

Fast Variance Prediction for Iteratively Reconstructed CT Images With Locally Quadratic Regularization

Stephen M. Schmitt, *Student Member, IEEE*, Mitchell M. Goodsitt, and Jeffrey A. Fessler*, *Fellow, IEEE*

Abstract—Predicting noise properties of iteratively reconstructed CT images is useful for analyzing reconstruction methods; for example, local noise power spectrum (NPS) predictions may be used to quantify the detectability of an image feature, to design regularization methods, or to determine dynamic tube current adjustment during a CT scan. This paper presents a method for fast prediction of reconstructed image variance and local NPS for statistical reconstruction methods using quadratic or locally quadratic regularization. Previous methods either require impractical computation times to generate an approximate map of the variance of each reconstructed voxel, or are restricted to specific CT geometries. Our method can produce a variance map of the entire image, for locally shift-invariant CT geometries with sufficiently fine angular sampling, using a computation time comparable to a single back-projection. The method requires only the projection data to be used in the reconstruction, not a reconstruction itself, and is reasonably accurate except near image edges where edge-preserving regularization behaves highly nonlinearly. We evaluate the accuracy of our method using reconstructions of both simulated CT data and real CT scans of a thorax phantom.

Index Terms—Bayes methods, computed tomography, optimization methods, reconstruction algorithms.

I. INTRODUCTION

ITERATIVE reconstruction (IR) methods for computed tomography (CT) are receiving increased attention for their improved resolution and noise properties compared to FBP [1]–[3]. However, the statistical properties of IR images are difficult to compute compared to FBP. Closed-form but computationally intractable matrix expressions exist [4] for the mean and covariance matrix of the reconstructed images, so faster prediction methods are desirable. Image variance information is useful for image analysis and regularization design [5]. In addition, adjusting X-ray tube current during

a scan has the potential to reduce dose [6], [7], and with sufficiently fast variance prediction, it could be feasible to compute the proper tube current adjustment during a scan to meet a certain variance target.

One way to determine the noise level in reconstructed images would be by finding the empirical variance from an ensemble of reconstructions. This method is extremely computationally intensive, requiring numerous reconstructions, and would require unacceptable X-ray dose for *in vivo* human data. The empirical approach determines the mean and variance of all voxels simultaneously, providing complete variance maps. Prior work on variance prediction has exploited the approximate shift-invariance of projection and back-projection operations to develop DFT-based approximations for the variance map of images reconstructed from scans having locally shift-invariant system geometries with sufficiently fine angular sampling. When it is possible to assume global shift-invariance of projection and back-projection (e.g., [8], [9], for PET), the DFT of projection and back-projection can be calculated once and applied to find the variance map of an entire volume. When one can assume only *local* shift invariance rather than *global* shift invariance, then DFT-based methods are useful for theoretical analysis but require projection and back-projection for each voxel of interest to calculate the DFT [10] and are impractical in general for producing a variance map for an entire large volume. For systems having sufficient symmetries (e.g., cylindrical 3D PET [10]), one can precompute and store a collection of DFT arrays; such computation and storage appears prohibitive for axial CT and inapplicable to short-scan and helical CT. There are methods specific for 2D fan-beam [11], 3D step-and-shoot [12], 3D axial CT [13], and 3D helical CT [14] that further approximate the DFT such that the computational load of predicting variance maps for an entire volume is greatly reduced.

The main contribution of this paper is a method for predicting the variance of iteratively reconstructed CT images with locally shift-invariant scan geometries. The proposed method is significantly faster than previous methods and does not require any estimate or reconstruction of the image, only the scan geometry and the weighting data to be used in the reconstruction. We also extend our previous methods [14] to deal with general weightings in the data-fit term and to accommodate space-variant regularizers such as the uniform-resolution regularizer described in [15]. Previous methods for specific geometries [11], [13], [14] are special cases of the formulation derived here. Section II specifies the general

Manuscript received April 12, 2016; revised June 16, 2016; accepted June 21, 2016. Date of publication July 19, 2016; date of current version December 29, 2016. This work was supported in part by NIH Grants R01 HL-098686 and U01EB018753 and equipment donations from Intel Corporation. *Asterisk indicates corresponding author.*

S. M. Schmitt is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: smschm@umich.edu).

M. M. Goodsitt is with the Department of Radiology, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: goodsitt@med.umich.edu).

*J. A. Fessler is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: fessler@umich.edu).

Digital Object Identifier 10.1109/TMI.2016.2593259

form of the CT reconstruction problem to which our methods apply and provides an overview of the first steps in many existing variance prediction methods. The general variance prediction methods described require the *local* frequency response (LFR) of the projection, weighting, and back-projection operator. Section III shows how we apply an approximation to the LFR, derived in the Appendix,¹ to variance prediction; Section III-B provides a further simplification specific to 3DCT geometries where all of the rays have a reasonably small cone-angle, such as in axial or helical scans. Section IV validates the variance predictions by comparing with the empirical variance of images reconstructed from repeated phantom scans, in both a simulation study and a real-world study.

II. BACKGROUND

A. Problem Domain

This paper develops methods for predicting the variance (i.e., $\text{var}(\hat{x}_j)$) of statistical image reconstructions that take the general form

$$\hat{\mathbf{x}} = \text{argmin}_{\mathbf{x}} L(\mathbf{Y}; \mathbf{x}) + \alpha R(\mathbf{x}). \quad (1)$$

Here, L is a data-fit term, commonly the negative log-likelihood of the N_{obs} vectorized observations \mathbf{Y} given an image vector \mathbf{x} composed of N_{vox} voxel attenuations. The observation vector \mathbf{Y} can represent either estimates of photon counts or of line integrals. The function $R(\mathbf{x})$ is a regularization penalty. We make the following assumptions:

- 1) The covariance of \mathbf{Y} is diagonal, and can be estimated from the data and knowledge of the instrumentation.
- 2) Given an image \mathbf{x} , the elements of \mathbf{Y} are statistically independent, and the likelihood of a particular observation Y_i is modeled in terms of the projection $[\mathbf{Ax}]_i \triangleq \sum_{j=1}^{N_{\text{vox}}} a_{ij} x_j$, such that

$$L(\mathbf{Y}; \mathbf{x}) = \sum_{i=1}^{N_{\text{obs}}} L_i(Y_i; [\mathbf{Ax}]_i); \quad (2)$$

the $N_{\text{obs}} \times N_{\text{vox}}$ system matrix \mathbf{A} has elements a_{ij} representing the projection of voxel j onto observation i . This form can account for both weighted least-squares data-fit terms, in which Y_i represents a line integral estimate and the log-likelihood is given by

$$L_i(Y_i; [\mathbf{Ax}]_i) = \frac{1}{2} w_i (Y_i - [\mathbf{Ax}]_i)^2, \quad (3)$$

and for a Poisson likelihood, in which Y_i represents a photon count, and the log-likelihood is given by [16]:

$$L_i(Y_i; [\mathbf{Ax}]_i) = \bar{Y}_i - Y_i \ln \bar{Y}_i, \quad \bar{Y}_i \triangleq b_i e^{-[\mathbf{Ax}]_i} + r_i. \quad (4)$$

It is important to note that any mismatch between L in (2) and a hypothetical “true” likelihood \check{L} that perfectly matches the true physics of a CT system does not affect our prediction methods. Even if L is a poor approximation of \check{L} , and the resulting reconstruction $\hat{\mathbf{x}}$ is a poor image, our methods can still predict the variance of $\hat{\mathbf{x}}$, so long as L is correctly characterized.

In particular, our method accurately predicts the variance even in the usual case where there is model mismatch between the system matrix \mathbf{A} and the actual scanner physics. An accurate characterization of the \mathbf{A} used in a particular reconstruction algorithm is much more important to predicting the variance of that reconstruction than whether \mathbf{A} accurately characterizes the physics. (Of course, the usefulness of the reconstructed image $\hat{\mathbf{x}}$ will depend on the accuracy of \mathbf{A} .)

- 3) The regularizer takes the general form

$$R(\mathbf{x}) = \sum_{d=1}^{N_C} r_d \sum_k \psi([\mathbf{C}_d \mathbf{x}]_k). \quad (5)$$

In the common case of a regularizer that penalizes first differences between neighboring voxels, d indexes the directions over which we take the differences, \mathbf{C}_d is a first differencing matrix between voxels in that direction, and r_d is the relative strength of the regularizer in that direction. We assume the regularizer penalty ψ is twice-differentiable at 0, and scaled such that $\psi''(0) = 1$.

These assumptions account for many common choices of data-fit terms and regularization penalties. Under these conditions on the reconstruction, we adapt [4] to the following form for the approximate covariance of the resulting reconstruction:

$$\text{cov}(\hat{\mathbf{x}}) \approx (\mathbf{A}^T \mathbf{W} \mathbf{A} + \alpha \nabla^2 R(\check{\mathbf{x}}))^{-1} \mathbf{A}^T \hat{\mathbf{W}} \mathbf{A}. \quad (6)$$

$$(\mathbf{A}^T \mathbf{W} \mathbf{A} + \alpha \nabla^2 R(\check{\mathbf{x}}))^{-1},$$

where the diagonal matrices \mathbf{W} and $\hat{\mathbf{W}}$ are defined as:

$$[\mathbf{W}]_{ii} \triangleq \left. \frac{\partial^2}{\partial y^2} L_i(Y_i; y) \right|_{y=[\mathbf{Ax}]_i} \quad (7)$$

$$[\hat{\mathbf{W}}]_{ii} \triangleq \text{var}(Y_i) \left(\left. \frac{\partial^2}{\partial y \partial Y_i} L_i(Y_i; y) \right|_{y=[\mathbf{Ax}]_i} \right)^2. \quad (8)$$

In (6)–(8), $\check{\mathbf{x}}$ denotes the reconstruction using noise-free data $\check{\mathbf{Y}}$. (Our final formula will not use $\check{\mathbf{x}}$.)

For a Gaussian log-likelihood in (3) with weight w_i for observation i , the weighting matrix \mathbf{W} in (7) is simply these weights as a diagonal matrix: $[\mathbf{W}]_{ii} = w_i$. The matrix $\hat{\mathbf{W}}$ has diagonal elements $[\hat{\mathbf{W}}]_{ii} = w_i^2 \text{var}(Y_i)$; if the weights are chosen so that $w_i \approx 1/\text{var}(Y_i)$, then $\hat{\mathbf{W}} \approx \mathbf{W}$.

For the Poisson log-likelihood given in (4), $[\mathbf{W}]_{ii} = Y_i / \bar{Y}_i^2$ and $[\hat{\mathbf{W}}]_{ii} = \text{var}(Y_i) / \bar{Y}_i^2$; both are approximately $1/Y_i$ [17].

For both of these statistical models, \bar{Y}_i and $\text{var}(Y_i)$ are of course unknown. However, our methods are not extremely sensitive to the particular values of \mathbf{W} and $\hat{\mathbf{W}}$, so estimates of these values are acceptable [4]. Typically those estimates do not require having $\hat{\mathbf{x}}$ or $\check{\mathbf{x}}$. We define the matrix \mathbf{P} as:

$$\mathbf{P} \triangleq \sum_{d=1}^{N_C} r_d \mathbf{C}_d^T \mathbf{C}_d. \quad (9)$$

We use \mathbf{P} as an approximation for the Hessian of the regularizer, $\nabla^2 R(\check{\mathbf{x}})$, in approximating (6). The actual Hessian, for regularizers of the form (5), is given by:

$$\nabla^2 R(\check{\mathbf{x}}) = \sum_{d=1}^{N_C} r_d \mathbf{C}_d^T \check{\Psi}_d(\check{\mathbf{x}}) \mathbf{C}_d, \quad (10)$$

¹Supplementary material in the supplementary files / multimedia tab.

where $\ddot{\Psi}_d(\mathbf{x})$ is a diagonal matrix of second derivatives of the penalty function, evaluated for an image \mathbf{x} :

$$[\ddot{\Psi}_d(\check{\mathbf{x}})]_{kk} \triangleq \ddot{\psi}([\mathbf{C}_d \check{\mathbf{x}}]_k). \quad (11)$$

For variance prediction, we use the approximation $\ddot{\Psi}_d(\check{\mathbf{x}}) \approx \mathbf{I}$, such that $\nabla^2 R(\check{\mathbf{x}}) \approx \mathbf{P}$. This approximation (for locally quadratic regularizers like the Huber function) is based on the idea that the majority of neighboring-voxel differences in the reconstruction $\check{\mathbf{x}}$ from noise-free projection data will be small, since the regularizer penalizes large neighboring-voxel differences. We hope that for these small differences, the second derivative of their penalties will be near 1. The utility of this assumption to our purposes of fast variance prediction is enormous. First, variance prediction using $\nabla^2 R(\check{\mathbf{x}})$ would require foreknowledge of the noiseless reconstruction $\check{\mathbf{x}}$. For non-phantom applications, $\check{\mathbf{x}}$ is unavailable. Second, even using a noisy reconstruction $\hat{\mathbf{x}}$ requires the time to compute $\hat{\mathbf{x}}$, which is much greater than the computation time of our fast methods. Using $\hat{\mathbf{x}}$ would diminish the practical utility of our methods. Any edge present breaks both the assumption that $\nabla^2 R(\check{\mathbf{x}})$ is shift-invariant, and our ability to use pre-computation to accelerate our algorithm. Eqn. (9) is exactly the Hessian of R for quadratic regularization.

