Published in final edited form as:

IEEE Trans Nucl Sci. 2015 February ; 62(1): 131–136. doi:10.1109/TNS.2014.2363776.

Revisit of the Ramp Filter

Gengsheng L. Zeng [Fellow, IEEE]

Department of Electrical Engineering, Weber State University, Ogden, UT 84408 USA and the
Department of Radiology, University of Utah, Salt Lake City, UT 84108 USA

Gengsheng L. Zeng: larryzeng@weber.edu

Abstract

An important part of the filtered backprojection (FBP) algorithm is the ramp filter. This paper derives the discrete version of the ramp filter in the Fourier domain and studies the windowing effects. When a window function is used to control the noise, the image amplitude will be affected and reduced. A simple remedy is proposed to improve the image accuracy when a window function must be used.

Index Terms

Filtered backprojection algorithm; analytic image reconstruction

I. Introduction

The filtered backprojection (FBP) algorithm has been serving as the driving engine in tomographic image reconstruction for many decades [1–4]. This algorithm consists of ramp filtering and backprojection procedures. Ramp filtering can be implemented as convolution in the spatial domain or multiplication in the Fourier domain.

The continuous ramp filter has a zero DC gain. Therefore, if one directly samples the continuous ramp filter to obtain a discrete version, the discrete version will also have a zero DC gain. It has been shown that this zero DC gain can cause a reconstruction bias error [5–7]. In fact, this error can be easily corrected by using a non-zero DC gain in the sampled ramp filter. This paper will give a systematic treatment of the issue and give a relationship between a continuous filter's transfer function and its exact discrete counterpart.

This paper will use the discrete/periodic duality [8] to derive an exact explicit expression for the discrete ramp filter. The discrete/periodic duality is a fundamental property for a Fourier transform pair. If a function is periodic in one domain, it is discrete in the other domain, and vice versa. Since the projections are to be filtered in the discrete Fourier domain, an exact circular convolution expression in the spatial domain must be obtained first. A circular convolution is nothing but a linear convolution with two periodic functions; these two functions have the same period. If these two periodic functions are also discrete, the circular convolution can be calculated via the Discrete Fourier Transform (DFT). The DFT can be efficiently computed by the Fast Fourier Transform (FFT).

In general, the FFT method is faster than the convolution method. However, in some special cases, the convolution method can be faster, for example, when the filter is spatially varying. The backprojection is still the dominating procedure in terms of computation cost.

Noise control in the FBP algorithm is achieved by the application of a window function, which is normally a low-pass filter with a unity DC gain. The window function can affect the accuracy of the FBP reconstruction. This effect has been ignored for the purpose of simplification in image analysis [9]. This paper will investigate how the window function can reduce the amplitude of the reconstructed image and propose a simple remedy.

II. Methods

A. Converting a continuous filter into a discrete filter

Linear filtering can be achieved either by convolution with a spatial domain kernel function (i.e., a convolver) or by multiplication with a Fourier domain transfer function. The transfer function is the Fourier transform of the spatial domain kernel function.

First of all, it is generally incorrect to obtain a discrete transfer function by sampling a continuous transfer function. Let the projection at a certain view angle θ be $p_\theta(n)$, where n is the index along the detector. The projection $p_\theta(n)$ is discretely sampled and has a finite support N . Without loss of generality, the sampling interval is assumed to be 1, and the projection can be treated as band-limited, with a bandwidth of 0.5.

In theory, a signal cannot simultaneously be band-limited in the Fourier domain and have a finite support in the spatial domain. However, in our case, the projection is already in the discrete form, forcing the projection be band-limited. In other words, only a band-limited projection can be recovered from these samples.

