Chapter 25

Tomographic system geometries

ap,geom

Contents

25.1 **Introduction** (s,geom,intro)

This chapter describes some details for a few tomographic imaging system geometries. These details are important for implementing efficient forward and backprojection operations.

Numerous papers have described tomographic system models and associated forward- and back-projection methods. This chapter only scratches the surface.

s,geom,line,perp

Distances between a point and a line are useful in deriving projectors. In Fig. 25.1.1, one can show that

$$\tau = \frac{-(x_0 - x_1)(y_2 - y_1) + (y_0 - y_1)(x_2 - x_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}$$

$$\tau_\perp = \frac{(x_0 - x_1)(x_2 - x_1) + (y_0 - y_1)(y_2 - y_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}.$$
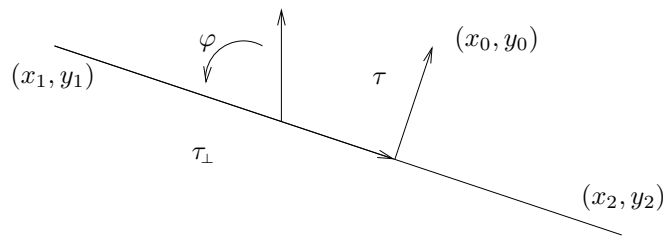
fig,geom,line,perp

Figure 25.1.1: Distance between a point and a line.

25.2 Parallel-beam 2D system models (s,geom,par2)

Consider the 2D object model based on a common basis function $\beta_0(x, y)$ superimposed on a $N \times M$ Cartesian grid as follows:

$$f(x, y) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f[n, m] \, \beta_0\left(\frac{x - x_{\mathrm{c}}[n]}{\Delta_{\mathrm{X}}}, \frac{y - y_{\mathrm{c}}[m]}{\Delta_{\mathrm{Y}}}\right), \tag{25.2.1}$$

e,geom,par2,fxy

where $(x_{\mathrm{c}}[n], y_{\mathrm{c}}[m])$ denotes the center of the basis function for $f[n, m]$. Typically

$$
\begin{aligned}
x_{\mathrm{c}}[n] &= (n - w_{\mathrm{X}})\Delta_{\mathrm{X}}, \quad n = 0, \ldots, N - 1 \\
y_{\mathrm{c}}[m] &= (m - w_{\mathrm{Y}})\Delta_{\mathrm{Y}}, \quad m = 0, \ldots, M - 1 \\
w_{\mathrm{X}} &= (N - 1)/2 + c_{\mathrm{X}} \\
w_{\mathrm{Y}} &= (M - 1)/2 + c_{\mathrm{Y}},
\end{aligned}
\tag{25.2.2}
$$

e,geom,par2,xon

where the user-selectable parameters $(c_{\mathrm{X}}, c_{\mathrm{Y}})$ denote a spatial offset for the image center. These are useful when the object is not positioned in the center of the field-of-view. For simplicity, we assume that $\Delta_{\mathrm{Y}} = \pm\Delta_{\mathrm{X}}$, and that $\beta_0(x, y)$ is symmetric in both arguments, *i.e.*, $\beta_0(\pm x, \pm y) = \beta_0(x, y)$. (The purpose of a negative Δ_{X} or Δ_{Y} value is to permit mirror reversals of the coordinate system.)

Consider *parallel-beam* or *parallel-ray* system geometries that have the property that for a given projection angle φ, the radial samples have equal spacing Δ_{R}, so that, *cf.* (4.3.1), we can write

$$\bar{y}_{\varphi}[i_{\mathrm{R}}] = h_{\varphi}(r) * p_{\varphi}(r)\Big|_{r=(i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}}} = \int h_{\varphi}((i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}} - r) \, p_{\varphi}(r) \, \mathrm{d}r, \tag{25.2.3}$$

e,geom,par2,yb

for $i_{\mathrm{R}} = 0, \ldots, n_{\mathrm{R}} - 1$, where $p_{\varphi}(r)$ denotes the Radon transform of $f(x, y)$. Note that for generality we allow the blur PSF $h_{\varphi}(r)$ to vary with projection view φ, although often it does not. The offset w_{R} is defined akin to w_{X} and w_{Y}, *e.g.*, $w_{\mathrm{R}} \triangleq (n_{\mathrm{R}} - 1)/2 + c_{\mathrm{R}}$. Typically the offset c_{R} is zero or $\pm 1/4$, the latter reflecting the *quarter detector offset* that is used in some CT systems to improve sampling, *e.g.*, [1–4].

The next subsection considers the specific case of a *strip-integral* model where $h_{\varphi}(r) = \frac{1}{w} \mathrm{rect}\left(\frac{r}{w}\right)$, where w denotes the detector width.

Let $g(t, \varphi)$ denote the Radon transform of $\beta_0(x, y)$, *i.e.*,

$$g(t, \varphi) \triangleq \int_{\mathcal{L}(t, \varphi)} \beta_0(x, y) \, \mathrm{d}\ell = \int_{-\infty}^{\infty} \beta_0(t \cos\varphi - \ell \sin\varphi, \ t \sin\varphi + \ell \cos\varphi) \, \mathrm{d}\ell. \tag{25.2.4}$$

e,geom,par2,fbasisradon

Then by the linearity, shift, and scaling properties of the Radon transform we have

$$p_{\varphi}(r) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f[n, m] \, \Delta_{\mathrm{X}} \, g\left(\frac{r - r_{\varphi}[n, m]}{\Delta_{\mathrm{X}}}, \varphi\right), \tag{25.2.5}$$

e,geom,par2,pangr

where

$$r_{\varphi}[n, m] \triangleq x_{\mathrm{c}}[n] \cos\varphi + y_{\mathrm{c}}[m] \sin\varphi. \tag{25.2.6}$$

e,geom,par2,rangnm

Substituting into (25.2.3) yields the linear model

$$\bar{y}_{\varphi}[i_{\mathrm{R}}] = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a[i_{\mathrm{R}}, \varphi; n, m] \, f[n, m], \tag{25.2.7}$$

e,geom,par2,yb,a,f

for $i_{\mathrm{R}} = 0, \ldots, n_{\mathrm{R}} - 1$, where the elements of the system matrix are given by[1]

$$
\begin{aligned}
a[i_{\mathrm{R}}, \varphi; n, m] &= \int h_{\varphi}((i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}} - r) \, \Delta_{\mathrm{X}} \, g\left(\frac{r - r_{\varphi}[n, m]}{\Delta_{\mathrm{X}}}, \varphi\right) \mathrm{d}r \\
&= \int h_{\varphi}((i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}} - r_{\varphi}[n, m] - r') \, \Delta_{\mathrm{X}} \, g(r'/\Delta_{\mathrm{X}}, \varphi) \, \mathrm{d}r' \\
&= F_{\varphi}\Big((i_{\mathrm{R}} - \tau_{\varphi}[n, m]) \, \Delta_{\mathrm{R}}; \Delta_{\mathrm{X}}\Big),
\end{aligned}
\tag{25.2.8}
$$

e,geom,par2,aainm,Fangr

---

[1] Note that for a sinogram with n_{R} radial bins and projection angles $\{\varphi_k\}$, the relation between the "linear algebra" notation a_{ij} and the notation in (25.2.8) is $a_{ij} = a[i \bmod n_{\mathrm{R}}, \varphi_{\lfloor i/n_{\mathrm{R}} \rfloor}; n(j), m(j)]$, where $n(j)$ and $m(j)$ were defined in (1.4.16).

where $r' = r - r_\varphi[n, m]$ and we define

$$\tau_\varphi[n, m] \triangleq w_{\mathrm{R}} + r_\varphi[n, m]/\Delta_{\mathrm{R}}, \tag{25.2.9}$$

and

$$F_\varphi(r; \Delta_{\mathrm{x}}) \triangleq \int h_\varphi(r - r')\, \Delta_{\mathrm{x}}\, g(r'/\Delta_{\mathrm{x}}, \varphi)\, \mathrm{d}r' = h_\varphi(r) *_{\Delta_{\mathrm{x}}} g(r/\Delta_{\mathrm{x}}, \varphi). \tag{25.2.10}$$

For simple basis functions and PSFs, one can determine the "blurred footprint" function(s) $F_\varphi(r; \Delta_{\mathrm{x}})$ analytically. For more complicated basis functions and PSFs, one can precompute the integral that defines $F_\varphi(r; \Delta_{\mathrm{x}})$ using very fine discretizations, and tabulate the results for a suitable range of arguments. Then each system matrix element corresponds simply to looking up in the appropriate table (one for each angle φ) for the desired argument.

For example, suppose we consider the pixel basis but use a (truncated) gaussian PSF. Each $g(t, \varphi)$ is a trapezoid, and a different trapezoid for each angle. So $F_\varphi(r; \Delta_{\mathrm{x}})$ is essentially the convolution of a trapezoid with a gaussian. The analytical form for that convolution is cumbersome, but such a convolution is very easy computed numerically to high accuracy and then tabulated, *e.g.*, [5].

25.2.1 Local bases

Practical models use local basis functions $\beta_0(x, y)$ having finite support and finite width PSF models $h_\varphi(r)$. For such local models, the blurred footprint function $F_\varphi(r; \Delta_{\mathrm{x}})$ is nonzero only the finite support interval $(-r_{\max}(\varphi), r_{\max}(\varphi))$. In such typical cases, most of the system matrix elements in (25.2.8) are zero. To determine which elements are nonzero for a given pixel location $[n, m]$ and projection view angle φ, *i.e.*, to determine the *bin support*, we must find the i_{R} values for which

$$-\tau_{\max}(\varphi) < i_{\mathrm{R}} - \tau_\varphi[n, m] < \tau_{\max}(\varphi),$$

where

$$\tau_{\max}(\varphi) \triangleq r_{\max}(\varphi)/\Delta_{\mathrm{R}}, \tag{25.2.11}$$

or equivalently:

$$\tau_\varphi[n, m] - \tau_{\max}(\varphi) < i_{\mathrm{R}} < \tau_\varphi[n, m] + \tau_{\max}(\varphi).$$

Using these bounds avoids unnecessary evaluations of $F_\varphi(r; \Delta_{\mathrm{x}})$. In particular, a loop over i_{R} values should cover

$$i_{\mathrm{R}}^{\min}(\tau_\varphi[n, m]) \le i_{\mathrm{R}} \le i_{\mathrm{R}}^{\max}(\tau_\varphi[n, m]), \tag{25.2.12}$$

where

$$\begin{aligned} i_{\mathrm{R}}^{\min}(\tau) &\triangleq \lfloor \tau - \tau_{\max}(\varphi) \rfloor + 1 \\ i_{\mathrm{R}}^{\max}(\tau) &\triangleq \lceil \tau + \tau_{\max}(\varphi) \rceil - 1. \end{aligned}$$

By (27.9.1), the number of nonzero i_{R} values in the interval (25.2.12) satisfies

$$1 + i_{\mathrm{R}}^{\max} - i_{\mathrm{R}}^{\min} \le \lceil 2\, \tau_{\max}(\varphi) \rceil. \tag{25.2.13}$$

25.2.2 Tabulation

One approach is to tabulate exhaustively all of the nonzero values of $F_\varphi(r; \Delta_{\mathrm{x}})$ needed for forward projection [6]. This is feasible for small problems, but for large problems the required memory can be excessive.

For reducing memory, the standard approach to tabulating $F_\varphi(r; \Delta_{\mathrm{x}})$ would be to evaluate it for finely spaced values of its argument, and then use either nearest neighbor or linear interpolation to compute (25.2.8). A disadvantage of this approach is that it requires nonsequential access of the table values as one indexes over i_{R}, which may degrade cache performance. In light of (25.2.13), for more sequential access of the table, first define

$$K = \max_\varphi \lceil 2\, \tau_{\max}(\varphi) \rceil,$$

and choose an over-sampling factor L. Then for a given view angle φ, precompute samples of $F_\varphi(r; \Delta_\mathrm{x})$ as follows[2]:

$$F[k, l] \triangleq F_\varphi\left(\left[i_\mathrm{R}^{\min}\left(\frac{l + \frac{1}{2}}{L}\right) + k - \left(\frac{l + \frac{1}{2}}{L}\right)\right]\Delta_\mathrm{R}; \Delta_\mathrm{x}\right), \qquad \begin{matrix} k = 0, \ldots, K - 1 \\ l = 0, \ldots, L - 1. \end{matrix} \qquad \text{(25.2.14)}$$

<div align="right">e,geom,par2,Fkl</div>

Then for $k = 0, \ldots, K - 1$, approximate $F_\varphi(r; \Delta_\mathrm{x})$ by its nearest tabulated value as follows:

$$F_\varphi\left(\left[i_\mathrm{R}^{\min}(\tau) + k - \tau\right]\Delta_\mathrm{R}; \Delta_\mathrm{x}\right) \approx F[k, l(\tau)],$$

where

$$l(\tau) \triangleq \lfloor(\tau - \lfloor\tau\rfloor)L\rfloor \in \{0, \ldots, L - 1\}.$$

For sequential access, we store $F[k, l]$ so that k varies fastest.

IRT *See* `Gtomo2_table.m`.

25.2.3 Incrementalism

To compute $\tau_\varphi[n, m]$ efficiently, note that the following recursion holds:

$$\tau_\varphi[n + 1, m] = \tau_\varphi[n, m] + \left(\frac{\Delta_\mathrm{x}}{\Delta_\mathrm{R}}\cos\varphi\right),$$

which is known as an *incremental* approach since we can precompute the parenthesized value, *e.g.*, [7, 8].

25.2.4 Computation analysis

Consider a typical $N \times N$ image and a single projection view with $n_\mathrm{R} = N$ samples. Computing the forward projection (25.2.7) for a single angle φ using the above tabulation approach requires $2KN^2$ flops.

Because the table (25.2.14) is finite, the above tabulation approach is equivalent mathematically to a generalization of the approach proposed by Schwinger *et al.* [9]. In that method one first projects each pixel value $f[n, m]$ onto the nearest neighbor in a finely spaced projection array, then convolves that finely spaced projection array with the footprint function $F_\varphi(r; \Delta_\mathrm{x})$, and then downsamples the filtered projection array to the sample spacing of the measurements. For an efficient implementation of that method, the convolution and down-sampling steps should be combined into a single step. The accuracy of this approach will be identical to our tabulation method if the over-sampling factor is the same L used in (25.2.14). For such an implementation, the flops required per projection view are $N^2 + 2LKN = (1 + 2LK/N)N^2$. The footprint tabulation approach is more efficient if $L/N > 1 - \frac{1}{2K}$.

<div style="font-size:small">s,geom,par2,aa</div>

25.2.5 Local angle approximation (s,geom,par2,aa)

It may be useful to make approximations to the "blurred footprint" function $F_\varphi(r; \Delta_\mathrm{x})$ in (25.2.10), *e.g.*, to reduce memory when tabulating or to reduce computation time.

Let $q(\varphi)$ denote a smooth function of φ, such as $q(\varphi) = g(0, \varphi)$. If $\varphi \approx \theta$, then

$$g(t, \varphi) \approx \frac{q(\varphi)}{q(\theta)}g\left(\frac{q(\varphi)}{q(\theta)}t, \theta\right).$$

<div align="right">e,geom,par2,aa,g,approx
(25.2.15)</div>

Note that both sides of this approximation integrate to the same value.

In (25.2.10), suppose that $h_\varphi(r) \approx h_\theta(r)$ for $\varphi \approx \theta$. (This holds exactly when $h_\varphi(r)$ is independent of φ.) Then we have the following approximation to the blurred footprint:

$$\varphi \approx \theta \implies F_\varphi(r; \Delta_\mathrm{x}) = \int h_\varphi(r - r')\Delta_\mathrm{x} g\left(\frac{r'}{\Delta_\mathrm{x}}, \varphi\right)\mathrm{d}r' \qquad \begin{matrix}\text{e,geom,par2,aa,g,approx}\\ \text{(25.2.16)}\end{matrix}$$

$$\approx \int h_\theta(r - r')\Delta_\mathrm{x}\frac{q(\varphi)}{q(\theta)}g\left(\frac{q(\varphi)}{q(\theta)}\frac{r'}{\Delta_\mathrm{x}}, \theta\right)\mathrm{d}r' \qquad \text{(25.2.17)}$$

$$= \left(\frac{q(\varphi)}{q(\theta)}\right)^2 F_\theta\left(r; \Delta_\mathrm{x}\frac{q(\theta)}{q(\varphi)}\right). \qquad \begin{matrix}\text{e,geom,par2,aa,F,approx}\\ \text{(25.2.18)}\end{matrix}$$

If the basis function $\beta_0(x, y)$ in (25.2.1) is circularly symmetric, then we can choose $q(\varphi) = 1$ and (25.2.18) holds exactly. Otherwise, a reasonable choice is $q(\varphi) = g(0, \varphi)$. See §25.2.10.

---

[2]The factor of 1/2 is to find the center value in an even number of intervals.

25.2.6 Mojette sampling (s,geom,par2,moj)

The tabulation approach becomes particularly simple *if* we can choose the radial sampling to be related to the projection angle φ as follows:

$$\Delta_{\mathrm{R}} = \frac{1}{L} |\Delta_{\mathrm{x}}| \max(|\cos\varphi|, |\sin\varphi|), \qquad\qquad (25.2.19)$$

where $L \in \mathbb{N}$ is an over-sampling factor, and we continue to assume that $|\Delta_{\mathrm{Y}}| = |\Delta_{\mathrm{x}}|$. This choice of sample spacing dates back many years and has been rediscovered many times [10–16]; recently it is associated with the *Mojette transform* [17–20]. See also the related *linogram* sampling in §3.8. Fig. 25.2.1 illustrates the radial spacing for a certain angle φ.
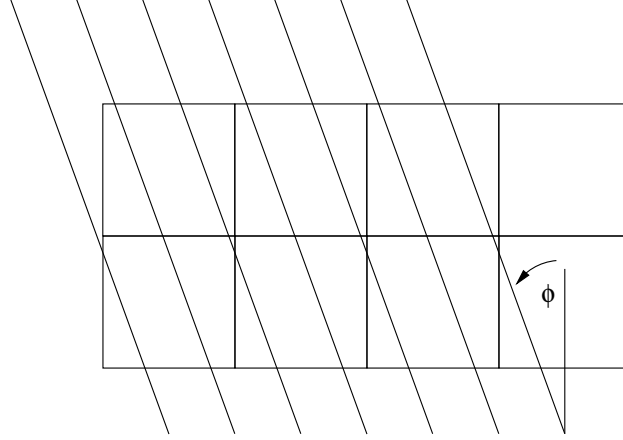
Figure 25.2.1: Illustration of Mojette radial sample spacing.

