The Raymond and Beverly Sackler Faculty of Exact Sciences
Tel Aviv University

Super-resolution SAXS based on PSF engineering and sub-pixel detector translations

Thesis submitted for the degree of Master of Science

by **Benjamin Gutman**

this work was carried out under the supervision of

Prof. Yoel Shkolnisky

March 2021

Abstract

Small-angle X-ray scattering (SAXS) is a technique that enables convenient nanoscopic characterization of various samples under various conditions. In a SAXS experiment, a sample is irradiated with an X-ray beam, resulting in a two-dimensional scattering pattern. Unfortunately, this scattering pattern is distorted by a point spread function (PSF), that can be controlled during the experiment.

In this thesis, we propose a deconvolution algorithm that takes multiple scattering patterns, each acquired with a different PSF, and recovers their underlying scattering pattern. The algorithm is based on casting the recovery as minimization problem, that is shown to have a unique solution, obtained using the conjugate gradients method. In addition, we propose cascading our algorithm with a sub-pixel super resolution algorithm [36]. We evaluate our algorithm using both numerical simulations as well as using experimental SAXS measurements.

Acknowledgements

I would like to thank my supervisor Prof. Yoel Shkolnisky for his professional support and endless encouragement and patience. Prof. Roy Beck Barkai who's laboratory was used to conduct the measurements and the computational process, and for many enlightening discussions and ideas for implementing the proposed algorithm. Michael Mrejen, Guy Jacoby, and Ram Avinery from prof. Beck's lab for helpful discussions and assistance with the measurements.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Small Angle X-ray Scattering (SAXS) is a method for studying the physical features of nanoscopic structures, which exploits the scattering pattern produced when X-ray photons scatter from a crystal structure [11]. Historically, SAXS experiments have been conducted in specialized synchrotron facilities, which harness their high-flux of X-ray photons to provide an adequate scattering signal, and till our days, important SAXS experiments are being conducted in such facilities. However, such experiments are not accessible to most industries and scientists, as synchrotron-based experiments are commonly over-booked world-wide. Recent advances in small scale lab-based X-ray sources and, in particular, in high-efficiency solid-state 2D X-ray detectors enable conducting some of the experiments using lab-based SAXS systems. Such systems are best suitable for the study of hours-long dynamics of a nanoscopic structural organization [18, 14, 19] or when SAXS is used to optimize sample preparation for a synchrotron experiment. Nevertheless, lab-based SAXS systems suffer from two major drawbacks. First, their X-ray beam has much lower flux compared to synchrotron-based systems, and second, they require large floor space. Below we discuss these two drawbacks, but we first describe SAXS image acquisition and formation.

Any camera, including an X-ray detector, consists of multiple pixels placed on a rectangular grid. Each pixel is a device that counts photons hitting it during a given period of time, known as the exposure time. At the end of the exposure time, counts

from all the pixels are stored on a matrix, called an image, representing a discrete version of the scene as captured by the camera [10]. A camera can be characterized by several attributes, but perhaps the most important one is it's ability to resolve separate objects in the image, termed "resolution". Image resolution depends on the number of pixels as well as their size.

Typically, the photon count in each pixel is characterized by Poisson statistics [38]. As a consequence, the signal to noise ratio (SNR) at each pixel is the square-root of the count ($n^{-1/2}$ where n is the photon count at the pixel). By the Central Limit Theorem, when n is large the Poisson distribution is well approximated by the Gaussian distribution [1]. Therefore, it is common to consider the noise in each pixel as Gaussian and additive. To obtain an adequate photon count in each pixel, for example in order to obtain a sufficiently high SNR, we need to properly set the exposure time, depending on the incoming flux of photons; the lower the incoming flux, the higher the exposure time required in order to obtain a given total photon count. However, long exposure times may be prohibitive due to budget constraints, as well as due to hardware related reasons (such as temperature limitations, power consumption, beam stability, etc [6]). So, low flux forces long exposure times, however, these may be limited. In order to overcome this, it is possible to use a different approach, aimed at holding fixed the total photon count.

Low flux is first and foremost a result of the low intensity X-ray source used by lab-based systems [46]. Moreover, after emitted from the source, the X-ray beam undergoes collimation prior to arriving to the sample, a process that further reduces the flux as explained next. X-ray sources emit photons in a range directions. In order to produce a collimated beam, it is passed through lead slits placed on the line between the source and the sample, which pass only photons moving in specific directions. At least two slits are placed between the source and the sample – one close to the source and the other close to the sample [11]. These slits block all photons not moving parallel to the line from the source to the sample. However, this photon-blocking process also reduces the flux of the beam arriving at the sample [31]. So, by compromising on the collimation of the beam, it is possible to increase the size of the lead slits, that in turn increases the number of photons passing through them,

thus increasing the flux. Increasing the size of the slits in lab-based SAXS systems results in a beam width which is 250 times wider than that of a synchrotron [39]. This increase in beam width blurs the acquired image, as explained next.

Suppose some real-world scene is to be acquired (in SAXS framework this scene is a scattering pattern, to be explained below). As explained earlier, due to the discrete nature of the pixels, any acquired image is a discrete version of the real-world scene. Furthermore, additional factors may affect the acquired images, such as the blur due to the non-collimated beam. Here, we denote by X the image that would have been acquired if no beam-blur occurred, and by Y the image that was actually acquired. Mathematically, we model the connection between X and the acquired image Y by [10]

$$Y = P * X + N, \tag{1.1}$$

where $*$ is 2D convolution operator, N is some noise, and P is a matrix that represents the point spread function (PSF) of the imaging device. In order to extract information from SAXS measurements, it is mandatory to retrieve the underlying X out of the acquired Y.

A tight connection exists between the slit shape and the PSF [32]. Suppose we acquire an image with no sample in front of the detector. Placing a slit right in front of the detector would form an image similar to the shape of the slit, since only photons that pass the slit arrive at the detector. So, having a small rectangular slit will form a small rectangle in the acquired image, with similar dimensions to that of the slit (assume the slit is much larger than the pixel size). Since photons move in straight lines, increasing the distance between the slit and the detector would produce a rectangular image, but with increasing dimensions. This occurs since the beam is never perfectly collimated, and photons with non-parallel motion pass the slit, and arrive at pixels beyond the close slit-to-detector case. Thus, the image of the beam is highly correlated with the shape of the slit. If we change the slit shape to a wider rectangle, the out-coming image would be a wider rectangle in the image. The naked beam acquisition is relevant to a scattering pattern acquisition in the following fashion: if a sample is placed between the last slit and the detector, the

beam shape hitting the sample is almost identical to the naked beam image, and so the naked beam acquisition can be considered as the PSF.

With respect to the previously mentioned slit size changing technique, it can be seen that by varying the slit's shape from very tight to increasingly wider, the operator P blurs X in respectively increasing manner, as the following thought experiment demonstrates. Assume a very strong source and a pinhole-like slit. The resulting PSF can be well approximated by a delta-function, and hence such a PSF will result in no blur, since convolution with a delta-function is the identity operator. The strong beam ensures high flux despite the given slit. On the other hand, increasing the size of the slit produces some other, wider, PSF, resulting in blurring X due to the convolution. The effect of P in blurring X can also be analyzed from the frequency point of view. Performing DFT on Eq. 1.1, we get that the convolution transforms into entry-wise multiplication, and multiplying the DFT of X with the DFT of P might result in attenuating some frequencies. In the delta function PSF case, the DFT transforms it into a constant, so the entire spectrum of X can be found in the spectrum of Y. On the other hand, any other PSF will attenuate frequencies or even zero some of them altogether. In other words, some frequencies of X might get lost in the acquisition process and would not be found in Y.

Sometimes the expected scattering pattern X in known, and so the expected DFT of X is known as well. If this spectrum is expected to have frequencies of small magnitude, a PSF can be engineered in order to ensure that these frequencies would be attenuated in the acquisition process. Attenuating insignificant frequencies in the DFT of X produces Y that has a spectrum similar to that of X, except for frequencies that have small impact on the underlying X. Knowing the required PSF for this process, it can be generated using the right slit-shape [11], and contemporary lab-based SAXS operators indeed use this knowledge in order to choose a roughly adequate slit shape.

However, the latter approach requires a very good knowledge of the expected scattering pattern, which is often unavailable. So X needs to be recovered from its blurry version Y by some digital deblurring algorithm. Several such algorithms have been proposed over the years [27, 35, 48]. A naive solution to the deblurring

problem is neglecting the additive noise and simply inverting Eq. 1.1. However, this solution is, in general, ill-conditioned. In order to see this, we turn once again to the spectral point of view. Performing DFT on Eq. 1.1 results in $\hat{Y} = \hat{P}\hat{X} + \hat{N}$, where $\hat{[\cdot]}$ represents the DFT of $[\cdot]$, and the multiplication is entry-wise. Assuming P is square, neglecting $\hat{N}$ and solving for $\hat{X}$ results in $\hat{X} = \hat{Y}/\hat{P}$, from which an estimate for X can be obtained using the inverse DFT of $\hat{Y}/\hat{P}$, where the division is entry-wise. Both Y and P are measured so computing $\hat{Y}/\hat{P}$ should be trivial. However, $\hat{P}$ may have values that are close to zero, which causes amplification of noise and measurement errors when computing $\hat{Y}/\hat{P}$. Since the DFT matrix is orthogonal, these errors persist when estimating X using the inverse DFT. We next present several deconvolution algorithms and their approach to overcoming this problem in estimating X.

A first attempt of avoiding this issue is the inverse filtering method. Instead of using $1/\hat{P}$, we look at $\hat{P}^*/(\hat{P}^*\hat{P} + \varepsilon)$ where $\hat{P}^*$ is the imaginary entry-wise conjugate of $\hat{P}$ and $\varepsilon > 0$, with all multiplications and divisions being entry-wise. If principle, an optimal ε can be obtained and this solution for $1/\hat{P}$ is adequate for simple tasks (i.e. when the noise can be neglected). However, it suffers from several major drawbacks: an optimal ε is not easy to find and a constant ε for all frequencies neglects the different contribution of frequencies to the image. In order to overcome these drawbacks, Wiener [48] proposed using prior knowledge on the noise (which in our setting we assumed to be Gaussian) and replacing the constant ε by a matrix corresponding to the SNR at each frequency. In order to do so, we define the reciprocal spectral SNR matrix: $\mathfrak{N} = \mathbb{E}[|\hat{N}|^2]/\mathbb{E}[|\hat{Y}|^2]$, and apply $(\hat{P}^*\hat{P} + \mathfrak{N})^{-1}\hat{P}^*$ on $\hat{Y}$. This results in a better estimate of $\hat{X}$ [2].

A different approach to the deconvolution problem is to use iterative algorithms, with the Richardson-Lucy algorithm [27, 35] being the most common algorithm of this class. The main idea of the Richardson-Lucy algorithm is to estimate X using the iterations $X^{(t+1)} = X^{(t)} \cdot (\frac{Y}{X^{(t)}P}\breve{P})$, where $\breve{P}$ is the flipped PSF, to be defined later, and both the multiplication and the division are entry-wise. It is shown in [27] that these iterations converge since the expression inside the parentheses approaches unity as t grows. The main drawback of this algorithm is that it has a relatively slow

convergence, and moreover, in the presence of noise, performing too many iterations might actually amplify the noise [37], so fixing the right number of iterations is critical. Several improvements to this algorithm were proposed, trying overcome these drawbacks [7, 49]. Nevertheless, iterative deconvolution algorithms SAXS data are performing poorly [45].

We now turn to discuss the large floor space issue mentioned above. To that end, we first explain what is a scattering pattern. Photons arriving at a sample from a fixed direction experience force arising from the atomic structure of the sample, and are being scattered in various directions. The photons are then collected by a camera, forming what is known as the scattering pattern of the sample. It turns out that these directions are characterized by a single angle, 2θ, measured relative to the X-ray beam (see Fig. 1.1), namely, there is an azimuthal symmetry of the scattering pattern around a well defined center [11]. The main goal of SAXS experiments is to find the distribution of scattered photons as a function of θ, with special interest in small values of θ. Since the scattering angle of each photon cannot be measured directly, the scattering pattern is analyzed in the following way: a photon arriving at some point on the detector is characterized by the distance between the arriving point and the center of the scattering pattern, and this distance is related to the scattering angle of the photon using simple trigonometry (see Fig. 1.1), so any photon is related to its scattering angle 2θ, and the distribution of angles can be inferred using multiple photons.

However, the discrete nature of the detector puts some reservations on this description. Since pixels have finite size, each of the pixels corresponds to a range of distances from the center, and hence to a range of scattering angles. So each pixel produces the photon count corresponding to a range of scattering angles. The sensitivity of a system in distinguishing small differences in θ is known as angular resolution, and can be refined using a larger sample to detector distance. The reason is once again trigonometry. Holding the angle fixed and increasing the sample to detector distance increases the distance on the detector, and hence angles that where indistinguishable are now mapped to different pixels. However, the long sample to detector distance required to achieve high angular resolution comes at the expanse

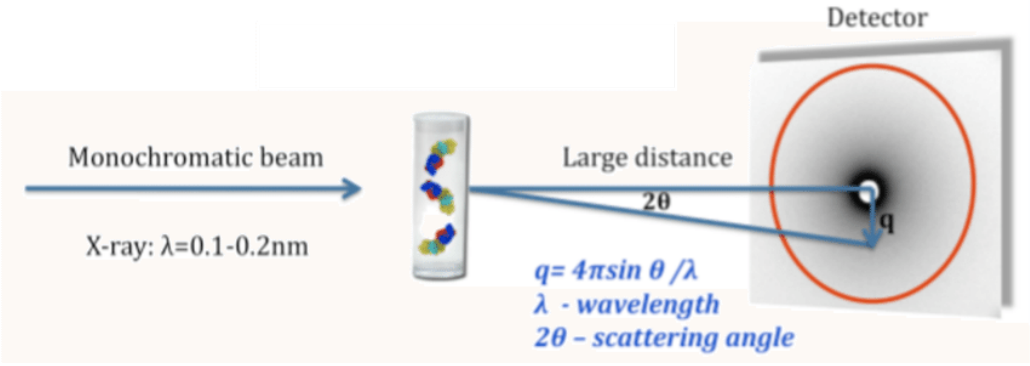

Figure 1.1: SAXS schematic setup. Image taken from [5]. Sample here is represented by the capsule between the beam and the detector.

of large floor space, which may not be available in all facilities.

We now turn back to image processing point of view. According to the Nyquist sampling theorem [28], the detector samples the scattering pattern at a spatial sampling rate given by the pixel size, and hence frequencies beyond this limit cannot be sampled. The multiframe super-resolution technique, originally proposed by Tsai and Huang [43], aims to enrich the acquired spectral information beyond the Nyquist limit, by taking several images of the same scene, while moving the detector in sub-pixel translations. This idea is explained in a toy model shown in Fig 1.2. Assume we acquire a single image (a) and find that a single pixel is illuminated. The red marker represents the location where photons hit, however, this information is unavailable since no sub-pixel information is at hand. Looking at image (a) without the red marker, we can't tell if the photons are uniformly spread across the entire pixel, or are distributed otherwise in the pixel. Moving the detector half a pixel size to the right (while holding the rest of the setup fixed) results in the image (b). Due to the displacement of the detector, the photons are now hitting another pixel, and it can be inferred that the photons are hitting the left half of the pixel they illuminate (in (a)). Note that if the photons were distributed uniformly across the pixel, image (b) would have had two illuminated pixels, with half intensity each. So, sub-pixel information was achieved. This idea can be further extended by moving the detector up, resulting in image (c), and up and right, resulting in image (d). The illuminated

pixel is different in each of the acquired images, and from its relative displacements between the acquisitions (frames) it can be inferred that all the photons are concentrated in the lower left corner of the illuminated pixel in (a). This technique is well studied [30, 34, 20, 24, 44, 17, 9] and explained in details in [36], but in all cases its underlying idea is to take several frames with sub-pixel displacement, align them to one another (as we did by knowing the relative displacement) and then fuse the frames into a single image. This enables us achieving better resolution. Returning to our discussion above regarding the trade-off between angular resolution and floor space, the improved resolution achieved by sub-pixel detector translations can be traded for smaller sample to detector distances, and hence smaller floor space.

In this thesis, we propose taking multiple snapshots of a single X while using different slit shapes. This results in multiple acquired signals Y, each corresponding to a different PSF. The general idea of our algorithm is to use the measured PSFs and the acquired Ys to set up a minimization problem, for which the underlying X is a solution, show that this solution is unique, and show how to compute it efficiently. This algorithm offers a solution to the low flux problem since is enables to capture a rich portion of X's spectrum and hence restore it with superior details. In addition, in order to overcome the large floor space requirement, we propose using an 'off the shelf' protocol called 'multiframe super-resolution' proposed by Farsiu et al. [36], and cascade it with our proposed deconvolution algorithm. To our best knowledge, our proposed algorithm is the first implementation of a multiframe super-resolution SAXS system.

The remaining of the thesis is organized as follows. In Chapter 2, we present our proposed algorithm. In Chapter 3, we present the numerical results for our algorithm using simulated data (Section 3.2) and using measured SAXS data (Section 3.3). Last, In Chapter 4, we summarize this work and propose some future directions.

