol.*, 191(2):245–262.

Tan, Y. Z., Baldwin, P. R., Davis, J. H., Williamson, J. R., Potter, C. S., Carragher, B., and Lyumkis, D. (2017). Addressing preferred specimen orientation in single-particle cryo-EM through tilting. *Nat. Methods*, 14(8):793–796.

te Velthuis, A. J. W. (2014). Common and unique features of viral RNA-dependent polymerases. *Cell. Mol. Life Sci.*, 71(22):4403–4420.

Tegunov, D. and Cramer, P. (2019). Real-time cryo-electron microscopy data preprocessing with Warp. *Nat. Methods*, 16(11):1146–1152.

Villani, C. (2009). *Optimal transport: old and new*, volume 338. Springer.

V'kovski, P., Kratzel, A., Steiner, S., Stalder, H., Thiel, V., V'kovski, P., Kratzel, A., Steiner, S., Stalder, H., and Thiel, V. (2021). Coronavirus biology and replication: implications for SARS-CoV-2. *Nat. Rev. Microbiol.*, 19(3):155–170.

Voss, N., Yoshioka, C., Radermacher, M., Potter, C., and Carragher, B. (2009). Dog picker and tiltpicker: software tools to facilitate particle selection in single particle electron microscopy. *Journal of structural biology*, 166(2):205–213.

Wang, H.-W. and Fan, X. (2019). Challenges and opportunities in cryo-em with phase plate. *Current Opinion in Structural Biology*, 58:175–182.

Wang, H.-W. and Wang, J.-W. (2017). How cryo-electron microscopy and x-ray crystallography complement each other. *Protein Science*, 26(1):32–39.

Wang, Q., Wu, J., Wang, H., Gao, Y., Liu, Q., Mu, A., Ji, W., Yan, L., Zhu, Y., Zhu, C., Fang, X., Yang, X., Huang, Y., Gao, H., Liu, F., Ge, J., Sun, Q., Yang, X., Xu, W., Liu, Z., Yang, H., Lou, Z., Jiang, B., Guddat, L. W., Gong, P., and Rao, Z. (2020). Structural Basis for RNA Replication by the SARS-CoV-2 Polymerase. *Cell*, 182(2):417–428.e13.

Ward, R., Wu, X., and Bottou, L. (2020). Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *The Journal of Machine Learning Research*, 21(1):9047–9076.

Wittenborn, E. C. and Marletta, M. A. (2021). Structural perspectives on the mechanism of soluble guanylate cyclase activation. *International journal of molecular sciences*, 22(11):5439.

Yan, L., Ge, J., Zheng, L., Zhang, Y., Gao, Y., Wang, T., Huang, Y., Yang, Y., Gao, S., Li, M., et al. (2021). Cryo-em structure of an extended sars-cov-2 replication and transcription complex reveals an intermediate state in cap synthesis. *Cell*, 184(1):184–193.

Yan, L., Zhang, Y., Ge, J., Zheng, L., Gao, Y., Wang, T., Jia, Z., Wang, H., Huang, Y., Li, M., Wang, Q., Rao, Z., and Lou, Z. (2020). Architecture of a SARS-CoV-2 mini replication and transcription complex. *Nat. Commun.*, 11(1).

Yi, X., Walia, E., and Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical image analysis*, 58:101552.

Yin, W., Mao, C., Luan, X., Shen, D.-D., Shen, Q., Su, H., Wang, X., Zhou, F., Zhao, W., Gao, M., Chang, S., Xie, Y.-C., Tian, G., Jiang, H.-W., Tao, S.-C., Shen, J., Jiang, Y., Jiang, H., Xu, Y., Zhang, S., Zhang, Y., and Xu, H. E. (2020). Structural basis for inhibition of the RNA-dependent RNA polymerase from SARS-CoV-2 by remdesivir. *Science (80-.).*, 368(6498):1499–1504.

Yip, K. M., Fischer, N., Paknia, E., Chari, A., and Stark, H. (2020). Atomic-resolution protein structure determination by cryo-em. *Nature*, 587(7832):157–161.

Zeng, H., Gao, X., Xu, G., Zhang, S., Cheng, L., Xiao, T., Zu, W., and Zhang, Z. (2022). Sars-cov-2 helicase nsp13 hijacks the host protein ewsr1 to promote viral replication by enhancing rna unwinding activity. *Infectious Medicine*, 1(1):7–16.

Zhai, Y., Sun, F., Li, X., Pang, H., Xu, X., Bartlam, M., and Rao, Z. (2005). Insights into SARS-CoV transcription and replication from the structure of the nsp7–nsp8 hexadecamer. *Nat. Struct. Mol. Biol.*, 12(11):980–986.

Zhang, C., Li, L., He, J., Chen, C., and Su, D. (2021). Nonstructural protein 7 and 8 complexes of SARS-CoV-2. *Protein Sci.*, 30(4):873–881.

Zhang, K. (2016). Gctf: Real-time ctf determination and correction. *Journal of structural biology*, 193(1):1–12.

Zheng, S. Q., Palovcak, E., Armache, J.-P., Verba, K. A., Cheng, Y., and Agard, D. A. (2017). Motioncor2: anisotropic correction of beam-induced motion for improved cryo-electron microscopy. *Nature methods*, 14(4):331–332.

Zhong, E. D., Bepler, T., Berger, B., and Davis, J. H. (2021a). Cryodrgn: reconstruction of heterogeneous cryo-em structures using neural networks. *Nature methods*, 18(2):176–185.

Zhong, E. D., Lerer, A., Davis, J. H., and Berger, B. (2021b). Cryodrgn2: Ab initio neural reconstruction of 3d protein structures from real cryo-em images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4066–4075.

Zivanov, J., Nakane, T., Forsberg, B. O., Kimanius, D., Hagen, W. J., Lindahl, E., and Scheres, S. H. (2018). New tools for automated high-resolution cryo-em structure determination in relion-3. *elife*, 7:e42166.

Zivanov, J., Nakane, T., and Scheres, S. H. (2020). Estimation of high-order aberrations and anisotropic magnification from cryo-em data sets in relion-3.1. *IUCrJ*, 7(2):253–267.

Zivanov, J., Nakane, T., and Scheres, S. H. W. (2019). A Bayesian approach to beam-induced motion correction in cryo-EM single-particle analysis. *IUCrJ*, 6:5–17.

Zúñiga, S., Sola, I., Alonso, S., and Enjuanes, L. (2004). Sequence Motifs Involved in the Regulation of Discontinuous Coronavirus Subgenomic RNA Synthesis. *J. Virol.*, 78(2):980–994.