When the projection is to be filtered by the ramp filter, only a band-limited ramp filter is required, since the projection is band limited by 0.5. The band-limited ramp filter is defined in the Fourier domain as

$$RAMP(\omega) = \begin{cases} |\omega| & \text{if } |\omega| \leq 0.5, \\ 0 & \text{if } |\omega| > 0.5. \end{cases} \quad (1)$$

The spatial domain continuous convolution kernel $h(t)$ is the inverse Fourier transform of (1):

$$\begin{aligned} ramp(t) &= \int_{-\infty}^{\infty} RAMP(\omega) e^{j2\pi\omega t} d\omega = 2 \int_0^{0.5} \omega \cos(2\pi\omega t) d\omega \\ &= \frac{1}{2} \frac{\sin(\pi t)}{\pi t} - \frac{1}{4} \left(\frac{\sin(\pi t/2)}{\pi t/2} \right)^2. \end{aligned} \quad (2)$$

According to the Nyquist sampling criterion, one can sample the function $ramp(t)$ with a sampling interval 1 without losing any information. The sampled (discrete) version of (2) is given as

$$\begin{aligned}
 \text{ramp}(n) &= \frac{1}{2} \frac{\sin(\pi n)}{\pi n} - \frac{1}{4} \left(\frac{\sin(\pi n/2)}{\pi n/2} \right)^2 \\
 &= \begin{cases} 1/4, & n=0, \\ 0, & n \text{ even}, \\ -1/(n\pi)^2, & n \text{ odd}. \end{cases} \quad (3)
 \end{aligned}$$

The filtered projection $q_\theta(n)$ can be accurately obtained as the discrete convolution of the discrete projection $p_\theta(n)$ and the discrete convolver $\text{ramp}(n)$ defined in (3):

$$q_\theta(n) = p_\theta(n) * \text{ramp}(n), \quad (4)$$

where $*$ denotes the linear convolution operator.

Next, the circular convolution will be used to implement the linear convolution (4). In other words, the two functions on the right-hand-side of (4) are changed into periodic functions with the same period. As a result, the function $q_\theta(n)$ on left-hand-side of (4) becomes a periodic function, with the same period as the other two functions have.

To make the idea clear, let us re-phrase the above discussion here. The projections are discrete measurements. An exact discrete convolution kernel for the ramp filter is available. A well-known approach to filter the projections with a ramp filter is to calculate the linear convolution. If the support of the projections has a length N , only the central N samples of the filtered projection are needed in backprojection. This requires that the minimal length of the convolution kernel be $2N-1$. In order to make both functions to have the same length, one must pad the projections with $N-1$ zeros. In general a length M must be selected which is at least $2N-1$ and is a power of 2. The projections are then padded with $M-N$ zeros. Both functions can now be treated as periodic with the same period M .

The user has freedom to choose M . In fact, a larger M results in smaller errors if the sampled ramp filter is to be used. After padding $M-N$ zeros, $p_\theta(n)$ is extended into a periodic function with a period M . A common approach is to let $M = 2N$ by padding N zeros.

Accordingly, the convolver $\text{ramp}(n)$ is truncated, keeping the central M samples with $\text{ramp}(0)$ at the center. Then the M -sample segment of $\text{ramp}(n)$ is extended into a periodic function with a period of M .

With this setup, the convolution can be calculated via the M -point FFT. The discrete filter's transfer function can be numerically obtained by calculating the FFT of the M samples of the convolution kernel, or by an explicit expression to be derived below.

Applying these two periodic functions to the right-hand-side of (4) yields a periodic output $q_\theta(n)$, which also has M samples in one period. It must be pointed out that among these M samples of $q_\theta(n)$, only the central N samples are accurate and can be used in the backprojection procedure. The other $M-N$ samples of $q_\theta(n)$ are not needed and can be discarded.

The Discrete Fourier Transform (DFT) method is able to accurately implement the circular convolution, by performing DFT, multiplication, and inverse DFT (IDFT). When M is a power of 2, the Fast Fourier Transform (FFT) can be used for an efficient implementation of the DFT.