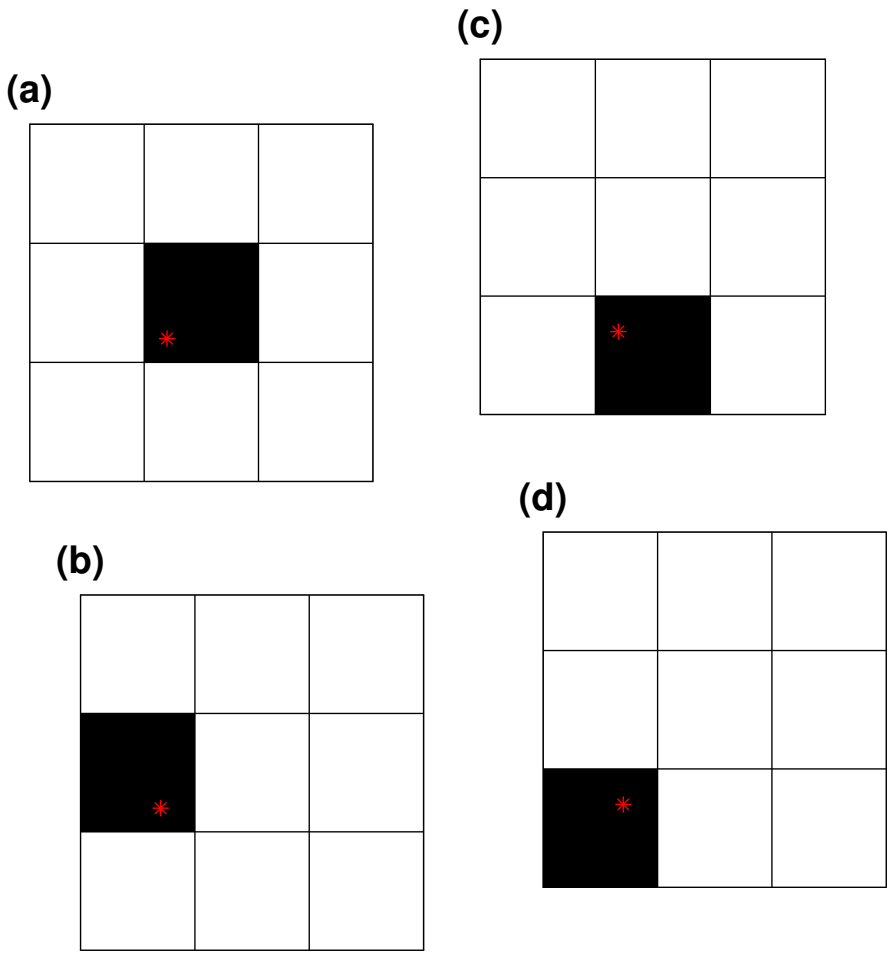

Figure 1.2: Toy example for the sub-pixel multiframe technique. (a) original image, (b) image with sub-pixel detector translation of 1/2 pixel to the right, (c) image with sub-pixel detector translation of 1/2 pixel up, and (d) image with sub-pixel detector translation of 1/2 pixel to both up and right directions. The red marker represents the true location of the hitting photons.

Chapter 2

Proposed algorithm

Following the discussion in the previous chapter regarding the blurring of the measured scattering pattern incurred by any practical slit, we present a deconvolution algorithm for estimating the scattering pattern of a given sample that is based on measuring it using various slits, each corresponding to a different PSF. First, we formulate the mathematical model underlying our estimation problem and propose some naive solution for it. Then, we extend this solution to a more robust one, and finally, we explain how to implement it efficiently.

As mentioned above, our approach is based on taking several images of the same sample while varying the shape of the slits, namely the corresponding PSFs. Following Eq. 1.1, we model the the acquired images as

$$Y_i = X * P_i + N_i, \tag{2.1}$$

where $i = 1, \ldots, m$ indexes the m acquired images, with image Y_i corresponding to the measured PSF P_i. Our goal is to recover a single image X common to all acquired images Y_i, which corresponds to the (noiseless and de-blurred) scattering pattern of the imaged sample.

A direct least squares solution of Eq. 2.1 can be formulated as

$$\bar{X} = \underset{X}{\operatorname{argmin}} \sum_i ||Y_i - P_i * X||_F^2, \tag{2.2}$$

where $||.||_F$ is the Frobenius norm given by $||A||_F = \sqrt{\sum_{i,j} A_{i,j}^2}$, and the convolution operator is such that both X and $P_i * X$ are of the dimensions of Y_i. However, this formulation has several drawbacks. First, it weights all the acquired images equivalently, despite the fact that some of them may contribute nothing but noise to the estimated solution. Such acquired images arise when their corresponding PSFs are too tight or too wide, as significant details of X are then either lost due to low SNR or are blurred due to aggressive blur. In addition, Eq. 2.2 may be ill-conditioned (as shown in [17] as well as below shortly), and does not take into account additional known information about the underlying image X [21].

Thus, the significance of each image Y_i in Eq. 2.1 needs to be controlled. This in done by imposing a different weight on each Y_i. The specific choice of these weights should depend on the underlying scattering image X, as the following thought experiment demonstrates. Assuming a scattering pattern that has fine structures, the PSFs used to resolve it must be as tight as possible, since wide PSFs will blur its fine structures. Therefore, wide PSFs should be given lower weights by the algorithm. On the other hand, if the scattering pattern is known to have no fine structures, images taken with tight PSFs will suffer from low SNR, while not contributing fine details to the recovered pattern. Our proposed weight function to be integrated into Eq. 2.2 is presented below.

As for the fact that Eq. 2.2 may be ill-conditioned, we first show where does this ill-conditioning come from. Since convolution is a linear operator, it can be represented by a corresponding matrix [13], and Eq. 2.2 becomes

$$\tilde{\bar{X}} = \underset{\tilde{X}}{\operatorname{argmin}} \sum_i ||\tilde{Y}_i - \Gamma_i \tilde{X}||_F^2, \tag{2.3}$$

where Γ_i is the matrix representation of P_i*, and $\tilde{X}$ and $\tilde{Y}_i$ are vectors given by column-stacking X and Y_i, respectively. Now, differentiating Eq. 2.3 with respect to $\tilde{X}$ and equating to zero, we get

$$2 \sum_i \Gamma_i^T (\tilde{Y}_i - \Gamma_i \tilde{X}) = 0, \tag{2.4}$$

or equivalently,

$$(\sum_i \Gamma_i^T \Gamma_i)\tilde{X} = \sum_i \Gamma_i^T \tilde{Y}_i. \tag{2.5}$$

The matrix $(\sum_i \Gamma_i^T \Gamma_i)$ is symmetric and positive semi-definite (SPSD). It is symmetric since $(\Gamma_i^T \Gamma_i)^T = \Gamma_i^T (\Gamma_i^T)^T = \Gamma_i^T \Gamma_i$, and it is positive semi-definite since

$$\langle \Gamma_i^T \Gamma_i \tilde{X}, \tilde{X} \rangle = \langle \Gamma_i \tilde{X}, \Gamma_i \tilde{X} \rangle = ||\Gamma_i \tilde{X}||^2 \geq 0. \tag{2.6}$$

Hence, all eigenvalues of $\Gamma_i^T \Gamma_i$ are non-negative. Since a sum of SPSD matrices is SPSD, we conclude that $(\sum_i \Gamma_i^T \Gamma_i)$ is a SPSD matrix. However, since some eigan-values of $(\sum_i \Gamma_i^T \Gamma_i)$ might be small or even zero, Eq. 2.2 may be ill-conditioned.

We therefore present two modifications to Eq. 2.2. First, we weight the i^{th} term in Eq. 2.2 by a weight $2\sigma_i^2$ (we choose the weights to be of this form instead of just σ_i to simplify subsequent derivations). The specific choice of the σ_is is discussed in Chapter 3. In addition, we add to Eq. 2.2 a ridge regularization term $\frac{\nu}{2}||X||_F^2$ [12], to ensure that the zero eigenvalue is avoided, as shown shortly (and again, the division by 2 is for simplicity of derivation). Such a regularization term is natural for images X that are smooth [21], i.e. have no sharp differences in intensities between neighboring pixels. The resulting optimization problem is thus

$$\bar{X} = \underset{X}{\operatorname{argmin}} \sum_i \frac{||Y_i - P_i * X||_F^2}{2\sigma_i^2} + \frac{\nu}{2}||X||_F^2. \tag{2.7}$$

In order to show the effect of the ridge regularization term on Eq. 2.7, we rewrite it using the notation of Eq. 2.3 to get

$$\bar{\tilde{X}} = \underset{\tilde{X}}{\operatorname{argmin}} \sum_i \frac{||\tilde{Y}_i - \Gamma_i \tilde{X}||_F^2}{2\sigma_i^2} + \frac{\nu}{2}||\tilde{X}||_F^2. \tag{2.8}$$

Differentiating the right-hand-side of Eq. 2.8 with respect to $\tilde{X}$ and equating the result to zero gives

$$\left(\sum_i \frac{\Gamma_i^T \Gamma_i}{\sigma_i^2}\right)\tilde{X} + \nu I \tilde{X} = \sum_i \frac{\Gamma_i^T \tilde{Y}_i}{\sigma_i^2}, \tag{2.9}$$

which is a linear equation in $\tilde{X}$. Moreover, choosing $\nu > 0$ we have

$$\begin{aligned}\langle(\sum_i \frac{\Gamma_i^T\Gamma_i}{\sigma_i^2} + \nu I)\tilde{X}, \tilde{X}\rangle &= \sum_i \langle\frac{\Gamma_i^T\Gamma_i}{\sigma_i^2}\tilde{X}, \tilde{X}\rangle + \langle\nu I\tilde{X}, \tilde{X}\rangle \\ &= \sum_i \langle\frac{\Gamma_i}{\sigma_i}\tilde{X}, \frac{\Gamma_i}{\sigma_i}\tilde{X}\rangle + \nu\langle\tilde{X}, \tilde{X}\rangle \\ &= \sum_i ||\frac{\Gamma_i}{\sigma_i}\tilde{X}||^2 + \nu||\tilde{X}||^2 \geq \nu||\tilde{X}||^2 > 0.\end{aligned}$$

Thus, the eigenvalues of $(\sum_i \frac{\Gamma_i^T\Gamma_i}{\sigma_i^2} + \nu I)$ are strictly positive (unlike those of $\sum_i \Gamma_i^T\Gamma_i$ that correspond to Eq. 2.3), and in particular, the solution of Eq. 2.7 is unique. Moreover, the regularization parameter ν can be used to control the condition number of the optimization problem in Eq. 2.7.

From a complexity point of view, a direct solution of Eq. 2.9 (using, for example, Gaussian elimination) is prohibitive, due to the size of the matrices involved. For example, the convolution of an $n \times n$ sized image with an $l \times l$ sized PSF transforms into a corresponding matrix multiplication representation in which an $(l^2+n^2-1)\times n^2$ sized matrix multiplies an $n^2 \times 1$ sized vector [13]. Assuming $n \gg l$, the complexity of this multiplication is $O(n^4)$ operations (for naive matrix multiplication), which is intractable for practical image sizes.

To efficiently solve Eq. 2.7, we rewrite it in a different form. For a set of measured PSFs $P_1, \ldots, P_m$, we define $\mathcal{P}$ to be the operator acting on images by

$$\mathcal{P}[\cdot] := \begin{pmatrix} P_1 * [\cdot] \\ P_2 * [\cdot] \\ \vdots \\ P_m * [\cdot] \\ \sqrt{\nu} I[\cdot] \end{pmatrix}, \tag{2.10}$$

and in a similar way, we stack the measured images as $\mathcal{Y} = [Y_1^T, Y_2^T, \ldots, Y_m^T, 0]^T$.

Using this notation, Eq. 2.7 becomes

$$\bar{X} = \underset{X}{\operatorname{argmin}} ||R(\mathcal{P}X - \mathcal{Y})||_F^2 = \underset{X}{\operatorname{argmin}} \left\| R \begin{pmatrix} P_1 * X & - & Y_1 \\ P_2 * X & - & Y_2 \\ \vdots & & \vdots \\ P_m * X & - & Y_m \\ \sqrt{\nu} I X & - & 0 \end{pmatrix} \right\|_F^2, \tag{2.11}$$

where R is a diagonal matrix given by

$$R = \frac{1}{\sqrt{2}} \operatorname{diag}(\sigma_1^{-1} I \quad \sigma_2^{-1} I \quad \dots \quad \sigma_m^{-1} I \quad I).$$

Note that the last row of Eq. 2.11 corresponds to the regularization term. Again, in order to solve Eq. 2.11, we differentiate its rightmost term with respect to X to get $(R\mathcal{P})^T R(\mathcal{P}X - \mathcal{Y})$, and by equating the latter to zero and using the diagonal form of R, we get the normal equations [8]

$$\mathcal{P}^T R^2 \mathcal{P} X = \mathcal{P}^T R^2 \mathcal{Y}, \tag{2.12}$$

where $\mathcal{P}^T$ is the adjoint operator of $\mathcal{P}$, given explicitly below in Lemma 2.1. Equation 2.12 is linear in X, and we intend to solve it using the conjugate gradients algorithm [15]. This algorithm requires the operator acting on the unknown vector to be symmetric and positive definite (SPD), as was shown above to hold in our case.

The conjugate gradients algorithm applies in each of its iterations the operator $\mathcal{P}^T R^2 \mathcal{P}$ (the operator on the left-hand-side of Eq. 2.12) to some vector. Since this operator consists of convolutions (see Eq. 2.10 and Lemma 2.1 below), we can efficiently apply it using the Fast Fourier Transform algorithm [4] using $O(n^2 \log^2 n)$ operations per iteration, where $n \times n$ in the size of the acquired images.

We next derive the explicit form of the operator $\mathcal{P}^T$ from Eq. 2.12, and show that it can be computed using convolutions.

Lemma 2.1. *Let $\mathcal{P}$ be the operator defined in Eq. 2.10. Its adjoint $\mathcal{P}^T$ is given by*

$$\mathcal{P}^T\mathcal{T} = \mathcal{P}^T \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{m+1} \end{pmatrix} = \Big(\sum_{k=1}^{m} T_k * \breve{P}_k \Big) + \sqrt{\nu} \cdot T_{m+1}, \tag{2.13}$$

where $[\breve{P}_k]_{i,j} := [P_k]_{-i,-j}$ is the k^{th} acquired PSF being flipped in both dimensions.

Proof. For a pair of images X and Y (indexed symmetrically around zero) of size $w \times z$, where w and z are odd positive integers, we define (assuming implicit zero padding whenever needed)

$$[X * Y]_{i,j} = \sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}} \sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}} X_{i-l,j-f} Y_{l,f}. \tag{2.14}$$

for $i \in \{-\frac{w-1}{2}, -\frac{w-1}{2}+1, \ldots, \frac{w-1}{2}\}$, and $j \in \{-\frac{z-1}{2}, -\frac{z-1}{2}+1, \ldots, \frac{z-1}{2}\}$. Using this indexing scheme, the product of X and Y as matrices is given by

$$[XY^T]_{i,j} = \sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}} X_{i,f} [Y^T]_{f,j}, \tag{2.15}$$

with same ranges for i and j as before. Note that this definition of matrix multiplication is consistent with the standard definition of matrix multiplication, but uses indexes that are symmetric around 0. With a slight abuse of notation (due to the use of negative indices), we denote $X, Y \in \mathbb{R}^{w \times z}$.

Next, we recall the definition of the Frobineus inner product. For a pair of matrices $\mathcal{X} = [X_1^T, X_2^T, \ldots, X_{m+1}^T]^T$ and $\mathcal{T} = [T_1^T, T_2^T, \ldots, T_{m+1}^T]^T$ with $X_i, T_i \in \mathbb{R}^{w \times z}, i \in \{1, \ldots, m+1\}$, we define

$$\langle \mathcal{X}, \mathcal{T} \rangle_F = \left\langle \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_{m+1} \end{pmatrix}, \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{m+1} \end{pmatrix} \right\rangle_F = \mathrm{Tr}(\mathcal{X}^T \mathcal{T}) = \sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} [\mathcal{X}^T \mathcal{T}]_{ii}. \tag{2.16}$$

Also, we can use the block structure of $\mathcal{X}$ and $\mathcal{T}$ in Eq. 2.16 to get

$$\begin{aligned} \langle \mathcal{X}, \mathcal{T} \rangle_F = \mathrm{Tr}(\mathcal{X}^T \mathcal{T}) &= \mathrm{Tr}\left(\begin{pmatrix} X_1^T & X_2^T & \cdots & X_{m+1}^T \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{m+1} \end{pmatrix} \right) \\ &= \mathrm{Tr}\left(\sum_{k=1}^{m+1} X_k^T T_k \right) = \sum_{k=1}^{m+1} \mathrm{Tr}(X_k^T T_k). \end{aligned} \tag{2.17}$$

Now, in order to prove that $\mathcal{P}^T$, given by Eq. 2.13, is the adjoint operator of $\mathcal{P}$, we need to show that for any $X \in \mathbb{R}^{w \times z}$ and any $\mathcal{T} = [T_1^T, \ldots, T_{m+1}^T]^T \in \mathbb{R}^{w \times z \times (m+1)}$, it holds that

$$\langle \mathcal{P}X, \mathcal{T} \rangle_F = \langle X, \mathcal{P}^T \mathcal{T} \rangle_F. \tag{2.18}$$

Using the definition of $\mathcal{P}X$ in Eq. 2.10 together with Eq. 2.17, we write

$$\langle \mathcal{P}X, \mathcal{T} \rangle_F = \left\langle \begin{pmatrix} P_1 * X \\ P_2 * X \\ \vdots \\ P_m * X \\ \sqrt{\nu}X \end{pmatrix}, \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_m \\ T_{m+1} \end{pmatrix} \right\rangle_F \tag{2.19}$$

$$= \sum_{k=1}^{m} \mathrm{Tr}\left((P_k * X)^T T_k\right) + \mathrm{Tr}(\sqrt{\nu} X^T T_{m+1}) \tag{2.20}$$

$$= \sum_{k=1}^{m} \sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} [(P_k * X)^T T_k]_{ii} + \sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} [\sqrt{\nu} X^T T_{m+1}]_{ii}, \tag{2.21}$$

where Eq. 2.20 follows from Eq. 2.17, and Eq. 2.21 follows from the definition of the trace function. For simplicity of notation, we won't carry the additive term $\sqrt{\nu}X^T \cdot T_{m+1}$ along the proof, but rather return to it at the end.