B. Methods

In general, the matrix $\mathbf{A}^T \mathbf{W} \mathbf{A}$ is not spatially shift-invariant. If it were shift-invariant, except for the truncation to the masked space spanned by the image vector \mathbf{x} , then $\mathbf{A}^T \mathbf{W} \mathbf{A}$ could be diagonalized with an n -dimensional DSFT (discrete space Fourier transform):

$$\mathbf{A}^T \mathbf{W} \mathbf{A} = \mathcal{F}^* \mathcal{D} \left\{ H^W \right\} \mathcal{F},$$

where $H^W(\vec{v})$ would be the n -dimensional DSFT of the impulse response of $\mathbf{A}^T \mathbf{W} \mathbf{A}$, and \mathcal{D} is a ‘‘diagonalization’’ operator:

$$(\mathcal{D}\{H\}X)(\vec{v}) = H(\vec{v})X(\vec{v}),$$

and \mathcal{F} is the DSFT with the spatial extent limited by the image support. It is defined as:

$$(\mathcal{F}\mathbf{x})(\vec{v}) = \sum_k x_k \exp(-j2\pi \vec{v}^T \vec{n}_k) \quad (12)$$

$$[\mathcal{F}^* X]_k = \int_{[-\frac{1}{2}, \frac{1}{2}]^n} X(\vec{v}) \exp(j2\pi \vec{v}^T \vec{n}_k) d\vec{v}, \quad (13)$$

where \vec{v} has units of cycles per sample. Each voxel x_j in an image \mathbf{x} is centered at a spatial position denoted \vec{x}_j , and we assume that these voxel centers are aligned on a grid such that

$$\vec{x}_j = \mathbf{V} \vec{n}_j + \vec{o}, \quad (14)$$

where \vec{n}_j is a voxel index with integer coordinates, \mathbf{V} is a voxel spacing matrix (e.g., $\mathbf{V} = \text{diag}(\Delta_x, \Delta_x, \Delta_z)$ for the common choice of rectangular-cuboid voxels), and \vec{o} is a spatial offset.

The local impulse response (LIR) of $\mathbf{A}^T \mathbf{W} \mathbf{A}$ for the voxel j is defined by

$$\mathbf{h}_j^W \triangleq \mathbf{A}^T \mathbf{W} \mathbf{A} \mathbf{e}_j, \quad (15)$$

where \mathbf{e}_j is defined as the unit vector with a single 1 at position j . This LIR can be written exactly as the impulse \mathbf{e}_j operated on by a frequency-domain filter $H_j^W(\vec{v})$:

$$\mathbf{h}_j^W = \mathcal{F}^* \mathcal{D} \left\{ H_j^W \right\} \mathcal{F} \mathbf{e}_j, \quad (16)$$

where

$$H_j^W = \mathcal{D} \left\{ \exp(j2\pi \vec{v}^T \vec{n}_j) \right\} \mathcal{F} \mathbf{h}_j^W. \quad (17)$$

The diagonal term ‘‘centers’’ the transform at the j th voxel using the shift property of the DSFT. We will refer to H_j^W as a *local* frequency response (LFR). In the region near voxel j , $\mathbf{A}^T \mathbf{W} \mathbf{A}$ is typically *approximately* spatially shift-invariant, leading us to an approximation

$$[\mathbf{A}^T \mathbf{W} \mathbf{A}]_{kj} \approx \mathbf{e}_k^T \mathcal{F}^* \mathcal{D} \left\{ H_j^W \right\} \mathcal{F} \mathbf{e}_j, \quad (18)$$

for voxel k near voxel j , which is suggested by (15) and (16). Except at the edges of the reconstructed image, \mathbf{P} can also be represented in terms of its frequency response $R(\vec{v})$:

$$\mathbf{P} = \mathcal{F}^* \mathcal{D} \{ R \} \mathcal{F}. \quad (19)$$

With both of these matrices diagonalized, (6) approximately simplifies, locally to a voxel j , to

$$\text{cov}(\hat{\mathbf{x}}) \approx \mathcal{F}^* \mathcal{D} \{ S_j \} \mathcal{F}, \quad (20)$$

where

$$S_j(\vec{v}) \triangleq \frac{H_j^{\hat{W}}(\vec{v})}{(H_j^W(\vec{v}) + \alpha R(\vec{v}))^2} \quad (21)$$

is the local noise power spectrum (NPS) of the noise in the reconstruction. Representing the covariance using the NPS in (20) makes the approximation that $\mathcal{F}\mathcal{F}^*$ is an identity operation. In practice it is not exactly an identity because of the finite spatial support of the image considered by the finite sum in (12), but we follow previous work that makes similar approximations [18]. Note here the distinction between H_j^W and $H_j^{\hat{W}}$; $H_j^{\hat{W}}$ comes from the same derivation as (15)–(18), but with $\hat{\mathbf{W}}$ substituted for \mathbf{W} .

If $\mathbf{A}^T \mathbf{W} \mathbf{A}$ were shift-invariant, so that (18) were exact and not a local approximation, (21) would be the global NPS of the noise. Extracting the variance of one voxel can be done by left- and right-multiplying the covariance matrix by unit vectors:

$$\text{var}(\hat{x}_j) = \mathbf{e}_j^T \text{cov}(\hat{\mathbf{x}}) \mathbf{e}_j;$$

plugging in the approximation (20) to this expression simplifies it to an integral of the local NPS:

$$\text{var}(\hat{x}_j) \approx \int_{[-\frac{1}{2}, \frac{1}{2}]^n} S_j(\vec{v}) d\vec{v} \quad (22)$$

Prior work used (22) for variance prediction; [9] and [19] find an empirical LFR from the Fourier transform of \mathbf{h}_j^W , found by projecting, weighting, and back-projecting \mathbf{e}_j . This empirical LFR H_j^W is then numerically integrated in (22). Other work derives a closed form for H_j^W based on the specific CT geometry and projection method used for \mathbf{A} and simplifies (22) for these specific realizations of H_j^W ; [11] has an approximate closed form for 2D fan-beam CT, and [12] has one for a

restricted subset of 3DCT geometries. The Appendix derives an approximate closed form for H_j^W for locally shift-invariant CT geometries that we use for variance prediction.

III. VARIANCE PREDICTION

A. Application to General n -Dimensional CT

Our variance prediction method uses the integral (22) with the NPS expression (21). The local frequency response H_j^W is approximated using (61), found in the supplementary material, which is a factorization of H_j^W into the two terms J and E_j^W :

$$H_j^W(\vec{v}) \approx J(\vec{v})E_j^W\left(\frac{\vec{v}}{\|\vec{v}\|}\right). \quad (23)$$

In the supplement we derive this approximate factorization by separating a continuous-operator approximation of $\mathbf{A}^T\mathbf{W}\mathbf{A}$ into a weighting-like term (that becomes E_j^W) and a continuous-operator analog of the unweighted operator $\mathbf{A}^T\mathbf{A}$. We then apply a first-order Taylor expansion that makes the projection look, local to a voxel of interest and a source position, like a single parallel-beam view with an extra magnification factor. The LFR of this single parallel-beam view can be represented as a slice through the J term; which slice that we take at a particular voxel and source position becomes part of E_j^W . The simplest version of J , for the usual rectangular-cuboid voxels, is given by (65) in the supplement as:

$$J(\vec{v}) = \frac{\text{sinc}^2(v_1)\text{sinc}^2(v_2)\text{sinc}^2(v_3)}{\|\vec{v}\|}.$$

This ratio contains the expected $1/\|\vec{v}\|$ term in the frequency response of projection followed by back-projection with no weighting. The E_j^W term contains the angle-dependent weighting and is given by (63) in the supplement as

$$E_j^W\left(\frac{\vec{v}}{\|\vec{v}\|}\right) = \sum_{\sigma \in \mathcal{B}_j\left(\frac{\vec{v}}{\|\vec{v}\|}\right)} \frac{\tilde{u}_{j,\sigma} \det(\mathbf{R}_{j,\sigma}) \det(\mathbf{V})}{\left(\frac{\partial}{\partial \sigma} \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \frac{\vec{v}}{\|\vec{v}\|}\right)}.$$

Here, $\tilde{u}_{j,\sigma}$ in (48) denotes the element of the statistical weights \mathbf{W} corresponding to the ray from the source at position σ through the j th voxel along ray direction $\hat{\theta}_{j,\sigma}$; $\det(\mathbf{R}_{j,\sigma})$ in (40) is related to the magnification at the j th voxel of the cone-beam geometry; \mathbf{V} in (14) denotes the voxel spacing; and $\mathcal{B}_j\left(\frac{\vec{v}}{\|\vec{v}\|}\right)$ in (50) is the set of source positions σ where $\hat{\theta}_{j,\sigma}$ is perpendicular to $\mathbf{V}^{-T}\vec{v}$.

The separation (23) assumes that $\mathbf{A}^T\mathbf{W}\mathbf{A}$ can be approximated by a continuous operator, and therefore that the detector elements are closely and regularly spaced. We assume that the geometry can be approximated local to a voxel by a Taylor expansion, i.e., that the function that, for any fixed source position, maps a spatial position to its corresponding detector position, is smooth. This is reasonable for standard X-ray CT geometries built from curved or flat detectors, but might not hold for some baggage CT systems that have unusual sets of detector segments with gaps. Such gaps could preclude local shift invariance.

The utility of this factorization is that the first term, J , does not depend on the voxel location j or weighting \mathbf{W} ; the second term, E_j^W does, but does not depend on the magnitude $\|\vec{v}\|$

of the location in frequency space and so has one fewer dimension than the argument of H_j^W . Representing \vec{v} in spherical coordinates, such that $\varrho \triangleq \|\vec{v}\|$ is the spatial frequency magnitude and $\vec{\Theta} \triangleq \vec{v}/\|\vec{v}\|$ is the frequency direction, (22) becomes (see supplementary materials for details):

$$\begin{aligned} \text{var}(\hat{x}_j) &\approx \int_{\mathbb{S}^n} \int_0^{\varrho_{\max}} \frac{H_j^W(\varrho, \vec{\Theta})}{(H_j^W(\varrho, \vec{\Theta}) + \alpha R(\varrho, \vec{\Theta}))^2} \varrho^{n-1} d\varrho d\vec{\Theta} \\ &\approx \int_{\mathbb{S}^n} \int_0^{\varrho_{\max}} \frac{J(\vec{v})E_j^W(\vec{\Theta})}{(J(\vec{v})E_j^W(\vec{\Theta}) + \alpha R(\varrho, \vec{\Theta}))^2} \varrho^{n-1} d\varrho d\vec{\Theta} \\ &= \alpha^{-1} \left(\int_{\mathbb{S}^n} \frac{E_j^W(\vec{\Theta})}{E_j^W(\vec{\Theta})} \right. \\ &\quad \left. \int_0^{\varrho_{\max}} \frac{\alpha^{-1} E_j^W(\vec{\Theta}) J(\vec{v})}{(\alpha^{-1} E_j^W(\vec{\Theta}) J(\vec{v}) + R(\varrho, \vec{\Theta}))^2} \varrho^{n-1} d\varrho d\vec{\Theta} \right) \\ &= \alpha^{-1} \int_{\mathbb{S}^n} \frac{E_j^W(\vec{\Theta})}{E_j^W(\vec{\Theta})} G(\alpha^{-1} E_j^W(\vec{\Theta}), \vec{\Theta}) d\vec{\Theta}, \quad (24) \end{aligned}$$

where $n = 3$ for 3D images and $n = 2$ for 2D images. Here, the integral over \mathbb{S}^n is taken over the surface of the n -dimensional sphere and $\vec{\Theta}$ represents a particular point on the surface, so (24) is an $(n-1)$ -dimensional integral. The object-independent function $G(\gamma, \vec{\Theta})$ is defined as

$$G(\gamma, \vec{\Theta}) \triangleq \int_0^{\varrho_{\max}(\vec{\Theta})} \frac{\gamma J(\varrho, \vec{\Theta})}{(\gamma J(\varrho, \vec{\Theta}) + R(\varrho, \vec{\Theta}))^2} \varrho^{n-1} d\varrho, \quad (25)$$

where $\varrho_{\max}(\vec{\Theta})$ is the maximum extent of ϱ in $[-1/2, 1/2]^n$:

$$\varrho_{\max}(\vec{\Theta}) = \frac{1}{2\|\vec{\Theta}\|_{\infty}}.$$

In general, G cannot be computed in a closed form, but it is well-behaved and only depends on, other than its arguments γ and $\vec{\Theta}$, the regularizer, which determines $R(\vec{v})$, and the scale-invariant form of the voxel, which determines $J(\vec{v})$. The J term will be the same for all cuboids that can tile space, for example, but not the same as the J term for an image reconstructed using blob-based voxels [20].

For a particular voxel shape and regularizer, we precompute a single table of values of G and use that table to predict variance maps via (24) for multiple voxels, any regularization parameter α , any weighting \mathbf{W} , any voxel spacing or scan geometry.

B. Application to 3D Axial and Helical Cone-Beam CT

Section VII-F of the Appendix in the supplemental materials derives the following further approximation to the two-term factorization (23) seen in the previous section that is specific to 3DCT geometries with small cone angles:

$$H_j^W(\vec{v}) \approx J_{\text{cyl}}(\vec{v})E_{j,\text{cyl}}^W(\Phi), \quad (26)$$

shown in (71) (in the supplementary materials). In this small cone-angle factorization, the first term $J_{\text{cyl}}(\vec{v}) = J(\vec{v}) \sec \Theta$

in (70) is again independent of voxel location j and weighting \mathbf{W} . The second term $E_{j,\text{cyl}}$ in (72) is:

$$E_{j,\text{cyl}}^W(\Phi) \approx \frac{D_{\text{sd}}^2 \Delta_x^3 \Delta_z}{D_{\text{so}}} \sum_{\sigma \in \mathcal{B}_j(\Phi)} \tilde{u}_{j,\sigma} \frac{\|\vec{\ell}_{j,\sigma}\|^2}{D_{2,j,\sigma}^3} |\csc(\Phi - \sigma)|,$$

where D_{sd} and D_{so} denote the distances between the source and the detector and to the isocenter respectively, $\vec{\ell}_{j,\sigma}$ in (66) denotes the ray segment from the source at σ to voxel position \vec{x}_j , and $D_{2,j,\sigma}$ in (67) denotes the length of the projection of that segment into the xy -plane. Unlike the E_j in the previous section, here it is a function of only one variable, the azimuthal angle Φ of \vec{v} in 3D cylindrical coordinates.

These terms are derived from the factorization used in (23), applying a further approximation that takes advantage of the fact that, for a small cone angle, this spherically separable approximation is nearly cylindrically separable except close to the missing cone in frequency space around the v_3 axis. Appendix Section VII-F shows that for the purposes of variance prediction, accuracy near this missing cone is less important.