The DFT of the M -sample segment of the discrete $ramp(n)$ is given as

$$H(k) = \sum_{n=0}^{M-1} ramp(n) e^{-j2\pi \frac{nk}{M}} \text{ for } k=0, 1, 2, \dots, M-1. \quad (5)$$

It can be shown that $H(k)$ is almost the same as the discrete samples of the $RAMP(k/M)$ (i.e., letting $\omega = k/M$ in (1) with $k = \pm 1, \pm 2, \dots, \pm 0.5M$), except for the DC gain and some low frequency components. In (1), $RAMP(0) = 0$, while in (5)

$$H(0) = \sum_{n=0}^{M-1} ramp(n). \quad (6)$$

Eq. (5) defines a discrete Fourier transform pair $\{ramp(n), H(k)\}$. Eq. (6) holds for any Fourier transform pair. Since $ramp(n)$ and $H(k)$ are periodic, the summation indices in (5) and (6) as well as the indexes k can any sequential integers with M numbers. Keep in mind that the M -sample segment of $ramp(n)$ in (6) is symmetric with $ramp(0)$ at the center. The DC gain $H(0)$ given by (6) depends on the value M , and is always positive. As M tends to infinity, $H(0)$ approaches to zero.

Eq. (5) can be generalized to derive a discrete transfer function $H(k)$, $k=0, 1, 2, \dots, M-1$, for any continuous transfer function $H_c(\omega)$, whose corresponding discrete convolution kernel is $h_c(n)$:

$$\begin{aligned} H(k) &= \sum_{n=0}^{M-1} h_c(n) e^{-j2\pi \frac{nk}{M}} = \sum_{n=0}^{M-1} \int_{-0.5}^{0.5} H_c(\omega) e^{j2\pi \omega n} d\omega e^{-j2\pi \frac{nk}{M}} \\ &= \int_{-0.5}^{0.5} H_c(\omega) \sum_{n=0}^{M-1} e^{j2\pi n(\omega - k/M)} d\omega = \int_{-0.5}^{0.5} H_c(\omega) \frac{1 - e^{j2\pi M(\omega - k/M)}}{1 - e^{j2\pi(\omega - k/M)}} d\omega \quad (7) \\ &= \int_{-0.5}^{0.5} H_c(\omega) \frac{\sin(\pi M(\omega - k/M))}{\sin(\pi(\omega - k/M))} e^{j\pi(M-1)(\omega - k/M)} d\omega. \end{aligned}$$

Here, $\frac{\sin(\pi M(\omega - k/M))}{\sin(\pi(\omega - k/M))}$, similar to the sinc function, has a removable singularity at $\omega =$

k/M , and it is a continuous function if one defines $\frac{\sin(\pi M(\omega - k/M))}{\sin(\pi(\omega - k/M))} = M$ at $\omega = k/M$. If $H_c(\omega)$ is an even function, (7) reduces to

$$H(k) = \int_{-0.5}^{0.5} H_c(\omega) \frac{\sin(\pi M(\omega - \frac{k}{M}))}{\sin(\pi(\omega - \frac{k}{M}))} \cos(\pi(M-1)(\omega - \frac{k}{M})) d\omega. \quad (8)$$

It is learned from (8) that in general

$$H(k) \neq H_c(k/M). \quad (9)$$

B. The effect of the window function

Many window functions in the Fourier domain have been suggested to suppress the noise, for example, the Metz window [10], the Hann window [11], the Hamming window [12], the Butterworth window [13], and so on. Recently, window functions have been derived to compensate for the projection noise on the ray-by-ray base [14]. Noise suppression is achieved by blurring the projections, and thus image contrast and magnitude are somehow reduced. This magnitude reduction has nothing to do with the DC gain $H(0)$ as discussed in Part A of this section. This magnitude reduction is caused by the narrow bandwidth of the window function.

Let us consider a general lowpass filter. The constraint of the DC gain $H(0) = 1$ of a lowpass filter maintains the integral of the entire image unchanged before and after filtering. When a lowpass filter with a narrow bandwidth is applied, image values are spreading out into the entire $(-\infty, \infty) \times (-\infty, \infty)$ 2D space. The total integral over the $(-\infty, \infty) \times (-\infty, \infty)$ 2D space is still conserved. However, over any finite region containing the object, the total integral is decreased.

