Fast and accurate registration of n point clouds

Background

Want a way to combine pairwise registrations that:

- Is fast and straightforward

Background

Want a way to combine pairwise registrations that:

- Is fast and straightforward
- Allows us flexibility to encode our confidence (or lack thereof) in particular scans or registrations

Background

Want a way to combine pairwise registrations that:

- Is fast and straightforward
- Allows us flexibility to encode our confidence (or lack thereof) in particular scans or registrations
- Is robust to missing pairwise registrations

Registration

An alignment between two scans, often a rotation and translation represented as a transformation matrix in $\mathbb{R}^{4\times 4}$.

I've just been working with *translations* without rotations, since that's what Lawrence's NCC method gives.

Registration

An alignment between two scans, often a rotation and translation represented as a transformation matrix in $\mathbb{R}^{4\times4}$.

I've just been working with *translations* without rotations, since that's what Lawrence's NCC method gives.

Let $\bar{D}_{ij} : \mathbb{R}^3 \to \mathbb{R}^3$ represent the transformation to scan i from scan j via application of a transformation matrix.

$$\mathbf{p}_i = \bar{D}_{ij}\mathbf{p}_j$$

Registration uncertainty

We expect some registrations to be better than others. Need to be able to encode this.

We measure how unsure we are about a registration with a *covariance matrix*.

Registration uncertainty

We expect some registrations to be better than others. Need to be able to encode this.

We measure how unsure we are about a registration with a *covariance matrix*.

To encode that we think we're about 0.1 m off in every direction, we set covariance to $0.1^2 I$.

Putting scans together

Say that for any registration between scans i and j we have:

- $C_{ij} \in \mathbb{R}^{3\times 3}$: Covariance
- $\bar{D}_{ij} : \mathbb{R}^3 \to \mathbb{R}^3$: Transform to scan i from scan j (as before)

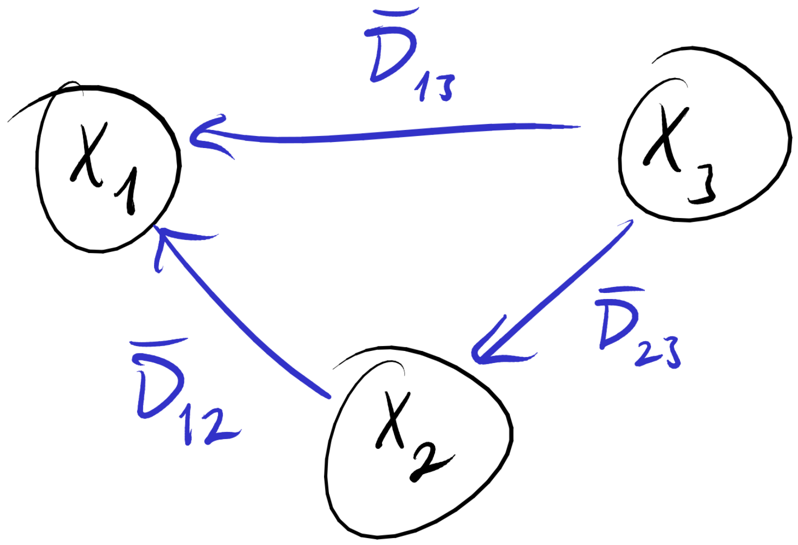
Putting scans together

Say that for any registration between scans i and j we have:

- $C_{ij} \in \mathbb{R}^{3\times3}$: Covariance
- $\bar{D}_{ij} : \mathbb{R}^3 \to \mathbb{R}^3$: Transform to scan i from scan j (as before)

Let $X_i \in \mathbb{R}^3$ be where scan i "belongs", or alternatively, its optimal global registration.