Expanding the product in Eq. 2.21, using Eq. 2.15 we get

$$\sum_{k=1}^{m} \left(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} [(P_k * X)^T T_k]_{ii} \right) = \sum_{k=1}^{m} \left(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} \left(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}} [(P_k * X)^T]_{ij} \cdot [T_k]_{ji} \right) \right). \tag{2.22}$$

Continuing with the right-hand-side of Eq. 2.22, expending the convolution term by using Eq. 2.14, we have

$$\begin{aligned} &\sum_{k=1}^{m} \left(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} \left(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}} [(P_k * X)^T]_{ij} \cdot [T_k]_{ji} \right) \right) = \\ &\sum_{k=1}^{m} \left(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}} \left(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}} \left(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}} \sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}} [P_k]_{j-l,i-f} \cdot X_{lf} \right) \cdot [T_k]_{ji} \right) \right). \end{aligned} \tag{2.23}$$

Since $X_{lf}, [P_k]_{j-l,i-f}$ and $[T_k]_{ji}$ are numbers, their multiplication is commutative. Switching their order and using the definition of $\breve{P}_k$ from Eq. 2.13, we get from

Eq. 2.23 that

$$\sum_{k=1}^{m}\Big(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}[P_k]_{j-l,i-f}\cdot X_{lf}\Big)\cdot[T_k]_{ji}\Big)\Big)=$$
$$\sum_{k=1}^{m}\Big(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot[\breve{P}_k]_{l-j,f-i}\Big)\cdot[T_k]_{ji}\Big)\Big). \tag{2.24}$$

Next, we switch the order of summation in Eq. 2.24, and by noting that

$$\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}}\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}}[\breve{P}_k]_{l-j,f-i}\cdot[T_k]_{ji}=[\breve{P}_k*T_k]_{lf},$$

we get

$$\sum_{k=1}^{m}\Big(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot[\breve{P}_k]_{l-j,f-i}\Big)\cdot[T_k]_{ji}\Big)\Big)=$$
$$\sum_{k=1}^{m}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot\Big(\sum_{i=-\frac{z-1}{2}}^{\frac{z-1}{2}}\sum_{j=-\frac{w-1}{2}}^{\frac{w-1}{2}}[\breve{P}_k]_{l-j,f-i}\cdot[T_k]_{ji}\Big)\Big)\Big)$$
$$\sum_{k=1}^{m}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot[\breve{P}_k*T_k]_{lf}\Big)\Big). \tag{2.25}$$

The innermost sum in Eq 2.25 is written using matrix multiplication (Eq. 2.15) as

$$\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot[\breve{P}_k*T_k]_{lf}=\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X^T_{fl}\cdot[\breve{P}_k*T_k]_{lf}=[X^T(\breve{P}_k*T_k)]_{ff},$$

and therefore, Eq. 2.25 becomes

$$\sum_{k=1}^{m}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big(\sum_{l=-\frac{w-1}{2}}^{\frac{w-1}{2}}X_{lf}\cdot[\breve{P}_k * T_k]_{lf}\Big)\Big)=\sum_{k=1}^{m}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T(\breve{P}_k * T_k)]_{ff}\Big). \quad (2.26)$$

Since X is independent of k, we can switch the order of summation once again to get

$$\sum_{k=1}^{m}\Big(\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T(\breve{P}_k * T_k)]_{ff}\Big)=\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T\Big(\sum_{k=1}^{m}\breve{P}_k * T_k\Big)]_{ff}. \quad (2.27)$$

Finally, we bring back the additive term $\sqrt{\nu}X^T\cdot T_{m+1}$ from Eq. 2.20, which could have been carried throughout the entire proof without affecting it. This results in

$$\begin{aligned}&\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T\Big(\sum_{k=1}^{m}\breve{P}_k * T_k\Big)]_{ff}+\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[\sqrt{\nu}X^T\cdot T_{m+1}]_{ff}=\\&\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}\Big([X^T(\sum_{k=1}^{m}\breve{P}_k * T_k)]_{ff}+[\sqrt{\nu}X^T\cdot T_{m+1}]_{ff}\Big)=\\&\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T\Big(\sum_{k=1}^{m}\breve{P}_k * T_k+\sqrt{\nu}T_{m+1}\Big)]_{ff},\end{aligned} \quad (2.28)$$

Recalling the definition of $\mathcal{P}^T$ from Eq. 2.13, $\mathcal{P}^T=\sum_{k=1}^{m}\breve{P}_k * T_k+\sqrt{\nu}T_{m+1}$, we get from Eq. 2.28 that

$$\begin{aligned}\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T\Big(\sum_{k=1}^{m}\breve{P}_k * T_k+\sqrt{\nu}T_{m+1}\Big)]_{ff}&=\sum_{f=-\frac{z-1}{2}}^{\frac{z-1}{2}}[X^T\big(\mathcal{P}^T\mathcal{T}\big)]_{ff}\\&=\mathrm{Tr}\left(X^T(\mathcal{P}^T\mathcal{T})\right)=\langle X,\mathcal{P}^T\mathcal{T}\rangle_F.\end{aligned} \quad (2.29)$$

Summing up Eqs. 2.19–2.29 gives us the equality in Eq. 2.18 as required.

□

Chapter 3

Experiments

In this chapter, we demonstrate the performance of our proposed algorithm using both simulated data, where scattering patterns are simulated according to a known model, as well as using experimental SAXS measurements. For all datasets (simulated and experimental), we apply two versions of the proposed algorithm. First, we apply the algorithm proposed in Chapter 2 as is, and second, we apply a super-resolution version of it, as described in Chapter 1 and explained again below. We apply both algorithms (with and without super-resolution) in order to evaluate whether our proposed algorithm has advantages on its own, and whether combining it with the super-resolution procedure further improves the results.

In Chapter 1, we have proposed cascading our algorithm with an "off the shelf" super-resolution algorithm (described in [36]). In the super-resolution algorithm, we take multiple images for each given PSF, while moving the detector in sub-pixel translations. The advantage of this algorithm is demonstrated in Fig. 1.2. Throughout our subsequent experiments, we apply sub-pixel translations of 1/3 pixel size, and use the super-resolution algorithm to combine the resulting 9 images into a single high-resolution image. Performing the above for each PSF separately results in several high resolution images, each corresponding to a different PSF. These high resolution images are the input to the algorithm proposed in Chapter 2.

In order to distinguish between the two different versions of the proposed algorithm (with and without super-resolution preprocessing), we label them as follows.

The algorithm described in Chapter 2 (without super-resolution preprocessing) is named 'Constrained Multi Deconvolution' (CMD), since the deconvolution is applied to a set of scattering patterns in which all of them are constrained by the same underlying scattering pattern X. The version where a super-resolution algorithm is applied prior to the CMD algorithm is named 'Super-Resolution SAXS' (srSAXS). For comparison, we also apply to the input data the super-resolution algorithm [36] without performing any further deconvolution. Such results are labelled 'SPS' for Sub-Pixel Sampling.

Before presenting the results of our algorithm, we first present in Section 3.1 the measures used to evaluate its performance. Then, we present simulation results in Section 3.2, and finally, we present results for experimental SAXS data in Section 3.3.

3.1 Performance evaluation

We use several measures to assess the quality of an estimated scattering pattern. If the underlying scattering pattern is known (X in Eq. 2.1), it is possible to compare the recovered scattering pattern $\bar{X}$ to X using the Frobenius norm

$$\varepsilon = \frac{||\bar{X} - X||_F}{||X||_F}. \tag{3.1}$$

However, for cases where X is unknown (for example, when applying our algorithms to experimental SAXS data), a different measure is required. To develop such a measure, we first lay some additional assumptions on X.

As previously discussed, many scattering patterns have radial symmetry, and therefore, we only need to recover their one-dimensional radial profile. To compute the radial profile of a scattering pattern, we first estimate the center of the latter using its two-dimensional auto-correlation. Then, we compute the one-dimensional radial profile of the scattering pattern using azimuthal integration around its center as follows. Denote by $(i_c, j_c) \in \{-\frac{w-1}{2}, \ldots, \frac{w-1}{2} \times -\frac{z-1}{2}, \ldots, \frac{z-1}{2}\}$ the pixel which is the estimated center of the scattering pattern. Then, the radial profile $I_X[r]$ of X is

given by

$$I_X[r] = |Q_r|^{-1} \sum_{p \in Q_r} X[p], \qquad r \in \{1, 1+s, 1+2s, \cdots, r_{max}\},$$
$$Q_r = \Big\{(i,j) \mid i \in \{-\frac{w-1}{2}, \ldots, \frac{w-1}{2}\},\ j \in \{-\frac{z-1}{2}, \ldots, \frac{z-1}{2}\}, \tag{3.2}$$
$$(i-i_c)^2 + (j-j_c)^2 \in ((r-s)^2, r^2]\Big\},$$

where $X \in \mathbb{R}^{w \times z}$ is the scattering pattern, s is the sampling interval of $I_X[r]$, set to $1/2$ in the remaining of the chapter, and r_{max} is set to $\min(i_c, j_c, w - i_c, z - j_c)$, representing the minimal distance of the center to the edges of the scattering pattern X.

The radial profile representation of a scattering pattern is very common in SAXS data processing [3, 14, 16, 18, 19, 23, 25, 26, 32, 41], as it is less noisy compared to the two-dimensional scattering pattern. The reason for the reduced noise is that while the signal in a two-dimensional scattering pattern has azimuthal symmetry, the noise in it is spread without any symmetry, and so by averaging in the azimuthal direction, the noise is being suppressed following the Central Limit Theorem.

Typically, $I_X[r]$ is characterized by several extrema points, whose magnitude decreases with increasing r [19, 23]. We use this fact to define a quality measure of a scattering pattern X using its radial profile $I_X[r]$, and denote this measure by ζ. In the following, we refer to local maxima as 'peaks', and to local minima as 'valleys'. Let X be a scattering pattern with radial profile $I_X[r]$, and let $\{I_1, \ldots, I_k\}$ be the heights of the peaks of $I_X[r]$, with corresponding radii $\{r_1, \ldots, r_k\}$. If $I_X[r]$ has a single peak (i.e. $k = 1$), we define our quality measure to be 0, that is, $\zeta = 0$. Else, we label the maximum over $\{I_1, \ldots, I_k\}$ by I_{p_1}, and its corresponding radius by r_{p_1}. Next, we define

$$r_{p_2} = \underset{\substack{i \in \{1,\ldots,k\} \\ r_i \neq r_{p_1}}}{\operatorname{argmin}} |r_i - r_{p_1}|,$$

and denote the height of the peak that corresponds to r_{p_2} by I_{p_2}. Since $I_X[r]$ is continuous, there must exist a valley between the two peaks I_{p_1} and I_{p_2}. We label

the value of $I_X[r]$ at this valley by I_v, and the radius of the valley by r_v. Finally we define the quality measure by

$$\zeta_X = \frac{I_{p_2} - I_v}{|r_{p_2} - r_v|}. \tag{3.3}$$

Note that $\zeta \geq 0$, since the numerator is positive (the peak is higher than the valley), and the denominator is an absolute value.

We demonstrate the intuition behind this measure using a toy example. Let X be a scattering pattern with a corresponding radial profile I_X, and suppose that $I_X[r]$ has only one pair of peaks with $I_{p_1} = I_{p_2} = 1$, with $|r_{p_1} - r_{p_2}| = 4$. Since $I_X[r]$ is continuous, it has a single valley, and we further assume that $I_v = 0.4$ and that $|r_{p_2} - r_v| = 2$. The quality measure of $I_X[r]$ is then calculated using Eq. 3.3 to be $\zeta_X = \frac{1-0.4}{2} = 3/10$. Figure 3.1 presents such a simulated $I_X[r]$ as the blue curve, with X shown in the inset. The exact model used to simulate X is described later, and is of no relevance for now. Next, let Y_1 and Y_2 be blurred scattering patterns that are generated from X according to Eq. 2.1. The PSFs that are used to generate Y_1 and Y_2 are

$$P_1 = \frac{1}{18}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \qquad P_2 = \frac{1}{8}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \tag{3.4}$$

respectively, where the coefficients $\frac{1}{18}$ and $\frac{1}{8}$ are discussed later. We present the corresponding radial profiles $I_{Y_1}[r]$ and $I_{Y_2}[r]$ as the red and yellow curves in Fig. 3.1, respectively. One can notice that the two separate peaks in $I_X[r]$ completely vanish in $I_{Y_1}[r]$, and hence the quality measure of $I_{Y_1}[r]$ is $\zeta_{Y_1} = 0$. As of $I_{Y_2}[r]$, it has two peaks with a valley between them, which is shallower than that of I_X. The quality measure of $I_{Y_2}[r]$ is $\zeta_{Y_2} = 0.09$, which is lower than that of I_X.

In order to calculate the quality measure of Eq. 3.3, we implement the following

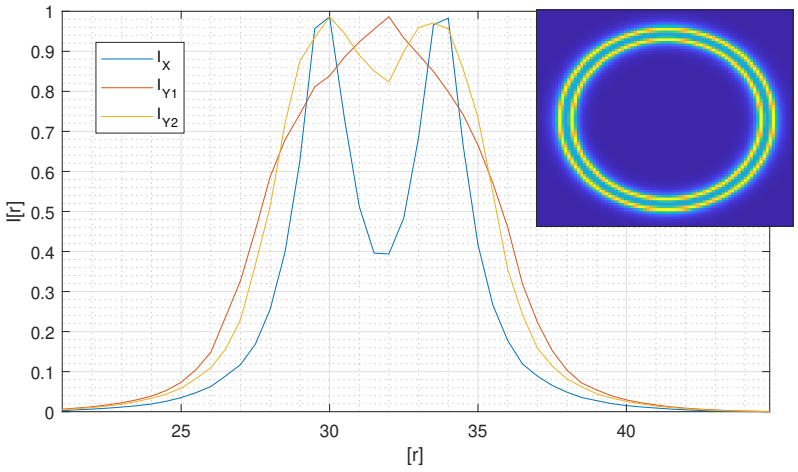

Figure 3.1: Radial profile demonstrating ζ of Eq. 3.3. Blue curve shows I_X, with good separation between the peaks and with $\zeta_X = 3/10$; red curve shows I_{Y_1} with no separation between the peaks, corresponding to $\zeta_{Y_1} = 0$; yellow curve shows I_{Y_2}, with poor separation between the peaks, and with corresponding $\zeta_{Y_2} = 0.09$. The inset shows the original scattering pattern X.

algorithmt. Let X be some scattering pattern. We first compute its radial profile $I_X[r]$ using Eq. 3.2. Next, we normalize $I_X[r]$ such that it's highest peak is equal to 1. This normalization assures that all radial profiles are of the same scale, and results in values of ζ which can be compared across different PSFs (as different PSFs might result in different scalings of $I_X[r]$). It also enables us to neglect the coefficients of P_1 and P_2 in Eq. 3.4 that where used for the same reason. In order to find the number of peaks in $I_X[r]$, we use Matlab's `findpeaks` function and set `MinPeakHeight= 0.3` to avoid spurious peaks due to noise. The `findpeaks` function results in peaks' values $\{I_1, \ldots, I_k\}$ and their corresponding radii $\{r_1, \ldots, r_k\}$. If the number of peaks is 1, we set $\zeta = 0$. Else, we find the maximal peak using Matlab's `max` function, and label the highest peak by I_{p_1} and it's corresponding r value by r_{p_1}. In order to find the closest peak to I_{p_1} among $\{I_1, \ldots, I_k\}$, we use `min(abs(`$r_i - r_{p_1}$`))` on the set $\{r_1, \ldots, r_k\}$, with r_{p_1} excluded from the set. The radius corresponding to the minimum of the latter expression is labeled r_{p_2} and the corresponding I_i is labeled I_{p_2}. Now we are at a position of finding the valley between the peaks. Having r_{p_1}

and r_{p_2} at hand, we find the minimal value over all $I_X[r]$ for $r \in \{r_{p_1}, \ldots, r_{p_2}\}$, using Matlab's `min` function, label it by I_v and the corresponding radius by r_v. Finally, we calculate ζ using Eq. 3.3.