It is known that the iterative maximum likelihood expectation maximization (MLEM) algorithm [15–16] is able to maintain the total count in the finite image array at every iteration. In other words, the sum of the forward projections equals to the sum of the measured ray-sum projections. This property makes the MLEM algorithm to have accurate reconstruction in terms of the total count within the image array. This, in turn, helps to achieve smaller local bias when the total count is correct. The MLEM algorithm thus guarantees the same total count in the image array, regardless the image resolution, while the FBP algorithm does not share this good behavior.

One can learn from the MLEM algorithm and suggests a remedy for the FBP algorithm to improve the accuracy of the FBP reconstructions when the window function has a narrow bandwidth. The remedy is to apply a simple scaling factor to the FBP reconstructed image. For the parallel-beam imaging geometry, this scaling factor is defined as

$$s = \frac{\text{sum of measured projections at one view}}{\text{sum of FBP reconstruction}}; \quad (10)$$

for non-parallel imaging geometry, this scaling factor is calculated as

$$s = \frac{\text{sum of measured projections at one view}}{\text{sum of forward projection of FBP reconstruction}}. \quad (11)$$

The scaling factor s is used to scale the FBP reconstruction so that the final image will have the same total image value sum as in the sum of the projection at each view. Due to noise, the projection view-sums are different for different views. An average view-sum can be used in evaluating the scaling factor.

III. Results

A. DC gain

A small simple (detector size $N = 8$) example is now used to illustrate the significance using a correct DC gain when performing ramp filtering in the Fourier domain. The 8-sample projection p is assumed to be

$$p = [0, 0, \frac{1}{2}, 1, 1, \frac{1}{2}, 0, 0] \quad (12)$$

and the ramp filtered results are listed in Table 1 using different methods. Only $N=8$ filtered values of $q(n)$, $n = 1, 2, \dots, 8$, are needed. Since $q(n)$ is symmetric, only $q(5)$ through $q(8)$ are listed in Table 1. The “Linear convolution” result is used as the gold standard.

Method 1—In Table 1, the “Linear convolution” result is obtained by convolving $p(n)$ in (12) with $ramp(n)$ in (3).

Method 2—In the “Circular convolution via FFT” method, the implementation steps are:

- First pad p in (12) with 8 zeros, resulting in a 16-sample projection

$$p_z = [0, 0, 0, 0, 0, 0, \frac{1}{2}, 1, 1, \frac{1}{2}, 0, 0, 0, 0, 0, 0]. \quad (13)$$

- Write the 16-element convolver from (3) in the form of

$$h = [\frac{1}{4}, \frac{-1}{\pi^2}, 0, \frac{-1}{(3\pi)^2}, 0, \frac{-1}{(5\pi)^2}, 0, \frac{-1}{(7\pi)^2}, 0, \frac{-1}{(7\pi)^2}, 0, \frac{-1}{(5\pi)^2}, 0, \frac{-1}{(3\pi)^2}, 0, \frac{-1}{\pi^2}]. \quad (14)$$

- Take the FFT of p_z in (13) obtaining P_z and take the FFT of h in (14) obtaining H .
- Multiply H and P_z element by element, and then take the inverse FFT (IFFT), obtaining $q(n)$. Select the central 8 elements and label them as $q(1)$ through $q(8)$.

Method 3—The “FFT with sampled ramp filter and $H(0) = 0$ ” method changes the H in Method 2 to

$$H_{ramp} = [0, 1, 2, 3, 4, 5, 6, 7, 8, 7, 6, 5, 4, 3, 2, 1]/16, \quad (15)$$

which is obtained by direct sampling of the continuous ramp filter.

Method 4—The only difference between the “FFT with sampled ramp filter and $H(0) = 0$ ” method and Method 3 is the DC gain $H(0)$. The DC gain in Method 3 is zero; it is the total sum of h in (14) for Method 4.

It is observed that Method 2, using (5) to calculate the discrete transfer function, gives exactly the same result as Method 1. Method 3, with direct sampling of the continuous ramp filter, produces errors. Method 4 reduces the DC shift error by using the correct DC gain from Method 2; however, some small errors still remain. Fortunately, the remaining small errors become even smaller as the size M increases. This point is shown in the following discussing and tables.

