Filter design for filtered back-projection guided by the interpolation model

Stefan Horbelt, Michael Liebling, and Michael Unser

Biomedical Imaging Group, Swiss Federal Institute of Technology, CH-1015 Lausanne EPFL

ABSTRACT

We consider using spline interpolation to improve the standard filtered back-projection (FBP) tomographic reconstruction algorithm. In particular, we propose to link the design of the filtering operator with the interpolation model that is applied to the sinogram. The key idea is to combine the ramp filtering and the spline fitting process into a single filtering operation. We consider three different approaches. In the first, we simply adapt the standard FBP for spline interpolation. In the second approach, we replace the interpolation by an oblique projection onto the same spline space; this increases the peak signal-noise ratio by up to 2.5 dB. In the third approach, we perform an explicit discretization by observing that the ramp filter is equivalent to a fractional derivative operator that can be evaluated analytically for splines. This allows for an exact implementation of the ramp filter and improves the image quality by an additional 0.2 dB. This comparison is unique as the first method has been published only for degree n=0, whereas the two other methods are novel. We stress that the modification of the filter improve the reconstruction quality especially at low (faster) interpolation degrees ($n = 1$); the difference between the methods becomes marginal for cubic or higher degrees ($n \geq 3$).

Keywords: tomographic reconstruction, filtered back-projection, ramp filter, fractional splines, oblique projection, spline interpolation

1. INTRODUCTION

Mathematically, the measurement process in X-ray tomography is conveniently described by the Radon transform[1]. An attractive feature of this transform is that it has an exact inversion formula. The digital implementation of this analytical formula leads to the standard filtered back-projection (FBP) algorithm, which goes back to the early 70s[2]. Despite the considerable research efforts devoted to alternative reconstruction techniques (in particular, algebraic or statistical ones), the FBP is still the method of choice used in commercial CT scanners. It owes its success to the fact that it is direct, fast and reasonably simple to implement. Even though the standard implementation uses a rather rudimentary discretization—at least by modern standards, it has not been much improved over the years, except for the aspect of filter design[3]. The filtering part of the algorithm is essential to avoid excessive smoothing and to suppress aliasing; in practice, the design is mostly guided by practical considerations, such as noise reduction.

An important aspect of FBP is the interpolation step that takes place during the back-projection part of the algorithm. Most practical implementations use linear interpolation to minimize computations, but there is also evidence that the performance can be improved be using higher order models. In this paper, we are especially interested in the interpolation aspect of the problem and we want to determine the extent to which high quality methods can make a difference. As a novelty, we are proposing to modify the filtering step of the algorithm so that it is best matched to the interpolation that is applied to the sinogram. We have chosen here to concentrate on B-splines since these functions were found to offer the best cost-performance tradeoff for the interpolation of medical images[4,5]. The quality of the polynomial spline model is determined by its degree; for $n = 1$, it is equivalent to linear interpolation and it gets closer and closer to the band-limited model (sinc interpolation) as n increases. Another advantage of splines is that the effect of ramp filtering can be determined analytically; a property that will be exploited in the third method that is considered in this paper. We will essentially compare three methods—all based on spline interpolation; practically, they will differ only by the type of filter being used. Interestingly, we will see that we can outperform the standard approach by selecting a filter that is different from the traditional Ram-Lak (or ramp) filter.

2. STANDARD FBP

We recall the standard filtered back-projection (FBP), which is based on the inverse of the Radon transform[1].

The Radon transform $R_\theta f$ of an image $f(\vec{x})$, $\vec{x} \in R^2$, is the set of line integrals along the direction $\vec{\theta}$ at the distance t from the origin

$$R_\theta\{f(\vec{x})\} = R_\theta f(t) = \int_{\vec{x}\in\mathbb{R}^2} f(\vec{x})\delta(t - \vec{x}^\top \cdot \vec{\theta})d\vec{x}, \tag{1}$$

where $\delta(t)$ is the Dirac impulse and $\vec{\theta} = (\cos\theta, \sin\theta)^\top$ specifies the direction of integration.

