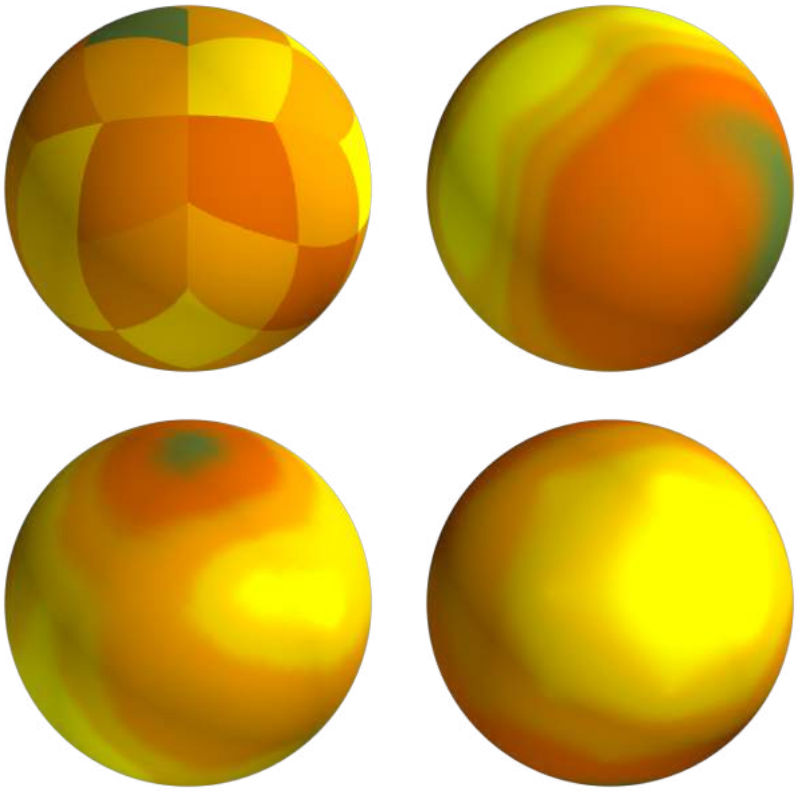


Local Representations for Small-angle X-ray Scattering Tensor Tomography

Radial Basis Functions and Spherical Pixels as Representations for Reciprocal Space Maps

Master's thesis in Physics

LUDVIG LINDAHL

DEPARTMENT OF PHYSICS
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023
www.chalmers.se

Master's thesis 2023

Local Representations for Small-angle X-ray Scattering Tensor Tomography

Radial Basis Functions and Spherical Pixels as Representations for Reciprocal Space Maps

LUDVIG LINDAHL

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Physics
Division of Materials Physics
Liebi research group
Chalmers University of Technology
Gothenburg, Sweden 2023

Local Representations for Small-angle X-ray Scattering Tensor Tomography
Radial Basis Functions and Spherical Pixels as Representations for Reciprocal Space Maps
LUDVIG LINDAHL

Supervisor: Leonard Nielsen, Department of Physics
Examiner: Paul Erhart, Department of Physics

Master's Thesis 2023
Department of Physics
Division of Materials Physics
Liebi research group
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Reciprocal space maps represented with (left to right, top to bottom) spherical pixels, Gaussian radial basis functions, Matérn spherical splines and Wendland spherical splines.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2023

Local Representations for Small-angle X-ray Scattering Tensor Tomography
Radial Basis Functions and Spherical Pixels as Representations for Reciprocal Space Maps
LUDVIG LINDAHL
Department of Physics
Chalmers University of Technology

Abstract

In small-angle X-ray scattering tensor tomography, a reconstruction of the reciprocal space map of a sample is performed using small-angle X-ray scattering measurements, providing information about the local reciprocal space in the volume of the sample. Previous methods of reconstruction have posed it as an inverse problem in linear and non-linear formulations using spherical harmonic representations of the reciprocal space map, solving it using quasi-Newton methods. Local representations have advantages, such as making the problem separable and allowing for positivity constraints on the reconstructed spherical functions. In this work, two local representations, one based on spherical pixels and one based on radial basis functions (including spherical splines), were implemented. The studied radial basis functions were based on Gaussian, Matérn and Wendland functions. In addition, regularization using the L1-norm, the L1-norm and the total variation were implemented. After selecting hyperparameters for the various reconstruction methods using the L-curve method and minimization of the residual, comparisons of the local representation methods and an existing method using a real spherical harmonic representation were carried out on simulated data as well as a trabecular bone sample. Studying the correlation between the reconstructed reciprocal space map and the true coefficients, the radial basis function representations were shown to perform comparably to the real spherical harmonics representation. Increases in correlation were seen under the application of regularization terms. The correlation between neighboring voxels of the orientation of features increased under regularization, where total variation regularization was shown to perform similarly to L1-regularization.

Keywords: small-angle X-ray scattering, small-angle X-ray scattering tensor tomography, tensor tomography, spherical spline, radial basis function, total variation, spherical pixel, HEALPix

Acknowledgements

I want to thank my supervisor Leonard Nielsen for all his guidance and input over the course of the project and for being present and responsive all the way through. I also want to thank Paul Erhart for handling administrative details and for his thorough proofreading. Thanks to the condensed matter physics division for lending me an office space and for the coffee breaks. And thanks to David Dagson for your company in the office. Last but not least, I want to thank Marianne Liebi and everyone at the Paul Scherrer Institute their hospitality and for being helpful during my time there.

Some of the results in this thesis have been derived using the healpy and HEALPix package.

Ludvig Lindahl, Gothenburg, June 2023

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

HEALPix	Hierarchical Equal Area isoLatitude Pixelation
L-BFGS-B	Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm with Boundaries
RBF	Radial Basis Function
RSH	Real Spherical Harmonic
RSM	Reciprocal Space Map
SAXS	Small-angle X-ray Scattering
SAXSTT	Small-angle X-ray Scattering Tensor Tomography
SH	Spherical Harmonic
SIGTT	Spherical Integral Geometric Tensor Tomography
TV	Total Variation

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Parameters

ϵ	Scale parameter
ϵ_{rel}	Relative scale parameter

Variables

$\mathbf{p}, \mathbf{q}$	Point on the unit sphere
M	Reciprocal space map
a_i	Coefficient of the basis function whose index is i
A_i	Integral of a_i along integration lines
i, j, q, s	Indices
j, k	Sample displacement coordinates

Functions

B_i	Basis function with index i
Y_l^m	Complex spherical harmonic of order l and degree m
Y_{lm}	Real spherical harmonic of order l and degree m
P_i	Pixel basis function with index i
K	Kernel function
D	Great-circle distance function
D_e	Euclidean distance function
$\mathbb{P}$	John transform
$\mathbb{D}$	Detector segment projection transform

Contents

List of Figures

List of Tables

1
Introduction

Small-angle X-ray Scattering (SAXS) is a technique used to study nano-scale features of samples by probing their Reciprocal Space Map (RSM). The technique is often used in materials science, biology and related fields [1]. A sample is illuminated with X-rays which are scattered, mainly by the electrons in the sample. The scattered component of the beam produces a diffraction pattern which is proportional to the Fourier transform of the auto-correlation function of the electron density. The diffraction pattern informs us of the reciprocal space of the sample, allowing for a characterization of features in various length scales. For small angles, less than around 5°, the features probed are in the scale of about 5 nm to 100 nm [2]. The information provided by a two-dimensional diffraction pattern represents an average of the interaction of the beam with the sample and can usually only be used to probe features orthogonal to the beam [2].

Scanning SAXS builds on SAXS by repeating measurements in a raster scan, providing structural information in each volume element of the two-dimensional grid, a limited reconstruction of the sample [3, 4].

Reconstruction techniques, long used in for example medical imaging with magnetic resonance imaging and computed tomography, can be used to reconstruct three-dimensional representations of a sample from many projections [5]. Examples of reconstruction techniques include direct back-projection, filtered back-projection as well as simultaneous algebraic reconstruction. Tomography refers to an imaging technique where a penetrating wave is used to probe a sample which is then reconstructed, providing information about the three-dimensional inner structure of the object. The name tomography implies that the technique provides information about slices of a sample.

A particular variant of tomography uses the information provided by SAXS measurements to perform a reconstruction which may be represented as a three-dimensional grid of volume elements, or voxels (the volumetric equivalent of pixels). Since every SAXS measurement provides information about a two-dimensional cross-section of the RSM of the sample, utilization of this data should allow for an approximate reconstruction of the reciprocal space of every voxel, the RSM.

This is the principle behind the technique known as Small-angle X-ray Scattering Tensor Tomography (SAXSTT). The reciprocal space of voxel is a function that needs to represented in some finite basis. This representation can be expressed as a tensor, and the reconstructed field is a tensor field. In one particular implementation of SAXSTT the reconstructions are constrained to

an approximately fixed magnitude scattering vector [6, 7, 8]. Under this constraint, the function that describes the reciprocal space of a voxel is a function on the surface of a sphere. Although there are several bases used for representing spherical functions, the most well-known and frequently used is the Spherical Harmonic (SH) basis. This is the canonical representation of band-limited spherical functions. Advantages of using this representation include orthogonality, convenient calculation of statistical quantities like the variance and covariance and symmetry assumptions being easily implemented. There are also some disadvantages. For one, it is not possible to enforce positive bounds on the coefficients since a general function on the sphere could not be represented by a non-negative linear combination of SH. Secondly, SHs are global in the sense of being non-compact. In a local basis of compact functions, the problem of reconstructing different regions of the RSM would be separable in the sense that different parts of reciprocal space could be reconstructed separately. Lastly, SH representation are band-limited, as mentioned above, which is not a property that we expects the reciprocal space of most materials to have. For these reasons, other representations are of interest.

The aim of this project is to investigate different types of local representations and to explore the benefits and limitations of such representations. A few examples of local representations include, spherical pixels or a basis of Voronoi cells, spherical splines, Slepian functions and spherical wavelets [9]. The first two are the subject of this thesis. They were implemented in Python for use with the MUMOTT Python package that implements a method for SAXSTT called Spherical Integral Geometric Tensor Tomography (SIGTT).

Other than implementing these representations and comparing them with SH, another goal of this project is to implement regularization terms and evaluate them. When solving inverse problems, regularization is usually indispensable since such problems are ill-posed, with no unique solution. Regularization terms are thus used for encoding certain prior knowledge and assumptions, such as the assumption that adjacent voxels should be correlated.

The project does not include any significant modification of the algorithm used for solving the reconstruction problem but rather focuses more specifically on the representations and the regularization terms. There are other limitations, such as those set by the available computational resources. The resolution in reciprocal space has for the most part been kept quite low for this reason.

2

Background

Prior to the discussion of implementation details and results, a brief theoretical introduction to the subject of SAXSTT and the canonical representation of RSMs will be given, starting with an overview of the technique and the mathematical formulation of it as an inverse problem. Furthermore, some details on the SH basis are recounted.

2.1 Small-angle X-ray Scattering Tensor Tomography

SAXS is a technique that probes the reciprocal space of a sample. SAXSTT uses SAXS to gather data about the reciprocal space of a sample in different orientations. The sample is rotated (β) and tilted (α) to provide a three-dimensional view of its reciprocal space. Furthermore, a raster scan is performed at each orientation. An illustration of the experimental setup, sample orientation can be seen in Figure 2.1. An illustration of a raster scan is found in Figure 2.2. The data acquired using this methodology can be used to perform a reconstruction of the reciprocal space in each point of the volume containing the sample. Before being used for a reconstruction, the measured patterns are divided into circular segments and the intensity integrated for each segment.

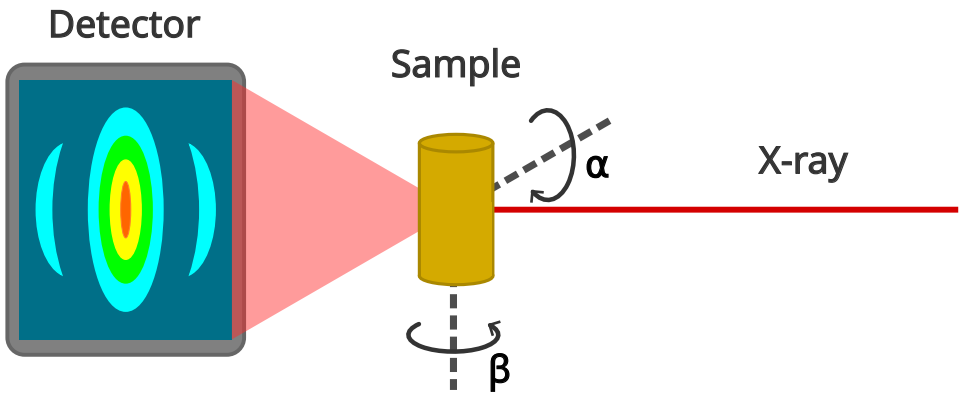
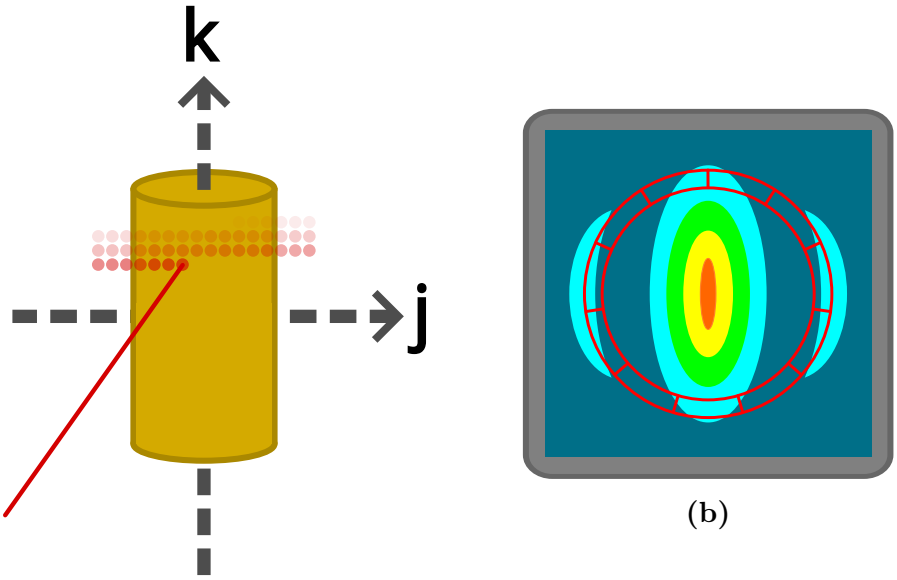


Figure 2.1: A simplified sketch of the experimental setup, illustrating how the sample is oriented with a tilt α and a rotation β.



(a) (b)

Figure 2.2: **(a)** A sketch of the raster scan, illustrating how the sample is translated orthogonally to the direction of the X-rays and measurements taken at points in a grid. **(b)** The measured SAXS patterns are segmented at an approximately constant magnitude scattering vector.

In practice, the data only allows for an approximate reconstruction, with a finite number of degrees of freedom both in real space and in reciprocal space. As such, a model needs to be assumed to account for these constraints. The model conventionally used, assumes a volumetric grid, a so-called voxel grid, over real space. The value associated with each voxel is a representation of reciprocal space, usually for a constant magnitude of the q-vector. Under this constraint, the probed part of reciprocal space is a spherical function as illustrated in Figure 2.3. [6, 7, 4]

The spherical function in reciprocal space, or the so-called RSM, can be represented in a number of bases. The conventional representation is of course SHs, in particular the so-called Real Spherical Harmonic (RSH) basis is especially suited for this task. However, there are a number of drawbacks to using a global representation like SHs as discussed in chapter 1.

2.1.1 Reconstruction

Following Nielsen [10, 8], for a basis on the sphere comprised of a set of basis functions $\{B_i\}_{i=1}^{n}$, with $\mathbf{r} = (x, y, z)$ as the spatial position and $\mathbf{p} = (\theta, \phi) \in \Omega$ as the position along the surface of the sphere which is the RSM, the RSM $M(\mathbf{r}, \mathbf{p})$ can be approximated as follows,

$$M(\mathbf{r}, \mathbf{q}) = \sum_{i=1}^{n} a_i(\mathbf{r}) B_i(\mathbf{p}) + E(\mathbf{r}, \mathbf{p}), \tag{2.1}$$

where $E(\mathbf{r}, \mathbf{p})$ is the approximation error and we have assumed that the amplitude a_i is the only parameter to be determined. This basis could be the basis of SHs, radial basis functions or any other appropriate basis for representing spherical functions. Equation (2.1) is the model of the RSM and the coefficients $\{a_i(\mathbf{r})\}_{i=1}^{n}$ are the reconstruction coefficients.

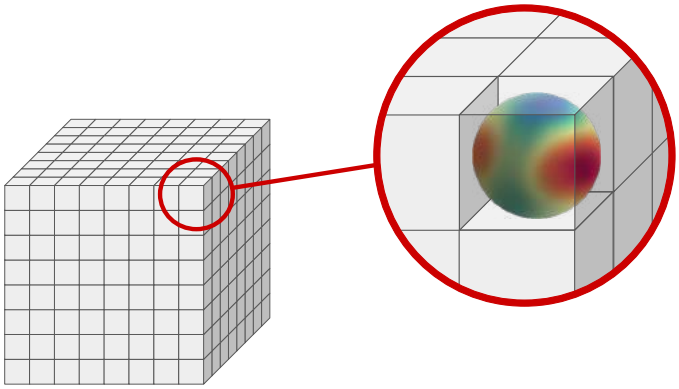


Figure 2.3: An illustration of the voxel grid, where the value of each voxel is a spherical function, the RSM, represented in some spherical basis.

