Image Reconstruction

A short tour from analytical to data-driven methods

N. Ducros [1,2]

[1]CREATIS, INSA-Lyon

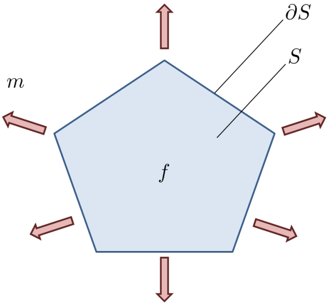
[2]Institut universitaire de France

Practical sessions: T Maitre

January 9, 2026

Natural vs Reconstructed Image

Inverse Problem



- Measurements m are taken on the boundary ∂S
- The (unknown) image f is defined on S
- How to estimate f? I.e., how to invert $m = \mathcal{A}(f)$ given $\mathcal{A}$?

Medical Imaging

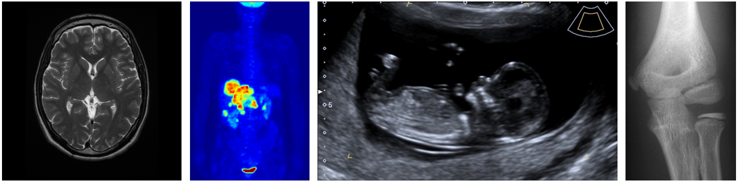


Figure: ( ) positron emission tomography ( ) photoacoustic imaging ( ) magnetic resonance ( ) ultrasonography ( ) radiography ( ) computed radiography

Main reconstruction approaches

I. Analytical methods

- Model the imaging process and invert it algebraically

$$\texttt{find } \mathcal{H} \texttt{ such that } f = \mathcal{H}(m) \tag{1}$$

II. Optimisation-based methods

- Build a cost function from prior knowledge about the solution and the measurements. Then, minimize it (algebraically or numerically)

$$\texttt{find and minimize } \mathcal{C} \texttt{ such that } \mathcal{C}(f; m) \texttt{ is small} \tag{2}$$

Main reconstruction approaches

III. Data-driven methods

- "Learn" to reconstruct

$$\texttt{learn } \mathcal{H}_\omega \texttt{ such that } f = \mathcal{H}_\omega(m) \tag{3}$$

Outline

Organisation

- 6 CM-TD
  https://github.com/nducros/image-reconstruction
- 2 TP salle Labex (TP 5GE)
- Évaluation : CR de TP + devoir surveillé (= QCM + Exo). Annales : https://moodle.insa-lyon.fr/course/section.php?id=13460
- Emploi du temps

Pioneers

Figure: Paul Langevin (left), Robert William Boyle (right)

Paul Langevin

- French physicist

Robert W Boyle

Pioneers

Figure: Paul Langevin (left), Robert William Boyle (right)

Paul Langevin

- French physicist
- First patents on ultrasonic submarine detection in 1916.

Robert W Boyle

Pioneers

Figure: Paul Langevin (left), Robert William Boyle (right)

Paul Langevin

- French physicist
- First patents on ultrasonic submarine detection in 1916.

Robert W Boyle

- Canadian physicist

Pioneers

Figure: Paul Langevin (left), Robert William Boyle (right)

Paul Langevin

- French physicist
- First patents on ultrasonic submarine detection in 1916.

Robert W Boyle

- Canadian physicist
- First working prototype of active sonar in 1917.

Recent echographic images

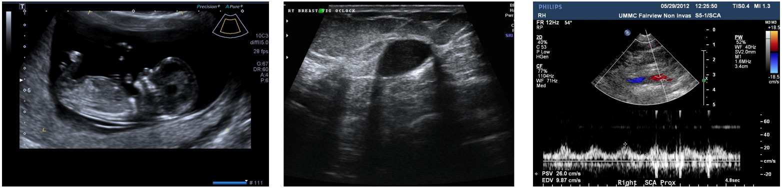
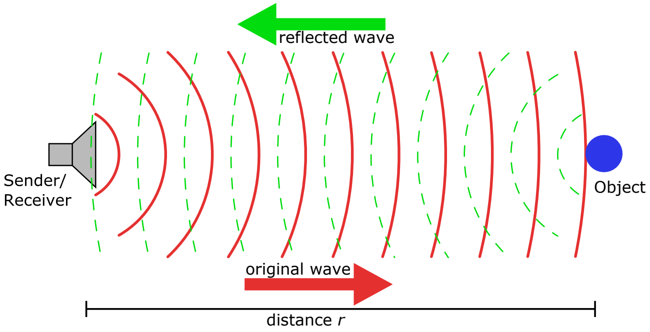


Figure: Nuchal translucency ultrasound, breast ultrasound, arterial Doppler. `https://radiopaedia.org/`

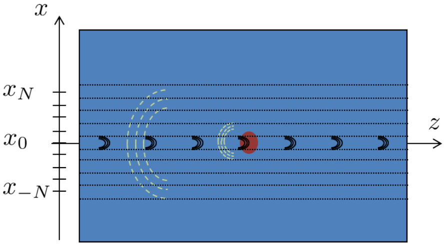
Image formation overview



Basics

- Ultrasonography consists in recording echos
- Contrast is due to changes in the tissue acoustic properties

Inverse problem



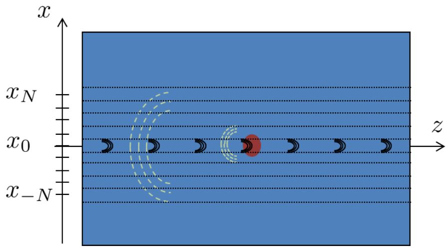
Measured data

- $\{m_n(t)\}$: set of signals measured after excitation along $x = x_0$
- $n \in \{-N, \dots, N\}$: index of the transducer element centered at x_n

To be recovered

- $f(x_0, z)$: reflectors along $x = x_0$

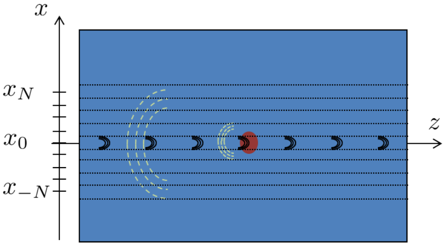
Inverse problem



Space is time!

- $\hat{f}_n(x_0, z) = G(z)\, m_n(t = \tau_n(z))$, with $\tau_n(z) = \tau^{\text{exci}}(z) + \tau_n^{\text{echo}}(z)$
- τ^{exci} is the source-reflector propagation time and τ_n^{echo} the propagation time from the reflector to the n-th element
- G is a gain that compensates for attenuation (and angular energy dispersion)

Inverse problem

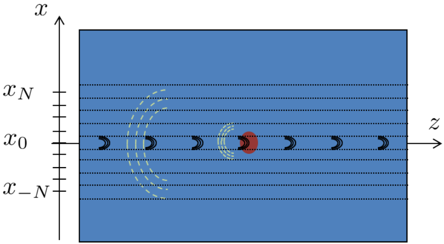


Exercise

- Calculate the delays $\tau^{\text{exci}}(z)$ and $\tau_n^{\text{echo}}(z)$

Solution

Inverse problem



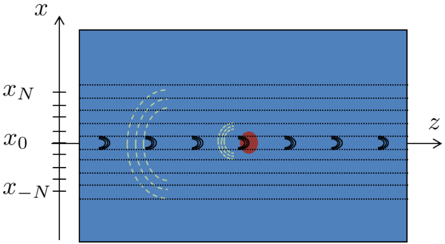
Exercise

- Calculate the delays $\tau^{\text{exci}}(z)$ and $\tau_n^{\text{echo}}(z)$

Solution

- $\tau^{\text{exci}}(z) = \frac{z}{c}$, assuming focusing (see slides 92–93)

Inverse problem



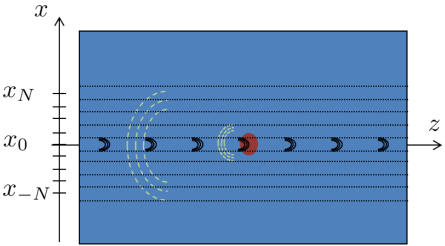
Exercise

- Calculate the delays $\tau^{\text{exci}}(z)$ and $\tau_n^{\text{echo}}(z)$

Solution

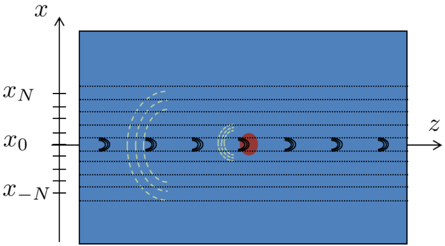
- $\tau^{\text{exci}}(z) = \frac{z}{c}$, assuming focusing (see slides 92–93)
- $\tau_n^{\text{echo}}(z) = \frac{1}{c}\sqrt{(x_0 - x_n)^2 + z^2}$

Inverse problem



Some thoughts

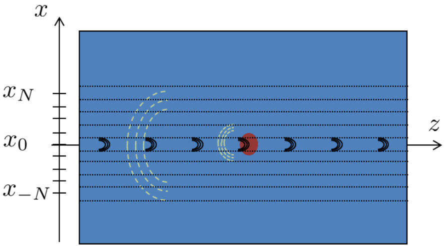
Inverse problem



Some thoughts

- Are the reflectivity maps $\hat{f}_n(x_0, z)$ estimated from different signals m_n the same?

Inverse problem



Some thoughts

- Are the reflectivity maps $\hat{f}_n(x_0, z)$ estimated from different signals m_n the same?
- Which one to choose?

Dynamic focusing

Philosophy

- Dynamic focusing answers the previous two questions

Dynamic focusing

Philosophy

- Dynamic focusing answers the previous two questions
- How ?

  1. Aligning (temporally) all the signals from the same echo being recorded at different positions x_n
  2. Averaging the aligned signals

Delay-and-Sum Algorithm

Governing equation

- Along a given beam direction, we define

$$m(t) = \frac{1}{2N+1} \sum_{n=-N}^{N} m_n(t - \Delta\tau_n)$$

where $\Delta\tau_n$ are appropriate delays that align the signals

Delay-and-Sum Algorithm

Exercise 1

- Calculate the delays $\Delta\tau_n$ to align the signal m_n with the signal v_0 recorded at x_0

Solution 1

Delay-and-Sum Algorithm

Exercise 1

- Calculate the delays $\Delta\tau_n$ to align the signal m_n with the signal v_0 recorded at x_0

Solution 1

- $\Delta\tau_n = \frac{z}{c}\left(1 - \sqrt{1 + \left(\frac{x_n - x_0}{z}\right)^2}\right)$

Delay-and-Sum Algorithm

Exercise 2

- How to estimate the reflectivity map $f(x_0, z)$ from the delay-and-sum signal m?
- Derive an expression of the reflectivity map $f(x_0, z)$ as a function of the raw signals m_n

Solution 2

Delay-and-Sum Algorithm

Exercise 2

- How to estimate the reflectivity map $f(x_0, z)$ from the delay-and-sum signal m?
- Derive an expression of the reflectivity map $f(x_0, z)$ as a function of the raw signals m_n

Solution 2

- Remember that: i) time is space and ii) all signals were aligned with m_0: $\hat{f}(x_0, z) = G(z)\, v\left(\frac{2z}{c}\right)$

Delay-and-Sum Algorithm

Exercise 2

- How to estimate the reflectivity map $f(x_0, z)$ from the delay-and-sum signal m?
- Derive an expression of the reflectivity map $f(x_0, z)$ as a function of the raw signals m_n

Solution 2

- Remember that: i) time is space and ii) all signals were aligned with m_0: $\hat{f}(x_0, z) = G(z)\, v\left(\frac{2z}{c}\right)$
- $\hat{f}(x_0, z) = G(z)\, \frac{1}{2N+1} \sum_{n=-N}^{N} m_n \left(\frac{1}{c} \left(z + \sqrt{z^2 + (x_n - x_0)^2} \right) \right)$

Wilhelm Röntgen

Figure: Mr. Röntgen (left), Mrs Röntgen (au centre), a colleague (right)

ID card

- German physicist.
- Pioneering work on X-rays in 1895.
- First recipient of the Nobel Price in Physisc in 1901.

Nowadays

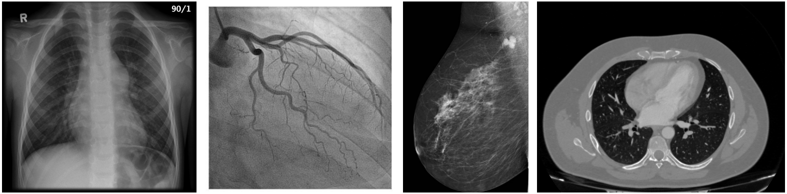
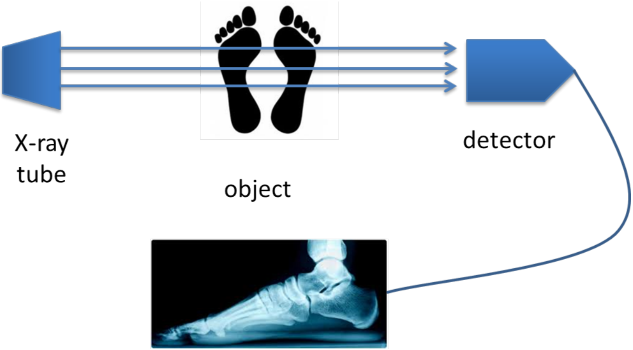


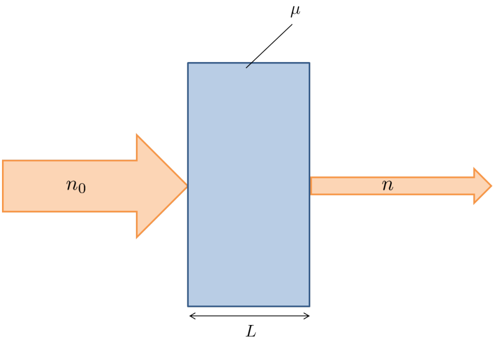
Figure: Chest radiography, left coronary artery (interventional cardiology), mammogram, thorax CT scan. https://radiopaedia.org/.

Image formation overview

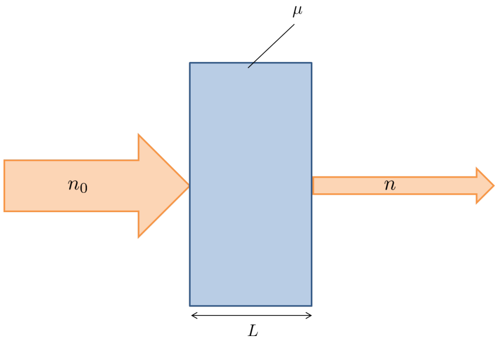


Contrast is related to the *X-ray attenuation* of the tissues

Beer-Lambert law



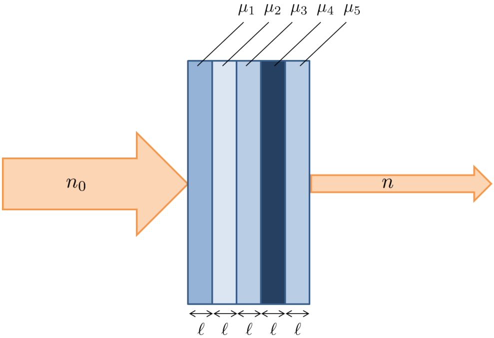
Beer-Lambert law



- μ is the linear attenuation coefficient

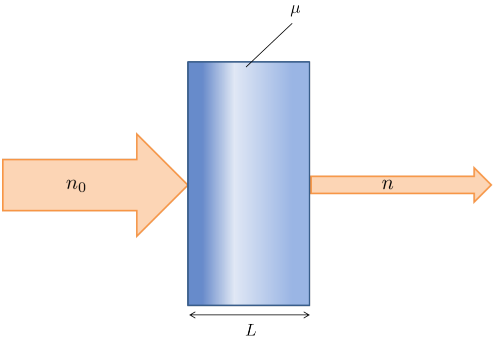
$$n = n_0 \exp\left(-\mu L\right)$$

Beer-Lambert law



$$n = n_0 \prod_i \exp\left(-\mu_i \ell\right) = n_0 \exp\left(-\sum_i \mu_i \ell\right)$$

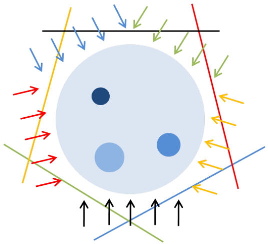
Beer-Lambert law



Remember

$$n = n_0 \exp\left(-\int_0^L \mu(\ell)\,\mathrm{d}\ell\right)$$

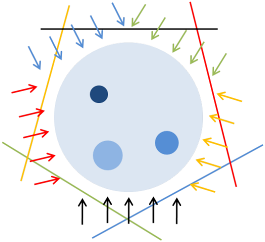
Forward problem



CT principle

- Radiographs are acquired under multiple view angles
- Post processing all the radiographs allows the object to be recovered, which is called *image reconstruction*

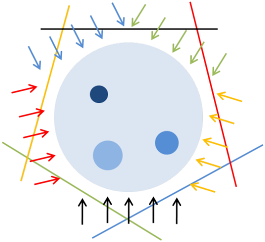
Forward problem



Measured data

- Line integrals (projections) of the attenuation coefficient (see slide 39): $\ln(n_0/n) = m = \int_{\mathcal{L}} f \, \mathrm{d}\ell$

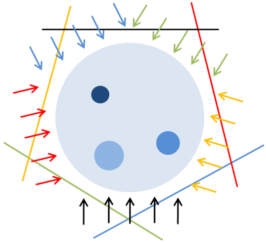
Forward problem



Measured data

- Line integrals (projections) of the attenuation coefficient (see slide 39): $\ln(n_0/n) = m = \int_{\mathcal{L}} f \, \mathrm{d}\ell$
- The line path $\mathcal{L}$ depends on the acquisition geometry

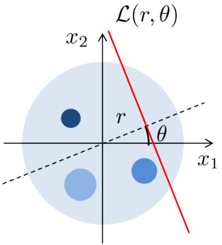
Forward problem



Measured data

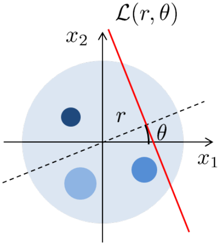
- Line integrals (projections) of the attenuation coefficient (see slide 39): $\ln(n_0/n) = m = \int_{\mathcal{L}} f \, \mathrm{d}\ell$
- The line path $\mathcal{L}$ depends on the acquisition geometry
- Parallel geometry here (in practice fan beam, cone beam, etc)

Forward problem



Measured data

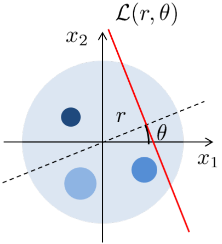
Forward problem



Measured data

- Let $\mathcal{L}(r,\theta)$ be the line path parametrized by (r,θ) as depicted above

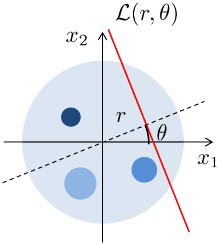
Forward problem



Measured data

- Let $\mathcal{L}(r,\theta)$ be the line path parametrized by (r,θ) as depicted above
- Let $m(r,\theta)$ be the line integral along $\mathcal{L}(r,\theta)$

Forward problem



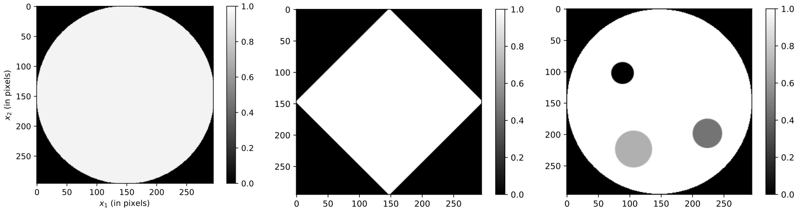
Measured data

- Let $\mathcal{L}(r,\theta)$ be the line path parametrized by (r,θ) as depicted above
- Let $m(r,\theta)$ be the line integral along $\mathcal{L}(r,\theta)$
- The set of line integrals for $(r,\theta) \in \mathbb{R} \times [0,\pi[$ is referred to as *sinogram*

