Diagnostic Medical Image Processing
Prof. Dr.-Ing. Andreas Maier
Exercises (DMIP-E)
WS 2016/17

# Reconstruction in 3-D

Frank Schebesch, Tobias Würfl, Matthias Utzschneider,
Yixing Huang, Asmaa Khdeir, Houman Mirzaalian

Exercise Sheet 7

## 22 Analytic and Iterative Reconstruction

(i) The X-ray transform models the process of how CT raw data is acquired. In 3-D, is there a difference between this transform and the Radon transform? Include the respective integrals in your answer.
*Hint*: How and in which geometry are the integrals computed?

(ii) The algebraic reconstruction technique (ART) is a practical application of the Kaczmarc method. To get an idea how it works, assume that the following two equations describe two lines in the solution space of a synthetic CT projection:

$$3y - x = 5,$$
$$11x + 4y = 19.$$

Compute two iteration steps using ART to find an approximate solution $\mathbf{X}^{(2)} \in \mathbb{R}^2$. Initialize your algorithm with $\mathbf{X}^{(0)} = (0, 0)^\mathsf{T}$.
How good is this estimate? Compute the exact solution of the linear system and compare. Comment on the convergence rate for this specific example.

(iii) What is the main drawback of the elementary ART? Name and explain in a few words three different techniques of how we can tackle this problem.

**2+2+2**

(i) The projection geometry for the Radon transform is parallel, the X-ray transform for the common way of acquiring CT data is using cone-beams.
The 3-D Radon transform is defined with plane integrals, while we still acquire line integrals in a real system.

- The 3-D Radon transform gets the parallel plane-integral data

$$p_{\text{plane}}(s,\boldsymbol{\theta}) = \mathcal{R}f(s,\boldsymbol{\theta}) = \int_{\boldsymbol{x}\cdot\boldsymbol{\theta}=s} f(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

  i.e., the integral of the object $f$ over the hyperplane perpendicular to $\boldsymbol{\theta}$ with signed distance $s$ from the origin.
- The X-ray transform gets the line integral data

$$p_{\text{line}}(\boldsymbol{x_0},\boldsymbol{\theta}) = \mathcal{X}f(\boldsymbol{x_0},\boldsymbol{\theta}) = \int_{\boldsymbol{L}} f\mathrm{d}\boldsymbol{L} = \int_{\mathbb{R}} f(\boldsymbol{x_0}+t\boldsymbol{\theta})\mathrm{d}t,$$

  where $\boldsymbol{x_0}$ is an initial point on the line and $\boldsymbol{\theta}$ is a unit vector giving the direction of the line $\boldsymbol{L}$ (cf. Wikipedia).

(ii) The system matrix and the vector of projection values are

$$\mathbf{A} = \begin{pmatrix} -1 & 3 \\ 11 & 4 \end{pmatrix}, \qquad \mathbf{P} = \begin{pmatrix} 5 \\ 19 \end{pmatrix}.$$

From the Kaczmarc method we find iteratively (one way to do it):

$$\begin{aligned}
\mathbf{X}^{(0)} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\
\mathbf{X}^{(1)} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \frac{5-0}{(-1)^2+3^2}\begin{pmatrix} -1 \\ 3 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} -1 \\ 3 \end{pmatrix} \\
\mathbf{X}^{(2)} &= \frac{1}{2}\begin{pmatrix} -1 \\ 3 \end{pmatrix} + \frac{19-\frac{1}{2}(11,4)^{\mathsf{T}}\begin{pmatrix} -1 \\ 3 \end{pmatrix}}{11^2+4^2}\begin{pmatrix} 11 \\ 4 \end{pmatrix} \approx \begin{pmatrix} 0.985 \\ 2.040 \end{pmatrix}
\end{aligned}$$

The exact solution is $(1,2)^{\mathsf{T}}$.
Possible comment on the convergence: Convergence is quite fast here, which cannot be expected at all for ART. The reason can be seen, e.g., in the fact that both lines in the solution space are "almost orthogonal", i.e., the angle $\alpha$ between both rows of $\mathbf{A}$ is small:

$$\alpha = \arccos\frac{(-1,3)^{\mathsf{T}}\begin{pmatrix} 11 \\ 4 \end{pmatrix}}{\sqrt{1370}} \approx 0.03.$$

(iii) Slow convergence is the main drawback of ART. Possible improvements are:

- Compute update using more than one projected pixel, e.g., simultaneous ART (SART): multiple updates at the same time and combination of the result.
- Compute update using more than one projected pixel, e.g., simultaneous iterative reconstruction technique (SIRT): compute update once per iteration.
- Use intelligent methods to select the order of the update equations, e.g., ordered subsets.

## 23 Data Completeness

(i) What is Orlov's condition? What is Tuy's condition? Show for each condition a trajectory that meets its criterion.

(ii) Use Tuy's condition to explain under which condition the FDK algorithm performs an exact reconstruction.

(iii) In the lecture, we have taken a closer look at one specific algorithm that can be used for a helical trajectory. Which reconstruction algorithm was that? This algorithm used a concept called $\pi$-lines. Explain them in your own words and discuss the treatment of redundant data.

**2+1+1**

(i) Both are conditions for data completeness in 3-D. Examples for trajectories are in the lecture slides.

- Orlov's condition: A complete data set can be obtained if every great circle intersects the trajectory (represented on the unit sphere).
- Tuy's condition: Every plane that intersects the object of interest must contain a cone beam focal point.

(ii) If the object does not vary in z-direction, a single plane of the object does contain all necessary information. Now Tuy's condition demands that every plane through the object contains a source point. If a single plane contains enough information to reconstruct the object, we have an exact algorithm.

(iii) The following points could be addressed in the student's answer:

- Katsevich's algorithm
- $\pi$-line: For all points inside the helix there is one line that passes the point and hits the helix at two points that are separated by less than one pitch.
- If redundant data occurs, it occurs three times.
- Redundancy is solved by assigning the weights 1, -1, 1.

**Total: 10**