The John transform (or X-ray transform) can be understood as a generalization of the so-called Radon transform [5]. In this generalized case, it projects a three-dimensional function onto a two-dimensional surface in a direction defined by the parameters α and β. When applied to an RSM, it models the X-ray scattering onto the detector plane. The coordinates j and k along the surface are the coordinates of the sample displacement in the raster scan. The John transform of a function $f(\mathbf{r})$ is

$$\mathbb{P}[f](j, k, \alpha, \beta) = \int_{-\infty}^{\infty} f(\mathbf{v}(j, k, \alpha, \beta) + s\mathbf{u}(\alpha, \beta))\, \mathrm{d}s, \tag{2.2}$$

where $\mathbf{u}$ is a vector parallel to the line of integration and $\mathbf{v}$ is a displacement vector orthogonal to that line [8] . Applying the John transform to $M(\mathbf{r}, \mathbf{p})$ we get,

$$\mathbb{P}[M](\mathbf{p}, j, k, \alpha, \beta) = \sum_{i=1}^{n} B_i(\mathbf{p}) \int_{-\infty}^{\infty} a_i(\mathbf{v}(j, k, \alpha, \beta) + s\mathbf{u}(\alpha, \beta))\, \mathrm{d}s = \tag{2.3}$$

$$= \sum_{i=1}^{n} B_i(\mathbf{p}) A_i(j, k, \alpha, \beta) = [B_i A_i]\,(\mathbf{p}, \mathbf{s}), \tag{2.4}$$

where the last step is expressed in Einstein notation and where we have introduced the notation $\mathbf{s} = (j, k, \alpha, \beta)$ for brevity and to avoid confusion between the coordinate j and the index j later on. in Figure 2.4, the application of the John transform to a voxel grid is illustrated. We can conclude from Eq. 2.4 that applying the transform to each coefficient and applying the inner

product to the integrated coefficients and the basis functions is equivalent to transforming the RSM $M(\mathbf{r}, \mathbf{p})$ itself.

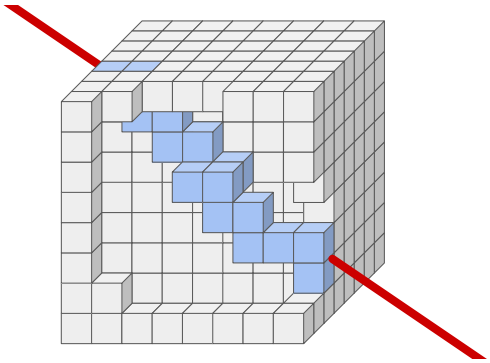


Figure 2.4: An illustration of the John transform of the voxel model at a certain sample orientation (α, β) and at a certain coordinate (j, k) in the projection plane.

Using the John transform, we can model the X-ray scattering onto the detector plane. What we get after applying the transform to our coefficients $a_i(\mathbf{r})$ is a coefficient vector A_i which represents a three-dimensional RSM in our chosen basis. However the measured SAXS patterns are two-dimensional. The part of reciprocal space probed by SAXS is the plane through the origin, orthogonal to the direction of the incident ray.

Keeping α, β, j and k fixed and given a fixed magnitude scattering vector $|q|$, the parameter $\mathbf{p} \in \Omega$ will be constrained to a circle $c(\varphi) \in \Omega$ in reciprocal space corresponding to the segment circle in the detector plane illustrated in Figure 2.2b. So, in that circle, a SAXS image will correspond to a cross-section of the spherical function $[B_i A_i]\,(c(\varphi))$ describing the RSM.

Applying the same segmentation as applied to the measured SAXS patterns to our projected model, we split the great-circle into segments $\{S_i\}_{i=1}^{m}$. For each segment, we determine a mean intensity from the representation of the sphere by numerically integrating each basis function over that segment,

$$\mathbb{D}_{ji} = \mathbb{D}[B_i](S_j) = C_j \int_{S_j} B_i(\mathbf{p})\, \mathrm{d}l(\mathbf{p}) \tag{2.5}$$

where C_j is a normalization constant chosen according to a normalization scheme. To get the theoretically expected intensity in that segment from the vector of coefficients that we get from the John transform A, we simply sum up the contributions of each basis function,

$$I_j^{\mathrm{th}}(j, k, \alpha, \beta) = [\mathbb{D}_{ji} A_i]\,(\mathbf{s}), \tag{2.6}$$

where I_j^{th} is the intensity in segment $j = 1, 2, ..., m$, D_{ji} is a $m \times n$-matrix and A_i is an n-vector.

We may now define a residual using experimentally determined intensities $I_j^{\mathrm{exp}}(\mathbf{s})$ which are transmission corrected such that

$$I_j^{\mathrm{N}}(\mathbf{s}) = \left[I_j^{\mathrm{exp}} / I^{\mathrm{T}}\right](\mathbf{s}), \tag{2.7}$$

where $I^{\mathrm{T}}(\mathbf{s})$ is the transmitted intensity. The squared residual is,

$$\mathcal{R}_j(\mathbf{s}) = \left[I^{\mathrm{th}} - I^{\mathrm{N}}\right]_j^2(\mathbf{s}). \tag{2.8}$$

This residual term is a vector over the circle segments and over multiple measurements. Introducing an index over measurements s such that $\mathbf{s} = (j, k, \alpha, \beta)$ can be indexed as $(j_p, k_p, \alpha_p, \beta_p)$, we write the objective function as follows,

$$\mathcal{O} = \sum_j \sum_s w_{js} \mathcal{R}_{js} + \Lambda, \tag{2.9}$$

where Λ represents the regularization terms. More details on them is presented in section 3.3 as certain other details will be needed first. By minimizing the objective function with respect to the set of coefficients $a_i(\mathbf{r})$, a reconstruction can be attained by

$$a_i^*(\mathbf{r}) = \arg\min_{a_i(\mathbf{r})} \mathcal{O}(a_i(\mathbf{r})). \tag{2.10}$$

Since in the model, the volume to be reconstructed is divided into a voxel grid, $a_i(\mathbf{r})$ will be constant for all $\mathbf{r}$ belonging to a certain voxel. We can then discretize even the spatial coordinate and instead use an index q refering to a voxel,

$$a_{iq}^* = \arg\min_{a_{iq}} \left[\sum_j \sum_s w_{js} \left[[\mathbb{D}_{ji} A_i]_s - I_{js}^{\mathrm{N}} \right]^2 + \Lambda \right], \tag{2.11}$$

where

$$A_i = \left[\mathbb{P}[a_i(\mathbf{r})]\right]_s = \sum_q v_q a_{iq}, \tag{2.12}$$

for some weights v_q. This minimization problem is an example of an inverse problem, where observations of some phenomenon are used to reconstruct the thing that caused them, using a model. There are many methods used for solving such problems. In SIGTT, quasi-newton methods are used and in this project built on SIGTT, the method known as Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm with Boundaries (L-BFGS-B) [11, 12] has been used.

2.1.2 Sampling Distribution in Reciprocal Space

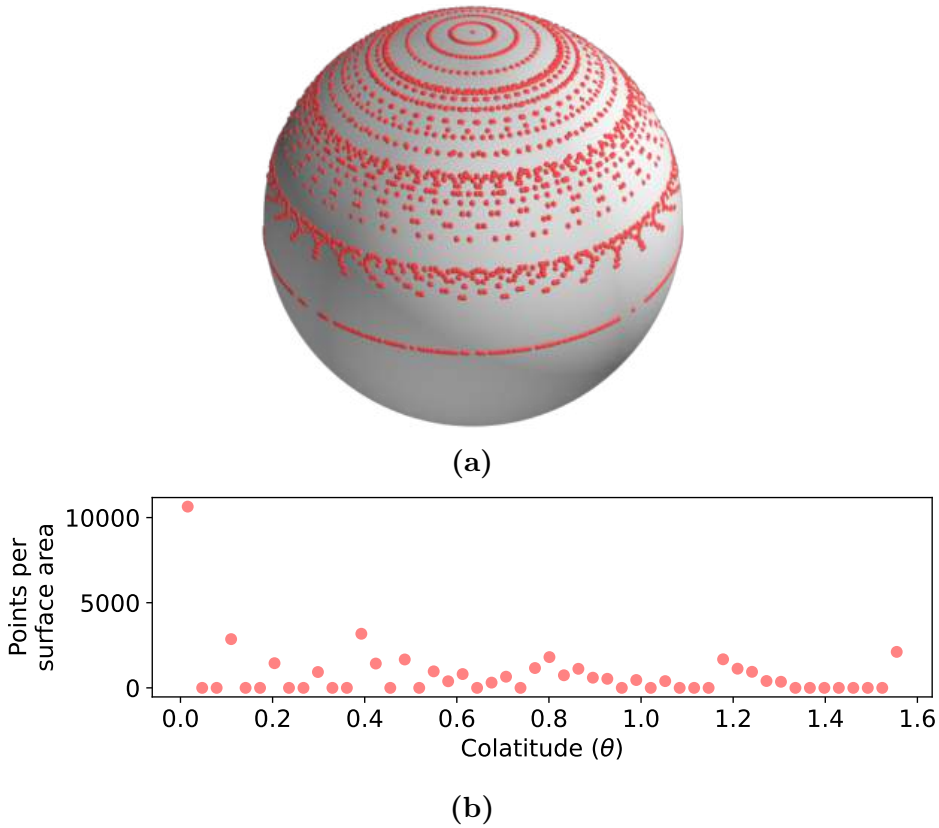


Figure 2.5: Due to the missing wedge problem the density of samples varies over reciprocal space. Generally, there is more data per surface element closer to the poles. **(a)** The colatitude θ of the center position of each segment, plotted on a sphere. These segments belong to different projection lines but the plot serves to illustrate that the density of angles is greater closer to the poles. Due to point symmetry, points may be mirrored in the origin and have the same effect on the reconstruction, explaining the empty segment close to the south pole. **(b)** The number of points per surface area to θ corresponds to the visualization above.

Figure 2.5 illustrates the distribution of probed points in reciprocal space. Each point on the sphere corresponds to the center point of a segment on the detector. It should be noted that not every point is probed for every voxel, rather the plot is the complete set of points in reciprocal space, neglecting spatial coordinates. Still, the plot clearly shows a sampling bias close to the poles of the sphere. This is a consequence of experimental limitations. The sample must be mounted on a sample holder that would block the X-rays at certain tilt intervals. For this reason, not all points in reciprocal space can be measured at all sample orientations at which they would be accessible. This leads to the so-called missing wedge problem in tomographic reconstruction.

2.2 Spherical Harmonics

SHs is a basis conventionally used for representing functions on the sphere, specifically the 2-sphere in our case (i.e., the three-dimensional sphere), defined as a set of eigenfunctions of the Laplace-Beltrami operator Δ^* [13, 9],

$$\Delta^* Y_l^m = -l(l+1)Y_l^m, \tag{2.13}$$

where Y is a SH of degree l with geometric multiplicity, or order, $-l \leq m \leq l$. The Laplace-Beltrami operator is related to the Laplace operator Δ by

$$\Delta = \left(\frac{\partial}{\partial r}\right)^2 \frac{2}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\Delta^*, \tag{2.14}$$

in a sense representing the angular part of the Laplace operator in the case of the unit sphere. In general, it generalizes the Laplace operator to functions on surfaces. The set of SHs is a restriction of the set of homogeneous harmonic polynomials $P \in \mathbb{R}^3$, satisfying Laplace's equation $\Delta P = 0$.

There are multiple conventions for normalization, leading to multiple different definitions of the complex SH functions. The following expression follows the orthonormalized convention [14],

$$Y_l^m(\theta, \phi) = \bar{P}_l^m(\cos(\theta)) \exp(im\phi), \tag{2.15}$$

$$\bar{P}_l^m(u) = \sqrt{\frac{(2l+1)}{4\pi}\frac{(l-m)!}{(l+m)!}} P_l^m(u), \tag{2.16}$$

$$P_l^m(u) = (-1)^m (1-u^2)^{m/2} \frac{\mathrm{d}^m}{\mathrm{d}u^m} P_l(u), \tag{2.17}$$

$$P_l(u) = \frac{1}{2^l l!}\frac{\mathrm{d}^l}{\mathrm{d}u^l}(u^2-1)^l, \tag{2.18}$$

where P_l^m is the associated Legendre function related to the ordinary Legendre function P_l.

However, these functions are complex and as such not necessarily suited as a basis for real functions on the sphere. A RSH basis can be expressed in terms of complex SHs with the above definition. Using this alternate approach simplifies the representation of RSMs. The relationship between complex SHs and RSHs is

$$Y_{lm} = \begin{cases} \sqrt{2}\,\Re[Y_l^m], & \text{for } m > 0, \\ Y_l^0, & \text{for } m = 0, \\ \sqrt{2}\,\Im[Y_l^{|m|}], & \text{for } m < 0, \end{cases} \tag{2.19}$$

where Y_{lm} is a RSH function (note the difference in notation) and the Condon-Shortley phase $(-1)^m$ has been left out. However, in SIGTT the so-called 4π-normalization was used, removing the $(4\pi)^{-1/2}$ factor from the definition of $\bar{P}_l^m$. Lastly, the functions to be represented are all point symmetric. This condition can be enforced by excluding all the odd degrees $l = 2n+1$.

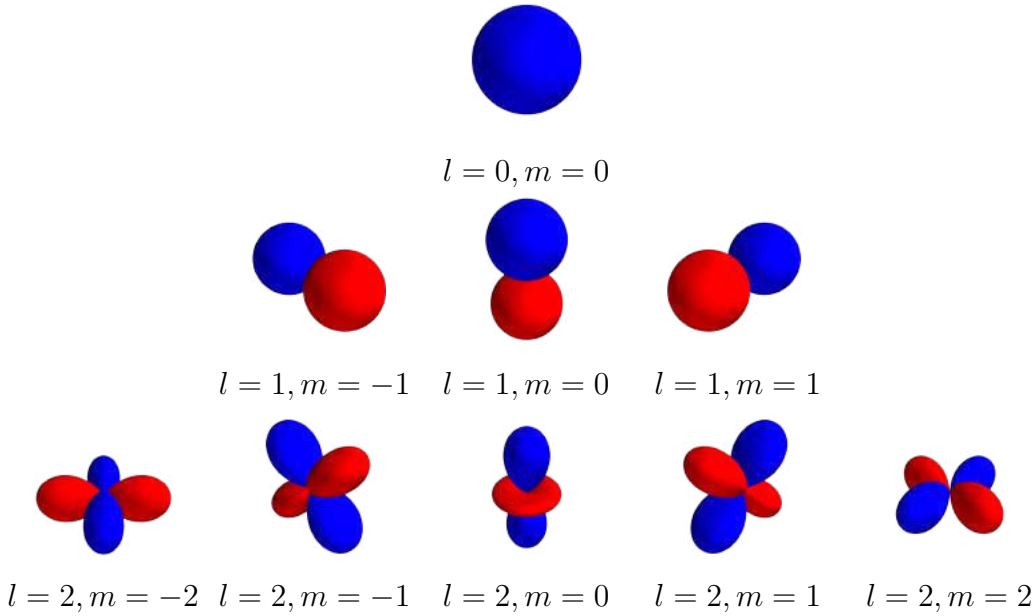


Figure 2.6: RSHs up to degree $l = 2$.

Having defined the RSH basis, we can now define a spherical Fourier decomposition of a spherical function $f(\theta, \phi)$ [15],

$$f(\theta, \phi) = \sum_{l \in \mathbb{N}_0} \sum_{m=-l}^{l} \hat{f}_{lm} Y_{lm}(\theta, \phi), \tag{2.20}$$

with Fourier coefficients

$$\hat{f}_{lm} = \frac{1}{4\pi} \int_{\Omega} f(\theta, \phi) Y_{lm}(\theta, \phi) \, \mathrm{d}\omega, \tag{2.21}$$

where Ω is the surface of the unit sphere. Under the assumption of point symmetry, the Fourier coefficients for odd degrees will be zero. An analogous Fourier decomposition exists for complex SHs.

In practice, it is usually not possible to determine Fourier coefficients analytically. The purpose of a discrete Fourier transform is to determine the Fourier coefficients of a function by sampling it discretely on some grid. Such a method is described by Driscoll and Healy [15], who also developed a fast algorithm for computing such a transform and contributed a sampling theorem akin to the well known Nyquist-Shannon sampling. Using their method, the Fourier coefficients of a band-limited function f for which $\hat{f}_l^m = 0$ with $l \geq b$ are determined by

$$\hat{f}_l^m = \frac{\sqrt{2\pi}}{2b} \sum_{j=0}^{2b-1} \sum_{k=0}^{2b-1} a_j^{(b)} f(\theta_j, \phi_k) \overline{Y_l^m}(\theta_j, \phi_k), \tag{2.22}$$

where $\overline{Y_l^m}$ is the complex conjugate of Y_l^m and where the weights $a_j^{(b)}$ are given by

$$a_j^{(b)} = \frac{2\sqrt{2}}{2b} \sin\left(\frac{\pi j}{2b}\right) \sum_{l=0}^{b-1} \frac{1}{2l+1} \sin\left([2l+1]\frac{\pi j}{2b}\right), \tag{2.23}$$

for $j = 0, 1, ..., 2b-1$ and $2b$ is a power of two. The sample points are given by $\theta_j = \pi j/2b$ and $\phi_k = \pi k/b$ for $j = k = 0, 1, ..., 2b-1$.

There are multiple ways of defining a convolution on the sphere, with different formulations in SHs [16]. The conventional definition is the isotropic convolution which has the following form in complex SHs,

$$(f * g)_l^m = \sqrt{\frac{4\pi}{2l+1}} \hat{f}_l^m \overline{\hat{g}_l^0}. \tag{2.24}$$

This will not be used explicitly in this thesis but is useful to know in chapter 3. Another useful equation is the following addition formula that allows the Dirac delta function on the sphere to be expressed in the following manner [16],

$$\delta_{\theta',\phi'}(\theta, \phi) = \sum_{lm} \overline{Y_l^m}(\theta', \phi') Y_l^m(\theta, \phi). \tag{2.25}$$

2.2.1 Correlation between Reciprocal Space Maps

Using the Fourier transform above, we may transform RSMs represented in other bases to the RSH basis. This is very useful for comparing different RSMs since the variance and covariance between two spherical functions in RSH can be expressed using the power spectrum and the cross power spectrum, respectively. The cross power spectrum is given by

$$S_l(f, g) = \sum_{m=-l}^{l} \hat{f}_{lm} \hat{g}_{lm} \tag{2.26}$$

under the 4π-normalization [14]. Setting $f = g$ we get the power spectrum $S_l(f, f)$. Furthermore, the total cross power per surface area fulfills the equality,

$$\frac{1}{4\pi} \int_\Omega f(\theta, \phi) g(\theta, \phi) \, \mathrm{d}\omega(\theta, \phi) = \sum_{l=0}^{\infty} S_l(f, g). \tag{2.27}$$

The definitions of variance and covariance on the sphere follow by substituting f and g by $f - \bar{f}$ and $g - \bar{g}$ where the bar represents the mean which has an analogous definition. It is easy to show that the variance is

$$\sigma_f^2 = \sum_{l=1}^{\infty} S_l(f, f) \tag{2.28}$$

and the covariance similarly,

$$\mathrm{cov}(f, g) = \sum_{l=1}^{\infty} S_l(f, g). \tag{2.29}$$

The Pearson correlation provides a way of comparing reconstructions, either with other reconstructions of the same sample or with some set of true coefficients describing the RSM in the case of simulated measurement data. Its definition is as follows

$$\rho_{fg} = \frac{\mathrm{cov}(f, g)}{\sigma_f \sigma_g}, \tag{2.30}$$

where f and g are RSMs with standard deviations σ_f and σ_g and covariance $\mathrm{cov}(X, Y)$. In this project, the Pearson correlation is only employed under the RSH representation, usually between reconstruction coefficients and true coefficients in the context of simulated data. Other representations are first transformed into a RSH representation before the correlation measure is applied.