3.2 Simulations

In this section, we first demonstrate our algorithm using simulated data, which follows the model in Eq. 2.1, that is,

$$Y_i = X * P_i + N_i.$$

We next describe how we simulate the "ground-truth" signal X, the PSFs P_i, and the noise N_i. We assume that the underlying signal X has radial symmetry (as discussed above), and model it as a linear combination of Cauchy distributions [1], as these are common in modeling scattering patterns [42]. Specifically, our ground-truth scattering pattern is given in polar coordinates by

$$X(r) = \sum_{i=1}^{2} I_i(r|r_0^i, \gamma^i, D^i), \qquad I_i(r|r_0^i, \gamma^i, D^i) = D^i \Big[\frac{(\gamma^i)^2}{(r - r_0^i)^2 + (\gamma^i)^2} \Big], \tag{3.5}$$

where D^i controls the height of the peak of I_i, r_0^i controls the location of the peak, and γ^i determines its width (see Fig 3.2). Note that in order to fix the height of the peak of $I_i(r|r_0^i, \gamma^i, D^i)$ to D^i, we use a different normalization than the traditional normalization of Cauchy distributions [1]. Our simulated ground-truth scattering pattern is obtained by sampling $X(r)$ of Eq. 3.5 on a discrete grid, with $r = 0$ corresponding to the center of the discrete grid. We denote the resulting discrete scattering pattern by X_{GT}. The parameters used for the different terms I_i in Eq. 3.5 are given in Table 3.1. To simulate the input to the super-resolution algorithm, we translate each term $I_i(r|r_0^i, \gamma^i, D^i)$ in Eq. 3.5 by $\{-1/3, 0, 1/3\}$ pixels in each dimension prior to sampling. This results in 8 additional ground-truth scattering patterns whose centers are displaced by 1/3 pixel in each dimension.

Next, we simulate the PSFs P_i. We describe the derivation of a one-dimensional

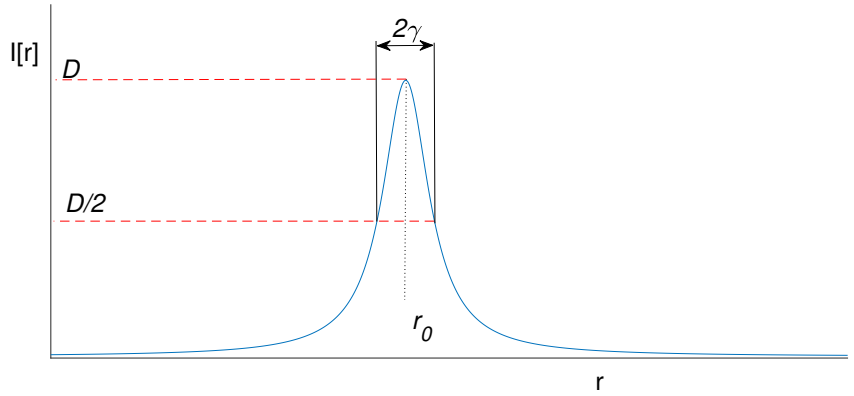

Figure 3.2: Cauchy distribution $I(r)$ used to generate the ground-truth scattering pattern X_{GT}. Following the notation in Eq. 3.5, D controls the height of the peak, r_0 its location, and γ its width (at $I(r) = D/2$).

γ	1	1
r_0	29	35
D	1	0.5

Table 3.1: Parameters used to generate X_{GT} using Eq. 3.5.

PSF, as its extension to two dimensions is trivial. Having a pinhole-like slit, it is common to assume its resulting PSF to be Gaussian [50], given by

$$P(x) = \frac{1}{\sqrt{\pi}\beta} \exp\Big(- \frac{x^2}{\beta^2}\Big), \tag{3.6}$$

where $P(x)$ is the intensity measured on a continuous detector, $x \in \mathbb{R}$ is the distance on the detector from the center of the beam, and $\beta > 0$ is a parameter that is referred to as the 'sharpness' of the PSF, as it controls the steepness of the Gaussian; since $\max(\frac{dP(x)}{dx}) = \sqrt{\frac{2}{\pi\beta^4}} \exp(-1/2)$, the smaller β the sharper the incline of $P(x)$ is.

However, the model in Eq. 3.6 is not sufficient for our setup, since our slit is not pinhole-like. As described in Chapter 1, we control the shape of the slit in order to produce a set of measured scattering patterns Y_i (Eq. 2.1) out of a single X, and hence, we need to simulate the PSFs P_is for rectangular slits of varying sizes.

Although the PSF of a system is defined as the measured signal on the detector for light arriving from a single point, we "abuse" this term and name all the following P_is of Eq. 2.1 as PSFs.

In the following, we derive the expression for the PSF of a given rectangular slit. Let Ł be a linear, shift-invariant operator that returns the PSF of a given slit. We derive an explicit formula for Ł for the case of a rectangular input slit. We start by deriving Ł for a pinhole-like slit. Let us model a pinhole-like slit by Dirac's delta function $\delta(x)$, that is defined to be 0 whenever $x \neq 0$ and by the identity

$$\int_{\mathbb{R}} \delta(x) f(x) dx = f(0), \tag{3.7}$$

for any function f that is continuous at the origin [29]. Since Eq. 3.6 specifies the PSF for a pinhole-like slit, we can write

$$Ł[\delta(x)] = \frac{1}{\sqrt{\pi}\beta} \exp\Big(-\frac{x^2}{\beta^2}\Big). \tag{3.8}$$

In order to find the PSF of a rectangular slit, we first model the rectangular slit by the rectangle function $\Pi(x/2\alpha)$, that is defined to be 1 when $|x| < \alpha$ and 0 otherwise [47]. Using Eq. 3.7, we deduce ([29]) that

$$f(x) = \int_{\mathbb{R}} f(u)\delta(x-u)du,$$

and specifically, for the rectangle function $\Pi(x/2\alpha)$ we have that

$$\Pi\Big(\frac{x}{2\alpha}\Big) = \int_{\mathbb{R}} \Pi\Big(\frac{u}{2\alpha}\Big)\delta(x-u)du. \tag{3.9}$$

Using the linearity of Ł together with Eqs. 3.8 and 3.9 we get

$$\begin{aligned}
Ł\Big[\Pi\Big(\frac{x}{2\alpha}\Big)\Big] &= Ł\bigg[\int_{\mathbb{R}} \Pi\Big(\frac{u}{2\alpha}\Big)\delta(x-u)du\bigg] \\
&= \int_{\mathbb{R}} \Pi\Big(\frac{u}{2\alpha}\Big) Ł\Big[\delta(x-u)\Big]du, \\
&= \int_{\mathbb{R}} \Pi\Big(\frac{u}{2\alpha}\Big)\frac{1}{\sqrt{\pi}\beta}\exp\Big(-\frac{(x-u)^2}{\beta^2}\Big)du. \qquad (3.10)
\end{aligned}$$

In order to get an explicit expression for $Ł\Big[\Pi\Big(\frac{x}{2\alpha}\Big)\Big]$, we change variable $u \to x - t$ in Eq. 3.10 to get

$$\begin{aligned}
Ł\Big[\Pi\Big(\frac{x}{2\alpha}\Big)\Big] &= \int_{\mathbb{R}} \Pi\Big(\frac{u}{2\alpha}\Big)\frac{1}{\sqrt{\pi}\beta}\exp\Big(-\frac{(x-u)^2}{\beta^2}\Big)du \\
&= \frac{1}{\sqrt{\pi}\beta}\int_{\infty}^{-\infty} \Pi\Big(\frac{x-t}{2\alpha}\Big)\exp\Big(-\frac{t^2}{\beta^2}\Big)d(-t) \\
&= \frac{1}{\sqrt{\pi}\beta}\int_{-\infty}^{\infty} \Pi\Big(\frac{x-t}{2\alpha}\Big)\exp\Big(-\frac{t^2}{\beta^2}\Big)dt \\
&= \frac{1}{\sqrt{\pi}\beta}\int_{x-\alpha}^{\alpha+x} \exp\Big(-\frac{t^2}{\beta^2}\Big)dt, \qquad (3.11)
\end{aligned}$$

where Eq. 3.11 follows since $\Pi\Big(\frac{x-u}{2\alpha}\Big) = 1$ whenever $|x-t| < \alpha$ and 0 otherwise. Thus, we get

$$\begin{aligned}
Ł\Big[\Pi\Big(\frac{x}{2\alpha}\Big)\Big] &= \frac{1}{\sqrt{\pi}\beta}\int_{x-\alpha}^{\alpha+x} \exp\Big(-\frac{t^2}{\beta^2}\Big)dt \\
&= \frac{1}{\sqrt{\pi}\beta}\bigg[\int_{x-\alpha}^{0} \exp\Big(-\frac{t^2}{\beta^2}\Big)dt + \int_{0}^{x+\alpha} \exp\Big(-\frac{t^2}{\beta^2}\Big)dt\bigg] \\
&= \frac{1}{\sqrt{\pi}\beta}\bigg[-\int_{0}^{\alpha-x} \exp\Big(-\frac{(-u)^2}{\beta^2}\Big)d(-u) + \int_{0}^{x+\alpha} \exp\Big(-\frac{t^2}{\beta^2}\Big)dt\bigg] \\
&= \frac{1}{\sqrt{\pi}\beta}\bigg[\int_{0}^{\alpha-x} \exp\Big(-\frac{u^2}{\beta^2}\Big)du + \int_{0}^{x+\alpha} \exp\Big(-\frac{u^2}{\beta^2}\Big)du\bigg]. \qquad (3.12)
\end{aligned}$$

label	1	2	3	4
α	1.0	1.5	2.0	2.5
β	0.1	1.6	3.1	4.6

Table 3.2: Parameters used to generate simulated PSFs using Eq. 3.13. The first row is used to label the different PSFs.

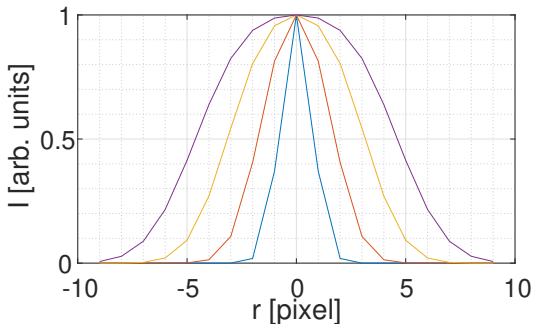

Figure 3.3: Different 1D PSFs used to generate 2D PSFs. Blue curve $(\alpha, \beta) = (1.0, 0.1)$; red curve $(\alpha, \beta) = (1.5, 1.6)$; yellow curve $(\alpha, \beta) = (2.0, 3.1)$; purple curve $(\alpha, \beta) = (2.5, 4.6)$.

Finally, we use the change of variable $u \to \beta t$ in Eq. 3.12 to conclude that

$$\begin{aligned} Ł\Big[\Pi\Big(\frac{x}{2\alpha}\Big)\Big] &= \frac{1}{\sqrt{\pi}\beta}\left[\int_0^{\alpha-x} \exp\Big(-\frac{u^2}{\beta^2}\Big)du + \int_0^{x+\alpha} \exp\Big(-\frac{u^2}{\beta^2}\Big)du\right] \\ &= \frac{1}{\sqrt{\pi}\beta}\left[\int_0^{\frac{\alpha-x}{\beta}} \exp\Big(-t^2\Big)\beta dt + \int_0^{\frac{x+\alpha}{\beta}} \exp\Big(-t^2\Big)\beta dt\right] \\ &= \frac{1}{2}\Big[\operatorname{erf}\Big(\frac{\alpha-x}{\beta}\Big) + \operatorname{erf}\Big(\frac{\alpha+x}{\beta}\Big)\Big], \end{aligned} \tag{3.13}$$

where $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x \exp(-t^2)dt$ is the error function.

Throughout this section, we use Eq. 3.13 to simulate 1D PSFs by sampling $x \in [-8, 8]$ at 19 equally spaced points, using the parameters α and β specified in Table 3.2. The resulting 1D PSFs are presented in Fig 3.3. In order to simulate a 2D PSF, two 1D PSFs are applied in the x and y directions independently, resulting in 16 different 2D PSFs.

Summing up the above, in order to simulate scattering patterns Y_i, we use Eq. 2.1 to form

$$Y_i = X_{GT} * P_i + N_i \tag{3.14}$$

for $i \in \{1, ..., 16\}$, where X_{GT} is the underlying scattering pattern generated using Eq. 3.5, and N_i consists of i.i.d Gaussian random samples with mean 0 and variance 1/4.

We next apply the CMD algorithm to our simulated data. Recall that the CMD algorithm solves (see Eq. 2.7)

$$\bar{X} = \underset{X}{\operatorname{argmin}} \sum_i \frac{||Y_i - P_i * X||_F^2}{2\sigma_i^2} + \frac{\nu}{2}||X||_F^2.$$

We determine σ_i (the weight corresponding to Y_i) and the regularization parameter ν as follows. Recalling that $Y_i \in \mathbb{R}^{w \times z}$, we define

$$S_i = \mathbb{1}_z^T Y_i \mathbb{1}_w, \qquad \text{where } \mathbb{1}_n = [\underbrace{1, 1, \ldots, 1}_{n}]^T, \tag{3.15}$$

that is S_i is the total intensity of Y_i, and set

$$\sigma_i = \exp\Big(- \frac{(S_i - \mu\bar{S})^2}{(\psi\bar{S})^2}\Big), \qquad \bar{S} = \frac{1}{16}\sum_{i=1}^{16} S_i, \tag{3.16}$$

where 16 is the number of input scattering patterns Y_i used by the CMD algorithm. We set $\psi = 3/10$, $\mu = 3/10$, and discuss this choice later in this section. The regularization parameter ν is set to 0.05. The method of choosing σ_i and ν is discussed later in this section. Using these parameters, we ran the CMD algorithm on the simulated scattering patterns Y_i, and calculated the quality of the recovered scattering pattern $\bar{X}$ to be $\zeta_{\bar{X}} = 0.13$ (using Eq. 3.3). For comparison, we calculated the quality of two additional simulated scattering patterns. The ground truth scattering pattern X_{GT} that was used to simulate the scattering patterns Y_i has $\zeta_{X_{GT}} = 0.17$. The scattering pattern Y^* that has the largest ζ among all Y_is has $\zeta_{Y^*} = 0.12$. The

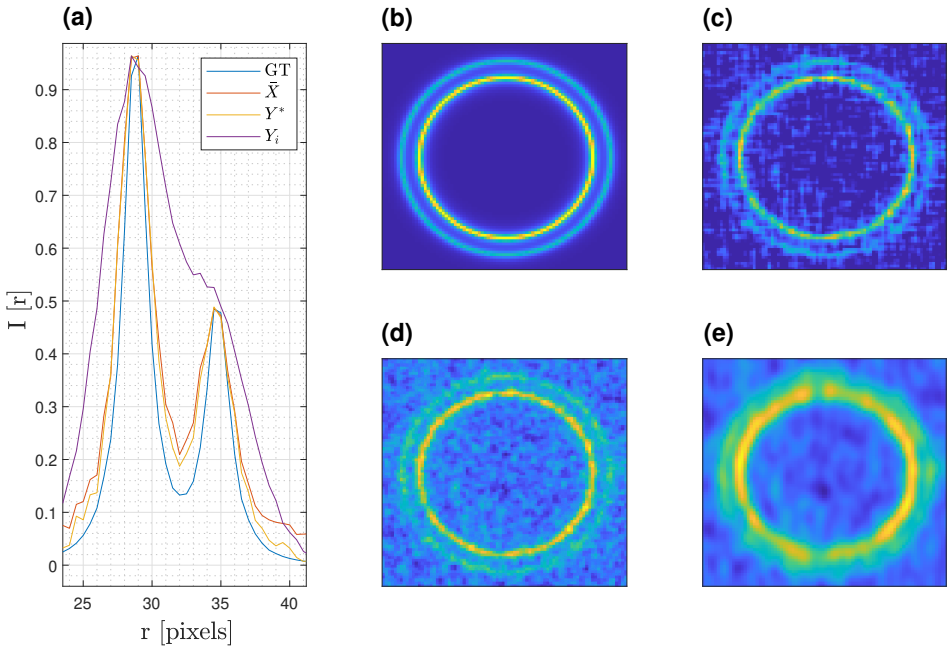

Figure 3.4: Results of the CMD algorithm on simulated data with Gaussian noise. (a) Radial profiles of (b)–(e). (b) Ground truth scattering pattern (X_{GT} in Eq. 3.14). (c) Scattering pattern recovered by the CMD algorithm, $\bar{X}$. (d) The scattering pattern Y^* that has the largest ζ among all Y_is. (e) Some other scattering pattern Y_i. One can notice that unlike the other curves in (a), the radial profile that corresponds to this Y_i (purple curve in (a)) exhibits no local minimum.

results of this experiment are summarized in Fig. 3.4.

These results are encouraging, since the value of $\zeta_{\bar{X}}$ for the scattering pattern estimated by the CMD algorithm exceeds that of Y^*, implying that using the CMD algorithm is advantageous. To further evaluate the performance of our algorithm, we would like to deviate from our noise model described in Chapter 1, and test the CMD algorithm under a more realistic model. As we have pointed out in Chapter 1, we are using Gaussian statistics as an approximation of a Poisson process taking place at each pixel of the detector. To get a more realistic model, we will generate our data Y_i using a simulation of this Poisson process. This will also allow us to take into account the exposure time t described in Chapter 1, which was not taken into

account so far.

