CLIC

Common Lines Implied Clustering

Donovan Webb

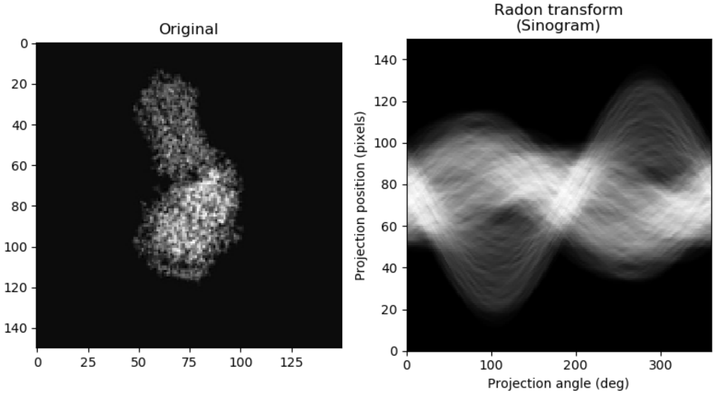
eBIC/University of Bath

March 23rd, 2020

Table of Contents

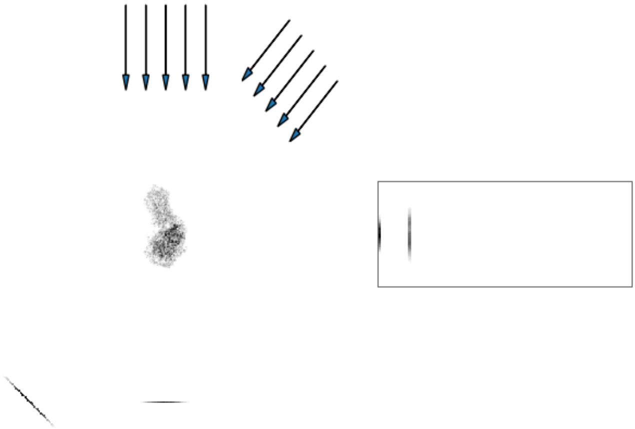
Single Lines

The Radon Transform

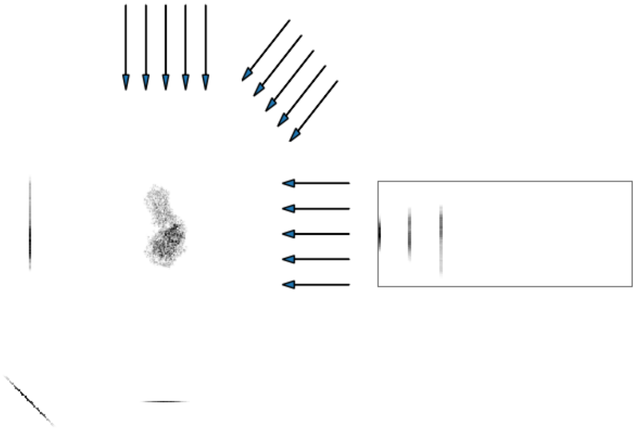


The Radon Transform

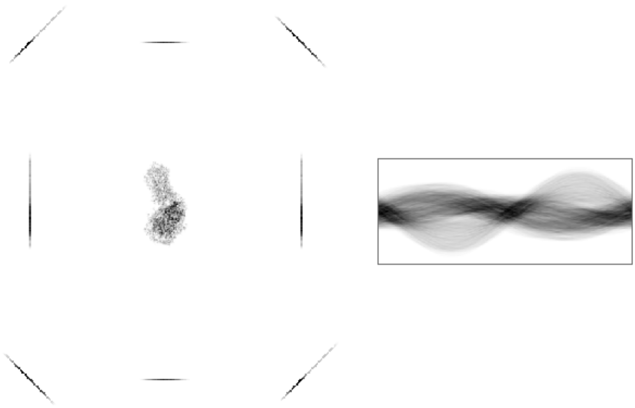
The Radon Transform



The Radon Transform

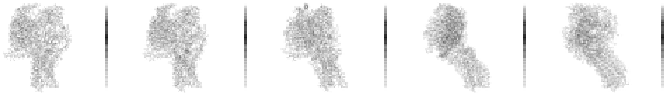


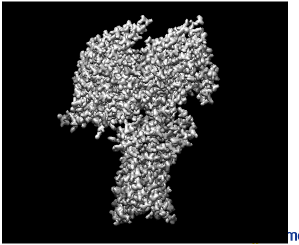
The Radon Transform



Common Lines

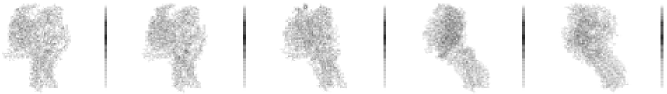
Two projections of the same 3D volume share at least one common line in the Radon transform

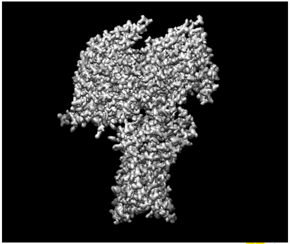




Common Lines

What about two different 3D volumes?

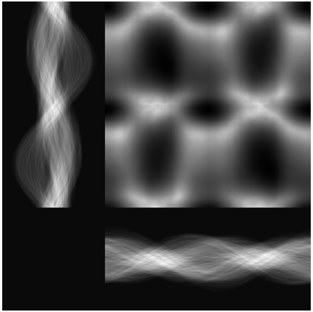




Finding Common Lines

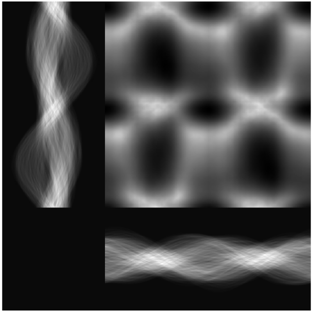
Sinogram Cross Correlation

Finding the common line between two sinograms



Sinogram Cross Correlation

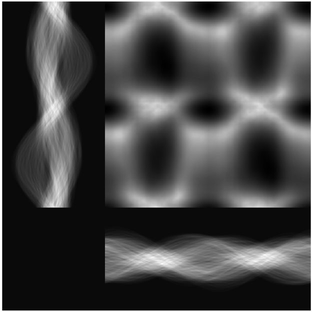
Finding the common line between two sinograms



But what about N sinograms?

Sinogram Cross Correlation

Finding the common line between two sinograms



But what about N sinograms?
What about N sinograms from a heterogenous dataset?

Pipeline

- Plot each line in high dimensional space
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms → Common line!
- Best scoring common lines → From the same model!

Pipeline

- Plot each line in high dimensional space
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms → Common line!
- Best scoring common lines → From the same model!

Slow.

Pipeline

- Plot each line in high dimensional space
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms → Common line!
- Best scoring common lines → From the same model!

Slow. Exhaustive.

Pipeline

- Plot each line in high dimensional space
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms $\rightarrow$ Common line!
- Best scoring common lines $\rightarrow$ From the same model!

Slow. Exhaustive. Doesn't handle noise well.

Dimensional Reduction

Find features - Reduce noise

Linear

PCA

Non-Linear

LLE

Isomap

TSNE

UMAP

Pipeline

- Plot each line in high dimensional space
- Apply dimensional reduction
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms $\rightarrow$ Common line!
- Best scoring common lines $\rightarrow$ From the same model!

Pipeline

- Plot each line in high dimensional space
- Apply dimensional reduction
- Find Euclidean Distances
- Smaller distances mean better agreement
- Best match between two sinograms → Common line!
- Best scoring common lines → From the same model!

But how do we assign clusters?

Clustering

Can heterogenaity be sorted by looking at common lines?