3

Local Representations

Local representations have some benefits over SHs. By choosing positive basis functions, it becomes simple to enforce positive bounds for the reconstruction coefficients and the RSM is guaranteed to be positive, which is a reasonable prior assumption to make. Local representations also decouple different parts of the sphere, making the reconstruction problem separable. In general, the SH spectrum of local basis functions is not band-limited. This may allow for a more accurate reconstruction since generally, the RSM of a sample cannot be assumed to be band-limited. In this chapter, we will cover various theoretical aspects of local representations and their implementation, including distance functions, spherical point grids and spherical splines. The latter lays a foundation for Radial Basis Functions (RBFs), which is one of the two representations investigated in this thesis, along with spherical pixels.

3.1 Distance on the Sphere

Before a local representation can be defined, a metric or distance function for measuring the distance between points on the sphere needs to be selected. The so-called great-circle distance or geodesic distance is the natural metric [17], for which the distance between two points is defined as the angle between the two points in a circle, multiplied by the radius of the sphere.

The great-circle distance is the shortest possible distance between two points along the surface of a sphere. In the case of the unit sphere it is simply the angle between the vectors going from the origin to each of the points. The distance between two points $\mathbf{p}, \mathbf{q}$ on a sphere with radius R is [18, page 221],

$$D(\mathbf{p}, \mathbf{q}) = R\cos^{-1}\left(\frac{\mathbf{p}\cdot\mathbf{q}}{R^2}\right) = R\cos^{-1}\left(\frac{R^2\cos(\theta)}{R^2}\right) = R\theta, \tag{3.1}$$

where θ is the angle between the vectors as described above. In Figure 3.1, an illustration of the great-circle distance between two points on the sphere can be seen.

Similarly, the Euclidean distance between two points can be defined as

$$D_e(\mathbf{p}, \mathbf{q}) = \sqrt{(\mathbf{p}-\mathbf{q})\cdot(\mathbf{p}-\mathbf{q})} = \tag{3.2}$$

$$= \sqrt{\mathbf{p}\cdot\mathbf{p} - 2\mathbf{p}\cdot\mathbf{q} + \mathbf{q}\cdot\mathbf{q}} = \sqrt{2R^2 - 2\mathbf{p}\cdot\mathbf{q}} \tag{3.3}$$

$$= 2R\sin\left(\frac{\theta}{2}\right), \tag{3.4}$$

for which $2R\sin(\theta/2) \approx R\theta$ when $\theta \gtrapprox 0$. So, as long as the distances in question are small, the Euclidean distance is a good approximation for the great-circle distance. Since the Euclidean distance can be calculated from the coordinates of two points without using trigonometric functions, there could be a computational benefit to using it instead of the great-circle distance in some cases. Another reason for using the Euclidean distance is that it is usually easier mathematically if, for example, the integral of a spherical function needs to be evaluated.

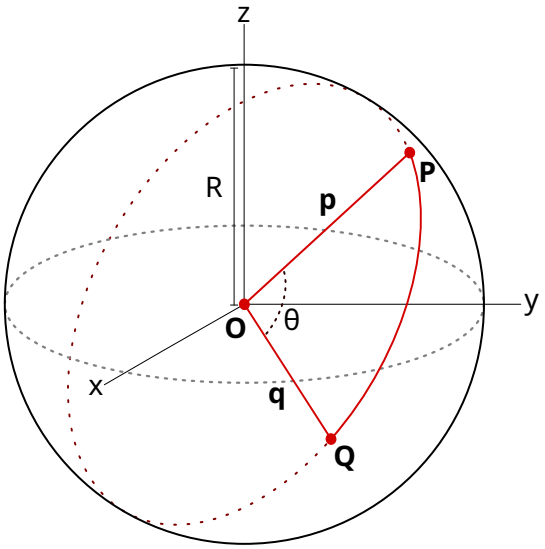


Figure 3.1: An illustration of the great-circle distance function applied to points **P** and **Q**, pointed to by vectors **p** and **q** centered in the origin. The great-circle distance is determined from the angle θ between the points, in the circle that intersects both of the points and is centered in the origin. The great circle distance is thus $R\theta$ or just θ on the unit sphere.

3.2 Discretization Schemes

For the representations investigated, a discretization of the unit sphere into a point grid is a prerequisite. Before the sphere can be represented using pixels, a point grid from which Voronoi cells, as detailed in subsection 3.2.1, can be determined is necessary. These cells would correspond to the pixels. As for the splines which are implemented as radial basis functions, these basis functions need to be centered in a set of points. As we will see, it is generally ideal for these points to be evenly distributed. However, there are only a limited number of discretizations of the sphere that distribute points evenly, and all such sets of points are too small to be of practical use. Hence, using pseudo-even distributions is the only practical option. There are a number of methods for generating such point sets, such as Hierarchical Equal Area isoLatitude Pixelation (HEALPix)[1] [19] and the Fibonacci spiral method and

[1] http://healpix.sourceforge.net

various similar methods.

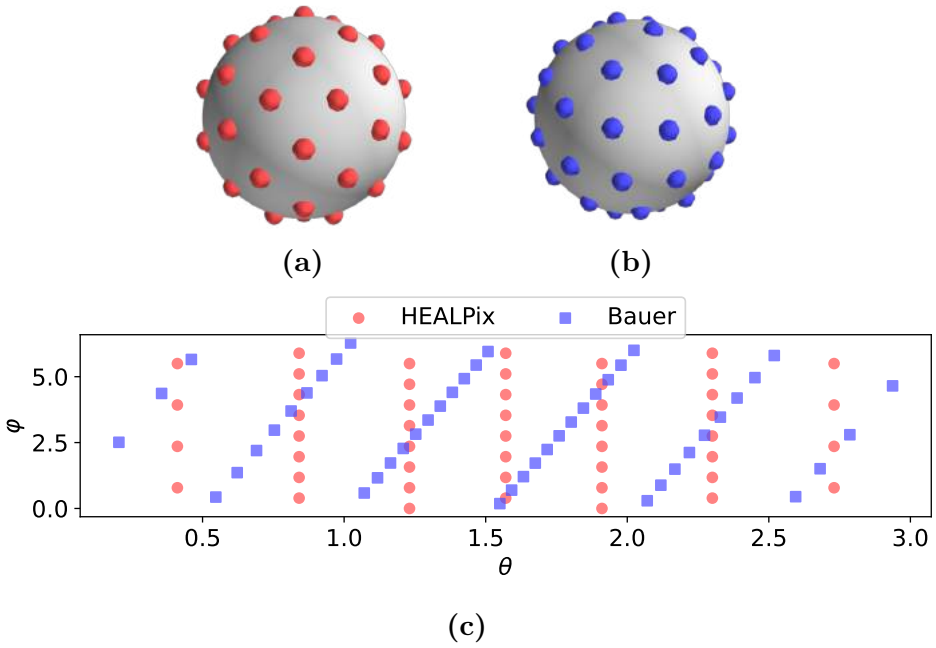


Figure 3.2: Illustration of different spherical point grids. **(a)** HEALPix. **(b)** Bauer. **(c)** The Cartesian projection of both point grids.

HEALPix is a discretization scheme that is conveniently implemented in the HEALPY Python package [20]. There are many benefits to using this scheme. The sphere is divided into equally sized cells, which are nested such that a higher-resolution grid can easily be mapped onto a lower-resolution grid, given that an appropriate indexing is assumed. This would be an advantage in particular for implementing spherical wavelets. One significant disadvantage is that HEALPix only works for certain numbers of points $N_{\text{point}} = 12N_{\text{side}}^2$ while retaining its useful properties, which puts some limits on the user. The parameter N_{side} refers to the number of segments along one side of one of the base pixels, of which there are twelve. Each base pixel, having four sides, will hence be divided into N_{side}^2 pixels.

Furthermore, HEALPix has octahedral symmetry that makes the enforcement of point symmetry in the pixel representation easily implemented and accurate. Many other discretization schemes do not allow for exact point symmetry since the points in these schemes are not distributed in a point symmetric manner.

Another such method is the Bauer spiral as described in [21]. This method is a type of Fibonacci spiral method, of which there are many. It can be

implemented using the following equations,

$$L = \sqrt{n\pi}, \tag{3.5}$$

$$z_k = 1 - \frac{2k-1}{n}, \tag{3.6}$$

$$\varphi_k = \cos^{-1}(z_k), \tag{3.7}$$

$$\theta_k = L\phi_k, \tag{3.8}$$

$$x_k = \sin(\varphi_k)\cos(\theta_k), \tag{3.9}$$

$$y_k = \sin(\varphi_k)\sin(\theta_k), \tag{3.10}$$

where $1 \leq k \leq n$. x_k, y_k and z_k are the coordinates of a set of n points.

3.2.1 Spherical Voronoi

Having decided on a spherical point grid, a corresponding set of cells can be determined using a spherical Voronoi diagram. Such a set of cells can be used in a pixel representation by binning a given point on the sphere and letting the value of a spherical function in that point be determined by what cell it is binned into.

In Voronoi diagrams and for a given set of points, a plane is partitioned such that each point in the plane, not necessarily belonging to the point set, belongs to the cell associated to its nearest neighbor in the point set. Spherical Voronoi diagrams then consist of the same kind of partition of points on the sphere, where the distance between points can be defined using the great-circle distance.

In the SCIPY Python package, there is a class SPHERICALVORONOI that allows for spherical Voronoi to be applied conveniently [22].

3.2.2 Spherical Splines

The following brief summary of the theory of spherical splines is based mostly on [13], as well as [9]. These basis functions are intimately tied to a certain spherical pseudo-differential operator by virtue of being the Green's functions of that operator. However, for a pseudo-differential operator to have a well defined set of splines it must be spline-admissible.

A pseudo-differential operator is a linear operator of the form,

$$\mathcal{D}[h] = \sum_{l \in \mathbb{N}_0} \hat{\mathcal{D}}_l \sum_{m=1}^{2l+1} \hat{h}_l^m Y_l^m, \tag{3.11}$$

where $\hat{h}_l^m$ is a Fourier coefficient of the function h. By comparing this equation with the definition of the convolution on the sphere, we see that this is essentially analogous to the application of a high-pass filter in the frequency domain of a time discrete signal.

The magnitude of the coefficients of the pseudo-differential operator are tightly bound to a power of the degree index,

$$|\hat{\mathcal{D}}_l| = \Theta(l^p), \tag{3.12}$$

where the spectral growth order of a pseudo-differential operator is p. As expected from a differential operator, the spectral growth order is positive $p \geq 0$, meaning that the application of the operator corresponds to the application of a high-pass filter. If the spectral growth order of an operator fulfills $p > 2$, that operator is spline-admissible.

The Moore-Penrose pseudo-inverse of a spherical pseudo-differential operator is given by

$$\mathcal{D}^{\dagger}[h] = \sum_{l \in \mathbb{N}_0, \hat{\mathcal{D}}_l \neq 0} \frac{1}{\hat{\mathcal{D}}_l} \sum_{m=1}^{2l+1} \hat{h}_l^m Y_l^m. \tag{3.13}$$

This is functionally the integral operator. We can now define the Green's function of $\mathcal{D}$ as

$$\Psi_{\mathbf{q}}^{\mathcal{D}} = \mathcal{D}^{\dagger} \delta_{\mathbf{q}}, \tag{3.14}$$

where $\delta_{\mathbf{q}}$ is a Dirac measure centered at $\mathbf{q} \in \Omega$. From this definition we get the following through the Fourier expansion with coefficients $\langle \mathcal{D}^{\dagger} \delta_{\mathbf{q}} | Y_l^m \rangle = \langle \delta_{\mathbf{q}} | \overline{\mathcal{D}^{\dagger}} Y_l^m \rangle$

$$\Psi_{\mathbf{q}}^{\mathcal{D}}(\mathbf{p}) = \sum_{l \in \mathbb{N}_0, \hat{\mathcal{D}}_l \neq 0} \frac{1}{\hat{\mathcal{D}}_l} \sum_{m=1}^{2n+1} Y_l^m(\mathbf{q}) Y_l^m(\mathbf{p}), \tag{3.15}$$

which is essentially a low-pass filter applied to a truncated SH representation of a delta function.[2] This can be shown to lead to

$$\langle \Psi_{\mathbf{q}}^{\mathcal{D}} | \varphi \rangle = \int_{\Omega} \psi_{\mathcal{D}}(\mathbf{p} \cdot \mathbf{q}) \varphi(\mathbf{p}) \, \mathrm{d}\mathbf{p}, \tag{3.16}$$

where $\psi_{\mathbf{q}}^{\mathcal{D}}$ is the so-called zonal Green kernel,

$$\psi_{\mathcal{D}}(\mathbf{p} \cdot \mathbf{q}) = \sum_{l \in \mathbb{N}_0, \hat{\mathcal{D}}_l \neq 0} \frac{2l+1}{4\pi \hat{\mathcal{D}}_l} C_l^{(1/2)}(\mathbf{p} \cdot \mathbf{q}), \tag{3.17}$$

where $C_l^{(1/2)}$ are ultraspherical or Gegenbauer polynomials. Now, given a point grid $\{\mathbf{q}_i\}_{i=1}^N$, a spline s is defined by

$$\mathcal{D}s = \sum_{i=1}^{N} a_i \delta_{\mathbf{q}_i} \tag{3.18}$$

and can be evaluated using

$$s(\mathbf{p}) = \sum_{i=1}^{N} a_i \psi_{\mathcal{D}}(\mathbf{p} \cdot \mathbf{q}), \tag{3.19}$$

given that there are no $\hat{\mathcal{D}}_l = 0$. Otherwise, the missing harmonics will still be included as follows,

$$s(\mathbf{p}) = \sum_{i=1}^{N} a_i \psi_{\mathcal{D}}(\mathbf{p} \cdot \mathbf{q}) + \sum_{l \in \mathbb{N}, \hat{\mathcal{D}}_l = 0} \sum_{m=-l}^{l} \hat{s}_l^m Y_l^m(\mathbf{p}), \tag{3.20}$$

[2] It is unclear from [13] what definition of the Dirac measure is used but seemingly it is the complex conjugate of the conventional definition of the delta function used in other works, $\langle \delta_{\mathbf{p}} | f(\mathbf{q}) \rangle = \bar{f}(\mathbf{p})$. Given that $\bar{Y}_l^m = (-1)^m Y_l^{-m}$, the difference in the addition formula is only a difference in sign.

It is therefore important to choose the right pseudo-differential operator when defining the spherical splines.

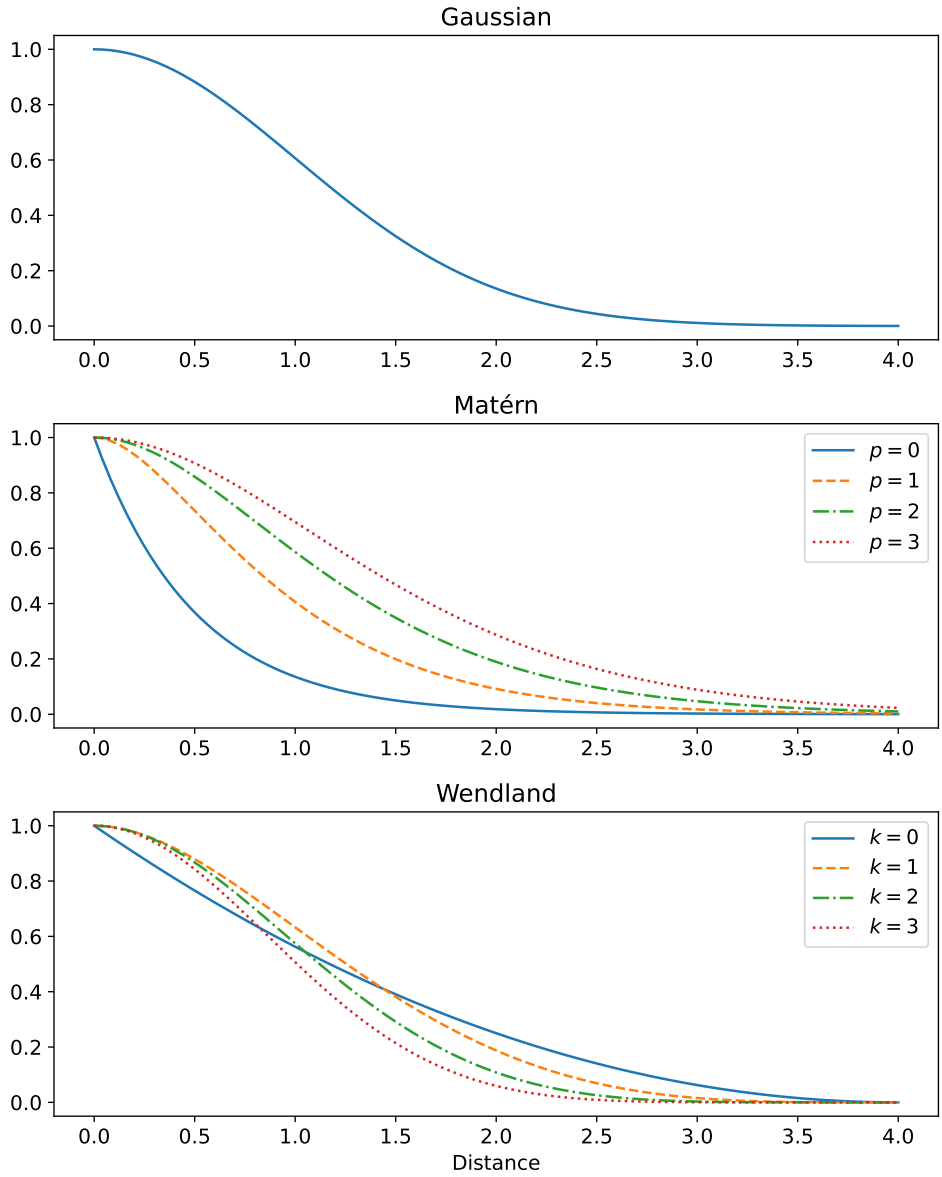


Figure 3.3: A visualization of the different kernel functions used, all normalized such that the maximum is one. The scale parameter ϵ was set to $\epsilon = 1$ for the Gaussian, $\epsilon = 0.5$ for the Matérn kernel functions and $\epsilon = 4$ for the Wendland kernel functions.