When M is chosen as an integer at least $2N-1$, (5) can be further written as

$$H(k) = \frac{1}{4} - \frac{2}{\pi^2} \sum_{\substack{n=1 \\ n=\text{odd}}}^{\frac{M}{2}-1} \frac{\cos \frac{2\pi nk}{M}}{n^2}. \quad (16)$$

Here $H(k)$ is even, is periodic with a period of M , and is the gain at frequency $\omega = k/M$. When using the DFT or FFT to implement the ramp filter, (16) is the correct formula to represent the ramp filter. Errors may occur if the sampled continuous ramp filter is used. The errors are expressed as

$$\text{Error}(\omega = \frac{k}{M}) = H(k) - \frac{|k|}{M}, \text{ for } \frac{|k|}{M} \leq 0.5. \quad (17)$$

Some numerical values according to (17) are listed in Tables 2 and 3 with $M = 2N$. It is observed from Tables 2 and 3 that if one uses the directly sampled ramp filter in DFT or FFT implementation, the DC gain needs to be corrected, while the errors for other frequency gains are small enough to ignore when the size M is large.

B. Post scaling

This section uses a computer simulated phantom to demonstrate the effectiveness of the post scaling method to improve the reconstruction accuracy. The parallel-beam imaging geometry is used to analytically generate projections with Poisson noise added. The number of views is 120 over 360° . There are 128 detector bins on the detector. The images are reconstructed in a 128×128 array with $M = 512$, where M is the length of the discrete convolution kernel used to evaluate the discrete filter as defined in (5). The results are shown in Table 4. These imaging parameters are typical values in our Siemens SPECT system, where the image resolution is approximately 1 cm which is a lot larger than the pixel size. The purpose of Table 4 is to compare the average image value in a large uniform region, and the image resolution is not considered.

The “Conventional FBP” method uses Method 4 defined in Part A of this section, without any window functions. The “Windowed FBP” methods uses the window function developed in [11], and is given as

$$WINDOW(\omega) = 1 - \left(1 - \frac{\alpha}{weight \times |\omega|}\right)^{iter}, \quad (18)$$

where $\alpha = 0.01$, $iter = 600$, and $weight$ = projection value of the current ray-sum. This window function models the Poisson noise in the projections. This window function changes from ray to ray. When $\omega = 0$, $WINDOW(0)$ is defined as 1.

When the filter is spatially variant, the Fourier domain filtering is carried out as follows. The spatially variant filter is quantized into 11 spatially invariant filters and apply these 11 filters to the projections, one filter at a time. Thus, 11 versions of the filtered projections are obtained. According to noise level of each ray, a filtered ray is selected from one of these 11 versions. At the end, only one backprojection is required to complete the FBP reconstruction.

The scaling factor in the “Scaled and windowed FBP” method is calculated according to (10). Due to noise, the value of the “sum of measured projections at one view” is chosen as the sum of all measurements divided by the number of views. The value of the “sum of FBP reconstruction” is sum of the 128×128 reconstruction array with all negative image values set to zero.

All images in Table 4 are displaying the same image value range of $[-0.1, 0.5]$, where -0.1 corresponds to black and 0.5 corresponds to white. The display gray scale is linear. The average value within the disc is labeled in the center of the image.

One last observation is that Method 4 gives large errors in Table 3 with $M = 16$, while the errors of Method 4 in Table 4 are invisible for $M = 512$ if no lowpass filter is applied. This observation verifies that the DC gain and low-frequency gain errors are visible only for a small M . The lowpass filter has a more significant effect on image quantitation for a large M .

IV. Discussion and Conclusions

This paper re-visits the well-known ramp filter, which is the driving engine in the FBP algorithm. This paper gives an exact formula (16) for the discrete ramp filter and also calculates the approximation error if one chooses to directly sample the continuous ramp filter. This common practice is then justified. An immediate result is that in order to reduce the errors, all one has to do is to pad more zeros (i.e., to increase M) to the projection data before transforming them into the Fourier domain.