Using the LFR approximation (26), we rearrange the variance prediction integral (22) by changing from Cartesian coordinates to cylindrical coordinates (ρ, Φ, v_3) , to be

$$\text{var}(\hat{x}_j) \approx \alpha^{-1} \int_0^{2\pi} \frac{E_{j,\text{cyl}}^W(\Phi)}{E_{j,\text{cyl}}^W(\Phi)} G_{\text{cyl}}(\alpha^{-1} E_{j,\text{cyl}}^W(\Phi), \Phi) d\Phi, \quad (27)$$

where we define another object-independent function G_{cyl} :

$$G_{\text{cyl}}(\gamma, \Phi) \triangleq \int_0^{\rho_{\text{max}}(\Phi)} \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{\gamma J_{\text{cyl}}(\vec{v})}{(\gamma J_{\text{cyl}}(\vec{v}) + R(\vec{v}))^2} \rho dv_3 d\rho. \quad (28)$$

In this case, $\rho_{\text{max}} = 1/(2\max\{|\cos \Phi|, |\sin \Phi|\})$. Again, G_{cyl} has no closed form but is a well-behaved function of only two parameters that we precompute and tabulate. We compute this table only once for a given differencing matrix \mathbf{C} and voxel shape; a particular image, weighting, system geometry, or regularization parameter α does not change the table G_{cyl} . Using the table, variance prediction via (27) simply requires looking up values of G_{cyl} and numerically integrating them in 1D. This integration can be evaluated using a coarse discretization of Φ with reasonably accurate predicted variance, especially given that the integrand is periodic and integrated over one period, a case in which numerical integration converges quickly [21]. While the method of derivation is changed, this is the form for fast variance prediction given in [14], which also reduces to the form given in [13] for quadratic regularization and an axial geometry.

For 3DCT geometries where (26) is an inaccurate approximation, such as those where a voxel under consideration has rays passing through it in directions that cover much of \mathbb{S}^3 , one must revert to (24) for fast variance prediction.

C. Spatially Varying Regularization

In this section we consider the effect on reconstruction variance of using a spatially varying regularizer such as that defined in [15], [22] designed to produce a reconstruction

with uniform spatial resolution. Each voxel has an associated factor κ_j^2 representing the 'certainty' of the voxel that multiplies the effect of α . This factor modulates the smoothing effect of the regularizer in otherwise less certain regions to promote uniform resolution at the cost of less uniform voxel variance. To adapt our variance prediction method to this situation, we simply define a per-voxel effective regularization parameter $\alpha_{\text{eff}} \triangleq \alpha \kappa_j^2$ and evaluate (27) with this α_{eff} .

The effect of using κ_j^2 in the regularizer is intuitive: assuming that the change in the value of G_{cyl} in (27) is small when α_{eff} is varied compared to the change in the α^{-1} multiplying the integral, the approximate variance decreases inversely with increasing certainty κ_j^2 [8].

D. Object Support Masking

Outside the support of the object there is significant approximation error because, being based on (6), our prediction method ignores the non-negativity constraint that is often used in solving the reconstruction problem (1). The empirical variance outside the object approaches zero, whereas the predicted variance is positive. We use a method similar to [23] to identify regions that are outside the support of the object and set the predicted variance in these regions to zero. Other compensation methods could be incorporated [10].

IV. RESULTS

To evaluate our fast variance prediction approach (27), we compared it to an empirical variance map in two cases. In one case, we computed the empirical variance of reconstructions from multiple realizations of simulations of noisy projection data of an XCAT phantom. In the other case, we repeatedly scanned a physical phantom and computed the empirical variance from the reconstructions of these scans. In both cases we used the weighted least-squares data-fit term (3).

A. Simulation Data

For the simulation study, we reconstructed a $512 \times 512 \times 320$ voxel section of the XCAT phantom [24] with voxel size $\Delta_x \times \Delta_z = 0.9764 \times 0.625\text{mm}$ that covers an anatomical section between the neck and mid-lungs. We simulated a GE third-generation helical system geometry with a 888×64 quarter-offset detector having detector element size $1.0239 \times 1.0964\text{mm}$; the detector went through three turns with a pitch of 1, taking 2952 views. Each reconstruction used 80 iterations of an ordered-subset method [25] using 64 subsets. In the regularizer, \mathbf{C} was a matrix that takes 3 first differences for each voxel, one each for the adjacent voxel in each axis. These differences were penalized by a Huber potential function:

$$\psi(x) = \begin{cases} x^2/2, & |x| \leq \delta \\ \delta|x| - \delta^2/2, & |x| > \delta, \end{cases} \quad (29)$$

where the value of δ was 10 Hounsfield units. We looked at two separate cases for regularization: one with the spatially varying regularization described in section III-C, and one without. The regularization parameter α was set empirically

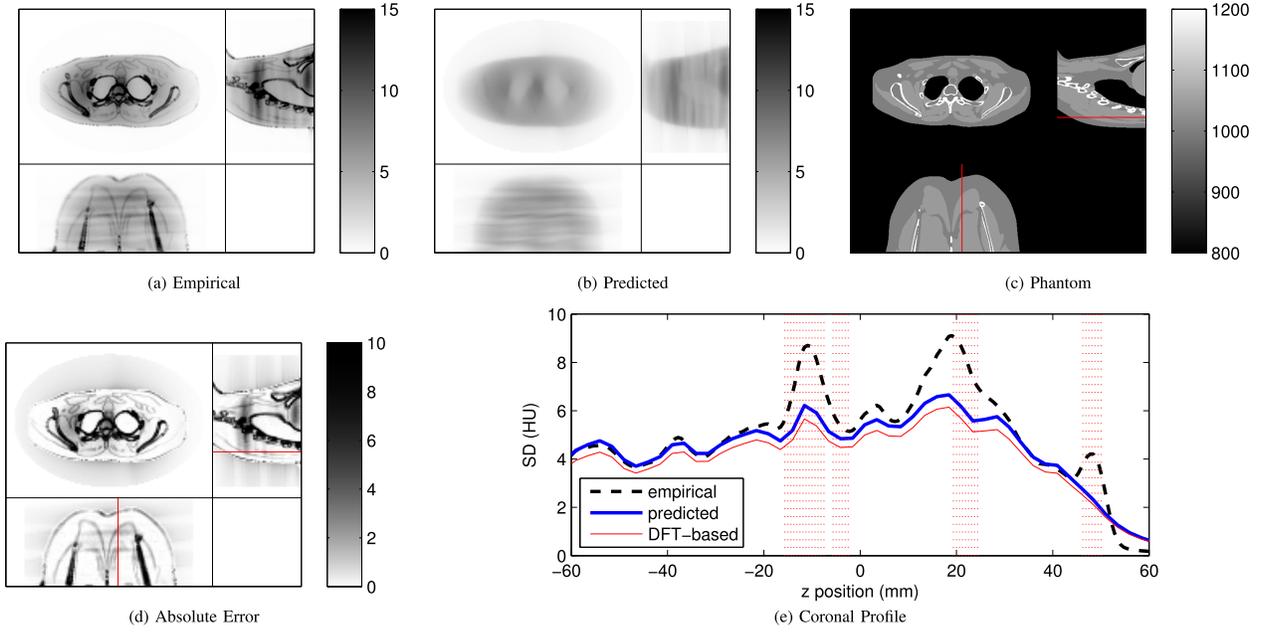


Fig. 1. Three slices of standard deviation maps for simulated reconstructions using spatially varying regularization; the red line in (c) and (d) indicates the profile used in (e). Scales in modified Hounsfield units (air is zero, water is 1000). In panels (a)-(d), the upper left is axial, the lower left is coronal, and the upper right is sagittal. Subsequent figures use the same convention.

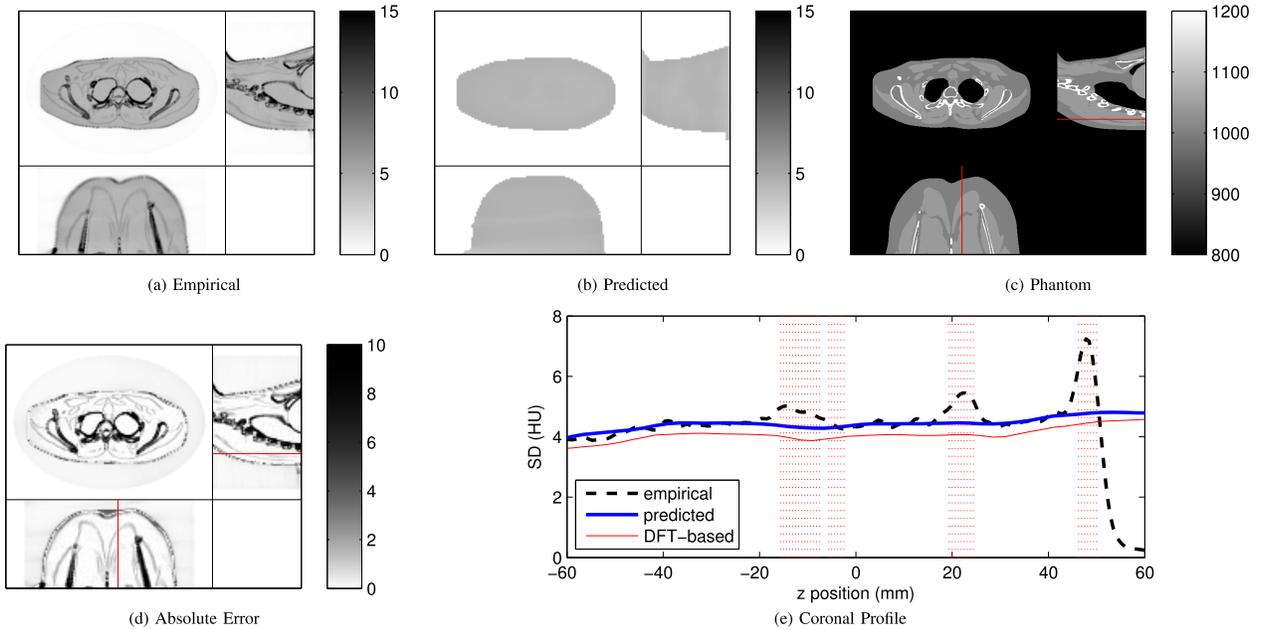


Fig. 2. Three slices of standard deviation maps for simulated reconstructions using uniform regularization; the red line in (c) and (d) indicates the profile used in (e). Scales in modified Hounsfield units.

to 2^6 in the non-spatially-varying case, and to 2^{14} in the spatially varying case. The weighting \mathbf{W} was normalized so that unattenuated rays had a weight of 1. Noise was applied to each of the simulated projections by realizing Poisson random variables with mean equal to the expected number of photons with a photon count incident on the phantom of 10^5 photons per view. The empirical standard deviation maps were produced using 89 realizations of the reconstruction in the case of non-spatially-varying regularization and 111 realizations in the case of spatially varying regularization. Figures 1(a) (with

spatially varying regularization) and 2(a) (uniform regularization) show axial, sagittal, and coronal slices of the 3D map of the empirical standard deviation from our reconstructions. Since the empirical standard deviation maps were noisy and the ground truth standard deviation is spatially slowly varying, we smoothed the empirical variance maps with a gaussian kernel with a FWHM of 3 voxels each in each direction. (Figure 7, in the supplementary material, shows a profile of the unsmoothed standard deviation.) Figures 1(b) and 2(b) show the corresponding slices through the 3D predicted

TABLE I
COMPUTATION TIME OF VARIANCE PREDICTION METHODS
(CPU SECONDS); BOTH CASES USED THE SPATIALLY
VARYING REGULARIZATION IN SECT. III-C.

	Empirical	DFT-based	Proposed
Simulation	$1.64 \cdot 10^7$ (111 realizations)	$7.23 \cdot 10^8$	$1.21 \cdot 10^3$
Real data	$3.63 \cdot 10^5$ (10 scans)	$1.07 \cdot 10^8$	$6.73 \cdot 10^2$

deviation map from (27). Since standard deviation varies slowly, we computed it once per $4 \times 4 \times 4$ block and used nearest-neighbor interpolation to fill in the rest. More sophisticated interpolation could be used, but the interpolation error is minimal compared to the intrinsic approximation error of our method. Figure 1(c) and Figure 2(c) show the ground truth XCAT phantom we used. Figure 1(d) and Figure 2(d) show the absolute magnitude of the error of our approximated standard deviation compared to the empirical results. The gray scale in these figures is transformed to better show the dynamic range of the error. Figure 1(e) and Figure 2(e) show the empirical and predicted standard deviation along a one-dimensional trans-axial profile through the image, behind the center of the spine in the phantom, along with the standard deviation as computed directly from (22) using (17) as the LFR (labeled 'DFT-based'). Along this profile, dotted red lines indicate where there is an edge within one voxel. Sections near the ends of the axial FOV were omitted in all images; the empirical variance becomes extreme due to a suboptimal OS algorithm implementation that is somewhat unstable in regions where the helical sampling is poor. The OS algorithm in [26] would reduce this instability and reduce the empirical variance in the (clinically unimportant) end slices.

The computation time of our method for the entire volume using $4 \times 4 \times 4$ downsampling was 1207 CPU-seconds using one core of an Intel Core i7-860 with 16 GB of memory. The empirical reconstructions took an average of 1.71 CPU-days each using one core of an Intel X5650 processor also with 16GB of memory. Table I compares the computation time required to find the empirical variance (using 111 realizations) with the computation time required to predict the variance for the entire volume using the DFT-based method and our methods. We used the DFT-based method only to produce the one-dimensional profiles shown in Figure 1(e) and Figure 2(e); since the computation time is large, we extrapolate to find the computation time for the entire volume for Table I.