In this thesis, two classes of spline kernel functions are used, the Matérn and the Wendland functions. The use of these functions in the context of inverse problems is described by Simeoni 2020 [13]. These functions are replacements for the Sobolev zonal Green kernel associated with Sobolev operators of the form

$$\mathcal{D}_\beta = [Id - \Delta^*]^\beta \tag{3.21}$$

where Id is the identity operator and Δ^*, as previously, the Laplace-Beltrami operator.

Furthermore, a Gaussian kernel function will be used. This function is not accounted for in the theory of spherical splines. If this is a necessary condition, one may use, e.g., high-order Matérn functions as these are almost indistinguishable from the Gaussian function. In fact, as the order of Matérn functions goes to infinity they approach a Gaussian function [13]. The definition of the Gaussian kernel is as follows

$$G(r) = \exp\left(-\frac{1}{2}\left(\frac{r}{\epsilon}\right)^2\right), \tag{3.22}$$

where the scale parameter ϵ is discussed further in the next section.

3.2.2.1 Matérn

Matérn functions are defined as [13],

$$S_\nu^\epsilon(r) := \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{r}{\epsilon}\right)^\nu K_\nu\left(\frac{r}{\epsilon}\right), \qquad \forall r > 0, \tag{3.23}$$

where Γ is the Gamma function, K_ν is the modified Bessel function of the second kind and the parameters are selected such that $\nu \geq 0$ and $1 \geq \epsilon > 0$. The scale parameter ϵ is selected with a parameter sweep as will be discussed further in chapter 4.

Given $\nu = p + 1/2$ where $p \in \mathbb{N}$, the Matérn function can be written as a polynomial multiplied by an exponential function. Following Simeoni, explicit formulas for a few orders can be written as follows [13],

$$\text{for } p = 0, \quad S_{1/2}^\epsilon(r) = \exp\left(-\frac{r}{\epsilon}\right), \tag{3.24}$$

$$\text{for } p = 1, \quad S_{3/2}^\epsilon(r) = \left[1 + \frac{r}{\epsilon}\right]\exp\left(-\frac{r}{\epsilon}\right), \tag{3.25}$$

$$\text{for } p = 2, \quad S_{5/2}^\epsilon(r) = \left[3 + 3\frac{r}{\epsilon} + \frac{r^2}{\epsilon^2}\right]\exp\left(-\frac{r}{\epsilon}\right), \tag{3.26}$$

$$\text{for } p = 3, \quad S_{7/2}^\epsilon(r) = \left[15 + 15\frac{r}{\epsilon} + 6\frac{r^2}{\epsilon^2} + \frac{r^3}{\epsilon^3}\right]\exp\left(-\frac{r}{\epsilon}\right). \tag{3.27}$$

These kernel functions are not normalized. This should be done before performing a reconstruction, either analytically or numerically. This is discussed further in Appendix B.

3.2.2.2 Wendland

Wendland functions on the 2-sphere are defined as follows [13]

$$\phi_k(r) = (\mathcal{I}^k\phi_l)(r), \tag{3.28}$$

where $k \in \mathbb{N}$, $l = k + 2$, ϕ_l is Askey's truncated power function and $\mathcal{I}$ is an integral operator. However, they can also be represented as

$$\phi_k(r) = \max(1 - r/\epsilon, 0)^{l+k} p_{k,l}(r/\epsilon), \qquad \forall r \geq 0 \tag{3.29}$$

where $p_{k,l}$ is a polynomial of degree k.

Under the constraint $k \in \mathbb{N}$ with $l = d$, the Wendland kernel functions of orders $k = 0, 1, 2, 3$ can be expressed as follows [13]

$$\text{for } k = 0, \quad \phi_0(r) = \max(1 - r/\epsilon, 0)^2, \tag{3.30}$$

$$\text{for } k = 1, \quad \phi_1(r) = \max(1 - r/\epsilon, 0)^4 \left(1 + 4\frac{r}{\epsilon}\right), \tag{3.31}$$

$$\text{for } k = 2, \quad \phi_2(r) = \max(1 - r/\epsilon, 0)^6 \left(3 + 18\frac{r}{\epsilon} + 35\left(\frac{r}{\epsilon}\right)^2\right), \tag{3.32}$$

$$\text{for } k = 3, \quad \phi_3(r) = \max(1 - r/\epsilon, 0)^8 \left(15 + 120\frac{r}{\epsilon} + 375\left(\frac{r}{\epsilon}\right)^2 + 480\left(\frac{r}{\epsilon}\right)^3\right). \tag{3.33}$$

Just like the Matérn functions, these are not normalized.

3.3 Regularization

L1-regularization, also known as least absolute shrinkage and selection operator (LASSO), refers to the use of the L1-norm of the parameters, in this case the reconstruction coefficients, as a regularization term, leading to the following problem formulation,

$$a^*_{i,q} = \arg\min_{a_{i,q}} \left[\left(\sum_j \sum_s w_{j,s} \mathcal{R}_{j,s} \right) + \lambda_{\mathrm{L1}} ||a_{i,q}||_1 \right], \tag{3.34}$$

where

$$\Lambda_{\mathrm{L1}} = \lambda_{\mathrm{L1}} ||a_{i,q}||_1 = \lambda_{\mathrm{L1}} \sum_{i,q} |a_{i,q}|. \tag{3.35}$$

Similarly for L2-regularization, also known as ridge regression and closely related to the more general concept of Tikhonov regularization, the regularization term in the inverse problem of Eq. (2.11) becomes,

$$\Lambda_{\mathrm{L2}} = \lambda_{\mathrm{L2}} ||a_{i,q}||_2^2 = \lambda_{\mathrm{L2}} \sum_{i,q} |a_{i,q}|^2. \tag{3.36}$$

Total variation regularization may refer to a number of related methods that can in general be described as L1-norms of derivatives [23]. The motivation for using this regularization term comes partly from its relation to the actively researched field of compressive sampling [24] and partly from recent applications in the context of electron tomography [25] with similar motivation. Total variation regularization applied to a voxel grid can be expressed as

follows [26]

$$\Lambda_{\mathrm{TV}}(a) = \lambda_{\mathrm{TV}} \sum_{ijkl} \sqrt{(a_i(x_j, y_k, z_l) - a_i(x_{j-1}, y_k, z_l))^2 + } \tag{3.37}$$

$$\overline{(a_i(x_j, y_k, z_l) - a_i(x_j, y_{k-1}, z_l))^2 + } \tag{3.38}$$

$$\overline{(a_i(x_j, y_k, z_l) - a_i(x_j, y_k, z_{l-1}))^2}, \tag{3.39}$$

where x_j, y_k and z_l refer to the positions of the different voxels. Using this regularization term tends to produce artifacts, known as staircasing artifacts, since it tends to incentivize piecewise flat volumes in the reconstruction coefficients. If such artifacts are prominent, one possible solution is to use higher-order Total Variation (TV) based terms, called generalized TV, that incorporate higher-order derivatives [27].

A common method for selecting hyperparameters is to perform reconstructions using different values for the hyperparameter, a parameter sweep, and then plot the residual and regularization term in a log-log plot. This plot often has an L-shape and the idea is to choose a value for the parameter that simultaneously minimizes the residual and regularization term [28].

4
Method

Building on the theoretical overview of the previous chapters, we will now focus on the application of these concepts. The aim of this chapter is to describe in some detail the various implementation steps. A brief overview of the code structure is provided as well. Furthermore, the reconstruction and test procedure used to compare different methods is described.

4.1 Implementation Details

Firstly, we will look at the main steps taken in creating a functional implementation. The section entails a high-level view of the different representations and segment projections thereof as well as the SH transform.

4.1.1 Spherical Pixel Representation

Once an implementation of a spherical grid $G = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}$ is available, we can create a spherical Voronoi diagram to define a set of cells $\{C_i\}_{i=1}^n$ associated with each point in the grid, such that the cells include all points on the sphere that are nearest to a particular grid point, $C_i = \{\mathbf{p} \mid D(\mathbf{p}, \mathbf{p}_i) \leq D(\mathbf{p}, \mathbf{p}_j),\ \forall j \neq i,\ \mathbf{p}_k, \mathbf{p} \in \Omega\}$. Each cell C_i has a surface area $\mathcal{A}_i$. A spherical function can now be approximated using a set of spherical pixel basis functions $\{P_i\}_{i=1}^n$ that are constant over a cell and zero outside of the cell. This is a box function with a given height, $P_i(\mathbf{p}) = h_i$ for $\mathbf{p} \in C_i$. The height associated with each cell can be selected freely, but most appropriate is the use of a normalization scheme.

The scalar product of two of the basis functions is

$$\frac{1}{4\pi}\int_\Omega P_i(\mathbf{p})P_j(\mathbf{p})\,\mathrm{d}\omega(\mathbf{p}) = \begin{cases} h_i^2\mathcal{A}_i, & \text{for } i = j \\ 0, & \text{for } i \neq j \end{cases}. \tag{4.1}$$

Setting $h_i^2\mathcal{A}_i = 1$ makes the set of pixels an orthonormal basis for the sphere. Another option would be to set the spherical mean of each basis function to one, making the computation of the spherical mean of a function equal to the sum of all basis coefficients.

The value of a function represented using these spherical pixels, for a particular point on the surface of the sphere, is determined by binning that point into one of the pixels. Figure 5.10 in chapter 5 illustrates an RSM represented using pixels.

4.1.1.1 Segment Projection

As explained in the background for SAXSTT, in order to reconstruct a sample from a data set, the reconstruction coefficients must be projected in a two-step process. The first step is the John transform which integrates voxels along a line of projection, resulting in the projection of the voxel grid onto the detector plane. The second step involves finding the contribution of each basis function to the circle segments in the plane which is orthogonal to the line of projection. For each type of representation, this segment projection needs to be determined.

In the case of the pixel representation, finding the projection is a question of finding the intersection between a segment and the cross section of a pixel. A given function on the sphere can be approximated as a sum of basis functions weighted by coefficients a_i,

$$f(\mathbf{p}) = \sum_{i=1}^{n} a_i P_i(\mathbf{p}) + e(\mathbf{p}) \tag{4.2}$$

where $e(\mathbf{p})$ is an error term dependent on the properties of the basis and the function. If the basis is orthogonal, the coefficients can be determined by

$$a_i = \int_\Omega f(\mathbf{p}) P_i(\mathbf{p}) \, \mathrm{d}\omega(\mathbf{p}), \tag{4.3}$$

and the average of the error over a cell will be zero which is easily seen by plugging Eq. (4.2) into Eq. (4.3). So, if f is a function within the space of functions spanned by the pixel basis, it can be expressed as $f = P_i a_i$ using Einstein notation. We compute the projection of f onto the detector plane as an integral along each segment,

$$\mathbb{D}[f(\mathbf{p})](S_j) = \int_{S_j} f(\mathbf{p}) \, \mathrm{d}l(\mathbf{p}) = \int_{S_j} P_i(\mathbf{p}) a_i \, \mathrm{d}l(\mathbf{p}) = \tag{4.4}$$

$$= h_i a_i \int_{S_j \cap C_i} \mathrm{d}l(\mathbf{p}) = \mathbb{D}_{ij} a_i, \tag{4.5}$$

where $\mathbb{D}_{ij}$ is the overlap of each pixel with each segment, the segment projection matrix, S_j is the segment whose index is j and, as explained above, h_i is the height of the pixel function P_i. Numerically, this is computed by sampling a number of points along each segment and approximating the integral above. The sample points are supplied by MUMOTT and can be tweaked by the user. In Equation D, some details about the normalization of the segment projection can be found.

4.1.2 Radial Basis Function Representation

RBFs are implemented similarly to the spherical pixels. For each point in the spherical grid G as defined above, a corresponding basis function is defined $K(D(\mathbf{p}_i, \mathbf{p}))$ where K is a kernel function and D is a distance function. The

kernel function could be a Gaussian, Matérn or Wendland function. The distance function is the Euclidean distance. A spherical function $f(\mathbf{p})$ can thus be represented by,

$$f(\mathbf{p}) = \sum_{i=1}^{n} a_i K(D(\mathbf{p}_i, \mathbf{p})) + e(\mathbf{p}). \tag{4.6}$$

The normalization used in the case of RBFs is slightly different to the one used for pixels since RBFs cannot be orthogonal to begin with. The used normalization is

$$\frac{1}{4\pi} \int_{\Omega} K(D(\mathbf{p}_i, \mathbf{p})) \, \mathrm{d}\omega(\mathbf{p}) = 1, \tag{4.7}$$

where the kernel function is scaled such that the mean of the RBF over the sphere is one. In the same way as for the pixel basis, we compute a segment projection matrix $\mathbb{D}_{ij}$,

$$\mathbb{D}[f(\mathbf{p})](S_j) = \int_{S_j} K(D(\mathbf{p}_i, \mathbf{p}))c_i \, \mathrm{d}l(\mathbf{p}) = \mathbb{D}_{ij}c_i, \tag{4.8}$$

4.1.3 Spherical Harmonic Transform

In order to analyze reconstructions they need to be converted into the same representation as the true coefficients. Since the simulations were done in RSH, to test methods based on SH representations, a general RSH transform is needed. The method of Driscoll and Healy is exact up to a degree b given a number of sample points $(2b)^2$, as detailed in section 2.2. The basic method is outlined in chapter 2. We perform the complex SH transform as defined in Eq. (2.22), using SH functions from the SCIPY python package [22] and apply the relationship between complex and real harmonics in Eq. (2.19).

In order to validate the transform, a reconstruction using simulated data is made and the mean and standard deviation computed for each voxel, before and after applying the transform. More details can be found in Appendix C.

4.1.4 Framework Structure

A framework for working with local representations was implemented in Python. The framework partly modifies existing code in the MUMOTT package and adds classes for working with spherical grids and representations as well as a SH transform. The structure of the framework is illustrated in a diagram in Figure 4.1. The regularization terms were included in MUMOTT by modifying the code in the initialization of the local representations framework, using the definitions provided in chapter 3.

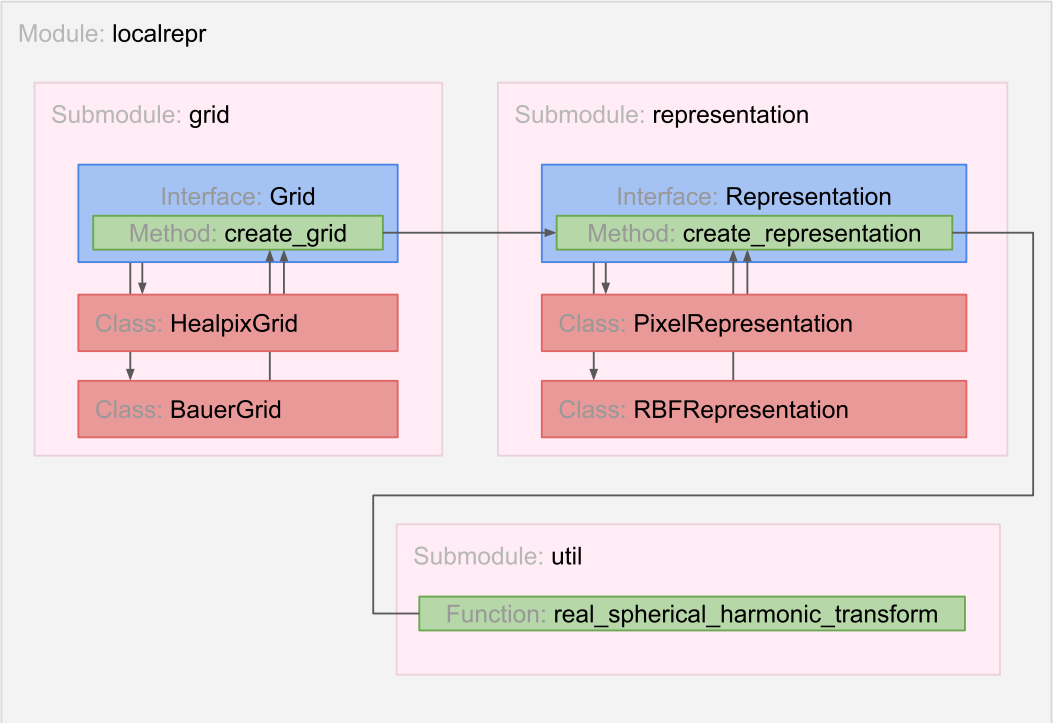


Figure 4.1: A diagram illustrating the structure of the local representations framework.

4.2 Tests and Comparisons

In this section, the method used to compare different reconstruction methods is explicated. Most of the comparisons are made using simulated data.

4.2.1 Simulated Data

The data used to evaluate and compare different reconstruction methods is comprised of simulated samples with various levels of Poisson noise. These simulations have been carried out previously by Nielsen and used in [8] to compare SAXSTT methods. There are three sets of data, each of which with five noise levels characterized by the Poisson rate parameter $k = 10^t$ where $t = \{2, 1.5, 1, 0.5, 0\}$. The data sets have the shape of the letters M and T as well as a mammoth and will hence be referred to as "M", "T" and "mammoth". Since both the signal and noise is known exactly for this data, the signal-to-noise ratio can be estimated and will be used to differentiate the different noise levels.

