NONLINEAR KERNEL BACKPROJECTION FOR COMPUTED TOMOGRAPHY

Hiroyuki Takeda and Peyman Milanfar

{htakeda,milanfar}@soe.ucsc.edu
University of California, Santa Cruz

ABSTRACT
In this paper, we propose a kernel backprojection method for computed tomography. The classical backprojection method estimates an unknown pixel value by the summation of the projection values with linear weights, while our kernel backprojection is a generalized version of the classic approach, in which we compute the weights from a kernel (weight) function. The generalization reveals that the performance of the backprojection operation strongly depends on the choice of the kernel, and a good choice of the kernels effectively suppresses both noise and streak artifacts while preserving major structures of the unknown phantom. The proposed method is a two-step procedure where we first compute a preliminary estimate of the phantom (a “pilot”), from which we compute the kernel weights. From these kernel weights we then re-estimate the phantom, arriving at a much improved result. The experimental results show that our approach significantly enhances the backprojection operation not only numerically but also visually.

Index Terms—tomography, projection reconstruction, filtered backprojection, nonlinear filters, kernel regression

1. INTRODUCTION
Filtered backprojection (FBP) is a widely-used method for image reconstruction from tomographic projections. Although it is simple, FBP is sensitive to noise due to the high-pass filtering of noise-ridden projections and also suffers from streak (star) artifacts unless the number of projections is sufficiently large. However, fewer projections are always preferable not only because of the scanning time, but also the amount of radiation exposure to patients.

In order to suppress both noise and streak effects, the iterative refinement using the total variation (TV) regularization in spatial domain is one of the most effective methods. Its performance is well demonstrated in [1] and also in [2] as the compressed sensing approach.

In this paper, first we study and generalize the backprojection operation and present a nonlinear backprojection method with improved performance in Section 2. The classical backprojection operation estimates a local value of the unknown phantom using the path integral of the given projections, which, in the discrete case, is implemented as a summation of the nearest or interpolated projection values along the path [3]. Bilinear and bicubic approaches are typical choices for interpolation. On the other hand, our nonlinear kernel backprojection (KBP) interpolates the projection values while suppressing the noise effect using nonlinear weights, which we learn from the initial guess of the underlying phantom. The proposed backprojection is a two-step approach: (i) We estimate the unknown phantom by the TV approach [1, 2] and compute local steering kernels (LSK) from this (“pilot”) estimated phantom, and then (ii) we reconstruct the phantom again with the nonlinear weights given by the local steering kernels.

Next, in Section 3, we describe the TV approach in conjunction with the proposed KBP in order to improve the tomographic reconstruction. Specifically, we replace the standard BP in the TV approach with KBP. After obtaining LSK from the estimated phantom by the TV approach, the TV iteration with KBP refines the estimated phantom further. The experimental results show that the proposed KBP operation significantly enhance the reconstruction performance numerically and visually.

2. NONLINEAR KERNEL BACKPROJECTION
First, we briefly review the basics of the projection reconstruction problem in the spatial domain, and derive kernel backprojection.

2.1. Review
Using an additive noise model, we define the data model as

\[
\tilde{y} = z + \varepsilon = Ru + \varepsilon.
\]

where \( \tilde{y} \in \mathbb{R}^{P \times Q} \) is the measured projections with noise, \( z \in \mathbb{R}^{P \times Q} \) is the noise-free projections, \( u \in \mathbb{R}^{N \times M} \) is the unknown phantom, \( R \in \mathbb{R}^{P \times Q \times N \times M} \) is the Radon operation, \( \varepsilon \in \mathbb{R}^{P \times Q} \) is zero-mean noise, and the matrices with underscore represent that they are lexicographically ordered into column-stack vectors (e.g. \( \tilde{y} \in \mathbb{R}^{PQ \times 1} \)). A direct solution of \( u \) is given by the least-squares estimator,

\[
\hat{u}_{LS} = \arg \min_{u} \| \tilde{y} - Ru \|_{2}^{2}
\]

as

\[
\hat{u}_{LS} = (R^{T}R)^{-1} R^{T} \tilde{y}.
\]

In general, we interpret \( R^{T} \tilde{y} \) as backprojection and \((R^{T}R)^{-1}\) as high-pass filter. It is noteworthy that, in FBP, using the projection slice theorem [4], we can reverse the order: we first
apply a high-pass filter on the projections and then perform backprojection. However, due to the fact that $R^T R$ is typically ill-conditioned, in practice, its inversion is problematic, and small changes (e.g., noise) in the given projections leads to huge distortions in the reconstructed images.