The basis for the inverse Radon transform is the well-known identity

$$f(\vec{x}) = R^*(q * R_\theta\{f(\vec{x})\}), \tag{2}$$

where q denotes the 1D ramp filter whose Fourier transform is $\hat{q}(\omega) = |\frac{\omega}{2\pi}|$; R^*, the adjoint of R, is the back-projection operator:

$$(R^*p)(\vec{x}) = \int_0^\pi p(\vec{x}^\top \cdot \vec{\theta}, \theta)d\theta,$$

The widely used FBP algorithm corresponds to the direct discretization of the right-hand side of (2). However, instead of the infinite ramp filter q, one usually uses an attenuated version $\hat{h}(\omega) = |\frac{\omega}{2\pi}|\widehat{\Omega}(\omega)$, where $\widehat{\Omega}(\omega)$ is a suitable spectral window (e.g. Shepp-Logan filter).

3. FRACTIONAL B-SPLINES AND THEIR DERIVATIVES

The basic building blocks for spline interpolation are the B-spline basis functions. Here, we describe an extension for fractional degree α; not necessarily integer[6]. The symmetric B-spline of degree α is defined in the Fourier domain by

$$\hat{\beta}_*^\alpha(\omega) = \frac{\left|1 - e^{j\omega}\right|^{\alpha+1}}{|j\omega|^{\alpha+1}}. \tag{3}$$

This definition is essentially the same as the one for the classical B-splines of integer degree n[7]. The difference is that $\alpha = n$ is allowed to be fractional; i.e., non integer. Here, we are especially interested in the fractional derivative properties of these B-splines.

The n-th derivative of the function $f(x)$ can be defined in the Fourier domain as $D^n f(x) \leftrightarrow (j\omega)^n \hat{f}(\omega)$, where $\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(x)e^{-j\omega x}dx$ denotes the Fourier transform of $f(x)$. By extension, we define a symmetric version of fractional derivatives $D_*^\gamma f(x) \leftrightarrow |\omega|^\gamma \hat{f}(\omega)$, where γ is any fractional number. Note that this derivative only corresponds to the usual one when γ is even.

The relevance for our purpose of the D_* operator is that it is a scaled version of the ramp filter $\hat{q}(\omega) = |\frac{\omega}{2\pi}|$. The key property is that we have a simple analytical formula for the fractional derivative of a fractional B-spline

$$D_*^\gamma \beta_*(x) = \Delta_*^\gamma \beta_*^{\alpha-\gamma}(x), \tag{4}$$

where Δ_*^γ is the fractional finite difference operator

$$\Delta_*^\gamma \leftrightarrow |1 - e^{-j\omega}|^\gamma.$$

The argument for the proof is as follows

$$D_*^\gamma \beta_*(x) \leftrightarrow |j\omega|^\gamma \left|\frac{1 - e^{j\omega}}{j\omega}\right|^{\alpha+1} = \left|1 - e^{-j\omega}\right|^\gamma \left|\frac{1 - e^{j\omega}}{j\omega}\right|^{\alpha+1-\gamma}.$$

Spline interpolation amounts to fitting a sequence $f(k)$ with a spline of the form

$$f(x) = \sum_{k \in Z} c(k) \beta_*^\alpha (x - k).$$

The $c(k)$'s are determined by inverse filtering[9]:

$$c(k) = \left((b_*^\alpha)^{-1} * f \right)(k), \tag{5}$$

where

$$(b_*^\alpha)^{-1}(k) \leftrightarrow \frac{1}{\sum_k \beta_*^\alpha(k) e^{-j\omega k}} = \frac{1}{\sum_{l \in Z} |\mathrm{sinc}(\frac{\omega}{2\pi} + l)|^{\alpha+1}}.$$

4. SPLINE FBP

Next, we show how to modify the standard FBP so that the filtered sinogram can be fitted using splines. We will first describe the general principle of the method and then derive the filters that combine the spline fitting process (e.g. interpolation) and the ramp filter in one step. We will consider three different approaches.

4.1. Spline-based FBP: general principle

Here, we assume that the projection data $p_\theta(x)$ is known in a continuous fashion for $x \in R$, but for a discrete set of N equidistant angles $\theta_i = i \cdot \pi / N$.

