To appear in
ICASSP 2010

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.

This work was supported in part by AFOSR Grant FA9550-07-1-0365.

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

$$\underline{\mathbf{y}} = \underline{\mathbf{z}} + \underline{\boldsymbol{\varepsilon}} = \mathbf{R}\underline{\mathbf{u}} + \underline{\boldsymbol{\varepsilon}}, \tag{1}$$

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

$$\underline{\hat{\mathbf{u}}}_{\mathrm{LS}} = \arg\min_{\underline{\mathbf{u}}} \left\|\underline{\mathbf{y}} - \mathbf{R}\underline{\mathbf{u}}\right\|_2^2 \tag{2}$$

as

$$\underline{\hat{\mathbf{u}}}_{\mathrm{LS}} = \left(\mathbf{R}^T\mathbf{R}\right)^{-1}\mathbf{R}^T\underline{\mathbf{y}}. \tag{3}$$

In general, we interpret $\mathbf{R}^T\underline{\mathbf{y}}$ as backprojection and $\left(\mathbf{R}^T\mathbf{R}\right)^{-1}$ 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 $\mathbf{R}^T\mathbf{R}$ 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

$$\underline{\hat{\mathbf{u}}}_{\mathrm{TV}} = \arg\min_{\underline{\mathbf{u}}} \left\|\underline{\mathbf{y}} - \mathbf{R}\underline{\mathbf{u}}\right\|_2^2 + \lambda\left(\|\mathbf{\Gamma}_{x_1}\underline{\mathbf{u}}\|_1 + \|\mathbf{\Gamma}_{x_2}\underline{\mathbf{u}}\|_1\right), \quad (4)$$

where we can iteratively estimate $\mathbf{u}$ with the steepest descent method as

$$\underline{\hat{\mathbf{u}}}_{\mathrm{TV}}^{(\nu+1)} = \underline{\hat{\mathbf{u}}}_{\mathrm{TV}}^{(\nu)} + \mu\Big[\mathbf{R}^T\Big(\underline{\mathbf{y}} - \mathbf{R}\underline{\hat{\mathbf{u}}}_{\mathrm{TV}}^{(\nu)}\Big) - \lambda\left\{\mathbf{\Gamma}_{x_1}^T\mathrm{sign}(\mathbf{\Gamma}_{x_1}\underline{\hat{\mathbf{u}}}_{\mathrm{TV}}^{(\nu)}) + \mathbf{\Gamma}_{x_2}^T\mathrm{sign}(\mathbf{\Gamma}_{x_2}\underline{\hat{\mathbf{u}}}_{\mathrm{TV}}^{(\nu)})\right\}\Big] \quad (5)$$

where λ is the regularization parameter, $\mathbf{\Gamma}_{x_1}$ and $\mathbf{\Gamma}_{x_2}$ are first derivative operators (say Sobel filters) along x_1- and x_2-axes, respectively, and μ 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 ($\mathbf{R}^T\underline{\mathbf{y}}$) and the high-pass filtering step $(\mathbf{R}^T\mathbf{R})^{-1}$. Due to the ill-condition of $\mathbf{R}^T\mathbf{R}$, it would be preferable to compute a stable estimate of the noise-free projections $\mathbf{z}$ first.

In the continuous case, we can express the backprojection $\underline{\tilde{\mathbf{u}}} = \mathbf{R}^T\underline{\mathbf{y}}$ as

$$\tilde{u}(\mathbf{x}) = \int_0^{\pi} y(t,\theta)d\theta, \quad \text{with } t = x_1\cos\theta + x_2\sin\theta, \quad (6)$$

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

$$\tilde{u}(\mathbf{x}_j) = \sum_{q=1}^{Q} y(t_j,\theta_q)\delta\theta, \quad \text{with } t_j = x_{1j}\cos\theta_q + x_{2j}\sin\theta_q, \quad (7)$$