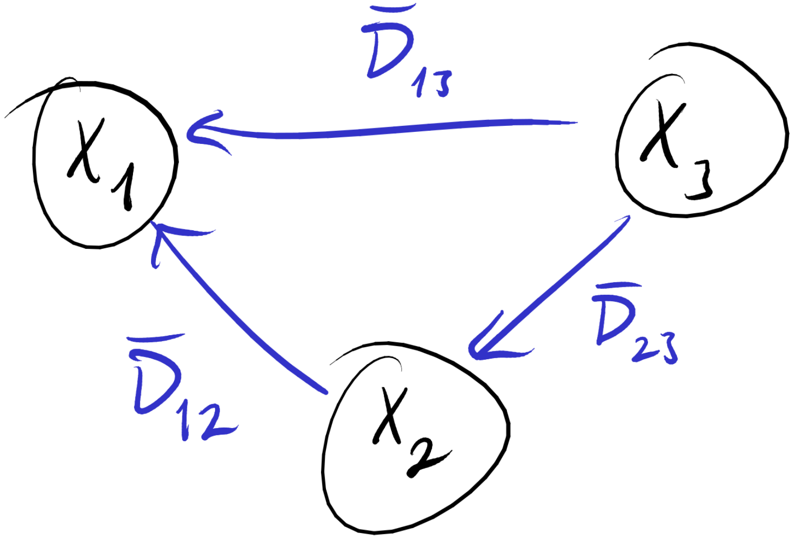
Putting scans together



We hope

$$\bar{D}_{13} \approx \bar{D}_{12}\bar{D}_{23}.$$

Putting scans together



In practice,

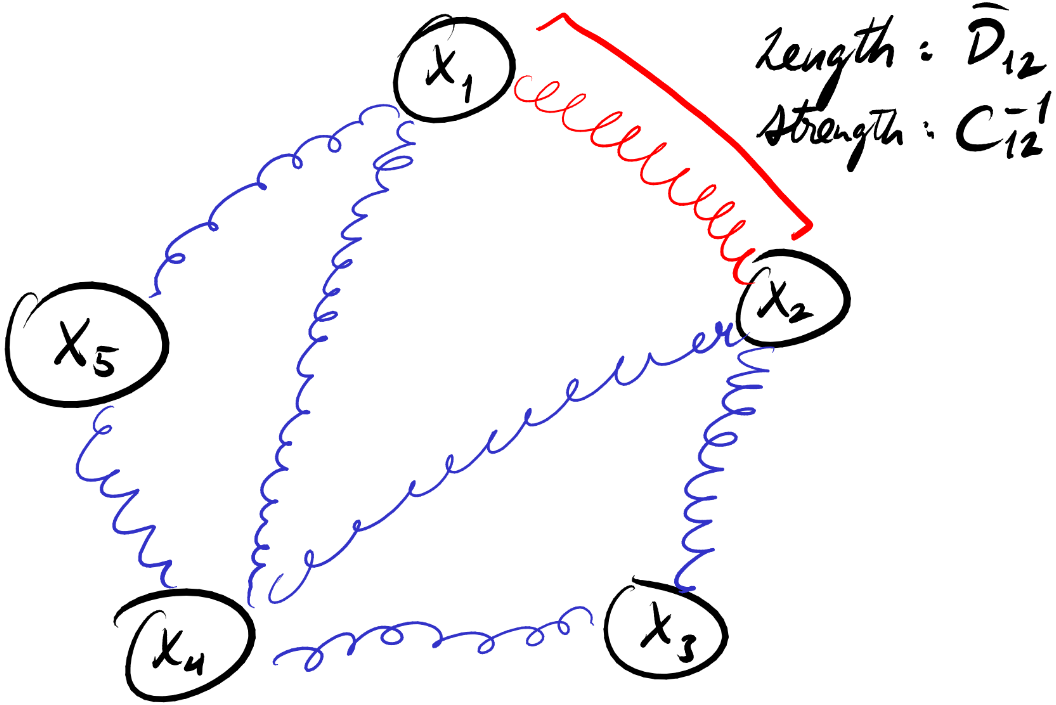
$$\bar{D}_{13} \neq \bar{D}_{12}\bar{D}_{23}.$$

Putting scans together

Lu and Milios give a way to determine an estimate for $X_1, \ldots, X_n$.

Putting scans together

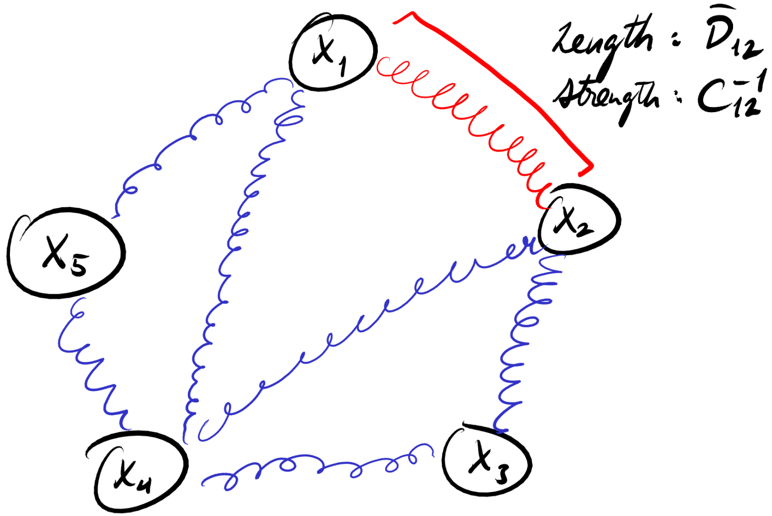
A set of pairwise registrations is like a web of springs.



Putting scans together

A set of pairwise registrations is like a web of springs.

$$\underset{\{X_1,\ldots,X_n\}}{\text{minimize}} \quad \sum_{i,j} \left(\bar{D}_{ij} - (X_i - X_j)\right)^T C_{ij}^{-1} \left(\bar{D}_{ij} - (X_i - X_j)\right)$$



Putting scans together

A set of pairwise registrations is like a web of springs.

$$\underset{\{X_1,\dots,X_n\}}{\text{minimize}} \quad \sum_{i,j} \left(\bar{D}_{ij} - \left(X_i - X_j\right)\right)^T C_{ij}^{-1} \left(\bar{D}_{ij} - \left(X_i - X_j\right)\right)$$

Easy to solve.

Putting scans together

A set of pairwise registrations is like a web of springs.

$$\underset{\{X_1,\ldots,X_n\}}{\text{minimize}} \quad \sum_{i,j} \left(\bar{D}_{ij} - (X_i - X_j)\right)^T C_{ij}^{-1} \left(\bar{D}_{ij} - (X_i - X_j)\right)$$

Easy to solve.

How do we combine pairwise registrations to get n globally optimal registrations?

Putting scans together

A set of pairwise registrations is like a web of springs.

$$\underset{\{X_1,\dots,X_n\}}{\text{minimize}} \quad \sum_{i,j} \left(\bar{D}_{ij} - (X_i - X_j)\right)^T C_{ij}^{-1} \left(\bar{D}_{ij} - (X_i - X_j)\right)$$

Easy to solve.

How do we combine pairwise registrations to get n globally optimal registrations?