Forward problem

Exercise 1

- Compute the Radon transform of the images below



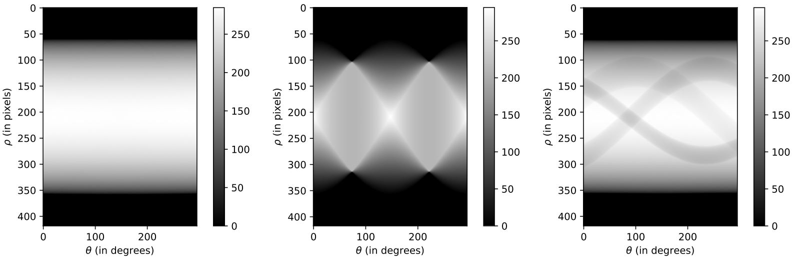
Forward problem

Solution 1

Forward problem

Solution 1

- We obtain



Forward problem

Exercise 2

- Give the equation of $\mathcal{L}(r,\theta)$ in the polar coordinate system $(\boldsymbol{u}_r, \boldsymbol{u}_\theta)$
- Derive the equation of $\mathcal{L}(r,\theta)$ in the Cartesian coordinate system $(\boldsymbol{u}_1, \boldsymbol{u}_2)$
- Derive the equation of $m(r,\theta)$ as a function of f

Solution 2

Forward problem

Exercise 2

- Give the equation of $\mathcal{L}(r,\theta)$ in the polar coordinate system $(\boldsymbol{u}_r, \boldsymbol{u}_\theta)$
- Derive the equation of $\mathcal{L}(r,\theta)$ in the Cartesian coordinate system $(\boldsymbol{u}_1, \boldsymbol{u}_2)$
- Derive the equation of $m(r,\theta)$ as a function of f

Solution 2

- $\mathcal{L}(r,\theta)$: $\boldsymbol{\ell} = r\boldsymbol{u}_r + \ell \boldsymbol{u}_\theta$, $\ell \in \mathbb{R}$

Forward problem

Exercise 2

- Give the equation of $\mathcal{L}(r,\theta)$ in the polar coordinate system $(\boldsymbol{u}_r, \boldsymbol{u}_\theta)$
- Derive the equation of $\mathcal{L}(r,\theta)$ in the Cartesian coordinate system $(\boldsymbol{u}_1, \boldsymbol{u}_2)$
- Derive the equation of $m(r,\theta)$ as a function of f

Solution 2

- $\mathcal{L}(r,\theta)$: $\boldsymbol{\ell} = r\boldsymbol{u}_r + \ell\boldsymbol{u}_\theta$, $\ell \in \mathbb{R}$
- $\mathcal{L}(r,\theta)$: $\boldsymbol{\ell} = (r\cos\theta - \ell\sin\theta)\boldsymbol{u}_1 + (r\sin\theta + \ell\cos\theta)\boldsymbol{u}_2, \ell \in \mathbb{R}$

Forward problem

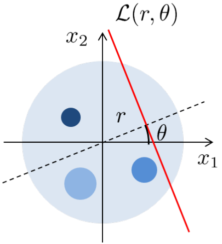
Exercise 2

- Give the equation of $\mathcal{L}(r,\theta)$ in the polar coordinate system $(\boldsymbol{u}_r, \boldsymbol{u}_\theta)$
- Derive the equation of $\mathcal{L}(r,\theta)$ in the Cartesian coordinate system $(\boldsymbol{u}_1, \boldsymbol{u}_2)$
- Derive the equation of $m(r,\theta)$ as a function of f

Solution 2

- $\mathcal{L}(r,\theta)$: $\boldsymbol{\ell} = r\boldsymbol{u}_r + \ell\boldsymbol{u}_\theta,\ \ell \in \mathbb{R}$
- $\mathcal{L}(r,\theta)$: $\boldsymbol{\ell} = (r\cos\theta - \ell\sin\theta)\boldsymbol{u}_1 + (r\sin\theta + \ell\cos\theta)\boldsymbol{u}_2, \ell \in \mathbb{R}$
- $m(r,\theta) = \displaystyle\int_{\mathbb{R}} f(r\cos\theta - \ell\sin\theta, r\sin\theta + \ell\cos\theta)\,\mathrm{d}\ell$

Inverse Problem



Measured data

- Sinogram $m(r,\theta)$

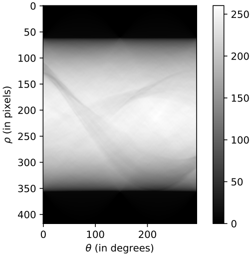
To be recovered

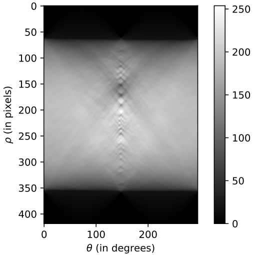
- Attenuation map $f(x_1, x_2)$

Inverse Problem

Exercise

- Compute the inverse Radon transform of the following images (guess the results first)





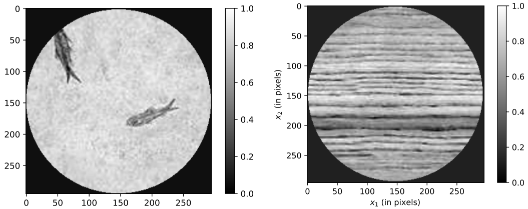
Inverse Problem

Solution

Inverse Problem

Solution

- We obtain



Projection-Slice Theorem

Theorem (a.k.a. Fourier slice theorem)

$$\hat{m}_\theta(\xi) = \hat{f}(\xi \cos\theta, \xi \sin\theta)$$

Where

Projection-Slice Theorem

Theorem (a.k.a. Fourier slice theorem)

$$\hat{m}_\theta(\xi) = \hat{f}(\xi \cos\theta, \xi \sin\theta)$$

Where

- $\hat{m}_\theta(\xi) = \int_{\mathbb{R}} m(r,\theta) \exp(-\mathrm{j}2\pi\xi r)\, \mathrm{d}r$ is a 1-D Fourier transform (with respect to r only)

Projection-Slice Theorem

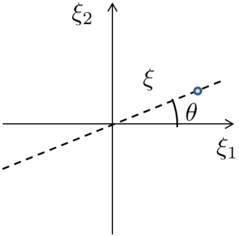
Theorem (a.k.a. Fourier slice theorem)

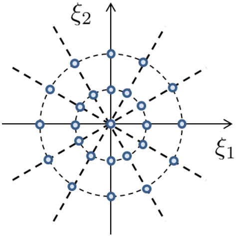
$$\hat{m}_\theta(\xi) = \hat{f}(\xi \cos\theta, \xi \sin\theta)$$

Where

- $\hat{m}_\theta(\xi) = \int_{\mathbb{R}} m(r,\theta) \exp(-\mathrm{j}2\pi\xi r)\, \mathrm{d}r$ is a 1-D Fourier transform (with respect to r only)
- $\hat{f}(\xi_1,\xi_2) = \iint_{\mathbb{R}^2} f(x_1,x_2) \exp\left(-\mathrm{j}2\pi(\xi_1 x_1 + \xi_2 x_2)\right) \mathrm{d}x_1\, \mathrm{d}x_2$ is a 2-D Fourier transform

Projection-Slice Theorem





- The Fourier domain is sampled on a polar grid
- Interpolation is required to recover f by inverse Fourier transformation

Projection-Slice Theorem

Sketch of the proof

- From the definition of $\hat{m}_\theta(\xi)$ find the appropriate change of variables
- Remember to compute the determinant of the Jacobian

Exercise

- Demonstrate the projection-slice theorem

Filtered Back-Projection

FBP formula

The image f is given by

$$f(x_1, x_2) = \int_0^\pi m_\theta^{\text{filt}}(x_1 \cos\theta + x_2 \sin\theta)\, \mathrm{d}\theta \tag{3}$$

where

$$\hat{m}_\theta^{\text{filt}}(\xi) = |\xi|\, \hat{m}_\theta(\xi) \tag{4}$$

Meaning

Filtered Back-Projection

FBP formula

The image f is given by

$$f(x_1, x_2) = \int_0^{\pi} m_\theta^{\text{filt}}(x_1 \cos\theta + x_2 \sin\theta)\, \mathrm{d}\theta \tag{3}$$

where

$$\hat{m}_\theta^{\text{filt}}(\xi) = |\xi|\, \hat{m}_\theta(\xi) \tag{4}$$

Meaning

- Equation (3) is a sum of 'back projection' of m_θ^{filt} over different view angles

Filtered Back-Projection

FBP formula

The image f is given by

$$f(x_1, x_2) = \int_0^\pi m_\theta^{\text{filt}}(x_1 \cos\theta + x_2 \sin\theta)\, \mathrm{d}\theta \tag{3}$$

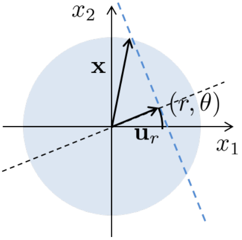
where

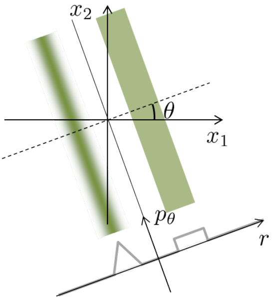
$$\hat{m}_\theta^{\text{filt}}(\xi) = |\xi|\, \hat{m}_\theta(\xi) \tag{4}$$

Meaning

- Equation (3) is a sum of 'back projection' of m_θ^{filt} over different view angles
- Equation (4) tells how to filter the projections m_θ

Filtered Back-Projection

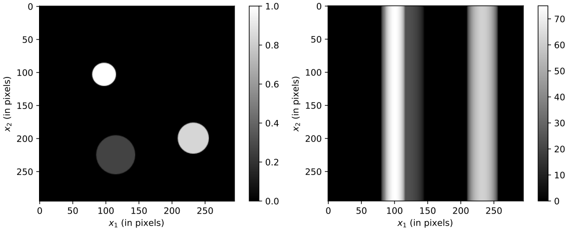




Back-projection

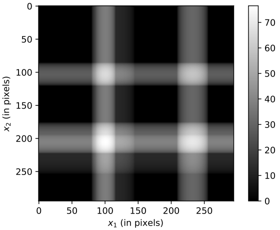
- $x_1 \cos\theta + x_2 \sin\theta = \boldsymbol{x} \cdot \boldsymbol{u}_r = r$ is the equation of a (straight) line
- The line is denoted by $\mathcal{L}(r, \theta)$ on slide 30

Filtered Back-Projection



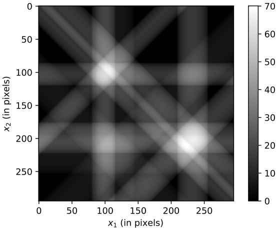
Back-projection

Filtered Back-Projection



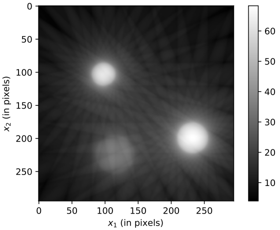
Back-projection

Filtered Back-Projection



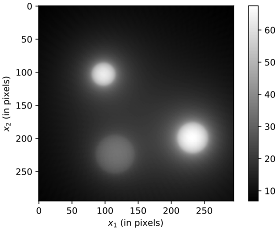
Back-projection

Filtered Back-Projection



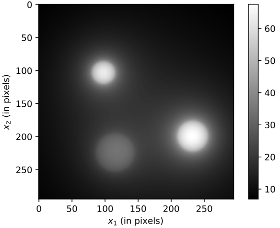
Back-projection

Filtered Back-Projection



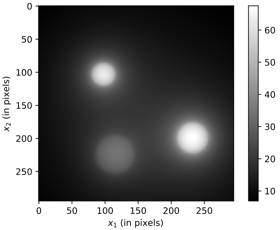
Back-projection

Filtered Back-Projection



Back-projection

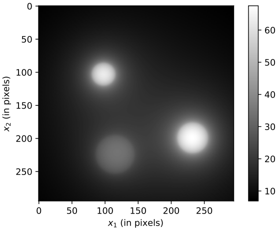
Filtered Back-Projection



Back-projection

- Summing back-projections almost recover the image

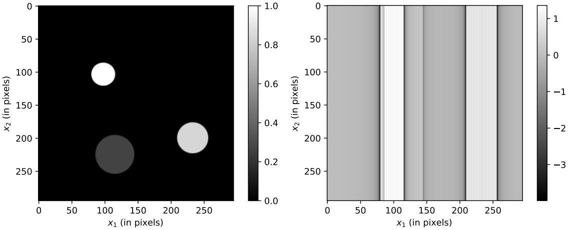
Filtered Back-Projection



Back-projection

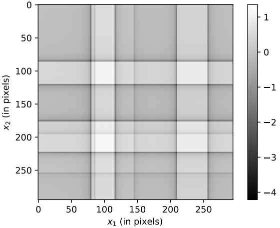
- Summing back-projections almost recover the image
- The unfiltered back-projection image is blurred

Filtered Back-Projection



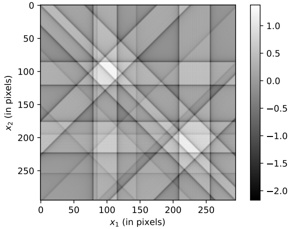
Filtering the projections (measurements)

Filtered Back-Projection



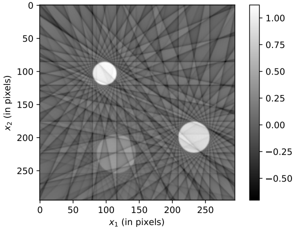
Filtering the projections (measurements)

Filtered Back-Projection



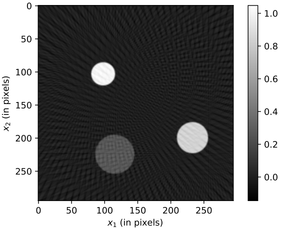
Filtering the projections (measurements)

Filtered Back-Projection



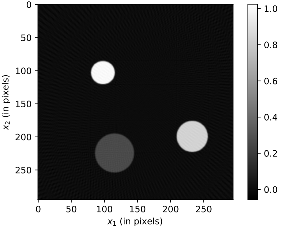
Filtering the projections (measurements)

Filtered Back-Projection



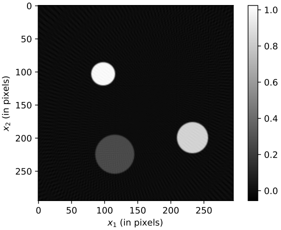
Filtering the projections (measurements)

Filtered Back-Projection



Filtering the projections (measurements)

Filtered Back-Projection



Filtering the projections (measurements)

- Back-projection now recovers the image perfectly!

Filtered Back-Projection

Sketch of the proof of the FBP formula

- Write f as a function of $\hat{f}$
- Use the projection-slice theorem

Exercise

- Derive the filtered back-projection formula

Filtered Back-Projection

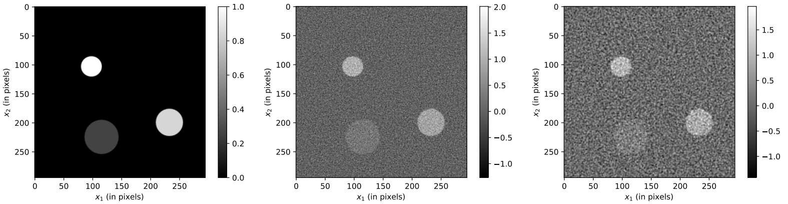


Figure: FBP reconstructions. (Left) ground truth, (middle) ramp filter, (right) ramp filter with Hann window.

Filtered back-projection

- Perfect recovery from noiseless data...
- ... but high-frequency noise is amplified by the ramp filter
- In practice, $\hat{m}_\theta^{\text{filt}}(\xi) = |\xi|\, \hat{m}_\theta(\xi)\hat{h}(\xi)$ where $\hat{h}(\xi)$ is a low-pass filter

Discretisation

Real world is discrete

Discretisation

Real world is discrete

- Finite number of detector pixels

Discretisation

Real world is discrete

- Finite number of detector pixels
- Finite number of view angles

Discretisation

Real world is discrete

- Finite number of detector pixels
- Finite number of view angles
- Dimensions of the reconstructed image are finite

Discretisation

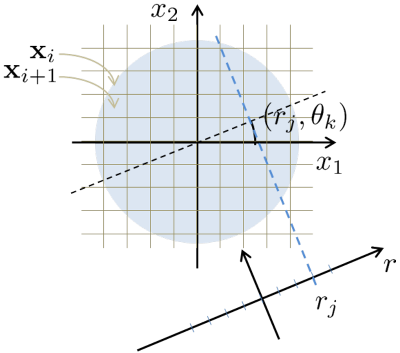
Real world is discrete

- Finite number of detector pixels
- Finite number of view angles
- Dimensions of the reconstructed image are finite

Two approaches

- Analytical is 'solve-and-discretize'
- Algebraic is 'discretize-and-solve'

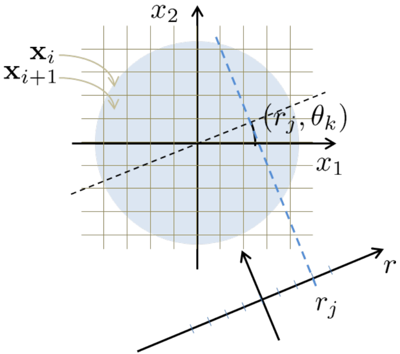
In the case of CT



Discretization of variables

- $r \in \{r_j\}, j \in \{1, \ldots, J\}$
- $\theta \in \{\theta_k\}, k \in \{1, \ldots, K\}$
- $\boldsymbol{x} = (x_1, x_2) \in \{\boldsymbol{x}_i\}, i \in \{1, \ldots, I = I_1 \times I_2\}$

In the case of CT

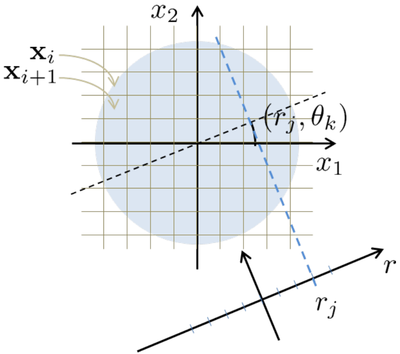


Discretization of functions

- Attenuation image: $f(\boldsymbol{x}) = \sum_i f_i b_i(\boldsymbol{x})$
- Projection at angle θ_k: $m_k(r) = \sum_j m_{k,j} b_j(r)$
- Where $b_i(\boldsymbol{x})$ and $b_j(r)$ are some basis functions

Examples of basis functions

In the case of CT



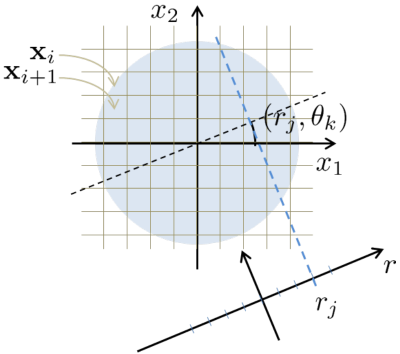
Discretization of functions

- Attenuation image: $f(\boldsymbol{x}) = \sum_i f_i b_i(\boldsymbol{x})$
- Projection at angle θ_k: $m_k(r) = \sum_j m_{k,j} b_j(r)$
- Where $b_i(\boldsymbol{x})$ and $b_j(r)$ are some basis functions

Examples of basis functions

- Sampling: $b_j(r) = \delta(r - r_j)$

In the case of CT



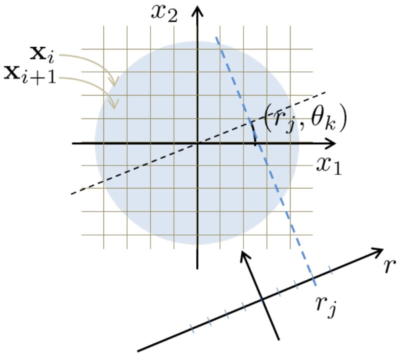
Discretization of functions

- Attenuation image: $f(\boldsymbol{x}) = \sum_i f_i b_i(\boldsymbol{x})$
- Projection at angle θ_k: $m_k(r) = \sum_j m_{k,j} b_j(r)$
- Where $b_i(\boldsymbol{x})$ and $b_j(r)$ are some basis functions

Examples of basis functions

- Sampling: $b_j(r) = \delta(r - r_j)$
- Indicator: $b_j(r) = \mathrm{rect}(r - r_j)$

In the case of CT



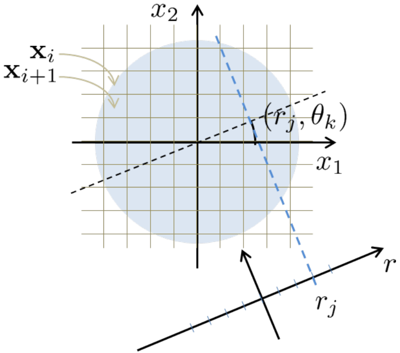
Discretization of functions

- Attenuation image: $f(\boldsymbol{x}) = \sum_i f_i b_i(\boldsymbol{x})$
- Projection at angle θ_k: $m_k(r) = \sum_j m_{k,j} b_j(r)$
- Where $b_i(\boldsymbol{x})$ and $b_j(r)$ are some basis functions