The true coefficients of the "M" data consist of RSHs up to $l_{\max} = 12$. The RSMs of "M" have been made to be approximately zonal, i.e., symmetric about an axis, mimicking a fiber-like material. The "T" data only contains

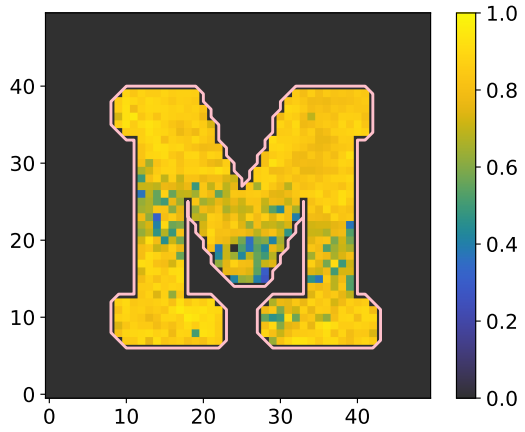


Figure 4.2: A slice of the correlation between a reconstruction and the true RSM for the simulated "M". The contour illustrates the main volume, selected by the mask.

RSHs with $l = 2$, a rank-2 tensor representation which was made to fit another SAXSTT method known as iterative reconstruction tensor tomography [29]. The "T" data is not used much in the results presented below. Lastly, "mammoth" has $l_{\text{max}} = 8$ with weak components at $l = 2$. It can be used to show the effectiveness of a method for the features of the RSMs other than the orientation features of rank-2 tensors. [8]

4.2.1.1 Comparison

When comparing the reconstruction coefficients of simulated data to its true coefficients, the reconstruction is transformed to RSHs after which the Pearson correlation between the reconstruction and the true data is determined. After calculating the correlation for each voxel, a mask that selects the main volume of the sample is applied. For each voxel in the true RSM, the mask is zero if the RSH coefficients of that voxel are zero and one otherwise. An illustration of the correlation and the mask is seen in Figure 4.2. After applying the mask, the mean and standard deviation of the correlation over the voxels is determined.

4.2.2 Parameter Selection

Before performing a reconstruction there are certain parameters that need to be specified. The underlying grid used for a representation will have slightly different properties, especially when the number of points per voxel is low. Fibonacci spiral methods are not inherently point symmetric, which will result in some error when point symmetry is enforced in the representation. For this reason, a HEALPix-based representation was used. A higher number of points per voxel will result in a higher resolution representation as well as decreasing the symmetry error in the case of Fibonacci spiral methods. However, since the main point of comparison of the various representations investigated in this project is the RSH representation for a certain l_{max}, the number of points

should be selected so that the number of coefficients is comparable to the RSH representation. For this reason we set the parameter $N_{\text{side}} = 2$, which gives us a HEALPix grid with 48 points and a representation with 24 coefficients, considering point symmetry. This is comparable to the 28 coefficients we get if $l_{\max} = 6$ in the RSH representation.

With a grid chosen, the pixel representation is determined except for the normalization scheme. The normalization scheme affects the magnitude of the reconstruction coefficients and the choice of regularization parameters but should not affect the quality of the reconstruction because of the linearity of the problem. However, the RBF representations involve one more parameter which is the scale parameter ϵ. The exact meaning of this parameter depends on the choice of kernel function and thus a selection procedure is needed. By varying the scale parameter and looking at the residual, the value that minimizes the residual can be selected as a good estimate. The best pick may be above this value if the data contains a significant amount of noise since the minimum of the residual will be for the parameter value that provides the best fit for both the signal and the noise. Picking a slightly higher value should have the effect of a low-pass filter and thus reduce the impact of the noise. However for the results below, the scale parameter was always chosen by minimizing the residual.

The scale parameter is of course dependent on the number of points in the underlying grid of a representation. By giving the scale parameter relative to the overall size of the cells, we can compare it between different grids. A useful metric for this purpose is the nodal width [13],

$$\Theta(\Xi) = \max_{\mathbf{p}\in\Omega} \min_{\mathbf{q}\in\Xi} D(\mathbf{p}, \mathbf{q}) \tag{4.9}$$

where $\Xi = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_N\}$ is a set of N points belonging to some grid. This is the maximum over all cells of the minimum distance between a point belonging to a cell and the center point of that cell. So essentially, if a spherical disk fills each cell then the nodal width is the maximum radius of all disks. Henceforth, the relative scale parameter is

$$\epsilon_{\text{rel}} = \epsilon / \Theta[\Xi]. \tag{4.10}$$

The scale parameter should be in the interval $0 < \epsilon \leq 1$ meaning that the relative scale parameter should fulfill $0 < \epsilon \leq 1/\Theta(\Xi)$.

Lastly, when using regularization terms, the relevant regularization hyper-parameters have to be determined. This can be done using the L-curve method where a parameter value that minimizes the residual and the regularization term simultaneously is picked [28].

5
Results and Discussion

Using the implementation described above, an investigation of the usage and effectiveness of the new representations and regularization terms was carried out for simulated datasets and for a data set of measurements done on a trabecular bone sample performed by Liebi *et al.* at the Paul Scherrer Institute [7]. The local representations were compared to the already implemented SIGTT method which uses RSH.

Firstly, appropriate parameters were selected for each representation. Once the appropriate usage of the representations was understood, aspects such as speed of convergence, robustness to initial conditions and the quality of reconstruction were investigated. Lastly, reconstructions of the trabecular bone were done, followed by a comparison of the effect of different methods.

5.1 Scale Parameter Selection

Kernel function	Relative relative scale parameter
Gaussian	0.7
Wendland, $k = 0$	2.3
Wendland, $k = 1$	2.65
Wendland, $k = 2$	3.1
Wendland, $k = 3$	3.5
Matérn, $p = 0$	0.7
Matérn, $p = 1$	0.45
Matérn, $p = 2$	0.35
Matérn, $p = 3$	0.3

Table 5.1: The relative scale parameter ϵ_{rel} given a HEALPix grid with $N_{\text{side}} = 2$.

For the Gaussian, Wendland and Matérn kernel functions different scale parameters are ideal. With four orders of each of the spline functions, there are in total nine options when selecting kernel function. In Table 5.1, the best relative scale parameters, as defined in Eq. (4.10), for each kernel function, selected by minimizing the residual at convergence are found. However, the nodal width for HEALPix with $N_{\text{size}} = 2$ is about 0.4, which means that the maximum allowed value of the scale parameter is about $1/0.4 = 2.5$, such that

some of the scale parameters are theoretically invalid. However, the corresponding kernel functions were not used for any of other results. The data set used in this comparison is the "M"-data set with the minimal noise setting.

If we refer back to Figure 3.3 in chapter 3, it is quite clear that the higher-order spline kernel functions are more similar to the Gaussian kernel. For that reason further tests will focus on the Gaussian and the lower-order spline kernel functions.

Another aspect of interest when it comes to the determination of the scale parameter is the variation with respect to the data set. As illustrated in Figure 5.1, the scale parameter does depend on the data set.

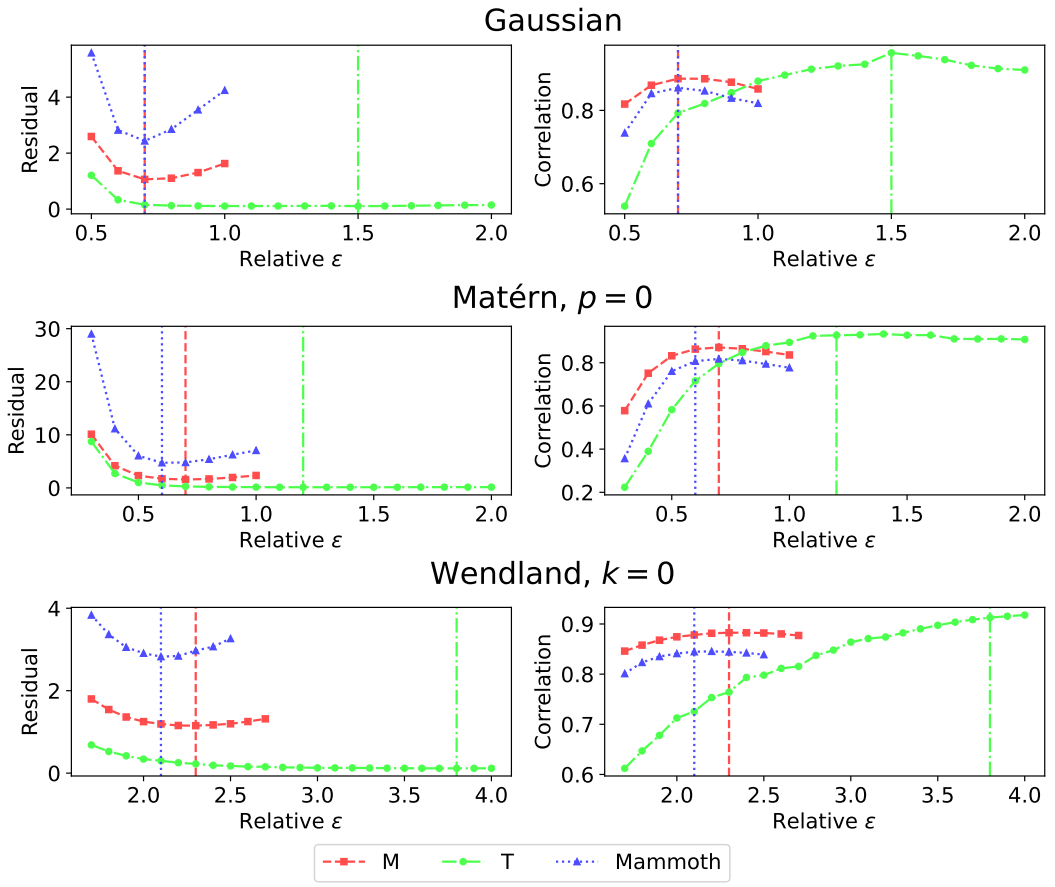


Figure 5.1: On the left, the residual term after convergence has been recorded for a number of different relative scale parameters. The vertical lines indicate the minimum residual value and the corresponding scale parameter. On the right, the mean correlation over the main volume is plotted. The point where the residual is minimal is once again indicated by the vertical lines and mostly corresponds to the maximum correlation. The data used had the lowest noise levels.

5.2 Regularization Parameter Selection

Once the scale parameters has been selected, the regularization hyperparameters are selected using the L-curve approach as illustrated in Figure 5.2. The same procedure is repeated for the "M", "T" and "Mammoth" data, always using the maximum amount of noise.

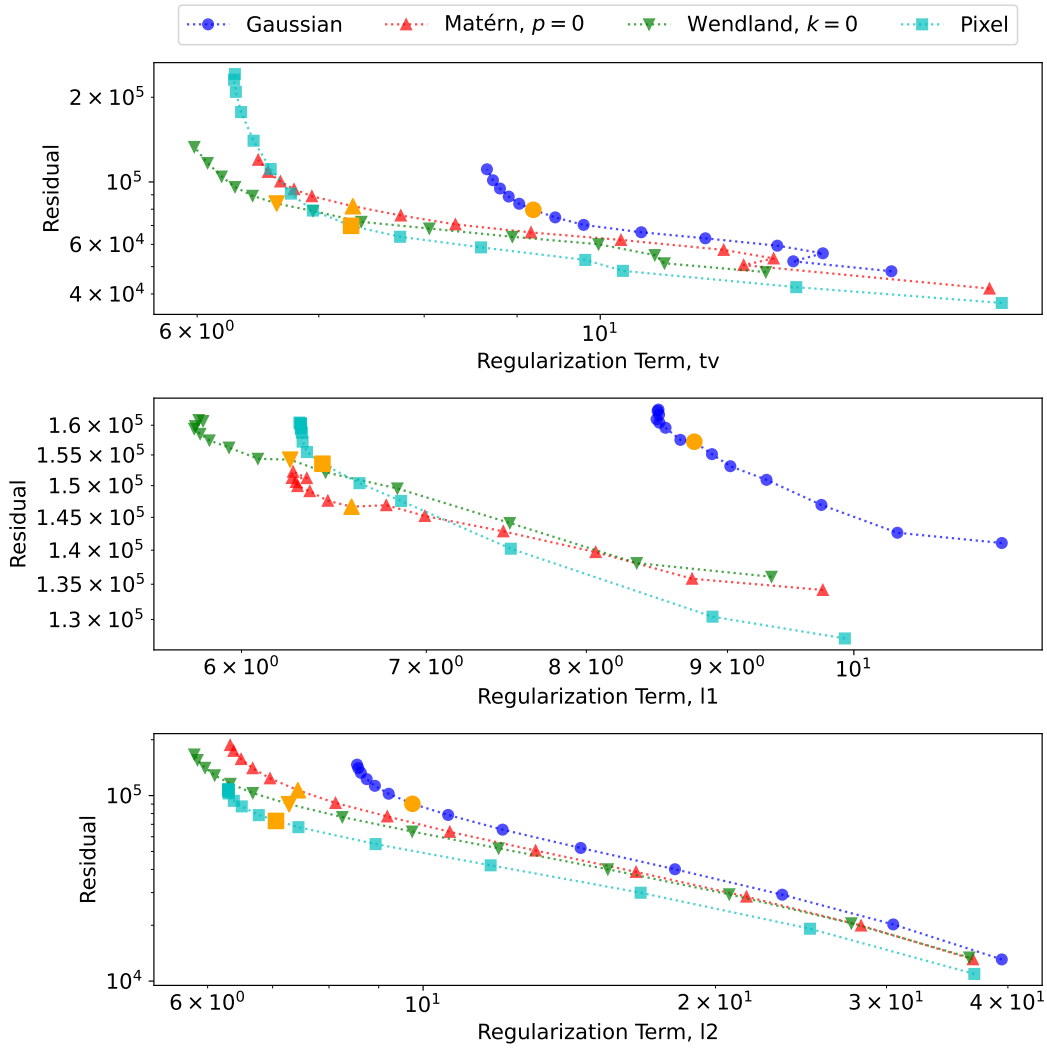


Figure 5.2: L-curves for RBFs with the different kernel functions and regularization terms for the "M" data with the maximum amount of noise. For each curve the point of the selected hyperparameter is indicated by an enlarged marker.

5.3 Initial Conditions

For most reconstructions, the initial conditions were set to 10^{-5}. As a simple test of the robustness of the reconstruction problem with regards to its initial conditions, the initial conditions were altered slightly by adding a random

coefficient vector in which every coefficient was picked from a uniform distribution in the interval $[0, 10^{-5})$. The correlations between the reconstruction with unaltered and altered initial conditions, respectively, was determined with a number of different settings. Since the correlations were high, the methods should be robust to small changes in initial conditions.

5.4 Comparison of Reconstructions on Simulated Data

In this section, comparisons are made of the different local representations and the RSH representation. For all of the local representations a HEALPix grid with $N_{\text{side}} = 2$ has been used. The three regularization terms described in chapter 3 are applied one-by-one.

For the RSH representation the maximum degree $l_{\max} = 6$. Furthermore, the regularization applied is an imitation of the regularization used by Nielsen in [10]. Three separate terms are applied after selecting their hyperparameters sequentially. The regularization terms used are the hybrid regularization, a nearest-neighbor L2-regularization and an L2-regularization. These terms have a likeness to the L1, TV and L2 terms applied for local representations. The nearest-neighbor L2-regularization in particular is defined as follows [10]

$$\Lambda_\rho = \sum_i \sum_j \sum_l \rho_{i,j}(l)(S_l(M_i, M_i) + S_l(M_j, M_j) - 2S_l(M_i, M_j)), \tag{5.1}$$

where $\rho_{i,j}(l)$ encodes the correlation length, M_i refers to the RSM of a voxel indexed i and the index j runs over the nearest neighbors of that voxel.

5.4.1 Convergence

We see in Figure 5.3a compared to Figure 5.3b that the RSH based method has comparable convergence to the other methods when regularization is applied. With no regularization it has a slower convergence, reaching lower residual values. This could be due to overfitting.

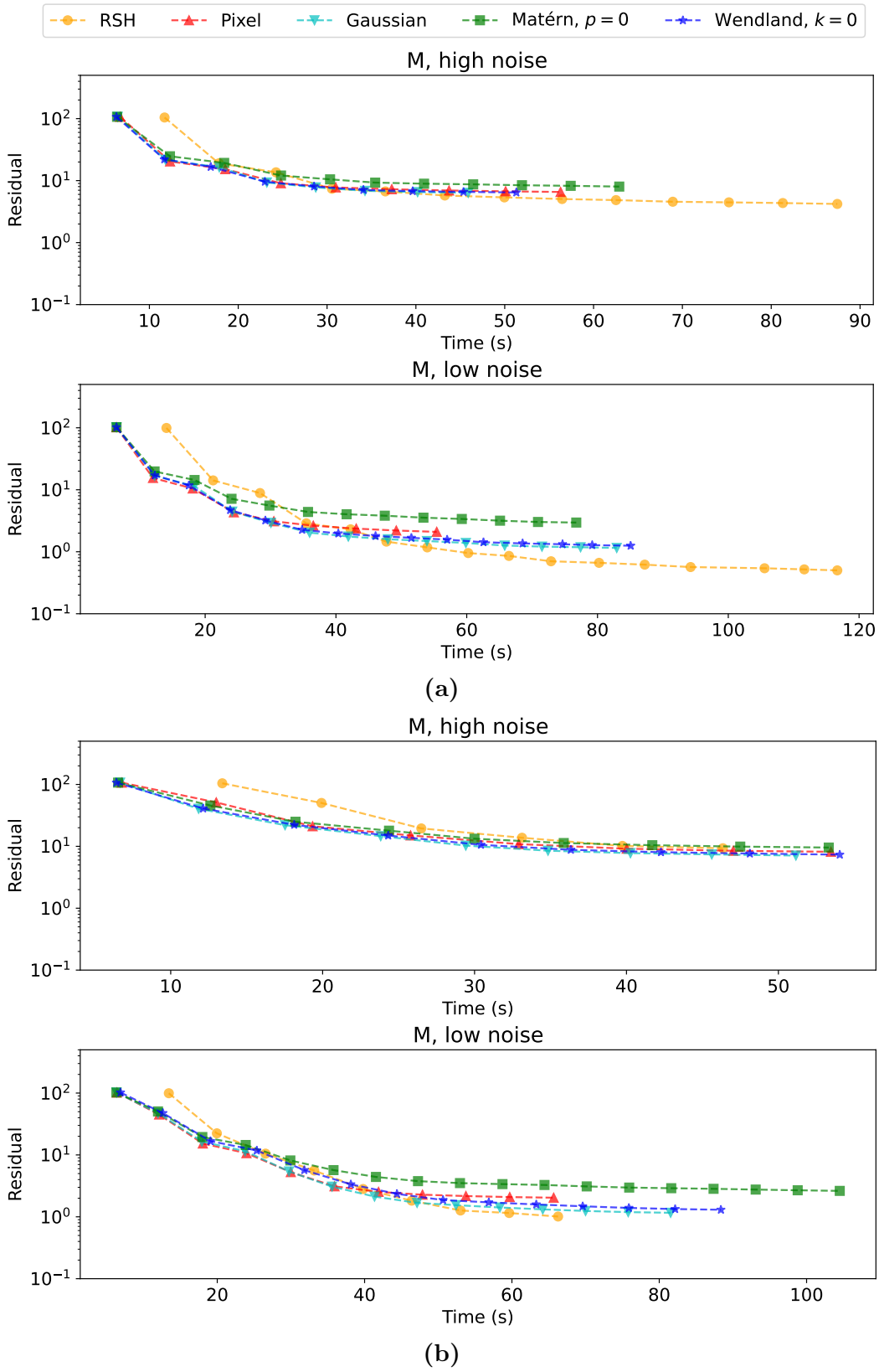


Figure 5.3: Convergence of reconstructions for the "M" data with high and low noise. **(a)** No regularization. RSH ends up with a lower residual than other methods. **(b)** TV-regularization has been applied for the local representations and the full regularization as described above for RSH.