To illustrate why such radial sample spacing is convenient, consider the case where $|\cos\varphi| \geq |\sin\varphi|$ and $\Delta_{\mathrm{x}} \cos\varphi > 0$, for which $\Delta_{\mathrm{R}} = \frac{1}{L}\Delta_{\mathrm{x}} \cos\varphi$. In this case, the system matrix elements (25.2.8) simplify as follows:

$$F_\varphi\Big((i_{\mathrm{R}} - \tau_\varphi[n,m]) \Delta_{\mathrm{R}}; \Delta_{\mathrm{x}} \Big) = F_\varphi\Big((i_{\mathrm{R}} - Ln - \tilde{\tau}_\varphi[m]) \frac{1}{L}\Delta_{\mathrm{x}} \cos\varphi; \Delta_{\mathrm{x}} \Big), \qquad (25.2.20)$$

where we define

$$\tilde{\tau}_\varphi[m] \triangleq w_{\mathrm{R}} - Lw_{\mathrm{x}} + L(m - w_{\mathrm{Y}})\frac{\Delta_{\mathrm{Y}} \sin\varphi}{\Delta_{\mathrm{x}} \cos\varphi}. \qquad\qquad (25.2.21)$$

Note that $\tilde{\tau}_\varphi[m+1] = \tilde{\tau}_\varphi[m] + L\frac{\Delta_{\mathrm{Y}} \sin\varphi}{\Delta_{\mathrm{x}} \cos\varphi}$. The attractive property of (25.2.20) is that it depends on $i_{\mathrm{R}} - n$ when $L = 1$, which we assume hereafter[3]. So we can tabulate (exactly) the nonzero values of $F_\varphi\Big((k - \tilde{\tau}_\varphi[m]) \Delta_{\mathrm{x}} \cos\varphi; \Delta_{\mathrm{x}} \Big)$ for each $m = 0, \ldots, M - 1$ and for each φ of interest. Furthermore, we can compute the forward projection summation (25.2.7) very efficiently because the summation over n is a 1D convolution for each m and φ. In particular, for each n the minimum relevant value of i_{R} is $n + i_{\mathrm{R}}^{\min}[m]$ where

$$i_{\mathrm{R}}^{\min}[m] \triangleq \left\lfloor \tilde{\tau}_\varphi[m] - \frac{r_{\max}(\varphi)}{\Delta_{\mathrm{R}}} \right\rfloor + 1.$$

Thus, for each angular sample φ, we tabulate

$$F_\varphi[k,m] \triangleq F_\varphi\Big((i_{\mathrm{R}}^{\min}[m] + k - \tilde{\tau}_\varphi[m]) \Delta_{\mathrm{x}} \cos\varphi; \Delta_{\mathrm{x}} \Big), \qquad \begin{matrix} k = 0, \ldots, K - 1 \\ m = 0, \ldots, M - 1. \end{matrix}$$

---

[3]For $L = 2$, we write $i_{\mathrm{R}} - 2n = 2(\lfloor i_{\mathrm{R}}/2 \rfloor - n) + i_{\mathrm{R}} \bmod 2$. So two tables suffice, one for even i_{R} and one for odd i_{R}. In general, we would need L tables, namely $F_\varphi\Big((k + l - \tilde{\tau}_\varphi[m]) \frac{1}{L}\Delta_{\mathrm{x}} \cos\varphi; \Delta_{\mathrm{x}} \Big)$ for $l = 0, \ldots, L - 1$, where "k" corresponds to $L(\lfloor i_{\mathrm{R}}/L \rfloor - n)$. Clearly $L = 1$ is simplest.

To simplify implementation, we can let $K = \max_\varphi \left\lceil 2\,\tau_{\max}(\varphi) \right\rceil = \max_\varphi \left\lceil 2\frac{r_{\max}(\varphi)}{\Delta_R} \right\rceil$, and then choose $r'_{\max}(\varphi) = K/2\Delta_R$, in which case

$$i_R^{\min}[m] = \left\lfloor \tilde{\tau}_\varphi[m] - \frac{K}{2} \right\rfloor + 1.$$

There are four possible cases depending on Δ_R in (25.2.19), each of which involves a similar simplification if we use appropriate rotations and/or mirror reversals of the discrete-space image $f[n,m]$.

Consider next the case where $|\cos\varphi| \geq |\sin\varphi|$ but $\Delta_X \cos\varphi < 0$, for which $\Delta_R = -\Delta_X \cos\varphi$. In this case, the system matrix elements involve footprint samples at

$$
\begin{aligned}
(i_R - \tau_\varphi[n,m])\,\Delta_R &= \left(i_R - w_R - \frac{(n-w_X)\Delta_X \cos\varphi + (m-w_Y)\Delta_Y \sin\varphi}{\Delta_R}\right)\Delta_R \\
&= \left(i_R - w_R + n - w_X + (m-w_Y)\frac{\Delta_Y \sin\varphi}{\Delta_X \cos\varphi}\right)\Delta_R \\
&= \left(i_R - n' - w_R + (N-1-w_X) + (m-w_Y)\frac{\Delta_Y \sin\varphi}{\Delta_X \cos\varphi}\right)\Delta_R \\
&= \left(i_R - n' - \tilde{\tau}'_\varphi[m]\right),
\end{aligned}
$$

where we define

$$\tilde{\tau}'_\varphi[m] \triangleq w_R - (N-1-w_X) + (m-w_Y)\frac{\Delta_Y \sin\varphi}{|\Delta_X \cos\varphi|}$$

and $n' = N - 1 - n$. This is just a mirror reversal of $f[n,m]$ over n.

If $|\sin\varphi| > |\cos\varphi|$ and $\Delta_Y \sin\varphi > 0$, then $\Delta_R = \Delta_Y \sin\varphi$ and

$$
\begin{aligned}
(i_R - \tau_\varphi[n,m])\,\Delta_R &= \left(i_R - w_R - \frac{(n-w_X)\Delta_X \cos\varphi + (m-w_Y)\Delta_Y \sin\varphi}{\Delta_R}\right)\Delta_R \\
&= \left(i_R - m - w_R + w_Y - (n-w_X)\frac{\Delta_X \cos\varphi}{\Delta_Y \sin\varphi}\right)\Delta_R.
\end{aligned}
$$

This is akin to (25.2.20) except with a transpose of $f[n,m]$.

Finally, if $|\sin\varphi| > |\cos\varphi|$ and $\Delta_Y \sin\varphi < 0$, then $\Delta_R = -\Delta_Y \sin\varphi$ and

$$
\begin{aligned}
(i_R - \tau_\varphi[n,m])\,\Delta_R &= \left(i_R - w_R - \frac{(n-w_X)\Delta_X \cos\varphi + (m-w_Y)\Delta_Y \sin\varphi}{\Delta_R}\right)\Delta_R \\
&= \left(i_R - w_R + m - w_Y + (n-w_X)\frac{\Delta_X \cos\varphi}{\Delta_Y \sin\varphi}\right)\Delta_R \\
&= \left(i_R - m' - w_R + (M-1-w_Y) + (n-w_X)\frac{\Delta_X \cos\varphi}{\Delta_Y \sin\varphi}\right)\Delta_R,
\end{aligned}
$$

where we define $m' = M - 1 - m$. This case involves both a transpose of $f[n,m]$ and then a mirror reversal of $f[m,n]$ over m.

For the specific case of a square pixel basis function and a strip integral detector response model, one convenient choice for the strip width is $w = \Delta_R$. In this case, $K = 3$ because

$$\frac{r_{\max}(\varphi)}{\Delta_R} = \frac{\frac{1}{2}(|\cos\varphi| + |\sin\varphi|)\Delta_X + \frac{1}{2}\Delta_R}{\Delta_R} = \frac{1}{2} + \frac{1}{2}\frac{|\cos\varphi| + |\sin\varphi|}{\max(|\cos\varphi|, |\sin\varphi|)} \leq \frac{3}{2}.$$

IRT *See* `Gtomo2_table.m`.

The sample spacing (25.2.19) is not immediately applicable to any known tomography system. Nevertheless, it may be useful for iterative image reconstruction, for example in one of the following two ways. We could take the measured sinogram and *interpolate* it into the sampling described by (25.2.19), and then perform iterative image reconstruction using that interpolated sinogram. This pre-interpolation has the risk of compromising image quality, but such interpolation is already done routinely in commercial X-ray CT systems; fan-beam CT sinograms are converted to parallel-beam both to simplify the backprojection process and to avoid noise amplifying effects of fan-beam FBP

weighting factors. Despite such fan-parallel rebinning, the images have been accepted clinically. A second concern is that such pre-interpolation will compromise the statistical models used for iterative image reconstruction. However, all the statistical models that are used for X-ray CT data involve many approximations, *e.g.*, [21]; furthermore, X-ray CT raw data usually is subjected to several precorrection steps that also affect the statistical properties.

If interpolating the data into the Mojette sampling (25.2.19) is unacceptable, there is an alternative. One can first forward project the image using the nonstandard method above, and then interpolate the resulting "Mojette sinogram" into the natural coordinate system of the measurements, *e.g.*, the fan-beam geometry. This approach can leave the data "untouched." The interpolations involved in the forward projection process may help approximate the blur (detector response etc.) of any realistic system, so the iterative reconstruction method may attempt to "deconvolve" this blur, thereby possibly improving, or at least not degrading, the final spatial resolution. Comparing these approaches is an *open problem*.

s,geom,par2,strip

25.2.7 Square-pixel / strip-integral system model (s,geom,par2,strip)

Consider the basis function consisting of a uniform unit pixel:

$$\beta_0(x, y) = \text{rect}(x) \text{rect}(y).$$

e,geom,par2,rect2
(25.2.22)

As discussed in Chapter 3, the Radon transform of this function has the trapezoidal form

$$g(t, \varphi) = l(\varphi) \cdot \begin{cases} \dfrac{d_{\max}(\varphi) + r}{d_{\max}(\varphi) - d_{\text{break}}(\varphi)}, & -d_{\max}(\varphi) < r < -d_{\text{break}}(\varphi) \\ 1, & |r| \leq d_{\text{break}}(\varphi) \\ \dfrac{d_{\max}(\varphi) - r}{d_{\max}(\varphi) - d_{\text{break}}(\varphi)}, & d_{\text{break}}(\varphi) < r < d_{\max}(\varphi) \\ 0, & \text{otherwise,} \end{cases}$$

e,geom,strip,g
(25.2.23)

where

$$\begin{aligned} l(\varphi) &= \frac{1}{\max(|\cos\varphi|, |\sin\varphi|)} \\ d_{\max}(\varphi) &= \frac{|\cos\varphi| + |\sin\varphi|}{2} \\ d_{\text{break}} &= \frac{1}{2} \left| |\cos\varphi| - |\sin\varphi| \right|. \end{aligned}$$

As in Chapter 3, φ denotes the projection angle (angle of the line integral relative to y axis) and r denotes the (signed) distance of the line to the origin.

Since a trapezoid has three pieces, we can rewrite the above projection function as follows:

$$g(t, \varphi) = g_1(r) + g_2(r) + g_3(r)$$

where

$$\begin{aligned} g_1(r) &= \begin{cases} l(\varphi)\frac{r + d_{\max}(\varphi)}{d_{\max}(\varphi) - d_{\text{break}}(\varphi)}, & -d_{\max}(\varphi) < r < -d_{\text{break}}(\varphi) \\ 0, & \text{otherwise} \end{cases} \\ g_2(r) &= \begin{cases} l(\varphi), & |r| \leq d_{\text{break}}(\varphi) \\ 0, & \text{otherwise} \end{cases} \\ g_3(r) &= \begin{cases} l(\varphi)\frac{d_{\max}(\varphi) - r}{d_{\max}(\varphi) - d_{\text{break}}(\varphi)}, & d_{\text{break}}(\varphi) < r < d_{\max}(\varphi) \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

Now consider a *strip-integral* projection model, for a sinogram with n_{R} radial bins spaced by Δ_{R}, and strip-width

w, corresponding to $h_\varphi(r) = \frac{1}{w} \operatorname{rect}\left(\frac{r}{w}\right)$. For this model, we will also need the integrals of the above trapezoid pieces:

$$
\begin{aligned}
\gamma(t_1, t_2) &= \int_{t_1}^{t_2} g(t, \varphi)\,\mathrm{d}t \\
&= \gamma_1(\max(t_1, -d_{\max}), \min(t_2, -d_{\text{break}})) \\
&\quad + \gamma_2(\max(t_1, -d_{\text{break}}), \min(t_2, d_{\text{break}})) \\
&\quad + \gamma_3(\max(t_1, d_{\text{break}}), \min(t_2, d_{\max})),
\end{aligned}
$$

$$\text{e,geom,strip,gam} \quad (25.2.24)$$

where $\gamma_k(t_1, t_2) \triangleq \int_{t_1}^{t_2} g_k(r)\,\mathrm{d}r$. Integrating yields:

$$
\begin{aligned}
\gamma_1(t_1, t_2) &= \frac{l(\varphi)}{2\left[d_{\max}(\varphi) - d_{\text{break}}(\varphi)\right]} \left[(t_2 + d_{\max}(\varphi))^2 - (t_1 + d_{\max}(\varphi))^2\right] \mathbb{1}_{\{t_2 > t_1\}} \\
\gamma_2(t_1, t_2) &= l(\varphi)(t_2 - t_1)\,\mathbb{1}_{\{t_2 > t_1\}} \\
\gamma_3(t_1, t_2) &= \frac{l(\varphi)}{2\left[d_{\max}(\varphi) - d_{\text{break}}(\varphi)\right]} \left[(t_1 - d_{\max}(\varphi))^2 - (t_2 - d_{\max}(\varphi))^2\right] \mathbb{1}_{\{t_2 > t_1\}}.
\end{aligned}
$$

The elements of the system matrix for a strip-integral model (for one projection angle) are given by (25.2.8), where

$$
\begin{aligned}
F_\varphi(r; \Delta_{\mathrm{x}}, w) &\triangleq \int h_\varphi(r - r')\,\Delta_{\mathrm{x}}\, g(r'/\Delta_{\mathrm{x}}, \varphi)\,\mathrm{d}r' \\
&= \frac{1}{w} \int_{r-w/2}^{r+w/2} \Delta_{\mathrm{x}}\, g(r'/\Delta_{\mathrm{x}}, \varphi)\,\mathrm{d}r' \\
&= \frac{\Delta_{\mathrm{x}}^2}{w} \int_{(r-w/2)/\Delta_{\mathrm{x}}}^{(r+w/2)/\Delta_{\mathrm{x}}} g(t, \varphi)\,\mathrm{d}t \\
&= \frac{\Delta_{\mathrm{x}}^2}{w} \gamma\left(\frac{r - w/2}{\Delta_{\mathrm{x}}}, \frac{r + w/2}{\Delta_{\mathrm{x}}}\right),
\end{aligned}
$$

$$\text{e,geom,par2,strip,aij} \quad (25.2.25)$$

where γ was defined in (25.2.24). For this model, the support of $F_\varphi(t\Delta_{\mathrm{R}}; \Delta_{\mathrm{x}}, w)$ is governed by (25.2.11) where

$$
\tau_{\max}(\varphi) = \frac{\Delta_{\mathrm{x}} d_{\max}(\varphi) + w/2}{\Delta_{\mathrm{R}}},
$$

for which

$$
M = \max_\varphi \lceil 2\,\tau_{\max}(\varphi) \rceil = \left\lceil \frac{\Delta_{\mathrm{x}}\sqrt{2} + w}{\Delta_{\mathrm{R}}} \right\rceil.
$$

IRT *See* `square_strip_int.m`.

Fig. 25.2.2 shows the entire system matrix A for a "strip integral" system model.

25.2.8 Siddon's line-integral method (s,geom,par2,siddon)

The trapezoidal form for the projection of a square, as given in (25.2.23), was derived in Example 3.2.6 using the Fourier-slice theorem, resulting in the convolution of two rectangles. Alternatively, one can derive an equivalent form geometrically, as illustrated in Fig. 25.2.3. This is the basis for Siddon's forward projection method [22].

As given in (3.2.2), the locus of points along the ray at angle φ and (signed) distance r from the origin is

$$
\{(r\cos\varphi, r\sin\varphi) + \ell(-\sin\varphi, \cos\varphi) : l \in \mathbb{R}\}.
$$

(The case $\ell = 0$ corresponds to the point along the ray that is closest to the origin.) For rays where φ is not a multiple of $\pi/2$, one can find the intersection of the ray with the vertical lines that bound the square pixel to the left and right using

$$
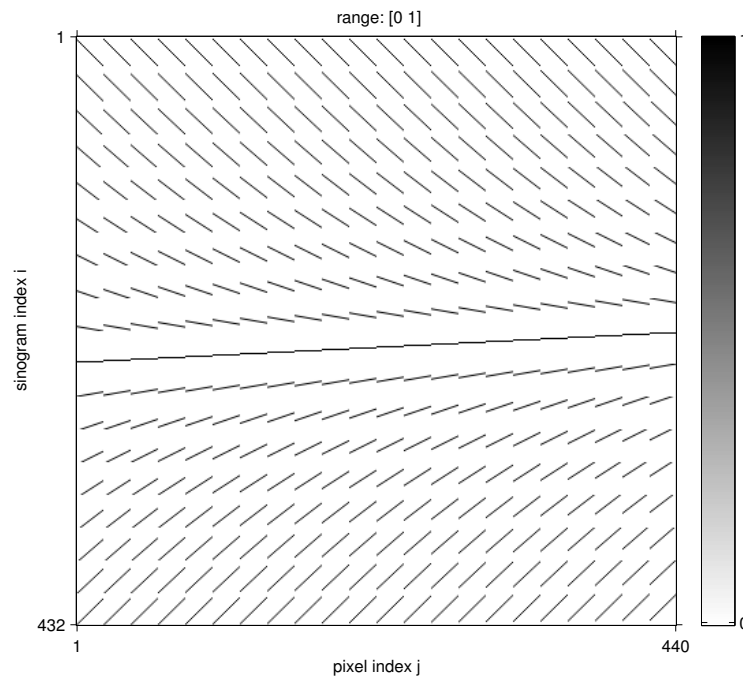\ell_\pm^{\mathrm{x}} = \frac{r\cos\varphi - x_\pm}{\sin\varphi},
$$

fig˙geom˙strip˙system
Figure 25.2.2: Elements of a system matrix A for a "strip integral" tomographic system model for a 20×22 image and a 24×18 sinogram with $\Delta_{\mathrm{X}} = 1$ and $\Delta_{\mathrm{R}} = 1$.

fig˙geom˙siddon2
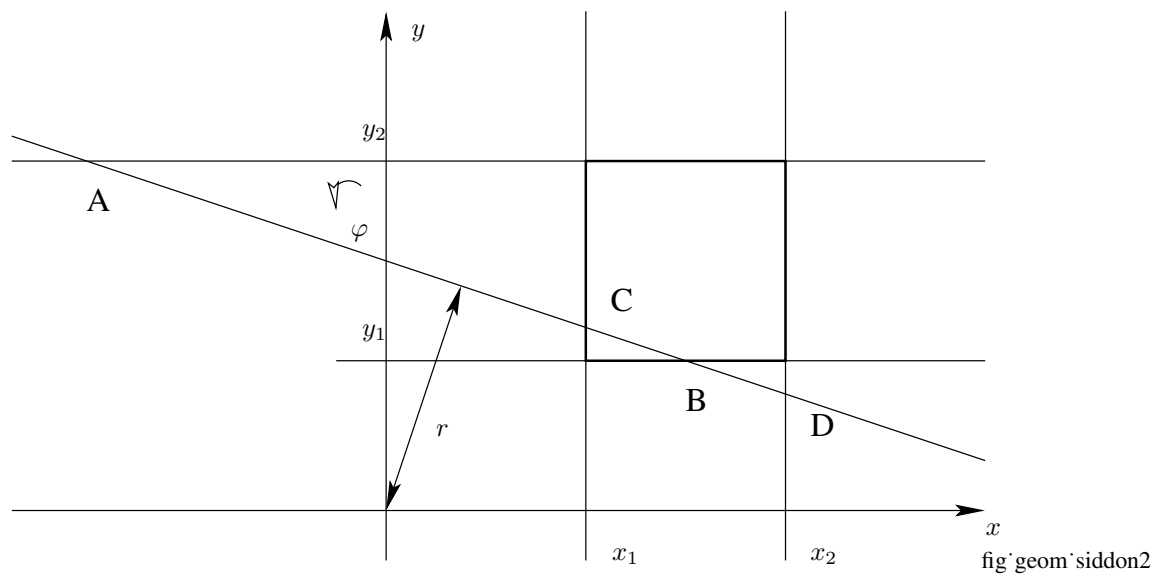
Figure 25.2.3: Siddon's line-integral method illustrated in 2D parallel-beam geometry.