The first step is to filter the sinogram $\hat{p}_\theta(\omega)$ in the Fourier domain with the ideal ramp filter (see Section 2):

$$\hat{g}_\theta(\omega) = \hat{p}_\theta(\omega) \cdot \hat{q}(\omega).$$

The second step is to fit the sinogram with a model that is represented as a linear combination of shifted basis functions $\varphi(t - k)$ (e.g. B-splines):

$$\tilde{g}_\theta(t) = \sum_{k \in Z} c_\theta(k) \cdot \varphi(t - k). \tag{6}$$

Typically, we have that $c_\theta(k) = \langle \tilde{\varphi}(t - k), g_\theta(t) \rangle$, where $\tilde{\varphi}$ is a suitable analysis function that is biorthogonal to $\varphi = \beta^n$ as described elsewhere[7]. In our implementation, the $c_\theta(k)$'s will be computed from the $p_\theta(k)$'s by filtering in the Fourier domain using FFTs:

$$c_\theta(k) = (h * p_\theta)(k) \leftrightarrow H(e^{j\omega}) \cdot P_\theta(e^{j\omega}).$$

What will make the difference between the three methods below is the choice of the filter $h \leftrightarrow H(e^{j\omega})$.

The last step is to calculate the back-projection $R^*\{\tilde{g}_\theta(t)\}$ at the pixel location (x, y)

$$\tilde{f}(x, y) = \int_0^\pi \tilde{g}_\theta((x, y) \cdot \vec{\theta}) d\theta \cong \frac{\pi}{N} \sum_{i=1}^{N} \tilde{g}_{\theta_i}(x \cos\theta_i + y \sin\theta_i),$$

where the right-hand side is the Riemann-sum approximation of the back-projection integral. This approximation is justifiable as long as the number N of projections is sufficiently large (typ., twice the size of the image). The arguments of the sum are computed using the 1D spline interpolation model (6).

4.2. Ramp filter with B-spline prefilter

In the first method, we compute the B-spline coefficients $c_{\theta_i}(k)$ such that the function $\tilde{g}_\theta(t)$ interpolates the integer samples of the filtered sinogram. This involves the application of the digital interpolation filter (5), which can be merged with the ramp-filtering step of the algorithm.

Here, the basic assumption is that the sinogram is band-limited; in this case, the ramp-filtered sinogram is band-limited as well, and represented by its samples. Thus, to get the B-spline coefficients of the filtered sinogram, these samples needs to be filtered with the prefilter $\widehat{\Omega}_1(\omega)$:

$$\widehat{\Omega}_1(\omega) = 1/B^n(e^{j\omega}) = 1/\sum_{l\in Z}\Big(\mathrm{sinc}(\frac{\omega}{2\pi}+l)\Big)^{n+1}. \tag{7}$$

We combine the prefilter $\widehat{\Omega}_1(\omega)$ with the ramp filter, and get the B-spline interpolating ramp filter

$$H_1(e^{j\omega}) = |\frac{\omega}{2\pi}|\widehat{\Omega}_1(\omega).$$

4.3. Ramp filter with oblique projection

Again, we assume that the sinogram is band-limited, which implies that the filtered sinogram is band-limited as well. Instead of B-spline interpolation, we use an oblique projection to get a continuous spline approximation of the sinogram. Because the computation needs to be performed in Fourier space, the projection is chosen to be perpendicular to the subspace of band-limited functions.

This projection can be derived as a direct application of Theorem 2 in the work of Unser and Aldroubi[8]. Specifically, the oblique projection of the filtered sinogram $g_\theta(t)$ is given by

$$\mathrm{Proj}\{g_\theta(t)\} = \sum_{k\in Z}\underbrace{c_1 * (a_{12})^{-1}(k)}_{=c_\theta(k)}\cdot\beta^n(x-k),$$

where $c_1(k) = \langle g_\theta(t), \mathrm{sinc}(t-k)\rangle \leftrightarrow |\frac{\omega}{2\pi}|P_\theta(e^{j\omega})$, and where a_{12} (the cross-correlation between the analysis and synthesis functions) is

$$a_{12}(k) = \langle \mathrm{sinc}(x), \beta^n(x-k)\rangle$$
$$\leftrightarrow A_{12}(e^{j\omega}) = \sum_{k\in Z}\mathrm{rect}(\omega+2\pi l)\cdot\hat{\beta}^n(\omega+2\pi k).$$