Ground truth: Good seperatation between two classes - but discontinuous

Heirarchical clustering

Pipeline

- Plot each line in high dimensional space
- Apply dimensional reduction
- Find Euclidean Distances
- Smaller distances mean better agreement - Create score
- Heirarchical Clustering
- Cut tree to produce clusters

Pipeline

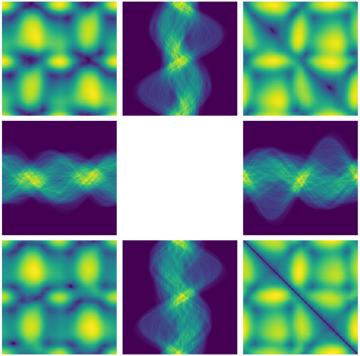
- Plot each line in high dimensional space
- Apply dimensional reduction
- Find Euclidean Distances
- Smaller distances mean better agreement - Create score
- Heirarchical Clustering
- Cut tree to produce clusters

Just a model left!

Reconstruction

Angular recovery from 3 Common lines

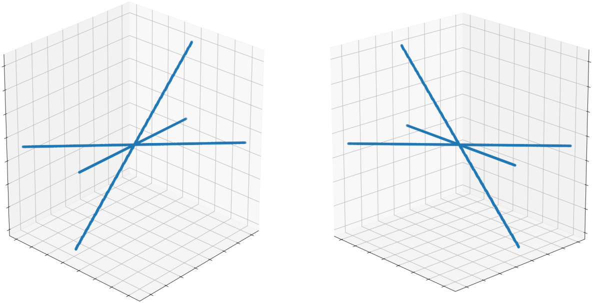
Common line gives axis of rotation. Three common lines gives 2 unique solutions for 3D orientation (One mirror of other)



Angular Reconstitution: A Posteriori Assignment of Projection Directions for 3d Reconstruction. Van Heel 1987

Angular recovery from 3 Common lines

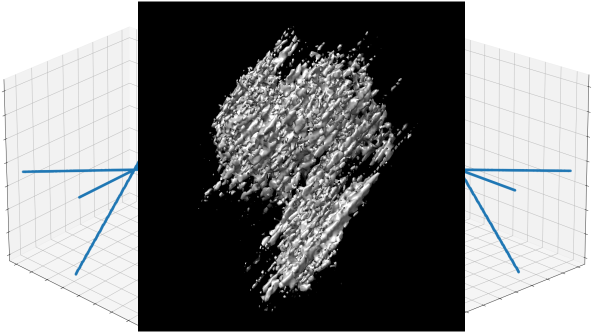
Common line gives axis of rotation. Three common lines gives 2 unique solutions for 3D orientation (One mirror of other)



Angular Reconstitution: A Posteriori Assignment of Projection Directions for 3d Reconstruction. Van Heel 1987

Angular recovery from 3 Common lines

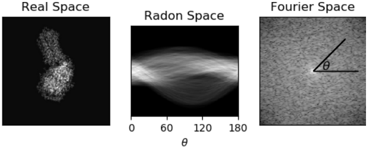
Common line gives axis of rotation. Three common lines gives 2 unique solutions for 3D orientation (One mirror of other)



Angular Reconstitution: A Posteriori Assignment of Projection Directions for 3d Reconstruction. Van Heel 1987

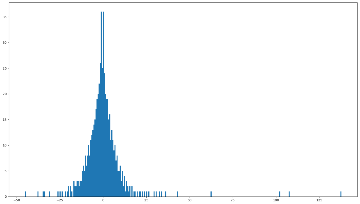
Eigenvector Relaxation

Aim: Given all common lines c for projections P, assign Rotation matrices R for each P to give greatest consensus volume.



$$max \sum_{i \neq j} R_i c_{ij} \cdot R_j c_{ji} \quad (2)$$

$c_{ij} = (cos$



Maths*! Make large ($2N \times 2N$) symmetric matrix S. Can recover R for each P from top 3 eigenvectors of S that maximise (2)!

*Three Dimensional Structure Determination from Common Lines in Cryo-EM by Eigenvectors and Semidefinite Programming, A.Singer and Y.Shkolnisky

Full Pipeline

A full pipeline of the procedure. 2d projs ¿ 2d sins ¿ 1d lines ¿ TSNE ¿ agglo ¿ clusters ¿ split into sep datasets ¿ find common lines ¿ eigenvector relaxation ¿ Models