where $x_\pm$ denote the x coordinates of the left and right edges of the pixel. Similarly, the intersections of the ray with the horizontal lines that bound the pixel from above and below are:

$$\ell_\pm^{\mathrm{y}} = \frac{y_\pm - r\sin\varphi}{\cos\varphi},$$

where $y_\pm$ denote the y coordinates of the top and bottom edges of the pixel. Thus the line segment intersecting the voxel is the intersection of the intervals $\left(\min(\ell_\pm^{\mathrm{x}}), \max(\ell_\pm^{\mathrm{x}})\right)$ and $\left(\min(\ell_\pm^{\mathrm{y}}), \max(\ell_\pm^{\mathrm{y}})\right)$. In Siddon's method, a_{ij} is the length of this intersection. This approach is applicable only for the ideal line-integral model with no detector blur, *i.e.*, $h_\varphi(r) = \delta(r)$.

s,geom,par2,lin1

25.2.9 Linear interpolation approximation (s,geom,par2,lin1)

Conventional pixel-driven back-projection methods can be put in the framework described here by making a particular choice for the basis function $\beta_0(x,y)$ in (25.2.1) and for the blur $h_\varphi(r)$ in (25.2.3). Specifically, consider using Dirac impulses for the basis "functions:" $\beta_0(x,y) = \delta_2(x,y)$, and a triangular function for the detector response: $h_\varphi(r) = \frac{1}{\Delta_{\mathrm{R}}}\operatorname{tri}\left(\frac{r}{\Delta_{\mathrm{R}}}\right)$. Substituting into (25.2.4) yields $g(t,\varphi) = \delta(r)$, which applied in (25.2.8) yields

$$a[i_{\mathrm{R}}, \varphi; n, m] = \Delta_{\mathrm{x}}^2\, h_\varphi((i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}} - r_\varphi[n,m]) = \frac{\Delta_{\mathrm{x}}^2}{\Delta_{\mathrm{R}}}\operatorname{tri}\left(\frac{(i_{\mathrm{R}} - w_{\mathrm{R}})\Delta_{\mathrm{R}} - r_\varphi[n,m]}{\Delta_{\mathrm{R}}}\right).$$

To within scale factors, this corresponds to the pixel-driven backprojector (3.5.5). That backprojector is adequate for FBP reconstruction, but the corresponding forward projector produces unacceptable aliasing artifacts due to the frequency spectrum of the Dirac impulse basis.

s,geom,par2,dd

25.2.10 Distance-driven approximation (s,geom,par2,dd)

De Man and Basu proposed a projection / backprojection method called *distance driven*. The 2D forward projector works by mapping pixel boundaries and detector boundaries to a common axis; one then computes lengths of intersecting intervals along that axis, and divides by the detector width and the cosine of the projection angle [23–25]. Fig. 25.2.4 illustrates the approximation. The method can be implemented with an efficient memory access pattern [26]. In 2D, the method is equivalent to replacing (25.2.25) with the following approximation:

$$F_\varphi(r; \Delta_{\mathrm{x}}, w) \approx \frac{F_0(r; \Delta_{\mathrm{x}}\max(|\cos\varphi|, |\sin\varphi|), w)}{\max^2(|\cos\varphi|, |\sin\varphi|)}.$$

One can verify that this approximation is a special case of (25.2.18) where $\theta = 0$. One can also show that the above approach is equivalent to the basis function model (25.2.1) for the particular basis function $\beta_0(x,y) = \operatorname{rect}(x)\,\delta(y)$ for projection angles $\varphi \in [-\pi/4, \pi/4]$ and for the basis function $\beta_0(x,y) = \delta(x)\operatorname{rect}(y)$ for projection angles $\varphi \in [\pi/4, 3\pi/4]$. In other words, it is equivalent to using *different* basis functions for different projection view angles. However, this is only a *mathematical* equivalence; the *procedure* proposed by De Man and Basu is quite different from the tabulation approach described in §25.2. An *open problem* is to analyze the error of the above approximation. Another *open problem* is to explore whether there are variations that would reduce that error without increasing computation, *e.g.*, by tuning Δ_{x} or w on the right-hand side.

The DD projector is not invariant to upsampling by pixel replication.

The distance-driven method uses a rectangular approximation to the trapezoidal footprint (25.2.23) of a rectangular image basis function. An earlier approach called *Joseph's method* [27] uses a triangular approximation to the trapezoidal footprint as follows:

$$g(t,\varphi) \approx l(\varphi)\operatorname{tri}(t\, l(\varphi)) = \frac{1}{\max(|\cos\varphi|, |\sin\varphi|)}\operatorname{tri}\left(\frac{t}{\max(|\cos\varphi|, |\sin\varphi|)}\right).$$

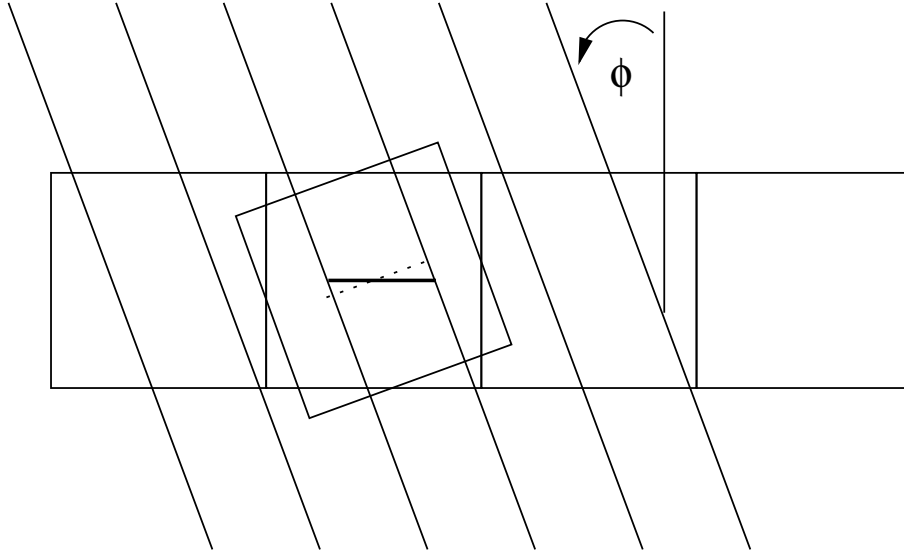The strip-integral generalization was considered in [28].

fig,geom,dd2

Figure 25.2.4: Illustration of approximation used in 2D distance-driven projection method.

s,geom,par2,nufft

25.2.11   Fourier-based projection / backprojection (s,geom,par2,nufft)

One natural approach to forward projection is to use the Fourier-slice theorem (3.2.24). That theorem relates the 1D FT of each projection to samples of the 2D FT of the object on a polar grid. The use of the Fourier-slice theorem as a tool for *reprojection* was noted in the late 1980's by Crawford *et al.* [29, 30], in the context of correction of beam hardening in X-ray CT, and by Stearns *et al.* [31,32], in the context of filling in missing projections in (noniterative) 3D PET image reconstruction. These approaches were apparently largely abandoned thereafter due in part to unacceptable image artifacts caused by the large interpolation errors associated with conventional "gridding" methods for converting between polar and Cartesian coordinates in frequency space. The importance of accurate interpolation for gridding has been analyzed rigorously [33, p. 119]. More recently, optimized Kaiser-Bessel interpolators for the nonuniform fast Fourier transform (NUFFT) [34–37] have been shown to largely eliminate interpolation artifacts, renewing interesting in Fourier-based reprojection methods, *e.g.*, [38, 39].

Frequency domain interpolation for NUFFT is closely related to "gridding" methods. Such methods have been considered both for tomography and for general applications, *e.g.*, [31, 40–67], and for MR imaging [68–73]. In most of these papers, the focus was using gridding to find a *non-iterative* approximate solution to an *inverse* problem. In contrast, Fourier-based reprojection is a tool for calculating the *forward* problem, thereby allowing an iterative reconstruction method to solve the inverse problem. The prospect of using graphics hardware to assist in this projection process is enticing [74–78].

Let $f(x, y)$ denote the 2D image whose projections we wish to compute. For the purposes of Fourier-based reprojection, the most useful expression for the Fourier-slice theorem is

$$f(x,y) \overset{\text{Radon}}{\leftrightarrow} p_\varphi(r) = \int F(\rho\cos\varphi, \rho\sin\varphi)\, e^{\imath 2\pi\rho r}\, d\rho, \qquad\qquad \begin{matrix}\text{e,geom,par2,nufft,pangr}\\ (25.2.26)\end{matrix}$$

where $p_\varphi(r)$ denotes the Radon transform of $f(x, y)$ and $F(u, v)$ its spectrum:

$$f(x,y) \overset{\text{2D FT}}{\longleftrightarrow} F(u,v) = \iint f(x,y)\, e^{-\imath 2\pi(xu+yv)}\, dx\, dy\,.$$

The complication here is that the preceding integral is in Cartesian coordinates whereas (25.2.26) essentially involves polar coordinates.

Now consider the usual basis function representation of $f(x, y)$ in (25.2.1), *i.e.*,

$$f(x,y) = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f[n,m]\, \beta_0\left(\frac{x - (n - w_\text{X})\Delta_\text{X}}{\Delta_\text{X}},\ \frac{y - (m - w_\text{Y})\Delta_\text{Y}}{\Delta_\text{Y}}\right).$$

By the shift property of the FT, the corresponding spectrum is exactly

$$
\begin{aligned}
F(u,v) &= \Delta_{\mathrm{X}}\Delta_{\mathrm{Y}}\, B_0(u,v)\, e^{\imath 2\pi(u\Delta_{\mathrm{X}} w_{\mathrm{X}} + v\Delta_{\mathrm{Y}} w_{\mathrm{Y}})}\, F_d(2\pi u\Delta_{\mathrm{X}}, 2\pi v\Delta_{\mathrm{Y}}) \\
&= B_1(u,v)\, F_d(2\pi u\Delta_{\mathrm{X}}, 2\pi v\Delta_{\mathrm{Y}}),
\end{aligned}
$$

e,geom,par2,nufft,Fuv
(25.2.27)

where $B_0(u,v)$ is the 2D FT of the basis function $\beta_0(x,y)$, we define

$$
B_1(u,v) \triangleq \Delta_{\mathrm{X}}\Delta_{\mathrm{Y}}\, B_0(u,v)\, e^{\imath 2\pi(u\Delta_{\mathrm{X}} w_{\mathrm{X}} + v\Delta_{\mathrm{Y}} w_{\mathrm{Y}})},
$$

e,geom,par2,nufft,B1
(25.2.28)

and the 2D *DSFT* of $f[n,m]$ is:

$$
F_d(\omega_1, \omega_2) = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f[n,m]\, e^{-\imath(\omega_1 n + \omega_2 m)}.
$$

e,geom,par2,nufft,dsft2
(25.2.29)

Both $B_1(u,v)$ and $F_d(\omega_1,\omega_2)$ are Hermitian symmetric spectra, since $\beta_0(x,y)$ and $f[n,m]$ are real. Although the discrete-space spectrum $F_d(\omega_1,\omega_2)$ is 2π-periodic, the corresponding continuous-space spectrum $F(u,v)$ is not, due to the spectral properties of the basis $\beta_0(x,y)$ (except in the unrealistic case of the Dirac impulse basis). We can compute (25.2.27) exactly using a (slow) 2D FT or approximately (to within the desired accuracy) by using a 2D NUFFT [37,79]. Instead of using an NUFFT designed for arbitrary sampling patterns, one could use instead a special purpose NUFFT designed for the polar sampling pattern, *e.g.*, [80].

As in (25.2.3) one can account for shift-invariant detector blur $h(r)$ having corresponding frequency response $H(\nu)$, so by the convolution property of the FT and the Fourier slice theorem (25.2.26) we wish to compute

$$
\begin{aligned}
\bar{y}_\varphi[i_{\mathrm{R}}] &= h(r) * p_\varphi(r)\Big|_{r=(i_{\mathrm{R}}-w_{\mathrm{R}})\Delta_{\mathrm{R}}} \\
&= \int_{-\infty}^{\infty} F(\rho\cos\varphi, \rho\sin\varphi)\, H(\rho)\, e^{\imath 2\pi\rho\Delta_{\mathrm{R}}(i_{\mathrm{R}}-w_{\mathrm{R}})}\, d\rho,
\end{aligned}
$$

e,geom,par2,nufft,int1
(25.2.30)

for $i_{\mathrm{R}} = 0,\ldots,n_{\mathrm{R}}-1$, where n_{R} is the number of radial samples, and Δ_{R} denotes the radial sampling distance. A simple choice is $h(r) = \frac{1}{w}\operatorname{rect}\!\left(\frac{r}{w}\right)$ for which $H(\nu) = \operatorname{sinc}(w\nu)$. This model accounts for the finite width of the detector elements.

For a forward projection, we must compute approximations to (25.2.30) for a finite collection of φ values. In general, $F(u,v)$ is not band-limited, so we must truncate the integral limits and discretize. For speed, we want to evaluate the resulting sum using an N_1-point inverse FFT for suitably chosen N_1, so we choose a sample spacing $\Delta_\rho = \frac{1}{N_1\Delta_{\mathrm{R}}}$, yielding the following approximation:

$$
\bar{y}_\varphi[i_{\mathrm{R}}] \approx \tilde{y}_\varphi[i_{\mathrm{R}}] \triangleq \frac{1}{N_1}\sum_{k=-K_1+1}^{K_1-1} Z_\varphi[k]\, e^{\imath 2\pi k i_{\mathrm{R}}/N_1},
$$

e,geom,par2,nufft,ytangir
(25.2.31)

for $i_{\mathrm{R}} = 0,\ldots,n_{\mathrm{R}}-1$, where

$$
Z_\varphi[k] \triangleq \frac{1}{\Delta_{\mathrm{R}}} H\!\left(\frac{k}{N_1\Delta_{\mathrm{R}}}\right) e^{-\imath 2\pi k w_{\mathrm{R}}/N_1}\, F(u_k, v_k),
$$

e,geom,par2,nufft,Zangk
(25.2.32)

where $u_k = \frac{k}{N_1\Delta_{\mathrm{R}}}\cos\varphi$ and $v_k = \frac{k}{N_1\Delta_{\mathrm{R}}}\sin\varphi$ for $k = -K_1+1,\ldots,K_1-1$. An alternative expression is

$$
\tilde{y}_\varphi[i_{\mathrm{R}}] = \tilde{y}_\varphi^+[i_{\mathrm{R}}] + \tilde{y}_\varphi^-[i_{\mathrm{R}}] = 2\operatorname{real}\!\left(\tilde{y}_\varphi^+[i_{\mathrm{R}}]\right),
$$

e,geom,par2,nufft,ytangir,2real
(25.2.33)

where

$$
\tilde{y}_\varphi^+[i_{\mathrm{R}}] \triangleq \frac{1}{N_1}\sum_{k=0}^{K_1-1} \tilde{Z}_\varphi[k]\, e^{\imath 2\pi k i_{\mathrm{R}}/N_1}.
$$

e,geom,par2,nufft,ytangir+
(25.2.34)

$$
\tilde{y}_\varphi^-[i_{\mathrm{R}}] \triangleq \frac{1}{N_1}\sum_{k=-K_1+1}^{0} \tilde{Z}_\varphi[k]\, e^{\imath 2\pi k i_{\mathrm{R}}/N_1} = \frac{1}{N_1}\sum_{k=0}^{K_1-1} \tilde{Z}_\varphi[-k]\, e^{-\imath 2\pi k i_{\mathrm{R}}/N_1}
$$

$$
\tilde{Z}_\varphi[k] \triangleq \left(1 - \frac{1}{2}\delta[k]\right) Z_\varphi[k].
$$

e,geom,par2,nufft,Ztangk
(25.2.35)

Because $F(u, v)$ and H are Hermitian symmetric, so is $Z_\varphi[k]$, so $\tilde{y}_\varphi^+[i_R] = \left(\tilde{y}_\varphi^-[i_R]\right)^*$.

If $K_1 = N_1$, then computing $\tilde{y}_\varphi^+[i_R]$ requires an ordinary N_1-point FFT. If $K_1 < N_1$, then we zero-pad $\tilde{Z}_\varphi[k]$ first. For high resolution imaging, we may have $K_1 > N_1$. For example if $K_1 = 2N_1$, then

$$\tilde{y}_\varphi^+[i_R] = \frac{1}{N_1} \sum_{k=0}^{N_1-1} \left(\tilde{Z}_\varphi[k] + \tilde{Z}_\varphi[k + N_1]\right) e^{i2\pi k i_R/N_1},$$

which again requires only a N_1-point inverse FFT.

The question remains then how to choose N_1 and K_1. Clearly we need $N_1 \geq n_R$. For a space-limited object the projections $p_\varphi(r)$ are themselves space-limited, spanning some field of view FOV_R. (Typically $\mathrm{FOV}_R \approx n_R \Delta_R$, although the widths of the detector response and the image basis function also may contribute.) From a sampling perspective, to avoid spatial aliasing of $p_\varphi(r)$, we need $N_1 \geq \mathrm{FOV}_R/\Delta_R$. Thus, we choose

$$N_1 = \max\{n_R, \mathrm{FOV}_R/\Delta_R\}.$$

If we chose sinc functions for the basis, then $F(u, v)$ would be band-limited to a maximum frequency $\sqrt{2}/(2\Delta_X)$, assuming $\Delta_Y = \pm\Delta_X$, so the natural choice for K_1 would satisfy $K_1 \Delta_\rho \approx \sqrt{2}1/(2\Delta_X)$, *i.e.*, $K_1 \approx \sqrt{2}\frac{N_1}{2}\frac{\Delta_R}{\Delta_X}$. However, for more reasonable space-limited basis functions, $F(u, v)$ is not band-limited, so the choice of K_1 involves a tradeoff between accuracy and speed.

$\underset{\sim}{?}$

25.2.11.1 Implementation

Here is a summary of the NUFFT-based reprojection method.

Step 1. Use the 2D NUFFT method with $\omega_1 = 2\pi\Delta_X \frac{k}{N_1\Delta_R} \cos\varphi$ and $\omega_2 = 2\pi\Delta_Y \frac{k}{N_1\Delta_R} \sin\varphi$ for $k = 1, \ldots, K_1-1$, for each φ value of interest, to compute an accurate approximation to the DSFT in (25.2.29). Compute the DC value exactly via $F_d(0, 0) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f[n, m]$. Use $u = \rho\cos\varphi$ and $v = \rho\sin\varphi$ with $\rho = k/(K\Delta_R)$ for $k = -N/2, \ldots, N/2 - 1$.

Step 2. Scale the NUFFT output by the factors in (25.2.28), (25.2.32) and (25.2.35) to compute $\tilde{Z}_\varphi[k]$.