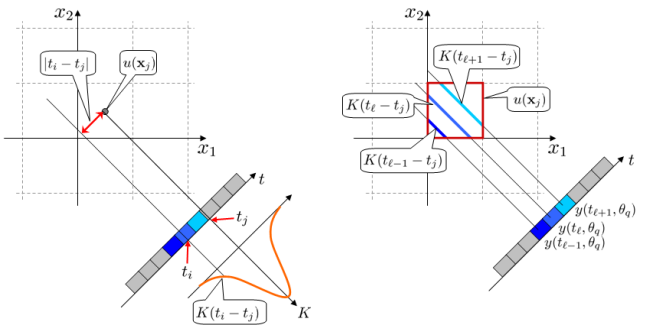
where, in general, $y(t_j,\theta_q)$ may be unavailable at every value of t_j and θ_q. Thus, we need to estimate $y(t_j,\theta_q)$ (or more precisely, its noise-free version $z(t_j,\theta_q)$) somehow from its neighbors. Typically, nearest, bilinear, bicubic interpolations are often used [4, 5].

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

$$y(t_i,\theta_q) = z(t_i,\theta_q) + \varepsilon(t_i,\theta_q), \quad i = 1,\cdots,P. \quad (8)$$

Assuming that the noise-free function $z(t,\theta_q)$ 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

$$\begin{aligned} z(t_i,\theta_q) &= z(t_j,\theta_q) + \frac{\partial z(t_j,\theta_q)}{\partial t}(t_i - t_j) \\ &\quad + \frac{1}{2!}\frac{\partial^2 z(t_j,\theta_q)}{\partial t^2}(t_i - t_j)^2 + \cdots \\ &= \beta_0 + \beta_1(t_i - t_j) + \beta_2(t_i - t_j)^2 + \cdots. \end{aligned} \quad (9)$$



(a) Kernel backprojection

(b) Kernels proportional to the length of overlapping segments

Fig. 1. A schematic representation of kernel backprojection and another kernel often used: (a) the kernel is the function of spatial distances, and (b) the kernels are proportional to the length of overlapping segments.

Using a local neighborhood, we solve for the desired projection value β_0 using a weighted least squares formulation which assigns higher weights to nearby samples as:

$$\min_{\beta_0,\beta_1,\cdots}\sum_{i=1}^{P}\left[y(t_i,\theta_q) - \beta_0 - \beta_1(t_i - t_j) - \beta_2(t_i - t_j)^2 - \cdots\right]^2 K_q(t_i - t_j). \quad (10)$$

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

$$\hat{z}(t_j,\theta_q) = \hat{\beta}_0 = \frac{\sum_i K_q(t_i - t_j)\, y(t_i,\theta_q)}{\sum_i K_q(t_i - t_j)}. \quad (11)$$

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

$$\tilde{u}_{\mathrm{KBP}}(\mathbf{x}_j) = \sum_{q=1}^{Q}\left\{\frac{\sum_i K_q(t_i - t_j)\, y(t_i,\theta_q)}{\sum_i K_q(t_i - t_j)}\right\}\delta\theta, \quad (12)$$

which we call it *kernel backprojection* (KBP); and rewrite it in matrix form as $\underline{\tilde{\mathbf{u}}} = \widetilde{\mathbf{R}}^T\underline{\mathbf{y}}$. 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

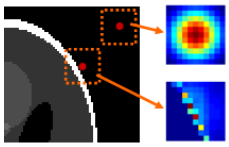


Fig. 2. Examples of local steering kernels at edge and flat regions.

effective tool for image restoration including denoising, interpolation, and deblurring [7, 8]. The LSK is defined as

$$k(\mathbf{x}_l - \mathbf{x}_j) = \sqrt{\det \mathbf{C}_l} \exp\left\{-\frac{(\mathbf{x}_l - \mathbf{x}_j)^T \mathbf{C}_l (\mathbf{x}_l - \mathbf{x}_j)}{2h^2}\right\}, \tag{13}$$

where $\mathbf{x}_j$ is the position where we compute the LSK, $\mathbf{x}_l$ is the neighboring pixel positions around $\mathbf{x}_j$, $\mathbf{C}_l$ is the 2×2 local covariance matrix of the local gradients around $\mathbf{x}_j$, 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 region[1].