1. Determine how uncertain each pairwise registration is
2. Make a “web of springs” with the pairwise registrations
3. Find node positions that minimize springs’ energy

Comparison to previous methods

- Here we are using pose graph optimization to *combine pairwise registrations.*
  - Graph nodes are scan positions inferred from arbitrary pairwise registrations
  - Graph edges are pairwise registrations with uncertainty parameter

Comparison to previous methods

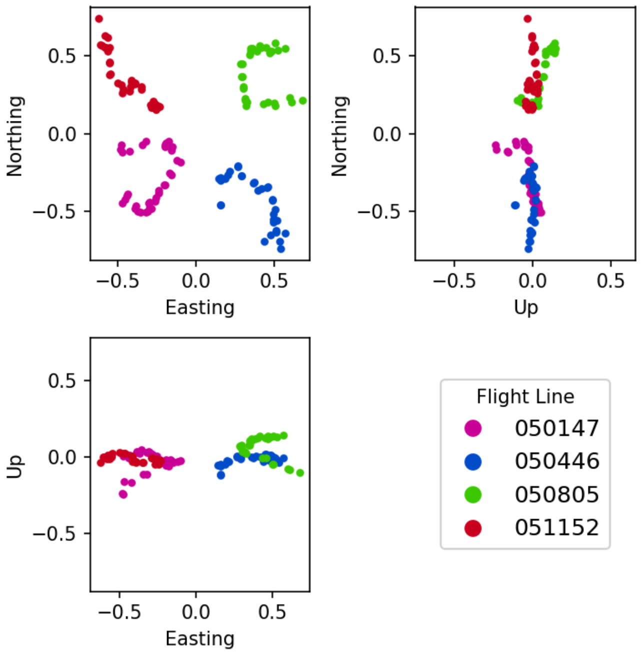
- Here we are using pose graph optimization to *combine pairwise registrations.*
  - Graph nodes are scan positions inferred from arbitrary pairwise registrations
  - Graph edges are pairwise registrations with uncertainty parameter
- MZICP used pose graph optimization *on the points in the scans* to create pairwise registrations and combine them.
  - Graph nodes were scan positions inferred from point correspondences
  - Graph edges were pairwise registrations inferred from point correspondences, with uncertainties inferred using a pointwise algorithm by Lu and Milios

Determining covariances (uncertainty) of pairwise registrations

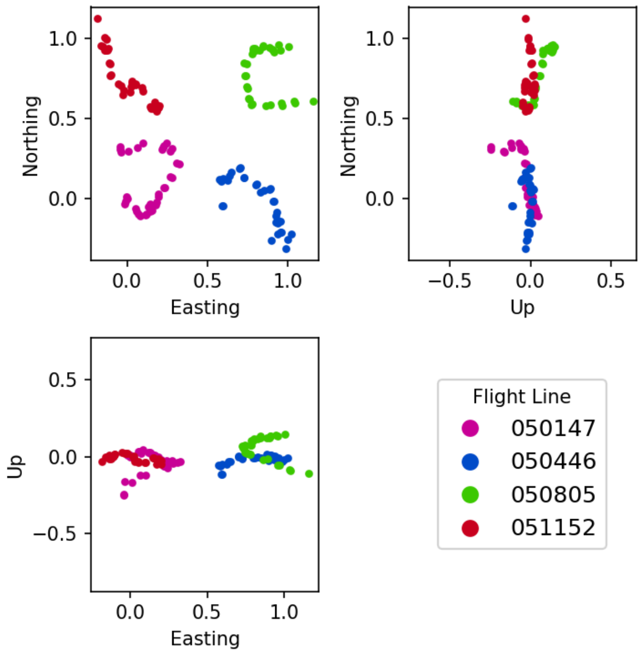
Difficult, but of utmost importance.