Examples of basis functions

- Sampling: $b_j(r) = \delta(r - r_j)$
- Indicator: $b_j(r) = \mathrm{rect}(r - r_j)$
- Gaussians, splines, etc

In the case of CT

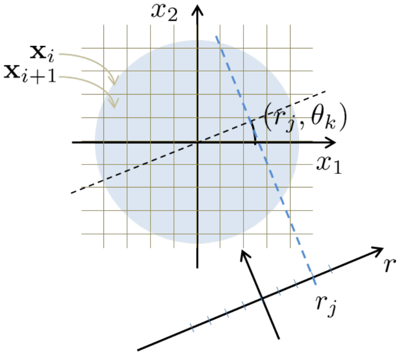


Linear forward problem

$$\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f}$$

where

In the case of CT



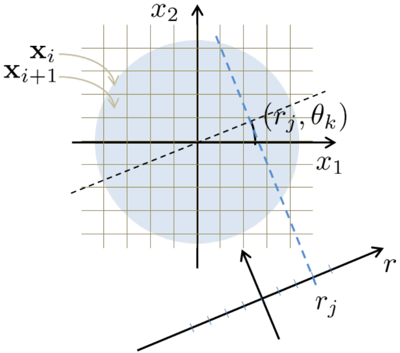
Linear forward problem

$$\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f}$$

where

- Attenuation vector $\boldsymbol{f} \in \mathbb{R}^{I \times 1}$ defined as $\boldsymbol{f} = [f_1, \ldots, f_I]^\top$

In the case of CT



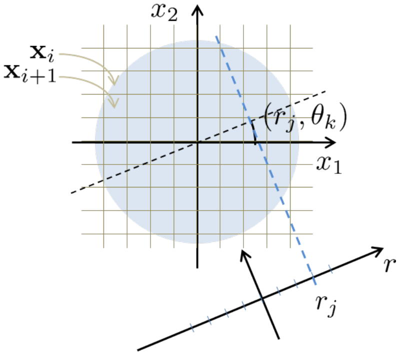
Linear forward problem

$$\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f}$$

where

- Attenuation vector $\boldsymbol{f} \in \mathbb{R}^{I \times 1}$ defined as $\boldsymbol{f} = [f_1, \ldots, f_I]^\top$
- Forward mapping $\boldsymbol{A} \in \mathbb{R}^{KJ \times I}$

In the case of CT



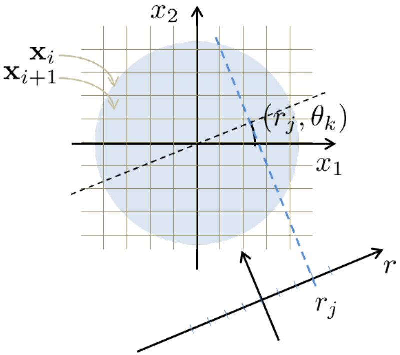
Linear forward problem

$$\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f}$$

where

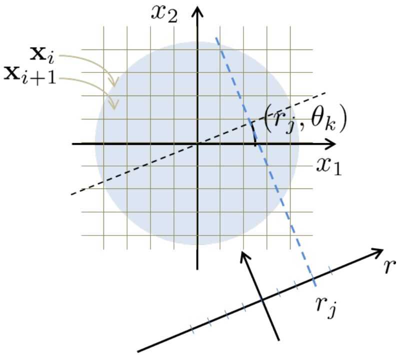
- Attenuation vector $\boldsymbol{f} \in \mathbb{R}^{I \times 1}$ defined as $\boldsymbol{f} = [f_1, \ldots, f_I]^\top$
- Forward mapping $\boldsymbol{A} \in \mathbb{R}^{KJ \times I}$
- Projection vector $\boldsymbol{m} \in \mathbb{R}^{KJ \times 1}$ defined as $\boldsymbol{m} = [\boldsymbol{m}_1^\top, \ldots, \boldsymbol{m}_K^\top]^\top$, with $\boldsymbol{m}_k = [m_{k,1}, \ldots, m_{k,J}]^\top$

In the case of CT



Sketch of the proof

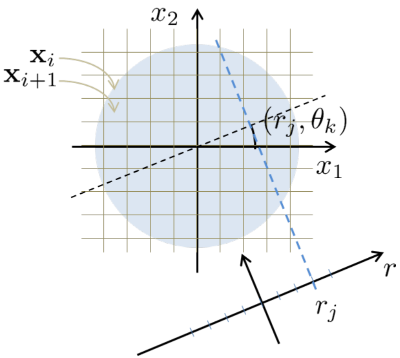
In the case of CT



Sketch of the proof

- Start from $m_k(r) = \int_{\mathbb{R}^2} f(\boldsymbol{x})\delta(r - \boldsymbol{x} \cdot \boldsymbol{u}_r)\,\mathrm{d}\boldsymbol{x}$

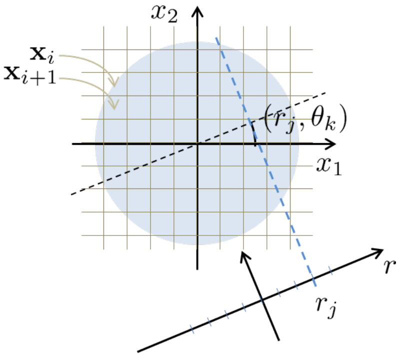
In the case of CT



Sketch of the proof

- Start from $m_k(r) = \int_{\mathbb{R}^2} f(\boldsymbol{x})\delta(r - \boldsymbol{x} \cdot \boldsymbol{u}_r)\,\mathrm{d}\boldsymbol{x}$
- Show that $m_{k,j} = \int_{\mathbb{R}^2} f(\boldsymbol{x}) b_{j,k}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$

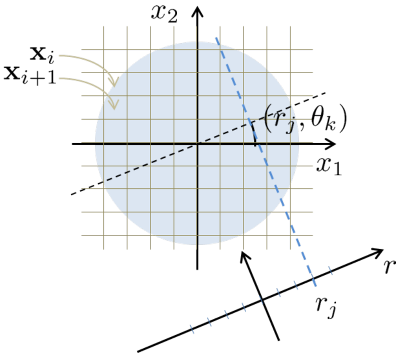
In the case of CT



Sketch of the proof

- Start from $m_k(r) = \int_{\mathbb{R}^2} f(\boldsymbol{x})\delta(r - \boldsymbol{x} \cdot \boldsymbol{u}_r)\,\mathrm{d}\boldsymbol{x}$
- Show that $m_{k,j} = \int_{\mathbb{R}^2} f(\boldsymbol{x}) b_{j,k}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$
- Hence $m_{k,j} = \sum_i r_{(k,j),i} f_i$

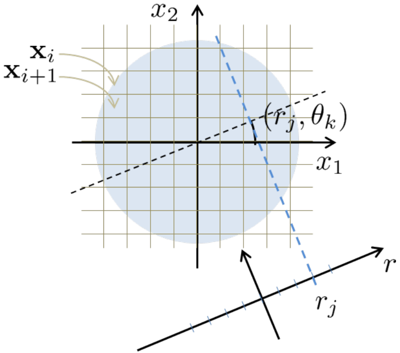
In the case of CT



Sketch of the proof

- Start from $m_k(r) = \int_{\mathbb{R}^2} f(\boldsymbol{x})\delta(r - \boldsymbol{x}\cdot\boldsymbol{u}_r)\,\mathrm{d}\boldsymbol{x}$
- Show that $m_{k,j} = \int_{\mathbb{R}^2} f(\boldsymbol{x}) b_{j,k}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$
- Hence $m_{k,j} = \sum_i r_{(k,j),i} f_i$
- $r_{(k,j),i}$ indicates the contribution of pixel i to measurement at angle k and position j

In the case of CT

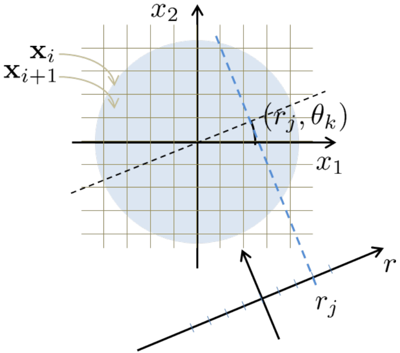


Exercise

- Following the sketch of the proof, derive an expression for $r_{(k,j),i}$

Solution

In the case of CT



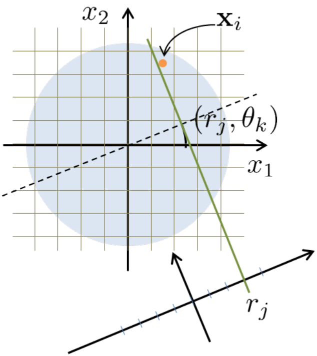
Exercise

- Following the sketch of the proof, derive an expression for $r_{(k,j),i}$

Solution

- $r_{(k,j),i} = \int_{\mathbb{R}^2} b_i(\boldsymbol{x}) b_{j,k}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}$, with $b_{j,k}(\boldsymbol{x}) = b_j(\boldsymbol{x} \cdot \boldsymbol{u}_r(\theta_k))$

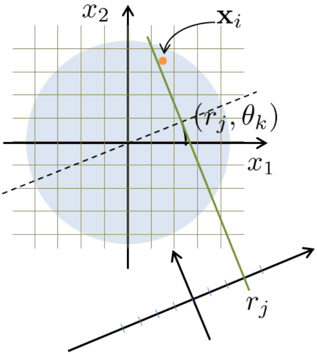
In the case of CT



Interpretation of $r_{h,i}$

- We define $h = (k, j)$

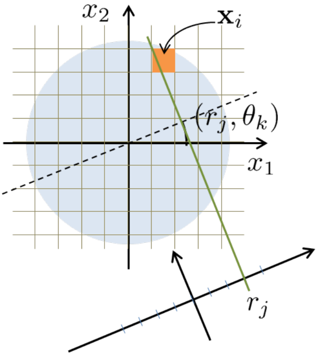
In the case of CT



Interpretation of $r_{h,i}$

- We define $h = (k, j)$
- For $b_i = b_j = \delta$, $r_{h,i}$ indicates if the center of pixel i is intersected by ray $h = (k, j)$

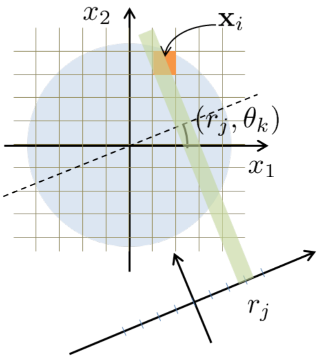
In the case of CT



Interpretation of $r_{h,i}$

- We define $h = (k, j)$
- For $b_i = b_j = \delta$, $r_{h,i}$ indicates if the center of pixel i is intersected by ray $h = (k, j)$
- For $b_i = \text{rect}$ and $b_j = \delta$, it is the length of ray h through pixel i

In the case of CT



Interpretation of $r_{h,i}$

- We define $h = (k, j)$
- For $b_i = b_j = \delta$, $r_{h,i}$ indicates if the center of pixel i is intersected by ray $h = (k, j)$
- For $b_i = \mathrm{rect}$ and $b_j = \delta$, it is the length of ray h through pixel i
- for $b_i = b_j = \mathrm{rect}$, it is the area of ray h over pixel i

In the case of CT

Exercise

- Construct the discrete forward operator corresponding to the Radon transform of a $I_1 \times I_2 = 32 \times 32$ image computed under $J = 40$ view angles over $[0, \pi)$ using a linear detector of $K = 45$ pixels.
  Hint: You can call the `radon` function from `skimage`.

In the case of CT

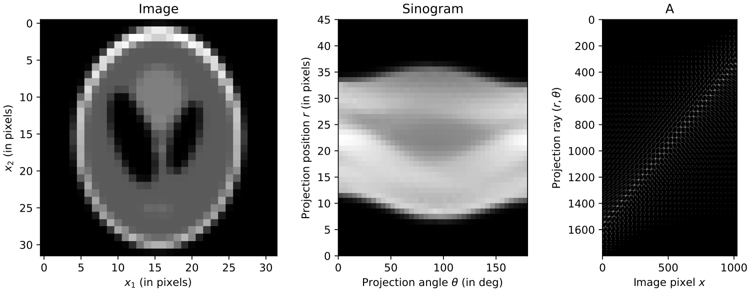


Figure: Discrete forward problem. (Left) attenuation *image*. (Middle) corresponding *sinogram*. (Right) discrete Radon transform $\boldsymbol{A}$. Here, $K \times J = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

Algebraic reconstruction

Reconstruction

- Recover f measuring m

However...

Algebraic reconstruction

Reconstruction

- Recover $\boldsymbol{f}$ measuring $\boldsymbol{m}$
- Basically, $\boldsymbol{f} = \boldsymbol{A}^{-1}\boldsymbol{m}$

However...

Algebraic reconstruction

Reconstruction

- Recover $\boldsymbol{f}$ measuring $\boldsymbol{m}$
- Basically, $\boldsymbol{f} = \boldsymbol{A}^{-1}\boldsymbol{m}$

However...

- $\boldsymbol{A}$ is huge. $\boldsymbol{A} \in \mathbb{R}^{H \times I}$, with $H = KJ$. Typically $H \times I = 256 \cdot 360 \times 256^2$, which requires 45 GiB for full storage.

Algebraic reconstruction

Reconstruction

- Recover $\boldsymbol{f}$ measuring $\boldsymbol{m}$
- Basically, $\boldsymbol{f} = \boldsymbol{A}^{-1}\boldsymbol{m}$

However...

- $\boldsymbol{A}$ is huge. $\boldsymbol{A} \in \mathbb{R}^{H \times I}$, with $H = KJ$. Typically $H \times I = 256 \cdot 360 \times 256^2$, which requires 45 GiB for full storage.
- Infinity of solutions (few angles available)

Algebraic reconstruction

Reconstruction

- Recover $\boldsymbol{f}$ measuring $\boldsymbol{m}$
- Basically, $\boldsymbol{f} = \boldsymbol{A}^{-1}\boldsymbol{m}$

However...

- $\boldsymbol{A}$ is huge. $\boldsymbol{A} \in \mathbb{R}^{H \times I}$, with $H = KJ$. Typically $H \times I = 256 \cdot 360 \times 256^2$, which requires 45 GiB for full storage.
- Infinity of solutions (few angles available)
- No solution (in the presence of noise, i.e., $\boldsymbol{m} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\eta}$)

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$

Standard iterative reconstruction algorithms

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number

Standard iterative reconstruction algorithms

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n

Standard iterative reconstruction algorithms

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n

Standard iterative reconstruction algorithms

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n
- Update can also be multiplicative

Standard iterative reconstruction algorithms

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n
- Update can also be multiplicative

Standard iterative reconstruction algorithms

- Algebraic reconstruction technique (ART)

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n
- Update can also be multiplicative

Standard iterative reconstruction algorithms

- Algebraic reconstruction technique (ART)
- Simultaneous Iterative Reconstructive Technique (SIRT)

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n
- Update can also be multiplicative

Standard iterative reconstruction algorithms

- Algebraic reconstruction technique (ART)
- Simultaneous Iterative Reconstructive Technique (SIRT)
- Simultaneous Algebraic Reconstruction Technique (SART)

Algebraic reconstruction

Iterative reconstruction

- $\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \boldsymbol{u}^{(n)}$
- n is the iteration number
- $\boldsymbol{f}^{(n)}$ is the attenuation image at iteration n
- $\boldsymbol{u}^{(n)}$ is the update at iteration n
- Update can also be multiplicative

Standard iterative reconstruction algorithms

- Algebraic reconstruction technique (ART)
- Simultaneous Iterative Reconstructive Technique (SIRT)
- Simultaneous Algebraic Reconstruction Technique (SART)
- ...

ART algorithm

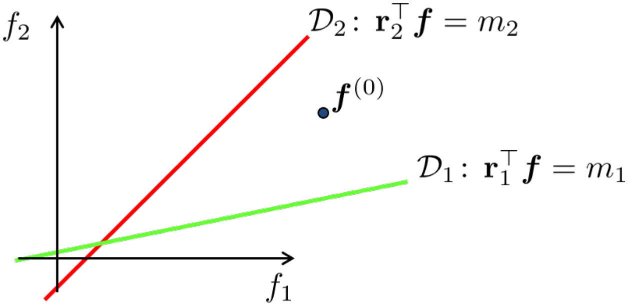


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

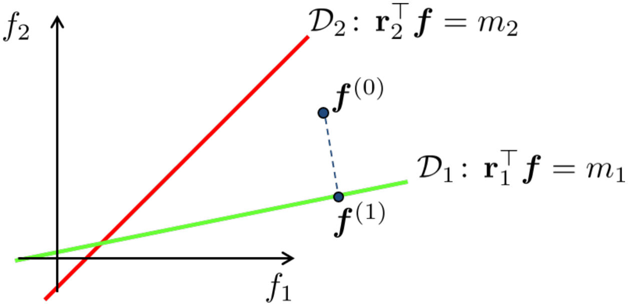


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

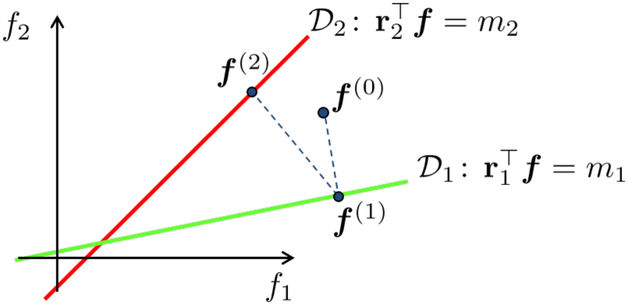


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

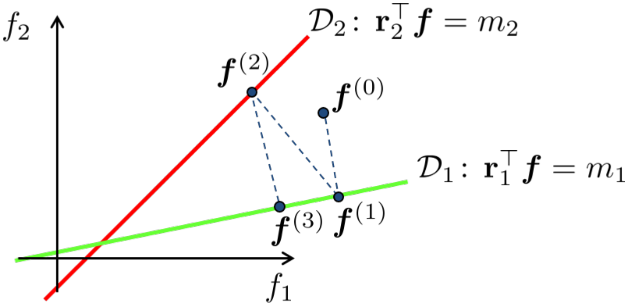


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

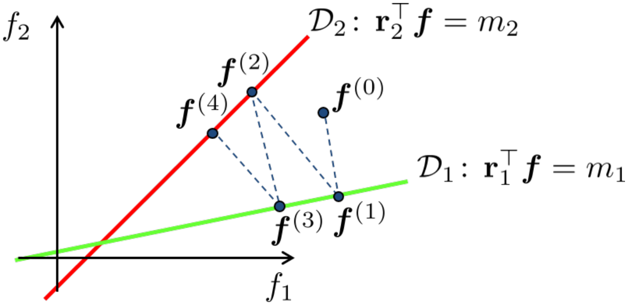


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

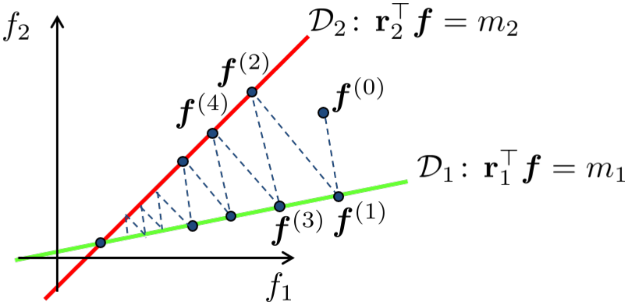


Figure: Illustration in 2D ($I = H = 2$), where $\boldsymbol{r}_h^\top$ is the h-th row of $\boldsymbol{A}$. Note that $\boldsymbol{r}_h$ is normal to $\mathcal{D}_h$.

ART algorithm

Update equation

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

Implementation

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

- The h-th equation is satisfied: $\boldsymbol{r}_h^\top \boldsymbol{f}^{(h)} = m_h$

Implementation

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

- The h-th equation is satisfied: $\boldsymbol{r}_h^\top \boldsymbol{f}^{(h)} = m_h$

Implementation

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

- The h-th equation is satisfied: $\boldsymbol{r}_h^\top \boldsymbol{f}^{(h)} = m_h$

Implementation

- One iteration of ART corresponds to a sequence of $h = H$ updates.

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

- The h-th equation is satisfied: $\boldsymbol{r}_h^\top \boldsymbol{f}^{(h)} = m_h$

Implementation

- One iteration of ART corresponds to a sequence of $h = H$ updates.
- A few iterations are typically required before convergence.