To define our improved model, we start by generating a realization of X_{GT} by

$$X_t[i,j] = \texttt{poissrnd}(t \cdot X_{GT}[i,j]), \tag{3.17}$$

where `poissrnd` is Matlab's function that realizes Poisson random variables, and t simulates the exposure time. While using this model for X_{GT} gives a more realistic noise model whenever a signal exists (when $X_{GT} > 0$), it reduces the noise to zero when the signal vanishes. This happens since $P(Poiss(0) = 0) = 1$, where $Poiss(\xi)$ is a Poisson random variable with parameter ξ [1], so using Eq. 3.17 gives $X_t[i,j] = 0$ whenever $X_{GT}[i,j] = 0$, independent of the exposure time t. Since this behavior is physically false, we introduce into our model additional background noise (N in Eq. 2.1). In order to do so, we start by defining a radial profile

$$I_{BKG}[r] = \begin{cases} Ar^{-B} & r > r_{min}, \\ 0 & \text{otherwise}, \end{cases} \tag{3.18}$$

where this form of $I_{BKG}[r]$ and the choice of A, B and r_{min} are discussed in the following. Next, we construct a matrix N_{BKG} out of $I_{BKG}[r]$ where $N_{BKG}[i,j] = I_{BKG}[\sqrt{i^2+j^2}]$, and finally, we use Eq. 3.17 to generate a Poisson realization of N_{BKG} given by $N_t[i,j] = \texttt{poissrnd}(t \cdot N_{BKG}[i,j])$.

In order to determine the functional form of I_{BKG} in Eq. 3.18, we used experimental scattering patterns Y_i (described later in Section 3.3), and specifically, their corresponding radial profiles $I_{Y_i}[r]$, calculated using Eq. 3.2. Recall that any such radial profile consists of peaks and valley (as discussed above Eq. 3.3, and demonstrated in Fig. 3.5). Having a radial profile $I_{Y_i}[r]$, we manually picked 2 points at each valley, for a total of 16 points, and fitted this set of points to the model of Eq. 3.18 using least squares fitting. Few I_{Y_i}s are presented in Fig. 3.5 (solid lines) together with the fitted functions I_{BKG} (dashed lines). The values $A = 3$ and $B = 0.3$ were chosen as the median over all As and Bs found for the experimental data. As one can notice, defining $I_{BKG}[r] = Ar^{-B}$ results in a singularity for $r \to 0$, and in order

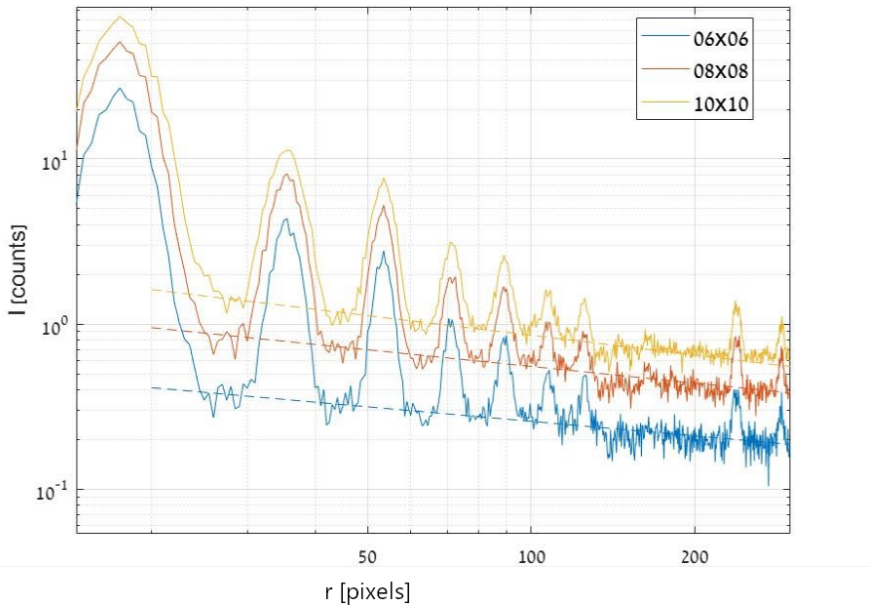

Figure 3.5: Radial profiles of measured AgBh samples corresponding to different PSFs, used to fit I_{BKG} of Eq. 3.18. The presented Y_is correspond to slit sizes of $0.6\ mm \times 0.6\ mm, 0.8\ mm \times 0.8\ mm$ and $1\ mm \times 1\ mm$. Dashed lines are the fitted I_{BKG} corresponding to $A = 0.99,\ 2.60,\ 5.30$ and $B = 0.29,\ 0.33, 0.39$, respectively.

to cope with it, we set $I_{BKG}[r < r_{min}] = 0$, where r_{min} was set to 20.

Summing up the above, in order to simulate a "measured" scattering pattern Y_i, we generate a ground truth scattering pattern X_{GT}, set an exposure time t, and use Eq. 3.17 to generate a scattering pattern X_t. Then, we generate N_{BKG} using $I_{BKG}[r]$ of Eq. 3.18, and its Poisson realization N_t using the same t as for X_t. Next, we simulate the different PSFs P_i by using the parameters of Table 3.2 in Eq. 3.13, as explained above, and use Eq. 2.1 to form

$$Y_i = [X_t]_i * P_i + [N_t]_i, \tag{3.19}$$

where $[X_t]_i$ and $[N_t]_i$ are X_t and N_t created separately for each i. In order to simulate input scattering patterns for the srSAXS algorithm, we modify X_{GT} as follows. So

far, $X_{GT}[i,j]$ was defied by $X_{GT}[i,j] = I_{X_{GT}}[\sqrt{i^2+j^2}]$, so in order to simulate sub-pixel translations, we set

$$X^k_{GT}[i,j] = I_{X_{GT}}[\sqrt{(i+\eta_k)^2+(j+\rho_k)^2}], \tag{3.20}$$

where $\eta_k, \rho_k \in \{\pm\frac{1}{3}, 0\}$, $k \in \{1,\ldots,9\}$. One can notice that by setting $\eta_1 = \rho_1 = 0$, we get X_{GT}, and by using 8 additional distinct pairs of η_k and ρ_k, we simulate the required sub-pixel translations. Next, we use Eq. 3.17 to generate a Poisson realization of X^k_{GT}, denoted X^k_t. The noise term N^k_{BKG} is simulated out of I_{BKG} (Eq. 3.18) in the same manner as Eq. 3.20, (9 different times with sub-pixel translations), and Eq. 3.17 is used to generate the Poisson realization N^k_t. Finally, a set of Y_is for the srSAXS algorithm is simulated by

$$Y^k_i = [X_t]^k_i * P_i + [N_t]^k_i,$$

where $k \in \{1,\ldots,9\}$ for the different sub-pixel translation, and $i \in \{1,\ldots,16\}$ for the different PSFs.

Using the simulated data sets, we are now at a position of tuning the various parameters of the CMD algorithm:

- the weight σ_i for each Y_i (see Eq. 2.7),
- the regularization parameter ν (see Eq. 2.7),
- the exposure time t of the simulation (see Eq. 3.17),
- number of Y_is.

In the following, we explain how these parameters were optimized to yield the best recovered scattering pattern $\bar{X}$ (Eq. 2.7), in terms of the largest $\zeta_{\bar{X}}$ (Eq. 3.3) or smallest ε (Eq. 3.1). We emphasize that the following optimization process should be applied to each set of Y_is separately.

We begin by recalling Eq. 2.7

$$\bar{X} = \underset{X}{\operatorname{argmin}} \sum_i \frac{||Y_i - P_i * X||_F^2}{2\sigma_i^2} + \frac{\nu}{2}||X||_F^2,$$

where σ_i and ν are parameters to be tuned. In order to demonstrate the relation between the parameters σ_i and ν and the quality measure $\zeta_{\bar{X}}$ of the recovered scattering pattern, we simulate a data set $\{Y_i\}_{i=1}^{16}$ using Eq. 3.19, where X_{GT} is given by Eq. 3.5 using the parameters in Table 3.1, and the set of PSFs is given by Eq. 3.13 using the parameters given in Table 3.2. Recalling that the total intensity of Y_i is S_i (Eq. 3.15), we repeat here Eq. 3.16, which defines the σ_is as

$$\sigma_i = \exp\left(- \frac{(S_i - \mu\bar{S})^2}{(\psi\bar{S})^2} \right), \qquad \bar{S} = \frac{1}{16}\sum_{j=1}^{16} S_j, \tag{3.21}$$

where 16 is the number of Y_is in the data set, and ψ and μ are parameters to be set. Three examples of sets $\{\sigma_i\}_{i=1}^{16}$ are presented in Fig. 3.6a for $\psi = 0.3; \mu = 0.3$ (blue curve), $\psi = 0.3; \mu = 1$ (red curve), and $\psi = 0.3; \mu = 3$ (green curve). In order to understand the relation between μ, ψ and ν and the resulting $\zeta_{\bar{X}}$, we repeatedly apply the CMD algorithm to the simulated data set $\{Y_i\}_{i=1}^{16}$, while using different values of μ, ψ and ν for each run of the CMD algorithm. This is done using logaritmicaly spaced values of $\mu \in [0.1, 3]$, $\psi \in [0.1, 1]$, and $\nu \in [10^{-5}, 0]$. As X_t and N_t that are used to simulate the data set $\{Y_i\}_{i=1}^{16}$, are realizations of Poisson random variables, we randomize 10 different X_ts and N_ts, simulate 10 corresponding data sets $\{Y_i\}_{i=1}^{16}$ (one for each X_t and N_t), and perform the above-mentioned procedure to each $\{Y_i\}_{i=1}^{16}$ independently. At the end of each run of the CMD algorithm, the quality measure $\zeta_{\bar{X}}$ of the recovered scattering pattern is calculated using Eq. 3.3. For three different sets of σ_i (with $\psi = 0.3$ and $\mu \in \{0.3, 1, 3\}$), we present in Fig. 3.6b the dependence of $\zeta_{\bar{X}}$ on ν, where the curve represents the mean ζ over the 10 experiments, and the error bars show the standard deviation of ζ over the 10 experiments. Having a grid of the parameters μ, ψ and ν, where each point on the grid is assigned with 10 different calculated ζs, we calculate the mean over the 10 ζs, and the best performing triplet

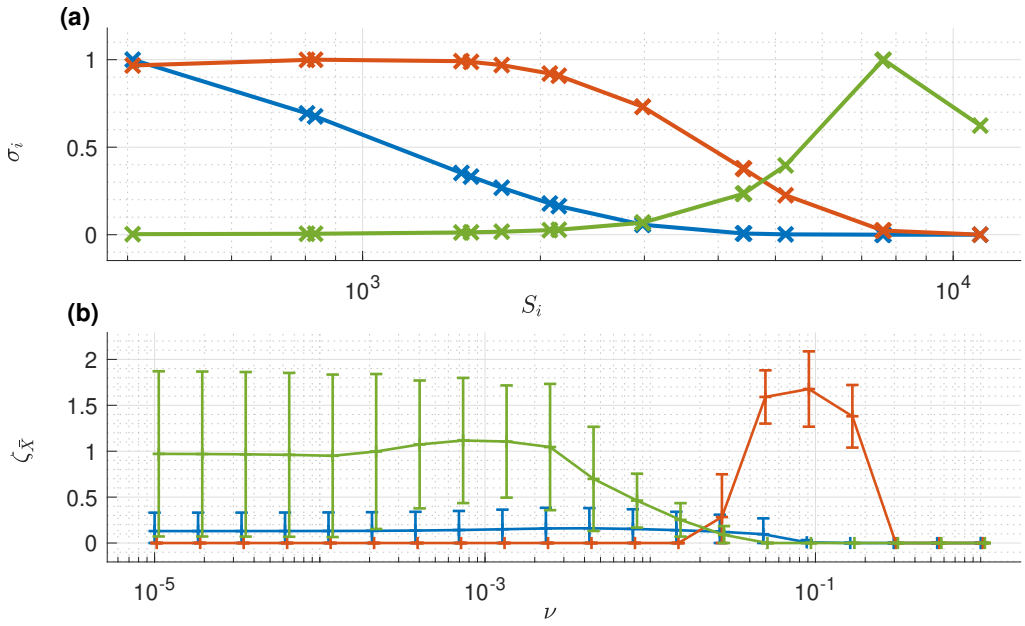

Figure 3.6: Optimizing the parameters of the CMD algorithm. (a) Three different sets of weights σ_i defined by Eq. 3.16 with $\psi = 0.3$ and $\mu = 0.3$ (blue), $\mu = 1$ (red), and $\mu = 3$ (green). (b) Performance of the CMD algorithm, measured by the value of $\zeta_{\bar{X}}$ (defined in Eq. 3.3) for various values of ν (see Eq. 2.7) and weights σ_i (colored as in (a)). Each curve represents the mean of $\zeta_{\bar{X}}$ calculated over 10 consecutive experiments, with error bars showing the standard deviation of the 10 experiments.

(μ, ψ, ν) is found using argmax on the mean ζs. The best-performing values found are $\mu = 0.3, \psi = 0.3$ and $\nu = 0.09$. We note that applying this optimization process to the above-mentioned simulations that are based on Eq. 3.14 (using a Gaussian noise model) resulted in $\mu = 0.3, \psi = 0.3$ and $\nu = 0.05$, as mentioned earlier.

Next, we examine how the exposure time (t in Eq. 3.17) and the number of Y_is affect the error (ε of Eq. 3.1) of the recovered scattering pattern ($\bar{X}$ of Eq. 2.7). To that end, for a given $t \in \{0.1,\ 0.8,\ 4\}$, we generate X_t using Eq. 3.17, and simulate a set of scattering patterns $\{Y_i\}_{i=1}^{16}$ using Eq. 3.19, with the PSFs and noise model described above Eq. 3.19. Next, for a given $k \in \{3, \ldots, 16\}$, we choose k scattering patterns Y_i at random, forming a subset of $\{Y_i\}_{i=1}^{16}$, labeled $\{Y_{i_j}\}_{j=1}^{k}$. We apply the CMD algorithm to $\{Y_{i_j}\}_{j=1}^{k}$, resulting in $\bar{X}$, calculate the recovery error ε of $\bar{X}$ using

Eq. 3.1, and label this recovery error by $\varepsilon_1^{t,k}$. For future reference, we note that the total exposure time used to produce $\{Y_{i_j}\}_{j=1}^{k}$ is $t \cdot k$, and hence, the exposure time that corresponds to $\varepsilon_1^{t,k}$ is tk. Next, using the same t and k, we repeat this procedure $\binom{16}{k} - 1$ times, each time with a different subset of $\{Y_i\}_{i=1}^{16}$ of size k (covering all possible subsets of $\{Y_i\}_{i=1}^{16}$ of size k), and denote the recovery error corresponding to the n^{th} subset by $\varepsilon_n^{t,k}$ for $n \in \{2, \ldots, \binom{16}{k}\}$. Finally, we define $\varepsilon(t,k) = min_n(\varepsilon_n^{t,k})$. In a similar way, we compute $\varepsilon(t,k)$ for all $t \in \{0.1,\ 0.8,\ 4\}$ and $k \in \{3, \ldots, 16\}$. As discussed in the previous experiment, since X_t and N_t that are used to simulate the data set $\{Y_i\}_{i=1}^{16}$, are realizations of Poisson random variables, we randomize 10 different X_ts and N_ts for a given t, simulate 10 corresponding data sets $\{Y_i\}_{i=1}^{16}$ (one for each X_t and N_t), perform the above-mentioned procedure to each $\{Y_i\}_{i=1}^{16}$ independently, and calculate the corresponding $\varepsilon(t,k)$. Finally, we define $\mathcal{E}(t,k)$ as the mean over the 10 values of $\varepsilon(t,k)$, and use $\mathcal{E}(t,k)$ to examine the dependence of the recovery error on the exposure time t and the number of Y_is k. In Fig. 3.7, we show $\mathcal{E}(t,k)$ using solid curves, where each curve represents $\mathcal{E}(t,k)$ as a function of k for a given t. The curves are colored in purple for $t = 0.1$, orange for $t = 0.8$, and gray for $t = 4$. The error bars in Fig. 3.7 correspond to the standard deviation of $\varepsilon(t,k)$ over the 10 repetitions. As one can notice, the recovery error ε decreases as the size of the subset of $\{Y_i\}_{i=1}^{16}$ (i.e. k) increases. In addition, longer exposure time results in a lower recovery error ε.