Constant covariance works much like Lawrence’s methods

Barrett Park, pairwise registrations from `20250527_ncc`



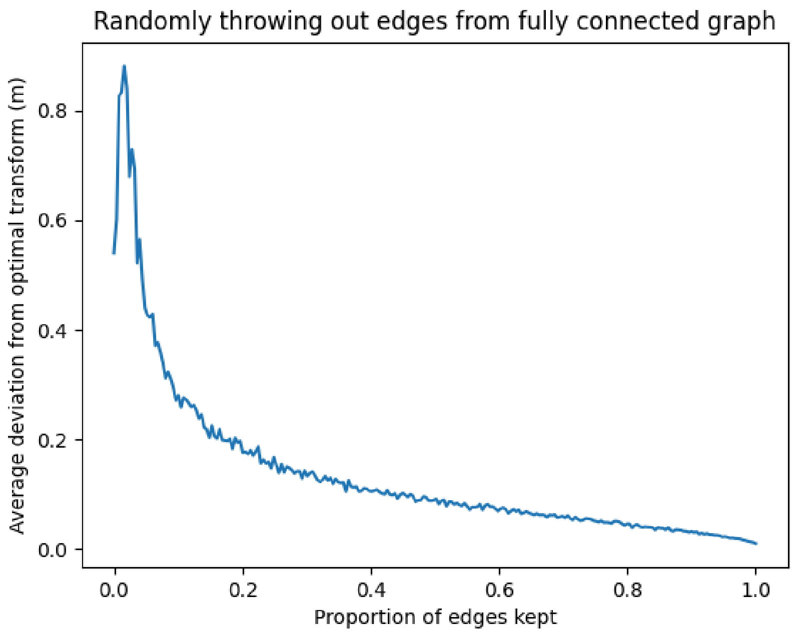
Pose graph, constant covariance



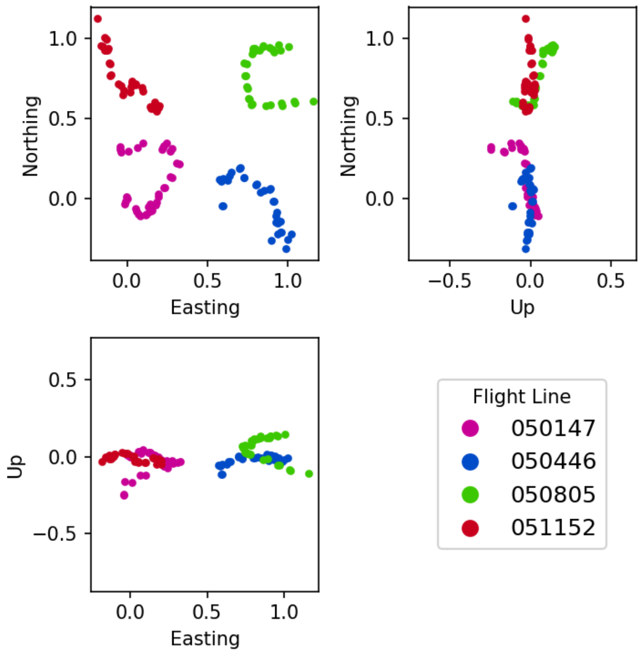
Jacob Lawrence’s `ncc_nxn.py`

Constant covariance works much like Lawrence's methods

Barrett Park, pairwise registrations from `20250527_ncc`



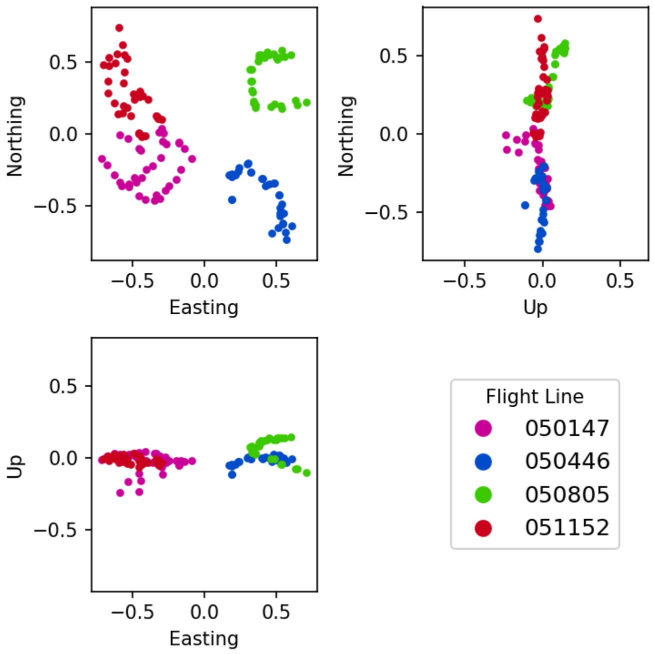
Pose graph, constant covariance



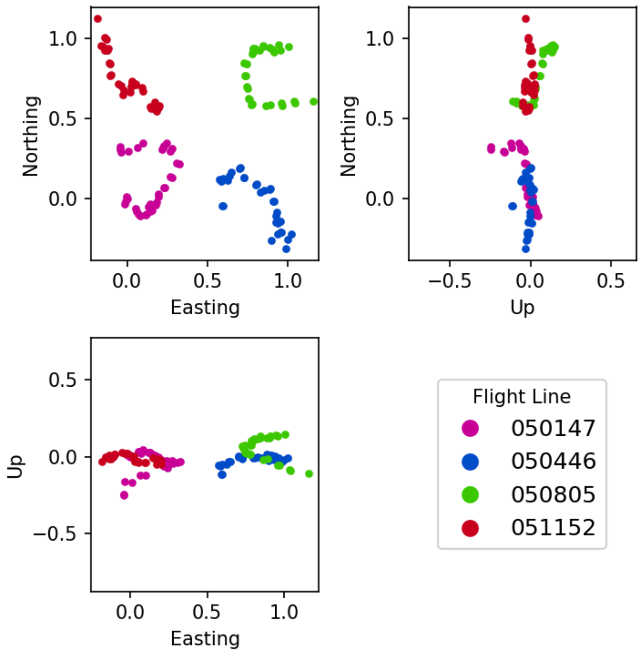
Jacob Lawrence's `ncc_nxn.py`
("Optimal transform")

Constant covariance works much like Lawrence’s methods

Barrett Park, pairwise registrations from `20250527_ncc`



Using pose graph, constant covariance, randomly keeping 75% of registrations.