The effect of the rect function is to suppress aliasing so that we have

$$A_{12}(e^{j\omega}) = \hat{\beta}^n(\omega)\,\mathrm{for}\,-\pi\le\omega\le\pi.$$

Combining the ramp filter and the projection filter $A_{12}^{-1}(e^{j\omega})$ in the Fourier domain, we get the oblique B-spline ramp-filter

$$H_2(e^{j\omega}) = |\frac{\omega}{2\pi}|/\widehat{\beta}^n(\omega) = |\frac{\omega}{2\pi}|/\mathrm{sinc}^{n+1}(\frac{\omega}{2\pi}).$$

We can already predict that this projection approach will be better than the more standard interpolation described in Section 4.2. It is essentially equivalent to the least-squares solution: first, because it is guaranteed to be asymptotically optimal; and second, because the angle between the spline and the Sinc spaces is small, especially for higher degrees n.

Filters	Frequency response $H(e^{j\omega})$ for $0 \leq \omega \leq 2\pi$
Shepp−Logan[3]	$h_0(k) \leftrightarrow \|\omega/(2\pi)\|\text{sinc}(\omega)$
Inter−polation	$h_1(k) \leftrightarrow \frac{\|\omega/(2\pi)\|}{B^n(e^{j\omega})} = \frac{\|\omega/(2\pi)\|}{\sum_{l\in Z}\text{sinc}^{n+1}(\frac{\omega}{2\pi}+l)}$
Oblique	$h_2(k) \leftrightarrow \frac{\|\omega/(2\pi)\|}{\text{sinc}^{n+1}(\frac{\omega}{2\pi})}$
Fractional	$h_3(k) \leftrightarrow \frac{\|1-e^{j\omega}\|/(2\pi)}{B_*^{\alpha}(e^{j\omega})} = \frac{\|\sin(\omega/2)\|/\pi}{\sum_{l\in Z}\|\text{sinc}(\frac{\omega}{2\pi}+l)\|^{\alpha+1}}$

Table 1: Modified ramp filters for splines. These filters project onto the B-spline space of degree n or $\alpha - 1$.

4.4. Ramp filter with fractional B-splines

In the two cases before (4.2 and 4.3), the band limitation assumption was necessary to justify the multiplication with the theoretical ramp filter $|\frac{\omega}{2\pi}|$ in the FFT domain. We now consider a third approach which does not require this hypothesis. The idea is to fit the sinogram with a fractional spline, and then determine the effect of the ramp filter analytically.

First, the unfiltered sinogram $p_\theta(t)$ is fitted using fractional B-splines of degree α:

$$p_\theta(t) = \sum_{k\in Z} d_\theta(k)\beta_*^{\alpha}(t-k). \tag{8}$$

This is achieved with the digital prefiltering technique described by (5).

Since the ramp filter corresponds to the fractional derivative operator D_*, we can use our theoretical formula (4) to differentiate (8) analytically. This allows to apply the ramp filtering step directly onto the fitted sinogram (8):

$$g_\theta(t) = D_*\{p_\theta(t)\}/(2\pi) = \sum_{k\in Z} c_\theta(k)\beta_*^{\alpha-1}(t-k), \tag{9}$$

which reduces the spline degree by one. We can obtain the spline coefficients $c_\theta(k)$ of the filtered sinogram in one step by applying a single digital filter which combines the fitting and finite differences operations:

$$c_\theta(k) = (h_3 * p_\theta)(k).$$

$$H_3(e^{j\omega}) = \frac{|1-e^{j\omega}|/(2\pi)}{B_*^{\alpha}(e^{j\omega})} = \frac{|\sin(\omega/2)|/\pi}{\sum_{l\in Z}|\text{sinc}(\frac{\omega}{2\pi}+l)|^{\alpha+1}}.$$

If the fractional spline degree α is even ($\alpha = 2, 4, \ldots$), then the B-spline in (9) will be of odd degree $\alpha - 1 = n = 1, 3, \ldots$, and the basis functions is compactly supported. This means that the same back-projection method as before is applicable.