Step 3. Take the inverse N_1-point FFT of each $\tilde{Z}_\varphi[\cdot]$ set (for each φ) using (25.2.34). Discard all but the samples $i_R = 0, \ldots, n_R - 1$. Take twice the real part to form $\tilde{y}_\varphi[i_R]$ per (25.2.33).

| IRT | *See* `Gtomo_nufft.m`.

25.2.11.2 Extensions

The inverse FFT (25.2.31) yields uniformly-spaced radial samples. Typical PET systems having circular geometries acquire nonuniform radial samples. For conventional FBP reconstruction prior to ramp filtering one interpolates the nonuniform radial samples onto equally-spaced samples, often called *arc correction*. Such "preprocessing" interpolation is suboptimal for iterative reconstruction since it destroys the statistical independence of the measurements. It is preferable to build the nonuniform spacing into the reprojection method.

The min-max NUFFT interpolation method described in [37] is applicable to cases where (25.2.31) must be replaced by nonuniform radial sampling. Similar considerations apply for *fan-beam* projection [81, 82].

25.2.11.3 Adjoint

The method described above is a linear operator and hence corresponds implicitly to some $(n_A \cdot n_R) \times (N \cdot M)$ matrix. Iterative algorithms usually also need the ability to compute matrix-vector multiplication by the transpose of that matrix, even though the matrix itself is not stored explicitly. It is straightforward to "reverse" (not invert!) the steps described above to develop an algorithm to perform multiplication by the transpose, corresponding to the adjoint of the forward operator, which is a form of backprojection.

25.2.12   Local spectrum of Gram operator (s,geom,par2,gram)

We can find the *local frequency response* of the *Gram matrix* $A'WA$ by taking the FFT of its jth column. However, it can also be useful to have approximate *analytical* expressions for that local frequency response [83].

Consider the image domain model (25.2.1) and the 2D parallel-beam geometry having system model (25.2.3). The elements a_{ij} of the corresponding system matrix A can be written as $a_{ij} = a(r_k, \varphi_l; n, m)$ where $j = 1 + n + mN$, $i = 1 + k + (l-1)n_{\mathrm{R}}$, for $k = 0, \ldots, n_{\mathrm{R}} - 1$, $l = 1, \ldots, n_{\mathrm{A}}$, and

$$a(r, \varphi; n, m) = a(r - r_\varphi[n, m], \varphi), \quad \text{where } a(r, \varphi) \triangleq \int h_\varphi(r - r') \Delta_{\mathrm{X}}\, g\left(\frac{r'}{\Delta_{\mathrm{X}}}, \varphi\right) \mathrm{d}r,$$

using (25.2.4). Rewrite the elements of weighing matrix $W = \mathrm{diag}\{w_i\}$ as $w_i = w(r_k, \phi_l)$. Then the elements of the *Gram matrix* are given exactly by

$$h_{\mathrm{d}}[n, m; n', m'] = [A'WA]_{jj'} = \sum_{l=1}^{n_{\mathrm{A}}} \sum_{k=1}^{n_{\mathrm{R}}} w(r_k, \varphi_l)\, a(r_k, \varphi_l; n, m)\, a(r_k, \varphi_l; n', m') \qquad \begin{array}{l}\text{e,geom,par2,gram,h,sum,kl}\\ (25.2.36)\end{array}$$

where $j' = 1 + n' + m'N$. For a pixel at a given location (n', m') we can compute the local frequency response of the Gram matrix by taking the DFT of $h_{\mathrm{d}}[n, m; n', m']$ with respect to (n, m), thereby enabling predictions of resolution and noise properties. However, when we want to do this for many such locations, then many DFT computations would be required. We derive next a simpler analytical approximation for the local frequency response.

The main impediment to analysis is that the expression (25.2.36) is not exactly shift invariant, because the ray samples r_k intersect each pixel somewhat differently. However, the aggregate effect of all those rays is quite similar from pixel to pixel, motivating the following approximation. As in (25.2.3), we assume the radial samples r_k are equally spaced, *i.e.*, $r_k = r_0 + k\Delta_{\mathrm{R}}$. We then use an integral (!) to approximate the summation[4] in (25.2.36) as follows:

$$h_{\mathrm{d}}[n, m; n', m'] \approx \sum_{l=1}^{n_{\mathrm{A}}} \frac{1}{\Delta_{\mathrm{R}}} \int_{-\infty}^{\infty} w(r, \varphi_l)\, a(r, \varphi_l; n, m)\, a(r, \varphi_l; n', m')\, \mathrm{d}r$$

$$= \sum_{l=1}^{n_{\mathrm{A}}} \frac{1}{\Delta_{\mathrm{R}}} \int_{-\infty}^{\infty} w(r, \varphi_l)\, a(r - r_{\varphi_l}[n, m], \varphi_l)\, a(r - r_{\varphi_l}[n', m'], \varphi_l)\, \mathrm{d}r$$

$$= \sum_{l=1}^{n_{\mathrm{A}}} w(\varphi_l; n, m; n', m') \frac{1}{\Delta_{\mathrm{R}}} \int_{-\infty}^{\infty} a(r - r_{\varphi_l}[n, m], \varphi_l)\, a(r - r_{\varphi_l}[n', m'], \varphi_l)\, \mathrm{d}r, \quad \begin{array}{l}\text{e,geom,par2,gram,h,sum,l}\\ (25.2.37)\end{array}$$

where

$$w(\varphi; n, m; n', m') \triangleq \frac{\int_{-\infty}^{\infty} w(r, \varphi)\, a(r - r_\varphi[n, m], \varphi)\, a(r - r_\varphi[n', m'], \varphi)\, \mathrm{d}r}{\int_{-\infty}^{\infty} a(r - r_\varphi[n, m], \varphi)\, a(r - r_\varphi[n', m'], \varphi)\, \mathrm{d}r}.$$

Because $w(r, \varphi)$ often varies slowly in r relative to the typically sharp peak of $a(r, \phi)$ at $r = 0$, we make the following approximation:

$$w(\varphi; n, m; n', m') \approx w(\varphi; n_0, m_0) \triangleq w(r_\varphi[n_0, m_0], \varphi),$$

where provided that (n, m) and (n', m') are "sufficiently close" to (n_0, m_0). Here, (n_0, m_0) denotes the coordinates of the jth pixel for which we want a local Fourier approximation, Substituting into (25.2.37) yields

$$h_{\mathrm{d}}[n, m; n', m'] \approx \sum_{l=1}^{n_{\mathrm{A}}} w(\varphi_l; n_0, m_0)\, h_{\varphi_l}[n, m; n', m'], \qquad \begin{array}{l}\text{e,geom,par2,gram,h,sum,wh}\\ (25.2.38)\end{array}$$

where

$$h_\varphi[n, m; n', m'] \triangleq \frac{1}{\Delta_{\mathrm{R}}} \int_{-\infty}^{\infty} a(r - r_\varphi[n, m], \varphi)\, a(r - r_\varphi[n', m'], \varphi)\, \mathrm{d}r.$$

Making the change of variables $r' = r - r_\varphi[n', m']$ shows that

$$\begin{aligned} h_\varphi[n, m; n', m'] &= q_\varphi(r_\varphi[n, m] - r_\varphi[n', m']) \\ &= q_\varphi(\Delta_{\mathrm{X}}(n - n')\cos\varphi + \Delta_{\mathrm{Y}}(m - m')\sin\varphi), \end{aligned}$$

---

[4] See [84] for an analogous approximation for approximating FBP variance.

where the following 1D autocorrelation is with respect to r:

$$q_\varphi(r) \triangleq \frac{1}{\Delta_R} \, a(r, \varphi) \star a(r, \varphi) \, .$$

Thus, thanks to the simple approximation (25.2.37), we now have a form that is shift-invariant (except for edge effects).

We have ignored the finite support *mask* that is present in any discrete-space image reconstruction method. Let $\mu(x, y)$ denote the reconstruction FOV. Then a more accurate model is

$$h_\varphi[n, m; n', m'] = q_\varphi(\Delta_X(n - n') \cos\varphi + \Delta_Y(m - m') \sin\varphi) \, \mu(x_c[n], y_c[m]) \, \mu(x_c[n'], y_c[m']),$$

where the pixel centers $(x_c[n], y_c[m])$ were defined in (25.2.2). Because this model is not completely shift-invariant, we must make a choice about how to define the *local impulse response* near pixel (n_0, m_0). One approach is to use the coordinate transformation recommended for analyzing quasistationary noise in [85, p. 870] as follows:

$$\begin{aligned}
\tilde{h}_\varphi[n, m; n_0, m_0] &\triangleq h_\varphi[n_0 + n/2, m_0 + m/2; n_0 - n/2, m_0 - m/2] \\
&= q_\varphi(n\Delta_X \cos\varphi + m\Delta_Y \sin\varphi) \, \mu_1(n\Delta_X, m\Delta_Y),
\end{aligned}$$

where $x_0 \triangleq x_c[n_0]$, $y_0 \triangleq y_c[m_0]$, and

$$\mu_1(x, y) \triangleq \mu(x_0 + x/2, y_0 + y/2) \, \mu(x_0 - x/2, y_0 - y/2) \, .$$

This approach yields a local impulse response that is symmetric in (n, m), thus ensuring that its spectrum is real. An alternative is to reference all displacements relative to the point (n_0, m_0) as follows:

$$\begin{aligned}
\tilde{h}_\varphi[n, m; n_0, m_0] &\triangleq h_\varphi[n_0 + n, m_0 + m; n_0, m_0] \\
&= q_\varphi(n\Delta_X \cos\varphi + m\Delta_Y \sin\varphi) \, \mu_2(n\Delta_X, m\Delta_Y),
\end{aligned}$$

where

$$\mu_2(x, y) \triangleq \mu(x_0 + x, y_0 + y) \, \mu(x_0, y_0) \, . \tag{25.2.39}$$

<div align="right">e,geom,par2,gram,mu2</div>

This choice is not symmetric in (n, m), but it better corresponds to the local Fourier analyses based on the DFT of $A'WAe_j$, so we focus on it hereafter.

In light of (25.2.38) and the linearity of the DFT, to determine the frequency response of $h_d[n, m; n', m']$ we must find the spectrum of $\tilde{h}_\varphi[n, m; n_0, m_0]$, *i.e.*, we want to determine

$$\begin{aligned}
H_\varphi(\omega_1, \omega_2) &= \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} \tilde{h}_\varphi[n, m; n_0, m_0] \, e^{-\imath(\omega_1 n + \omega_2 m)} \\
&= \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} s_\varphi(n\Delta_X, m\Delta_Y) \, e^{-\imath(\omega_1 n + \omega_2 m)} \\
&\approx \frac{1}{\Delta_X} \frac{1}{\Delta_Y} \iint s_\varphi(x, y) \, e^{-\imath(\omega_1 x/\Delta_X + \omega_2 y/\Delta_Y)} \, dx \, dy \\
&= \frac{1}{\Delta_X} \frac{1}{\Delta_Y} S_\varphi\left(\frac{\omega_1}{2\pi\Delta_X}, \frac{\omega_2}{2\pi\Delta_Y}\right),
\end{aligned}$$

where we define the following "strip like" function:

$$s_\varphi(x, y) \triangleq q_\varphi(x \cos\varphi + y \sin\varphi) \, \mu_2(x, y),$$

and $s_\varphi(x, y) \overset{\text{2D FT}}{\longleftrightarrow} S_\varphi(u, v)$.

We assume hereafter that the field of view is a convex and symmetric set. Thus the intersection of the ray through point (x_0, y_0) at angle φ and the reconstruction support is a line segment with width FOV_φ. For example, if the support is a disk with radius R, then one can show that $\text{FOV}_\varphi = 2\sqrt{R^2 - (x_0 \cos\varphi + y_0 \sin\varphi)^2}$. Then because a is sharply peaked, we make the approximation[5]

$$s_\varphi(x, y) \approx q_\varphi(x \cos\varphi + y \sin\varphi) \, \text{rect}\left(\frac{-x \sin\varphi + y \cos\varphi}{\text{FOV}_\varphi}\right), \tag{25.2.40}$$

<div align="right">e,geom,par2,gram,mask</div>

---

[5]For a circular support, a better approximation would be $\text{rect}\left(\frac{-x \sin\varphi + y \cos\varphi - \ell_0}{\text{FOV}_\varphi}\right)$, where $\ell_0 \triangleq -x_0 \sin\varphi + y_0 \cos\varphi$. This shift by ℓ_0 arises from the fact that the choice (25.2.39) is not symmetric, so we disregard it hereafter.

where FOV_φ denotes the width of the profile at angle φ across the field of view through the point (x_0, y_0). Fig. 25.2.5 illustrates the accuracy of this approximation.
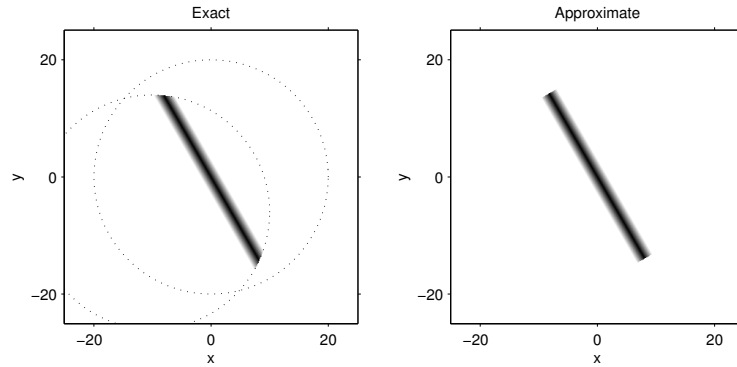
<div style="text-align:right">fig˙geom˙gram˙mask</div>

Figure 25.2.5: Exact version of $s_\varphi(x, y)$ versus its approximation (25.2.40).

The 2D FT of $s_\varphi(x, y)$ is easiest to see for the case $\varphi = 0$:

$$s_0(x, y) \quad = \quad q_0(x) \operatorname{rect}\left(\frac{y}{\mathrm{FOV}_0}\right)$$

$$\overset{\text{2D FT}}{\longleftrightarrow} \quad S_0(u, v) = \frac{1}{\Delta_\mathrm{R}} |A_0(u)|^2 \, \mathrm{FOV}_0 \operatorname{sinc}(\mathrm{FOV}_0 v),$$

where $q_\varphi(\cdot) \overset{\text{FT}}{\longleftrightarrow} \frac{1}{\Delta_\mathrm{R}} |A_\varphi(\nu)|^2$ by the autocorrelation property of the 1D FT. Similarly, by the rotation property of the 2D FT:

$$S_\varphi(\rho, \Phi) \approx \frac{1}{\Delta_\mathrm{R}} |A_\varphi(\rho \cos(\Phi - \varphi))|^2 \, \mathrm{FOV}_\varphi \operatorname{sinc}(\mathrm{FOV}_\varphi \rho \sin(\Phi - \varphi)).$$

Combining all of the above approximations yields the following result for the local frequency response around a point (n_0, m_0):

$$H_\mathrm{d}(\omega_1, \omega_2) \approx H_1(\rho, \Phi)\Big|_{\rho = \sqrt{(\omega_1/2\pi\Delta_\mathrm{X})^2 + (\omega_2/2\pi\Delta_\mathrm{Y})^2}, \ \Phi = \angle_\pi(\omega_1, \omega_2)}$$

where

$$H_1(\rho, \Phi) \quad \triangleq \quad \frac{1}{\Delta_\mathrm{R}\Delta_\mathrm{X}\Delta_\mathrm{Y}} \sum_{l=1}^{n_\mathrm{A}} w(\varphi_l; n_0, m_0) \tag{25.2.41}$$

<div style="text-align:right">e,geom,par2,gram,HqA,1</div>

$$\cdot \, |A_{\varphi_l}(\rho \cos(\Phi - \varphi_l))|^2 \, \mathrm{FOV}_{\varphi_l} \operatorname{sinc}(\mathrm{FOV}_{\varphi_l} \rho \sin(\Phi - \varphi_l)). \tag{25.2.42}$$

The frequency response $H_\mathrm{d}(\omega_1, \omega_2)$ can be sampled to determine the corresponding DFT coefficients.

Note that as $\mathrm{FOV}_\varphi \to \infty$, one can show that

$$\mathrm{FOV}_\varphi \operatorname{sinc}(\mathrm{FOV}_\varphi \rho \sin(\Phi - \varphi)) \to \delta(\rho \sin(\Phi - \varphi)) = \frac{1}{|\rho|} \delta((\Phi - \varphi) \bmod \pi),$$

as expected from Problem 3.5. Therefore the sinc term is sharply peaked near $\Phi = \varphi$, so we consider the further simplifying approximation

$$w(\varphi; n_0, m_0) \, S_\varphi(\rho, \Phi) \approx w(\Phi; n_0, m_0) \frac{1}{\Delta_\mathrm{R}} |A_\Phi(\rho)|^2 \, \mathrm{FOV}_\varphi \operatorname{sinc}(\mathrm{FOV}_\varphi \rho \sin(\Phi - \varphi)).$$

This approximation suggests

<div style="text-align:right">e,geom,par2,gram,HqA,2</div>

$$H_1(\rho, \Phi) \approx H_2(\rho, \Phi) \triangleq \frac{1}{\Delta_\mathrm{R}\Delta_\mathrm{X}\Delta_\mathrm{Y}} w(\Phi; n_0, m_0) |A_\Phi(\rho)|^2 \sum_{l=1}^{n_\mathrm{A}} \mathrm{FOV}_{\varphi_l} \operatorname{sinc}(\mathrm{FOV}_{\varphi_l} \rho \sin(\Phi - \varphi_l)). \tag{25.2.43}$$

,par2,gram,square **Example 25.2.1** *Consider the case of a square pixel basis* $\beta_0(x, y) = \mathrm{rect}_2(x, y)$ *and a rectangular detector response* $h_\varphi(r) = \frac{1}{w} \mathrm{rect}\left(\frac{r}{w}\right)$ *. By (3.2.25) we have*

$$A_\varphi(\nu) = \mathrm{sinc}(w\nu)\,\Delta_\mathrm{x}\Delta_\mathrm{Y}\,\mathrm{sinc}(\nu\Delta_\mathrm{x}\cos\varphi)\,\mathrm{sinc}(\nu\Delta_\mathrm{Y}\sin\varphi)\,.$$

Fig. 25.2.6 shows an example where samples of $H_\mathrm{d}(\omega_1, \omega_2)$ were computed analytically using (25.2.43) and by the FFT of (25.2.36), for a non-center pixel within a circular FOV. *There is reasonable agreement despite using several approximations in the derivation.*
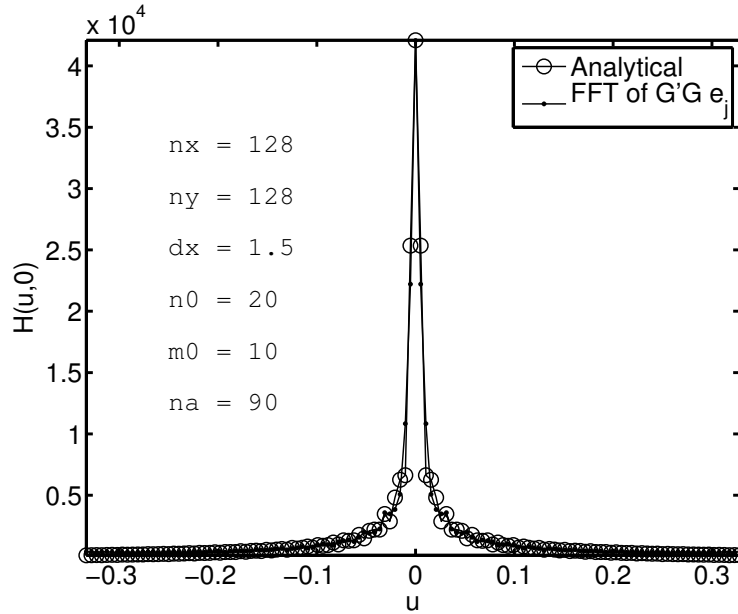
Figure 25.2.6: Comparison of profiles through the analytical local frequency response (25.2.43) and the numerical local frequency response computed via the FFT.

An interesting case arises for a circular field of view, for which FOV is a constant value when $x_0 = y_0 = 0$. In this special case,

$$\sum_{l=1}^{n_\mathrm{A}} \mathrm{FOV}\,\mathrm{sinc}(\mathrm{FOV}\rho\sin(\Phi - \varphi_l)) \quad \approx \quad \frac{n_\mathrm{A}}{\pi} \int_0^\pi \mathrm{FOV}\,\mathrm{sinc}(\mathrm{FOV}\rho\sin(\Phi - \varphi))\,\mathrm{d}\varphi$$

$$= \quad \frac{n_\mathrm{A}}{\pi} \int_0^\pi \mathrm{FOV}\,\mathrm{sinc}(\mathrm{FOV}\rho\sin\varphi)\,\mathrm{d}\varphi\,.$$

This final expression is independent of Φ, so if it also happens that $A_\varphi(\rho)$ is independent of φ, then $H(\rho, \Phi)$ ends up being approximately separable. This is a convenient result because such separability occurs in the pure continuous-space form $w(\Phi)/|\rho|$, and that separability is exploited in both regularization design [86, 87] and in variance predictions [88].

Evidently the following integral is central to the approximations above:

$$f(u) \triangleq \frac{1}{\pi} \int_0^\pi \mathrm{sinc}(u\sin\varphi)\,\mathrm{d}\varphi\,.$$

One empirical approximation for this integral is:

$$f(u) \approx \begin{cases} 1 - 0.6u^2, & 0 \le u < 1 \\ \dfrac{1}{(1 + (\pi u)^2)^{1/2} + \sqrt{2u}\cos(\pi(u + 1/4))}, & u \ge 1. \end{cases}$$

For large u, this function is approximately $1/(\pi u)$. An *open problem* is to find simpler or more accurate approximations and to use them in the context of resolution and noise analysis.

25.3   Non-parallel-beam 2D system models (s,geom,non2)

Now consider a non-parallel 2D geometry, such as a fan-beam geometry, indexed by two coordinates (s, β). We assume there is an invertible relationship (such as (3.9.5)) relating each (s, β) pair to corresponding parallel-beam coordinates $(r(s, \beta), \varphi(s, \beta))$.

Assume that the detector blur is shift invariant, independent of β, and acts only along the s coordinate. The mean projections satisfy

$$\bar{y}_\beta[i_\mathrm{s}] = \int h((i_\mathrm{s} - w_\mathrm{s})\Delta_\mathrm{s} - s)\, p_{\varphi(s,\beta)}(r(s, \beta))\, \mathrm{d}s$$

for $i_\mathrm{s} = 0, \ldots, n_\mathrm{s} - 1$, where Δ_s is the sample spacing in s, w_s is defined akin to w_R, and $p_\varphi(r)$ is the 2D Radon transform of $f(x, y)$.

Using the same basis expansion model (25.2.1) for the object leads to the linear model

$$\bar{y}_\beta[i_\mathrm{s}] = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a_\beta[i_\mathrm{s}; n, m]\, f[n, m],$$

where from (25.2.5) the system matrix elements are

$$a_\beta[i_\mathrm{s}; n, m] = \int h((i_\mathrm{s} - w_\mathrm{s})\Delta_\mathrm{s} - s)\, \Delta_\mathrm{x}\, g\left(\frac{r(s, \beta) - r_{\varphi(s,\beta)}[n, m]}{\Delta_\mathrm{x}}, \varphi(s, \beta)\right) \mathrm{d}s\,.$$

To simplify this expression, note that typically $\beta_0(x, y)$ and h are relatively narrow functions. Define $s_0 = s_0[n, m; \beta]$ to be the solution to

$$r(s_0, \beta) = r_{\varphi(s_0,\beta)}[n, m]\,. \tag{25.3.1}$$

Also define $\varphi_0 \triangleq \varphi(s_0, \beta)$. As a concrete example, in the fan-beam geometry of §3.9, one can show (see Problem 25.1) that the solution satisfies

$$\gamma(s_0) = \gamma_0 \triangleq \arctan\left(\frac{x_\mathrm{c}[n]\cos\beta + y_\mathrm{c}[m]\sin\beta - r_\mathrm{off}}{D_\mathrm{s0} + x_\mathrm{c}[n]\sin\beta - y_\mathrm{c}[m]\cos\beta}\right), \tag{25.3.2}$$

and $\varphi_0 = \beta + \gamma_0$, where $\gamma(s)$ was defined in (3.9.1). In particular, for the equiangular (arc) geometry of 3rd-generation CT, we have $s_0 = D_\mathrm{sd}\gamma_0$, whereas for the equidistant geometry $s_0 = D_\mathrm{sd}\tan\gamma_0$. (Geometrically, s_0 is the location of the projection of the pixel center onto the detector.) As $D_\mathrm{s0} \to \infty$, note that $\varphi_0 \to \beta$ and $s_0 \to x_\mathrm{c}[n]\cos\beta + y_\mathrm{c}[m]\sin\beta - r_\mathrm{off}$, as expected when the fan-beam geometry approaches the parallel-beam case.

For small basis functions and narrow blurs h, the angles of the rays that intersect the basis function are very similar to φ_0 defined above. We thus make our first approximation:

$$a_\beta[i_\mathrm{s}; n, m] \approx \int h((i_\mathrm{s} - w_\mathrm{s})\Delta_\mathrm{s} - s)\, \Delta_\mathrm{x}\, g\left(\frac{r(s, \beta) - r_{\varphi(s,\beta)}[n, m]}{\Delta_\mathrm{x}}, \varphi_0\right) \mathrm{d}s\,.$$

(This approximation is *exact* for circularly symmetric basis functions.)

It follows from (25.3.1) that the first-order Taylor series for $r(s, \beta)$ about s_0 is

$$r(s, \beta) - r_{\varphi(s,\beta)}[n, m] \approx \mu_0\, (s - s_0),$$

where we define the following *magnification factor*:

$$\mu_0 = \mu_0[n, m; \beta] \triangleq \left.\frac{\partial}{\partial s}\left(r(s, \beta) - r_{\varphi(s,\beta)}[n, m]\right)\right|_{s=s_0}.$$

Substituting above yields

$$
\begin{aligned}
a_\beta[i_\mathrm{s}; n, m] \quad &\approx \quad \int h((i_\mathrm{s} - w_\mathrm{s})\Delta_\mathrm{s} - s)\, \Delta_\mathrm{x}\, g\left(\frac{s - s_0}{\Delta_\mathrm{x}/\mu_0}, \varphi_0\right) \mathrm{d}s \\
&= \quad \int h((i_\mathrm{s} - \tau[n, m; \beta])\Delta_\mathrm{s} - s')\, \Delta_\mathrm{x}\, g\left(\frac{s'}{\Delta_\mathrm{x}/\mu_0}, \varphi_0\right) \mathrm{d}s',
\end{aligned}
$$

where $s' = s - s_0$ and

$$\tau[n, m; \beta] \triangleq w_{\mathrm{s}} + s_0[n, m; \beta]/\Delta_{\mathrm{s}}.$$

Defining

$$F(t, \mu; \varphi) \triangleq \int h(t\Delta_{\mathrm{s}} - s) \, \Delta_{\mathrm{x}} \, g\left(\frac{s}{\Delta_{\mathrm{x}}/\mu}, \varphi\right) \mathrm{d}s$$

yields our final approximation for the system matrix elements:

$$a_\beta[i_{\mathrm{s}}; n, m] \approx F(i_{\mathrm{s}} - \tau[n, m; \beta], \, \mu_0[n, m; \beta]; \, \varphi_0[n, m; \beta]) \,.$$

s,geom,non2,fan **Example 25.3.1** *For the fan-beam geometry we have*

$$\mu_0 = [(D_{\mathrm{s}0} \cos \gamma_0 - r_{\mathrm{off}} \sin \gamma_0) - (-x_{\mathrm{c}}[n] \sin \varphi_0 + y_{\mathrm{c}}[m] \cos \varphi_0)] \, \dot{\gamma}(s_0).$$

In the arc case we have $\gamma(s) = s/D_{\mathrm{sd}}$, *so* $\dot{\gamma}(s_0) = 1/D_{\mathrm{sd}}$.
In the flat case, we have $\gamma(s) = \arctan(s/D_{\mathrm{sd}})$, *so* $\dot{\gamma}(s_0) = \frac{1}{D_{\mathrm{sd}}} \frac{1}{1+(s_0/D_{\mathrm{sd}})^2}$.

IRT *See* Gtomo2_strip.m.
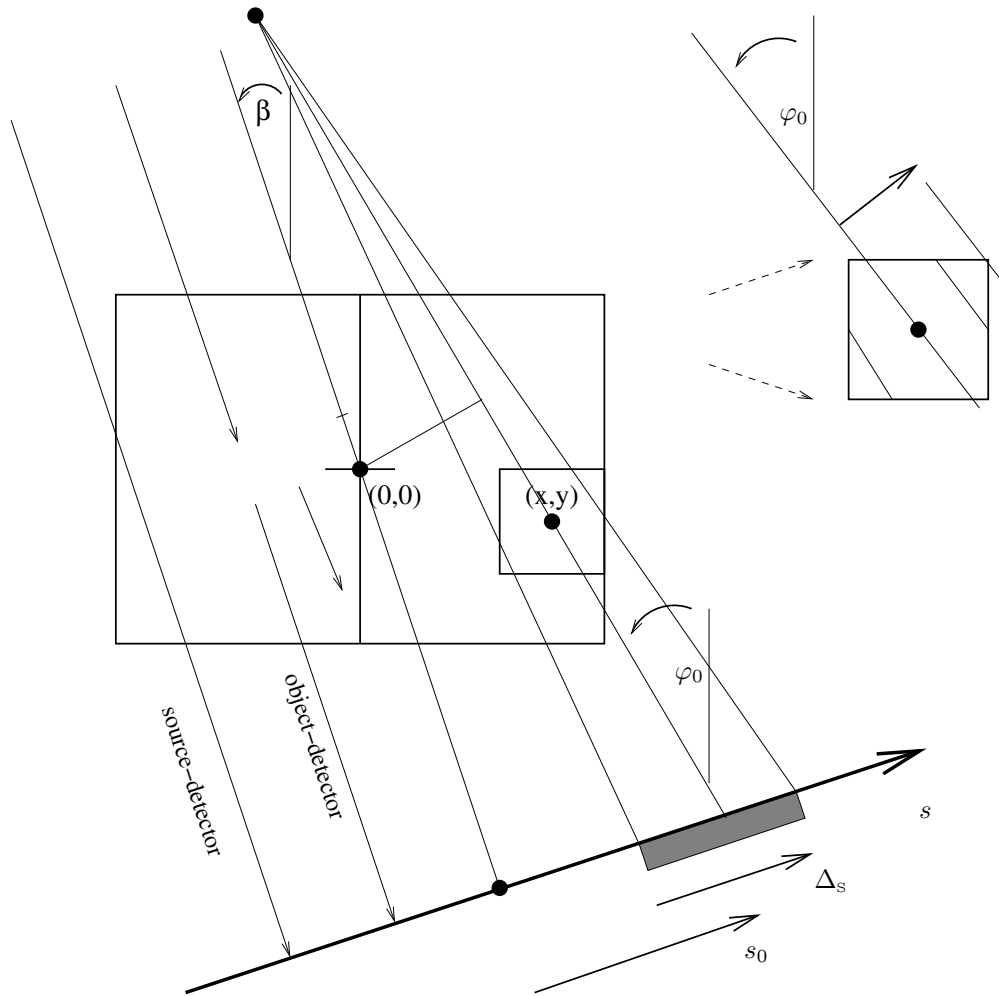
fig,geom,fan2

Figure 25.3.1: Geometry for approximate fan-beam system matrix.

todo: *rectification* [89] todo: shear-warp [90]

25.4   **Fan beam transmission system matrix** (s,geom,fan)

This section describes the geometrical considerations in fan-beam transmission system models, using the coordinates in Fig. 25.4.1.
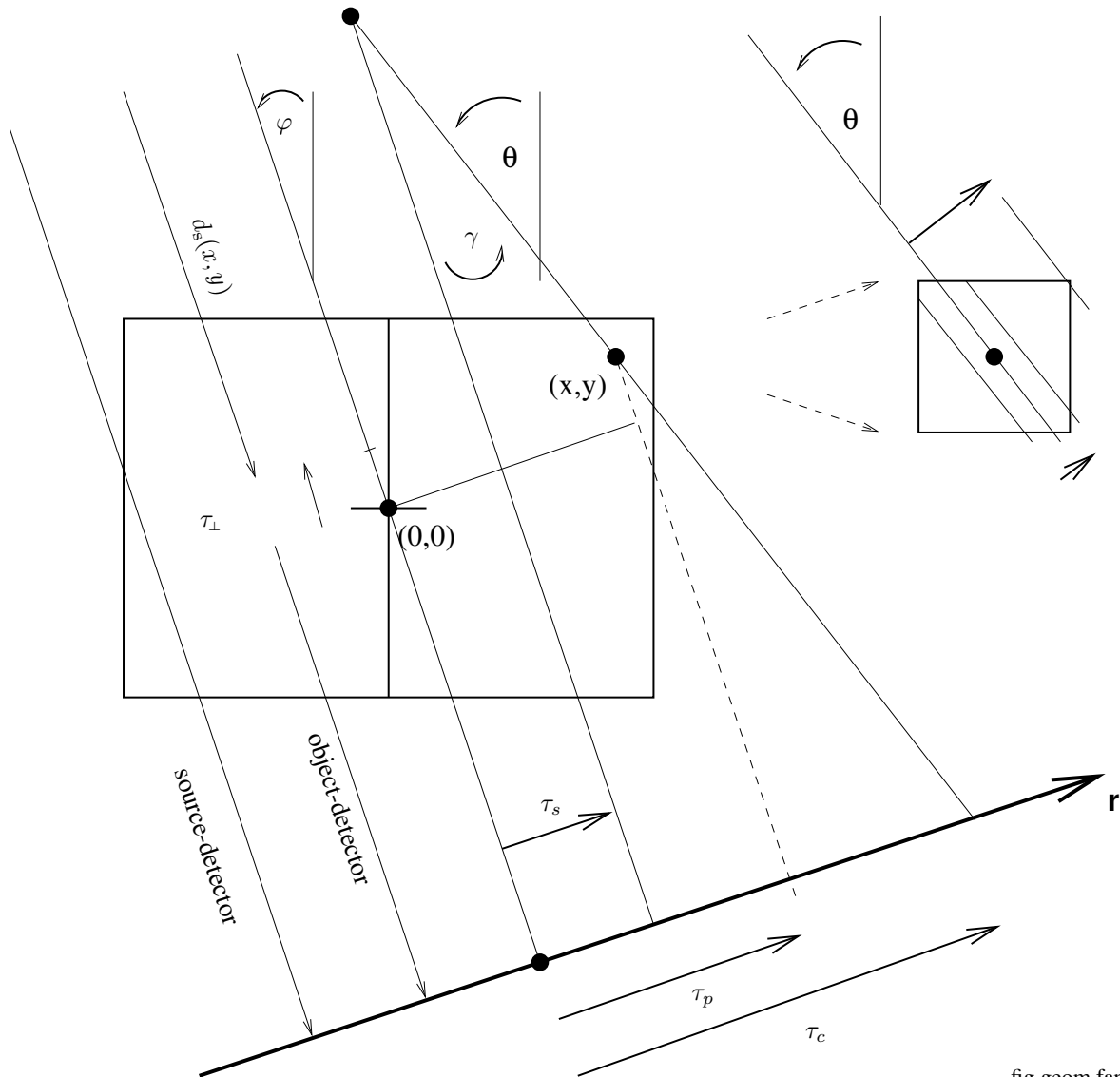
fig,geom,fan,offset

Figure 25.4.1: Coordinate system for "strip-integral" fan-beam projections.

If φ denotes the source rotation angle, and (x, y) denotes the pixel center coordinates, then usual "parallel beam" radial distance is

$$\tau_p(x, y) = x \cos\varphi + y \sin\varphi \,.$$

todo: pbox? The offset in the other direction is

$$\tau_\perp(x, y) = -x \sin\varphi + y \cos\varphi \,.$$

todo: pbox? We must have $\tau_\perp < D_{s0}$ or else the source is inside the object. We must have $-\tau_\perp < D_{0d}$ or else the object is on the other side of the detector. Combining, we see that the distance from the source to the "plane" of the point (x, y) is:

$$d_s(x, y) \triangleq D_{s0} - \tau_\perp(x, y) \in (0, D_{sd}) \,.$$

25.4.1   Bin support

The object magnification factor at (x, y) is

$$m(x, y) = \frac{D_{\mathrm{sd}}}{d_{\mathrm{s}}(x, y)} = \frac{D_{\mathrm{sd}}}{D_{\mathrm{sd}} - D_{0\mathrm{d}} - \tau_\perp(x, y)} = \frac{D_{\mathrm{sd}}}{D_{\mathrm{s}0} + x \sin\varphi - y \cos\varphi}.$$

The incidence angle of the ray from the source through the pixel center located at (x, y) is

$$\gamma(x, y) = \tan^{-1}\left(\frac{\tau_p(x, y) - \tau_s}{d_{\mathrm{s}}(x, y)} \right) = \tan^{-1}\left(m(x, y) \frac{\tau_{ps}(x, y)}{D_{\mathrm{sd}}} \right),$$

where

$$\tau_{ps}(x, y) \triangleq \tau_p(x, y) - \tau_s.$$

(In the context of collimated emission imaging, $|\gamma|$ should have some collimator-limited maximum.)

The "projection angle" θ through the center of this pixel is

$$\theta = \varphi + \gamma = \varphi + \tan^{-1}\left(\frac{\tau_{ps}}{d_{\mathrm{s}}(x, y)} \right).$$

25.4.1.1   Flat detector

The flat-detector fan-beam projection of the pixel center is located at

$$
\begin{aligned}
\tau_c(x, y) &= \tau_s + m(x, y)\left[\tau_p(x, y) - \tau_s \right] \\
&= \tau_s + \frac{D_{\mathrm{sd}}}{d_{\mathrm{s}}(x, y)}\left(x \cos\varphi + y \sin\varphi - \tau_s \right).
\end{aligned}
$$

To determine the *bin support*, we must find the largest and smallest members of the four-element set

$$\left\{ \tau_c\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}, \right) \right\}.$$

Defining

$$\tau_f(s_x, s_y) = \tau_c\left(x + s_x \frac{\Delta_x}{2}, y + s_y \frac{\Delta_x}{2} \right) - \tau_s$$
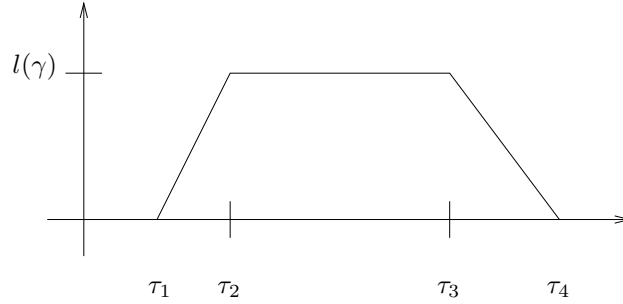
where $s_x, s_y = \pm 1$, we have

$$
\begin{aligned}
\tau_f(s_x, s_y) &= m\left(x + s_x \frac{\Delta_x}{2}, y + s_y \frac{\Delta_x}{2} \right)\left[\tau_p\left(x + s_x \frac{\Delta_x}{2}, y + s_y \frac{\Delta_x}{2} \right) - \tau_s \right] \\
&= \frac{D_{\mathrm{sd}}}{D_{\mathrm{s}0} + (x + s_x \frac{\Delta_x}{2}) \sin\varphi - (y + s_y \frac{\Delta_x}{2}) \cos\varphi}\left[\tau_{ps}(x, y) + \Delta_x \frac{s_x \cos\varphi + s_y \sin\varphi}{2} \right] \\
&= D_{\mathrm{sd}} \frac{\tau_{ps}(x, y) + \frac{1}{2}\Delta_x(s_x \cos\varphi + s_y \sin\varphi)}{d_{\mathrm{s}}(x, y) + \frac{1}{2}\Delta_x(s_x \sin\varphi - s_y \cos\varphi)}.
\end{aligned}
$$

By trying all four combinations of signs and sorting the resulting τ_f's, we can find the four breakpoints of the piecewise-linear approximation of the projection function shown below, having height

$$l(\theta) \triangleq \frac{1}{\max\{|\cos\theta|, |\sin\theta|\}}.$$

todo: pbox?

Since $\tau = (i - w_b - c)\Delta_b$ for $i = 0, \ldots, n_b - 1$, where $w_b = (n_b - 1)/2$ and c denotes the channel offset, the exact *bin support* will be the integer interval

$$\left[1 + \left\lfloor w_b' + \frac{\tau_1}{\Delta_b} - \frac{w}{2\Delta_b} \right\rfloor, \left\lceil w_b' + \frac{\tau_4}{\Delta_b} + \frac{w}{2\Delta_b} \right\rceil \right),$$

where $w_b' = w_b + \frac{\tau_s}{\Delta_b} + c$. Note the open right interval for simplicity of implementation.

25.4.1.2 Arc detector

$$\gamma_c(s_x, s_y) = \gamma\left(x + s_x \frac{\Delta_x}{2}, y + s_x \frac{\Delta_x}{2} \right) = \tan^{-1}\left(\frac{\frac{\tau_{ps}(x,y)}{\Delta_x} + s_x \frac{\cos\varphi}{2} + s_y \frac{\sin\varphi}{2}}{\frac{d_s(x,y)}{\Delta_x} + s_x \frac{\sin\varphi}{2} + s_y \frac{-\cos\varphi}{2}} \right).$$

25.4.1.3 First-order approximation

To save computation (and sorting) it may be reasonable to use first-order Taylor series[6], yielding

$$
\begin{aligned}
\tau_f(s_x, s_y) &= \left(\frac{D_{sd}}{d_s(x,y)} \right) \left(\frac{1}{1 - \frac{1}{2} \frac{s_y \cos\varphi - s_x \sin\varphi}{d_s(x,y)} \Delta_x} \right) \left[\tau_{ps}(x,y) + \Delta_x \frac{s_x \cos\varphi + s_y \sin\varphi}{2} \right] \\
&\approx \left(\frac{D_{sd}}{d_s(x,y)} \right) \left[\tau_{ps} + \Delta_x \left(\frac{s_x \cos\varphi + s_y \sin\varphi}{2} + \frac{1}{2} \frac{s_y \cos\varphi - s_x \sin\varphi}{d_s(x,y)} \tau_{ps} \right) \right] \\
&= m(x,y) \left[\tau_{ps} + \Delta_x (s_x h_x + s_y h_y) \right],
\end{aligned}
$$

where

$$
\begin{aligned}
h_x &\triangleq \frac{1}{2} \cos\varphi - \frac{1}{2} \frac{\tau_{ps}}{d_s(x,y)} \sin\varphi \\
h_y &\triangleq \frac{1}{2} \sin\varphi + \frac{1}{2} \frac{\tau_{ps}}{d_s(x,y)} \cos\varphi.
\end{aligned}
$$

Idris: analyze the accuracy of the above approximation.

Thus, the maximum and minimum values of τ_f are (approximately) located at

$$\tau_{out}^{\pm} \triangleq \tau_s + m(x,y) \left[\tau_p - \tau_s \pm \Delta_x h(x,y) \right]$$

todo: pbox? where

$$h(x,y) \triangleq |h_x(x,y)| + |h_y(x,y)|.$$

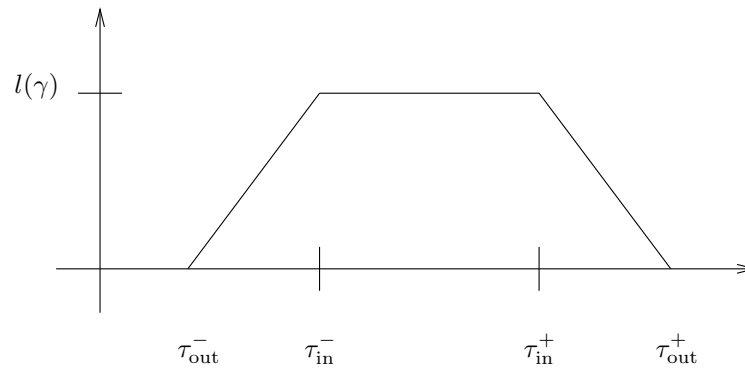Thus, the half-width of the trapezoid shown below is

$$m(x,y)\Delta_x h(x,y).$$

The projection is approximately a trapezoid of height $l(\gamma)$ defined above

---

[6]$f(\Delta) = \frac{\tau + \beta\Delta}{1 - \alpha\Delta} \approx \tau + (\beta + \alpha\tau)\Delta.$

and with inner breakpoints at

$$\tau_{\text{in}}^{\pm} \triangleq \tau_s + m(x, y) \left[\tau_p - \tau_s \pm \Delta_x \big| |h_{\text{x}}| - |h_{\text{y}}| \big| \right].$$

todo: pbox?

25.4.1.4   Old corner discussion

In particular, to find the projected position τ_c^+ of the corner of the pixel (which is an extreme point if $0 \le \theta \le \pi/2$) we see

$$
\begin{aligned}
\tau_c^+ &\triangleq \tau_c\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) \\
&= \tau_s + m\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) \tau_{ps}\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) \\
&= \tau_s + \frac{D_{sd}}{D_{s0} + (x \pm \frac{\Delta_x}{2})\sin\varphi - (y \pm \frac{\Delta_x}{2})\cos\varphi}\left(\tau_{ps}(x,y) + \Delta_x \frac{\cos\varphi + \sin\varphi}{2}\right) \\
&= \tau_s + \frac{D_{sd}}{d_s(x,y) - \Delta_x \chi(\varphi)}\left[\tau_{ps}(x,y) + \Delta_x d_{\max}(\varphi)\right],
\end{aligned}
$$

where

$$
d_{\max}(\varphi) \triangleq \frac{|\cos\varphi| + |\sin\varphi|}{2}
$$

and

$$
\chi(\varphi) \triangleq \cos\varphi - \sin\varphi = \sqrt{2}\cos(\varphi + \pi/4).
$$

In the usual case where the pixel size is much smaller than D_{s0}, the variation of the magnification factors over the area of a pixel will be quite small, so at first glance it would seem reasonable to replace $m\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right)$ with simply $m(x, y)$, yielding the approximation:

$$
\tau_c^+ \approx \tau^+ \triangleq \tau_s + m(x,y)\left[\tau_p(x,y) + \Delta_x d_{\max}(\varphi) - \tau_s\right].
$$

Although this seems reasonable, it turns out to be unacceptably inaccurate towards the edges of a large field-of-view. Since

$$
\begin{aligned}
m\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) - m(x,y) &= \frac{D_{sd}}{d_s(x,y) - \Delta_x \chi(\varphi)} - \frac{D_{sd}}{d_s(x,y)} \\
&= \frac{D_{sd}}{[d_s(x,y) - \Delta_x \chi(\varphi)]\,[d_s(x,y)]}\Delta_x \chi(\varphi),
\end{aligned}
$$

the approximation error resulting from the above approximation would be:

$$
\begin{aligned}
\tau_c^+ - \tau^+ &= \left[m\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) - m(x,y)\right]\left[\tau_{ps}(x,y) + \Delta_x d_{\max}(\varphi)\right] \\
&= \left(\frac{D_{sd}}{d_s(x,y)}\right)\left(\frac{\tau_{ps}(x,y) + \Delta_x d_{\max}(\varphi)}{d_s(x,y) - \Delta_x \chi(\varphi)}\right)\Delta_x \chi(\varphi).
\end{aligned}
$$

The parenthesized terms are of order unity for large $|\tau_p|$, so the overall error is approximately Δ_x. Since the pixel size Δ_x is usually approximately the same as the detector sample spacing, an error of order Δ_x is unacceptable.

25.4.1.5 Old incomplete reasoning...

When is this latter approximation reasonable? Define

$$m(x, y; \Delta_x) = m\left(x \pm \frac{\Delta_x}{2}, y \pm \frac{\Delta_x}{2}\right) = \frac{D_{\text{sd}}}{D_{\text{s0}} + (x \pm \frac{\Delta_x}{2})\sin\varphi - (y \pm \frac{\Delta_x}{2})\cos\varphi},$$

so $m(x, y) = m(x, y; 0)$.

$$\frac{\partial}{\partial \Delta_x} m(x, y; \Delta_x) = \frac{D_{\text{sd}}}{\left[D_{\text{s0}} + (x \pm \frac{\Delta_x}{2})\sin\varphi - (y \pm \frac{\Delta_x}{2})\cos\varphi\right]^2} (\cos\varphi - \sin\varphi).$$

Thus

$$m(x, y; \Delta_x) \approx m(x, y) + \frac{m^2(x, y)}{D_{\text{sd}}} \Delta_x.$$

We can ignore the second term if it is much much smaller than the first term, *i.e.*, if $\frac{m^2(x,y)}{D_{\text{sd}}}\Delta_x \ll m(x, y)$ or equivalently

$$\Delta_x \ll \frac{D_{\text{sd}}}{m(x, y)},$$

which means that the pixel size is very small relative to the distance D_{sd}.

 need to analyze this more!

 this is incomplete because it ignores how large the terms are that are being magnified!

The projected width of the pixel at angle θ is (approximately) $2\Delta_x d_{\text{max}} m$. Thus

$$\tau_\pm = \tau_c \pm \Delta_x d_{\text{max}}(\varphi)m,$$

Thus:

$$i_\pm = w_b + \frac{\tau_\pm}{\Delta_b} = w_b + \frac{\tau_c \pm \Delta_x d_{\text{max}} m}{\Delta_b} = w_b + \frac{\tau_s + m(\tau_p - \tau_s) \pm \Delta_x d_{\text{max}} m}{\Delta_b}$$

$$= \left[w_b + \frac{\tau_s + m(\tau_p - \tau_s)}{\Delta_b}\right] \pm m\frac{d_{\text{max}}}{\Delta_b/\Delta_x}$$

The bracketed term is `ib_cent`, and the last term is `db`.
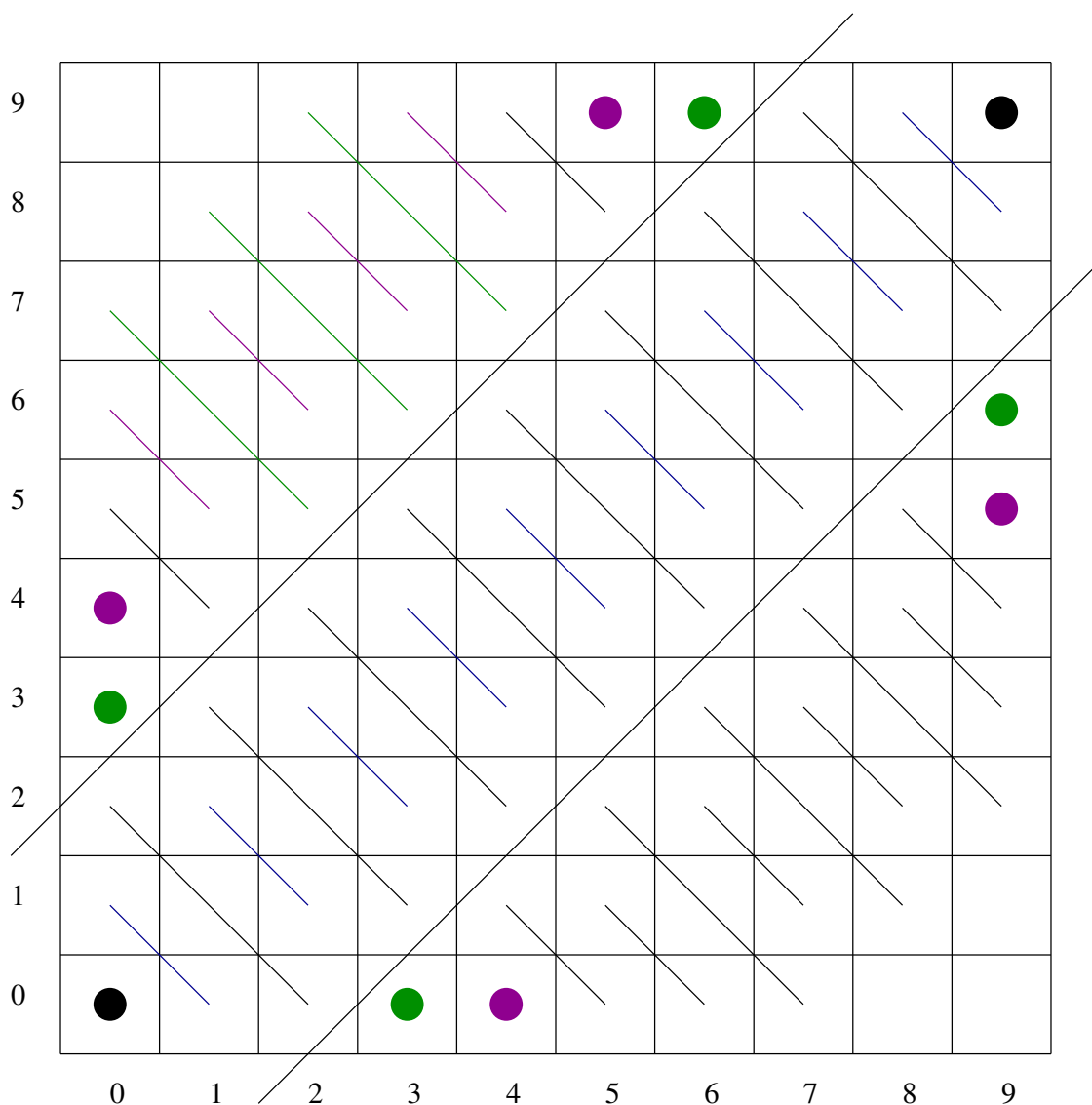
 For a given ray, we have

$$\tau_c^\pm = (i - w_b)\Delta_b \pm w/2.$$

We need to look at this in the non-magnified τ_p coordinates to use `ray_int`, using $\tau_c = \tau_s + m(\tau_p - \tau_s)$.

$$\tau_p^\pm = \tau_s + \frac{\tau_c^\pm - \tau_s}{m} = \tau_s + \frac{(i - w_b)\Delta_b \pm w/2 - \tau_s}{m} = \left[\tau_s + \frac{(i - w_b)\Delta_b - \tau_s}{m}\right] \pm \frac{w}{2m}.$$
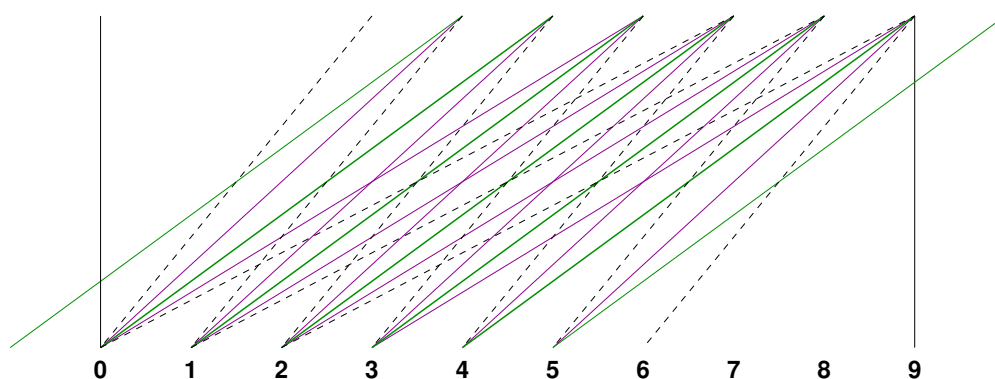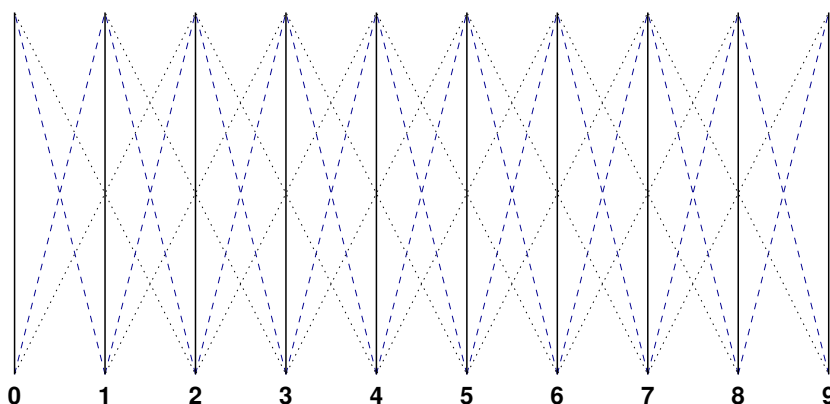
The bracketed term is `tau` in `weight_value_13` and the last term is `delta_b`.

25.5   3D PET mashing/span/tilt (s,geom,span,3d)

Michelogram

$nz = 2\ nring - 1$

in 1st segment: $nv = nz - 2 * (span+1)/2 = nz - (span+1)$

for 2nd segment (and each additional segment after that), we lose another 2*span worth: $nv = nz - (span+1) - 2*span = nz - 3\ span - 1$

for Exact (nring=24, nz=47, span=7): $nv1 = 47 - (7+1) = 39$, $nv2 = 47 - 3(7)-1 = 25$.

$d_c = 0.675 = 2d_z$, $\tan\theta = span\ d_c/d$

25.6   **todo** (s,geom,todo)

recursive hierarchical approach that is $O(N \log n \log \log n)$ [91]

correction for system model mismatch resulting from deliberate approximations [92]

[93] exact volume of intersection of a pyramidal ray with a cubic voxel for cone-beam CT

[94] exact volume of intersection of a tube with a cubic voxel for PET

3d cone-beam projection via shear-warp factorization, [95, 96] also called *rectification* of the views [97].

todo: Unser says that tensor-product cubic b-splines are almost circularly symmetric and hence suitable for fast reprojection ala blobs.

[19] Mojette angular sampling (rational m/n arctan?)