As the total exposure time that corresponds to each $\varepsilon(t,k)$ is tk (as discussed in the previous paragraph), we would like the examine the recovery error ε of a single scattering pattern, taken with an exposure time of tk and no PSF blur. In order to do so, we define

$$\tilde{X}_{t,k} = \texttt{poissrnd}\big(tk \cdot (X_{GT} + N_{BKG})\big), \tag{3.22}$$

where X_{GT} and N_{BKG} are simulated as in the previous setup, with the indexing t, k explained below. For a given X_{GT}, N_{BKG}, t and k, we simulate 10 independent realizations $\tilde{X}_{t,k}$, use Eq. 3.1 to calculate the corresponding recovery error ε of each $\tilde{X}_{t,k}$, and label it $\tilde{\varepsilon}(t,k,n)$ for $n \in \{1, ..., 10\}$. Finally we calculate $\tilde{\mathcal{E}}(t,k) = \frac{1}{10}\sum_{n=1}^{10} \tilde{\varepsilon}(t,k,n)$ for a fixed t and k. Repeating this for $t \in \{0.1,\ 0.8,\ 4\}$

and $k \in \{3, \ldots, 16\}$ we end up with the dependence of $\tilde{\mathcal{E}}(t,k)$ on t and k. This dependence is shown in Fig. 3.7 as dashed curves, where each curve shows $\tilde{\mathcal{E}}(t,k)$ as a function of k for a given t. The curves are colored by purple for $t = 0.1$, orange for $t = 0.8$, and gray for $t = 4$. As before, the curves in Fig. 3.7 represent $\tilde{\mathcal{E}}(t,k)$ and the error bar represents the standard deviation of the 10 consecutive $\tilde{\varepsilon}(t,k)$. This experiment presents the main advantage of the CMD algorithm. For short exposure times, which are common and desired in SAXS experiments [18, 22, 33], applying the CMD algorithm on a set of scattering patterns Y_i, each taken with a different PSF, outperforms taking a single scattering pattern with an equivalent exposure time and an unrealistic blur-less system. We have used in this experiment the recovery error ε of Eq. 3.1 instead of the quality measure ζ of Eq. 3.3, as the former better presents the point demonstrated here.

Finally, we conduct an experiment that shows a limitation of the CMD algorithm. We simulate X_{GT} with two peaks using Eq. 3.5, and a set of scatting patterns $\{Y_i\}_{i=1}^{16}$ using Eq. 3.19 and the surrounding paragraph. We then apply the CMD algorithm on $\{Y_i\}_{i=1}^{16}$, and examine how close the two peaks of X_{GT} can be, while still being able to identify a valley between the two peaks in $I_{\bar{X}}[r]$, where $\bar{X}$ is the recovered scattering pattern using the CMD algorithm (Eq. 2.7). We study this using the following setup. First, we simulate X_{GT} with two terms $I(r|r_0^i, \gamma^i, D^i)$ (defined in Eq. 3.5) where $D^{1,2} = 1$, $\gamma^{1,2} = 0.7$, $r_0^1 = 30$ pixels, and r_0^2 varies between 31 to 45 pixels. Next, for each of the resulting X_{GT}, we simulate a set of scattering patterns Y_i using Eq. 3.19 and the set of PSFs defined by Eq. 3.13, with parameters given in Table 3.2. We then apply the CMD algorithm to the resulting set of Y_is. Finally, we calculate $\zeta_{\bar{X}}$ for each $\bar{X}$ that results from the runs of the CMD algorithm. Note that each run uses a set of Y_is that correspond to a different X_{GT}, as r_0^2 is different for different runs. We define $\Delta R = r_0^2 - r_0^1$, and present $\zeta_{\bar{X}}$ as a function of ΔR in Fig. 3.8 using black markers. For clarity, we denote $\zeta_{\bar{X}}$ by $\zeta(\Delta R)$ to emphasize its dependence on ΔR. For comparison, $\zeta(\Delta R)$ can be calculated analytically from

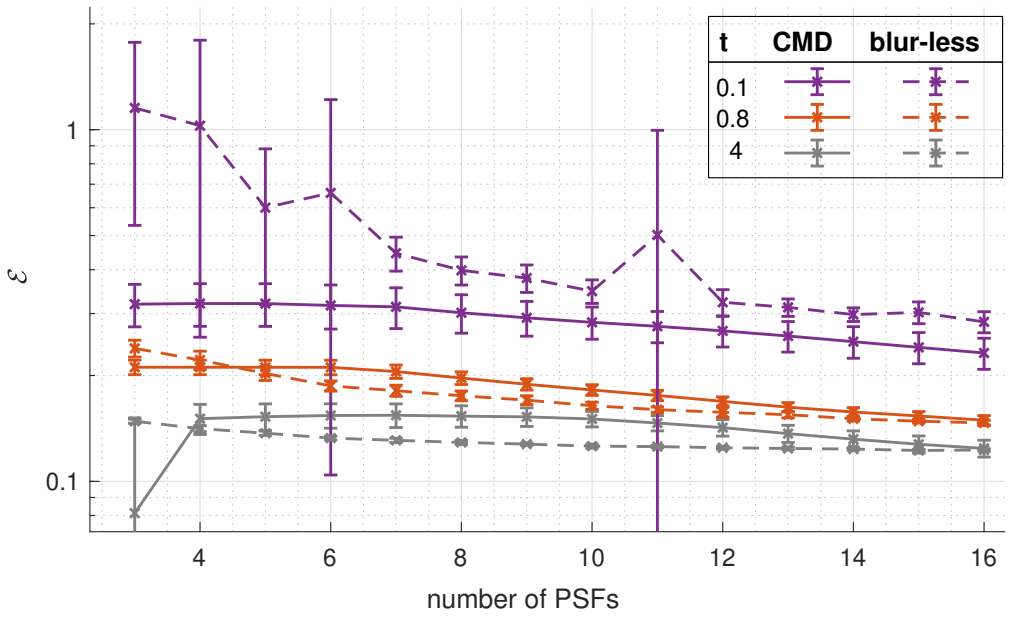

Figure 3.7: Recovery error (Eq. 3.1) for the CMD algorithm as a function of the number of PSFs used and exposure time (t of Eq. 3.17) (solid lines), using the optimal ν ($\nu = 0.09$) and a σ_i function defined by Eq. 3.16 with $\mu = \psi = 0.3$ (presented as the red curve in Fig. 3.6a). For comparison, ε (Eq. 3.1) of a scattering pattern defined by Eq. 3.22 with an equivalent exposure time is presented in dashed lines. The different lines represent short, intermediate and long simulated exposure times with $t = 0.1,\ 0.8,\ 4$, respectively, as defined in Eqs. 3.17 and 3.22. Error bars show the standard deviation of the recovery error over 10 independent simulations.

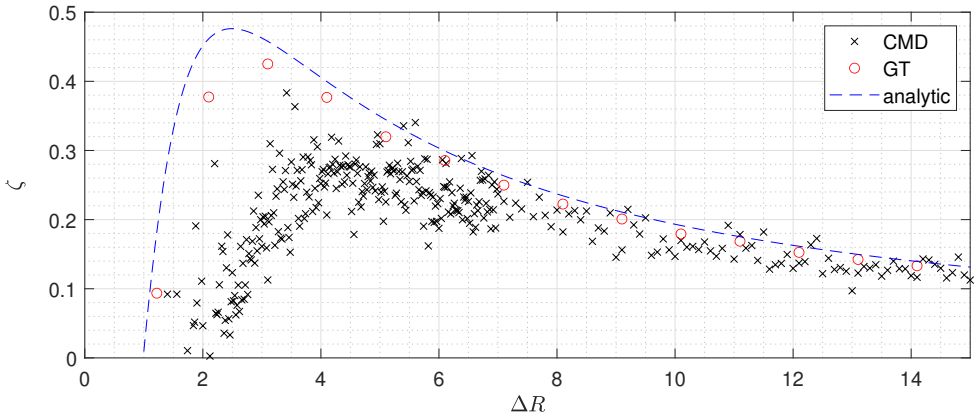

Figure 3.8: Quality measure ζ (Eq. 3.3) for various distances ΔR between the two peaks in the radial profile. Black markers correspond to $\zeta_{\bar{X}}$ of scattering patterns recovered by the CMD algorithm. Red circles correspond to $\zeta_{X_{GT}}$ of X_{GT} used to simulate the different Y_is used by the CMD algorithm. Dashed line corresponds to the analytical value of ζ as given by Eq. 3.23.

Eq. 3.5 to be

$$\zeta_A(\Delta R) = \frac{2D\Delta R\big((\Delta R)^2 - 2\gamma^2\big)}{\big(\gamma^2 + (\Delta R)^2\big)\big(4\gamma^2 + (\Delta R)^2\big)}, \quad \Delta R > 0, \tag{3.23}$$

where ζ_A stands for *analytic*, and we assume the same D and γ for both peaks, as in our setup. The value of $\zeta_A(\Delta R)$ is shown as a dashed line in Fig. 3.8. In addition, we numerically calculated $\zeta_{X_{GT}}$ for the scattering patterns X_{GT} that were used to simulate the different sets of Y_is. In order to verify that these values of $\zeta_{X_{GT}}$ approach the analytical values of $\zeta_A(\Delta R)$, we present $\zeta_{X_{GT}}$ for $\Delta R \in \{1.1, 2.1, \ldots, 14.1\}$ using red circles in Fig. 3.8, labelled 'GT' for ground truth. As one can notice, the values of $\zeta(\Delta R)$ approach their corresponding analytical values for $\Delta R > 5$. However, for $\Delta R \in (1.7, 5)$, $\zeta(\Delta R)$ doesn't follow the analytical curve. In addition, while the analytical values of $\zeta_A(\Delta R)$ remain positive down to $\Delta R = 1$, which is related to the existence of a valley between the two peaks (see explanation above Eq. 3.3), the calculated $\zeta(\Delta R)$ vanishes for $\Delta R < 1.8$.

In Fig. 3.9, we demonstrate the added value of the srSAXS algorithm using sim-

ulated data. The procedure for simulating the data for the srSAXS algorithm is described by Eq. 3.20 and its surrounding paragraph. Applying the srSAXS algorithm to the simulated Y_is results in a recovered scattering pattern in which the nearby peaks can be clearly identified. The quality measure of the recovered pattern is $\zeta_{\bar{X}} = 0.026$. Remarkably, $\zeta_{\bar{X}}$ is larger than ζ_{Y_i} for all scattering patterns Y_i used by the srSAXS algorithm, among which the largest is $\zeta_{Y^*} = 0.022$.

3.3 SAXS measurments

Motivated by our simulated data results, we next test the CMD algorithm using measure SAXS data. Our test samples are Silver behenate (AgBh) powder, showing lamellar scattering signals, and DOPE phospholipids in solution, showing self-assembled inverted hexagonal phase. Additional information regarding the sample preparation procedure can be found in Appendix A.1. Both samples exhibit concentric scattering rings, that appear as peaks and valleys in the corresponding radial profiles (Eq. 3.2), and follow our assumption in the paragraph above Eq. 3.3. As common in the SAXS literature [3, 14, 16, 18, 19, 23, 25, 26, 32, 41], the r-axis of the radial profiles, which was previously presented in units of $pixels$, is presented in this section in q space (see Fig. 1.1). The dependence between r and q is approximately linear, and throughout this section we use

$$q[nm^{-1}] = C \cdot r[pixels], \tag{3.24}$$

with $C = 0.12\ \frac{nm^{-1}}{pixels}$. The precise dependence between r and q, together with the derivation of Eq. 3.24, the calculation of C, and additional information on the SAXS setup are given in Appendix A.2.

Throughout this section, 6 different PSFs are used to capture the different scattering patterns Y_i of Eq. 2.1. The PSFs are determined by the following slit sizes (measured in mm^2): 0.6×0.6 ; 0.6×0.8 ; 0.6×1.0 ; 0.8×0.6 ; 0.8×0.8 ; 0.8×1.0. Fig 3.10 presents these PSFs as they were captured on the detector, as well as the middle row/column of each captured PSF that show the shape of the beam. In Ta-

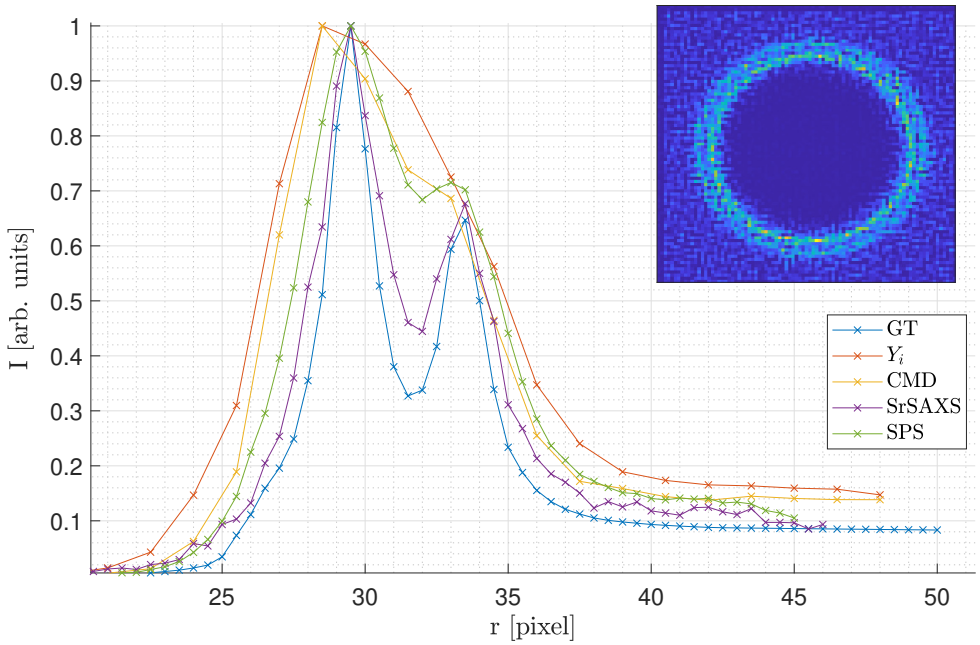

Figure 3.9: Radial profiles of simulated and recovered scattering patterns, demonstrating the added value of the srSAXS algorithm. Different radial profiles are presented for comparison: the blue curve corresponds to the radial profile of the scattering pattern X_{GT} used to simulate the Y_is; the red curve corresponds to the radial profile of Y_i produced with some PSF simulated using Eq. 3.13 and Table 3.2; the yellow curve corresponds to the radial profile of the scattering pattern recovered by the CMD algorithm; the purple curve corresponds to the radial profile of the scattering pattern recovered by the srSAXS algorithm; and the green curve corresponds to the radial profile of the super resolution algorithm of [36] used without their proposed deconvolution process, labeled SPS; in addition, the scattering pattern recovered by the srSAXS algorithm is presented in the inset.

PSF $[mm \times mm]$	Area $[\times 10^{-2}\, mm^2]$	Flux $[\times 10^{6}\, counts/sec]$
0.2×0.2	4	0.258
0.4×0.4	16	2.807
0.2×0.8	16	2.185
0.6×0.4	24	4.959
0.4×0.6	24	5.006
0.8×0.4	32	6.840
0.4×0.8	32	7.137
0.6×0.6	36	9.013
1.0×0.4	40	8.577
0.4×1.0	40	9.705
0.6×0.8	48	12.625
0.8×0.6	48	13.360
0.6×1.0	60	16.295
0.8×0.8	64	18.693
0.8×1.0	80	25.248
1.0×1.0	100	28.799

Table 3.3: Values of incoming flux as measured directly on the detector. Various slit shapes are presented, where the slits that were used in our experiments are highlighted in green. A slit that was used for additional comparisons is highlighted in yellow.

ble 3.3, we present the values of incoming flux as measured directly on the detector for various PSFs, highlighting in green the PSFs that where used to measure the different Y_is. In addition, for various comparisons throughout this section, we use a slit size of $0.2 \times 0.2\ mm^2$ in addition to the set of PSFs described above. This is highlighted in yellow in Table 3.3.

In order to capture scattering patterns for the srSAXS algorithm, we move the detector in sub-pixel translations of $\pm 172/3\ \mu m$ between consecutive captures, as the pixel size of our apparatus is $172 \times 172\ \mu m^2$ (specified in Appendix A.2). This is done repeatedly for each PSF, and the resulting data set is of the form $\{Y_i^k\}$, where $k \in \{1, \ldots, 9\}$ for the different sub-pixel translations, and $i \in \{1, \ldots, 6\}$ for the different PSFs.

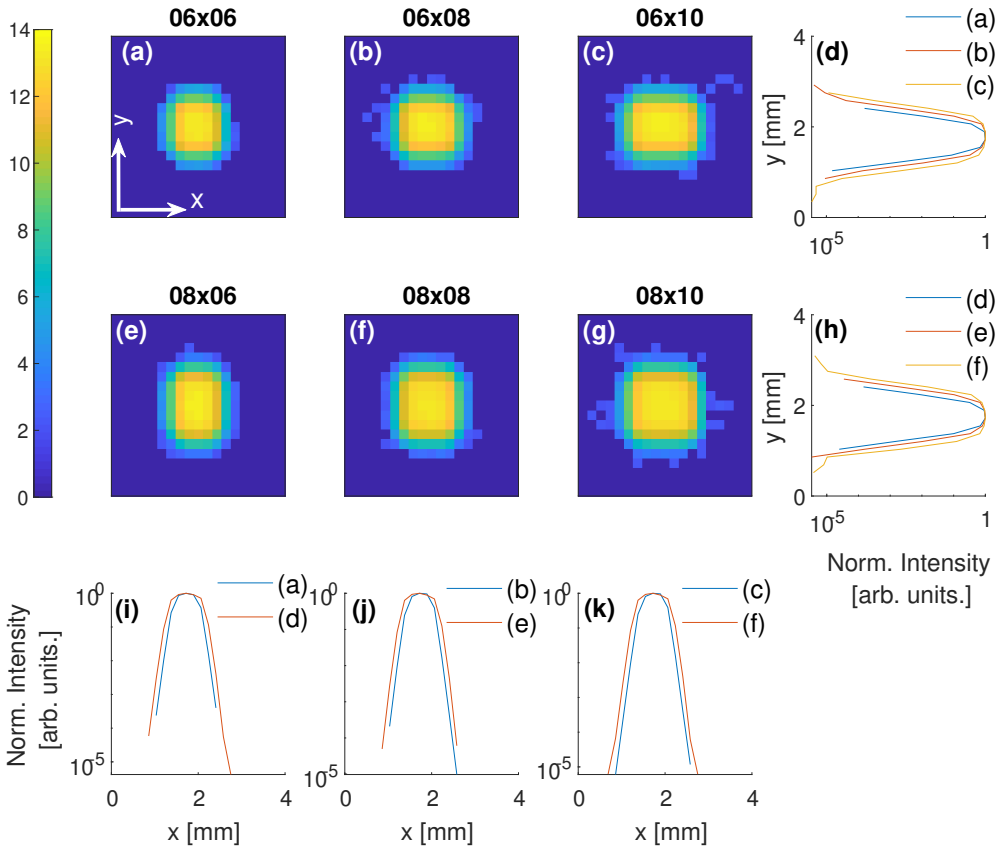

Figure 3.10: Measured PSFs. Panels (a) – (c) and (e) – (g) show (on a log scale) PSFs corresponding to slits sizes of 0.6×0.6 ; 0.6×0.8 ; 0.6×1.0; 0.8×0.6 ; 0.8×0.8 ; $0.8 \times 1.0\, mm^2$, respectively. (d), (h),(i)–(k) Middle row/column of the PSFs, normalized to have a maximum of 1 to demonstrate the beam shaping effect.