Jacob Lawrence’s `ncc_nxn.py`

Modeling covariance manually

Pre-registration features

- Scan area overlap
- Difference between scan angles
- Time between scans

Post-registration features

- Pointwise scan overlap
- Normalized cross-correlation

Modeling covariance manually

Pre-registration features

- Scan area overlap
- Difference between scan angles
- Time between scans

Post-registration features

- Pointwise scan overlap
- Normalized cross-correlation

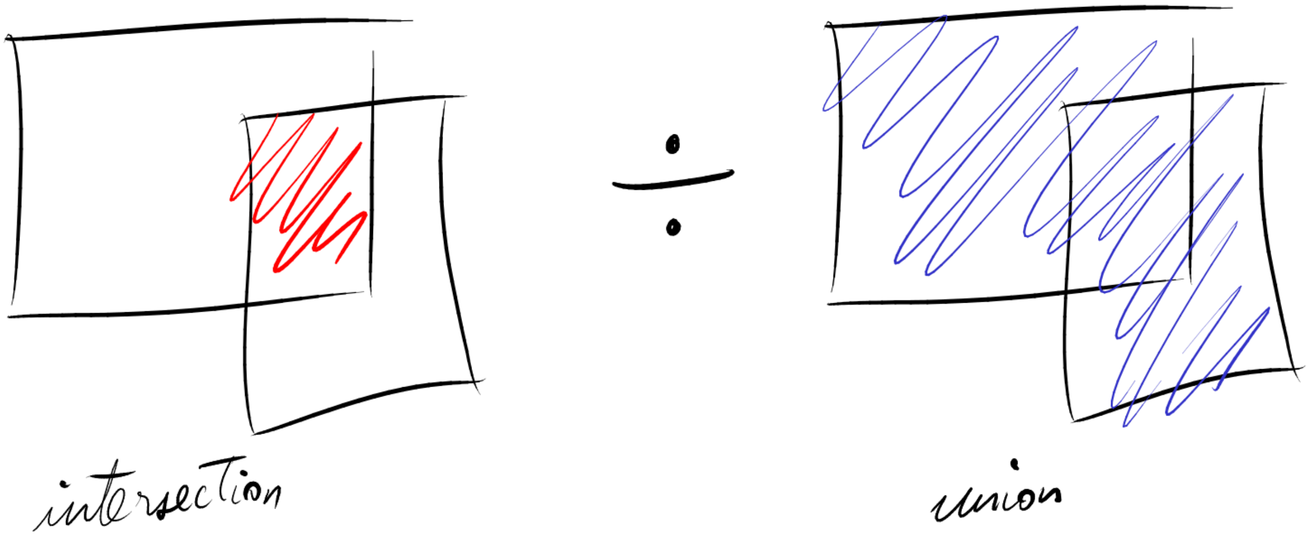
Modeling covariance manually

Pre-registration features	Post-registration features
• Scan area overlap • Difference between scan angles • Time between scans	• Pointwise scan overlap • Normalized cross-correlation

We tend to believe a pairwise registration less if:

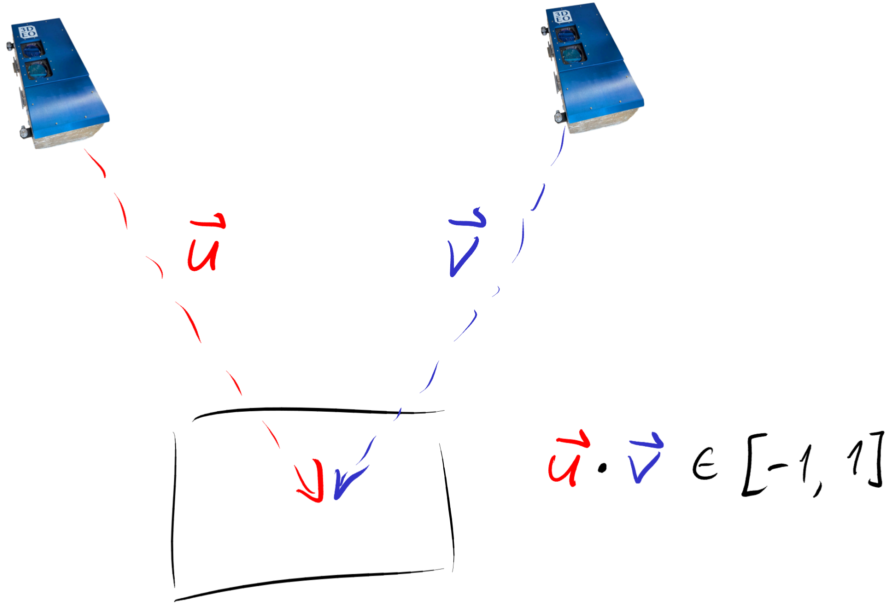
- Scan area overlap is low
- Scan angles are very different

Modeling covariance manually



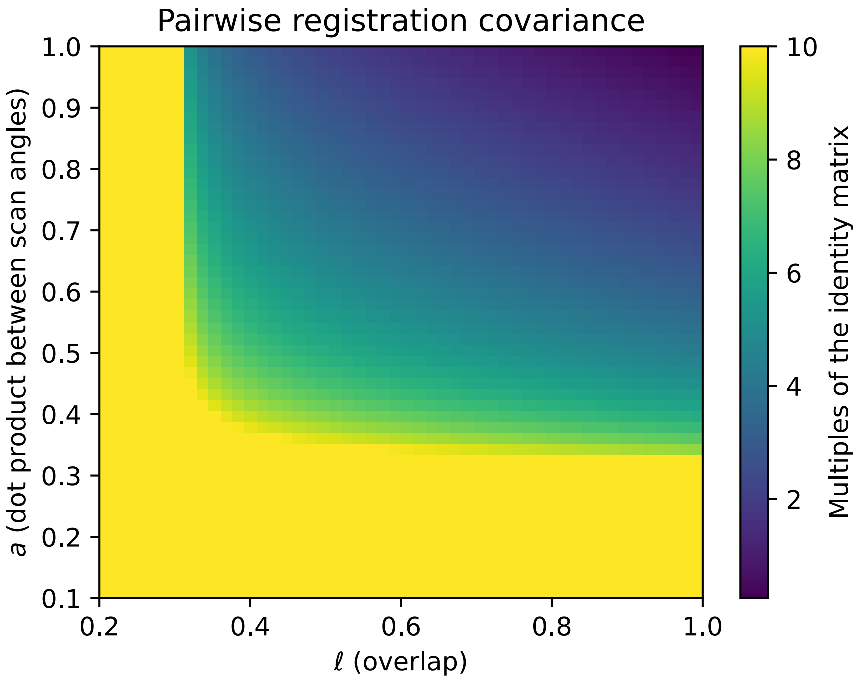
Measure scan area overlap with intersection over union