5.4.2 Correlation

Performing many reconstructions and studying the correlation with the true coefficients, we notice a few things. In Figure 5.4, the reconstructions are made with no regularization. The cases of RSH and pixels underperform in all cases. In Figure 5.5, TV has been used for the local representations and the more complicated regularization scheme for the RSH basis. In this case, RSHs perform well, usually outperforming the other methods.

These results are not exactly a one-to-one comparison since for RSHs, slightly more coefficients per voxel have been used (28 rather than 24) and the regularization is more extensive, involving multiple terms. Still, the RBF methods are comparable.

In Appendix A, the results for L1- and L2-regularization can be found. These results are similar to the TV results above.

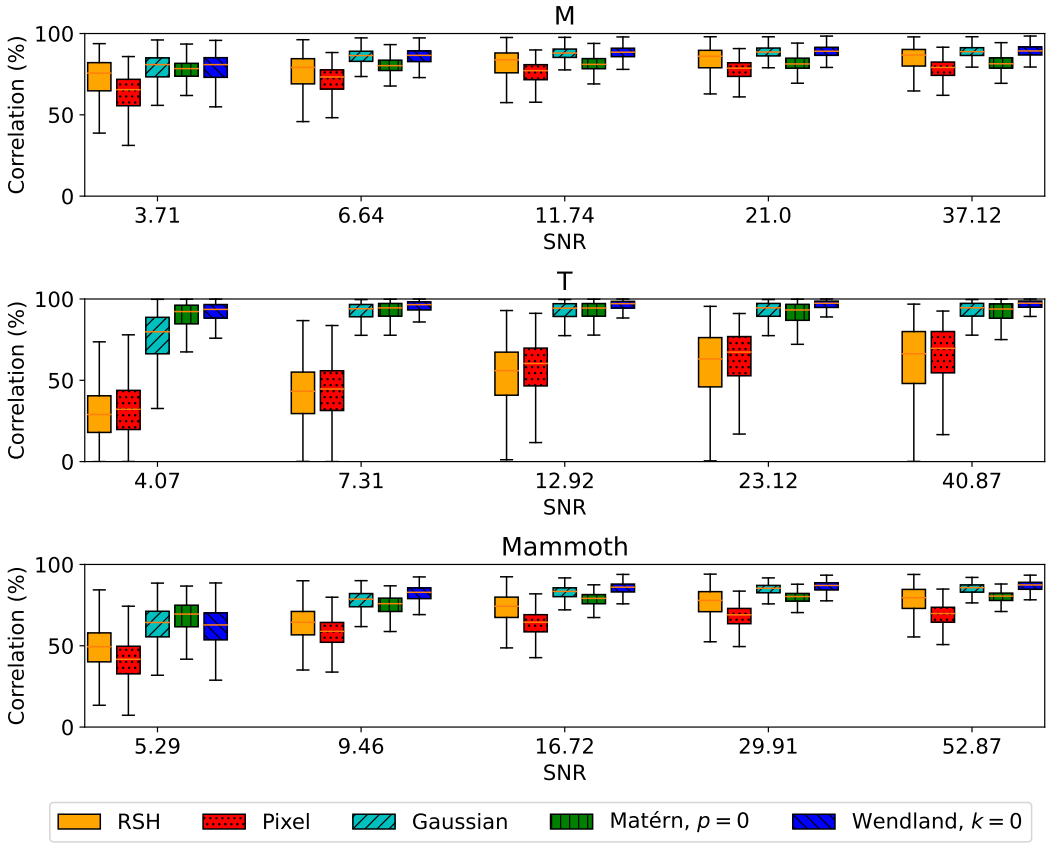


Figure 5.4: A Boxplot of correlations over the main volume of a series of reconstructions. In this case, no regularization has been applied.

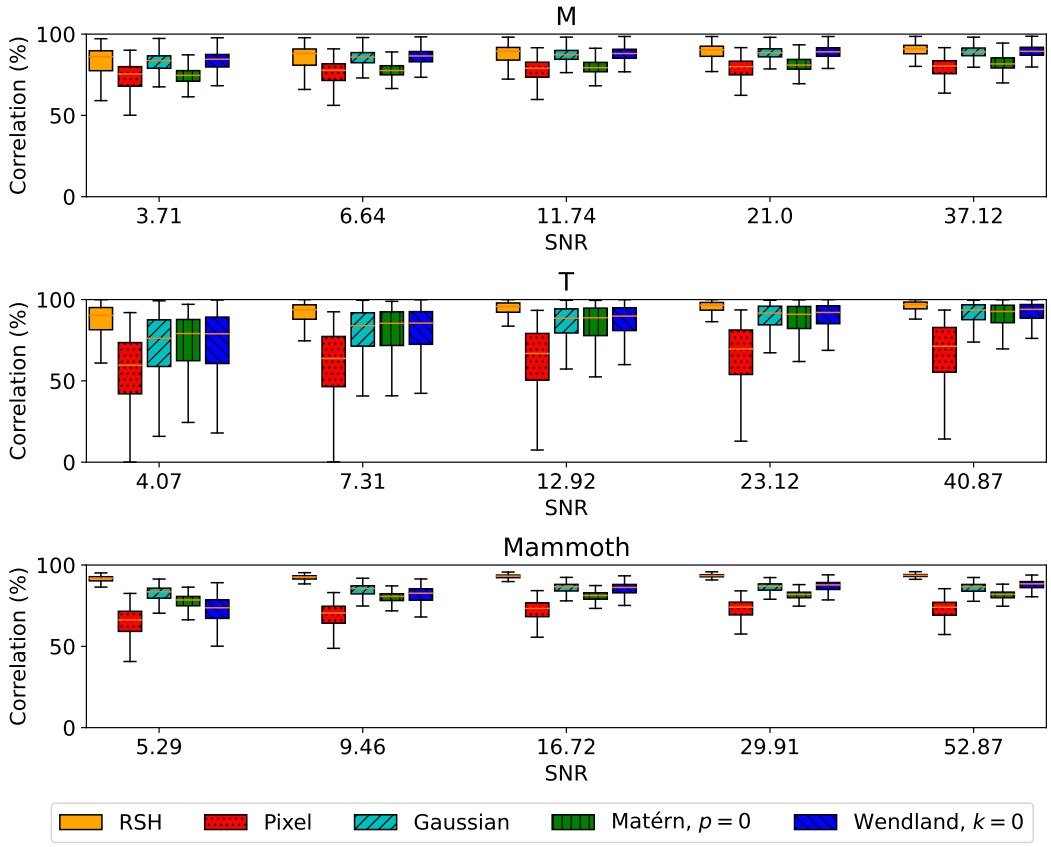


Figure 5.5: A box plot of correlations over the main volume of a series of reconstructions. In this case, TV regularization has been applied.

5.4.3 Inspection of Reciprocal Space Maps

We now pick a voxel per data set and plot the RSM of that voxel over different reconstructions. All RSMs are normalized such that their maximum is unity. In Figure 5.6, we see that the effect of the L1-regularization term is to make the reconstructed RSM more similar to the low-noise case, which is expected since the L1-regularization is a total generalized variation regularization on the RSM [13]. In fact, this is true for the other regularization terms as well, suggesting that they all have a similar effect on the RSM.

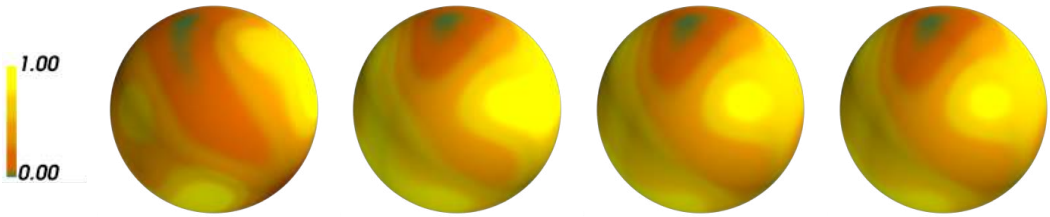
In Figure 5.7, we compare different regularization approaches. The TV-regularization is similar to the other two but stands out by including features that the other two do not. This is expected since it explicitly acts on several voxels at once and thus correlates adjacent voxels.

Comparing Figure 5.8 to Figure 5.6, we see that all of the RBF based methods produce similar reconstructions. Generally, Matérn functions are smeared over a larger surface area and Wendland functions have a sharper peak and are less smeared and more confined to one particular area on the sphere. These peaks are a clear form of artifact related to this particular representation, although the other RBF representations will have similar but less visible artifacts due to being constrained by the underlying point grid.

The set of functions that can be represented only includes functions that have their peaks at the grid points.

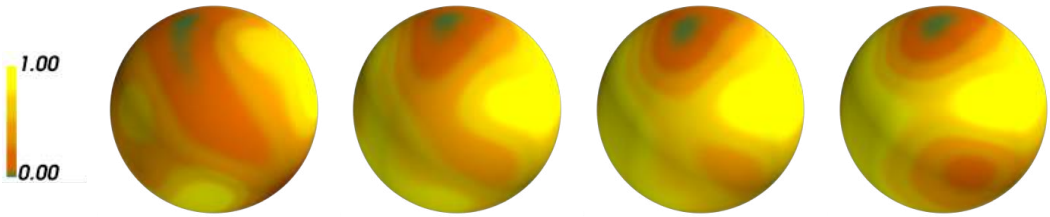
In Figure 5.9, the corresponding series of images as shown in Figure 5.6 is shown for RSH. We can see that the ones that have regularization applied are similar to the ones above.

We see in Figure 5.10 that the pixel based representation do maintain similar features. The spot of high intensity is visible for the minimum noise reconstruction for example. However, it is clear why this representation displays much lower correlation consistently, especially at this low number of pixels. The representation is not smooth and not very accurate with detailed features.



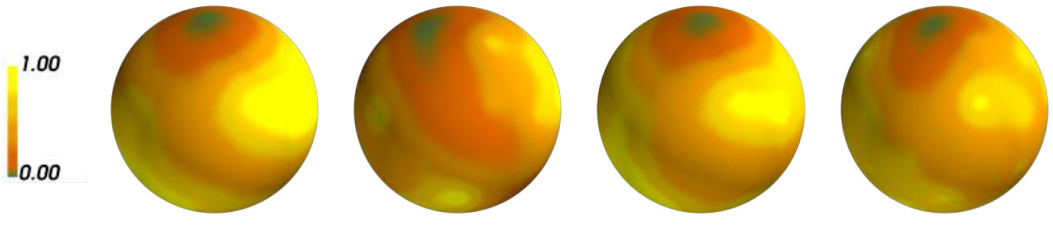
(a) **(b)** **(c)** **(d)**

Figure 5.6: The images show the RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. The RSM has been reconstructed using different settings and noise levels. **(a)** Gaussian RBF without regularization, for maximum noise level. **(b)** Gaussian RBF with TV-regularization, for maximum noise level. **(c)** Gaussian RBF without regularization, for minimum noise level. **(d)** Gaussian RBF with TV-regularization, for minimum noise level.



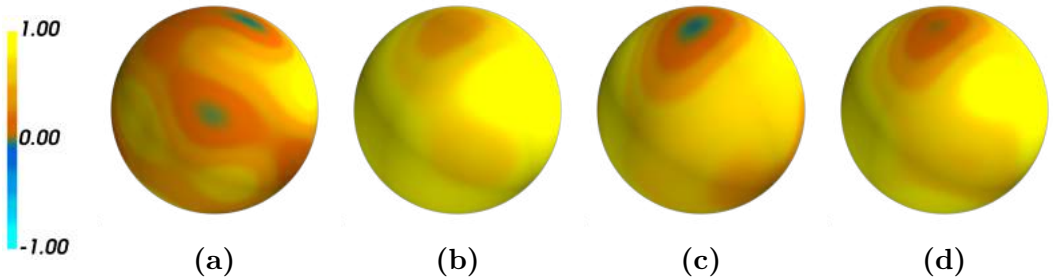
(a) **(b)** **(c)** **(d)**

Figure 5.7: The images show the RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. The RSM has been reconstructed using different settings at the maximum noise level. **(a)** Gaussian RBF without regularization. **(b)** Gaussian RBF with L1-regularization. **(c)** Gaussian RBF with L2-regularization. **(d)** Gaussian RBF with TV-regularization.



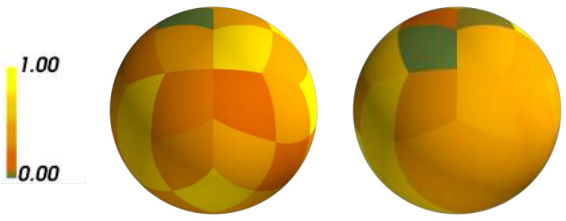
(a) **(b)** **(c)** **(d)**

Figure 5.8: The images show the RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. The RSM has been reconstructed using different settings and noise levels. No regularization has been applied. **(a)** Matérn RBF, for maximum noise level. **(b)** Wendland RBF, for maximum noise level. **(c)** Matérn RBF, for minimum noise level. **(d)** Wendland RBF, for minimum noise level.



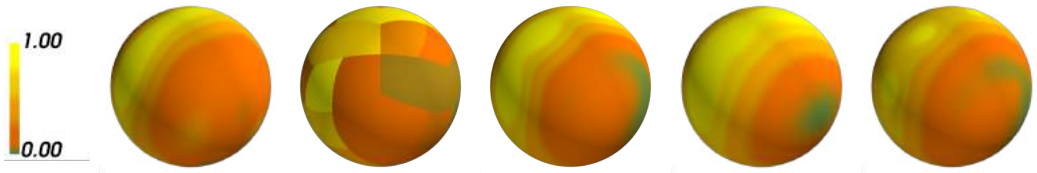
(a) **(b)** **(c)** **(d)**

Figure 5.9: The images show the RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. Negative values are indicated both by color and a dotted texture and can be seen in (a) and (c). The RSM has been reconstructed using different settings and noise levels. **(a)** RSH without regularization, for maximum noise level. **(b)** RSH with regularization, for maximum noise level. **(c)** RSH without regularization, for minimum noise level. **(d)** RSH with regularization, for minimum noise level.



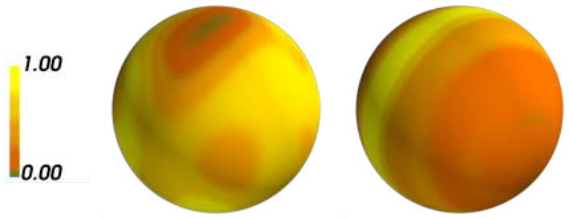
(a) **(b)**

Figure 5.10: The images show the RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. The RSM has been reconstructed for different noise levels. No regularization has been applied in any of these images. **(a)** Pixels for maximum noise level. **(b)** Pixels for minimum noise level.



(a) (b) (c) (d) (e)

Figure 5.11: The images show the RSM of the voxel at $(15, 20, 10)$ in the "M" data. The RSM has been reconstructed for the minimum noise level and no regularization has been applied in any of these images. **(a)** RSH. **(b)** Pixels. **(c)** Gaussian RBFs. **(d)** Matérn RBFs. **(e)** Wendland RBFs.



(a) **(b)**

Figure 5.12: Both images are for the minimum noise level. **(a)** The true RSM of the voxel at $(38, 30, 44)$ in the "mammoth" data. **(b)** The true RSM of the voxel at $(15, 20, 10)$ in the "M" data.

In Figure 5.11 we see for the "M" data, all representations with no regularization and for the minimum noise level. This data is made to have fiber-like features that should be seen as a ring on the RSM. It seems like all representations can pick up the fiber direction in this case.

Lastly, in Figure 5.12, we see the true RSMs for the "mammoth" and "M" data. It is clear that all representations manage to approximate these once regularization has been applied and in the low noise case. Although, clearly there are different artifacts associated with the different representations.

5.5 Comparisons of Reconstructions on Experimental Data

In all figures of this section, a constant threshold has been applied to the mean of the RSMs. Inspecting different reconstruction methods applied to experimentally measured data of a trabecular bone sample, we see in Figure 5.13 that the regularization has an effect on the smoothness of the reconstruction and the amount of correlation with neighboring voxels, especially in regards to orientation. Both the SH regularization, which includes a correlation regularization term for this specific purpose, and the TV-regularization exhibit this feature to varying degrees.