Throughout this section, we calculate the quality measure ζ (Eq. 3.3) of the scattering patterns that correspond to the DOPE sample. The reason for using this sample is that its radial profile is characterized by a small distance between the second and third peaks (counting outwards). In order to calculate ζ out of the DOPE scattering patterns, we excluded the first peak from their radial profiles, and calculated the quality measure (using Eq. 3.3) of the remaining of the radial profile.

In order to apply the CMD algorithm to measured SAXS data, 6 different scattering patterns were taken for the DOPE sample (one for each PSF), with exposure times of $t = 10$ *sec* each. As discussed in the previous section, the optimal performance of the CMD algorithm depends of several parameters, and in order to optimize them, we have conducted a grid search as described earlier in the paragraph surrounding Eq. 3.21. Recalling Eq. 3.21

$$\sigma_i = \exp\left(- \frac{(S_i - \mu\bar{S})^2}{(\psi\bar{S})^2} \right), \qquad \bar{S} = \frac{1}{6}\sum_{j=1}^{6} S_j,$$

we found that the optimal performance of the CMD algorithm is achieved for $\mu = 0.7$, $\psi = 1.2$ and $\nu = 0.04$. The recovered scattering pattern is presented in Fig. 3.11b, and its corresponding quality measure is calculated (using Eq. 3.3) to be $\zeta_{\bar{X}} = 1.15$. In addition, since the total exposure time used to measure the Y_is used by the CMD algorithm is 10×6 *sec*, we measured a raw scattering pattern Y^* with a PSF that corresponds to a slit size of 0.6×0.6 mm^2, and with an exposure time of $t = 60$ *sec*. Note that this slit is the tightest that was used to capture the Y_is. Y^* is presented in Fig. 3.11a, and its corresponding quality measure is calculated to be $\zeta_{Y^*} = 0.57$. For comparison, the radial profiles of the scattering pattern $\bar{X}$ recovered by the CMD algorithm, and the raw scattering pattern Y^* are presented in Fig. 3.11c. One can find the second and third peaks in $q \in (1.5, 2.2)$ nm^{-1}, and notice that the valley between the two peaks is indeed shallow for $I_{Y^*}[q]$ (blue curve) while deep for $I_{\bar{X}}[q]$ (red curve).

Next, 6 scattering patterns where taken for the AgBh sample (one for each PSF), with exposure times of $t = 10$ *sec* each. Using the same parameters for the CMD al-

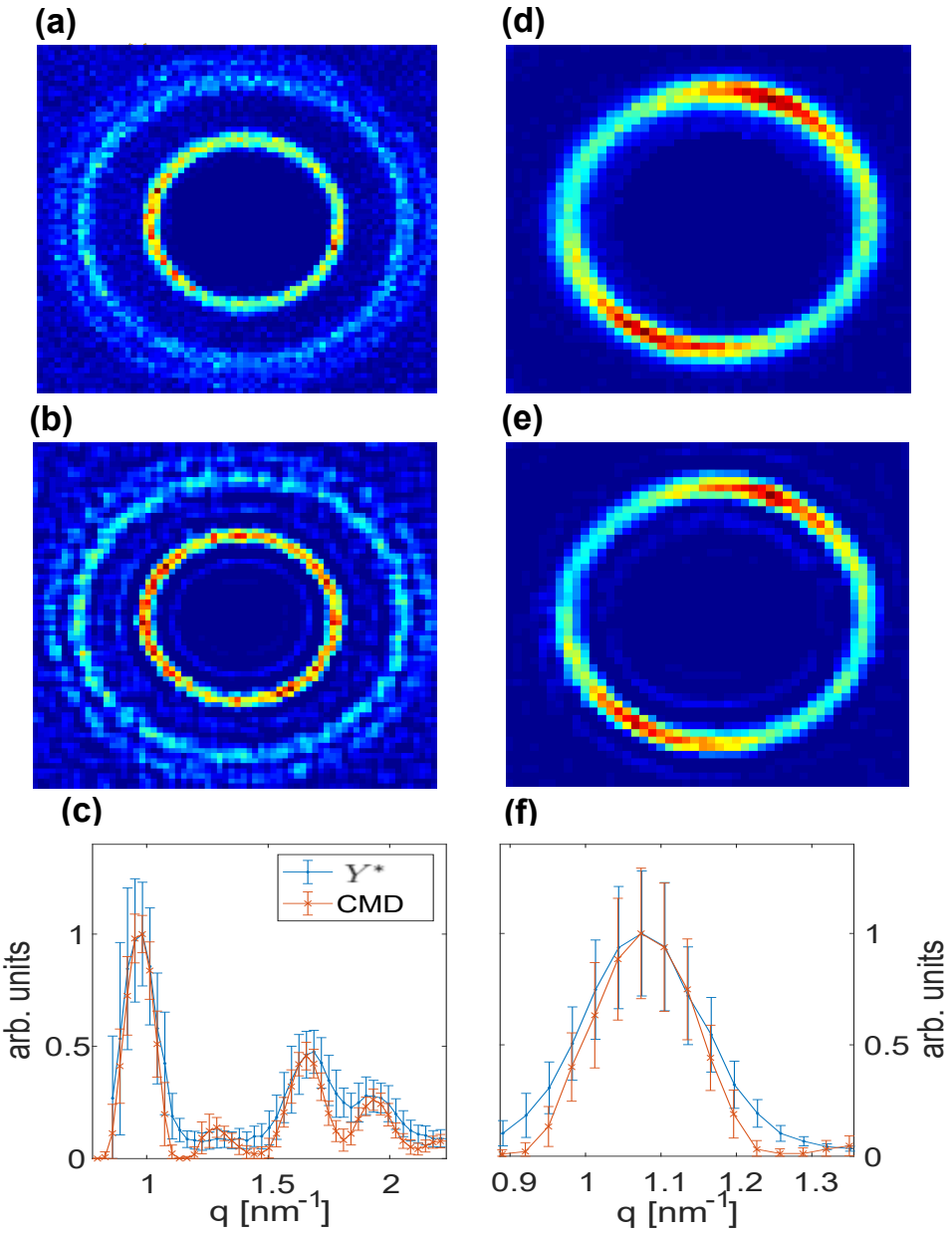

Figure 3.11: CMD algorithm applied to measured data: (a)–(c) DOPE sample, (d)–(f) AgBh sample. (a), (d) Measurements with PSF that corresponds to a slit of size 0.6×0.6 mm^2, and with an exposure time of 60 *sec*, labeled Y^*. (b), (e) Scattering pattern recovered by the CMD algorithm using 6 different PSFs, each with an exposure time of 10 seconds. (c), (f) Radial profiles of (a)–(b), (d)–(e). The error bars represent standard deviation for 10 independent measurements of each sample.

gorithm as above ($\mu = 0.7, \psi = 1.2$ and $\nu = 0.04$), the resulting recovered scattering pattern is presented in Fig. 3.11e. The scattering pattern Y^* for this sample (measured as above) is presented in Fig. 3.11d, and for comparison, the corresponding radial profiles are presented in Fig. 3.11f

In Fig. 3.12, we present the results of the srSAXS algorithm for the scattering data of DOPE sample, measured at 9 different sub-pixel translations, using 6 different PSFs, and with exposure times of 10 *sec* each. Measuring the DOPE sample and processing the measured Y_i^ks by the srSAXS algorithm results in $\bar{X}_{srSAXS}$, shown in Fig. 3.12c. We use the $srSAXS$ subscript for $\bar{X}$ to emphasize the algorithm that was used to recover $\bar{X}$. The corresponding quality measure ζ (Eq. 3.3) is $\zeta_{\bar{X}_{srSAXS}} = 1.22$. This quality measure is larger than the value $\zeta_{\bar{X}} = 1.15$ reported for the previous experiment. For additional comparison, we measured another two raw scattering patterns Y_i. First, similarly to the previous experiment, we measured a raw scattering pattern Y^* with the tightest slit used by the srSAXS algorithm ($0.6 \times 0.6\ mm^2$) and with an equivalent total exposure time of 540 *sec* (that is 9 translations $\times$ 6 PSFs $\times$ 10 *sec*). The quality measure (Eq. 3.3) of Y^* is calculated to be $\zeta_{Y^*} = 0.59$. In addition, we measured a raw scattering pattern Y^{**} using the tightest slit possible in our setup ($0.2 \times 0.2\ mm^2$), using an exposure time of 1080 *sec*, which is twice the total exposure time used to generate the Y_i^k data set for the srSAXS algorithm. Y^{**} is presented in Fig. 3.12a, and its quality measure is calculated to be $\zeta_{Y^{**}} = 1.21$. The corresponding radial profiles are presented in Fig. 3.12g.

These results manifest the added value of the srSAXS algorithm. However, we do notice that the CMD and srSAXS algorithms produce additional small spurious peaks. One way of solving this is by choosing another regularization method, such as L_1 or some Tikhonov Regularization [12]. Our experiments with such regularizers show different results, but not with great significance, and this remains to be further researched.

In addition, we applied the srSAXS algorithm to the scattering data of the AgBh sample, measured at 9 different sub-pixel translations, using 6 different PSFs, and with exposure times of 10 *sec* each. The scattering pattern recovered by the srSAXS

algorithm is presented in Fig. 3.12f. In a similar way to the previous experiment, we capture Y^* and Y^{**}, where the later is presented in Fig. 3.12d. For comparison, the corresponding radial profiles are presented in Fig. 3.12h.

We compared our srSAXS algorithm with other deconvolution techniques applied to the measured scattering patterns of the DOPE sample. The results are presented in Fig. 3.13. In this paragraph, we subscript the calculated quality measures (ζ of Eq. 3.3) by the algorithm used to recover the corresponding scattering pattern. We implemented the Richardson-Lucy (RL) algorithm [27, 35], which is discussed in Chapter 1, and present its recovered scattering pattern in Fig. 3.13b. The corresponding radial profile is presented as the red curve in Fig. 3.13e, and the quality measure was calculated to be $\zeta_{RL} = 1.19$. An additional comparison is presented using the full implementation of the algorithm of Farsiu et. al (FA), including their proposed deconvolution [36], in Fig. 3.13d. The corresponding radial profile is presented as the green curve in Fig. 3.13e, and the quality measure was calculated to be $\zeta_{FA} = 1.17$. A raw scattering pattern Y^{**}, which is taken with the tightest slits opening possible in our setup ($0.2 \times 0.2\ mm^2$) and with exposure time of 1080 *sec* is presented in 3.13a. The corresponding radial profile is presented as the blue curve in Fig. 3.13e, and the quality measure was calculated to be $\zeta_{Y^{**}} = 1.21$. For comparison, the results of our srSAXS algorithm are presented in Fig. 3.13c. Its corresponding radial profile is presented as the purple curve in Fig. 3.13e, and shows better performance with $\zeta_{srSAXS} = 1.22$.

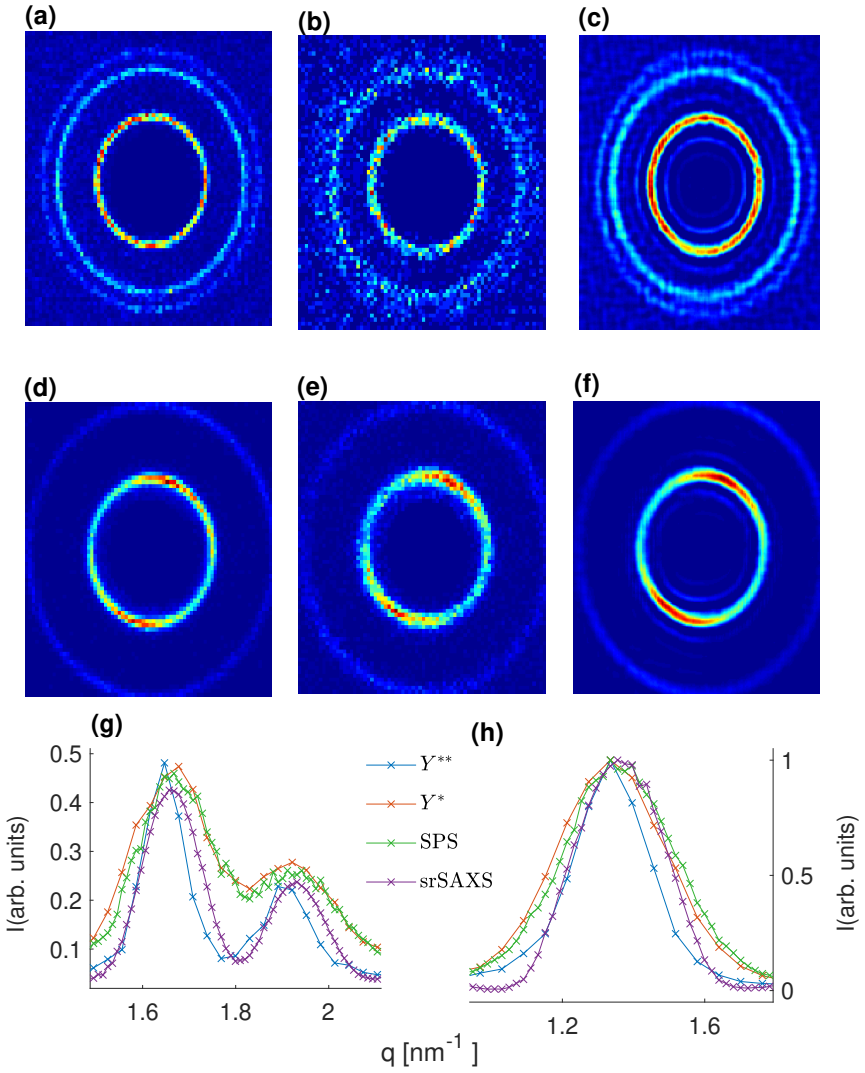

Figure 3.12: Measurements and recovered scattering patterns using the srSAXS algorithm for (a)–(c), (g) DOPE and (d)–(f), (h) AgBh samples. (a), (d) Raw scattering patterns Y^{**}, taken with the tightest slit size possible in the setup ($0.2 \times 0.2\ mm^2$), and exposure time of 18 minutes. (b), (e) Scattering patterns with slit size of $0.6 \times 0.6\ mm^2$ and exposure time of 10 *sec* (raw data for the CMD algorithm). (c), (f) SrSAXS recovered scattering patterns with 9 sub-pixel translations and 6 different PSFs. (g), (h) Radial profiles. Blue curves correspond to Y^{**} (shown in (a), (d)), red curves correspond to the raw scattering patterns Y^{*}, taken with slit size of $0.6 \times 0.6\ mm^2$ and exposure times of 540 *sec*, green curves correspond to the super resolution algorithm of [36] applied without their deconvolution procedure, labeled SPS, and purple curves correspond to the scattering patterns recovered by the srSAXS algorithm.