B. Real CT Scans of a Thorax Phantom

For our real-world dataset, we scanned a phantom 10 times with a GE Discovery CT750 HD scanner and reconstructed each of the 10 sinograms separately and produced an empirical variance map of the reconstruction. The phantom was a custom modified CIRS (Norfolk, VA) Model 003 lung nodule simulator phantom for quantitative CT [27], [28]. The geometry of the system is the same as the simulated geometry used in the previous section, with the exception of performing an axial scan using a 16-row detector and 984 views. Since we

could not ensure that each scan began at the same starting angle, using multiple realizations of the same helical scan to produce an empirical variance map was not possible with our physical CT scanner. For reconstructing the axial scans, we used a projection matrix \mathbf{A} that was correctly aligned to the starting angle of each scan so that each reconstruction was aligned to the same voxel grid. We used two different tube currents (40mA, 200mA) for a low-dose and a high-dose scan, and in all cases the tube voltage was 120 kVp and the scan time was 0.5 seconds. We reconstructed each of the 10 sinograms using statistical reconstruction methods. The size of the reconstruction was $512 \times 512 \times 32$ voxels with voxel size $\Delta_x \times \Delta_z = 0.9764 \times 0.625\text{mm}$, as in the simulated phantom reconstructions. Each reconstruction used 100 iterations of an ordered-subset method [25] using 64 subsets. We performed the reconstructions using two different regularizers. In the first case, the regularization used a quadratic penalty and was spatially varying using the method of Section III-C. Figure 3(c) shows three slices of a sample reconstruction using these parameters. In the second case, the penalty function used the Huber potential (29) with a threshold δ of 10 Hounsfield units and was not spatially varying. Figure 4(c) shows three slices of a sample reconstruction using these parameters. In the quadratic-penalty case, the regularization parameter α was 2^{-14} ; in the Huber-penalty case, $\alpha = 2^{28}$. In both cases, the elements of the weighting matrix \mathbf{W} corresponded to the CT scanner's estimate of the inverse of the variance of each ray given the scanner-specific corrections used [29]. Given that we have several repeated scans of the same object, we computed the empirical variance of the observations \mathbf{y} from this data. Using this empirical observation variance for the purposes of evaluating variance prediction would be unrealistic, since in a clinical setting we do not have this data. When using elements of the matrix $\hat{\mathbf{W}}$ for variance prediction, we estimate the observation variance from the inverse of the scanner-provided weight.

Figures 3(a) (with spatially varying, quadratic regularization) and 4(a) (uniform, Huber-penalized regularization) show axial, sagittal, and coronal slices of the 3D map of the empirical standard deviation from our real reconstructions. As in the simulated empirical standard deviation maps, the empirical maps were noisy, so we smoothed the empirical variance maps with a 2D gaussian kernel with a FWHM of 5 voxels each in each direction. (Noisy, unsmoothed maps are shown in the supplemental material.) Figures 3(b) and 4(b) show the corresponding slices through the 3D predicted standard deviation map from (27). We computed the standard deviation once per $4 \times 4 \times 1$ block and used nearest-neighbor interpolation to fill in the rest. Figures 3(d) and 4(d) show the absolute magnitude of the error of our approximated standard deviation compared to the empirical results. Figures 3(e) and 4(e) show the empirical and predicted standard deviation along a one-dimensional coronal profile through the center of the image, along with the standard deviation as computed from (22) using a DFT variant of (17) as the LFR (labeled 'DFT-based').

Table I compares the computation time required to find the empirical variance, in both the simulation study and the real-world study, with the computation time required to

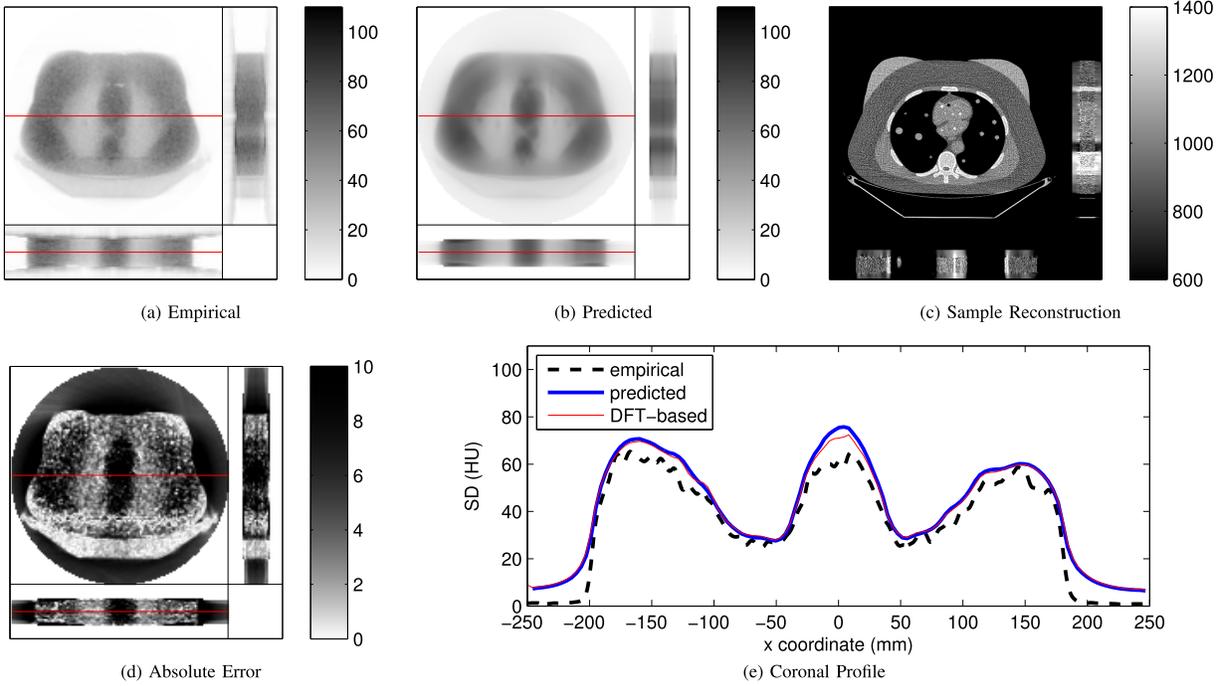


Fig. 3. Three slices of standard deviation maps for real reconstructions using spatially varying, quadratic regularization (Hounsfield units). Coronal and sagittal slices were stretched in the trans-axial direction by a factor of two for visualization. In (c), the scale is modified HU. In (e), the thin black line indicates empirical SD; the blue line indicates the DFT-based prediction; the thick black line indicates our predicted SD.

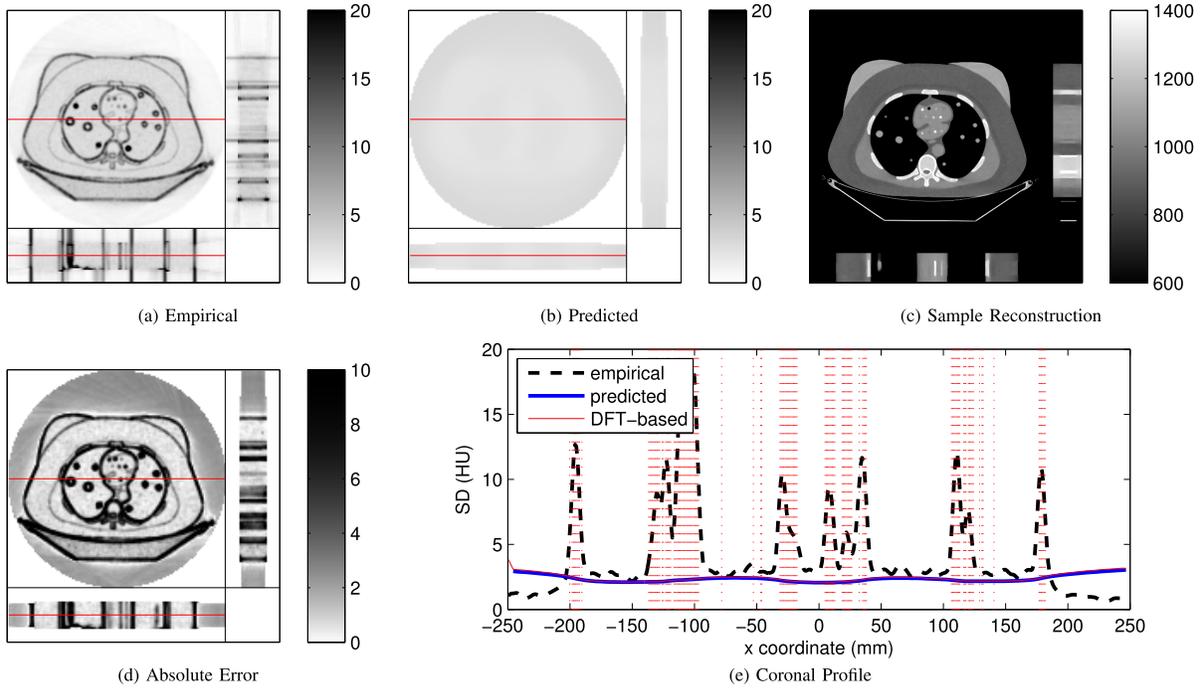


Fig. 4. Three slices of standard deviation maps for real reconstructions using spatially uniform, Huber-penalized regularization (Hounsfield units). Coronal and sagittal slices were stretched in the trans-axial direction by a factor of two for visualization. In (c), the scale is modified HU. In (e), the thin black line indicates empirical SD; the blue line indicates the DFT-based prediction; the thick black line indicates our predicted SD. Dashed red lines indicate locations of edges between voxels larger than 10 HU.

predict the variance for the entire volume using the DFT-based method and our methods. We used the DFT-based method only to produce the one-dimensional profiles shown

in Figures 1(e), 2(e), 3(e), and 4(e); since the computation time is large, we extrapolate to find the computation time for the entire volume for Table I.

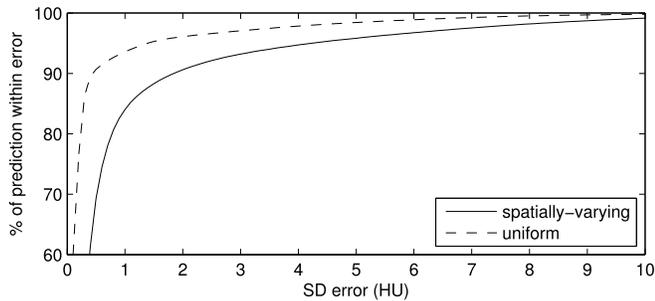


Fig. 5. Cumulative distribution of error of predicted standard deviation versus empirical standard deviation for simulated reconstructions in Figs. 1–Figs 2.

V. DISCUSSION

The presented methods are able to predict the standard deviation of most voxels in the simulated reconstructed images within an error of one Hounsfield unit in both the spatially varying regularization case (about 85% within 1HU) and the uniform regularization case (about 95% within 1HU) in less time than empirical measurement by a factor of over 10^4 . Figure 5 shows, for the entire CT volume, the percentage of the image that had an error within a specified bound in both the spatially varying and uniform regularization cases. Whether the tradeoff for time at the expense of accuracy provided by our method is acceptable depends on the application.

Figure 3 shows very good qualitative agreement between empirical and predicted standard deviation, even away from the plane of X-ray source rotation, until the furthest end slices of the reconstruction, which have insufficient data coverage and would not be presented clinically. Inside the phantom, the region with the highest error is the center-most section. Curiously, the DFT-based method has trouble with this region as well, although not as much as our prediction. The reason for this discrepancy is unknown. One possibility is that the OS algorithm had not converged completely. Figure 4 shows good qualitative agreement between the empirical and predicted standard deviation, except near edges. Except near the edges, the standard deviation is nearly constant in both the empirical and predicted reconstructions. The predicted standard deviation has a slight variation in the profile that we can see that seems like it follows a ground-truth slight variation that exists away from edges. The prediction slightly overestimates variance in Figure 3 and underestimates Figure 4; we are not sure what causes this discrepancy. It is possible that this is a function of the regularization parameter α (which is much higher in the Huber case, to compensate for the edge-preserving effect), and that for an α somewhere between these two cases the bias crosses zero. The comparisons in Figures 1(e), 2(e), 3(e), and 4(e) demonstrate that the majority of the error incurred by our methods occurs in the assumptions of quadratic-like regularization and local shift-invariance that ultimately lead to (22), rather than our approximations that transform (22) into the more computationally tractable (27). There is very little difference between the fast prediction and DFT-based prediction in the profile in Figure 4(e). This is reasonable considering that the error in the approximation (23) is highest when the magnitude of the

frequency, $||\vec{v}||$, is large, but in these regions the regularization tends to be strongest, suppressing the error in the resulting approximate local NPS.

Since we approximate the regularizer as being locally quadratic, it is not surprising that the main locations of error within the support of the object are near edges. This effect of edges on noise properties has also been seen for 2D fan-beam CT in [30], which postulates that the source of the high variance near edges is the uncertainty of edge position. Accurate variance prediction near edges might require *a priori* knowledge of edge location. We have investigated small cone angle geometries that allow us to approximate variance using the single-integral form (27). For a wider range of CT geometries, such as C-arm CT, where we cannot make a small cone angle approximation, we could instead use (24), although this double-integral form would afford less acceleration than the single-integral form.

In these geometries, more careful attention must also be paid to \mathcal{B} , defined in (50) (in supplementary material), which represents the set of source positions such that a ray from the source through a voxel of interest is perpendicular to a particular frequency. For our specific geometries, (69) is an approximate closed form for (50). For more general geometries, it may be necessary to find \mathcal{B} numerically or to find another approximation specific to the geometry.

VI. CONCLUSIONS AND FUTURE WORK

We presented a method for approximating the variance of each voxel of a statistically reconstructed 3DCT image with quadratic regularization. This method has a computational cost that is smaller by several orders of magnitude compared to existing variance prediction methods for helical CT, while maintaining a reasonable error within regions of interest away from image edges. Generalizing the methods for nonquadratic regularizers is a challenging open problem [31]. There are a few opportunities for further acceleration. When using a spatially varying regularizer, the change in variance as a function of space is often more due to the change in the effective regularization parameter than a change in weighting or location parameters. It may be possible, then, to evaluate the integral in (27) on a coarser grid and then interpolate to all image voxels, with the factor of α_{eff} multiplying the integral producing much of the spatial variation. We also have not performed a complete analysis of strategies to numerically evaluate the integral in (27), although pilot tests have suggested that as few as 24 values of Φ suffices.