5. RESULTS

The Shepp-Logan phantom[3] of size 128×128 was used as our test image. Its Radon transform was computed over $K = 256$ equidistant angles. Figures 1 and 2 display the reconstructed images (with details) and the errors for the three methods for linear and cubic degrees, respectively. The FBP reconstruction error with the Shepp-Logan filter and linear interpolation is 29.16 dB. With linear splines ($n = 1$), the FBP reconstruction errors are 30.98 dB (interpolation filter), 32.91 dB (oblique projection filter), and 33.10 dB (fractional derivative filter).

Filter type \ degree	$n = 1$	$n = 3$
Shepp-Logan	29.16	32.49
B-spline interpolation	30.98	34.69
Oblique projection	32.91	34.80
Fractional derivative	33.10	34.90

Table 2. FBP reconstruction error (PSNR given in dB) for different spline interpolating ramp filters with various interpolation degrees n. The image size is $N = 128$, the angular resolution is $K = 256$ and the sampling step on the sinogram is 1.

Note that the first case here also corresponds to the standard approach: Ram-Lak filter with linear interpolation since $B^1(e^{j\omega}) = 1$. With cubic splines ($n = 3$), the corresponding results are 32.49 dB (Shepp-Logan filter), 34.69 dB (interpolation filter), 34.80 dB (oblique projection filter), and 34.90 dB (fractional derivative filter). See Table 2.

When looking at the error images, the edges are sharper for the interpolation filter and more diffuse for the oblique projection. Remember that the less visible the error, the better the result. The fractional derivative filter is slightly better than the two others. These results demonstrate that there is a clear improvement when using ramp filters that adapt to the interpolation method. Further, it appears that the optimization of the filter is especially useful when the interpolation model is low. This is especially interesting for applications where computational speed is a key issue.

6. CONCLUSION

We compared three different versions of the filtered back-projection based on spline interpolation. The first approach uses interpolation combining the ramp and direct B-spline filter in to a single FFT filtering operation. The second implements an oblique projection of the sinogram into a spline space. It outperforms the first by up to 2 dB, because it better takes into account the band-limited nature of the sinogram. The third filter is based on an explicit discretization using fractional splines. The advantage is that the ramp-filtering of the sinogram produces a spline of reduced degree, which leads to an implementation perfectly coherent with the underlying model. This yields another gain in quality for the same computational cost.

In conclusion, for low interpolation degrees ($n = 1$) we recommend to use either the method based on the oblique projection or fractional splines in order to achieve a quality improvement of up to 2.5 dB. Switching to higher degree—cubic instead of linear—pays off by an additional 2-4 dB in all three cases, but the quality differences between the methods are less significant.

A software demo (JAVA applet) of the methods developed in this paper is available at:
`http://bigwww.epfl.ch/demo/jtomography/`

REFERENCES

1. D. Ludwig, "The Radon transform on Euclidean space," *Comm. Pure and Appl. Math.,* vol.19, pp.49-81, 1966.
2. G. N. Ramachandran and A. V. Lakshminarayanan, "3D reconstructions from radiographs and electron micrographs: Appl. of convolution instead of Fourier trans.," *Proc. Nat. Acad. Sci.,* vol.68, pp.2236-40, 1971.
3. L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nucl. Sci.,* vol.21, pp.21-43, 1974.
4. E. H. W. Meijering, "Spline Interpolation in Medical Imaging: Comparison with other convolution-based approaches," *EUSIPCO*, vol. 4, pp. 1989-1996, 2000.
5. P. Thévenaz, T. Blu, M. Unser, "Interpolation revisited," *IEEE Trans. MI*, vol. 9, no. 17, pp. 739-758, 2000.
6. M. Unser, T. Blu, "Fractional Splines and Wavelets," *SIAM Review*, vol. 42, no. 1, pp. 43-67, March 2000.
7. M. Unser, "Sampling—50 Years After Shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569-587, 2000.
8. M. Unser, A. Aldroubi, "A general sampling theory for nonideal acquisition devices", Signal Processing, IEEE Trans. on Signal Processing, vol. 42, no. 11 , pp. 2915-2925, Nov. 1994.
9. M. Unser, "Splines: A Perfect Fit for Signal and Image Processing," *IEEE Signal Processing Magazine*, vol. 16, no. 6, pp. 22-38, Nov. 1999.