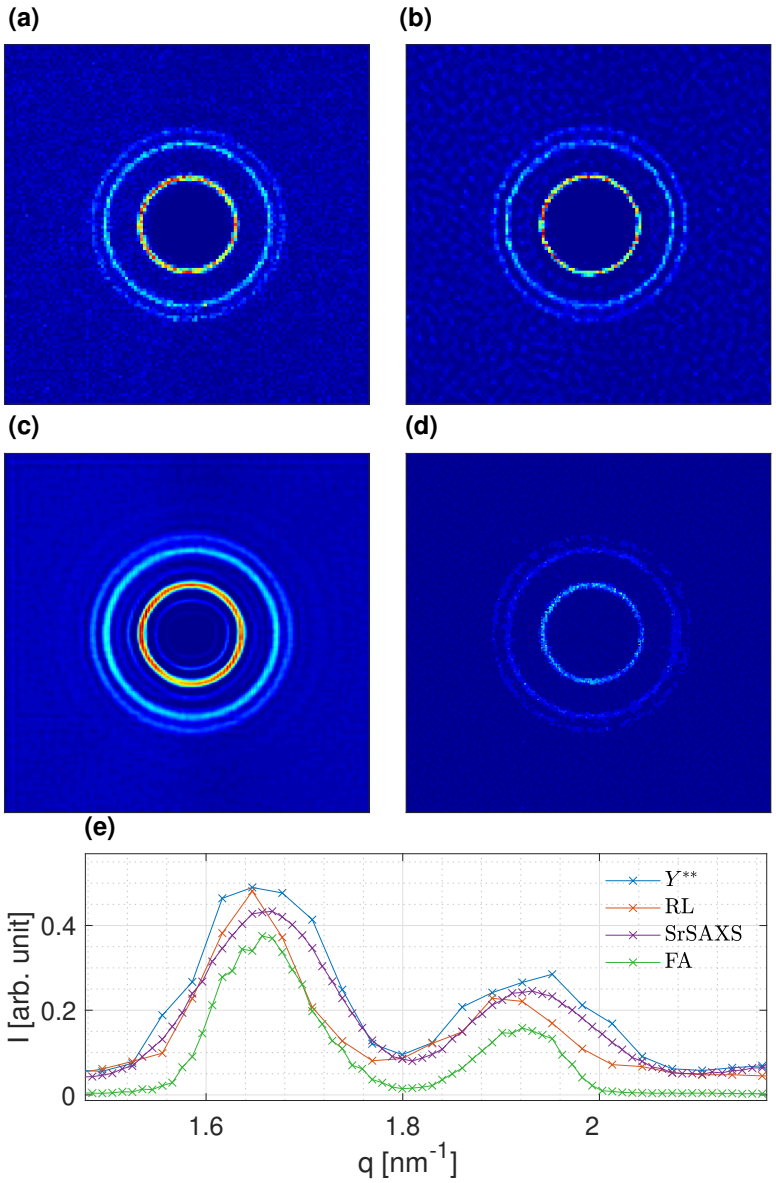

Figure 3.13: Comparison of different deconvolution techniques. (a) Raw scattering pattern Y^{**}, taken with an exposure time of 18 minutes, and with slit size of 0.2×0.2 mm^2. (b) Richardson-Lucy deconvolution technique, using the 0.6×0.6 mm^2 PSF and an exposure time of 9 minutes (RL). (c) Our srSAXS algorithm with a total exposure time of $6 \times 9 \times 10$ seconds (similar to Fig. 3.12c). (d) Full implementation of Farsiu algorithm (FA) [36] (including their proposed deconvolution technique) using the 0.6×0.6 mm^2 slit and an exposure time of 9 minutes. (e) Radial profiles.

Chapter 4

Summary

In this thesis, we propose a deconvolution algorithm that takes as input multiple scattering patterns, each acquired with a different PSF, and recovers their underlying scattering pattern. This is done by using the acquired scattering patterns to cast a minimization problem. We show that this minimization problem has a unique solution, and solve it using the conjugate gradients method. In addition, we propose cascading our algorithm with a sub-pixel super resolution algorithm [36].

In order the test the proposed algorithm, we conducted experiments that are based on two types of data sets: numerical simulations, and measured SAXS data. In order to evaluate the recovered scattering patterns, we phrased a quality measure, that is based on the azimuthal symmetry of the scattering pattern. We simulated scattering patterns with various parameters, such as the number of the scattering patterns and the exposure times, and tested how the quality measure depends on these. In addition, we showed a limitation that the proposed algorithm has in recovering specific scattering patterns. Finally, we showed the added value of cascading our proposed algorithm with a super resolution algorithm. Furthermore, we applied our proposed algorithm upon measured scattering patterns and demonstrated the value of the proposed algorithm by comparing it to other deconvolution algorithms.

While our results demonstrate the added value of our algorithms, we do notice that the CMD and srSAXS algorithms produce additional small spurious peaks. One way of solving this may involve choosing another regularization method, such as L_1

or some Tikhonov Regularization [12]. Our experiments with such regularizers are not conclusive, and this remains to be further researched. In addition, we point that our cascaded algorithm is somewhat brute-force, and a more natural approach would be to combine the two phases of the cascaded algorithm into a single unified framework. Finally, despite the fact that super resolution algorithms exist for 30 years, to our best knowledge, this is the first proposal of applying a super resolution algorithm to SAXS measurements, and applying it in a synchrotron facility may result in tremendous advancement.

Bibliography

[1] J.V. Beck and K.J. Arnold. *Parameter Estimation in Engineering and Science*. Probability and Statistics Series. Wiley, 1977.

[2] Jonathan M. Blackledge. Chapter 12 - image restoration and reconstruction. In Jonathan M. Blackledge, editor, *Digital Image Processing*, Woodhead Publishing Series in Electronic and Optical Materials, pages 404 – 438. Woodhead Publishing, 2005.

[3] Clement E Blanchet and Dmitri I Svergun. Small-angle x-ray scattering on biological macromolecules and nanocomposites in solution. *Annual review of physical chemistry*, 64:37–54, 2013.

[4] James Cooley and John Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, 19(90):297–301, 1965.

[5] Cameron Danesh. *SURFACTANT FORMULATIONS FOR WATER-BASED PROCESSING OF A POLYTHIOPHENE DERIVATIVE*. the Faculty of California Polytechnic State University, San Luis Obispo, 06 2013.

[6] W.H. de Jeu and W.H. Jeu. *Basic X-ray Scattering for Soft Matter*. Oxford University Press, 2016.

[7] Flavio Dell'Acqua, Paola Scifo, Giovanna Rizzo, Marco Catani, Andrew Simmons, Giuseppe Scotti, and Ferruccio Fazio. A modified damped richardson–lucy algorithm to reduce isotropic background effects in spherical deconvolution. *Neuroimage*, 49(2):1446–1458, 2010.

[8] Yadolah Dodge. *Normal Equations*, pages 380–382. Springer New York, New York, NY, 2008.

[9] Michael Elad and Arie Feuer. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. *IEEE transactions on image processing*, 6(12):1646–1658, 1997.

[10] Donald Geman and George Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Transactions on pattern analysis and machine intelligence*, 14(3):367–383, 1992.

[11] O. Glatter, O. Kratky, and H.C. Kratky. *Small Angle X-ray Scattering*. Academic Press, 1982.

[12] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 4th edition, 2013.

[13] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Pearson Education, 2011.

[14] Maria Herbst, Eddie Hofmann, and Stephan Forster. Nucleation and growth kinetics of zno nanoparticles studied by in situ microfluidic saxs/waxs/uv–vis experiments. *Langmuir*, 35(36):11702–11709, 2019.

[15] Magnus R Hestenes, Eduard Stiefel, et al. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409–436, 1952.

[16] Wen-Qiang Hua, Yu-Zhu Wang, Ping Zhou, Tao Hu, Xiu-Hong Li, Feng-Gang Bian, and Jie Wang. Microfocus small-angle x-ray scattering at ssrf bl16b1. *Chinese Physics C*, 41(4):048001, Apr 2017.

[17] Michal Irani and Shmuel Peleg. Improving resolution by image registration. *CVGIP: Graphical Models and Image Processing*, 53(3):231 – 239, 1991.

[18] Guy Jacoby, Keren Cohen, Kobi Barkan, Yeshayahu Talmon, Dan Peer, and Roy Beck. Metastability in lipid based particles exhibits temporally deterministic and controllable behavior. *Scientific Reports*, 5:9481, mar 2015.

[19] Guy Jacoby, Irina Portnaya, Dganit Danino, Haim Diamant, and Roy Beck. Delayed nucleation in lipid particles. *Soft Matter*, 16:247–255, 2020.

[20] R. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981.

[21] Mahmoud M Khattab, Akram M Zeki, Ali A Alwan, and Ahmed S Badawy. Regularization-based multi-frame super-resolution: A systematic review. *Journal of King Saud University - Computer and Information Sciences*, 32(7):755 – 762, 2020.

[22] Nigel Kirby, Nathan Cowieson, Adrian M. Hawley, Stephen T. Mudie, Duncan J. McGillivray, Michael Kusel, Vesna Samardzic-Boban, and Timothy M. Ryan. Improved radiation dose efficiency in solution SAXS using a sheath flow sample environment. *Acta Crystallographica Section D*, 72(12):1254–1266, Dec 2016.

[23] M. Kornreich, R. Avinery, and R. Beck. Modern x-ray scattering studies of complex biological systems. *Current Opinion in Biotechnology*, 24(4):716–723, 2013.

[24] Xin Li and M. T. Orchard. New edge-directed interpolation. *IEEE Transactions on Image Processing*, 10(10):1521–1527, 2001.

[25] Y Li, R Beck, T Huang, M C Choi, and M Divinagracia. Scatterless hybrid metal–single-crystal slit for small-angle x-ray scattering and high-resolution x-ray diffraction. *Journal of Applied Crystallography*, 41(6):1134–1139, December 2008.

[26] Jan Lipfert and Sebastian Doniach. Small-angle x-ray scattering from rna, proteins, and protein complexes. *Annual Review of Biophysics and Biomolecular Structure*, 36(1):307–327, 2007.

[27] Leon Lucy. An iterative technique for the rectification of observed distributions. *Astronomical Journal*, page 745–754, 1974.

[28] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

[29] A. Papoulis. *Signal Analysis*. Dover Books on Electrical Engineering. Dover Publications, Incorporated, 2018.

[30] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *IEEE signal processing magazine*, 20(3):21–36, 2003.

[31] Brian Richard Pauw. Everything SAXS: small-angle scattering pattern collection and correction. *Journal of Physics: Condensed Matter*, 25(38):383201, aug 2013.

[32] J. S. Pedersen, D. Posselt, and K. Mortensen. Analytical treatment of the resolution function for small-angle scattering. *Journal of Applied Crystallography*, 23(4):321–333, August 1990.

[33] Javier Pérez and Yoshinori Nishino. Advances in x-ray scattering: from solution saxs to achievements with coherent beams. *Current opinion in structural biology*, 22(5):670–678, 2012.

[34] Seunghyeon Rhee and Moon Gl Kang. Discrete cosine transform based regularized high-resolution image reconstruction algorithm. *Optical Engineering*, 38(8):1348–1356, August 1999.

[35] William Hadley Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society in America*, pages 55–59, 1972.

[36] M. Elad S. Farsiu, M. D. Robinson and P. Milanfar. Fast and robust multiframe super resolution. *IEEE Transactions on Image Processing*, 13(10):1327–1344, Oct 2004.

[37] Manjeet Singh, Uma Shanker Tiwary, and Kim Young-Hoon. An adaptively accelerated lucy-richardson method for image deblurring. *EURASIP Journal on Advances in Signal Processing*, 2008, 02 2008.

[38] Donald L. Snyder, Carl W. Helstrom, Aaron D. Lanterman, Mohammad Faisal, and Richard L. White. Compensation for readout noise in ccd images. *J. Opt. Soc. Am. A*, 12(2):272–283, Feb 1995.

[39] Diamond Light Source. https://www.diamond.ac.uk/instruments/soft-condensed-matter/small-angle/i22/specs.html.

[40] M. Spivak. *Calculus*. Calculus. Cambridge University Press, 2006.

[41] Pablo Székely, Avi Ginsburg, Tal Ben-Nun, and Uri Raviv. Solution X-ray scattering form factors of supramolecular self-assembled structures. *Langmuir*, 26(16):13110–29, aug 2010.

[42] H.H. Telle, A.G. Ureña, and R.J. Donovan. *Laser Chemistry: Spectroscopy, Dynamics and Applications*. Wiley, 2007.

[43] R Tsai. Multiframe image restoration and registration. *Advance Computer Visual and Image Processing*, 1:317–339, 1984.

[44] Hanoch Ur and Daniel Gross. Improved resolution from subpixel shifted pictures. *CVGIP: Graphical Models and Image Processing*, 54(2):181 – 186, 1992.

[45] Thomas Vad and W. Sager. Comparison of iterative desmearing procedures for one-dimensional small-angle scattering data. *Journal of Applied Crystallography*, 44(1):32–42, Feb 2011.

[46] Chengqing Wang, Kwang-Woo Choi, Wei-En Fu, Derek L. Ho, Ronald L. Jones, Christopher Soles, Eric K. Lin, Wen-Li Wu, James S. Clarke, and Benjamin Bunday. CD-SAXS measurements using laboratory-based and synchrotron-based instruments. In John A. Allgair and Christopher J. Raymond, editors, *Metrology, Inspection, and Process Control for Microlithography XXII*, volume 6922, pages 845 – 851. International Society for Optics and Photonics, SPIE, 2008.

[47] R. Wang. *Introduction to Orthogonal Transforms: With Applications in Data Processing and Analysis*. Cambridge University Press, 2012.

[48] Norbert Wiener. Extrapolation, interpolation and smoothing of stationary. *Time Series, with Engineering Appli cations*, 1949.

[49] Jiunn-Lin Wu, Chia-Feng Chang, and Chun-Shih Chen. An improved richardson-lucy algorithm for single image deblurring using local extrema filtering. In *2012 International Symposium on Intelligent Signal Processing and Communications Systems*, pages 27–32. IEEE, 2012.

[50] Bo Zhang, Josiane Zerubia, and Jean-Christophe Olivo-Marin. Gaussian approximations of fluorescence microscope point-spread function models. *Appl. Opt.*, 46(10):1819–1829, Apr 2007.

Appendix A

Data generation

A.1 Samples preparation

Commercial AgBh powder (Thermo Fisher Scientific) was used without any further purification. 1,2-dioleoyl-sn-glycero-3-phosphoethanolamine (DOPE) was purchased from Avanti Polar Lipids Inc. The lipids were dissolved in water (DDW), total lipid concentration was 30 mg/ml per sample. Samples were homogenized using a vortexer for 5 minutes at 3000 RPM. Samples were then placed in quartz capillaries (1.5 mm in diameter), containing about $40\mu l$.

A.2 SAXS measurement setup

Measurements were performed using a lab-based X-ray scattering system, with a GeniX (Xenocs) low divergence Cu K_α radiation source (wavelength of $\lambda = 1.54$Å) and a scatterless slits setup [25]. The beam's full width half maximum (size) is $1mm$ in diameter, and the measured divergence is 0.0372 degrees (2θ). Samples were measured at distance of $d_s = 117mm$ using Pilatus 300K detector (Dectris), having pixel size of $172 \times 172\,\mu m^2$. The detector, sample stage, and slits were motorized using stepper motors with a positioning accuracy of $1\mu m$ and were controlled by SPEC software. In all the measured images, only valid pixels were used. Gaps between

the separate detector's modules were removed in post-processing. All the scattering patterns that are presented in Section 3.3 were cropped to show the relevant area.

For the r axis of the radial profiles (Eq. 3.2), the connection between q $[nm^{-1}]$ and r $[pixels]$ is given by [11]

$$q = \frac{4\pi}{\lambda} \sin\Big(\frac{1}{2} \arctan\big(\frac{Pr}{d_s}\big)\Big),$$

where P is the size of the pixel, and λ and d_s are given in the previous paragraph. However, as $\frac{P}{d_s} \propto 10^{-3}$, we use the fact that $\sin(\frac{1}{2}\arctan(x)) = \frac{x}{2} + o(x^3)$ [40], in order to approximate $q \approx \frac{4\pi}{\lambda}\frac{P}{2d_s}r$. This makes the linear coefficient of Eq. 3.24 to be

$$C = \frac{2\pi P}{\lambda d_s} \Big[\frac{nm^{-1}}{pixels}\Big],$$

from which C is calculated to be $C = 0.12$ $\frac{nm^{-1}}{pixels}$.

סופר רזולוציה בפיזור רנטגן בזווית קטנה מבוססת על שינוי פונקציית פיזור הנקודה ותזוזה תת פיקסלית של המצלמה

חיבור זה הוגש כחלק מהדרישות לקבלת התואר
באוניברסיטת תל-אביב .M.Sc - "מוסמך אוניברסיטה"
החוג למתמטיקה שימושית

על ידי
בנימין גוטמן

העבודה הוכנה בהדרכתו של
פרופסור יואל שקולניצקי

תקציר:

פיזור רנטגן בזווית קטנה (Small-angle X-ray scattering - SAXS) היא טכניקה המאפשרת אפיון ננוסקופי של דגימות שונות בתנאים שונים. בניסוי SAXS מקרינים דגימה באמצעות קרן רנטגן, וכתוצאה מכך נוצרת תבנית פיזור דו מימדית. למרבה הצער, דפוס פיזור זה מעוות על ידי פונקציית התפשטות נקודה (point spread function - PSF), שניתן לשלוט בה במהלך הניסוי.

בתזה זו אנו מציעים אלגוריתם של דה-קונבולוציה שלוקח דפוסי פיזור מרובים, כל אחד מהם נרכש עם PSF אחר, ומשחזר את דפוס הפיזור הבסיסי שלהם. האלגוריתם מבוסס על שחזור התמונה המקורית כבעיית מזעור, מוצג שיש לו פיתרון ייחודי, המתקבל בשיטת שיפועי הצמידה (conjugate gradients). בנוסף, אנו מציעים לשלב את האלגוריתם שלנו באלגוריתם סופר-רזולוציית תת-פיקסלי [36]. אנו בוחנים את האלגוריתם שלנו באמצעות סימולציות נומריות וכן באמצעות מדידות SAXS ניסיוניות.