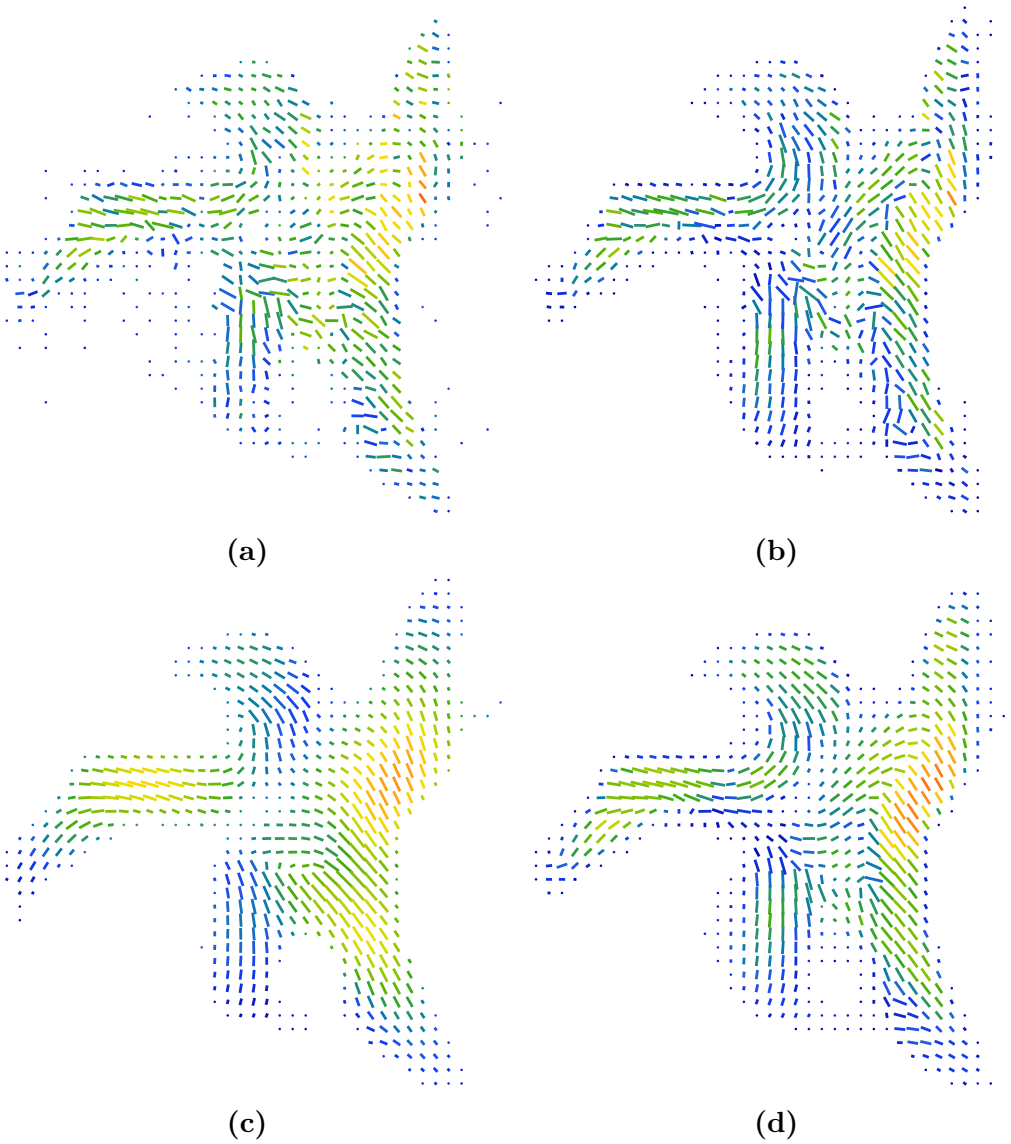


Figure 5.13: The images show the main orientation of fiber-like structures as the orientation of the cylinders. The mean of the RSM is shown as the scale of each cylinder. The standard deviation is shown as color with red implying high and blue low. The images come from slices of reconstructions of a trabecular bone sample made using different settings. The settings used were as follows. **(a)** RSHs without regularization. **(b)** Gaussian RBFs without regularization. **(c)** RSHs with nearest-neighbor L2-regularization. **(d)** Gaussian RBFs with TV-regularization. The regularized reconstructions are more coherent, especially in regards to orientation.

Surprisingly, the L1-regularization seems to perform as well as the TV-regularization as seen in Figure 5.14. This is unexpected since the L1-regularization targets each voxel independently and should not correlate neighboring voxels. It could be that it is more important to regularize the RSM to be able to accurately reconstruct the underlying structure. Another explanation is that

since the residual is computed for the measured intensities and the projected, using the John transform, model, the residual does not target voxels independently and could implicitly correlate them along projection lines. One difference between L1 and TV is that L1 penalizes the magnitude of coefficients and therefore shrinks the reconstruction if too much of it is applied.

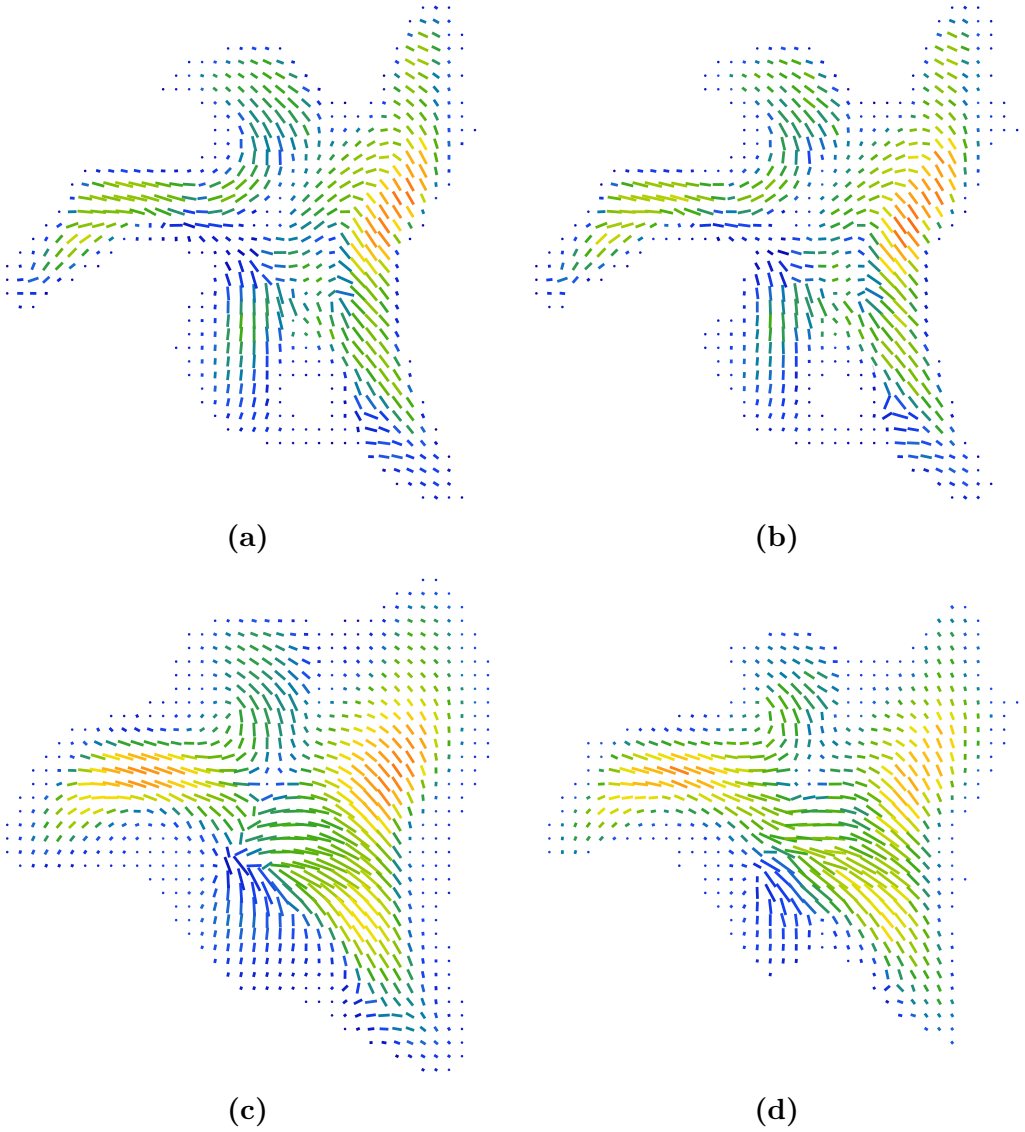


Figure 5.14: The images show the main orientation of fiber-like structures as the orientation of the cylinders. The mean of the RSM is shown as the scale of each cylinder. The standard deviation is shown as color with red implying high and blue low. The images come from slices of reconstructions of a trabecular bone sample made using different settings. The settings used were as follows. **(a)** Gaussian RBFs with L1-regularization. **(b)** Gaussian RBFs with too much L1-regularization. **(c)** Gaussian RBFs with TV-regularization. **(d)** Gaussian RBFs with too much TV-regularization. The overregularized reconstructions are smeared. Too much L1 has a shrinking effect.

6
Concluding Remarks

Of the different local representations that have been studied, those based on RBFs have been the most effective. They are comparable in performance to the RSH-based method. Based on the reconstructions of simulated data, the RBF based methods seem to perform better than RSH in the case of no regularization. This is unexpected since the simulations were done in RSH and could be a case of overfitting as indicated by the convergence plots.

The TV-regularization displays a similar effect of enforcing correlation between neighboring voxels as the nearest-neighbor L2-regularization, which is expected since they both work on differences between neighboring voxels, although in different ways. However, the L1-regularization is almost identical.

6.1 Outlook

Fiber-like features are seen as great-circles in reciprocal space. If a sample is known to contain such structures and the goal of a reconstruction is to study such features, one alternative, that has been brought up by others, would be to perform the reconstruction in a basis of functions symmetric about the fiber axis and the plane orthogonal to that axis, in other words a basis of ring-like functions. One simple way of implementing such a representation is to define an equatorial distance functions,

$$D(\mathbf{p}, \mathbf{q}) = R\left|\sin^{-1}\left(\frac{\mathbf{p}\cdot\mathbf{q}}{R^2}\right)\right| = R\theta'. \tag{6.1}$$

This distance function is similar to the great-circle distance but with the difference that the angle θ' is measured from the xy-plane instead of the z-axis, meaning that the distance is measured from the equator. Using this distance function in conjunction with the kernel functions discussed previously, such as Gaussians or Matérn functions, a ring-like function is defined. These functions are normalized numerically. Having performed some test runs using this setup, reconstructions were made successfully.

When it comes to point grids, it may be worthwhile to investigate different options and what effect they have on reconstructions under different circumstances and with different samples. It should be possible to construct grids that fulfill various symmetries accurately, which may be useful should these symmetries constitute prior knowledge about some sample.

Another topic that could be relevant to future research is compressive sampling. Compressive sampling, or compressed sensing, refers to a mathematical

theory of sampling. The signal is reconstructed in a basis where it is sparse or compressed and sampled in a basis incoherent with the reconstruction basis. This approach leads to fewer sample points being needed to perform the reconstruction [24]. L1 and TV-regularization are often used in the context of compressive sampling, however a full exploration of the approach would involve both the representation and the sampling methodology.

Bibliography

[1] Samuel S. Welborn and Eric Detsi. "Small-angle X-ray scattering of nanoporous materials". In: *Nanoscale Horiz.* 5 (1 2020), pp. 12–24. DOI: 10.1039/C9NH00347A. URL: http://dx.doi.org/10.1039/C9NH00347A.

[2] "Introduction". In: *Introduction to Soft Matter*. John Wiley & Sons, Ltd, 2007. Chap. 1, pp. 1–37. ISBN: 9780470517338. DOI: https://doi.org/10.1002/9780470517338.ch1. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9780470517338.ch1. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470517338.ch1.

[3] P. Fratzl et al. "Position-Resolved Small-Angle X-ray Scattering of Complex Biological Materials". In: *Journal of Applied Crystallography* 30.5-2 (1997), pp. 765–769. DOI: https://doi.org/10.1107/S0021889897001775. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1107/S0021889897001775. URL: https://onlinelibrary.wiley.com/doi/abs/10.1107/S0021889897001775.

[4] J. M. Feldkamp et al. "Recent developments in tomographic small-angle X-ray scattering". In: *physica status solidi (a)* 206.8 (2009), pp. 1723–1726. DOI: https://doi.org/10.1002/pssa.200881615. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pssa.200881615. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pssa.200881615.

[5] S. Kim and A.K. Khambampati. "12 - Mathematical concepts for image reconstruction in tomography". In: *Industrial Tomography*. Ed. by Mi Wang. Woodhead Publishing Series in Electronic and Optical Materials. Woodhead Publishing, 2015, pp. 305–346. ISBN: 978-1-78242-118-4. DOI: https://doi.org/10.1016/B978-1-78242-118-4.00012-5. URL: https://www.sciencedirect.com/science/article/pii/B9781782421184000125.

[6] Marianne Liebi et al. "Nanostructure surveys of macroscopic specimens by small-angle scattering tensor tomography". In: *Nature* 527.7578 (Nov. 2015), pp. 349–352. ISSN: 1476-4687. DOI: 10.1038/nature16056. URL: https://doi.org/10.1038/nature16056.

[7] Marianne Liebi et al. "Small-angle X-ray scattering tensor tomography: model of the three-dimensional reciprocal-space map, reconstruction algorithm and angular sampling requirements". In: *Acta Crystallographica. Section A, Foundations and Advances* 74 (2018), pp. 12–24.

[8] Leonard C. Nielsen et al. *Small-angle scattering tensor tomography algorithm for robust reconstruction of complex textures*. 2023. arXiv: 2305.07750 [cond-mat.mtrl-sci].

[9] Volker Michel. Boston, MA: Birkhäuser, 2012. ISBN: 978-0-8176-8403-7. DOI: https://doi.org/10.1007/978-0-8176-8403-7.

[10] Leonard C. Nielsen. "Theoretical and computational advances in small-angle x-ray scattering tensor tomography". Licentiate Thesis. Chalmers University of Technology Gothenburg, Sweden, 2022.

[11] Richard H. Byrd et al. "A Limited Memory Algorithm for Bound Constrained Optimization". In: *SIAM Journal on Scientific Computing* 16.5 (1995), pp. 1190–1208. DOI: 10.1137/0916069. eprint: https://doi.org/10.1137/0916069. URL: https://doi.org/10.1137/0916069.

[12] Ciyou Zhu et al. "Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization". In: *ACM Trans. Math. Softw.* 23.4 (Dec. 1997), pp. 550–560. ISSN: 0098-3500. DOI: 10.1145/279232.279236. URL: https://doi.org/10.1145/279232.279236.

[13] Matthieu Martin Jean-Andre Simeoni. "Functional Inverse Problems on Spheres: Theory, Algorithms and Applications". PhD thesis. Lausanne: IINFCOM, 2020, p. 326. DOI: 10.5075/epfl-thesis-7174. URL: http://infoscience.epfl.ch/record/275337.

[14] Mark A. Wieczorek and Matthias Meschede. "SHTools: Tools for Working with Spherical Harmonics". In: *Geochemistry, Geophysics, Geosystems* 19.8 (2018), pp. 2574–2592. DOI: https://doi.org/10.1029/2018GC007529. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018GC007529. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GC007529.

[15] J.R. Driscoll and D.M. Healy. "Computing Fourier Transforms and Convolutions on the 2-Sphere". In: *Advances in Applied Mathematics* 15.2 (1994), pp. 202–250. ISSN: 0196-8858. DOI: https://doi.org/10.1006/aama.1994.1008. URL: https://www.sciencedirect.com/science/article/pii/S0196885884710086.

[16] Patrick J. Roddy and Jason D. McEwen. "Sifting Convolution on the Sphere". In: *IEEE Signal Processing Letters* 28 (2021), pp. 304–308. DOI: 10.1109/lsp.2021.3050961. URL: https://doi.org/10.1109%5C%2Flsp.2021.3050961.

[17] Kerstin Hesse, Ian H. Sloan, and Robert S. Womersley. "Numerical Integration on the Sphere". In: *Handbook of Geomathematics*. Ed. by Willi Freeden, M. Zuhair Nashed, and Thomas Sonar. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 2671–2710. ISBN: 978-3-642-54551-1. DOI: 10.1007/978-3-642-54551-1_40. URL: https://doi.org/10.1007/978-3-642-54551-1_40.

[18] Lennart Råde and Bertil Westergren. Studentlitteratur AB, 2004. ISBN: 9789144031095. DOI: https://doi.org/10.1002/9780470517338.ch1.

[19] K. M. Gorski et al. “HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere”. In: *The Astrophysical Journal* 622.2 (Apr. 2005), pp. 759–771. DOI: 10.1086/427976. URL: https://doi.org/10.1086%5C%2F427976.

[20] Andrea Zonca et al. “healpy: equal area pixelization and spherical harmonics transforms for data on the sphere in Python”. In: *Journal of Open Source Software* 4.35 (Mar. 2019), p. 1298. DOI: 10.21105/joss.01298. URL: https://doi.org/10.21105/joss.01298.

[21] M Kemoli Arthur. “Point Picking and Distributing on the Disc and Sphere”. In: 2015.

[22] Pauli Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[23] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. “Nonlinear total variation based noise removal algorithms”. In: *Physica D: Nonlinear Phenomena* 60.1 (1992), pp. 259–268. ISSN: 0167-2789. DOI: https://doi.org/10.1016/0167-2789(92)90242-F. URL: https://www.sciencedirect.com/science/article/pii/016727899290242F.

[24] Emmanuel J. Candes and Michael B. Wakin. “An Introduction To Compressive Sampling”. In: *IEEE Signal Processing Magazine* 25.2 (2008), pp. 21–30. DOI: 10.1109/MSP.2007.914731.

[25] Rowan Leary et al. “Compressed sensing electron tomography”. In: *Ultramicroscopy* 131 (2013), pp. 70–91. ISSN: 0304-3991. DOI: https://doi.org/10.1016/j.ultramic.2013.03.019. URL: https://www.sciencedirect.com/science/article/pii/S0304399113000892.

[26] Yan Zhang, Yuanyuan Wang, and Chen Zhang. “Total variation based gradient descent algorithm for sparse-view photoacoustic image reconstruction”. In: *Ultrasonics* 52.8 (2012), pp. 1046–1055. ISSN: 0041-624X. DOI: https://doi.org/10.1016/j.ultras.2012.08.012. URL: https://www.sciencedirect.com/science/article/pii/S0041624X12001655.