[98] D W Wilson H H Barrett The effects of incorrect modeling on noise and resolution properties of ML-EM images

[99] Irani-Peleg algorithm converges ? even with what might be called a mismatched "projector" ?

[25] Deman and Basu's clever distance-based projector. Need to examine accuracy of its approximation!

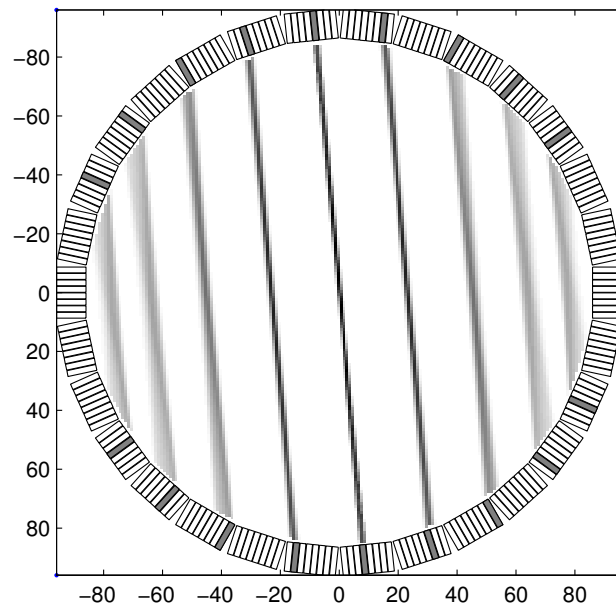[100] intermediate radiance distribution

[101] clever form of partial shift invariance for thick-septa collimators, boils down to $\int h_i(\vec{x}) b_j(\vec{x})$ needed computed only for some i,j combinations!

[102] tabulate projection of basis (splatting) for subsequent table lookup!

[103] describes a "2D+1" model for SPECT that uses a depth-dependent gaussian blur in the transaxial plane, with a depth-*independent* blur in the z-direction. this could be implemented easy using a combination of the "system 12" projector followed by simple convolution in z. may be useful for some approximate analysis too? see sys12 subdirectory

[104] complicated pinhole

[105] discussion of [106] interleaving approach in circular PET system, with pictures.

Figure 25.6.1: Stayman's rat PET system model

[107] On various approximations for the projectors in iterative reconstruction algorithms for 3D PET @an piecewise linear approximation for reprojection. interesting rotation-invariance argument!

25.7   **Photon/detector interaction: Planar detector** (s,emis,plane,cauchy)

A point source at a distance d from a planar detector emits photons isotropically (within a plane for simplicity). We are interested in the *intensity* "recorded" on the detector.
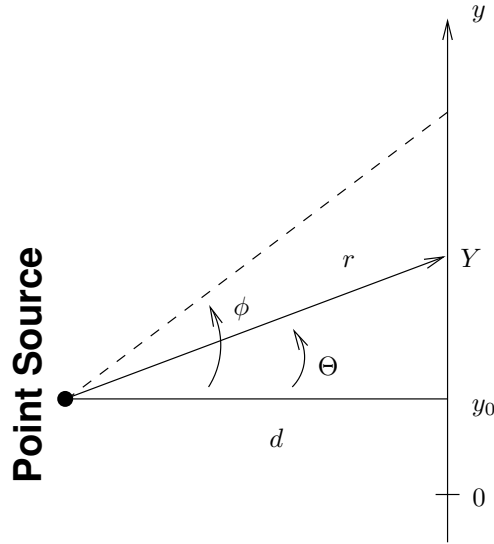
The angle of emission is $\Theta \sim \text{Uniform}[-\pi, \pi]$.

Some detectors only accept photons that are incident within a certain *critical angle* ϕ, where $0 < \phi \leq \pi/2$. (For other detectors we can simply let $\phi = \pi/2$ in our final expressions.)

Provided $|\Theta| \leq \phi$, the photon strikes the detector at position

$$Y = y_0 + d \tan \Theta.$$

We define the intensity on the detector to be

$$I(y) = f_{Y|C}(y|C)P(C)$$

where $f_{Y|C}(y|C)$ is the conditional *probability density function (pdf)* of Y and where C is the event

$$C = [|\Theta| \leq \phi].$$

Interpretation of intensity: if the source emits a photon, the probability that a photon emitted from the source will strike the detector between y_1 and y_2 is:

$$P[y_1 \leq Y \leq y_2] = \int_{y_1}^{y_2} I(y) \, dy.$$

From elementary probability:

$$f_\Theta(\theta) = \begin{cases} \frac{1}{2\pi}, & |\theta| \leq \pi \\ 0, & \text{otherwise,} \end{cases} \quad \text{and } f_{\Theta|C}(\theta|C) = \frac{f_\Theta(\theta)}{P(C)} \, 1_{\{|\theta| \leq \phi\}} = \begin{cases} \frac{1}{2\phi}, & |\theta| \leq \phi \\ 0, & \text{otherwise,} \end{cases}$$

where $P(C) = (2\phi)/(2\pi) = \phi/\pi$.

For $|\Theta| \leq \phi$ there is a monotone differentiable relationship between Y and Θ. Thus this is a simple transformation of random variables problem. From elementary transformation of random variables:

$$f_{Y|C}(y|C) = f_{\Theta|C}(\theta|C) \left| \frac{d\theta}{dy} \right| \Bigg|_{\theta = \tan^{-1}[(y-y_0)/d]}.$$

Since $d/dx \tan^{-1} x = 1/(1 + x^2)$,

$$\frac{d\theta}{dy} = \frac{d}{d^2 + (y - y_0)^2} = \frac{\cos^2 \theta}{d} = \frac{\cos \theta}{r},$$

which can be interpreted as a $1/r$ falloff of intensity (since circumference of circle is $2\pi r \propto r$) times an *obliquity factor* $\cos\theta$ "due to angle of incidence between photons and detector". (More on that later.)

Thus

$$f_{Y|C}(y|C) = \begin{cases} \frac{1}{2\phi}\frac{d}{d^2+(y-y_0)^2}, & |y-y_0| \leq d\tan\phi \\ 0, & \text{otherwise.} \end{cases}$$

Alternative derivation:

$$F_{Y|C}(y|C) = P[Y \leq y \,|\, C] = P[y_0 + d\tan\Theta \leq y \,|\, C] = P[\Theta \leq \tan^{-1}((y-y_0)/d) \,|\, C]$$

$$= F_{\Theta|C}(\tan^{-1}((y-y_0)/d)|C)$$

then take derivative of both sides w.r.t. y.

Mean:

$$E[Y|C] = \int_{-\infty}^{\infty} y f_{Y|C}(y|C)\,dy = \int_{y_0-d\tan\phi}^{y_0+d\tan\phi} y\frac{1}{2\phi}\frac{d}{d^2+(y-y_0)^2}\,dy = y_0$$

where the integral is y_0 due to the symmetry of the pdf $f_{Y|C}(y|C)$. However, for $\phi = \pi/2$ the mean is *undefined*. (For $\phi = \pi/2$ the pdf is a *Cauchy* distribution, for which the mean is not defined.)

Variance (a measure of the spread of the PSF):

$$\mathsf{Var}\{Y|C\} = E[(Y - E[Y|C])^2|C] = \int_{-\infty}^{\infty} (y-y_0)^2 f_{Y|C}(y|C)\,dy$$

$$= \int_{y_0-d\tan\phi}^{y_0+d\tan\phi} d[(y-y_0)/d]^2 \frac{1}{2\phi}\frac{1}{1+[(y-y_0)/d]^2}\,dy$$

$$= d^2\int_{-\tan\phi}^{+\tan\phi} \frac{1}{2\phi}x^2\frac{1}{1+x^2}\,dx = \frac{d^2}{2\phi}(x - \tan^{-1}x)\Big|_{-\tan\phi}^{+\tan\phi} = \frac{d^2}{\phi}(\tan\phi - \phi).$$

where $x = (y-y_0)/d$ and $dy = d\,dx$. The variance approaches infinity as $\phi \to \pi/2$.

x,photon **Example 25.7.1** *Suppose $n = 1000$ photons are emitted. Let X denote the number that strike detector between y_1 and y_2. Find the PMF of X.*
The probability that a given photon strikes between y_1 and y_2 is $p = \int_{y_1}^{y_2} I(y)\,dy$. Assuming photons are independent, X is a Binomial *random variable:*

$$P[X = k] = \binom{n}{k} p^k q^{n-k}$$

where $q = 1 - p$. Mean of X is $E[X] = np$, variance is $\sigma_X^2 = npq$.

25.8 Photon/detector interaction: General detector (s,emis,photon,general)

Again consider the 2D case for simplicity, but now suppose that the emitter is "surrounded" by detecting material with linear attenuation coefficient $\mu(\vec{x})$.

Define D to be the event that an emitted photon is detected, *i.e.*, interacts with the detecting material.

$$P\{D|\Theta = \theta\} = 1 - e^{-\int_0^\infty \mu(\ell\vec{x}_\theta)\,d\ell} \triangleq p_\theta$$

where, in 2D:

$$\vec{x}_\theta \triangleq [\cos\theta, \sin\theta]$$

is a *direction cosine* ???

In 2D

$$f_\Theta(\theta) = \begin{cases} 1/2\pi, & |\theta| \leq \pi \\ 0, \text{otherwise} \end{cases}$$

and

$$P\{D\} = \int P\{D|\Theta = \theta\}\, f_\Theta(\theta)\, d\theta = \frac{1}{2\pi} \int_{-\pi}^\pi p_\theta\, d\theta$$

so by Bayes' rule:

$$f_\Theta(\theta|D) = \frac{P\{D|\Theta = \theta\}\, f_\Theta(\theta)}{P\{D\}} = \frac{p_\theta/2\pi}{\frac{1}{2\pi}\int_{-\pi}^\pi p_\phi\, d\phi}.$$

Let $\vec{X}$ denote the interaction point for a detected emission (and for completeness, define $\vec{X} = \vec{0}$ for a non-detected emission.

We are most interested in

$$f_{\vec{X}}(\vec{x}|D)$$

Let $R = \left\|\vec{X}\right\|$ denote the distance that the photon first interacts with the detector (*e.g.*, is absorbed or scattered), or $R = 0$ for undetected photons.

$$P\{R \leq r|\Theta = \theta, D\} = \frac{1 - e^{-\int_0^r \mu(\ell\vec{x}_\theta)\,d\ell}}{p_\theta}$$

so

$$f_R(r|\theta, D) = \frac{\mu(r\vec{x}_\theta)\, e^{-\int_0^r \mu(\ell\vec{x}_\theta)\,d\ell}}{p_\theta}$$

and

$$P\{R \leq r|D\} = \int P\{R \leq r|\Theta = \theta, D\}\, f_\Theta(\theta|D)\, d\theta = \frac{\frac{1}{2\pi}\int_{-\pi}^\pi \left[1 - e^{-\int_0^r \mu(\ell\vec{x}_\theta)\,d\ell}\right] d\theta}{\frac{1}{2\pi}\int_{-\pi}^\pi p_\phi\, d\phi},$$

for $r \geq 0$, so

$$f_R(r|D) = \frac{\frac{1}{2\pi}\int_{-\pi}^\pi \left[\mu(r\vec{x}_\theta)\, e^{-\int_0^r \mu(\ell\vec{x}_\theta)\,d\ell}\right] d\theta}{\frac{1}{2\pi}\int_{-\pi}^\pi p_\phi\, d\phi}$$

Joint density:

$$f(r,\theta|D) = f_R(r|\theta, D)f(\theta|D) = \frac{\mu(r\vec{x}_\theta)\, e^{-\int_0^r \mu(\ell\vec{x}_\theta)\,d\ell}}{\int_{-\pi}^\pi p_\phi\, d\phi}.$$

From this we can find $f_{\vec{X}}(\vec{x}|D)$ by pdf transformation with appropriate Jacobian.

25.9 SPECT (s,geom,spect)

In *single photon emission computed tomography* (*SPECT*), the system matrix $\boldsymbol{A}$ usually is less sparse than in X-ray imaging because the collimator causes a system response that gets wider as distance to the collimator increases. Therefore, usually it is impractical to precompute and store all the elements a_{ij} of the system matrix. Instead, matrix multiplication by $\boldsymbol{A}$ or $\boldsymbol{A}'$ are performed using forward and back-projection subroutines. Numerous forward and back-projection methods have been proposed for SPECT. Of these, a particularly convenient approach for SPECT with a parallel hole collimator is based on image *rotation* operations, *e.g.*, [108–110]. To compute $\boldsymbol{A}\boldsymbol{x}$ with this type of approach, one first uses image interpolation to rotate the 3D image $\boldsymbol{x}$ and the corresponding 3D *attenuation map* transaxially (around z) by the angle corresponding to one of the projection views. One then convolves each y, z plane at each distance from the detector with the appropriate detector/collimator response. Often a 2D FFT is used for each such convolution. Finally one sums the blurred image along the x direction, accounting for the cumulative effects of *attenuation* to form the projection view at that angle. This uses a *central ray* approximation for the effects of attenuation. Multiple projection views can be computed simultaneously.

Many of the "high quality" methods for image rotation use basis functions that have negative values [111]. These are problematic if used for ML algorithms for SPECT reconstruction because those algorithms generally are derived assuming that $a_{ij} \geq 0$.

Most rotation methods use a continuous-space interpolation model of the form (25.2.1). However, such methods usually use point sampling of the rotated image model rather than reprojecting the rotated model back into a basis-function representation of the form (25.2.1). Using such a representation both before and after rotating is an interesting *open problem*.

For a possibly more accurate approximation, one could combine the rotation method with a partial system matrix where one precomputes and stores only those elements corresponding to a single projection angle, *e.g.*, $\varphi = 0$.

For SPECT with fan-beam or cone-beam collimators, more sophisticated methods are required, *e.g.*, [109] [112]. Pinhole collimation presents a further challenge, *e.g.*, [113].

25.10 Problems (s,geom,prob)

Problem 25.1 *Prove (25.3.2).*

Problem 25.2 *Often a gaussian shape is used to model the PSF of PET systems. Determine the "blurred foot-print" function $F_\varphi(r; \Delta_\mathrm{x})$ in (25.2.10) for the case of square pixels for $\beta_0(x, y)$ as in (25.2.22) and a gaussian PSF $h_\varphi(r) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x/\sigma)^2/2}$. Compare numerically the blurred footprint for a gaussian PSF to the blurred footprint for a rectangular PSF having the same FWHM as the gaussian, for various values of σ.* (Need typed.)

Bibliography

peters:77:ctw [1] T. M. Peters and R. M. Lewitt. Computed tomography with fan beam geometry. *J. Comp. Assisted Tomo.*, 1(4):429–36, October 1977.

natterer:93:sif [2] F. Natterer. Sampling in fan beam tomography. *SIAM J. Appl. Math.*, 53(2):358–80, April 1993.

hsieh:99:rbr [3] J. Hsieh. Reconstruction bias resulting from weighted projection and iso-center misalignment. In *Proc. SPIE 3661, Medical Imaging 1999: Image. Proc.*, pages 442–9, 1999.

lariviere:04:saa [4] P. J. LaRiviere and X. Pan. Sampling and aliasing consequences of quarter-detector offset use in helical CT. *IEEE Trans. Med. Imag.*, 23(6):738–49, June 2004.

hanson:85:lbf [5] K. M. Hanson and G. W. Wecksung. Local basis-function approach to computed tomography. *Appl. Optics*, 24(23):4028–39, December 1985.

miller:92:fml [6] T. R. Miller and J. W. Wallis. Fast maximum-likelihood reconstruction. *J. Nuc. Med.*, 33(9):1710–11, September 1992.

cho:90:iaa [7] Z. H. Cho, C. M. Chen, and S. Y. Lee. Incremental algorithm—A new fast backprojection scheme for parallel geometries. *IEEE Trans. Med. Imag.*, 9(2):207–17, June 1990.

chen:96:afi [8] C-M. Chen, Z-H. Cho, and C-Y. Wang. A fast implementation of the incremental backprojection algorithms for parallel beam geometries. *IEEE Trans. Nuc. Sci.*, 43(6):3328–34, December 1996.

schwinger:86:awc [9] R. B. Schwinger, S. L. Cool, and M. A. King. Area weighted convolutional interpolation for data reprojection in single photon emission computed tomography. *Med. Phys.*, 13(3):350–5, May 1986.

herman:72:tdm [10] G. T. Herman. Two direct methods for reconstructing pictures from their projections: a comparative study. *Comp. Graphics and Im. Proc.*, 1(2):123–44, August 1972.

oppenheim:74:maa [11] B. E. Oppenheim. More accurate algorithms for iterative 3-dimensional reconstruction. *IEEE Trans. Nuc. Sci.*, 21(3):72–7, June 1974.

katz:78:qou [12] M. Katz. Questions of uniqueness and resolution in reconstruction from projections, 1978. Lecture Notes in Biomathematics, 26, Springer-Verlag, 1978.

donohue:89:asa [13] K. D. Donohue and J. Saniie. A scanning and sampling scheme for computationally efficient algorithms of computer tomography. *IEEE Trans. Acoust. Sp. Sig. Proc.*, 37(3):402–14, March 1989.

brady:93:nmp [14] S. D. Brady. *New mathematical programming approaches to the problem of image reconstruction from projections*. PhD thesis, Dept. of Operations Research, Stanford, June 1993.

nuyts:94:eso [15] J. Nuyts, P. Dupoint, C. Schieper, and L. Mortelmans. Efficient storage of the detection probability matrix for reconstruction in PET. *J. Nuc. Med. (Abs. Book)*, 35(5):187, May 1994.

schmidlin:94:iii [16] P. Schmidlin. Improved iterative image reconstruction using variable projection binning and abbreviated convolution. *Eur. J. Nuc. Med.*, 21(9):930–6, September 1994.

guedon:97:mta [17] J. V. Guedon and N. Normand. Mojette transform: applications for image analysis and coding. In *Proc. SPIE 3024, Vis. Comm. Im. Proc.*, pages 873–84, 1997.

guedon:04:msc [18] J. Guedon, M. Servieres, S. Beaumont, and N. Normand. Medical software control quality using the 3D Mojette projector. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, volume 1, pages 836–9, 2004.

subirats:04:aao [19] P. Subirats, M. Servières, N. Normand, and J. Guédon. Angular assessment of the Mojette filtered back projection. In *Proc. SPIE 5370, Medical Imaging 2004: Image Proc.*, pages 1951–60, 2004.

guedon:05:tmt [20] JP. Guédon and N. Normand. The Mojette transform: The first ten years. In E. Andres et al., editors, *Discrete Geometry for Computer Imagery*, pages 79–. Springer-Verlag, 2005. LNCS 3429.

elbakri:03:eaa [21] I. A. Elbakri and J. A. Fessler. Efficient and accurate likelihood for iterative image reconstruction in X-ray computed tomography. In *Proc. SPIE 5032, Medical Imaging 2003: Image Proc.*, pages 1839–50, 2003.

siddon:85:fco [22] R. L. Siddon. Fast calculation of the exact radiological path for a three-dimensional CT array. *Med. Phys.*, 12(2):252–5, March 1985.

deman:02:ddp [23] B. De Man and S. Basu. Distance-driven projection and backprojection. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 1477–80, 2002.

deman:04:ddp [24] B. De Man and S. Basu. Distance-driven projection and backprojection in three dimensions. *Phys. Med. Biol.*, 49(11):2463–75, June 2004.