Modeling covariance manually



Measure scan angle difference with dot product between scan angles

Modeling covariance manually



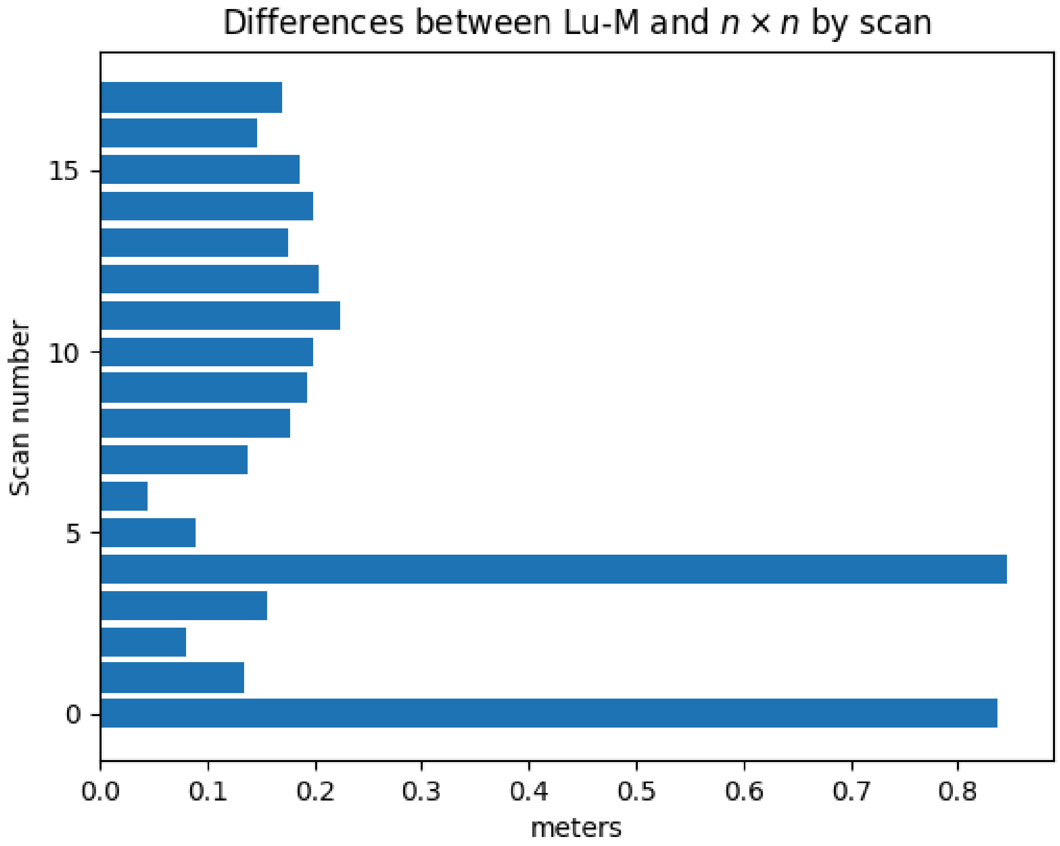
Covariance C_{ij} is some base covariance plus extra if overlap and/or angles are bad.

Modeling covariance manually

Not totally successful. Tended to perform worse than constant covariance.