ART algorithm

Update equation

- We have

$$\boldsymbol{f}^{(h)} = \boldsymbol{f}^{(h-1)} - \frac{\boldsymbol{r}_h^\top \boldsymbol{f}^{(h-1)} - m_h}{\boldsymbol{r}_h^\top \boldsymbol{r}_h} \boldsymbol{r}_h$$

- The h-th equation is satisfied: $\boldsymbol{r}_h^\top \boldsymbol{f}^{(h)} = m_h$

Implementation

- One iteration of ART corresponds to a sequence of $h = H$ updates.
- A few iterations are typically required before convergence.
- There are variants that depend on the order the measurements are processed (e.g., random-ART) or the way image is updated (e.g., simultaneous ART)

ART algorithm

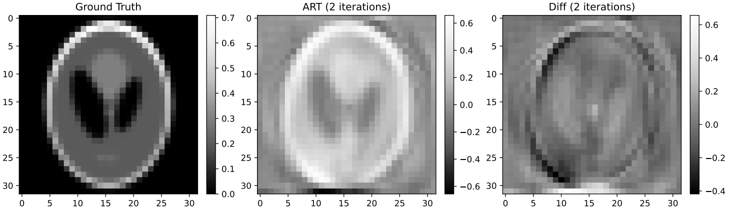


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

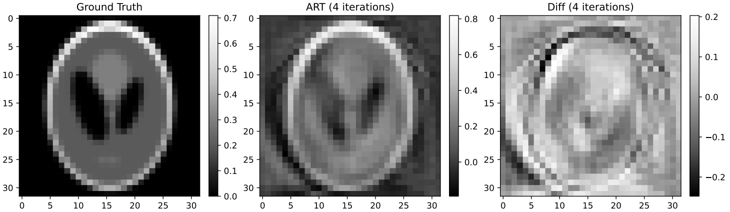


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

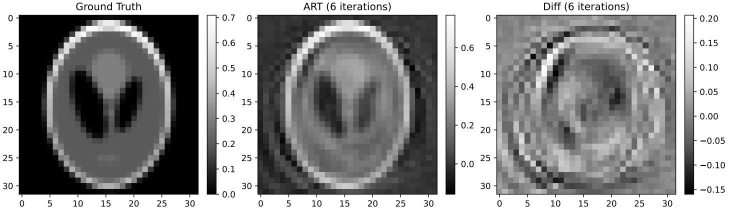


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

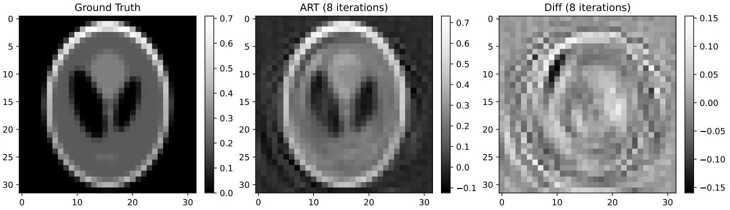


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

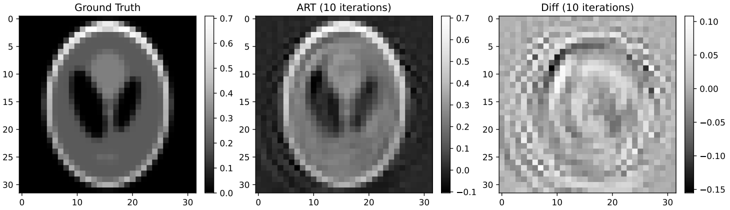


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

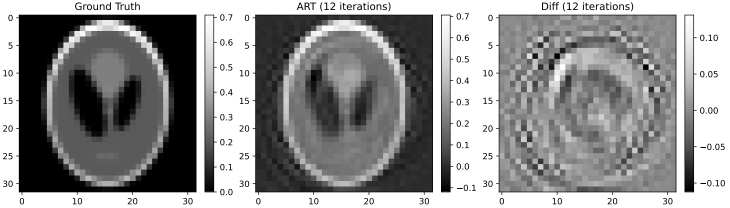


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

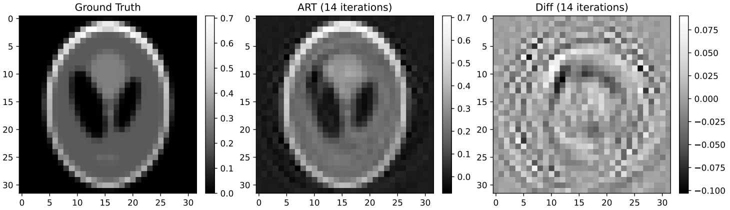


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

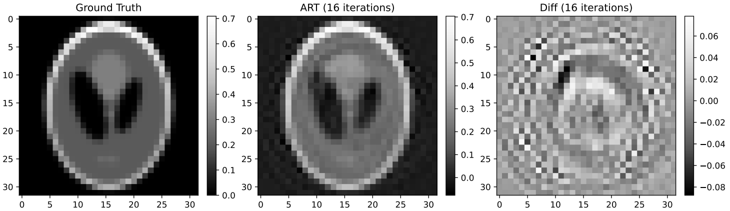


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

ART algorithm

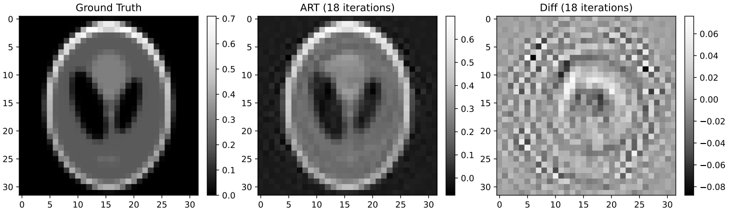


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. As in slide 55, $H = 40 \times 45 = 1800$ and $I = 32 \times 32 = 1024$.

Data fidelity optimisation

Reconstruction as optimization problem

Data fidelity optimisation

Reconstruction as optimization problem

- Look for small data misfits/residuals, i.e. $\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$

Data fidelity optimisation

Reconstruction as optimization problem

- Look for small data misfits/residuals, i.e. $\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$
- Minimize the cost/objective function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|$$

Data fidelity optimisation

Reconstruction as optimization problem

- Look for small data misfits/residuals, i.e. $\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{0}$
- Minimize the cost/objective function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|$$

- where $\|\cdot\|$ is some norm (e.g., L^2 or L^1)

Data fidelity optimisation

Analytical solution

Data fidelity optimisation

Analytical solution

- A minimizer of the least-squares cost function $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$, satisfies

$$\nabla \mathcal{C} = \mathbf{0}$$

Data fidelity optimisation

Analytical solution

- A minimizer of the least-squares cost function $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$, satisfies

$$\nabla \mathcal{C} = \mathbf{0}$$

- Therefore, it is a solution of the system

$$\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{f} = \boldsymbol{A}^\top \boldsymbol{m}$$

Data fidelity optimisation

Analytical solution

- A minimizer of the least-squares cost function $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$, satisfies

$$\nabla \mathcal{C} = \mathbf{0}$$

- Therefore, it is a solution of the system

$$\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{f} = \boldsymbol{A}^\top \boldsymbol{m}$$

- When $\boldsymbol{A}^\top \boldsymbol{A}$ is invertible, the least-squares solution is given by

$$\boldsymbol{f}_{\text{ls}} = (\boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{m}$$

It is also known as the pseudo-inverse solution.

Data fidelity optimisation

Exercise

- Derive an expression for the gradient of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$
- (Clue: First notice that $\|\boldsymbol{f}\|^2 = \boldsymbol{f}^\top\boldsymbol{f}$; second, derive its gradient; finally generalize to the remaining terms in $\mathcal{C}(\boldsymbol{f})$.)

Solution

Data fidelity optimisation

Exercise

- Derive an expression for the gradient of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$
- (Clue: First notice that $\|\boldsymbol{f}\|^2 = \boldsymbol{f}^\top \boldsymbol{f}$; second, derive its gradient; finally generalize to the remaining terms in $\mathcal{C}(\boldsymbol{f})$.)

Solution

- We first write

$$
\begin{aligned}
\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 &= (\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f})^\top(\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}) \\
&= \boldsymbol{m}^\top\boldsymbol{m} - \boldsymbol{m}^\top\boldsymbol{A}\boldsymbol{f} - (\boldsymbol{A}\boldsymbol{f})^\top\boldsymbol{m} + (\boldsymbol{A}\boldsymbol{f})^\top\boldsymbol{A}\boldsymbol{f} \\
&= \boldsymbol{m}^\top\boldsymbol{m} - 2\boldsymbol{m}^\top\boldsymbol{A}\boldsymbol{f} + (\boldsymbol{A}\boldsymbol{f})^\top\boldsymbol{A}\boldsymbol{f}
\end{aligned}
$$

Data fidelity optimisation

Exercise

- Derive an expression for the gradient of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2$
- (Clue: First notice that $\|\boldsymbol{f}\|^2 = \boldsymbol{f}^\top \boldsymbol{f}$; second, derive its gradient; finally generalize to the remaining terms in $\mathcal{C}(\boldsymbol{f})$.)

Solution

- We first write

$$\begin{aligned}
\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 &= (\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f})^\top (\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}) \\
&= \boldsymbol{m}^\top \boldsymbol{m} - \boldsymbol{m}^\top \boldsymbol{A}\boldsymbol{f} - (\boldsymbol{A}\boldsymbol{f})^\top \boldsymbol{m} + (\boldsymbol{A}\boldsymbol{f})^\top \boldsymbol{A}\boldsymbol{f} \\
&= \boldsymbol{m}^\top \boldsymbol{m} - 2\boldsymbol{m}^\top \boldsymbol{A}\boldsymbol{f} + (\boldsymbol{A}\boldsymbol{f})^\top \boldsymbol{A}\boldsymbol{f}
\end{aligned}$$

- $\nabla \mathcal{C}^{(n)} = 0 - 2(\boldsymbol{m}^\top \boldsymbol{A})^\top + 2\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{f}^{(n)} = -2\boldsymbol{A}^\top (\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}^{(n)})$

Data fidelity optimisation

Gradient descent

Data fidelity optimisation

Gradient descent

- Iterative algorithm

$$\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} - \gamma \nabla \mathcal{C}^{(n)}$$

Data fidelity optimisation

Gradient descent

- Iterative algorithm

$$\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} - \gamma \nabla \mathcal{C}^{(n)}$$

- $\nabla \mathcal{C}^{(n)}$ is the gradient of the cost function at $\boldsymbol{f}^{(n)}$

Data fidelity optimisation

Gradient descent

- Iterative algorithm

$$\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} - \gamma \nabla \mathcal{C}^{(n)}$$

- $\nabla \mathcal{C}^{(n)}$ is the gradient of the cost function at $\boldsymbol{f}^{(n)}$
- γ is the step length

Data fidelity optimisation

Gradient descent

- Iterative algorithm

$$\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} - \gamma \nabla \mathcal{C}^{(n)}$$

- $\nabla \mathcal{C}^{(n)}$ is the gradient of the cost function at $\boldsymbol{f}^{(n)}$
- γ is the step length
- γ is chosen small enough to guarantee convergence but large enough to converge in few iterations

Gradient descent (no noise)

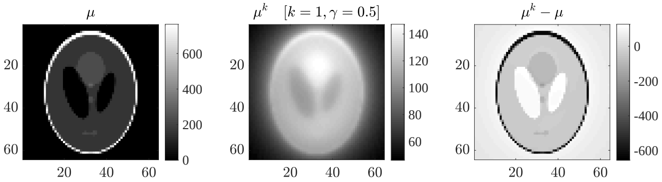


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

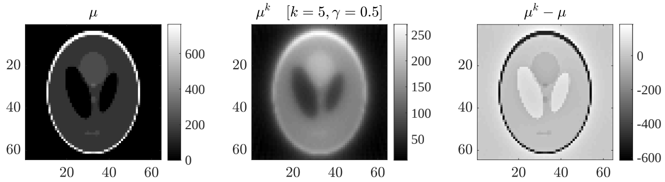


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

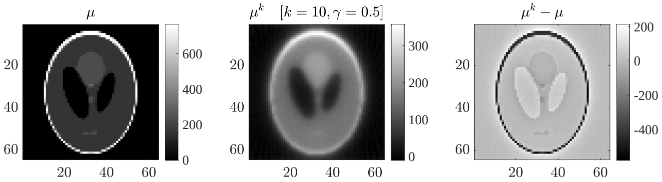


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

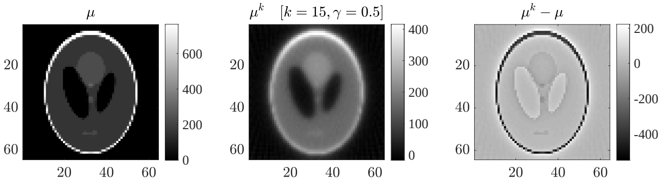


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

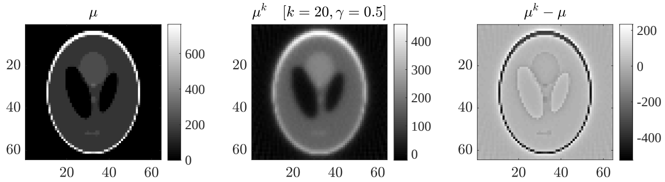


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

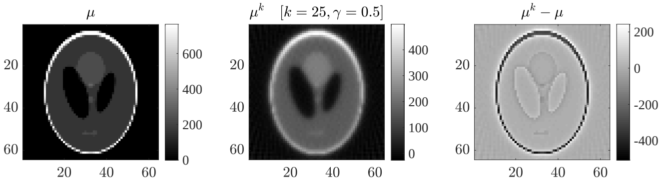


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

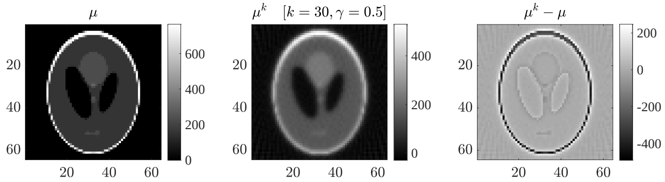


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

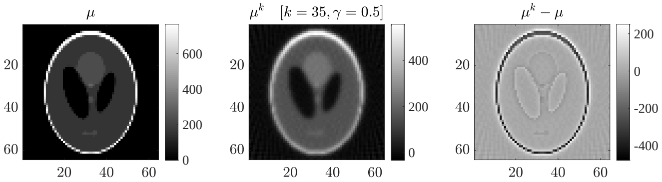


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

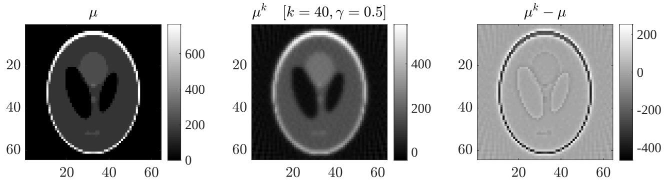


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

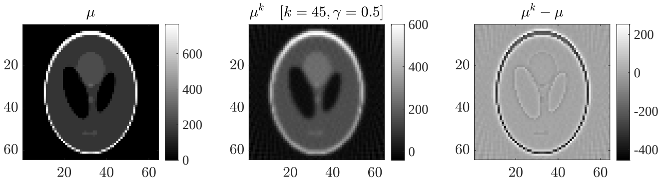


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (no noise)

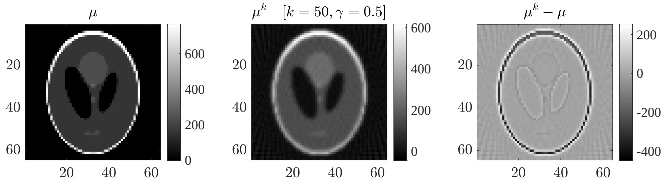


Figure: Reconstruction from noiseless data. Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

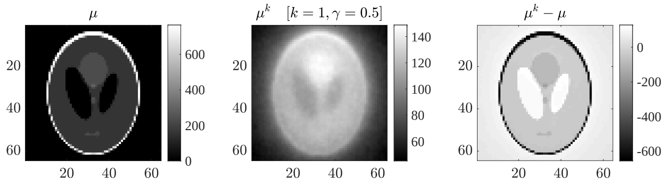


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

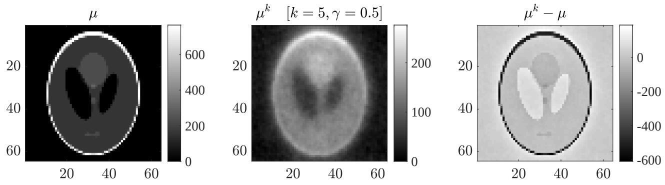


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

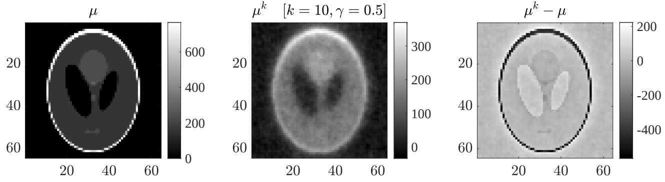


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

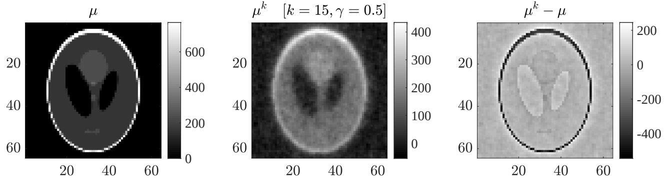


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

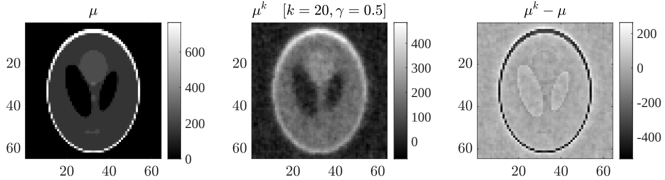


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

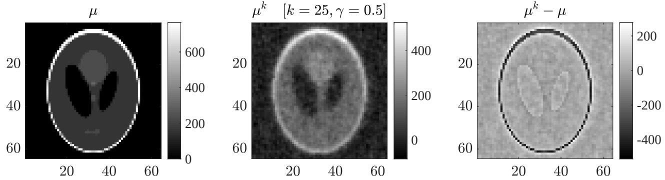


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

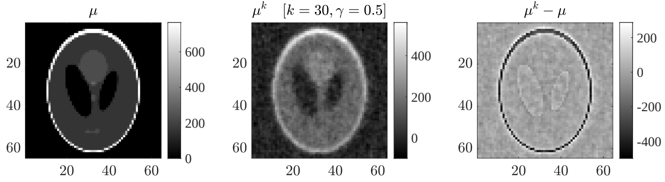


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

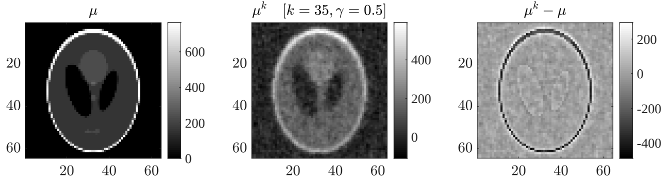


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

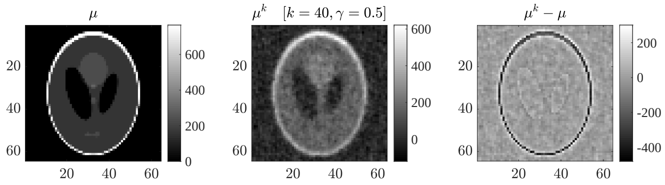


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

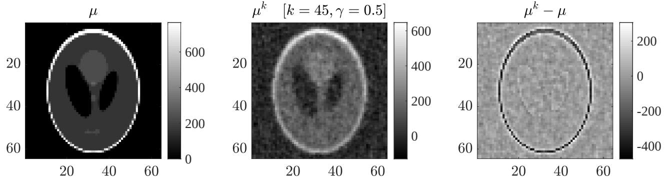


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Gradient descent (noisy data)

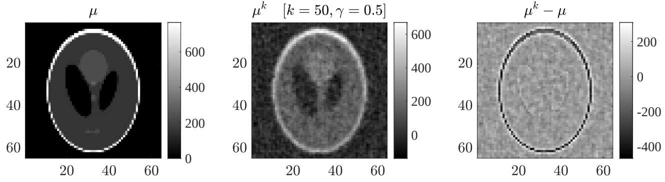


Figure: Reconstruction from noisy data (5% of additive Gaussian noise). Left: ground truth; Middle: reconstruction; Right: error. $n_x = n_y = 64$, $n_\theta = 36$, $\Delta\theta = 5\,^\circ$, $\gamma = 0.5$.

Regularisation

Cost function

Regularisation

Cost function

- In the case where $\boldsymbol{A}$ is poorly conditioned, we can minimize the regularized cost function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\mathcal{R}(\boldsymbol{f})$$

Regularisation

Cost function

- In the case where $\boldsymbol{A}$ is poorly conditioned, we can minimize the regularized cost function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\mathcal{R}(\boldsymbol{f})$$

- α is the regularization parameter

Regularisation

Cost function

- In the case where $\boldsymbol{A}$ is poorly conditioned, we can minimize the regularized cost function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\mathcal{R}(\boldsymbol{f})$$

- α is the regularization parameter
- $\mathcal{R} : \mathbb{R}^N \mapsto \mathbb{R}$ is the regularizer

Regularisation

Cost function

- In the case where $\boldsymbol{A}$ is poorly conditioned, we can minimize the regularized cost function

$$\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\mathcal{R}(\boldsymbol{f})$$

- α is the regularization parameter
- $\mathcal{R} : \mathbb{R}^N \mapsto \mathbb{R}$ is the regularizer
- A 'good' regularizer is small for acceptable images but large otherwise

Tikhonov regularisation

Analytical solution

Tikhonov regularisation

Analytical solution

- A standard choice is $\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{L}\boldsymbol{f}\|^2$, with typical $\boldsymbol{L}$ including the identity or differential operators (e.g., discrete gradient or Laplacian).

Tikhonov regularisation

Analytical solution

- A standard choice is $\mathcal{R}(\boldsymbol{f}) = \|\boldsymbol{L}\boldsymbol{f}\|^2$, with typical $\boldsymbol{L}$ including the identity or differential operators (e.g., discrete gradient or Laplacian).
- By setting the gradient of the cost function to zero, we obtain

$$\boldsymbol{f}_{\text{rls}} = (\boldsymbol{A}^\top \boldsymbol{A} + \alpha \boldsymbol{L}^\top \boldsymbol{L})^{-1} \boldsymbol{A}^\top \boldsymbol{m}$$

It is also known as the Tikhonov solution.

Tikhonov regularisation

Exercise

- Derive the expression of the minimizer of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\|\boldsymbol{L}\boldsymbol{f}\|^2$

Solution

Tikhonov regularisation

Exercise

- Derive the expression of the minimizer of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\|\boldsymbol{L}\boldsymbol{f}\|^2$

Solution

- $\nabla\left(\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2\right) = -2\boldsymbol{A}^\top(\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f})$

Tikhonov regularisation

Exercise

- Derive the expression of the minimizer of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\|\boldsymbol{L}\boldsymbol{f}\|^2$

Solution

- $\nabla\left(\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2\right) = -2\boldsymbol{A}^\top(\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f})$
- $\nabla\left(\alpha\|\boldsymbol{L}\boldsymbol{f}\|^2\right) = 2\alpha\boldsymbol{L}^\top\boldsymbol{L}\boldsymbol{f}$

Tikhonov regularisation

Exercise

- Derive the expression of the minimizer of $\mathcal{C}(\boldsymbol{f}) = \|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha\|\boldsymbol{L}\boldsymbol{f}\|^2$

Solution

- $\nabla\left(\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2\right) = -2\boldsymbol{A}^\top(\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f})$
- $\nabla\left(\alpha\|\boldsymbol{L}\boldsymbol{f}\|^2\right) = 2\alpha\boldsymbol{L}^\top\boldsymbol{L}\boldsymbol{f}$
- $\nabla\left(\|\boldsymbol{m} - \boldsymbol{A}\boldsymbol{f}\|^2\right) + \nabla\left(\alpha\|\boldsymbol{L}\boldsymbol{f}\|^2\right) = 0$ leads to the solution

Philosophy

Data-driven methods

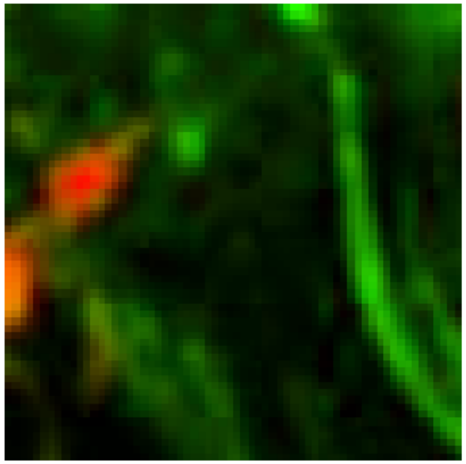
- "Learn" to reconstruct, i.e.,

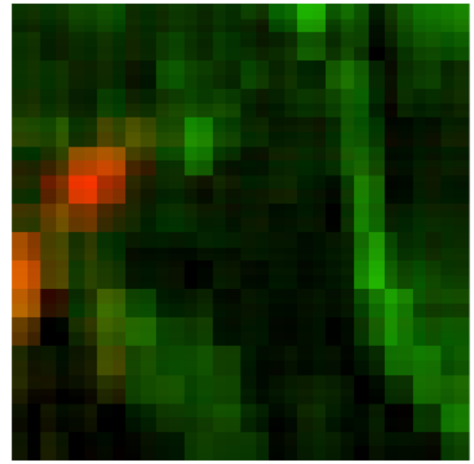
$$\texttt{find and train } \mathcal{H}_\omega \texttt{ such that } \mathcal{H}_\omega(m = \mathcal{A}(f)) \approx f \qquad (3)$$

- **Optimization- vs learning-based methods**

N = 64 × 64 image
M = 333 measurements
N / *M* ≈ 8%

Ground-Truth



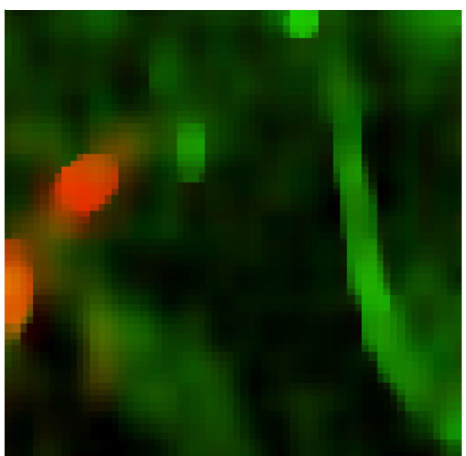


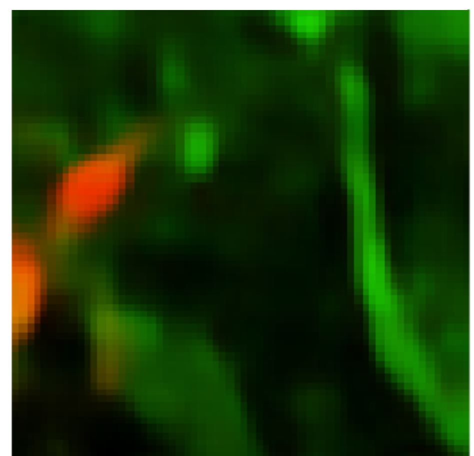
$\Psi(\boldsymbol{f}) = \|\boldsymbol{f}\|_2^2$

(pinv)

$\Psi(\boldsymbol{f}) = \|\nabla \boldsymbol{f}\|_1$

+2.5 dB (wrt pinv)





CNN

+3.5 dB (wrt pinv)

[N. Ducros et al., IEEE ISBI, 2019]

3. Learning-Based Methods

- **Our dream is to find**

$$\mathcal{R}^* : \mathbb{R}^M \mapsto \mathbb{R}^N \text{ such that } \mathcal{R}^*(\boldsymbol{m}) = \boldsymbol{f}^{\text{true}}$$

… able to reconstruct well any image, i.e., something like

$$\mathcal{R}^* \in \arg\min_{\mathcal{R}} \frac{1}{L} \sum_{\ell} \|\mathcal{R}(\boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

Minimum mean square error (MMSE) estimator

… Often intractable

- **We have to reduce the dimension of the solution space**
  - E.g.,

$$\mathcal{R}(\boldsymbol{m}) = \boldsymbol{W}\boldsymbol{m} + \boldsymbol{b},$$

Linear MMSE estimator

3. Learning-Based Methods

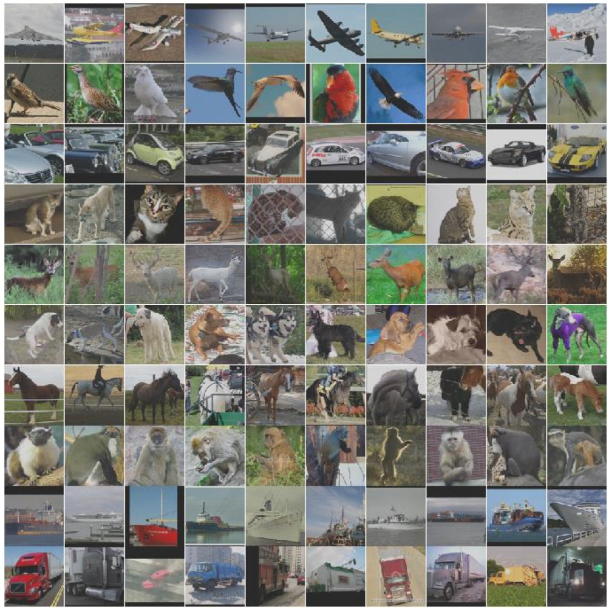
- **Linear MMSE**

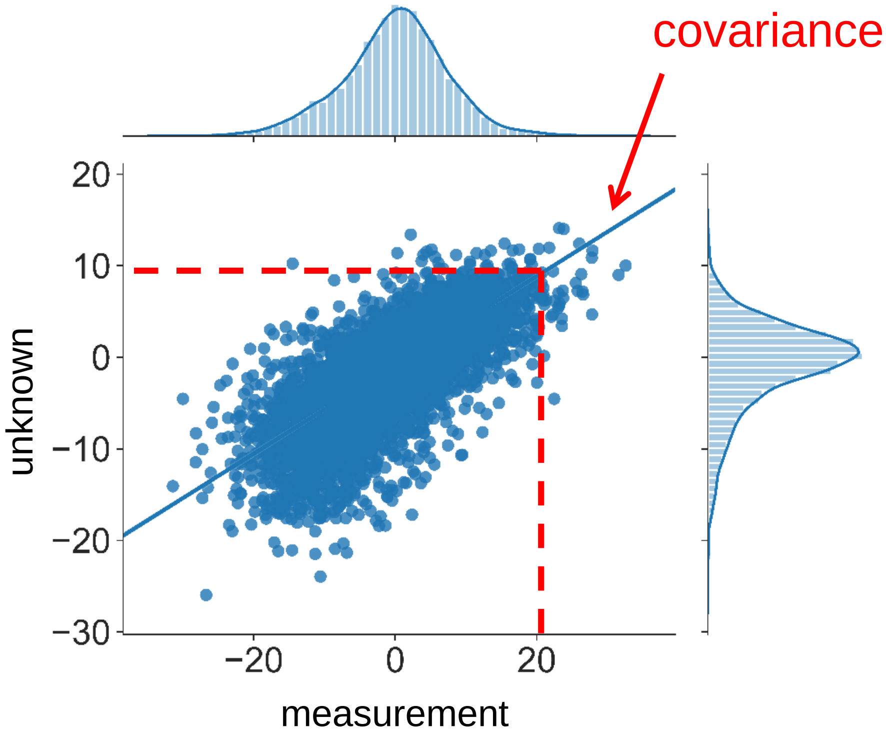
$$\boldsymbol{f}^* = \bar{\boldsymbol{f}} + \boldsymbol{\Sigma}_{f,m}\boldsymbol{\Sigma}_m^{-1}(\boldsymbol{m} - \bar{\boldsymbol{m}})$$

Covariance between measurements and unknowns

Covariance of measurements

slope → covariance





3. Learning-Based Methods

- **Learning approaches *only* reduce the dimension of the solution space to a family of *non linear* mappings**

$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L} \sum_{\ell} \|\mathcal{R}(\boldsymbol{\theta}; \boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

- Training phase
  - Image-measurement pairs $\{\boldsymbol{f}^{(\ell)}; \boldsymbol{m}^{(\ell)})\}_{1\leq\ell\leq L}$
  - Loss (e.g., MSE)
  - Optimization machinery (i.e., PyTorch or TensorFlow)

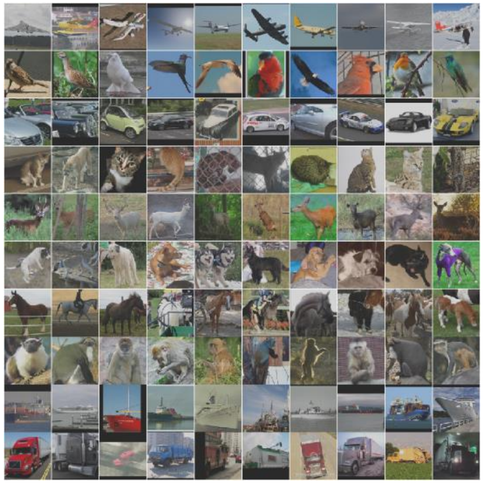
D.P. Kingma and J.L Ba, ICRL, 2015 (> 215k citations)

A. Paszke *et al.*, NEURIPS, 2019 (> 22k citations)

- Reconstruction phase

$$\boldsymbol{f}^* = \mathcal{R}_{\boldsymbol{\theta}^*}(\boldsymbol{m})$$

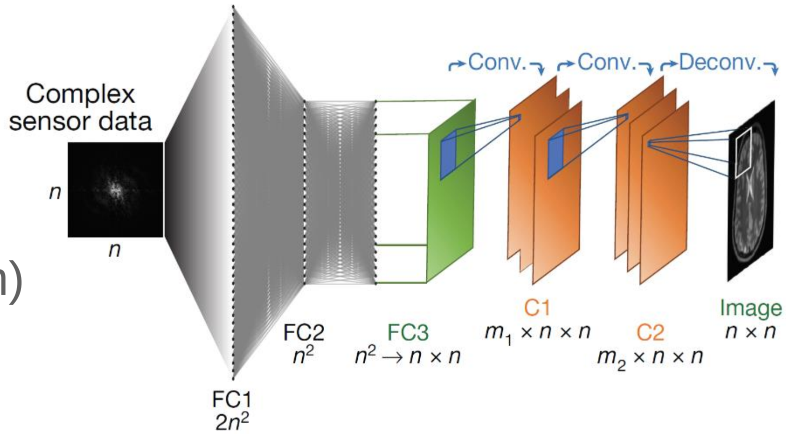
STL-10 dataset



3. Learning-Based Methods

- **Pros**
  - Reconstruction performance
    - Empirically excellent (i.e., almost always outperform optimization-based approaches)
  - Computation times
    - Training phase is slow, i.e., several hours or days
    - Inference is fast, i.e., tens or hundreds of milliseconds
- **Cons**
  - No reconstruction guarantees (*mathematicians don't like it*)
  - Black box (*radiologists don't like it*)
  - Practical issue
    - How to choose the model?

“Automap”
(> 6.10^9 param)



[B. Zhu et al., Nature Letters, 2018] (> 1.5k citations)

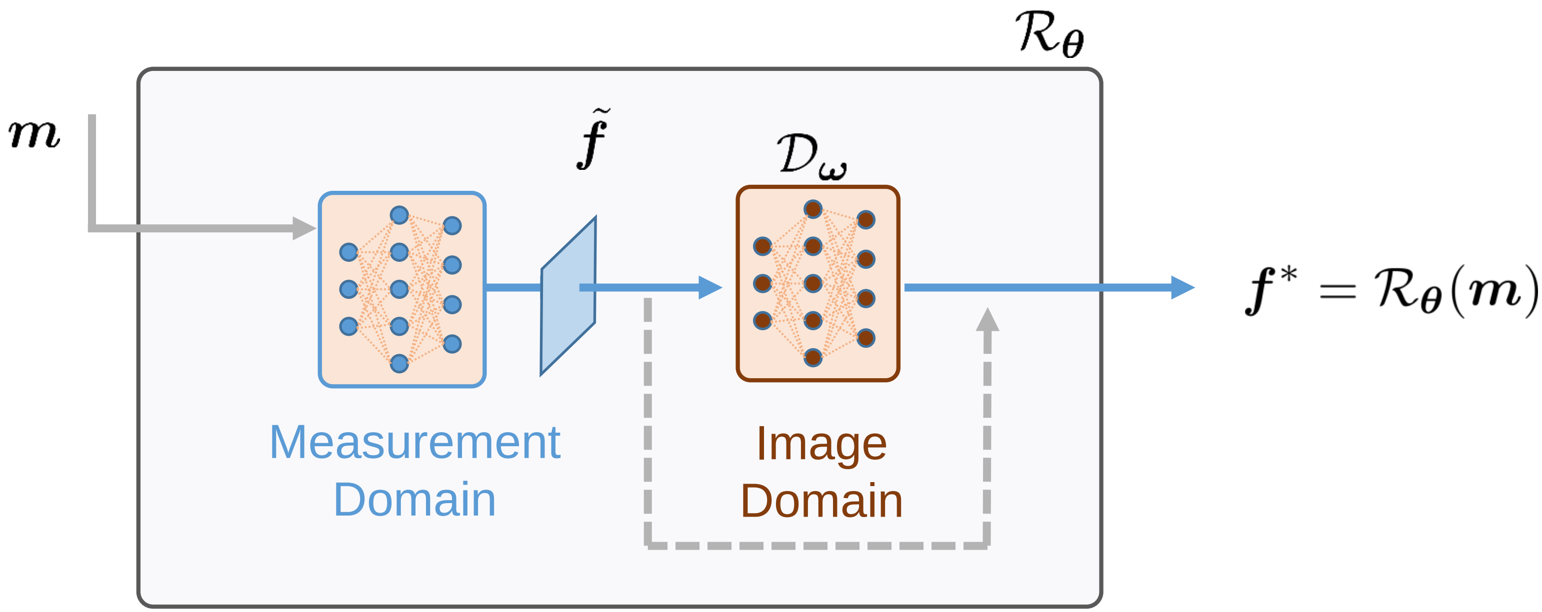
3.1. Post-Processing

- **Two-step methods**

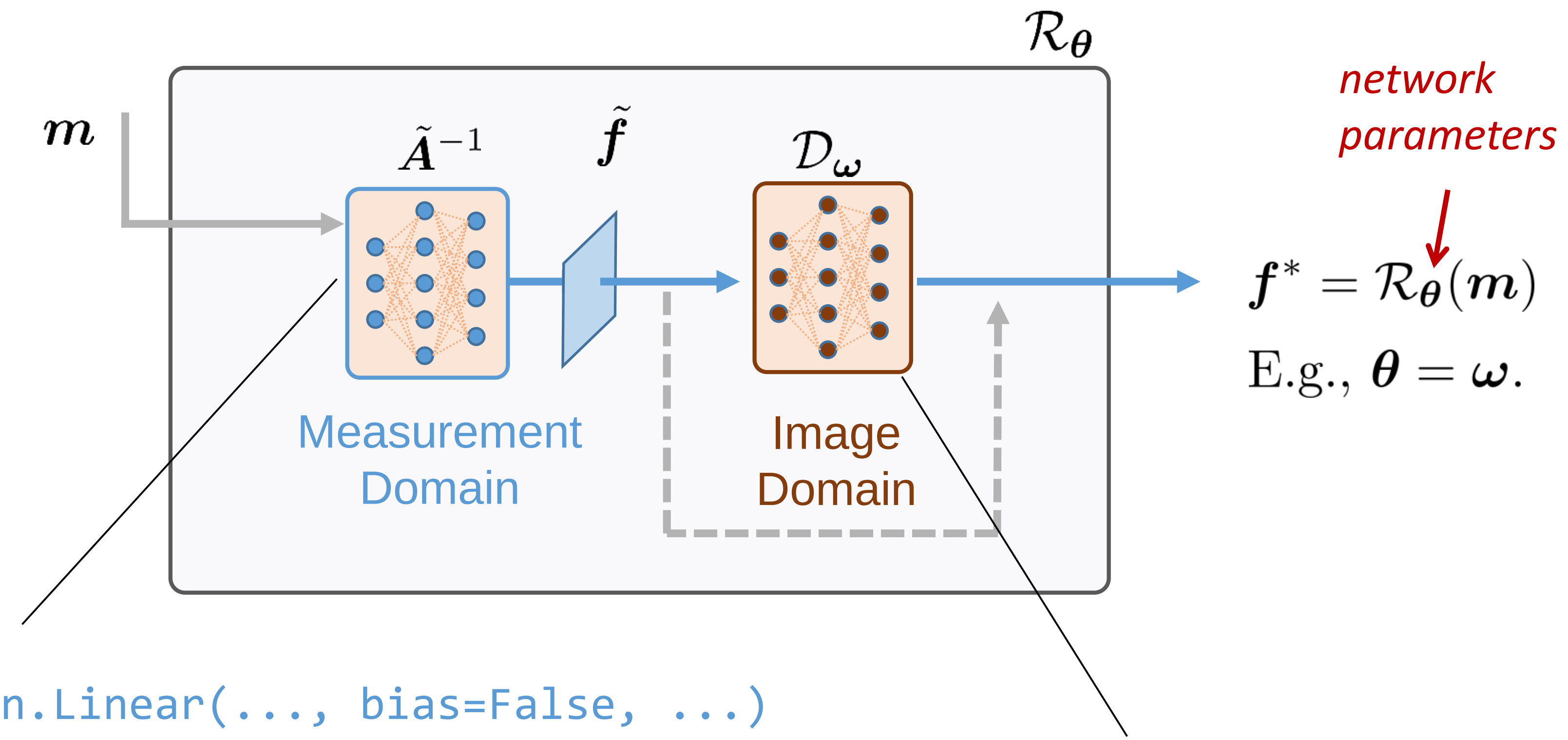
$$\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{A}}^{-1}\boldsymbol{m}$$

$$\boldsymbol{f}^* = \mathcal{D}_{\boldsymbol{\omega}}(\tilde{\boldsymbol{f}}) + \tilde{\boldsymbol{f}}$$

where $\tilde{\boldsymbol{A}^{-1}}$ is an approximate inverse of the forward, i.e., $\tilde{\boldsymbol{A}}^{-1}\boldsymbol{A}\boldsymbol{f} \approx \boldsymbol{f}$



- **Equivalent to a neural network with a frozen layer**




```
Atilde = nn.Linear(..., bias=False, ...)
Atilde.weight.requires_grad = False
```

$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\top$$
$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\dagger \text{ (E.g., } \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^{-1} \text{ for full row rank } A)$$
$$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{I})^{-1}$$

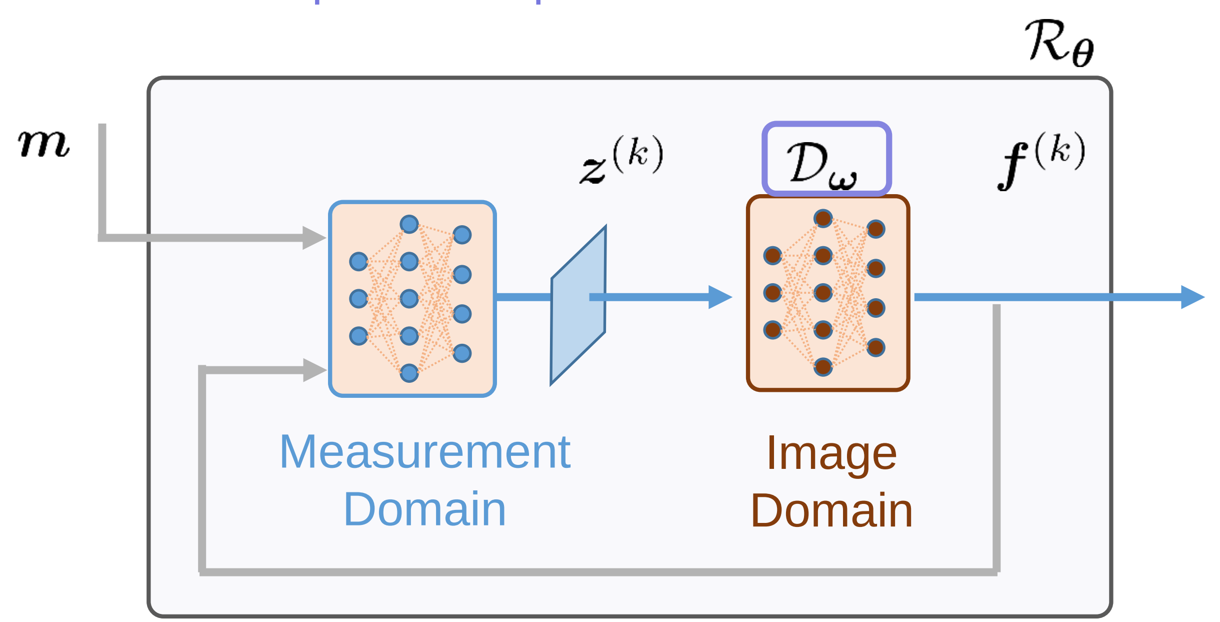
```
D = nn.Module(...)
requires_grad = True
```

Iterative methods

$$z^{(k)} = f^{(k-1)} - \eta \boxed{A^\top (A f^{(k-1)} - m)}$$
$$f^{(k)} = \boxed{\mathrm{prox}_{\lambda\Psi}}(z^{(k)})$$

data fidelity

proximal operator



network parameters

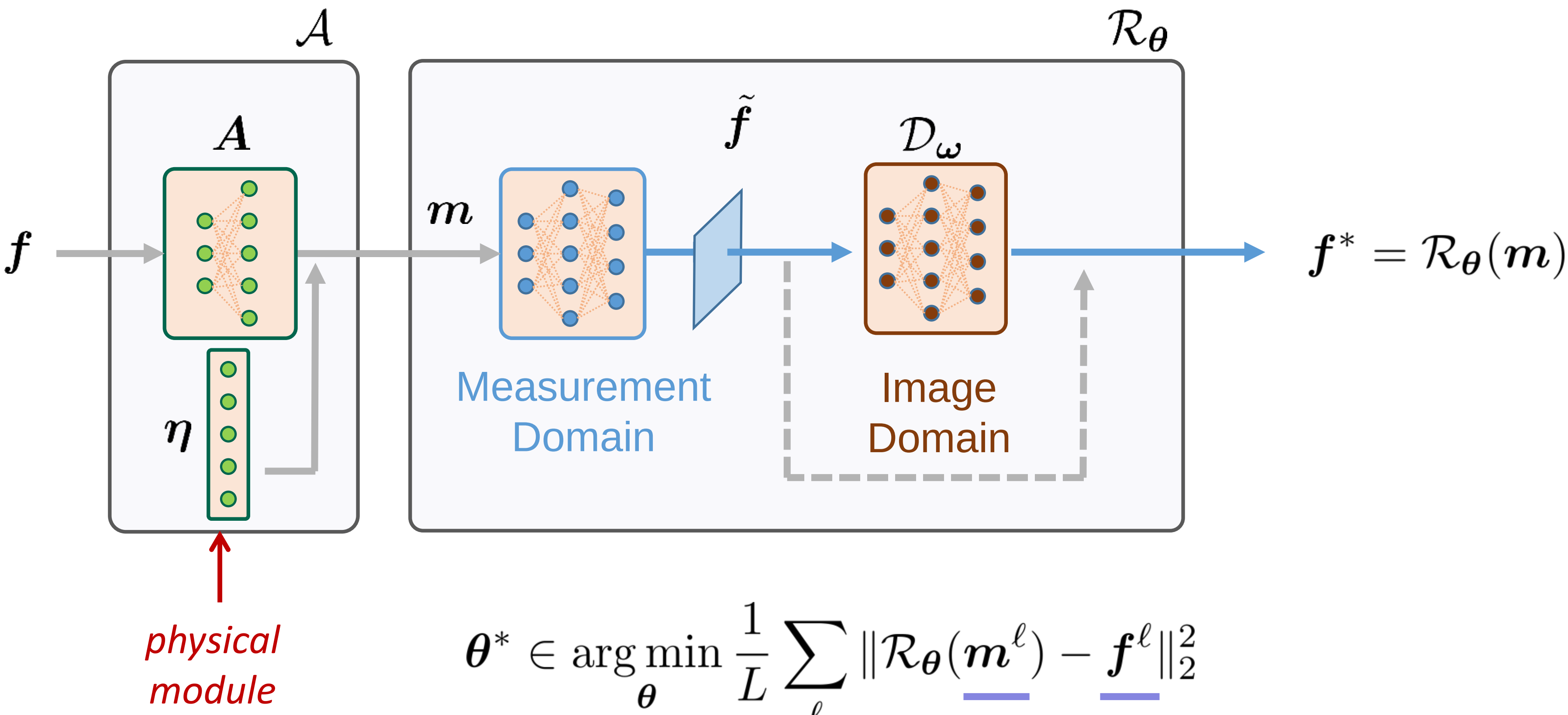
$$f^* = \mathcal{R}_\theta(m) = f^{(K)}$$

Parameters can be shared across iterations or not

E.g., $\theta = \omega$, or $\theta = [\omega^{(1)}, \dots, \omega^{(K)}]$.

3.3. Training

- **With a "physical" module: no need for meas/image pairs**



$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L}\sum_{\ell} \|\mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{m}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

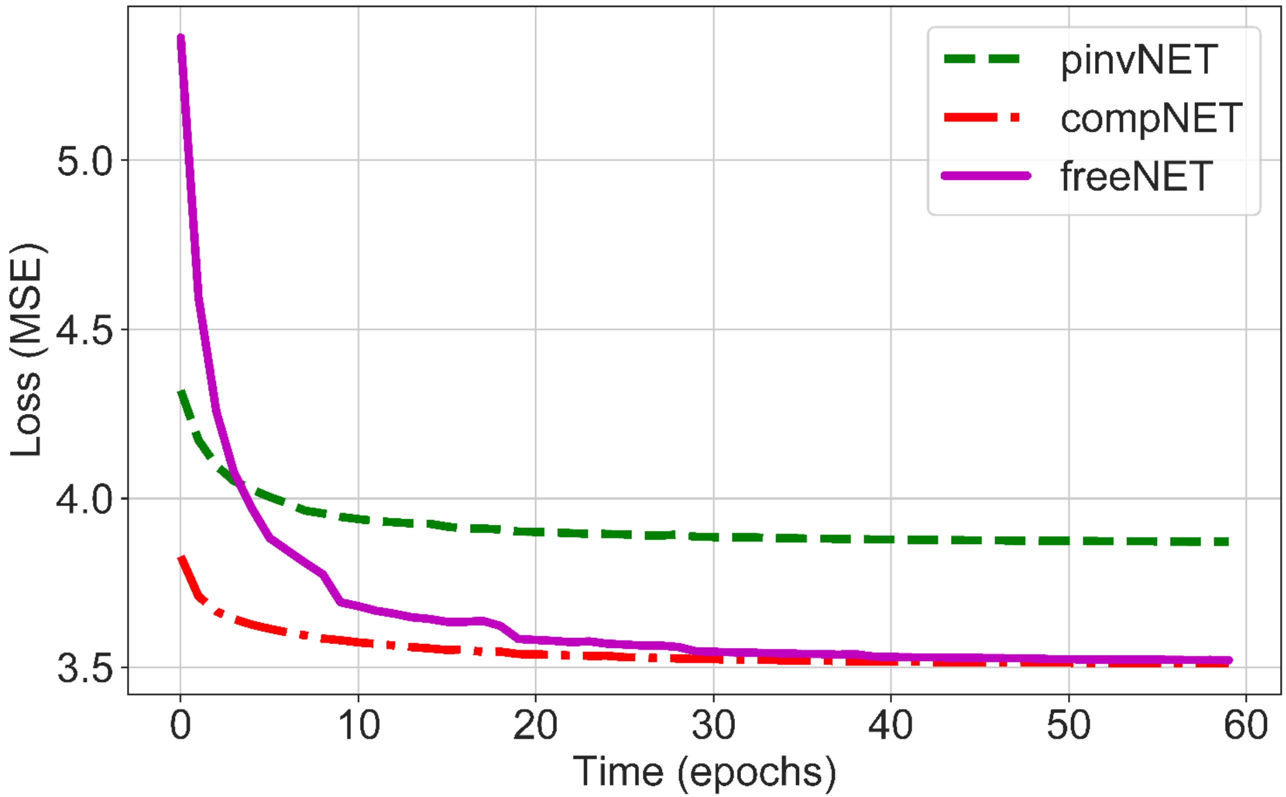
$$\mathcal{A}(\boldsymbol{f}) = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\eta}$$

$$\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \frac{1}{L}\sum_{\ell} \|(\mathcal{R}_{\boldsymbol{\theta}} \circ \mathcal{A})(\boldsymbol{f}^{\ell}) - \boldsymbol{f}^{\ell}\|_2^2$$

3.3. Training

- **STL-10 (training: ~100k images; test: 8k images)**

$$\sum_{\ell \in \mathcal{I}_{\text{test}}} \|\boldsymbol{f}^{(\ell)} - \mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{m}^{(\ell)})\|^2$$



$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^{\dagger}$
$\tilde{\boldsymbol{A}}^{-1} = \boldsymbol{A}^{\top}(\boldsymbol{A}\boldsymbol{A}^{\top} + \boldsymbol{\Sigma})^{-1}$
$\tilde{\boldsymbol{A}}^{-1} = \texttt{Linear}$

1.3M vs 8k trainable parameters

[N. Ducros et al., IEEE ISBI, 2019]

3.4. Plug-and-Play

- **Idea: use denoisers (e.g., BM3D) in place of proximal operators**

$$z^{(k)} = f^{(k-1)} - \eta A^\top (A f^{(k-1)} - m)$$

$$f^{(k)} = \boxed{\mathtt{prox}_{\lambda\Psi}}(z^{(k)}) \quad \Longrightarrow \quad f^{(k)} = \boxed{\mathtt{Denoi}}(z^{(k)}; \sigma)$$

proximal operator — Denoiser — *Noise level* (σ)

- **The denoiser can be data-driven. E.g.** $\mathtt{Denoi} = \mathtt{CNN}_{\theta^*}$ **with**

$$\theta^* \in \arg\min_{\theta} \frac{1}{L} \sum_{\ell} \| \mathtt{CNN}_{\theta}(f^\ell + \underline{\sigma\epsilon}; \sigma) - f^\ell \|_2^2$$

Gaussian noise with variance σ^2 $\quad \epsilon \sim \mathcal{N}(0,1)$

3.4. Plug-and-Play

- **Idea: use denoisers in place of proximal operators**

$$\boldsymbol{z}^{(k)} = \boldsymbol{f}^{(k-1)} - \eta \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{f}^{(k-1)} - \boldsymbol{m})$$

$$\boldsymbol{f}^{(k)} = \boxed{\texttt{Denoi}}(\boldsymbol{z}^{(k)}; \sigma)$$

- **Pros**
  - Training is independent of the forward model
    - Flexibility
    - Applies to any inverse problem
  - Adapt to varying noise levels via hyperparameter
  - Convergence can be guaranteed

- **Cons**
  - Manual tuning of hyperparameter
  - Many iterations required compared to supervised methods (E.g., K = 100—1,000 vs K = 1—10)
    - Longer reconstruction times
    - Higher memory requirement
  - Convergence is not always guaranteed (e.g., Lipschitz constant of denoiser)
  - Underlying prior not always known

3.5. Untrained/unsupervised

- **Deep Generative Models**

$$\boldsymbol{f} = \mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{z})$$

Random vector

$$\boldsymbol{z}^* \leftarrow \min_{\boldsymbol{z}} \|\boldsymbol{A}\mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{z}) - \boldsymbol{m}\|_2^2$$

$$\boldsymbol{f}^* = \mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{z}^*)$$

- **Pros**
  - Only requires measurements from a single acquisition
  - Theoretical guarantees (based on compressed sensing, e.g., considering Gaussian random matrices)
- **Cons**
  - Long and challenging reconstruction
  - Training of DGM is challenging (lots of data/long times)
  - Stability issues (arbitrary forward models, out-of-distribution images, etc.)

3.5. Untrained/unsupervised

- **Deep Image Priors**

$$\boldsymbol{f} = \mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{z})$$

Fixed random vector

$$\boldsymbol{\theta}^* \leftarrow \min_{\boldsymbol{\theta}} \|\boldsymbol{A}\mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{z}) - \boldsymbol{m}\|_2^2$$

$$\boldsymbol{f}^* = \mathcal{R}_{\boldsymbol{\theta}^*}(\boldsymbol{z})$$

Note: Minimization must be stopped before convergence (tends to noise otherwise)

- **Pros**
  - Only requires measurements from a single acquisition
  - The reconstruction quality is surprisingly good

- **Cons**
  - Long reconstruction times
  - No guarantees

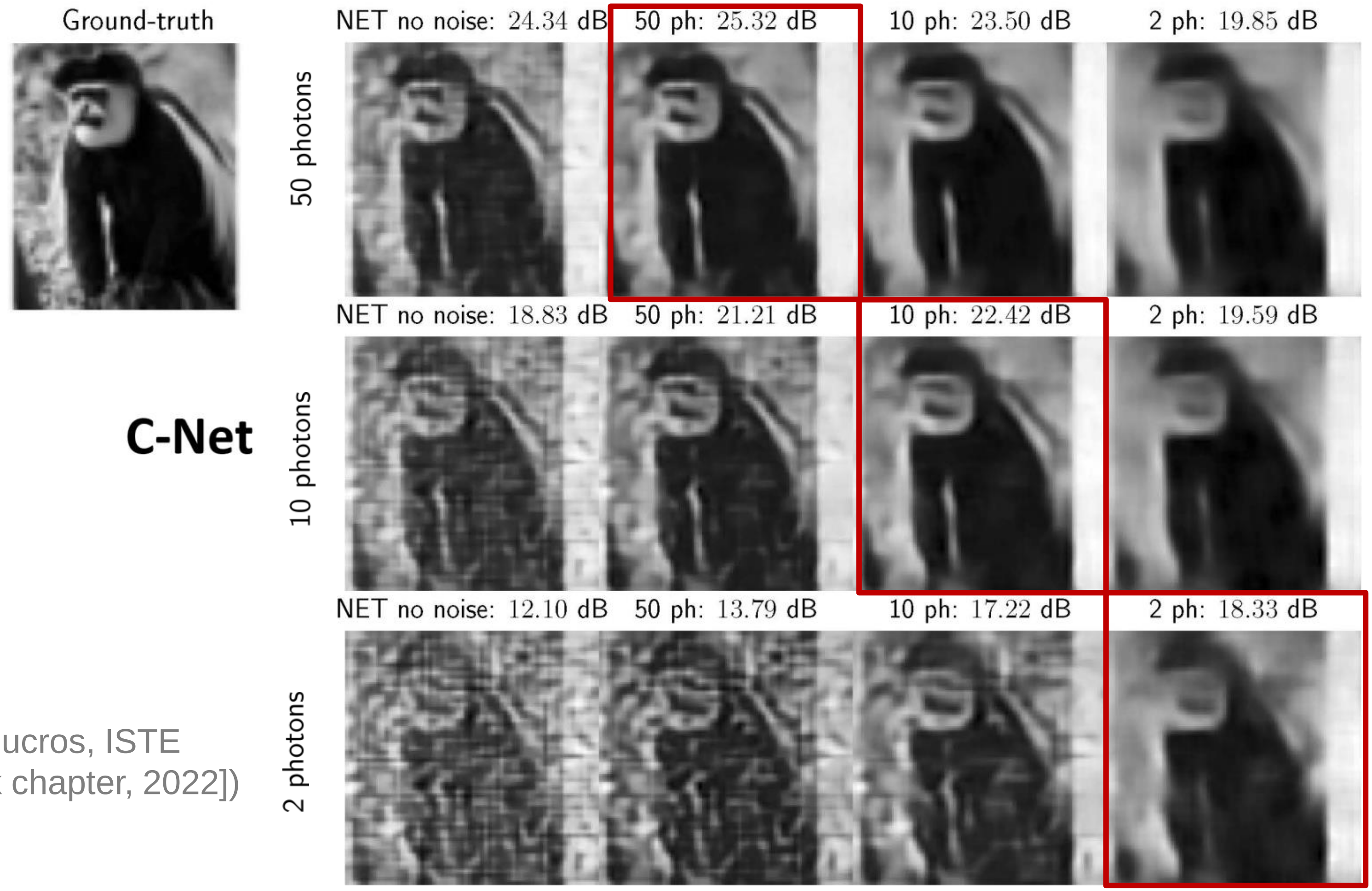
Conclusions

Benchmark

	Memory Requirement	Recon-struction Time (inference)	Training	Hyperparam/ Comment
Supervised	Low to intermediate	1—10	$\boldsymbol{A}$ $\{\boldsymbol{f}^\ell\}$	No adaptation (forward, noise)
PnP	Intermediate to high	100—1,000	$\{\boldsymbol{f}^\ell\}$	Noise level
Untrained	Usually low	> 1,000	—	Number of iterations

Noise robustness

Increasing training noise



C-Net

Increasing test noise

[N. Ducros, ISTE Book chapter, 2022])

Conclusions

- **Data-driven DL-based approaches for image reconstruction are**
  - Powerful!
  - No longer black boxes
  - Supervised, PnP, based on generative models, untrained, etc.

- **Supervised vs PnP methods**
  - Supervised methods usually require fewer parameters
  - Supervised methods performs very well
  - PnP methods adapts to different
    - Imaging modalities (i.e., forward models)
    - Noise levels

- **Warning**
  - Handling noise is still an issue.
    - Evaluate the robustness to noise level deviations
    - Train with noise (supervised)
    - Tune hyperparameters (PnP)

→ **Hands-on session**

https://github.com/openspyrit/spyrit-examples/tree/master/2025_DLMIS

Acoustic wave

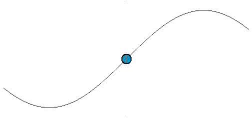


Figure: Wave

$$p(x,t) = p_0 \sin\left(2\pi f\left(t - \frac{x}{v}\right)\right) = p_0 \sin\left(\omega t - kx\right)$$

Mechanical wave

- Wave of pressure *p(x,t)* (and displacement)
- Compressible medium
- Particles in the medium have only minor displacement

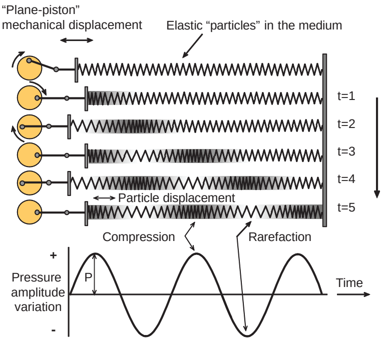
Wave

- f is the frequency
- v is the (phase) velocity
- $\lambda = v/f$ is the wavelength
- $k = 2\pi/\lambda$ is the wave number

Acoustic wave

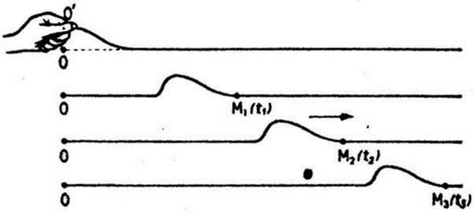
Longitudinal wave

- Also known as *compression* wave
- Directions of oscillation and propagation are the same



Adapted from [JT Bushberg, 2011, chap.14]

Acoustic wave



Transverse wave

Acoustic wave

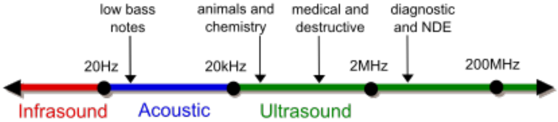


Figure: Frequency range corresponding to ultrasounds. [wikipedia]

Wave frequency f

- Medical ultrasound in the range 2–10 MHz
- Unaffected by changes in acoustic properties

Acoustic wave

Wave velocity v

- Velocity of the wavefront
- *Not* the velocity of the particle

Acoustic wave

Wave velocity v

- Depends on the propagation medium

$$v = \sqrt{\frac{K}{\rho}}$$

- ρ is the density (in g/cm^3), K is the bulk modulus (in Pa)
- K measures of the stiffness (resistance of a medium to being compressed)

Acoustic wave

Exercice

- Given the following velocities in different media, compute the resulting resolutions for a 5-MHz probe
- Air: $v =$330 m/s, soft tissue: $v =$1540 m/s, skull bone: 4080 m/s.

Solution

Acoustic wave

Exercice

- Given the following velocities in different media, compute the resulting resolutions for a 5-MHz probe
- Air: $v =$330 m/s, soft tissue: $v =$1540 m/s, skull bone: 4080 m/s.

Solution

- $\lambda_{\text{air}} \approx 66\ \mu$m

Acoustic wave

Exercice

- Given the following velocities in different media, compute the resulting resolutions for a 5-MHz probe
- Air: $v =$330 m/s, soft tissue: $v =$1540 m/s, skull bone: 4080 m/s.

Solution

- $\lambda_{\mathrm{air}} \approx 66$ μm
- $\lambda_{\mathrm{soft}} \approx 308$ μm

Acoustic wave

Exercice

- Given the following velocities in different media, compute the resulting resolutions for a 5-MHz probe
- Air: v =330 m/s, soft tissue: v =1540 m/s, skull bone: 4080 m/s.

Solution

- $\lambda_{\text{air}} \approx 66\ \mu$m
- $\lambda_{\text{soft}} \approx 308\ \mu$m
- $\lambda_{\text{bone}} \approx 816\ \mu$m

Ultrasound-Matter interaction

Acoustic Impedance

- For plane waves, defined as

$$Z = \rho v$$

- Related to the 'flexibility' of the medium (think to springs)
- The SI unit is kg/(m^{2}s), referred to as rayl
- Typical values ($\times 10^6$ rayls). Air: $Z = 4 \times 10^{-4}$, water: $Z = 1.48$, liver: $Z = 1.65$, muscle: $Z = 1.71$, skull bone: $Z = 7.8$.

Acoustic Intensity

- Sound power per unit area in W/m^2
- For plane waves

$$I = \frac{p^2}{\rho v}$$

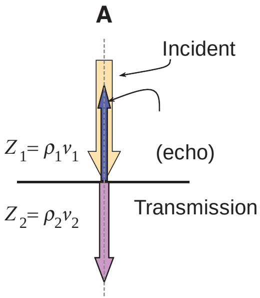
Ultrasound-Matter interaction

Reflection

- Intensity reflection coefficient defined as

$$R = \left(\frac{Z_2 - Z_1}{Z_2 + Z_1}\right)^2$$

- Percentage of the incident amount of power per unit area that is reflected back



Adapted from [JT Bushberg, 2011, chap.14]

Ultrasound-Matter interaction

Exercice

- Compute the reflection coefficient of an air-water and a water-liver interface
- Water: $Z = 1.48$, liver: $Z = 1.65$

Solution

Ultrasound-Matter interaction

Exercice

- Compute the reflection coefficient of an air-water and a water-liver interface
- Water: $Z = 1.48$, liver: $Z = 1.65$

Solution

- $R_{\text{air}-\text{water}} \approx 0.9999$

Ultrasound-Matter interaction

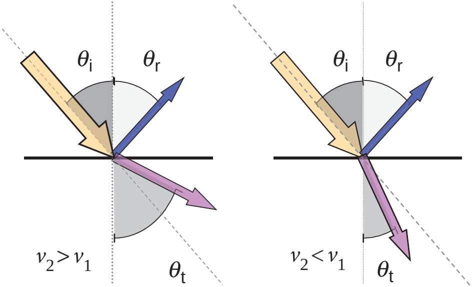
Exercice

- Compute the reflection coefficient of an air-water and a water-liver interface
- Water: $Z = 1.48$, liver: $Z = 1.65$

Solution

- $R_{\text{air}-\text{water}} \approx 0.9999$
- $R_{\text{water}-\text{liver}} \approx 0.0029$

Ultrasound-Matter interaction



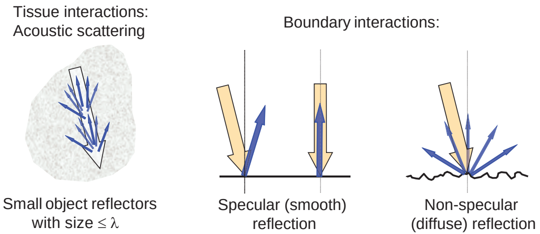
Adapted from [JT Bushberg, 2011, chap.14]

Refraction

- Change in the direction of the transmitted wave
- Snell–Descartes law

$$\frac{\sin \theta_{\mathrm{t}}}{\sin \theta_{\mathrm{i}}} = \frac{v_2}{v_1}$$

Ultrasound-Matter interaction



Adapted from [JT Bushberg, 2011, chap.14]

Scattering

- Due to object/interfaces about the size or smaller than wavelength
- Non specular reflectors reflect in all directions
- Resulting echos have a small amplitude
- Scattering from non specular reflectors increase with frequency

Ultrasound-Matter interaction

Attenuation

- Loss of acoustic energy during propagation
- Originates from many different effects, e.g., scattering, absorption, refraction
- Denoted μ and expressed in dB/m

$$I = I_0 \exp(-\mu x)$$

- Attenuation depends linearly on the wave frequency, with the common approximation

$$\mu(f) = \frac{\ln 10}{10} \hat{\mu} f$$

with $\hat{\mu}$ a constant expressed in (dB/m)/Hz

Ultrasound-Matter interaction

Exercise

- Assuming $\hat{\mu} = 0.5$ (dB/cm)/MHz for soft tissues, calculate the intensity of the echo generated by a 10-MHz wave having travelled 5 cm in the liver before being reflected at the boundary with a water pocket.

Solution

Ultrasound-Matter interaction

Exercise

- Assuming $\hat{\mu} = 0.5$ (dB/cm)/MHz for soft tissues, calculate the intensity of the echo generated by a 10-MHz wave having travelled 5 cm in the liver before being reflected at the boundary with a water pocket.

Solution

- $I_1 = I_0 \exp -(\hat{\mu} f x)$, with $10 \log \left(I_0 / I_1 \right) = 0.5 \times 5 \times 10 =$ 25 dB

Ultrasound-Matter interaction

Exercise

- Assuming $\hat{\mu} = 0.5$ (dB/cm)/MHz for soft tissues, calculate the intensity of the echo generated by a 10-MHz wave having travelled 5 cm in the liver before being reflected at the boundary with a water pocket.

Solution

- $I_1 = I_0 \exp -(\hat{\mu} f x)$, with $10 \log (I_0/I_1) = 0.5 \times 5 \times 10 =$ 25 dB
- $I_2 = R I_1$, with $10 \log (I_1/I_2) = -10 \log R \approx$ 25 dB

Ultrasound-Matter interaction

Exercise

- Assuming $\hat{\mu} = 0.5$ (dB/cm)/MHz for soft tissues, calculate the intensity of the echo generated by a 10-MHz wave having travelled 5 cm in the liver before being reflected at the boundary with a water pocket.

Solution

- $I_1 = I_0 \exp -(\hat{\mu} f x)$, with $10 \log (I_0/I_1) = 0.5 \times 5 \times 10 =$ 25 dB
- $I_2 = R I_1$, with $10 \log (I_1/I_2) = -10 \log R \approx$ 25 dB
- $I_3 = I_2 \exp(-\hat{\mu} f x)$, with $10 \log (I_2/I_3) = 0.5 \times 5 \times 10 =$ 25 dB

Ultrasound-Matter interaction

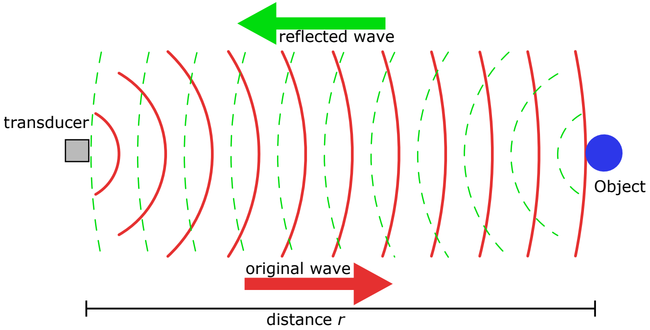
Exercise

- Assuming $\hat{\mu} = 0.5$ (dB/cm)/MHz for soft tissues, calculate the intensity of the echo generated by a 10-MHz wave having travelled 5 cm in the liver before being reflected at the boundary with a water pocket.

Solution

- $I_1 = I_0 \exp -(\hat{\mu} f x)$, with $10 \log (I_0/I_1) = 0.5 \times 5 \times 10 = 25$ dB
- $I_2 = R I_1$, with $10 \log (I_1/I_2) = -10 \log R \approx 25$ dB
- $I_3 = I_2 \exp(-\hat{\mu} f x)$, with $10 \log (I_2/I_3) = 0.5 \times 5 \times 10 = 25$ dB
- The total attenuation is about 75 dB

Ultrasound transducer



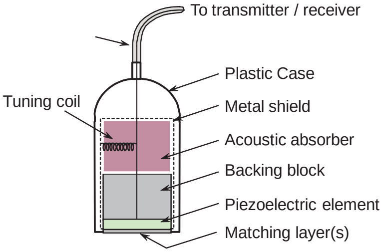
Transducer basics

- Both produces and detects ultrasounds
- Made of ceramic elements that can convert electrical energy into mechanical energy (and the opposite)

Ultrasound transducer

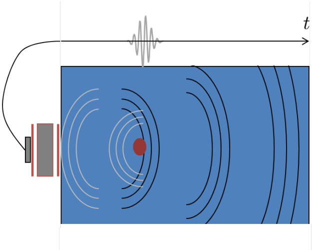
The piezoelectric element is the main component

- Usually lead-zirconate-titanate (PZT)
- Characterized by a natural resonance frequency f_0 (that depends on its thickness)
- Can generate an acoustic pulse by applying a voltage spike (∼150V during 1 μs)



Adapted from [JT Bushberg, 2011, chap.14]

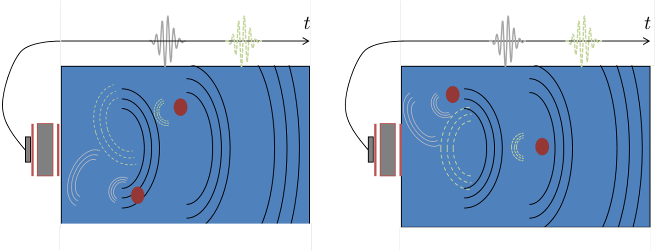
Pulse-echo ultrasound



Basic idea

- Pulse delays tell about the location of the reflectors

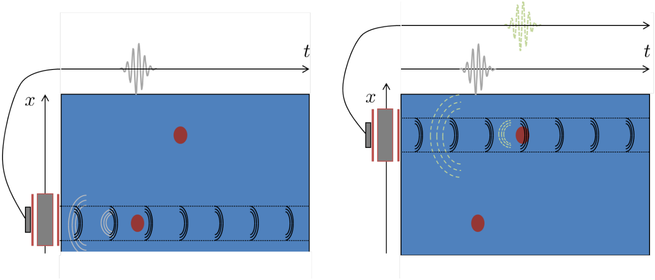
Pulse-echo ultrasound



Challenge

- Pulse delays only tell about the *distance* of the reflectors
- Same signals can be obtained in the different media

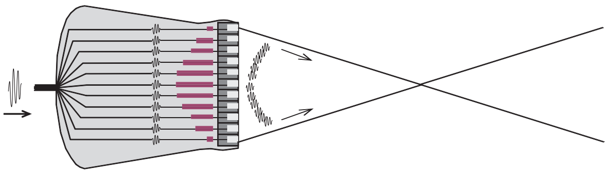
Pulse-echo ultrasound



Solution

- Use converging beams that focus energy in the medium
- Scan, i.e., deliver acoustic energy along multiple lines

Pulse-echo ultrasound

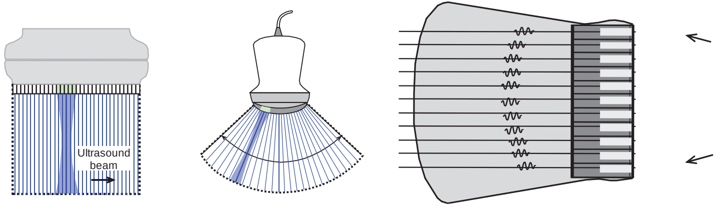


Adapted from [JT Bushberg, 2011, chap.14]

Phased-array transducers

- Use multiple elements at the same time to focus ultrasounds
- Different pulse delays are applied to elements
- Tuning delays permits to select focusing depth and/or angle

Pulse-echo ultrasound



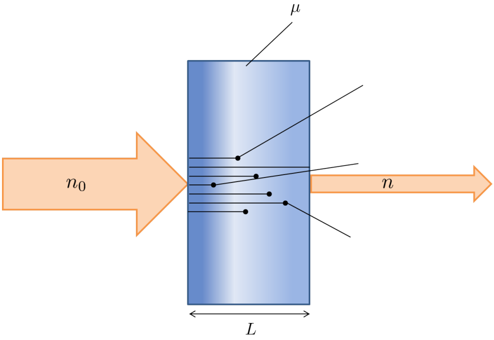
Adapted from [JT Bushberg, 2011, chap.14]

Measured signal

- For a given excitation direction, all the elements record simultaneously
- A set of time-resolved signal are collected

$$\{m_n(t)\},\ n \in \{-N, \dots, N\} \tag{5}$$

Light-matter interaction (Attenuation)



Photon can be absorbed

- Photoelectric effect
- Pair production

Photon can be deflected

- Compton scattering
- Rayleigh scattering

Electromagnetic radiation

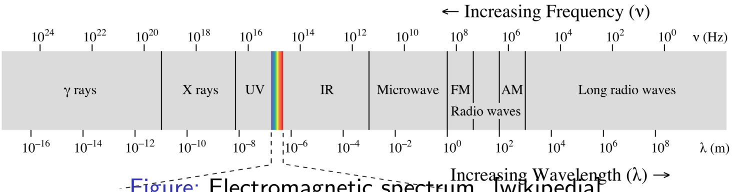


Figure: Electromagnetic spectrum. [wikipedia]

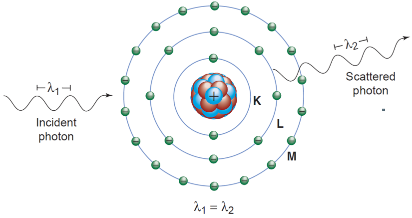
Main relations

- $E = h\nu$ is the photon energy (in eV)
- $\lambda = c/\nu$ is the wavelength

Constants and units

- $h = 6.63 \times 10^{-34}$ J.s is the Planck constant
- $1\text{eV} = 1.60 \times 10^{-19} J$
- $c = 3 \times 10^{8}$ ms^{-1} (in vacuum)

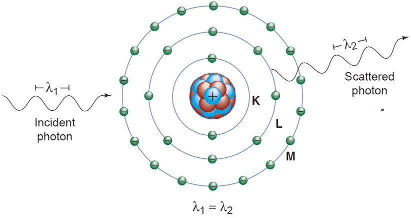
Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

Rayleigh scattering

Light-matter interaction

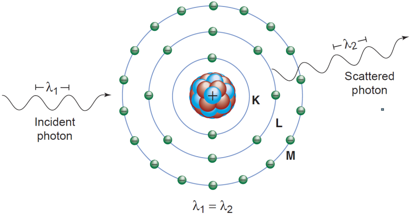


Adapted from [JT Bushberg, 2011, chap.3]

Rayleigh scattering

- Elastic scattering, i.e., kinetic energy is conserved

Light-matter interaction

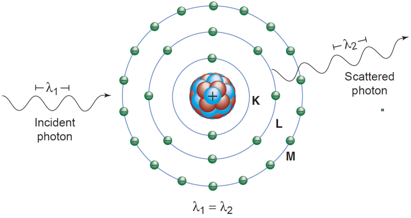


Adapted from [JT Bushberg, 2011, chap.3]

Rayleigh scattering

- Elastic scattering, i.e., kinetic energy is conserved
- $E_2 = E_1$

Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

Rayleigh scattering

- Elastic scattering, i.e., kinetic energy is conserved
- $E_2 = E_1$
- small scattering angle

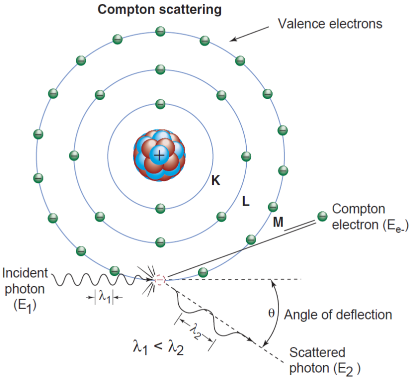
Light-matter interaction

Compton scattering

Light-matter interaction

Compton scattering

- Loosely bound electron

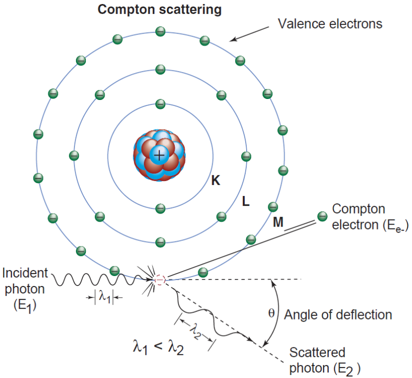


Adapted from [JT Bushberg, 2011, chap.3]

Light-matter interaction

Compton scattering

- Loosely bound electron
- Inelastic scattering, i.e., kinetic energy of scattered particle *not* conserved



Adapted from [JT Bushberg, 2011, chap.3]

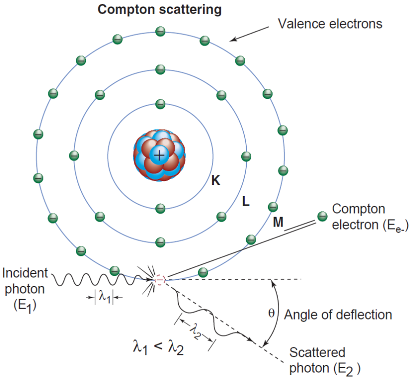
Light-matter interaction

Compton scattering

- Loosely bound electron
- Inelastic scattering, i.e., kinetic energy of scattered particle *not* conserved
- Energy and momentum conservation lead to

$$\frac{1}{E_2} - \frac{1}{E_1} = \frac{1 - \cos\theta}{E_e}$$

with $E_e = m_e c^2$



Adapted from [JT Bushberg, 2011, chap.3]

Light-matter interaction

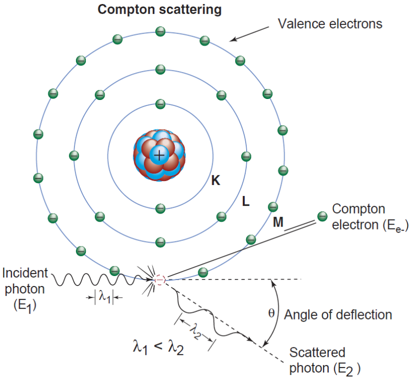
Compton scattering

- Loosely bound electron
- Inelastic scattering, i.e., kinetic energy of scattered particle *not* conserved
- Energy and momentum conservation lead to

$$\frac{1}{E_2} - \frac{1}{E_1} = \frac{1-\cos\theta}{E_e}$$

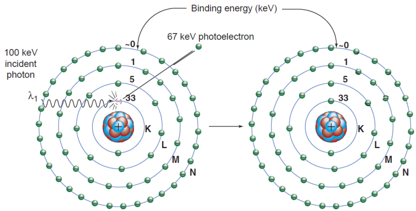
with $E_e = m_e c^2$

- Continuum of scattering angle and energy



Adapted from [JT Bushberg, 2011, chap.3]

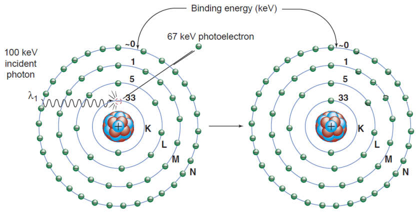
Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

Photoelectric effect

Light-matter interaction

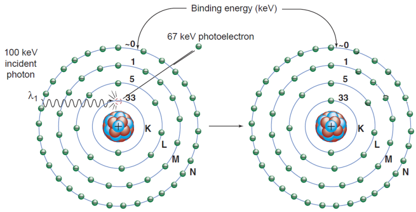


Adapted from [JT Bushberg, 2011, chap.3]

Photoelectric effect

- Absorption of photon/emission of a (photo)electron

Light-matter interaction

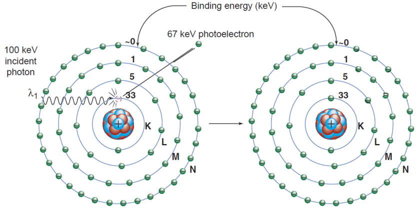


Adapted from [JT Bushberg, 2011, chap.3]

Photoelectric effect

- Absorption of photon/emission of a (photo)electron
- Electronic shell discontinuities (Quantum effect)

Light-matter interaction

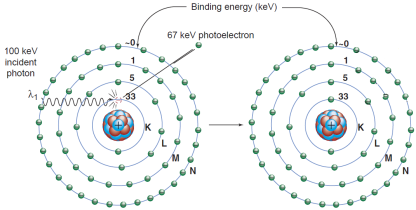


Adapted from [JT Bushberg, 2011, chap.3]

Photoelectric effect

- Absorption of photon/emission of a (photo)electron
- Electronic shell discontinuities (Quantum effect)
- Atom ionization

Light-matter interaction

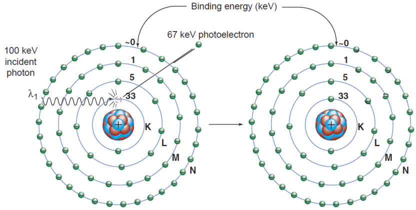


Adapted from [JT Bushberg, 2011, chap.3]

Photoelectric effect

- Absorption of photon/emission of a (photo)electron
- Electronic shell discontinuities (Quantum effect)
- Atom ionization
- Radiative atomic relaxation (fluorescence and Auger)

Light-matter interaction



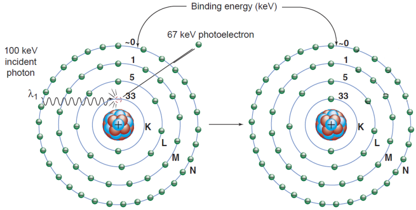
Adapted from [JT Bushberg, 2011, chap.3]

Exercise

- Calculate the energy and wavelength of the emitted radiations (fluorescence X-ray)

Solution

Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

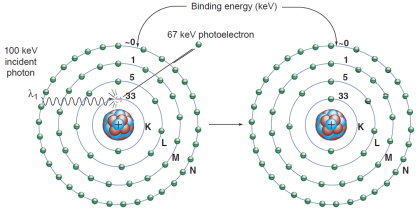
Exercise

- Calculate the energy and wavelength of the emitted radiations (fluorescence X-ray)

Solution

- K_α: $33 - 5 = 28$ keV; K_β: $33 - 1 = 32$ keV; K_γ: $33 - 0 = 33$ keV

Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

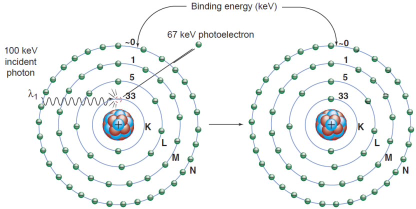
Exercise

- Calculate the energy and wavelength of the emitted radiations (fluorescence X-ray)

Solution

- K_α: $33 - 5 = 28$ keV; K_β: $33 - 1 = 32$ keV; K_γ: $33 - 0 = 33$ keV
- L_α: $5 - 1 = 4$ keV; L_β: $5 - 0 = 5$ keV

Light-matter interaction



Adapted from [JT Bushberg, 2011, chap.3]

Exercise

- Calculate the energy and wavelength of the emitted radiations (fluorescence X-ray)

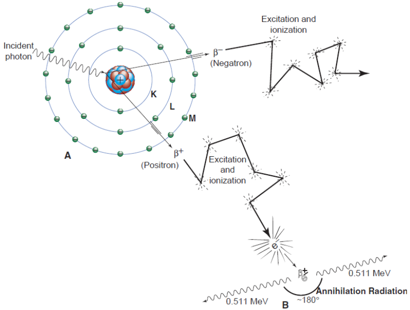
Solution

- K_α: $33 - 5 = 28$ keV; K_β: $33 - 1 = 32$ keV; K_γ: $33 - 0 = 33$ keV
- L_α: $5 - 1 = 4$ keV; L_β: $5 - 0 = 5$ keV
- M_α: $1 - 0 = 1$ keV

Light-matter interaction

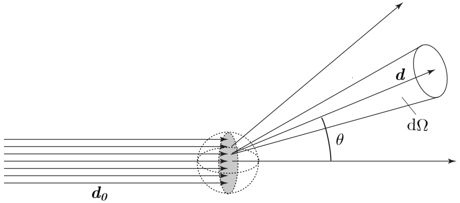
Electron pair production

- Positron-negatron pair is created
- $E_{\text{incident}} > 2 m_e c^2 = 1.02$ MeV



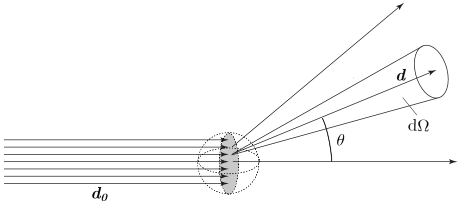
Adapted from [JT Bushberg, 2011, chap.3]

Cross-section



Definition

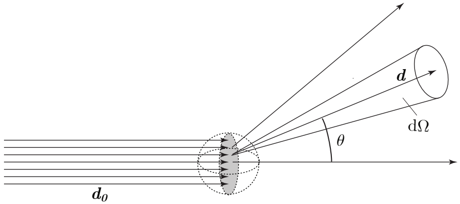
Cross-section



Definition

- We assume that a single particle is illuminated by a plane wave propagating along $\boldsymbol{d}_0$ with an intensity I_0 (W.cm^{-2})

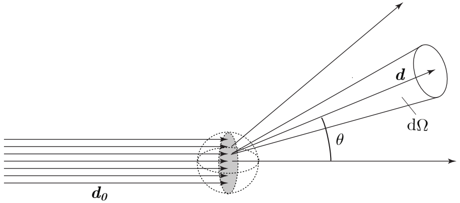
Cross-section



Definition

- We assume that a single particle is illuminated by a plane wave propagating along $\boldsymbol{d}_0$ with an intensity I_0 (W.cm^{-2})
- The power I (in W) is measured along $\boldsymbol{d}$ such that $\boldsymbol{d} \cdot \boldsymbol{d}_0 = \cos\theta$

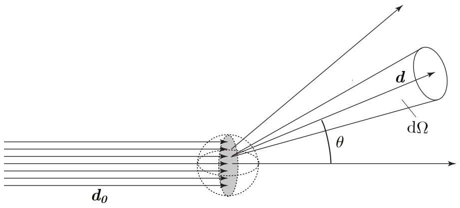
Cross-section



Definition

- We assume that a single particle is illuminated by a plane wave propagating along $\boldsymbol{d}_0$ with an intensity I_0 (W.cm^{-2})
- The power I (in W) is measured along $\boldsymbol{d}$ such that $\boldsymbol{d} \cdot \boldsymbol{d}_0 = \cos\theta$
- The (total) cross-section is defined by $\sigma(\theta) = \frac{I(\theta)}{I_0}$

Cross-section



Definition

- We assume that a single particle is illuminated by a plane wave propagating along $\boldsymbol{d}_0$ with an intensity I_0 (W.cm^{-2})
- The power I (in W) is measured along $\boldsymbol{d}$ such that $\boldsymbol{d} \cdot \boldsymbol{d}_0 = \cos\theta$
- The (total) cross-section is defined by $\sigma(\theta) = \frac{I(\theta)}{I_0}$
- The cross-section is an area

Cross-section

Link to attenuation

- For a homogeneous medium with density ρ (in cm^{-3}), we have:

$$\mu = \sigma\rho$$

- Mean free path is defined as $l = \mu^{-1}$ (in cm).
- Common units: $[\sigma]$ = cm^2.g^{-1}, $[\rho]$ = g.cm^{-3}, and $[\sigma]$ = b where b $= 10^{-28}$ m^2 stands for the barn.

- All attenuation effects sum up

$$\sigma = \sigma_{\text{pe}} + \sigma_{\text{coh}} + \sigma_{\text{incoh}} + \sigma_{\text{pair}}$$

Cross-section is energy-dependent

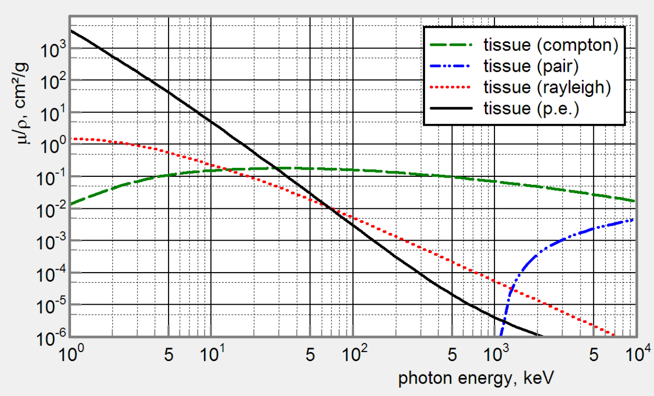


Figure: Soft tissue mass attenuation. Data and plot obtained using `XMuDat` [Nowotny, 1998]

Cross-section is energy-dependent

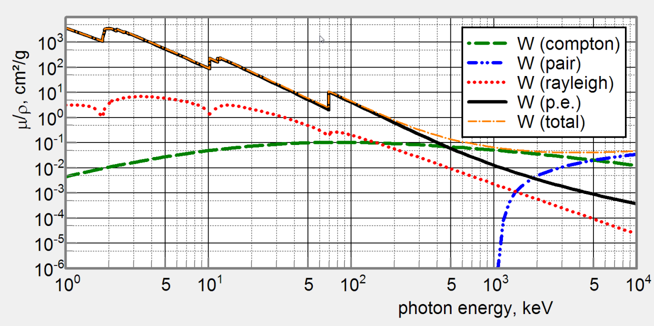


Figure: Tungsten mass attenuation. Data and plot obtained using `XMuDat` [Nowotny, 1998]

- Hard spheres collision is a wrong mental image.Think 'likelihood'.

X-ray tube

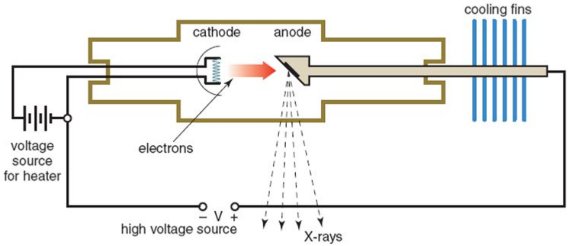


Figure: Principle of a Coolidge tube (X-ray tube)

Basics

- Generation of electrons (cathode)
- High voltage electron beam
- X-rays emission (anode)

Typical parameters

- Peak voltage V_{p}: 20–150 kV
- Tube current: 10–1000 mA
- Focal spot size: 1 μm—5 mm

X-ray tube

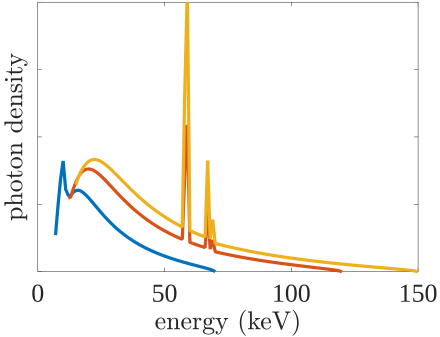
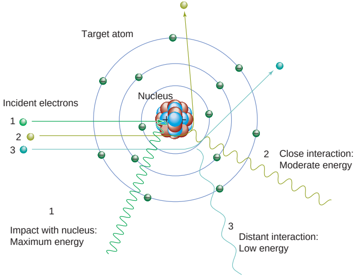
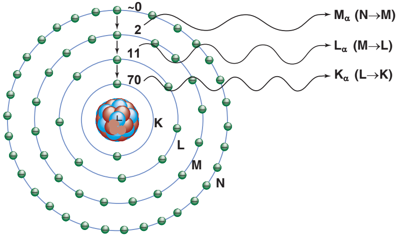


Figure: spectra from a tungsten anode x-ray tube operating for different peak voltages V_p (70, 120, and 150 kV). Data from [Poludniowski et al., 2009].

X-ray source





Adapted from [JT Bushberg, 2011, chap.6]

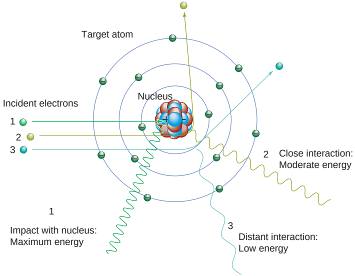
Bremsstrahlung

- Incident electrons deflected in the electric field of the nucleus
- Emission of a photon
- Continuous spectrum from $E_{\min}$ to $E_{\max} = eV_{\mathrm{p}}$

Characteristic X-rays

- Electron-matter interaction
- Electron vacancy results in energy levels transition

X-ray source



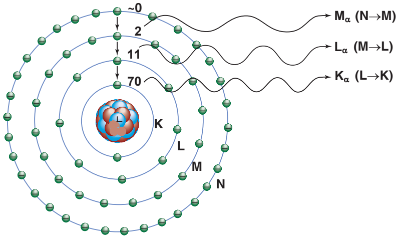


Figure: Tungsten energy-level diagram

Exercise

- Compute the characteristics X-ray emitted by a tungsten anode tube. Is it consistent with the plot on slide 107?

X-ray detector

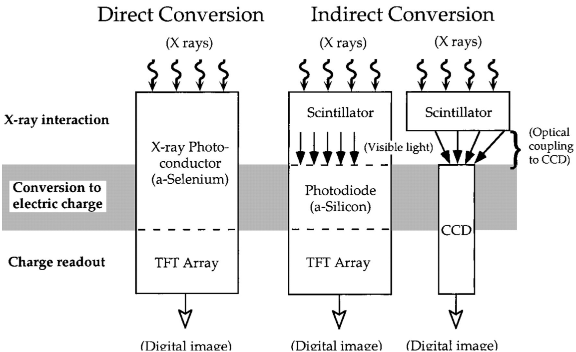


Figure: Direct vs indirect detection [Chotas et al., 1999]. TFT = Thin-film transistor, CCD = charge-coupled device.

X-ray detector

Indirect conversion

- X-rays converted into visible light by scintillators, e.g., CsI(TI)
- Visible photons converted into an electric charge by photodiode arrays

Direct conversion

- X-ray photons generate electron-hole pairs via photoelectric effect
- A bias voltage conducts electrons and holes to electrodes

X-ray detector

Hybrid pixel array detectors

- Direct conversion within semiconductor materials
- Bonded pixel-by-pixel to a readout application-specific integrated circuit (ASIC)
- Two mode of operations available: integration or counting

X-ray detector

Integrating detectors

- The incoming photo-generated current is integrated onto a feedback capacitor
- The measured signal is proportional to the attenuated X-ray *energy* [Huang et al., 2012]

$$s = \int_{E_{\min}}^{E_{\max}} q_{\mathrm{s}}(E) n_{\mathrm{a}}(E) E \, \mathrm{d}E$$

$$n_{\mathrm{a}}(E) = n_0(E)(1 - \exp[-\mu_{\mathrm{s}}(E) L_{\mathrm{s}}])$$

- $q_{\mathrm{s}}(E)$ includes the fractional energy absorbed in the scintillator, the scintillator conversion efficiency, and the optical coupling efficiency.

X-ray detector

Photon-counting detectors

- Absorption of an X-ray photon generate a current pulse
- One hit is recorded if the height pulse is above some (energy) threshold
- The measured signal is proportional to the attenuated X-ray *photon numbers*.

$$s = \int_{E_{\min}}^{E_{\max}} q(E) n_{\mathrm{a}}(E) \, \mathrm{d}E$$

- q includes amplification gains and energy distortion effects, e.g., fluorescence, share sharing, or pulse pile-up
- Energy discrimination capabilities based on pulse height analysis

Noise model

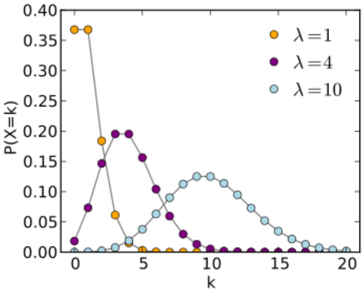
Basics

- Photon measurements are intrinsically corrupted by Poisson noise

$$\mathcal{P}(\lambda)$$

- For large photon numbers λ, we have

$$\mathcal{P}(\lambda) \approx \mathcal{N}(\mu = \lambda, \sigma = \sqrt{\lambda})$$



Noise model

Measured signal noise

- A complicated cascaded analysis is required to statistically model the conversion of incident X-ray photons into electrons (possibly via visible photons)
- An empirical model is often retained

$$\tilde{s} \sim \mathcal{N}(\mu = s, \sigma = \beta\sqrt{s})$$

where β depends on the detector

- The output signal is usually computed as

$$\log\left(\frac{s}{s^0}\right),$$

which makes things even more complicated...

Bibliographie II

Chotas, H. G., James T. Dobbins, I., and Ravin, C. E. (1999).
Principles of digital radiography with large-area, electronically readable detectors: A review of the basics.
Radiology, 210(3):595–599.
PMID: 10207454.

Huang, S.-Y., Yang, K., Abbey, C. K., and Boone, J. M. (2012).
A semiempirical linear model of indirect, flat-panel x-ray detectors.
Medical Physics, 39(4):2108–2118.

JT Bushberg, JA Seibert, E. L. (2011).
Essential Physics of Medical Imaging.
LIPPINCOTT RAVEN, third edition.

Nowotny, R. (1998).
Xmudat https://www-nds.iaea.org/publications/iaea-nds/iaea-nds-0195.htm.

Poludniowski, G., Landry, G., DeBlois, F., Evans, P. M., and Verhaegen, F. (2009).