[27] Kristian Bredies, Karl Kunisch, and Thomas Pock. “Total Generalized Variation”. In: *SIAM Journal on Imaging Sciences* 3.3 (2010), pp. 492–526. DOI: 10.1137/090769521. eprint: https://doi.org/10.1137/090769521. URL: https://doi.org/10.1137/090769521.

[28] Per Christian Hansen and Dianne P. O’Leary. “The Use of the L-Curve in the Regularization of Discrete Ill-Posed Problems”. In: *SIAM J. Sci. Comput.* 14 (1993), pp. 1487–1503.

[29] Zirui Gao et al. “High-speed tensor tomography: iterative reconstruction tensor tomography (IRTT) algorithm”. In: *Acta Crystallographica Section A* 75.2 (Mar. 2019), pp. 223–238. DOI: 10.1107/S2053273318017394. URL: https://doi.org/10.1107/S2053273318017394.

A

Further Results

A.1 Correlation

In Figure A.1 and Figure A.2 the correlation results under L1- and L2-regularization are plotted. The results are similar under all regularization terms tested.

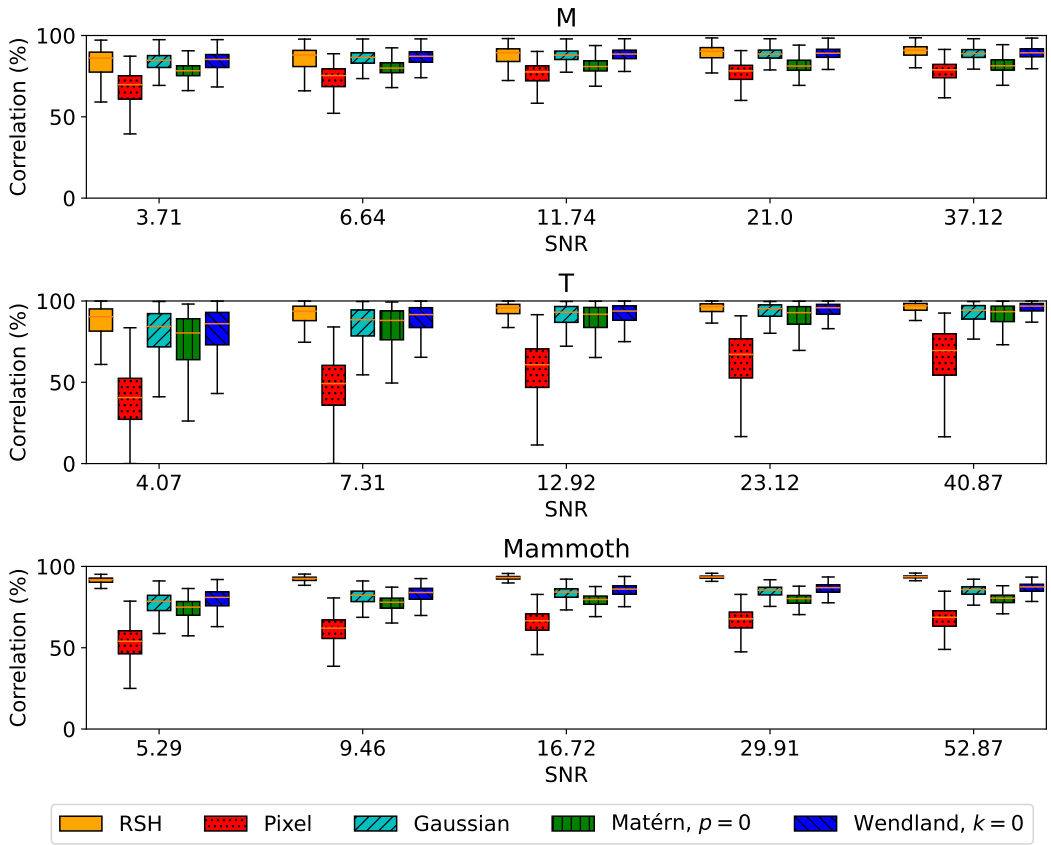


Figure A.1: A Boxplot of correlations over the main volume of a series of reconstructions. In this case, L1-regularization has been applied.

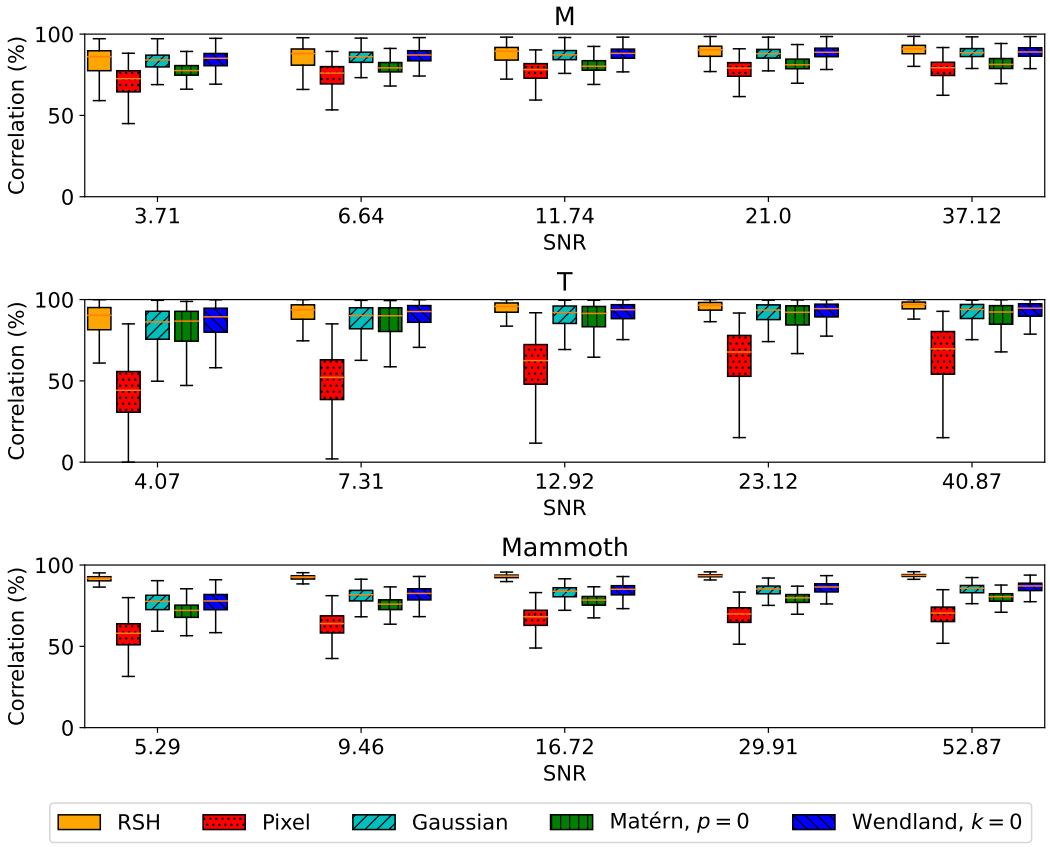


Figure A.2: A Boxplot of correlations over the main volume of a series of reconstructions. In this case, L2-regularization has been applied.

B

Normalization of Radial Basis Functions

RBFs can be normalized numerically or analytically. In the end, the former approach was applied. In what follows, the method used to numerically normalize an RBF is described and error estimates provided.

In regards to analytic normalization, there is one theorem which would be useful should this procedure be necessary. Theorem 4.16 from Michel 2012 states [9, chapter],

$$\int_{\Omega} G(\xi \cdot \eta)\,\mathrm{d}\omega(\eta) = 2\pi \int_{-1}^{1} G(t)\,\mathrm{d}t, \tag{B.1}$$

where $\xi, \eta \in \Omega$. To normalize an RBF using a Euclidean distance function, the scalar product would be expressed in terms of that distance.

B.1 Numerical Normalization

The mean of a function $f(\mathbf{p})$ over a sphere can be defined,

$$\bar{f} = \frac{1}{4\pi} \int_{\Omega} f(\mathbf{p})\,\mathrm{d}\omega(\mathbf{p}). \tag{B.2}$$

For a given spherical function, the mean can be evaluated using the cubature rule in Eq. (17) in [17]. Using a pseudo-even point grid $G = \{\mathbf{p}_i\}_{i=1}^{n}$ with corresponding cells $\{C_i\}_{i=1}^{n}$ and cell areas $\{\mathcal{A}_i\}_{i=1}^{n}$, the mean is,

$$\bar{f} = \frac{1}{4\pi} \sum_{i=1}^{n} a_i f(\mathbf{p}_i) + e_{mean}, \tag{B.3}$$

where the error e_{mean} is estimated using,

$$|e_{mean}| \leq \frac{1}{4\pi} \sum_{i=1}^{n} a_i \sup_{\mathbf{p},\mathbf{q} \in C_i} |f(\mathbf{p}) - f(\mathbf{q})| \leq \max_{i=1,2,\ldots,n} \sup_{\mathbf{p},\mathbf{q} \in C_i} |f(\mathbf{p}) - f(\mathbf{q})|. \tag{B.4}$$

Applying the above equations on an RSM and assuming that the representation is exact (neglecting the approximation error) such that $f(\mathbf{p}) = M(\mathbf{r}, \mathbf{p})$ at a fixed $\mathbf{r}$, we can determine its mean,

$$\bar{M}(\mathbf{r}) = \frac{1}{4\pi} \int_{\Omega} \sum_{i=1}^{n} a_i(\mathbf{r}) K(D(\mathbf{p}_i, \mathbf{p}))\,\mathrm{d}\omega(\mathbf{p}) = \tag{B.5}$$

$$= \sum_{i=1}^{n} a_i(\mathbf{r}) \frac{1}{4\pi} \int_{\Omega} K(D(\mathbf{p}_i, \mathbf{p}))\,\mathrm{d}\omega(\mathbf{p}), \tag{B.6}$$

where a_i is the coefficient of each basis function $K(D(\mathbf{p}_i, \mathbf{p}))$ with kernel function K and distance function D, centered in grid point $\mathbf{p}_i$. So if the basis functions B_i are normalized such that their respective mean is one, the mean of the RSM is the sum of the coefficients a_i. The error for a given basis function $K(D(\mathbf{p}_k, \mathbf{p}))$ is bounded by,

$$|e_{mean}| \leq \max_{i=1,2,\ldots,n} \sup_{\mathbf{p}\in C_i} |K(D(\mathbf{p}_k, \mathbf{p})) - K(D(\mathbf{p}_k, \mathbf{p}))| \leq \tag{B.7}$$

$$\leq \left|\max_x K^{'}(x)\right| \left|\max_{i=1,2,\ldots,n} \sup_{\mathbf{p}\in C_i} D(\mathbf{p}_k, \mathbf{p})\right| = |K^{'}_{\max} D_{\max}|, \tag{B.8}$$

which is simply the maximum of the derivative of the kernel function multiplied by the maximum diameter of any cell. The three kernel functions used the most were the Gaussian, Matérn ($p = 0$) and Wendland ($k = 0$) kernels. The maximum absolute derivatives are provided in Tab. B.1.

Kernel function	Maximum absolute derivative
Gaussian	$(1/\epsilon)\exp(-1/2)$
Matérn, $p = 0$	$1/\epsilon$
Wendland, $k = 0$	$2/\epsilon$

Table B.1: Maximum absolute derivatives of the kernel function most used in the project.

The integration grid had 2000 points per basis function, which is 48000 points when the representation to normalize is based on HEALPix with $N_{\text{side}} = 2$ as for most of the results in this thesis. The integration grid was of the Bauer type, and with 48000 points the maximum diameter of any cell is approximately 0.023. We set the relative scale parameter to 0.7 in the case of the Gaussian, 0.7 for the Matérn kernel and 2.3 for the Wendland kernel. The scale parameter is then the product of the relative scale parameter and the nodal width which is about 0.41 in this case. In Tab. B.2, the error estimates are given, both the absolute errors and the relative errors. The relative errors are the absolute errors divided by the approximation of the mean of the corresponding basis function. The relative error for the Matérn function may seem a bit high, although it should be remembered that this is an upper bound.

Kernel function	Absolute error	Relative error (%)
Gaussian	0.029	3
Matérn, $p = 0$	0.080	8.2
Wendland, $k = 0$	0.049	4.5

Table B.2: Maximum errors of the kernel function most used in the project.

C

Validation of the Spherical Harmonics Transform

In order to validate the RSH transform, we may compare the mean and variance of a function before and after the transform is applied. For the mean of a given function, these measures can be determined using the approach described in section B.1. The variance of a spherical function can similarly be defined,

$$\sigma_f^2 = \frac{1}{4\pi}\int_\Omega \left(f(\mathbf{p}) - \bar{f}\right)^2 \mathrm{d}\omega(\mathbf{p}) = \tag{C.1}$$

$$= \frac{1}{4\pi}\int_\Omega \left(f^2(\mathbf{p}) - 2\bar{f}f(\mathbf{p}) + \bar{f}^2\right) \mathrm{d}\omega(\mathbf{p}) = \tag{C.2}$$

$$= \frac{1}{4\pi}\int_\Omega f^2(\mathbf{p})\,\mathrm{d}\omega(\mathbf{p}) - \bar{f}^2. \tag{C.3}$$

The variance can be determined numerically using,

$$\sigma_f^2 = \frac{1}{4\pi}\sum_{i=1}^{n} a_i f^2(\mathbf{p}_i) - \left(\frac{1}{4\pi}\sum_{i=1}^{n} a_i f(\mathbf{p}_i)\right)^2 + e_{var}, \tag{C.4}$$

where the error is given analogously to the case of the numerical calculation of the mean.

Now, we need to determine the mean and variance of a spherical function in the RSH representation. The mean simply corresponds to the Y_{00} component, whereas the variance is determined from the sum of components in the power spectrum where $l \neq 0$. Comparing these measures before and after the transformation is applied to a given representation gives us an indication as to whether the transformation is correctly implemented.

D

Normalization of the Pixel Segment Projection

We have already calculated the overlap of pixels along the circle and the detector segments. This overlap is expressed in radians. In order to calculate the pseudo-inverse of this projection and to normalize the projection matrix, we need to analyze it sequentially. Firstly, if we normalize the spherical pixel basis with the square root of the pixel area, the resulting basis on the sphere is orthonormal. Secondly, a function represented in this basis is projected onto the circle corresponding to the detector plane. This basis of pixels along the circle is not normalized since the height of the box functions is constant over the circle but the width, i.e. the cross-section of a pixel with the circle, is not. So to get an orthonormal basis we need to rescale this basis using the square root of the width of each box function. Lastly, the function is projected from the circular pixel basis to the segment basis which likewise needs to be normalized.

A function on the sphere can be expressed as,

$$\boldsymbol{f} = \boldsymbol{P}\mathbf{a} = \boldsymbol{P}\boldsymbol{A}^{-1}\boldsymbol{A}\mathbf{a}, \tag{D.1}$$

where $\boldsymbol{f}$ is the function, $\boldsymbol{P}$ is the spherical pixel basis, where each column is a box function with height 1, $\mathbf{a}$ is a vector of coefficients for the box functions and $\boldsymbol{A}$ is the square root of the pixel area for each pixel. $\boldsymbol{A}$ is a diagonal matrix where each diagonal element is the same if HEALPix is used. The rescaled coefficients can be denoted $\mathbf{a}' = \boldsymbol{A}\mathbf{a}$.

Now the full projection can be expressed,

$$\boldsymbol{s} = \boldsymbol{V}^{-1}\boldsymbol{V}\boldsymbol{s} = \boldsymbol{M}\mathbf{a} = \boldsymbol{M}\boldsymbol{W}^{-1}\boldsymbol{W}\boldsymbol{A}^{-1}\boldsymbol{A}\mathbf{a}, \tag{D.2}$$

where $\boldsymbol{s}$ is a vector of coefficients for the function on the circle in the detector segment basis with basis functions whose height is 1, $\boldsymbol{W}$ is the square root of the width of each pixel along the circle and $\boldsymbol{V}$ is the square root of the width of each segment along the circle. We will denote $\boldsymbol{s}' = \boldsymbol{V}\boldsymbol{s}$ which allows us to write,

$$\boldsymbol{s}' = \boldsymbol{V}\boldsymbol{M}\boldsymbol{W}^{-1}\boldsymbol{W}\boldsymbol{A}^{-1}\mathbf{a}'. \tag{D.3}$$

So we have a projection from one orthonormal basis to another but clearly the function represented in one basis can not have the same norm as the one represented in the other. However, $\boldsymbol{W}\boldsymbol{A}^{-1}\mathbf{a}'$ actually represents the projection

to the orthonormal pixel basis on the circle. A function represented in this basis would not in general be representable in the detector segment basis, but the norm should be retained. So an indefinite number of projections and back-projections between theses bases should not diverge or go to zero, although as is easily realized on inspection, the function will be smeared out in each iteration.

So, it should be true that $(\boldsymbol{VMW}^{-1})^{-1} \approx (\boldsymbol{VMW}^{-1})^T$ which leads to,

$$(\boldsymbol{VMW}^{-1}\boldsymbol{WA}^{-1})^{-1}\boldsymbol{s}' = (\boldsymbol{WA}^{-1})^{-1}(\boldsymbol{VMW}^{-1})^T\boldsymbol{s}' = \mathbf{1}_{\uparrow}\mathbf{a}', \tag{D.4}$$

where $\mathbf{1}_{\uparrow}$ is more or less an upsampled identity matrix where each 1 becomes a block matrix. It represents the smearing effect of the back-projection.

Expanding this equation we get,

$$\boldsymbol{AW}^{-1}\boldsymbol{W}^{-1}\boldsymbol{M}^T\boldsymbol{V}\boldsymbol{s}' = \mathbf{1}_{\uparrow}\mathbf{a}', \tag{D.5}$$

where $\boldsymbol{W}$ and $\boldsymbol{V}$ are symmetric. Lastly we get,

$$\boldsymbol{AW}^{-1}\boldsymbol{W}^{-1}\boldsymbol{M}^T\boldsymbol{V}\boldsymbol{V}\boldsymbol{s} = \mathbf{1}_{\uparrow}\boldsymbol{A}\mathbf{a}, \tag{D.6}$$

although this is equivalent and not necessarily more convenient.

The approach described in this appendix might be adaptable to RBFs but the fact that these bases are not orthogonal makes the analysis difficult.

DEPARTMENT OF PHYSICS
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY