Discretization of CT Problems

Joost Batenburg, University of Leiden

Per Christian Hansen and Jakob Sauer Jørgensen

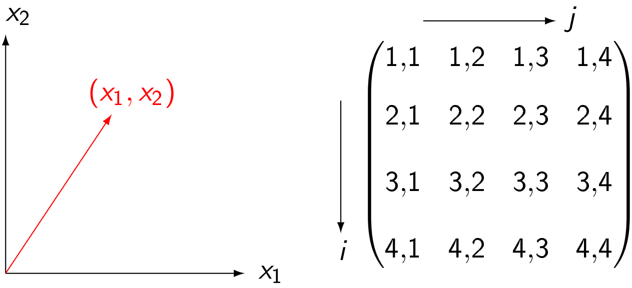
DTU Compute
Department of Applied Mathematics and Computer Science
Technical University of Denmark

Introduction

To represent the scanned object and the projection data in the computer, both must be discretized.

- **Object:** the physical reality is typically continuous, the way of discretization can be chosen freely.
- **Projection data:** the measurements are already discrete (finite set of detector elements, finite set of angles).
- **Forward transform:** the Radon transform must be discretized, defining how line integrals are computed or approximated on discrete images.

Image Coordinate Systems

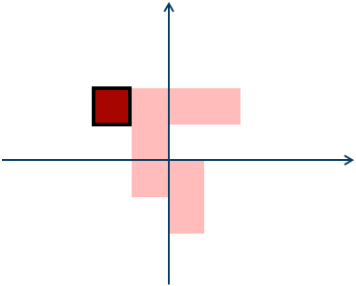


Left: Euclidean systems. Right: array “coordinates” or indices.

A Pixel as a Constant Function

A *pixel* is a rectangular domain

$$\pi = \left\{(x_1, x_2) \in \mathbb{R}^2 : x_1^{\text{left}} \leq x_1 < x_1^{\text{right}} \text{ and } x_2^{\text{bottom}} \leq x_2 < x_2^{\text{top}}\right\} .$$



The *pixel indicator function* is defined by

$$\chi_\pi(x_1, x_2) = \begin{cases} 1 & \text{if } (x_1, x_2) \in \pi \ , \\ 0 & \text{otherwise.} \end{cases}$$

Discrete Images

A *discrete image* $I_\boxplus$ is a pair $(\Pi, \boldsymbol{f})$ where $\Pi = \{\pi_1, \ldots, \pi_n\}$ is a regular pixel grid of n pixels, and the vector

$$\boldsymbol{f} = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} \in \mathbb{R}^n$$

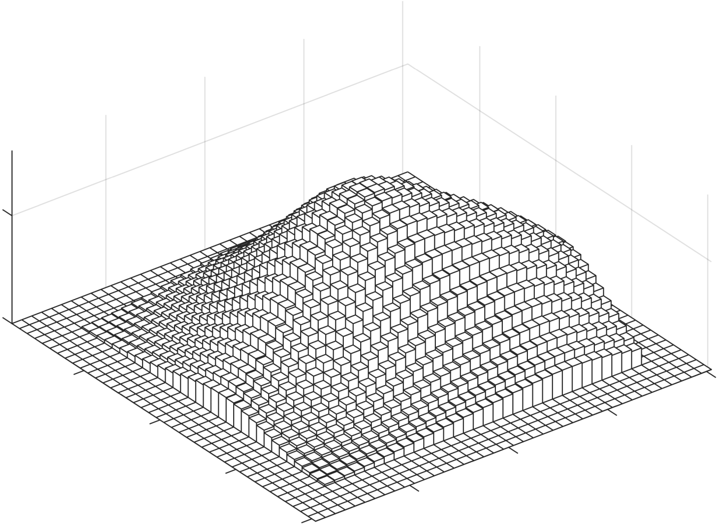
holds the pixel values of the image.

The discrete image $I_\boxplus$ induces an associated *pixelated image function:*

$$f_\boxplus(x_1, x_2) = \sum_{j=1}^{n} f_j \, \chi_{\pi_j}(x_1, x_2) \ .$$

This is a piecewise constant function which is constant in each pixel.