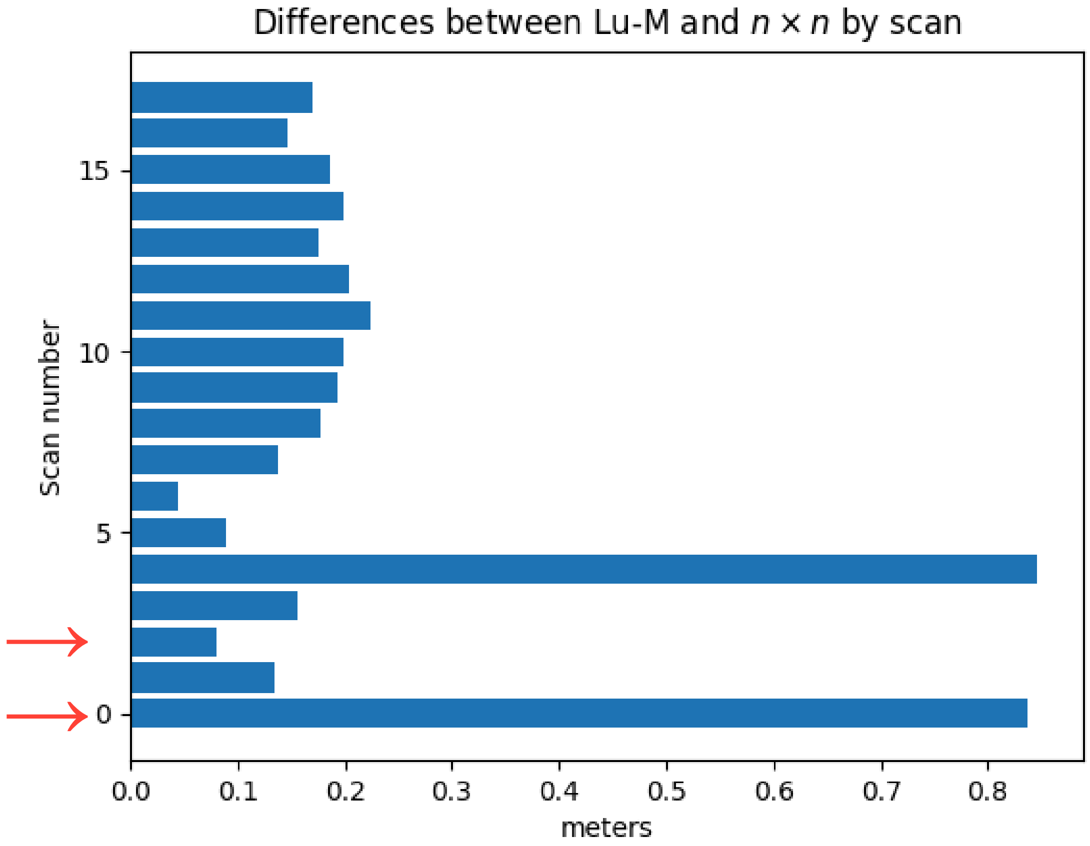
Modeling covariance manually

NTH001_030



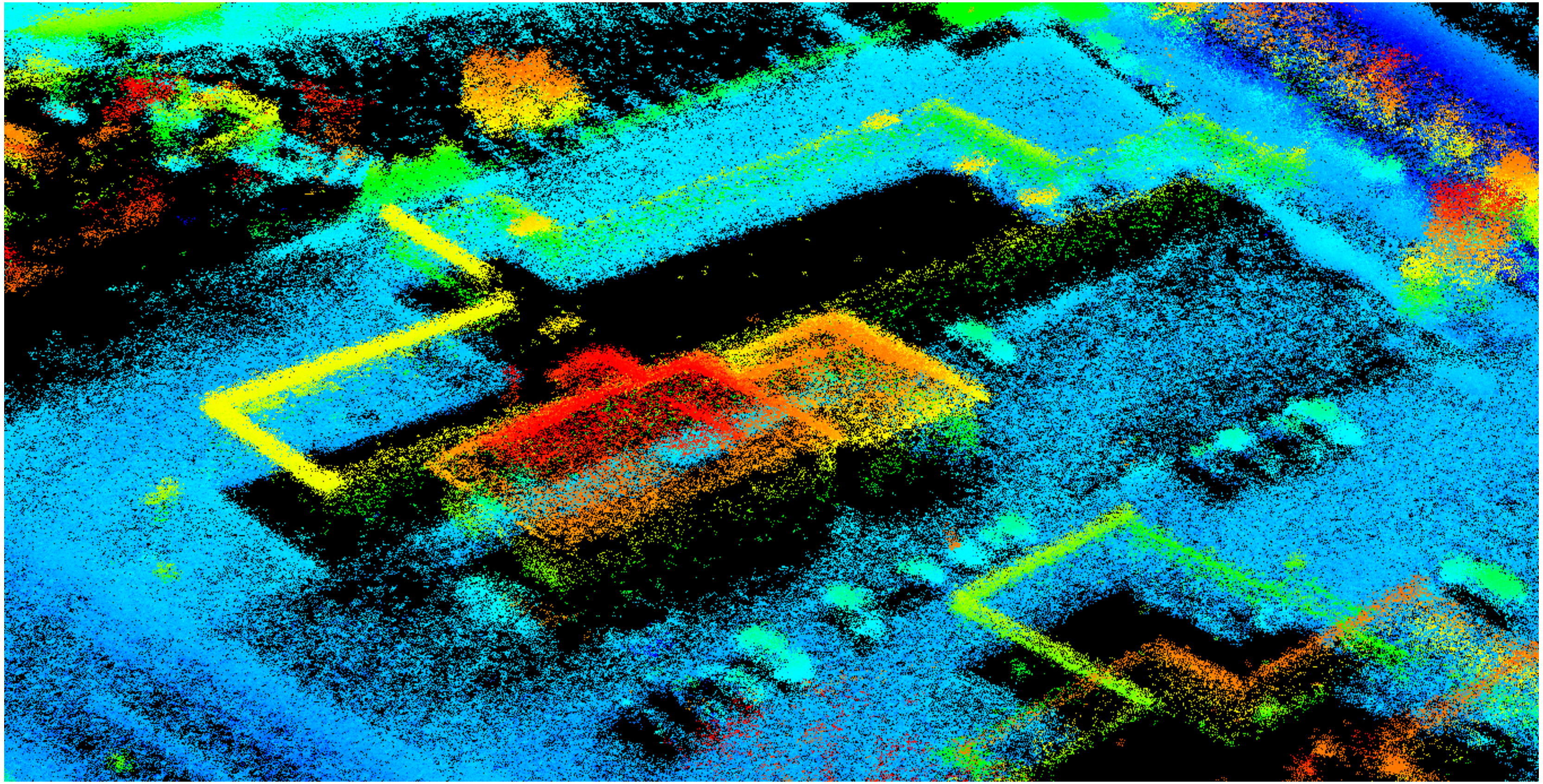
Modeling covariance manually

NTH001_030