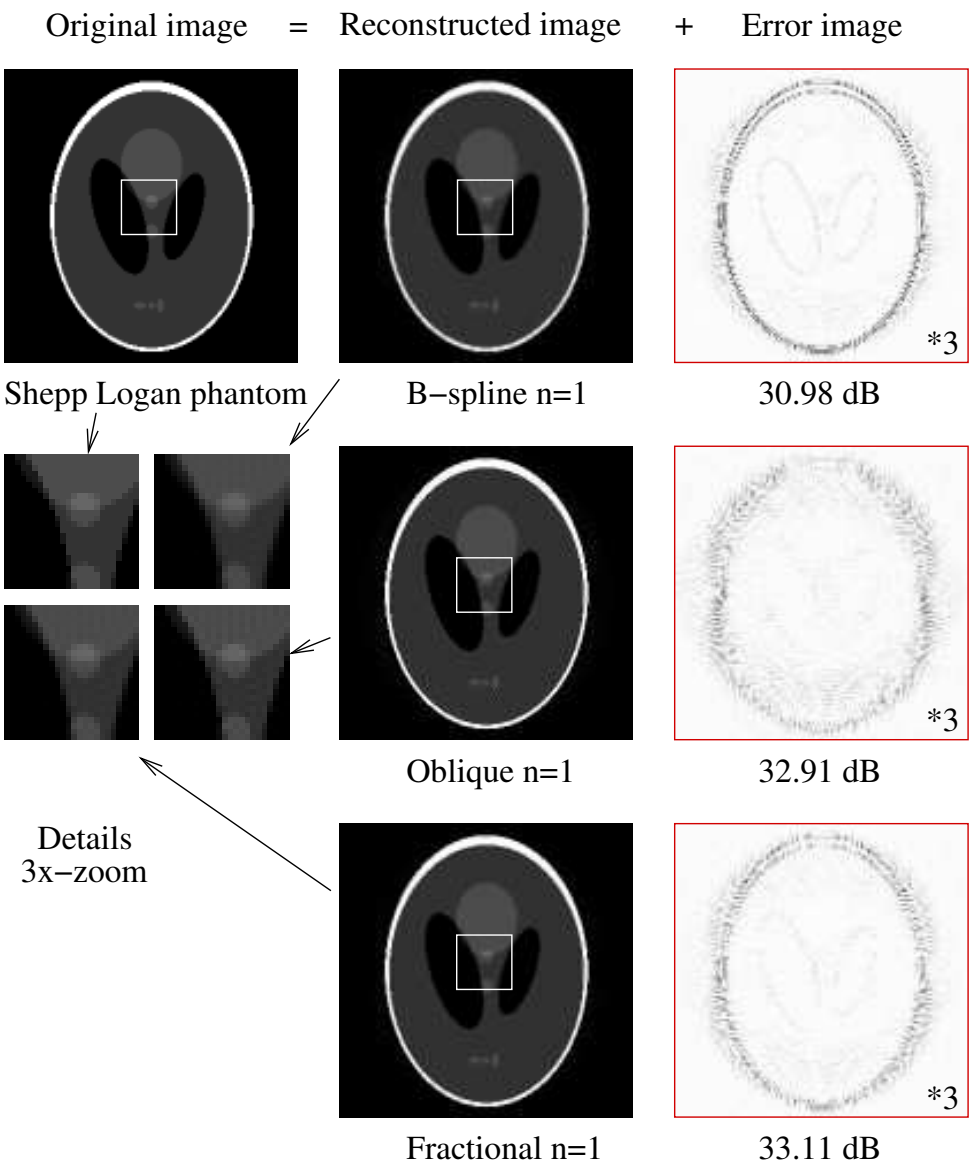

Figure 1. Results of Spline-filtered back-projection with linear interpolation model (degree $n = 1$) and the three different ramp filters. To highlight differences, the absolute error images have been amplified by a factor of 3.