We are also investigating using our approximate factorization (23) of the LFR in other applications that could benefit from the acceleration it can provide. For example, the performance of an algorithmic observer for detecting the presence of a feature can be approximately expressed in the frequency domain [32, Ch. V]. With a fast prediction for this performance, enabled by the factorization, we could potentially optimize automatic tube current modulation methods for a particular feature-detection task, e.g., for low-contrast features in locally smooth backgrounds where the quadratic regularizer approximation is reasonable.

REFERENCES

- [1] J.-B. Thibault, K. D. Sauer, C. A. Bouman, and J. Hsieh, "A three-dimensional statistical approach to improved image quality for multislice helical CT," *Med. Phys.*, vol. 34, pp. 4526–4544, Nov. 2007.
- [2] G. Wang, M. W. Vannier, and P. C. Cheng, "Iterative X-ray cone-beam tomography for metal artifact reduction and local region reconstruction," *Microsc. Microanal.*, vol. 5, pp. 58–65, Jan. 1999.
- [3] B. De. Man, J. Nuyts, P. Dupont, G. Marchal, and P. Suetens, "An iterative maximum-likelihood polychromatic algorithm for CT," *IEEE Trans. Med. Imag.*, vol. 20, no. 10, pp. 999–1008, Oct. 2001.
- [4] J. A. Fessler, "Mean and variance of implicitly defined biased estimators (such as penalized maximum likelihood): Applications to tomography," *IEEE Trans. Image Process.*, vol. 5, no. 3, pp. 493–506, Mar. 1996.
- [5] J. H. Cho and J. A. Fessler, "Quadratic regularization design for 3D axial CT," in *Proc. Int. Meeting Fully 3D Image Reconstruct. Radiol. Nucl. Med.*, 2013, pp. 78–81.
- [6] M. Gies, W. A. Kalender, H. Wolf, and C. Suess, "Dose reduction in CT by anatomically adapted tube current modulation. I. Simulation studies," *Med. Phys.*, vol. 26, pp. 2235–2247, Nov. 1999.
- [7] W. A. Kalender, H. Wolf, and C. Suess, "Dose reduction in CT by anatomically adapted tube current modulation. II. Phantom measurements," *Med. Phys.*, vol. 26, pp. 2248–2253, Nov. 1999.
- [8] J. A. Fessler, "Approximate variance images for penalized-likelihood image reconstruction," in *Proc. IEEE Nucl. Sci. Symp.*, Nov. 2. 1997, pp. 949–952, vol. 2.
- [9] J. Qi and R. M. Leahy, "A theoretical study of the contrast recovery and variance of MAP reconstructions from PET data," *IEEE Trans. Med. Imag.*, vol. 18, no. 4, pp. 293–305, Apr. 1999.
- [10] J. Qi and R. M. Leahy, "Resolution and noise properties of MAP reconstruction for fully 3-D PET," *IEEE Trans. Med. Imag.*, vol. 19, no. 5, pp. 493–506, May 2000.
- [11] Y. Zhang-O'Connor and J. A. Fessler, "Fast predictions of variance images for fan-beam transmission tomography with quadratic regularization," *IEEE Trans. Med. Imag.*, vol. 26, no. 3, pp. 335–346, Mar. 2007.
- [12] Y. Zhang-O'Connor and J. A. Fessler, "Fast variance predictions for 3D cone-beam CT with quadratic regularization," *Proc. SPIE*, vol. 6510, pp. 65105W-1–65105W-10, Mar. 2007.
- [13] S. M. Schmitt and J. A. Fessler, "Fast variance computation for quadratically penalized iterative reconstruction of 3D axial CT images," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf.*, Oct./Nov. 2012, pp. 3287–3292.
- [14] S. M. Schmitt and J. A. Fessler, "Fast variance prediction for iterative reconstruction of 3D helical CT images," in *Proc. 12th Int. Meeting Fully 3D Image Reconstruct. Radiol. Nucl. Med.*, 2013, pp. 162–165.
- [15] J. A. Fessler and W. L. Rogers, "Spatial resolution properties of penalized-likelihood image reconstruction: Space-invariant tomographs," *IEEE Trans. Image Process.*, vol. 5, no. 9, pp. 1346–1358, Sep. 1996.
- [16] J. A. Fessler, "Statistical image reconstruction methods for transmission tomography," in *Handbook of Medical Imaging: Medical Image Processing and Analysis*, M. Sonka and J. M. Fitzpatrick, Eds. Bellingham, WA, USA: SPIE, 2000, pp. 1–70.
- [17] K. Sauer and C. Bouman, "A local update strategy for iterative reconstruction from projections," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 534–548, Feb. 1993.
- [18] J. A. Fessler and S. D. Booth, "Conjugate-gradient preconditioning methods for shift-variant PET image reconstruction," *IEEE Trans. Image Process.*, vol. 8, no. 5, pp. 688–699, May 1999.
- [19] J. W. Stayman and J. A. Fessler, "Regularization for uniform spatial resolution properties in penalized-likelihood image reconstruction," *IEEE Trans. Med. Imag.*, vol. 19, no. 6, pp. 601–615, Jun. 2000.
- [20] S. Matej and R. M. Lewitt, "Practical considerations for 3-D image reconstruction using spherically symmetric volume elements," *IEEE Trans. Med. Imag.*, vol. 15, no. 1, pp. 68–78, Feb. 1996.
- [21] J. A. C. Weideman, "Numerical integration of periodic functions: A few examples," *Am. Math. Monthly*, vol. 109, no. 1, pp. 21–36, 2002.
- [22] J. H. Cho and J. A. Fessler, "Regularization designs for uniform spatial resolution and noise properties in statistical image reconstruction for 3-D X-ray CT," *IEEE Trans. Med. Imag.*, vol. 34, no. 2, pp. 678–689, Feb. 2015.
- [23] J. Gregor, "Data-driven problem reduction for image reconstruction from projections using gift wrapping," *IEEE Trans. Nucl. Sci.*, vol. 58, no. 3, pp. 724–729, Jun. 2011.
- [24] W. P. Segars, M. Mahesh, T. J. Beck, E. C. Frey, and B. M. W. Tsui, "Realistic CT simulation using the 4D XCAT phantom," *Med. Phys.*, vol. 35, pp. 3800–3808, Aug. 2008.
- [25] H. Erdogan and J. A. Fessler, "Ordered subsets algorithms for transmission tomography," *Phys. Med. Biol.*, vol. 44, pp. 2835–2851, Nov. 1999.
- [26] D. Kim, D. Pal, J.-B. Thibault, and J. A. Fessler, "Improved ordered subsets algorithm for 3D X-ray CT image reconstruction," in *Proc. 2nd Int. Meet. Image Formation X-Ray CT*, 2012, pp. 378–381.
- [27] E. A. Zerhouni *et al.*, "A standard phantom for quantitative CT analysis of pulmonary nodules," *Radiology*, vol. 149, no. 3, pp. 767–773, 1983.
- [28] M. M. Goodsitt *et al.*, "Accuracy of the CT numbers of simulated lung nodules imaged with multi-detector CT scanners," *Med. Phys.*, vol. 33, pp. 3006–3017, Aug. 2006.
- [29] Z. Chang, R. Zhang, J.-B. Thibault, K. Sauer, and C. Bouman, "Statistical X-ray computed tomography from photon-starved measurements," *Proc. SPIE*, vol. 9020, p. 90200G, Mar. 2014.
- [30] T. Koehler and R. Proksa, "Noise properties of maximum likelihood reconstruction with edge-preserving regularization in transmission tomography," in *Proc. Int. Meet. Fully 3D Image Reconstruct. Radiol. Nucl. Med.*, 2009, pp. 263–266.
- [31] S. Ahn and R. M. Leahy, "Analysis of resolution and noise properties of nonquadratically regularized image reconstruction methods for PET," *IEEE Trans. Med. Imag.*, vol. 27, no. 3, pp. 413–424, Mar. 2008.
- [32] S. Schmitt, "Fast variance prediction for iteratively reconstructed CT with applications to tube current modulation," Ph.D. dissertation, Univ. Michigan, Ann Arbor, MI, USA, 2015.

Fast Variance Prediction for Iteratively Reconstructed CT Images with Locally Quadratic Regularization: Supplementary Material

Stephen M. Schmitt, *Student Member, IEEE*, Mitchell M. Goodsitt, and Jeffrey A. Fessler, *Fellow, IEEE*

This supplementary material for [1] provides further details (in sections VII-A through VII-E) of the derivation of our factored approximation for the local frequency response of CT projection, weighting, and back-projection. Section VII-F derives further approximations for CT geometries with small cone angles.

VII. APPENDIX: LFR DERIVATION

A. The ray and projection transforms

We begin by examining one element of \mathbf{h}_j^W , the LIR of $\mathbf{A}^T \mathbf{W} \mathbf{A}$ applied to voxel j . Writing the matrix multiplication that defines this element as a sum over views i gives:

$$[\mathbf{h}_j^W]_k = [\mathbf{A}^T \mathbf{W} \mathbf{A}]_{kj} = \sum_i [\mathbf{A}]_{ik} [\mathbf{W}]_{ii} [\mathbf{A}]_{ij}. \quad (30)$$

To specify the projection matrix \mathbf{A} , we first use the definition of a ray transform operator \mathcal{R} that transforms a function f into line integrals through it:

$$(\mathcal{R}f)(\vec{x}, \hat{\theta}) \triangleq \int_{\mathbb{R}} f(\vec{x} + \tau \hat{\theta}) d\tau, \quad (31)$$

where \vec{x} is a point on a ray and $\hat{\theta}$ is a unit vector representing a ray direction. We define our projection matrix as a sampling of a projection operator \mathcal{P} that is simply a rebinning of \mathcal{R} :

$$(\mathcal{P}f)(\vec{s}, \sigma) \triangleq (\mathcal{R}f)(\vec{x}(\vec{s}, \sigma), \hat{\theta}(\vec{s}, \sigma)); \quad (32)$$

this continuous-domain projection $\mathcal{P}f$ is defined over the $(n-1)$ -dimensional detector position \vec{s} and 1-dimensional source position σ . The function $\vec{x}(\vec{s}, \sigma)$ maps a source position and detector position pair to a point on the ray that connects the pair. The function $\hat{\theta}(\vec{s}, \sigma)$ maps these positions to a unit vector that lies along the ray direction. We will also use $\vec{s}(\vec{x}, \sigma)$ to represent the detector coordinate corresponding to the ray that passes through the source at position σ and the spatial position

\vec{x} . These functions are defined by the geometry of the CT system under consideration, and are assumed to be known.

With this definition for \mathcal{P} , we approximate the elements of \mathbf{A} as discretized samples of the continuous projection:

$$[\mathbf{A}]_{ik} \approx (\mathcal{P}R_k)(\vec{s}_i, \sigma_i), \quad (33)$$

that is, the i th observation of a Kronecker impulse at voxel k is approximately the projection of the k th basis voxel, $R_k(\vec{x})$, sampled at one detector position \vec{s}_i with the source at one position σ_i . The approximation (33) for one element of the projection matrix differs from the \mathbf{A} matrix typically used in implementing the reconstruction (1) in two ways. First, the approximation treats the projection as being measured at exactly one point, neglecting detector blur. Second, (33) is defined to be an *exact* projection at this particular location, whereas an implemented system matrix \mathbf{A} (e.g. [2]) will use approximations to the integral through a basis function.

In the same way that we replace \mathbf{A} with a samples of a continuous function, we define $w(\vec{s}, \sigma)$ to denote a continuous (i.e., interpolated) version of the elements of the weighting matrix \mathbf{W} . The actual method used for interpolation is relatively unimportant, but we assume that w is a function such that $w(\vec{s}_i, \sigma_i) = [\mathbf{W}]_{ii}$. With these continuous analogs for \mathbf{A} and \mathbf{W} , we rewrite the LIR sum (30) as:

$$[\mathbf{h}_j^W]_k = \sum_i (\mathcal{P}R_j)(\vec{s}_i, \sigma_i) (\mathcal{P}R_k)(\vec{s}_i, \sigma_i) w(\vec{s}_i, \sigma_i). \quad (34)$$

For typical clinical CT scans (but not for sparse view acquisitions), this sum is usually sufficiently finely sampled that we can approximate it with an integral over \vec{s} and σ :

$$[\mathbf{h}_j^W]_k \approx \int_{\Sigma} \int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_k)(\vec{s}, \sigma) u(\vec{s}, \sigma) d\vec{s} d\sigma. \quad (35)$$

Here, Σ represents a continuous approximation to the domain of σ for which we collect observations; for all the system geometries we have examined this is a single interval $[\sigma_{\min}, \sigma_{\max}]$, but for applications such as gated CT this may be the union of multiple intervals. The term u in (35) is the product of w with a Jacobian term representing the “size” of a sample in the sum (34). In the usual case where the detector pixels have a constant area $\Delta_{\vec{s}}$ and the spacing (in radians)

This work was supported in part by NIH grants R01 HL-098686 and U01 EB018753 and equipment donations from Intel Corporation.

Stephen M. Schmitt and Jeffrey A. Fessler are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (smschm@umich.edu, fessler@umich.edu)

Mitchell M. Goodsitt is with the Department of Radiology, University of Michigan, Ann Arbor, MI 48109 USA (goodsitt@med.umich.edu)

Copyright (c) 2016 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

between source positions is a constant Δ_σ , then

$$u(\vec{s}, \sigma) \triangleq \frac{w(\vec{s}, \sigma)}{\Delta_{\vec{s}} \Delta_\sigma}. \quad (36)$$

However, for geometries where the spacing is nonuniform, u will not simply be a scaling of w . The approximation (35) also includes expanding the domain of detector positions \vec{s} to \mathbb{R}^{n-1} ; to compensate for this we consider the weighting $w(\vec{s}, \sigma)$, and hence $u(\vec{s}, \sigma)$, to be zero in the regions where we have no observations.

B. Linearization of projection transform

If we fix a spatial position \vec{x}_j and source position σ , this also fixes a detector location $\vec{s}_{j,\sigma} \triangleq \vec{s}(\vec{x}_j, \sigma)$ and ray direction $\hat{\theta}_{j,\sigma} \triangleq \hat{\theta}(\vec{s}_{j,\sigma}, \sigma)$, such that $(\mathcal{R}f)(\vec{x}_j, \hat{\theta}_{j,\sigma}) = (\mathcal{P}f)(\vec{s}_{j,\sigma}, \sigma)$. The coordinate mappings between \mathcal{P} and \mathcal{R} are usually regular enough that a first-order Taylor expansion is quite accurate for small perturbations $\Delta\vec{x}$:

$$(\mathcal{R}f)(\vec{x}_j + \Delta\vec{x}, \hat{\theta}_{j,\sigma}) \approx (\mathcal{P}f)(\vec{s}_{j,\sigma} + \mathbf{B}_{j,\sigma} \Delta\vec{x}, \sigma), \quad (37)$$

where

$$\mathbf{B}_{j,\sigma} \triangleq \nabla_{\vec{x}} \vec{s}(\vec{x}, \sigma^*)|_{\vec{x}=\vec{x}_j, \sigma^*=\sigma} \quad (38)$$

is a $(n-1) \times n$ matrix that ‘‘linearizes’’ the system geometry, locally to \vec{x}_j and σ , to a parallel-beam, flat-panel geometry. (Adding further Taylor expansion terms to this linearization makes further derivation considerably more complicated, and for the geometries we have considered, just one term has adequate accuracy.) We can also reverse (37) to find a spatial shift that corresponds to a particular detector position shift:

$$\begin{aligned} (\mathcal{R}f)(\vec{x}_j + \mathbf{B}_{j,\sigma}^+ \Delta\vec{s}, \hat{\theta}_{j,\sigma}) &\approx (\mathcal{P}f)(\vec{s}_{j,\sigma} + \mathbf{B}_{j,\sigma} \mathbf{B}_{j,\sigma}^+ \Delta\vec{s}, \sigma) \\ &= (\mathcal{P}f)(\vec{s}_{j,\sigma} + \Delta\vec{s}, \sigma), \end{aligned} \quad (39)$$

where $\mathbf{B}_{j,\sigma}^+$ is the $n \times (n-1)$ pseudo-inverse of $\mathbf{B}_{j,\sigma}$.

We will also need the QR factorization of $\mathbf{B}_{j,\sigma}$:

$$\mathbf{B}_{j,\sigma} = \mathbf{R}_{j,\sigma} \mathbf{Q}_{j,\sigma}, \quad (40)$$

where $\mathbf{R}_{j,\sigma}$ is a $(n-1) \times (n-1)$ matrix and $\mathbf{Q}_{j,\sigma}$ is a $(n-1) \times n$ matrix with orthonormal rows; each of these rows is also orthogonal to $\hat{\theta}_{j,\sigma}$.

C. Towards local shift invariance

We define a footprint correlation function as:

$$c_{kj,\sigma} \triangleq \int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_k)(\vec{s}, \sigma) d\vec{s} \quad (41)$$

i.e., the integral over the detector plane of the product of the continuous projections of the voxel basis functions R_j and R_k for a specific source position σ , and related weighting factors:

$$\check{u}_{kj,\sigma} \triangleq \frac{\int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_k)(\vec{s}, \sigma) u(\vec{s}, \sigma) d\vec{s}}{\int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_k)(\vec{s}, \sigma) d\vec{s}}, \quad (42)$$

so that the approximation to the LIR in (35) becomes

$$[\mathbf{h}_j^W]_k \approx \int_{\Sigma} c_{kj,\sigma} \check{u}_{kj,\sigma} d\sigma. \quad (43)$$

Next we simplify each term in the integrand using approximations that remove most of their dependence on k .

1) *Simplifying $c_{jk,\sigma}$* : We make the usual assumption that the basis functions for voxels k and j have the same shape, and differ only by translation, i.e.,

$$R_k(\vec{x}) = R_j(\vec{x} - (\vec{x}_k - \vec{x}_j)).$$

Given this spatial relationship, their ray transforms are related by translation:

$$\begin{aligned} (\mathcal{R}R_k)(\vec{x}, \hat{\theta}) &= \int R_k(\vec{x} + \tau\hat{\theta}) d\tau \\ &= \int R_j(\vec{x} + \tau\hat{\theta} - (\vec{x}_k - \vec{x}_j)) d\tau \\ &= (\mathcal{R}R_j)(\vec{x} - (\vec{x}_k - \vec{x}_j), \hat{\theta}). \end{aligned} \quad (44)$$

Using the Taylor expansion in (37), the projection footprint of nearby voxels are approximately related by the following translation:

$$(\mathcal{P}R_k)(\vec{s}, \sigma) \approx (\mathcal{P}R_j)(\vec{s} - \mathbf{B}_{j,\sigma}(\vec{x}_k - \vec{x}_j), \sigma); \quad (45)$$

that is, for a fixed σ , the projection of voxel k can be approximated as a shift of the projection of voxel j , so long as \vec{x}_k is sufficiently close to \vec{x}_j . Using (45), we rewrite the footprint correlation $c_{kj,\sigma}$ as the autocorrelation of just the projection of the j th voxel:

$$\begin{aligned} c_{kj,\sigma} &\triangleq \int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_k)(\vec{s}, \sigma) d\vec{s} \\ &\approx \int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}, \sigma) (\mathcal{P}R_j)(\vec{s} - \mathbf{B}_{j,\sigma}(\vec{x}_k - \vec{x}_j), \sigma) d\vec{s} \\ &= (\mathcal{A}\mathcal{P}R_j)(\mathbf{B}_{j,\sigma}(\vec{x}_k - \vec{x}_j), \sigma) \\ &\triangleq \check{c}_{kj,\sigma}, \end{aligned} \quad (46)$$

$$\triangleq \check{c}_{kj,\sigma}, \quad (47)$$

where \mathcal{A} denotes an autocorrelation operator:

$$(\mathcal{A}f)(\vec{x}) \triangleq \int f(\vec{t}) f(\vec{x} + \vec{t}) d\vec{t}.$$

That is, this correlation between the projections of voxels j and k is approximately a function of the autocorrelation of the projection of just voxel j , and the dependence on k is only via its location relative to j .

2) *Simplifying $\check{u}_{kj,\sigma}$* : For a given source position σ , the expression for \check{u} in (42) is the integral over the detector of the product of the continuous projections of R_j and R_k with the weighting function u , normalized by the integral of just the projections of R_j and R_k . The projections of each of R_j and R_k have a small support, and so does their product; we can therefore approximate the effect of (42) as a sifting that selects one value of u much like a Dirac impulse. We assume that u in (42) varies slowly with respect to \vec{s} , which is often true for typical choices of \mathbf{W} . The peak of $\mathcal{P}R_j$ is located near $\vec{s}(\vec{x}_j, \sigma)$, the detector location that corresponds to a ray originating at the detector at position σ and passing through the center of the j th voxel location, \vec{x}_j . For any other voxel k , if the integrand of (34) is non-zero, the peak of the projection of k must overlap the projection of j , meaning it is close enough to the peak of the projection of k that we can make the approximation:

$$\check{u}_{kj,\sigma} \approx \tilde{u}_{j,\sigma} \triangleq u(\vec{s}(\vec{x}_j, \sigma), \sigma). \quad (48)$$

We compute \tilde{u} by looking up the value of w for the detector element closest to $\vec{s}(\vec{x}_j, \sigma)$ and using (36). With approximations (46) and (48), the LIR in (43) becomes:

$$[\mathbf{h}_j^W]_k \approx \int \tilde{u}_{j,\sigma} (\mathcal{A}PR_j)(\mathbf{B}_{j,\sigma}(\vec{x}_k - \vec{x}_j), \sigma) d\sigma. \quad (49)$$

This is our final ‘‘space domain’’ approximation to the original LIR expression in (30). In short, this LIR expression uses the autocorrelation of the footprints of the voxel evaluated at projection view coordinates that account for the magnification through the j th voxel and integrated over all X-ray source positions. Next we move to the frequency domain to find the corresponding local frequency response.

D. Local frequency response (LFR) of $\mathbf{A}^T \mathbf{W} \mathbf{A}$

We use the form (49) for the LIR $[\mathbf{h}_j^W]_k$ to find a corresponding LFR, in (51) through (55) (at the end of this section), using (17). In (51), we take (17) and insert the approximation (49) for \mathbf{h}_j^W . In (52), we interchange the sum over voxels k and the integral over source position σ , and move the $\tilde{u}_{j,\sigma}$ term out of the sum, as it does not depend on k . In (53), we approximate the sum over k with an integral. This replacement assumes that, first, the summand term is slowly varying enough that we can make an integral approximation, and second, that the summand decays rapidly enough beyond a region of interest that we can replace it with an integral not just over the image support, but over \mathbb{R}^n . This integral is the continuous Fourier transform of the footprint $(\mathcal{A}PR_j)(\mathbf{B}_{j,\sigma} \mathbf{V} \vec{n})$ over \vec{n} in \mathbb{R}^n .

In (54), we replace this inner integral in (53). The integral represents the continuous n -dimensional Fourier transform of an $(n-1)$ -dimensional function, so we must be careful. We use the derivation found in section VII-G in the Appendix, where (73) is the inner integral in (53). Here, $\mathbf{X} = \mathbf{B}_{j,\sigma} \mathbf{V}$, $f = \mathcal{A}PR_j$, and $\vec{x} = \hat{\theta}_{j,\sigma}$. The resulting replacement in (54) corresponds to the expression in (74). To complete the transition to (54), we note that

$$\det \left(\begin{bmatrix} \mathbf{B}_{j,\sigma} \\ \hat{\theta}_{j,\sigma}^T \end{bmatrix} \right) = \det(\mathbf{R}_{j,\sigma}),$$

using the $\mathbf{R}_{j,\sigma}$ from the QR factorization (40).

Finally, in (55), we use the Dirac impulse in (54) to sift out only a finite number of σ values in the integral such that $\hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{v} = 0$. We define this set to be $\mathcal{B}_j \left(\frac{\vec{v}}{\|\vec{v}\|} \right)$, as follows:

$$\mathcal{B}_j \left(\frac{\vec{v}}{\|\vec{v}\|} \right) \triangleq \left\{ \sigma : \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{v} = 0 \right\}. \quad (50)$$

Note that \mathcal{B}_j depends on spatial frequency variable \vec{v} only via its direction, $\vec{v}/\|\vec{v}\|$, and not its magnitude.

$$\begin{aligned} H_j^W(\vec{v}) &= \mathcal{D} \left\{ \exp(j2\pi \vec{v}^T \vec{n}_j) \right\} \mathcal{F} \mathbf{h}_j^W \\ &= \exp(j2\pi \vec{v}^T \vec{n}_j) \sum_k [\mathbf{h}_j^W]_k \exp(-j2\pi \vec{v}^T \vec{n}_k) \\ &\approx \sum_k \left(\int \tilde{u}_{j,\sigma} (\mathcal{A}PR_j)(\mathbf{B}_{j,\sigma}(\vec{x}_k - \vec{x}_j), \sigma) d\sigma \right) \\ &\quad \exp(-j2\pi \vec{v}^T (\vec{n}_k - \vec{n}_j)) \end{aligned} \quad (51)$$

$$= \int \tilde{u}_{j,\sigma} \left[\sum_k (\mathcal{A}PR_j)(\mathbf{B}_{j,\sigma} \mathbf{V} (\vec{n}_k - \vec{n}_j)) \exp(-j2\pi \vec{v}^T (\vec{n}_k - \vec{n}_j)) \right] d\sigma \quad (52)$$

$$\approx \int \tilde{u}_{j,\sigma} \left[\int (\mathcal{A}PR_j)(\mathbf{B}_{j,\sigma} \mathbf{V} (\vec{n} - \vec{n}_j)) \exp(-j2\pi \vec{v}^T (\vec{n} - \vec{n}_j)) d\vec{n} \right] d\sigma \quad (53)$$

$$= \int \frac{\tilde{u}_{j,\sigma}}{\det(\mathbf{V}) \det(\mathbf{R}_{j,\sigma})} (\mathcal{F}^{n-1} \mathcal{A}PR_j)(\mathbf{B}_{j,\sigma}^{+T} \mathbf{V}^{-T} \vec{v}) \delta(\hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{v}) d\sigma \quad (54)$$

$$= \sum_{\sigma \in \mathcal{B}_j \left(\frac{\vec{v}}{\|\vec{v}\|} \right)} \frac{\tilde{u}_{j,\sigma} (\mathcal{F}^{n-1} \mathcal{A}PR_j)(\mathbf{B}_{j,\sigma}^{+T} \mathbf{V}^{-T} \vec{v})}{\det(\mathbf{V}) \det(\mathbf{R}_{j,\sigma})} \left(\frac{\partial}{\partial \sigma} \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{v} \right) \quad (55)$$

E. Final LFR Approximation

We further simplify the LFR (55) by approximating $\mathcal{F}^{n-1} \mathcal{A}PR_j$. We use the Fourier identity for autocorrelation:

$$(\mathcal{F}^n \mathcal{A}x)(\vec{v}) = |(\mathcal{F}^n x)(\vec{v})|^2,$$

and a Taylor-expansion-based approximation for $\mathcal{F}^{n-1} \mathcal{P}R_j$:

$$\frac{(\mathcal{F}^{n-1} \mathcal{P}R_j)(\vec{u}; \sigma)}{\exp(j2\pi \vec{u}^T \vec{s}_{j,\sigma})}$$

$$= \int_{\mathbb{R}^{n-1}} (\mathcal{P}R_j)(\vec{s}_{j,\sigma} + \Delta \vec{s}; \sigma) \exp(-j2\pi \vec{u}^T \Delta \vec{s}) d\Delta \vec{s} \quad (56)$$

$$\approx \int_{\mathbb{R}^{n-1}} (\mathcal{R}R_j)(\vec{x}_j + \mathbf{B}_{j,\sigma}^+ \Delta \vec{s}, \hat{\theta}_{j,\sigma}) \exp(-j2\pi \vec{u}^T \Delta \vec{s}) d\Delta \vec{s}$$

$$= \int_{\mathbb{R}^{n-1}} \int_{\mathbb{R}} R_j(\vec{x}_j + \mathbf{B}_{j,\sigma}^+ \Delta \vec{s} + \tau \hat{\theta}_{j,\sigma}) \exp(-j2\pi \vec{u}^T \Delta \vec{s}) d\tau d\Delta \vec{s} \quad (57)$$

$$= \int_{\mathbb{R}^n} R_j(\vec{x}_j + \tilde{\mathbf{B}}_{j,\sigma}^{-1} \vec{w}) \exp(-j2\pi \vec{u}^T \mathbf{S}_{n-1} \vec{w}) d\vec{w} \quad (58)$$