Example of a Pixelated Image Function



Intersection of Line and Pixel

Recall our definition of the line

$$L_{\theta,s} = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 \cos\theta + x_2 \sin\theta = s\} \ .$$

For any pixel π and line $L_{\theta,s}$, the intersection $L_{\theta,s} \cap \pi$ is a (possibly empty) line segment, which we call the **intersection segment** of $L_{\theta,s}$ and π.

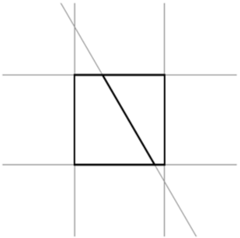
The **intersection length** $L_{\theta,s}^{(\pi)}$ (for line $L_{\theta,s}$ and pixel π) is the length of this segment. If $L_{\theta,s} \cap \pi = \emptyset$ (the line does not intersect the pixel) then we define $L_{\theta,s}^{(\pi)} = 0$.

The Radon transform $\mathcal{R}[\chi_\pi](\theta, s)$ of the indicator function for the pixel π, evaluated at (θ, s), is the line integral within π along $L_{\theta,s}$.

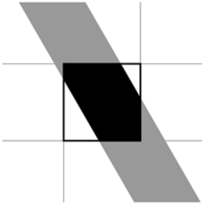
Since the indicator function is a constant function 1 within this pixel, the line integral is equal to the length of the line segment in the pixel:

$$\mathcal{R}[\chi_\pi](\theta, s) = L_{\theta,s}^{(\pi)} \ .$$

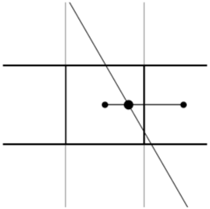
Projection Models Covered Here



Line model



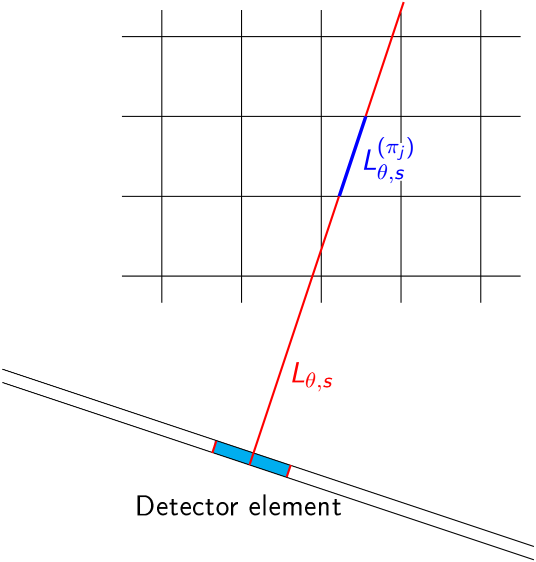
Strip model



Interpolation model

The interpolation model is also commonly known as the Joseph model.

The Line Model – Geometry



The Line Model – Details

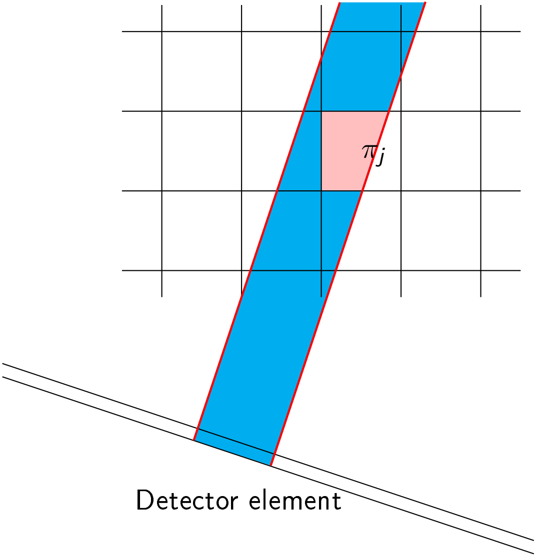
Given the pixelized image function $f_{\boxplus}$, its Radon transform can (due to its linearity) be expressed as

$$\mathcal{R}\{f_{\boxplus}\}(\theta, s) = \sum_{j=1}^{n} f_j \, \mathcal{R}[\chi_{\pi_j}](\theta, s) = \sum_{j=1}^{n} L_{\theta,s}^{(\pi_j)} \, f_j \ .$$

By computing the sum over the pixel intensities f_j in all pixels of $I_{\boxplus}$, weighted by their intersection lengths $L_{\theta,s}^{(\pi_j)}$, we obtain the value of the Radon transform of $f_{\boxplus}$, sampled at (θ, s) in the sinogram.

This process is often referred to as "Siddon's method" (although Siddon's contribution was a clever way to arrange the computations for a 3D grid).

The Strip Model – Geometry



The Strip Model – Details

Each detector element measures the integral of the sinogram $g(\theta,s)$ for all lines $L_{\theta_k,s}$ that intersect with it.

For parallel-beam CT, the lines that intersect the ℓth element form a *strip* defined by the set of parallel lines $\{L_{\theta_k,s} : s_\ell^{\text{left}} \leq s \leq s_\ell^{\text{right}}\}$. Given $f_\boxplus$ the data recorded in the ℓth pixel at angle θ_k is therefore given by the integral

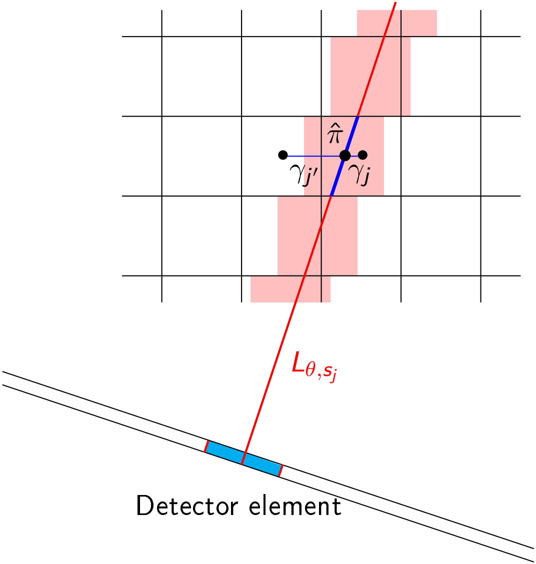
$$\int_{s=s_\ell^{\text{left}}}^{s_\ell^{\text{right}}} g(\theta_k, s)\,\mathrm{d}s = \int_{s=\ln_\ell^{\text{left}}}^{s_\ell^{\text{right}}} \mathcal{R}[f_\boxplus](\theta_k, s)\,\mathrm{d}s$$

$$= \sum_{j=1}^{n} f_j \int_{s=s_\ell^{\text{left}}}^{s_\ell^{\text{right}}} \mathcal{R}[\chi_{\pi_j}](\theta_k, s)\,ds = \sum_{j=1}^{n} \int_{s=s_\ell^{\text{left}}}^{s_\ell^{\text{right}}} L_{\theta_k,s}^{(\pi_j)}\,\mathrm{d}s\, f_j\ .$$

Hence, the contribution from pixel π_j at angle θ_k equals f_j times the *area of intersection* $\int_{s=s_\ell^{\text{left}}}^{s_\ell^{\text{right}}} L_{\theta_k,s}^{(\pi_j)} f_j\,\mathrm{d}s$ between pixel π_j and the strip, which is 0 if the strip does not overlap with the pixel.

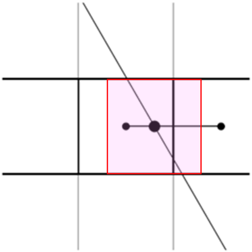
The total contribution to the detector element is the sum over all pixels.

The Interpolation Model (or Joseph Model) – Geometry



The Interpolation (or Joseph) Model – The Basic Idea

Key idea: put an artificial pixel $\hat{\pi}$ over the line L_{θ,s_j}, in such a way that we can use the line model within this artificial pixel.



The intensity value associated with the artificial pixel is found by linear interpolation between the pixel values f_j and $f_{j'}$ in two neighbour pixels π_j and $\pi_{j'}$, either in the same row or column – depending on θ.

See the details next slide.
(The Joseph model was originally presented without these details.)

The Interpolation (or Joseph) Model – The Groovy Details

Assume $\theta \in [-45°, 45°]$. The intersection length associated with $\hat{\pi}$ is $L^{\hat{\pi}}_{\theta,s_j} = \gamma/\cos\theta$, where γ is the width of the pixels.

Let γ_j and $\gamma_{j'}$ denote the lengths between the pixel centers and the center of the artificial pixel (obviously, $\gamma_j + \gamma_{j'} = \gamma$).

Using linear interpolation the pixel value of the artificial pixel is then

$$f_{\text{interp}} = \frac{\gamma_{j'}}{\gamma}\, f_j + \frac{\gamma_j}{\gamma}\, f_{j'} \ .$$

The contributions from the tow neighbour-pixels to the line integral are then obtained by multiplication with the intersection lengths:

$$\frac{\gamma_{j'}}{|\cos\theta|}\, f_j \qquad \text{and} \qquad \frac{\gamma_j}{|\cos\theta|}\, f_{j'} = \frac{\gamma - \gamma_{j'}}{|\cos\theta|}\, f_{j'} \ .$$

To compute γ_j and $\gamma_{j'}$ we need the horizontal coordinate of the artificial pixel's center, which is equal to the coordinate for the adjacent row shifted by a constant amount ς. By Pythagoras we have $\left(L^{\hat{\pi}}_{\theta,s_j}\right)^2 = \gamma^2 + \varsigma^2$, hence

$$\varsigma = L^{\hat{\pi}}_{\theta,s_j}\, \sin\theta = \gamma\, \tan\theta \ .$$

The System Matrix

All projection models take the form

$$\text{measured data for } \theta_k \text{ and } s_\ell \text{ is given by} \sum_{j=1}^{n} a_{k\ell}^{(j)} f_j \ .$$

Different discretization schemes leading to different expressions for $a_{k\ell}^{(j)}$.

The measurements $\boldsymbol{b} \in \mathbb{R}^m$ (stacking the elements in the sinogram) and the vector $\boldsymbol{f}$ of pixel values are related by a system of linear equations:

$$\boldsymbol{b} = \boldsymbol{A}\, \boldsymbol{f} \qquad \boldsymbol{A} \in \mathbb{R}^{m \times n} \ .$$

The elements of the *system matrix* $\boldsymbol{A}$ are given by

$$a_{ij} = a_{k\ell}^{(j)} \qquad \text{with} \qquad i = (k-1) N_s + \ell \ ,$$

where j, k and ℓ are associated with pixel π_j, projection angle θ_k and detector coordinate s_ℓ, respectively.

The number of rows in $\boldsymbol{A}$ equals the number $m = N_\theta \, N_s$ of lines in all views; the number n of columns equals the number n of image pixels.

Storage Issues

For a 100×100 image, using 100 views and 100 detector pixels, we have a matrix of 10^8 elements. This is challenging.

But there are few nonzero elements in the ith row of $\boldsymbol{A}$. For an $N \times N$ image, each line intersects with at most $2N$ pixels, meaning that there are at most $2N$ nonzero elements in each row.

We say that the system matrix is sparse, meaning that most of its elements are zero. This is conveniently used to reduce the memory requirement.

Even storing $\boldsymbol{A}$ in a sparse format can be problematic. If we collect 1000 views of 1000×1000 detector pixels then for a 3D grid with $N \times N \times N$ voxels and $N = 1000$ we have $m = 1000^3 = 10^9$ and there is at most $3N = 3000$ nonzeros per row, so the number of non-zeros in the system matrix is of the order $m\,N = 10^{12}$.