Next, using the LSK, we compute the kernels for backprojection (12) by taking Radon transform of the LSK at the projection angle θ_q:

$$K_q(t_i - t_j) = R_{\theta_q}\{k(\mathbf{x}_l - \mathbf{x}_j)\}. \tag{14}$$

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 ($\widetilde{\mathbf{R}}^T$). We replace the standard backprojection $\mathbf{R}^T$ in (5) with KBP $\widetilde{\mathbf{R}}^T$, and estimate the unknown phantom again. Fig. 3 shows the summary of our reconstruction algorithm with KBP.

[1]It is worth noting that since each location $\mathbf{x}_l$ receives its own covariance matrix, the shape of the kernel is not a simple Gaussian.

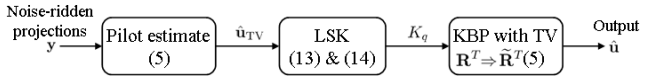
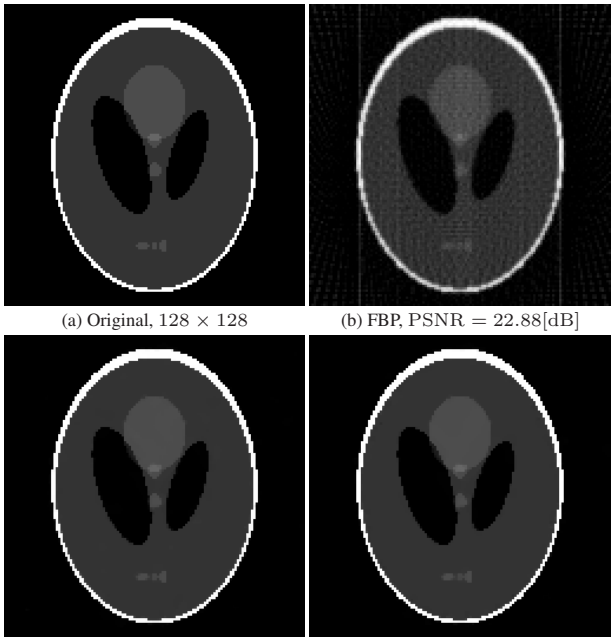


Fig. 3. Block diagram of the projection reconstruction using KBP.



(a) Original, 128×128 (b) FBP, PSNR $= 22.88$[dB]

(c) BP with TV, PSNR $= 44.38$[dB] (d) KBP with TV, PSNR $= 45.72$[dB]

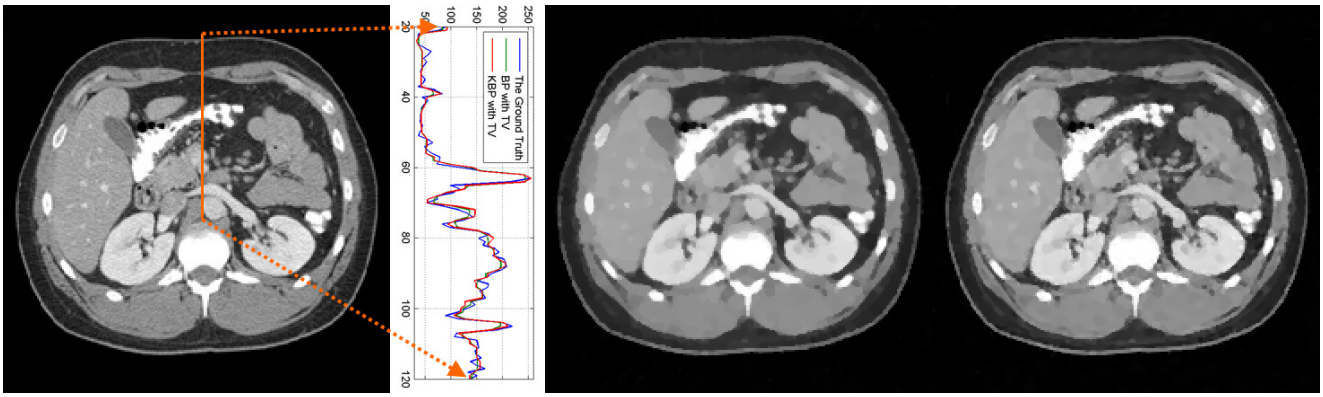
Fig. 4. Shepp-Logan example: we generate 60 projections from Shepp-Logan phantom shown in (a) with equally spaced angles and add white Gaussian noise (SNR $= 45$[dB]) to the projections, and the figures (b)-(d) are the reconstructed phantoms by the standard filtered backprojection, the TV approach (5), 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.