In order to stabilize the solution, the regularization technique is an immediate choice, and TV is the most widely used. That is, we have

$$u_{TV} = \arg \min_u \| y - R u \|^2 + \lambda \left( \| \Gamma_{x_1} u \|_1 + \| \Gamma_{x_2} u \|_1 \right),$$

where we can iteratively estimate $u$ with the steepest descent method as

$$u_{TV}^{(v+1)} = u_{TV}^{(v)} + \mu \left[ R^T (y - R u_{TV}^{(v)}) - \lambda \left\{ \Gamma_{x_1}^T \text{sign}(\Gamma_{x_1} u_{TV}^{(v)}) + \Gamma_{x_2}^T \text{sign}(\Gamma_{x_2} u_{TV}^{(v)}) \right\} \right],$$

where $\lambda$ is the regularization parameter, $\Gamma_{x_1}$ and $\Gamma_{x_2}$ are first derivative operators (say Sobel filters) along $x_1$- and $x_2$-axes, respectively, and $\mu$ is the step size.

### 2.2. Kernel Backprojection

As explained in (3), we can regard the standard tomographic reconstruction as a two-step process: the backprojection step ($R^T y$) and the high-pass filtering step ($R^T R)^{-1}$. Due to the ill-condition of $R^TR$, it would be preferable to compute a stable estimate of the noise-free projections $x$ first.

In the continuous case, we can express the backprojection $u = R^T y$ as

$$\tilde{u}(x) = \int_0^\pi y(t, \theta) d\theta, \text{ with } t = x_1 \cos \theta + x_2 \sin \theta,$$

where $\tilde{u}(\cdot)$ is the unknown function before the high-pass filtering. But in practice, we implement it as a discrete form,

$$\tilde{u}(x) = \sum_{q=1}^Q y(t_q, \theta_q) \delta \theta, \text{ with } t_q = x_1 \cos \theta_q + x_2 \sin \theta_q,$$

where, in general, $y(t_q, \theta_q)$ may be unavailable at every value of $t_q$ and $\theta_q$. Thus, we need to estimate $y(t_i, \theta_q)$ (or more precisely, its noise-free version $z(t_i, \theta_q)$ somehow from its neighbors. Typically, nearest, bilinear, bicubic interpolations are often used [4, 5].

In this paper, we use the kernel regression framework [6] to estimate $z(t_i, \theta_q)$. On the projection at the angle $\theta_q$, we have $P$ noise-ridden samples, $y(t_i, \theta_q)$ with $i = 1, \ldots, P$:

$$y(t_i, \theta_q) = z(t_i, \theta_q) + \varepsilon(t_i, \theta_q), \text{ with } t_i = x_1 \cos \theta_q + x_2 \sin \theta_q.$$

Assuming that the noise-free function $z(t_i, \theta_q)$ is locally smooth and using Taylor expansion as the local representation, between $z(t_j, \theta_q)$ and $z(t_i, \theta_q)$, we have the following relationship

$$z(t_i, \theta_q) = z(t_j, \theta_q) + \frac{\partial z(t_j, \theta_q)}{\partial t}(t_i - t_j) + \frac{1}{2!} \frac{\partial^2 z(t_j, \theta_q)}{\partial t^2}(t_i - t_j)^2 + \cdots = \beta_0 + \beta_1 (t_i - t_j) + \beta_2 (t_i - t_j)^2 + \cdots.$$

Using a local neighborhood, we solve for the desired projection value $\beta_0$ using a weighted least squares formulation which assigns higher weights to nearby samples as:

$$\min_{\beta_0, \beta_1, \ldots} \sum_{i=1}^P \left[ y(t_i, \theta_q) - \beta_0 - \beta_1 (t_i - t_j) - \beta_2 (t_i - t_j)^2 - \cdots \right] K_\theta(t_i - t_j).$$

This is the 1-D kernel regression formulation, and when we ignore all the higher order terms ($\beta_1, \beta_2, \ldots$), the optimization yields the (zeroth-order) estimator of $z(t_j, \theta_q)$ as

$$\hat{z}(t_j, \theta_q) = \hat{\beta}_0 = \frac{\sum_i K_\theta(t_i - t_j) y(t_i, \theta_q)}{\sum_i K_\theta(t_i - t_j)}.$$

Now, plugging $\hat{z}(t_j, \theta_q)$ into the backprojection (7), we have

$$\hat{u}_{\text{KBP}}(x) = \sum_{q=1}^Q \left\{ \sum_i K_\theta(t_i - t_j) y(t_i, \theta_q) \right\} \delta \theta,$$

which we call it kernel backprojection (KBP); and rewrite it in matrix form as $\hat{u} = R^T y$. Fig. 1 illustrates a schematic representation of kernel backprojection (12) and a typical choice of weights [5].