The only alternative is to avoid storing the system matrix and instead utilize the projection models for computing the matrix multiplications "on the fly."

The Columns and Rows of the System Matrix

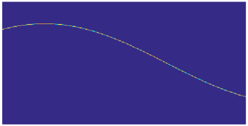
Recall $\boldsymbol{A}$ maps the discretized absorption coefficients in the pixels (the vector $\boldsymbol{f}$) to the data in the detector elements (the elements of the vector $\boldsymbol{b}$):

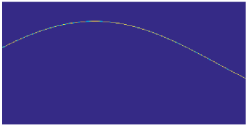
$$\boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \boldsymbol{A}\,\boldsymbol{f} = \underbrace{f_1\,\boldsymbol{c}_1 + f_2\,\boldsymbol{c}_2 + \cdots + f_n\,\boldsymbol{c}_n}_{\text{linear combination of columns}} . \tag{1}$$

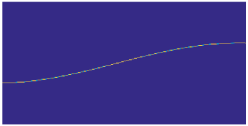
If the image consists of zeros except for a single pixel π_j with pixel value 1, then the corresponding vector $\boldsymbol{f}$ is all zeros except for a single element $f_j = 1$ in position j. The corresponding right-hand side is

$$\boldsymbol{b} = 0\,\boldsymbol{c}_1 + \cdots 0\,\boldsymbol{c}_{j-1} + 1\,\boldsymbol{c}_j + 0\,\boldsymbol{c}_{j+1} \cdots + 0\,\boldsymbol{c}_n = \boldsymbol{c}_j \ , \quad j = 1, 2, \ldots, n \ .$$

Hence $\boldsymbol{c}_j$, when reshaped, is the sinogram for a single pixel.







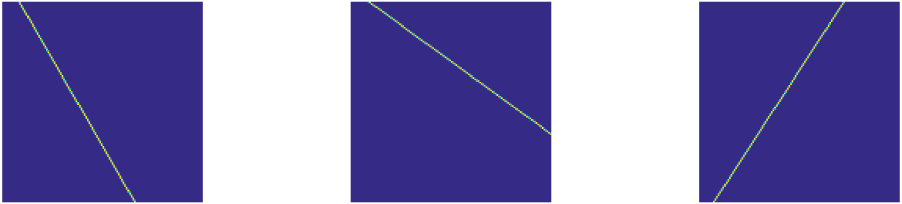
The Rows of the System Matrix

Now consider the ith row of $\boldsymbol{A}$ which maps the pixel values in $\boldsymbol{f}$ to the ith detector element:

$$b_i = \boldsymbol{r}_i^T \boldsymbol{f} = \sum_{j=1}^{n} a_{ij} f_j \ , \qquad i = 1, 2, \ldots, m \ . \tag{2}$$

For the line model, this inner product approximates the line integral in the Radon transform, and the nonzeros of $\boldsymbol{r}_i$ correspond to those image pixels that are intersected by the corresponding ray.

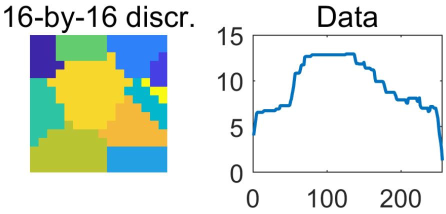
Hence, if we reshape $\boldsymbol{r}_i$ and plot it as a 2D image then we get a picture of the ray's path through the object.



Fooled by Discretization – One-Projection Reconstruction

For some combinations of a single projection angle θ, detector size, and number of detector elements, the system matrix $\boldsymbol{A}$ is square and nonsingular. Hence, it appears that we can compute the reconstruction $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ from a single projection.

Ex: $N = 16, \theta = 7^\circ$, N^2 detector elements, detector size = image size.



Above is $f_{\boxplus}$ and the detector data in the single projection. The resulting system matrix $\boldsymbol{A}$ has full rank and we can thus reconstruct the image from a single projection. This may seem sensational, but $f_{\boxplus}$ is a very poor representation of an actual grainy object.