In (57), in the argument to R_j , $\hat{\theta}_{j,\sigma}$ is perpendicular to each column of $\mathbf{B}_{j,\sigma}^+$, and so together $\mathbf{B}_{j,\sigma}^+ \Delta \vec{s} + \tau \hat{\theta}_{j,\sigma}$ spans all of \mathbb{R}^n ; by defining

$$\vec{w} \triangleq \begin{bmatrix} -\Delta \vec{s} \\ \tau \end{bmatrix}, \tilde{\mathbf{B}}_{j,\sigma} \triangleq \begin{bmatrix} \mathbf{B}_{j,\sigma}^+ \\ \hat{\theta}_{j,\sigma} \end{bmatrix},$$

and \mathbf{S}_{n-1} to be the first $n-1$ rows of the $n \times n$ identity matrix, we can combine everything into an n -dimensional integral (58) over \mathbb{R}^n . By defining $\vec{z} \triangleq \vec{x}_j + \tilde{\mathbf{B}}_{j,\sigma}^{-1} \vec{w}$ and changing

the variable of integration in (58) to \vec{z} , we find that:

$$\begin{aligned} & \int_{\mathbb{R}^n} R_j(\vec{x}_j + \tilde{\mathbf{B}}_{j,\sigma}^{-1}\vec{w}) \exp(-j2\pi\vec{u}^T \mathbf{S}_{n-1}\vec{w}) \\ &= \det(\tilde{\mathbf{B}}_{j,\sigma}) \exp(j2\pi\vec{u}^T \mathbf{B}_{j,\sigma}\vec{x}_j) \cdot \\ & \int R_j(\vec{z}) \exp(-j2\pi\vec{u}^T \underbrace{\mathbf{S}_{n-1}\tilde{\mathbf{B}}_{j,\sigma}}_{=\mathbf{B}_{j,\sigma}} \vec{z}) d\vec{z} \\ &= \det(\tilde{\mathbf{B}}_{j,\sigma}) \exp(j2\pi\vec{u}^T \mathbf{B}_{j,\sigma}\vec{x}_j) (\mathcal{F}^n R_j)(\mathbf{B}_{j,\sigma}^T \vec{u}). \end{aligned} \quad (59)$$

By taking the absolute value of our approximation (59):

$$|(\mathcal{F}^{n-1} \mathcal{P} R_j)(\vec{u}; \sigma)|^2 \approx \det(\mathbf{R}_{j,\sigma})^2 |(\mathcal{F}^n R_j)(\mathbf{B}_{j,\sigma}^T \vec{u})|^2, \quad (60)$$

using the fact that $\det(\tilde{\mathbf{B}}_{j,\sigma}) = \det(\mathbf{R}_{j,\sigma})$.

The approximation (60) simplifies (55) to the following LFR expression:

$$\begin{aligned} H_j^W(\vec{\nu}) &\approx \frac{|(\mathcal{F}^n R_j)(\mathbf{V}^{-T}\vec{\nu})|^2}{\det(\mathbf{V})} \sum_{\sigma \in \mathcal{B}_j(\frac{\vec{\nu}}{\|\vec{\nu}\|})} \frac{\tilde{u}_{j,\sigma} \det(\mathbf{R}_{j,\sigma})}{\left(\frac{\partial}{\partial \sigma} \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{\nu}\right)} \\ &= \frac{|(\mathcal{F}^n R_j)(\mathbf{V}^{-T}\vec{\nu})|^2}{\det(\mathbf{V}) \|\vec{\nu}\|} \sum_{\sigma \in \mathcal{B}_j(\frac{\vec{\nu}}{\|\vec{\nu}\|})} \frac{\tilde{u}_{j,\sigma} \det(\mathbf{R}_{j,\sigma})}{\left(\frac{\partial}{\partial \sigma} \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \frac{\vec{\nu}}{\|\vec{\nu}\|}\right)} \\ &= J(\vec{\nu}) E_j^W\left(\frac{\vec{\nu}}{\|\vec{\nu}\|}\right), \end{aligned} \quad (61)$$

where we define the terms:

$$J(\vec{\nu}) \triangleq \frac{|(\mathcal{F}^n R_j)(\mathbf{V}^{-T}\vec{\nu})|^2}{\det(\mathbf{V})^2 \|\vec{\nu}\|} \quad (62)$$

$$E_j^W\left(\frac{\vec{\nu}}{\|\vec{\nu}\|}\right) \triangleq \sum_{\sigma \in \mathcal{B}_j(\frac{\vec{\nu}}{\|\vec{\nu}\|})} \frac{\tilde{u}_{j,\sigma} \det(\mathbf{R}_{j,\sigma}) \det(\mathbf{V})}{\left(\frac{\partial}{\partial \sigma} \hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \frac{\vec{\nu}}{\|\vec{\nu}\|}\right)}. \quad (63)$$

We simplify $J(\vec{\nu})$ by assuming that each R_j is equal to a basis unit voxel R_{basis} (for example, a unit cube) scaled by the voxel spacing \mathbf{V} and shifted by \vec{x}_j , that is,

$$R_j(\vec{x}) = R_{\text{basis}}(\mathbf{V}^{-1}(\vec{x} - \vec{x}_j)).$$

In this case, J becomes:

$$J(\vec{\nu}) = \frac{|(\mathcal{F}^n R_{\text{basis}})(\vec{\nu})|^2}{\|\vec{\nu}\|}. \quad (64)$$

In particular, for the usual choice of a unit cube basis, (64) further simplifies to the following final expression:

$$J(\vec{\nu}) = \frac{\text{sinc}^2(\nu_1) \text{sinc}^2(\nu_2) \text{sinc}^2(\nu_3)}{\|\vec{\nu}\|}. \quad (65)$$

Importantly, in the factorization of (61), J is independent of voxel j , the weighting (via u), or even the specific CT geometry used. The E_j^W term depends on all of these factors, but depends only on the frequency via its direction, not its magnitude. This factorization is key to computational efficiency. To use (63), one must consider the CT geometry of interest, as shown in Section III-B.

F. Application to 3DCT with small cone angle

In this section, we derive a further approximation to (61) specific to 3DCT geometries with small cone angles.

In a third-generation helical CT system, the source position \vec{p}_s as a function of the source angle σ is given by

$$\vec{p}_s(\sigma) = \begin{pmatrix} -D_{\text{so}} \cos \sigma \\ -D_{\text{so}} \sin \sigma \\ p_3 \sigma \end{pmatrix},$$

where D_{so} is the source-isocenter distance, D_{sd} is the source-detector distance, and p_3 is the helical pitch, in units of length per radian; axial CT is a special case where $p_3 = 0$. We denote the position of the center of voxel j in Cartesian and cylindrical coordinates as

$$\vec{x}_j = \begin{pmatrix} x_{j,1} \\ x_{j,2} \\ x_{j,3} \end{pmatrix} = \begin{pmatrix} r_j \cos \phi_j \\ r_j \sin \phi_j \\ x_{j,3} \end{pmatrix}.$$

The ray from the source at σ to the position \vec{x}_j is denoted

$$\vec{\ell}_{j,\sigma} = \vec{x}_j - \vec{p}_s(\sigma). \quad (66)$$

The first coordinate of the detector position $s_1(\vec{x}_j; \sigma)$ is given by $s_1(\vec{x}_j; \sigma) = D_{\text{sd}} \text{atan2}(\ell_{2,j,\sigma}, \ell_{1,j,\sigma}) - \sigma$. The second coordinate is given by

$$s_2(\vec{x}_j; \sigma) = \frac{D_{\text{sd}}}{D_{2,j,\sigma}} \ell_{3,j,\sigma},$$

where

$$D_{2,j,\sigma} \triangleq \|\text{diag}(1, 1, 0) \vec{\ell}_{j,\sigma}\| \quad (67)$$

denotes the length of the projection of $\vec{\ell}_{j,\sigma}$ into the xy -plane. From these coordinates, we find the matrix $\mathbf{B}_{j,\sigma}$, defined in (38), for this geometry:

$$\mathbf{B}_{j,\sigma} = D_{\text{sd}} \begin{bmatrix} -\frac{\ell_{2,j,\sigma}}{D_{2,j,\sigma}^2} & \frac{\ell_{1,j,\sigma}}{D_{2,j,\sigma}^2} & 0 \\ -\frac{\ell_{1,j,\sigma}\ell_{3,j,\sigma}}{D_{2,j,\sigma}^3} & -\frac{\ell_{2,j,\sigma}\ell_{3,j,\sigma}}{D_{2,j,\sigma}^3} & \frac{1}{D_{2,j,\sigma}} \end{bmatrix},$$

which has a QR-type factorization

$$\begin{aligned} \mathbf{B}_{j,\sigma} &= \mathbf{R}_{j,\sigma} \mathbf{Q}_{j,\sigma} \\ \mathbf{R}_{j,\sigma} &= \begin{bmatrix} D_{\text{sd}}/D_{2,j,\sigma} & 0 \\ 0 & D_{\text{sd}}\|\vec{\ell}_{j,\sigma}\|/D_{2,j,\sigma}^2 \end{bmatrix} \\ \mathbf{Q}_{j,\sigma}^T &= \begin{bmatrix} -\ell_{2,\sigma}/D_{2,j,\sigma} & -\ell_{1,\sigma}\ell_{3,\sigma}/\|\vec{\ell}_{j,\sigma}\|D_{2,j,\sigma} \\ \ell_{1,\sigma}/D_{2,j,\sigma} & -\ell_{2,\sigma}\ell_{3,\sigma}/\|\vec{\ell}_{j,\sigma}\|D_{2,j,\sigma} \\ 0 & D_{2,j,\sigma}/\|\vec{\ell}_{j,\sigma}\| \end{bmatrix} \end{aligned}$$

For 3D CT, we use (Φ, Θ) to denote the angular coordinates of 3D spherical spatial frequencies, such that

$$\frac{\vec{\nu}}{\|\vec{\nu}\|} = \vec{\Theta} = \begin{pmatrix} \cos \Theta \cos \Phi \\ \cos \Theta \sin \Phi \\ \sin \Theta \end{pmatrix}.$$

If the voxel spacing is equal to the common choice of $\mathbf{V} = \text{diag}(\Delta_x, \Delta_x, \Delta_z)$, the argument to the Dirac impulse in (54)

is equal to

$$\hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{\Theta} = \frac{1}{\|\vec{\ell}_{j,\sigma}\|} \left(\frac{\cos \Theta}{\Delta_x} (r \cos(\Phi - \phi) + D_{\text{so}} \cos(\Phi - \sigma)) + \frac{\sin \Theta}{\Delta_z} (x_3 - p_3 \sigma) \right). \quad (68)$$

For typical clinical 3DCT system geometries, the majority of the noise power in LFR approximations to $S_j(\vec{\nu})$ is located in regions of spatial frequency with relatively small Θ . Since we are particularly interested in the region where Θ is small, and since $(x_3 - p_3 \sigma)$ is often small, we make the following approximation to (68):

$$\hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{\Theta} \approx \frac{\cos \Theta}{\Delta_x \|\vec{\ell}_{j,\sigma}\|} (r_j \cos(\Phi - \phi_j) + D_{\text{so}} \cos(\Phi - \sigma)),$$

so that $\mathcal{B}_j(\vec{\Theta})$, defined in (50) as the set of σ for which $\hat{\theta}_{j,\sigma}^T \mathbf{V}^{-T} \vec{\Theta} = 0$, can be approximated as

$$\mathcal{B}_j(\vec{\Theta}) \approx \Phi \pm \cos^{-1}(-r_j(\Phi - \phi_j)/D_{\text{so}}). \quad (69)$$

This approximation is useful because (68) has no closed-form solution for $\mathcal{B}_j(\vec{\Theta})$. It also allows us to approximate $E_j^W(\Phi, \Theta) \cos \Theta \approx E_j^W(\Phi, 0)$. This facilitates using a 1D integral instead of a 2D integral for variance prediction. Defining:

$$J_{\text{cyl}}(\vec{\nu}) \triangleq J(\vec{\nu}) \sec \Theta \quad (70)$$

$$E_{j,\text{cyl}}^W(\Phi) \triangleq E^W(\Phi, 0) \approx E_j^W(\Phi, \Theta) \cos \Theta,$$

leads to the following approximation to the LFR in (61):

$$H_j^W(\vec{\nu}) \approx J(\vec{\nu}) E_j^W(\Phi, \Theta) \approx J_{\text{cyl}}(\vec{\nu}) E_{j,\text{cyl}}^W(\Phi). \quad (71)$$

With this geometry, the expression $E_{j,\text{cyl}}^W(\Phi)$ simplifies to

$$E_{j,\text{cyl}}^W(\Phi) \approx \frac{D_{\text{sd}}^2 \Delta_x^3 \Delta_z}{D_{\text{so}}} \sum_{\sigma \in \mathcal{B}_j(\Phi)} \tilde{u}_{j,\sigma} \frac{\|\vec{\ell}_{j,\sigma}\|^2}{D_{2,j,\sigma}^3} |\csc(\Phi - \sigma)|. \quad (72)$$

This approximation is reasonably accurate until $|\Theta|$ approaches $\pi/2$ minus the largest cone angle used in the CT system. For the purposes of variance prediction, the inaccuracy of the LFR in large- Θ regions has a negligible effect, since for small cone angle systems, this inaccurate region is a very small fraction of the entire frequency space that is integrated in (24).