deman::ahs [25] B. De Man and S. Basu. A high-speed, low-artifact approach to projection and backprojection, 2002.

deman:04:rab [26] B. De Man, K. Bernard, and S. Basu. Reprojection and backprojection methods and algorithms for implementation thereof, 2004. US Patent 6,724,856.

joseph:82:aia [27] P. M. Joseph. An improved algorithm for reprojecting rays through pixel images. *IEEE Trans. Med. Imag.*, 1(3):192–6, November 1982.

sunnegardh:07:ana [28] J. Sunnegårdh and P-E. Danielsson. A new anti-aliased projection operator for iterative CT reconstruction. In *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, pages 124–7, 2007.

crawford:86:sfr [29] C. R. Crawford. System for reprojecting images using transform techniques, 1986. US Patent 4,616,318. Filed 1983-6-7. Elscint.

crawford:88:hsr [30] C. R. Crawford, J. G. Colsher, N. J. Pelc, and A. H. R. Lonn. High speed reprojection and its applications. In *Proc. SPIE 914, Med. Im. II: Im. Formation, Detection, Processing, and Interpretation*, pages 311–8, 1988.

stearns:87:tdi [31] C. W. Stearns, D. A. Chesler, and G. L. Brownell. Three-dimensional image reconstruction in the Fourier domain. *IEEE Trans. Nuc. Sci.*, 34(1):374–8, February 1987.

stearns:90:air [32] C. W. Stearns, D. A. Chesler, and G. L. Brownell. Accelerated image reconstruction for a cylindrical positron tomograph using Fourier domain methods. *IEEE Trans. Nuc. Sci.*, 37(2):773–7, April 1990.

natterer:86 [33] F. Natterer. *The mathematics of computerized tomography*. Teubner-Wiley, Stuttgart, 1986.

fessler:01:amm [34] J. A. Fessler and B. P. Sutton. A min-max approach to the multidimensional nonuniform FFT: Application to tomographic image reconstruction. In *Proc. IEEE Intl. Conf. on Image Processing*, volume 1, pages 706–9, 2001.

sutton:01:amm [35] B. P. Sutton, J. A. Fessler, and D. Noll. A min-max approach to the nonuniform N-D FFT for rapid iterative reconstruction of MR images. In *Proc. Intl. Soc. Mag. Res. Med.*, page 763, 2001.

fessler:02:tir [36] J. A. Fessler and B. P. Sutton. Tomographic image reconstruction using the nonuniform FFT. In *SIAM Conf.*

Imaging Sci., Abstract Book, page 21, 2002.

fessler:03:nff [37] J. A. Fessler and B. P. Sutton. Nonuniform fast Fourier transforms using min-max interpolation. *IEEE Trans. Sig. Proc.*, 51(2):560–74, February 2003.

matej:01:3fd [38] S. Matej and R. M. Lewitt. 3D-FRP: direct Fourier reconstruction with Fourier reprojection for fully 3-D PET. *IEEE Trans. Nuc. Sci.*, 48(4-2):78–1385, August 2001.

matej:01:po3 [39] S. Matej, M. E. Daube-Witherspoon, and J. S. Karp. Performance of 3D RAMLA with smooth basis functions on fully 3D PET data. In *f3d01*, pages 193–6, 2001.

mersereau:74:dro [40] R. M. Mersereau and A. V. Oppenheim. Digital reconstruction of multidimensional signals from their projections. *Proc. IEEE*, 62(10):1319–38, October 1974.

mersereau:74:rms [41] R. M. Mersereau. Recovering multidimensional signals from their projections. *Comp. Graphics and Im. Proc.*, 1(2):179–85, October 1974.

mersereau:76:dft [42] R. M. Mersereau. Direct Fourier transform techniques in 3-D image reconstruction. *Computers in Biology and Medicine*, 6(4):247–58, October 1976.

stark:81:aio [43] H. Stark, J. W. Woods, I. Paul, and R. Hingorani. An investigation of computerized tomography by direct Fourier inversion and optimimum interpolation. *IEEE Trans. Biomed. Engin.*, 28(7):496–505, July 1981.

stark:81:dfr [44] H. Stark, J. W. Woods, I. Paul, and R. Hingorani. Direct Fourier reconstruction in computer tomography. *IEEE Trans. Acoust. Sp. Sig. Proc.*, 29(2):237–44, April 1981.

lewitt:83:rat [45] R. M. Lewitt. Reconstruction algorithms: transform methods. *Proc. IEEE*, 71(3):390–408, March 1983.

niki:83:ahs [46] N. Niki, R. T. Mizutani, Y. Takahasi, and T. Inouye. A high-speed computerized tomography image reconstruction using direct two-dimensional Fourier transform method. *Syst. Comput. Controls*, 14(3):56–65, 1983.

natterer:85:fri [47] F. Natterer. Fourier reconstruction in tomography. *Numerische Mathematik*, 47(3):343–53, September 1985.

osullivan:85:afs [48] J. D. O'Sullivan. A fast sinc function gridding algorithm for Fourier inversion in computer tomography. *IEEE Trans. Med. Imag.*, 4(4):200–7, December 1985.

pan:83:acs [49] S. X. Pan and A. C. Kak. A computational study of reconstruction algorithms for diffraction tomography: interpolation versus filtered backprojection. *IEEE Trans. Acoust. Sp. Sig. Proc.*, 31(5):1262–75, October 1983.

peng:87:dfr [50] H. Peng and H. Stark. Direct Fourier reconstruction in fan-beam tomography. *IEEE Trans. Med. Imag.*, 6(3):209–19, September 1987.

lawton:88:anp [51] W. Lawton. A new polar Fourier transform for computer-aided tomography and spotlight synthetic aperture radar. *IEEE Trans. Acoust. Sp. Sig. Proc.*, 36(6):931–3, June 1988.

matej:90:ahs [52] S. Matej and I. Bajla. A high-speed reconstruction from projections using direct Fourier method with optimized parameters-an experimental analysis. *IEEE Trans. Med. Imag.*, 9(4):421–9, December 1990.

cheung:91:mfr [53] W. K. Cheung and R. M. Lewitt. Modified Fourier reconstruction method using shifted transform samples. *Phys. Med. Biol.*, 36(2):269–77, February 1991.

jackson:91:soa [54] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski. Selection of a convolution function for Fourier inversion using gridding. *IEEE Trans. Med. Imag.*, 10(3):473–8, September 1991.

magnusson:92:aar [55] M. Magnusson, P-E. Danielsson, and P. Edholm. Artefacts and remedies in direct Fourier tomographic reconstruction. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 2, pages 1138–40, 1992.

tabei:92:bbu [56] M. Tabei and M. Ueda. Backprojection by upsampled Fourier series expansion and interpolated FFT. *IEEE Trans. Im. Proc.*, 1(1):77–87, January 1992.

defrise:94:aps [57] M. Defrise, A. Geissbuhler, and D. W. Townsend. A performance study of 3D reconstruction algorithms for positron emission tomography. *Phys. Med. Biol.*, 39(3):305–20, March 1994.

bellon:95:adf [58] P. L. Bellon and S. Lanzavecchia. A direct Fourier method (DFM) for X-ray tomographic reconstructions and the accurate simulation of sinograms. *Int. J. Bio-Med. Comput*, 38:55–69, 1995.

schomberg:95:tgm [59] H. Schomberg and J. Timmer. The gridding method for image reconstruction by Fourier transformation. *IEEE Trans. Med. Imag.*, 14(3):596–607, September 1995.

dusaussoy:96:vav [60] N. J. Dusaussoy. VOIR: a volumetric image reconstruction algorithm based on Fourier techniques for inversion of the 3-D Radon transform. *IEEE Trans. Im. Proc.*, 5(1):121–31, January 1996.

lanzavecchia:96:eti [61] S. Lanzavecchia and P. L. Bellon. Electron tomography in conical tilt geometry. The accuracy of a direct Fourier method (DFM) and the suppression of non-tomographic noise. *Ultramicroscopy*, 63(3-4):247–61, July 1996.

bellon:97:fdf [62] P. L. Bellon and S. Lanzavecchia. Fast direct Fourier methods, based on one- and two-pass coordinate transformations, yield accurate reconstructions of x-ray CT clinical images. *Phys. Med. Biol.*, 42(3):443–64, March 1997.

choi:98:dfr [63] H. Choi and D. C. Munson. Direct-Fourier reconstruction in tomography and synthetic aperture radar. *Intl. J. Imaging Sys. and Tech.*, 9(1):1–13, 1998.

gottlieb:00:otd [64] D. Gottlieb, B. Gustafsson, and P. Forssen. On the direct Fourier method for computer tomography. *IEEE Trans. Med. Imag.*, 19(3):223–32, March 2000.

walden:00:aot [65] J. Waldén. Analysis of the direct Fourier method for computer tomography. *IEEE Trans. Med. Imag.*, 19(3):211–22, March 2000.

potts:00:nfr [66] D. Potts and G. Steidl. New Fourier reconstruction algorithms for computerized tomography. In *Proc. SPIE 4119, Wavelet Appl. in Signal and Image Proc. VIII*, pages 13–23, 2000.

potts:01:anl [67] D. Potts and G. Steidl. A new linogram algorithm for computerized tomography. *IMA J. Numer. Anal.*, 21(3):769–82, July 2001.

herman:92:ftd [68] G. T. Herman, D. Roberts, and L. Axel. Fully three-dimensional reconstruction from data collected on concentric cubes in Fourier space: implementation and a sample application to MRI. *Phys. Med. Biol.*, 37(3):673–89, March 1992.

meyer:92:fsc [69] C. H. Meyer, B. S. Hu, D. G. Nishimura, and A. Macovski. Fast spiral coronary artery imaging. *Mag. Res. Med.*, 28(2):202–13, December 1992.

rosenfeld:98:aoa [70] D. Rosenfeld. An optimal and efficient new gridding algorithm using singular value decomposition. *Mag. Res. Med.*, 40(1):14–23, July 1998.

rasche:99:rod [71] V. Rasche, R. Proksa, R. Sinkus, P. Bornert, and H. Eggers. Resampling of data between arbitrary grids using convolution interpolation. *IEEE Trans. Med. Imag.*, 18(5):385–92, May 1999.

sedarat:00:oto [72] H. Sedarat and D. G. Nishimura. On the optimality of the gridding reconstruction algorithm. *IEEE Trans. Med. Imag.*, 19(4):306–17, April 2000.

wajer:01:sff [73] F. T. A. W. Wajer, R. Lethmate, J. A. C. van Osch, D. Graveron-Demilly, M. Fuderer, and D. van Ormondt. Simple formula for the accuracy of gridding. In *Proc. Intl. Soc. Mag. Res. Med.*, page 776, 2001.

dunne:90:fro [74] S. Dunne, S. Napel, and B. Rutt. Fast reprojection of volume data. In *Proc. of First Conf on Visualiztion in Biomedical Computing*, pages 11–8, 1990.

malzbender:93:fvr [75] T. Malzbender. Fourier volume rendering. *ACM Trans. on Graphics*, 12(3):233–50, July 1993.

totsuka:93:fdv [76] T. Totsuka and M. Levoy. Frequency-domain volume rendering. In *Siggraph*, pages 271–8, 1993.

mueller:99:otu [77] K. Mueller and R. Yagel. On the use of graphics hardware to accelerate algebraic reconstruction methods. In *Proc. SPIE 3659, Phys. of Medical Imaging*, pages 615–25, 1999.

mueller:00:r3d [78] K. Mueller and R. Yagel. Rapid 3-D cone-beam reconstruction with the simultaneous algebraic reconstruction technique (SART) using 2-D texture mapping hardware. *IEEE Trans. Med. Imag.*, 19(12):1227–37, December 2000.

matej:04:iti [79] S. Matej, J. A. Fessler, and I. G. Kazantsev. Iterative tomographic image reconstruction using Fourier-based forward and back- projectors. *IEEE Trans. Med. Imag.*, 23(4):401–12, April 2004.

averbuch::fss [80] A. Averbuch, D. L. Donoho, R. R. Coifman, M. Israeli, and Y. Shkolnisky. Fast slant stack: A notion of Radon transform for data in cartesian grid which is rapidly computable, algebrically exact, geometrically faithful and invertible. *siam-sc*, 2004. to appear? date?

zhang:04:fbf [81] Y. Zhang and J. A. Fessler. Fourier-based forward and back-projectors in iterative fan-beam tomographic image reconstruction. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 364–7, 2004.

zangoconnor:06:fbf [82] Y. Zhang-O'Connor and J. A. Fessler. Fourier-based forward and back-projectors in iterative fan-beam tomographic image reconstruction. *IEEE Trans. Med. Imag.*, 25(5):582–9, May 2006.

older:93:mfo [83] J. K. Older and P. C. Johns. Matrix formulation of computed tomogram reconstruction. *Phys. Med. Biol.*, 38(8):1051–64, August 1993.

hanson:80:oto [84] K. M. Hanson. On the optimality of the filtered backprojection algorithm. *J. Comp. Assisted Tomo.*, 4(3):361–63, June 1980.

barrett:03 [85] H. H. Barrett and K. J. Myers. *Foundations of image science*. Wiley, New York, 2003.

fessler:03:aat [86] J. A. Fessler. Analytical approach to regularization design for isotropic spatial resolution. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 2022–6, 2003.

shi:05:qrd [87] H. Shi and J. A. Fessler. Quadratic regularization design for fan beam transmission tomography. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 2023–33, 2005.

zhang:05:fvi [88] Y. Zhang, J. A. Fessler, and J. Hsieh. Fast variance image predictions for quadratically regularized statistical image reconstruction in fan-beam tomography. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 4, pages 1929–33, 2005.

riddell:06:rfc [89] C. Riddell and Y. Trousset. Rectification for cone-beam projection and backprojection. *IEEE Trans. Med. Imag.*, 25(7):950–62, July 2006.

lacroute:95:fvr [90] P. Lacroute. *Fast volume rendering using a shear-warp factorization of the viewing transformation*. PhD thesis,

Stanford, Stanford, 1995.

brandt:99:fco [91] A. Brandt and J. Dym. Fast calculation of multiple line integrals. *SIAM J. Sci. Comp.*, 30(4):1417–29, 1999.

fu:08:arc [92] L. Fu and J. Qi. A residual correction method for iterative reconstruction with inaccurate system model. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 1311–4, 2008.

sen:95:etf [93] A. Sen, H-H. Hsiung, M. Patel, B. A. Schueler, J. E. Holte, and X. Hu. Exact technique for weighting function calculation in 3D cone-beam reconstruction. In *Proc. SPIE 2434, Med. Im. 1995: Phys. of Med. Im.*, pages 616–26, 1995.

scheins:06:aco [94] J. J. Scheins, F. Boschen, and H. Herzog. Analytical calculation of volumes-of-intersection for iterative, fully 3-D PET reconstruction. *IEEE Trans. Med. Imag.*, 25(10):1363–9, October 2006.

lacroute:94:fvr [95] P. Lacroute and M. Levoy. Fast volume rendering using a shear-warp factorization of the viewing transformation. In *Siggraph*, pages 451–8, 1994.

weese:99:fvb [96] J. Weese, R. Goecke, G. P. Penney, P. Desmedt, T. M. Buzug, and H. Schumann. Fast voxel-based 2D/3D registration algorithm using a volume rendering method based on the shear-warp factorization. In *Proc. SPIE 3661, Medical Imaging 1999: Image. Proc.*, pages 802–10, 1999.

riddell:05:rfc [97] C. Riddell and Y. Trousset. Rectification for cone-beam projection and backprojection. In *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, pages 175–9, 2005.

wilson:02:teo [98] D. W. Wilson and H. H. Barrett. The effects of incorrect modeling on noise and resolution properties of ML-EM images. *IEEE Trans. Nuc. Sci.*, 49(3):768–73, June 2002.

irani:93:maf [99] M. Irani and S. Peleg. Motion analysis for image enhancement: resolution, occlusion, and transparency. *J. Visual Comm. Im. Rep.*, 4(4):324–35, December 1993.

bouwens:01:rrf [100] L. R. Bouwens, H. Gifford, R. V. . Walle, M. A. King, I. Lemahieu, and R. A. Dierckx. Resolution recovery for list-mode reconstruction in SPECT. *Phys. Med. Biol.*, 46(8):2239–54, August 2001.

gonzaleztrotter:00:ii1 [101] D. E. Gonzalez Trotter, J. E. Bowsher, and R. J. Jaszczak. Improved I-131 SPECT resolution through modeling individual medium-energy collimator holes. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 22/12–15, 2000.

mueller:99:fio [102] K. Mueller, R. Yagel, and J. J. Wheller. Fast implementations of algebraic methods for three-dimensional reconstruction from cone-beam data. *IEEE Trans. Med. Imag.*, 18(6):538–48, June 1999.

bonetto:00:tir [103] P. Bonetto. *Tomographic imaging: reconstruction and qualitative evaluation*. PhD thesis, Dipartimento di Informatica e Scienze dell'Informazione, Universita degli Studi di Genova, December 2000.

metzler:01:ado [104] S. D. Metzler, J. E. Bowsher, M. F. Smith, and R. J. Jaszczak. Analytic determination of pinhole collimator sensitivity with penetration. *IEEE Trans. Med. Imag.*, 20(8):730–41, August 2001.

kouris:81:spo [105] K. Kouris, E. S. Garnett, and G. T. Herman. Sampling properties of stationary and half-rotation rings in positron emission tomography. *J. Comp. Assisted Tomo.*, 5(5):744–54, October 1981.

huesman:80:dar [106] R. H. Huesman and J. L. Cahoon. Data acquisition, reconstruction, and display for the Donner 280-crystal positron tomograph. *IEEE Trans. Nuc. Sci.*, 27(1):474–8, February 1980.

thielemans:99:ova [107] K. Thielemans, M. W. Jacobson, and D. Belluzzo. On various approximations for the projectors in iterative reconstruction algorithms for 3D PET. In *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, pages 232–5, 1999.

zeng:92:fdi [108] G. L. Zeng and G. T. Gullberg. Frequency domain implementation of the three-dimensional geometric point response correction in SPECT imaging. *IEEE Trans. Nuc. Sci.*, 39(5-1):1444–53, October 1992.

zeng:94:ara [109] G. L. Zeng, Y-L. Hsieh, and G. T. Gullberg. A rotating and warping projector/backprojector for fan-beam and cone-beam iterative algorithm. *IEEE Trans. Nuc. Sci.*, 41(6):2807–11, December 1994.

dibella:96:aco [110] E. V. R. Di Bella, A. B. Barclay, R. L. Eisner, and R. W. Schafer. A comparison of rotation-based methods for iterative reconstruction algorithms. *IEEE Trans. Nuc. Sci.*, 43(6):3370–6, December 1996.

unser:95:cbi [111] M. Unser, P. Thevenaz, and L. Yaroslavsky. Convolution-based interpolation for fast, high quality rotation of images. *IEEE Trans. Im. Proc.*, 4(10):1371–81, October 1995.

zeng:00:upb [112] G. L. Zeng and G. T. Gullberg. Unmatched projector/backprojector pairs in an iterative reconstruction algorithm. *IEEE Trans. Med. Imag.*, 19(5):548–55, May 2000.

passeri:93:pmg [113] A. Passeri, A. R. Formiconi, and U. Meldolesi. Physical modelling (geometrical system response, Compton scattering and attenuation) in brain SPECT using the conjugate gradients reconstruction method. *Phys. Med. Biol.*, 38(12):1727–44, December 1993.