Now the proposed approach is compared with that presented in [6]. The main difference is as follows. In Crawford’s paper, the discrete Fourier domain filter is obtained by sampling the continuous ramp filter. Since the corresponding spatial-domain convolution kernel has an infinite span, discrete sampling in the Fourier domain with any sampling interval will cause aliasing effects. Crawford’s paper presents an expression for the error this sampled filter has.

This current paper realizes that the span of the projection data is finite and the spatial-domain convolution kernel can be truncated without introducing any calculation errors. Thus an exact discrete filter function in the Fourier domain can be obtained by performing the Fourier transform of the truncated convolution kernel.

The explicit forms in (8) and (16) proposed in this paper are new. These results are equivalent to the sampled filter corrected by the error term that is proposed in [6]. The error term proposed in [6] is a summation with infinite terms and may not be accurately calculated in practice.

Another equivalent procedure is suggested in (eq. (68), page 74) Kak and Slaney's book [17], in which the sampled ramp filter is never used. Instead, the discrete Fourier transform of the convolution kernel is used. This approach is exact and it avoids sampling error caused by direct sampling of the continuous ramp filter.

A third new equation is given in (17), which is the discrepancy between the correct discrete Fourier domain ramp filter and the (incorrect) sampled ramp filter. Eq. (17) indicates that the errors become smaller as the filter size gets larger. Eq. (17) explains why the method of Stearns *et al* works [7].

This paper also investigates another factor that may affect the image accuracy. When a window function is introduced to suppress the noise and when the bandwidth of the window function is too narrow, some counts will be distributed outside the image array, causing a decreased total count within the *finite* image array and resulting in reduced image amplitude.

A simple remedy is suggested to improve the image accuracy. This remedy is to scale the reconstructed image with a scaling factor that is the ratio of the average per-view total count to the total count in the reconstructed image. This simple scaling method guarantees the total count conservation on the finite image array.

This is another important aspect that our viewpoint differs from others. When image quantitative errors occur, people automatically blame the errors on the DC gain of the filter. This is only partially justified. The lowpass filter used for noise suppression may affect image quantitation more, even though the DC gain of the lowpass filter is set to unity. This claim is supported by the results presented in our paper.

The post scaling method is not related to the DC gain problem; it is to correct for the error caused by a lowpass filter. In Table 1, there is no lowpass filter involved, and the post scaling method does not apply. The purpose of Table 1 is to show that Methods 1 and 2 are equivalent and they both give the exact true solution. However, if the ramp filter is directly sampled, the results are wrong. The errors cannot be fixed by simply using the correct $H(0)$ value; other values of H are still wrong.

This paper is not restricted to 2D parallel-beam imaging. The results can be extended to any imaging geometry. If the exact continuous Fourier domain transfer function $H_c(\omega)$ is given, its exact discrete counterpart $H(k)$ can be readily calculated according to (8). The post

scaling method can be applied to any reconstructed image, regardless the imaging geometry and the reconstruction algorithm.

Acknowledgments

This work was partially supported by the U.S. National Institute of Health under Grant NIH 1R01HL108350.