### 2.3. Nonlinear Kernel Functions

In this section, we take one step further toward better estimation of $\hat{z}(t_j, \theta_q)$ (11). Specifically, we employ a choice of the kernel function that not only estimates the unknown projection value but also effectively suppresses the streak and noise effects while preserving local structures (texture and orientations).

The desired kernels adapt to local structures of the phantom, and in order to obtain such kernels, we use the local steering kernel (LSK) technique. The LSK captures local orientation structures, and, as we have shown earlier, it is an
effective tool for image restoration including denoising, interpolation, and deblurring [7, 8]. The LSK is defined as
\[ k(x_l - x_j) = \sqrt{\det C_l} \exp \left\{ -\frac{(x_l - x_j)^T C_l (x_l - x_j)}{2h^2} \right\}, \tag{13} \]
where \( x_j \) is the position where we compute the LSK, \( x_l \) is the neighboring pixel positions around \( x_j \), \( C_l \) is the \( 2 \times 2 \) local covariance matrix of the local gradients around \( x_j \), and \( h \) is the smoothing parameter that controls the width of the LSK. Fig. 2 shows examples of LSK near an edge region and flat regions.

Next, using the LSK, we compute the kernels for back-projection (12) by taking Radon transform of the LSK at the projection angle \( \theta_q \):
\[ K_q(t_l - t_j) = R_{\theta_q}\{k(x_l - x_j)\}. \tag{14} \]
Now, the kernel \( K_q \) inherits the local structures of the phantom and it reflects the local structures in the backprojection operation with suppressing noise and streak artifacts. Of course, in the expression above, in order to compute the kernels \( k(\cdot) \) we need to have access to the underlying phantom, which is unrealistic given only the projections. As such, we produce a “pilot” estimate of the phantom using an independent, low-complexity algorithm, and use this phantom to calculate the desired kernel weights. We demonstrate this in the next section with several experiments.

3. EXPERIMENTS

Having proposed the nonlinear KBP, in this section, we explain how we obtain a good set of LSK from the given projections, and show some examples along with comparisons to the standard filtered BP, and the TV approach (5).

3.1. Implementation

In order to obtain a good set of LSK, we estimate the unknown phantom first by the TV approach (5). This pilot estimate tells us the local structures of the unknown phantom and their spatial locations. Next, using (13) and (14), we compute the LSK for all the pixels of the estimated phantom, and then we have the nonlinear kernel \( K_q \) by taking Radon transform of the LSK. Once, the kernels \( K_q \) are available, we can perform KBP (\( \tilde{R}^T \)). We replace the standard backprojection \( R^T \) in (5) with KBP \( \tilde{R}^T \), and estimate the unknown phantom again. Fig. 3 shows the summary of our reconstruction algorithm with KBP.

It is worth noting that since each location \( x_l \) receives its own covariance matrix, the shape of the kernel is not a simple Gaussian.

3.2. Examples

The first example is the familiar Shepp-Logan phantom. Using this Shepp-Logan phantom \((128 \times 128)\) shown in Fig. 4(a), we generate 60 projections with equally spaced angles and add white Gaussian noise \((\text{SNR} = 45 [\text{dB}])\) to the projections, and the figures (b)-(d) are the reconstructed phantoms by the standard filtered backprojection, the TV approach (5), and the KBP method, respectively. The corresponding PSNR values and SSIM indexes are (b) 22.88 [dB], 0.5489, (c) 44.38 [dB], 0.9963, and (d) 45.72 [dB], 0.9970, respectively.

A MATLAB implementation is freely available at http://www.ece.uwaterloo.ca/~z70wang/research/ssim/ssim.m
PSNR values and SSIM indexes are (b) 28.16[dB], 0.8506 and (c) 29.32[dB], 0.8683, respectively. The plot next to Fig. 5(a) shows the cross section (indicated by the orange line) of the original and the reconstructed images.

4. CONCLUSION AND FUTURE WORKS

We presented a nonlinear KBP approach and its implementation. The experimental results show that our method outperforms the standard FBP and the TV approach not only numerically, but also visually.

Although the KBP works well, some important issues remain to be studied: (i) the kernels are not specifically designed with sharp boundaries in mind. (ii) Other kernel techniques to capture the local image structures are also possible, for example bilateral kernel [9]. (iii) In this paper, we ignored the higher order terms in the kernel regression (10). Including the higher order terms will result in a sharpening effect in the restored image. This property might enhance the performance of KBP as well. (iv) Reduction of the computational complexity is also an important issue to study. The computational load of the naive implementation of KBP is proportional to the support size of the kernel function. After obtaining the LSK with setting the diameter of the kernel support d-pixels, the computational load of KBP is approximately d times the computational load of standard BP.

5. REFERENCES