Figure 6 shows estimates of the local projection-weight-backprojection frequency responses H_j^W and the corresponding approximate noise power spectra S_j for a weighting matrix \mathbf{W} corresponding to an XCAT chest phantom [3]. Three cases are shown: the actual LFR found from (17), the spherically-separable approximation (61) for general CT geometries, and the cylindrically-separable approximation (71) specific to small cone-angle 3DCT. Comparing the first row of this figure to the second row, the approximate local frequency response in (61) closely matches the overall appearance of the DFT-based frequency response except for large- Θ regions. These are of less importance to approximating the noise power spectrum, as can be seen in the figures comparing them. Furthermore, comparing the third row, the result of (71) also

matches the DFT LFR except for large Θ , but again this error has a diminished effect on the NPS.

G. Fourier Transform of a Projected Function

Let \mathbf{X} be an $(n-1) \times n$ matrix, \vec{x} be a unit vector perpendicular to the rows of \mathbf{X} , and $\tilde{\mathbf{X}}$ be the $n \times n$ extension of \mathbf{X} made by appending \vec{x} to the bottom:

$$\tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ \vec{x}^T \end{bmatrix}.$$

The function \tilde{f} , a function of an n -dimensional vector, is an extension of f , a function of an $(n-1)$ -dimensional vector, that simply evaluates f using the first $(n-1)$ components of its argument. We can see from these definitions that $\tilde{f}(\tilde{\mathbf{X}}\vec{n}) = f(\mathbf{X}\vec{n})$ $1(\vec{x}^T \vec{n}) = f(\mathbf{X}\vec{n})$, where $1(\cdot)$ is a constant function evaluating to one.

With these definitions, the n -dimensional Fourier transform of $f(\mathbf{X}\vec{n})$ is given by:

$$\begin{aligned} & \int f(\mathbf{X}\vec{n}) \exp(-j2\pi\vec{\nu}^T \vec{n}) d\vec{n} \quad (73) \\ &= \int f(\mathbf{X}\vec{n}) 1(\vec{x}^T \vec{n}) \exp(-j2\pi\vec{\nu}^T \vec{n}) d\vec{n} \\ &= \int \tilde{f}(\tilde{\mathbf{X}}\vec{n}) \exp(-j2\pi\vec{\nu}^T \vec{n}) d\vec{n} \\ &= \det(\tilde{\mathbf{X}})^{-1} \int \tilde{f}(\vec{m}) \exp(-j2\pi\vec{\nu}^T \tilde{\mathbf{X}}^{-1} \vec{m}) d\vec{m} \\ &= \det(\tilde{\mathbf{X}})^{-1} (\mathcal{F}^n \tilde{f})(\tilde{\mathbf{X}}^{-T} \vec{\nu}) \\ &= \det(\tilde{\mathbf{X}})^{-1} (\mathcal{F}^{n-1} f)(\mathbf{X}^{+T} \vec{\nu}) (\mathcal{F}^1 1)(\vec{x}^T \vec{\nu}), \quad (74) \end{aligned}$$

where \mathbf{X}^{+T} is the transpose of the pseudo-inverse of \mathbf{X} . The last line above uses a separation property of the FT; separate the identity matrix into an arbitrary top half and bottom half:

$$\mathbf{I}_n = \begin{bmatrix} \mathbf{S}_t \\ \mathbf{S}_b \end{bmatrix}$$

where \mathbf{S}_t is an $k \times n$ matrix that keeps the first k elements and discards the rest and \mathbf{S}_b is an $(n-k) \times n$ matrix that discards the first k . Then, if $h(\vec{x}) = f(\mathbf{S}_t \vec{x}) \cdot g(\mathbf{S}_b \vec{x})$,

$$(\mathcal{F}^n h)(\vec{\nu}) = (\mathcal{F}^k f)(\mathbf{S}_t^T \vec{\nu}) \cdot (\mathcal{F}^{n-k} g)(\mathbf{S}_b^T \vec{\nu}).$$

In this case, $k = n-1$, $h = \tilde{f}$, and $g = 1$, a 1D function trivially equal to 1 everywhere. We also use the fact that

$$\tilde{\mathbf{X}}^{-T} = \begin{bmatrix} \mathbf{X}^+ \\ \vec{x}^T \end{bmatrix}.$$

As for the determinant, if we take a QR factorization $\mathbf{X} = \mathbf{R}\mathbf{Q}$, where \mathbf{R} is a $(n-1) \times (n-1)$ matrix and \mathbf{Q} is $(n-1) \times n$ with orthonormal rows, then $\det(\tilde{\mathbf{X}}) = \det(\mathbf{R})$.

REFERENCES

- [1] S. M. Schmitt and J. A. Fessler, "Fast variance prediction for iteratively reconstructed CT images with locally quadratic regularization," *IEEE Trans. Med. Imag.*, 2015. Submitted.
- [2] Y. Long, J. Fessler, and J. Balter, "3D forward and back-projection for X-Ray CT using separable footprints," *Medical Imaging, IEEE Transactions on*, vol. 29, no. 11, pp. 1839–1850, 2010.

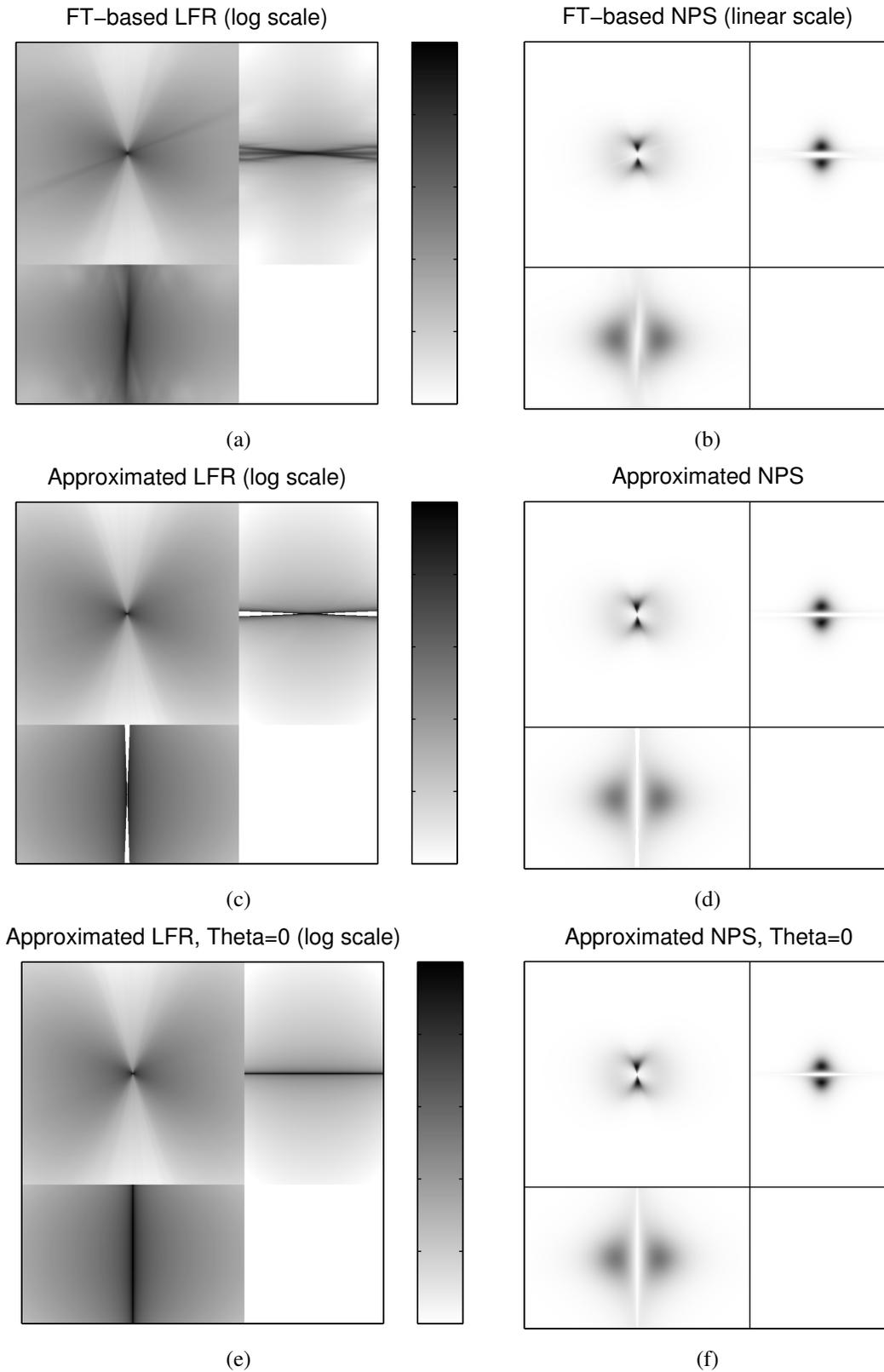


Fig. 6: Planes cut through local frequency responses (left) and corresponding noise power spectra (right). In each image, the upper-left portion is the (ν_1, ν_2) plane where $\nu_3 = 0$; top-right is (ν_3, ν_2) , $\nu_1 = 0$, and bottom-left is (ν_1, ν_3) , $\nu_2 = 0$. First row (a,b): from (17), the Fourier transform of the directly computed LIR

Middle row (c,d): from (61), a spherically-separable approximation

Bottom row (e,f): from (71), a cylindrically-separable approximation

Note: In figures (b,d,f), the bottom section ($\nu_2 = 0$) is scaled by a factor of 4 for visibility.

- [3] W. P. Segars, M. Mahesh, T. J. Beck, E. C. Frey, and B. M. W. Tsui, "Realistic CT simulation using the 4D XCAT phantom," *Med. Phys.*, vol. 35, pp. 3800–8, Aug. 2008.

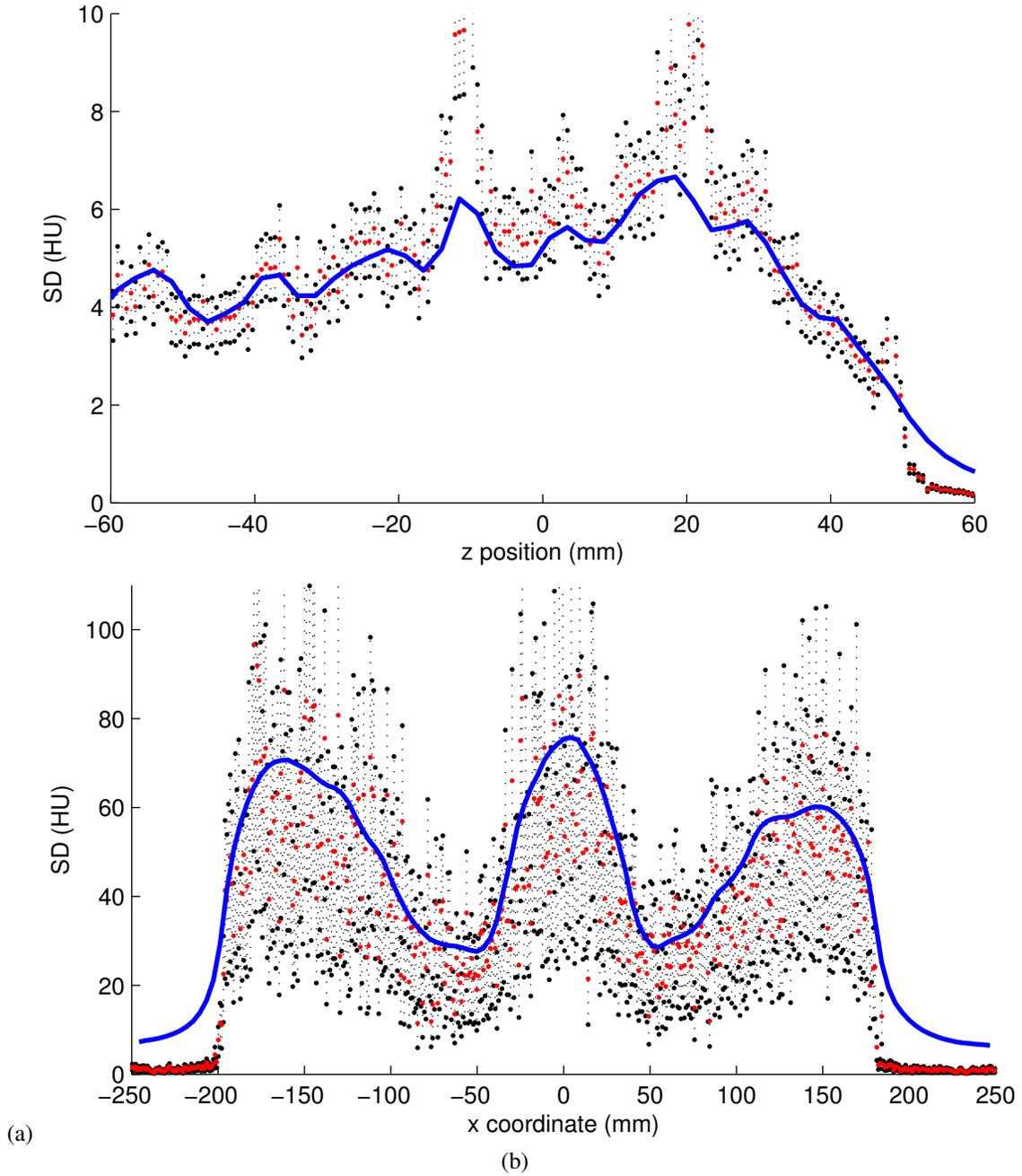


Fig. 7: Unsmoothed standard deviation profiles corresponding to simulation in Figure 1(e) (top, a) and real phantom data in Figure 3(e) (bottom, b). Red points indicate the sample standard deviation for each voxel. Black points indicate the 2.5% and 97.5% quantiles of the distribution of the sample standard deviation, assuming the sample standard deviation is the ground truth and sample realizations are Gaussian. The solid blue line is our proposed standard deviation prediction.