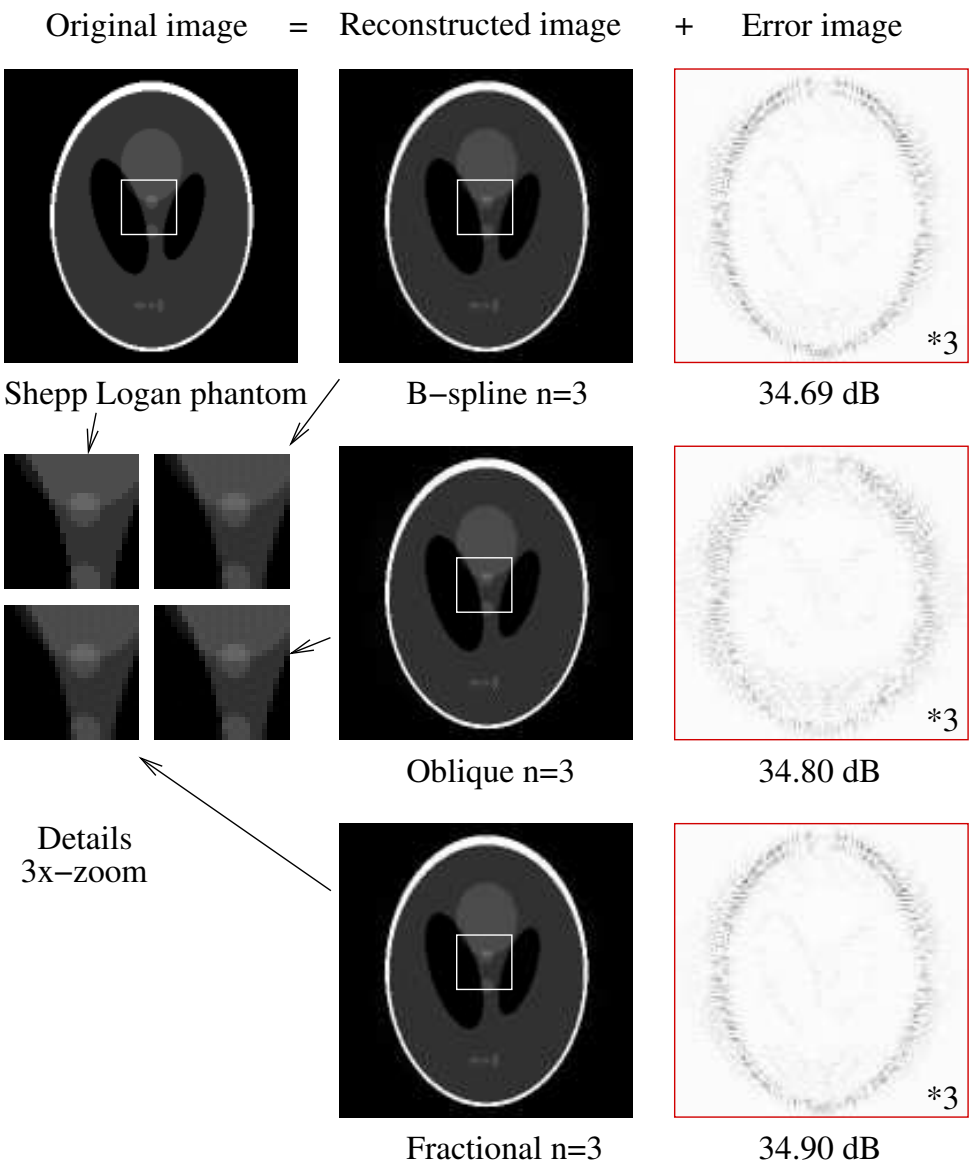

Figure 2. Results of Spline-filtered back-projection with cubic interpolation model (degree $n = 3$) and three different ramp filters. To highlight differences, the absolute error images have been amplified by a factor of 3.