References

1. Bracewell RN. Strip integration in radio astronomy. *Aust J Phys.* 1956; 9:198–217.
2. Crowther RA, DeRosier DJ, Klug A. The reconstruction of a three-dimensional structure from its projections and its application to electron microscopy. *Proc Roy Soc London.* 1970; A317:319–340.
3. Ramachandran GN, Lakshminarayanan AV. Three-dimensional reconstructions from radiographs and electron micrographs: Application of convolution instead of Fourier transforms. *Proc Nat Acad Sci.* 1971; 68:2236–2240. [PubMed: 5289381]
4. Shepp LA, Logan BF. The Fourier reconstruction of a head section. *IEEE Trans Nucl Sci.* 1974; NS-21:21–43.
5. Rosenfeld, A.; Kak, AC. *Digital Picture Processing.* 2. New York, NY: Academic Press; 1982.
6. Crawford CR. CT filtration aliasing artifacts. *IEEE Trans Med Imag.* 1991; 10(1):99–102.
7. Stearns CW, Crawford CR, Hu H. Oversampled filters for quantitative volumetric PET reconstruction. *Phys Med Biol.* 1994; 39:381–387. [PubMed: 15551587]
8. Lathi, BP. *Signal Processing and Linear Systems.* Berkeley-Cambridge Press; Carnichael, California: 1998.
9. Hanson KM. On the optimality of the filtered backprojection algorithm. *J Computer Assisted Tomography.* 1980; 4:361–363.
10. Metz CE, Beck RN. Quantitative effects of stationary linear image processing on noise of resolution of structure images. *J Nucl Med.* 1974; 15:164–170. [PubMed: 4813223]
11. Chesler DA, Riederer SJ. Ripple suppression during reconstruction in transverse tomography. *Phys Med Biol.* 1975; 20:632–636. [PubMed: 1187793]
12. Hamming, RW. *Digital Filters.* Prentice Hall; Englewood Cliffs: 1977.
13. Butterworth S. On the theory of filter amplifiers. *Experimental Wireless and the Wireless Engineer.* 1930; 7:536–541.
14. Zeng GL, Zamyatin A. A filtered backprojection algorithm with ray-by-ray noise weighting. *Med Phys.* 40:031113. <http://dx.doi.org/10.1118/1.4790696> (7 pages), Online Publication Date: 28 February 2013. [PubMed: 23464293]
15. Shepp LA, Vardi Y. Maximum likelihood reconstruction for emission tomography. *IEEE Trans Med Imag.* Oct; 1982 1(2):113–122.
16. Vardi Y, Shepp LA, Kaufman L. A statistical model for positron emission tomography. *J Amer Stat Assoc.* 1985; 80:8–20.
17. Kak, AC.; Slaney, Malcolm. *Principles of Computerized Tomographic Imaging.* Society of Industrial and Applied Mathematics; 2001.

Table 1

Ramp filtering results with different methods

Methods	$q(5)$	$q(6)$	$q(7)$	$q(8)$
Linear convolution	0.0980	0.0180	-0.0619	-0.0133
Circular convolution via FFT	0.0980	0.0180	-0.0619	-0.0133
FFT with sampled ramp filter and $H(0) = 0$	0.0960	0.0160	-0.0640	-0.0155
FFT with sampled ramp filter and $H(0) = 0$	0.0984	0.0184	-0.0616	-0.0131

Table 2Errors for $\omega = 0$, $M = 2N$.

N	k	$H(k)$	Sampled ramp filter value	Error
8	0	1.26×10^{-2}	0	1.26×10^{-2}
16	0	6.32×10^{-3}		6.32×10^{-3}
32	0	3.17×10^{-3}		3.17×10^{-3}
64	0	1.58×10^{-3}		1.58×10^{-3}
128	0	7.92×10^{-4}		7.92×10^{-4}
256	0	3.96×10^{-4}		3.96×10^{-4}
512	0	1.99×10^{-4}		1.99×10^{-4}
1024	0	9.89×10^{-5}		9.89×10^{-5}
2948	0	4.95×10^{-5}		4.95×10^{-5}

Table 3

Errors for $\omega = 1/16 = 0.0625$, $M = 2N$.

N	k	$H(k)$	Sampled ramp filter value	Error
8	1	6.11×10^{-2}	6.25×10^{-2}	-1.41×10^{-3}
16	2	6.28×10^{-3}		2.50×10^{-4}
32	4	6.25×10^{-3}		3.64×10^{-5}
64	8	6.25×10^{-3}		4.79×10^{-6}
128	16	6.25×10^{-3}		6.07×10^{-7}
256	32	6.25×10^{-3}		7.61×10^{-8}
512	64	6.25×10^{-3}		9.52×10^{-9}
1024	128	6.25×10^{-3}		1.19×10^{-9}
2048	256	6.25×10^{-3}		1.49×10^{-10}

Table 4

FBP reconstruction of a uniform disc

Method	FBP reconstruction	Central profile
True
Conventional FBP
Windowed FBP
Scaled and windowed FBP