3.2. Examples

The first example is the familiar Shepp-Logan phantom. Using this Shepp-Logan phantom (128×128) shown in Fig. 4(a), we generate 60 projections with equally spaced angles in $[0, \pi]$ and add white Gaussian noise (SNR $= 45$[dB]) to the projections. The estimated phantoms by the standard FBP, the TV approach (5) with $\lambda = 0.003$, and the KBP with $\lambda = 0.0006$ are shown in Figs. 4(b)-(d), respectively. The corresponding PSNR values are (b)22.88[dB], (c)44.38[dB], and (d)45.72[dB]. Also the SSIM indexes[2] are (b)0.5489, (c)0.9963, and (d)0.9970

Next, we use a more realistic phantom for the second example: an abdominal (216×216) phantom shown in Fig. 5(a). In these examples, we generate 60 projections with equally spaced angles and add white Gaussian noise (SNR $= 45$[dB]) to the projections. Again, the estimated phantoms by the the TV approach (5) with $\lambda = 0.002$ and the KBP with $\lambda = 0.001$ are shown in Figs. 5(b)-(c), respectively. The corresponding

[2]A MATLAB implementation is freely available at `http://www.ece.uwaterloo.ca/~z70wang/research/ssim/ssim.m`



(a) Original, 216×216 (b) BP with TV, PSNR $= 28.16$[dB] (c) KBP with TV, PSNR $= 29.32$[dB]

(d) Original (e) BP with TV (f) KBP with TV

Fig. 5. Abdominal CT example: we generate 60 projections from the abdominal phantom shown in (a) with equally spaced angles and add white Gaussian noise (SNR $= 45$[dB]) to the projections, and the figures (b) and (c) are the reconstructed phantoms by the TV approach (5) and the KBP method, respectively. Figures (d)-(f) show enlarged middle portions of (a)-(c), respectively. The corresponding PSNR values and SSIM indexes are (b)28.16[dB], 0.8506 and (c)29.32[dB], 0.8683, respectively.

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

[1] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, February 2006.

[2] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Processing Magazine*, pp. 72–82, March 2008.

[3] R. L. Siddon, "Fast calculation of the exact radiological path for a three-dimensional CT array," *Medical Physics*, vol. 12, no. 2, pp. 252–255, March 1985.

[4] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*, SIAM (Society for Industrial and Applied Mathematics), New York, 1988.

[5] P. P. Bruyant, "Analytic and iterative reconstruction algorithms in SPECT," *The Journal of Nuclear Medicine*, vol. 43, no. 10, pp. 1343–1358, October 2002.

[6] M. P. Wand and M. C. Jones, *Kernel Smoothing*, Monographs on Statistics and Applied Probability. Chapman and Hall, London; New York, 1995.

[7] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, February 2007.

[8] H. Takeda, S. Farsiu, and P. Milanfar, "Deblurring using regularized locally-adaptive kernel regression," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 550–563, April 2008.

[9] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *Proceeding of the 1998 IEEE International Conference of Compute Vision, Bombay, India*, pp. 836–846, January 1998.