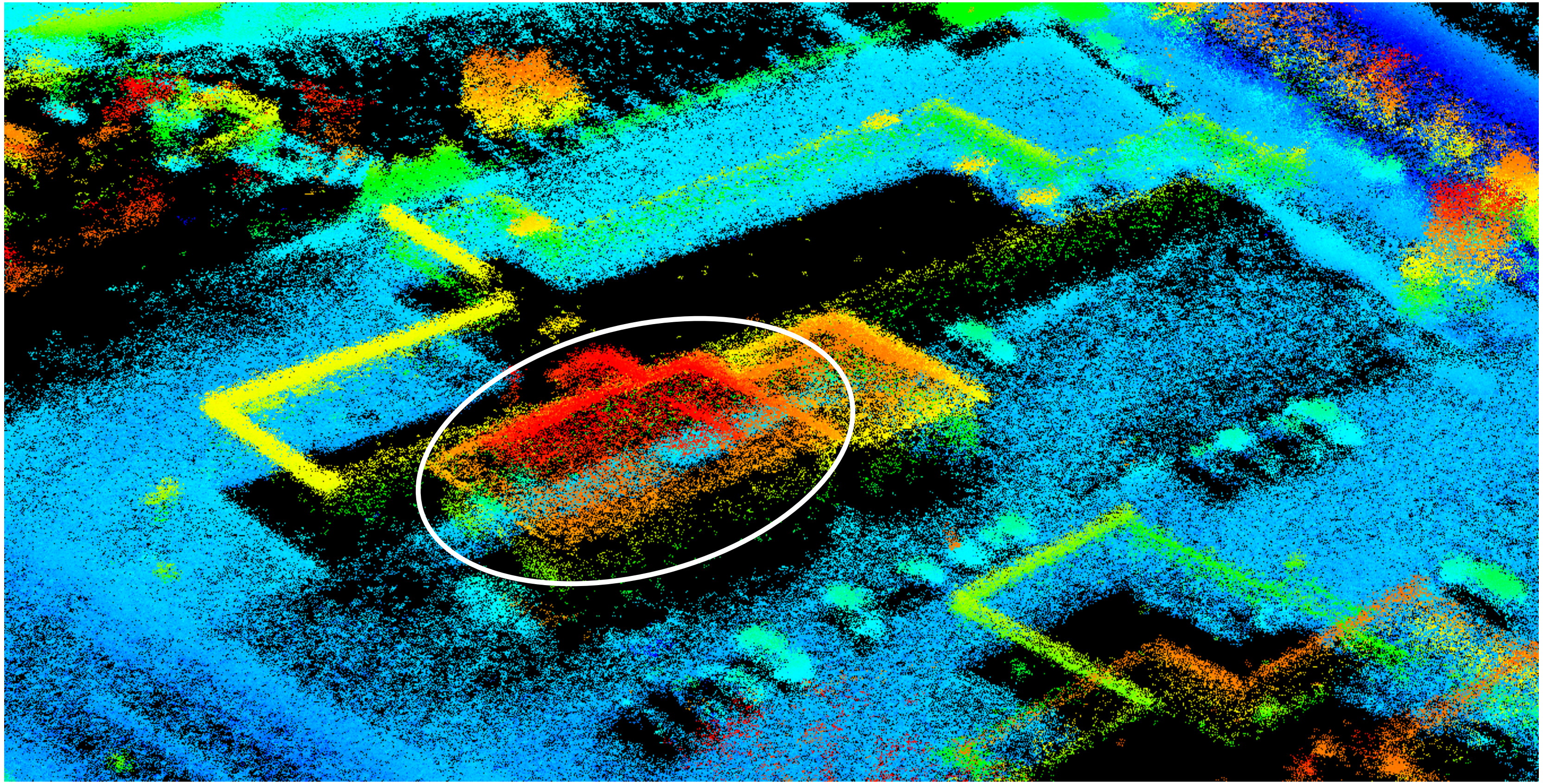
Modeling covariance manually

NTH001_030



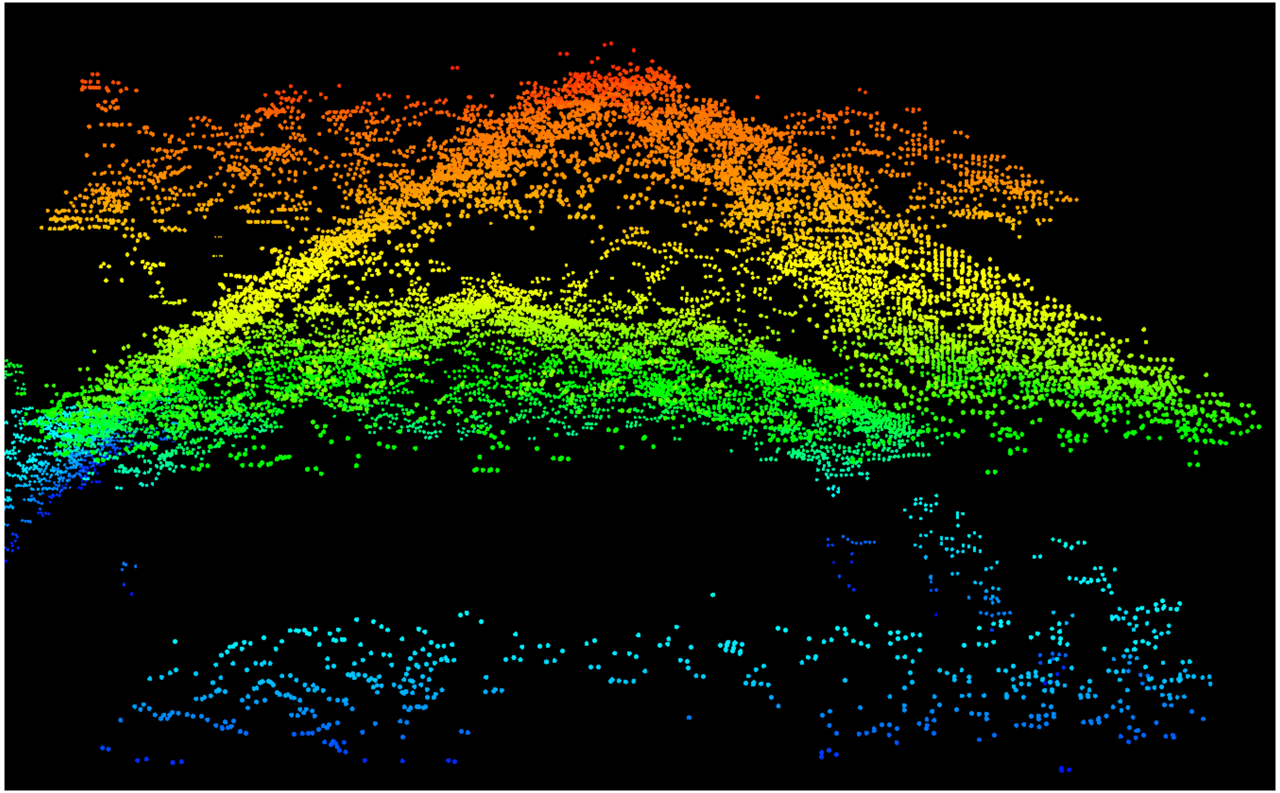
Modeling covariance manually

NTH001_030



Modeling covariance manually

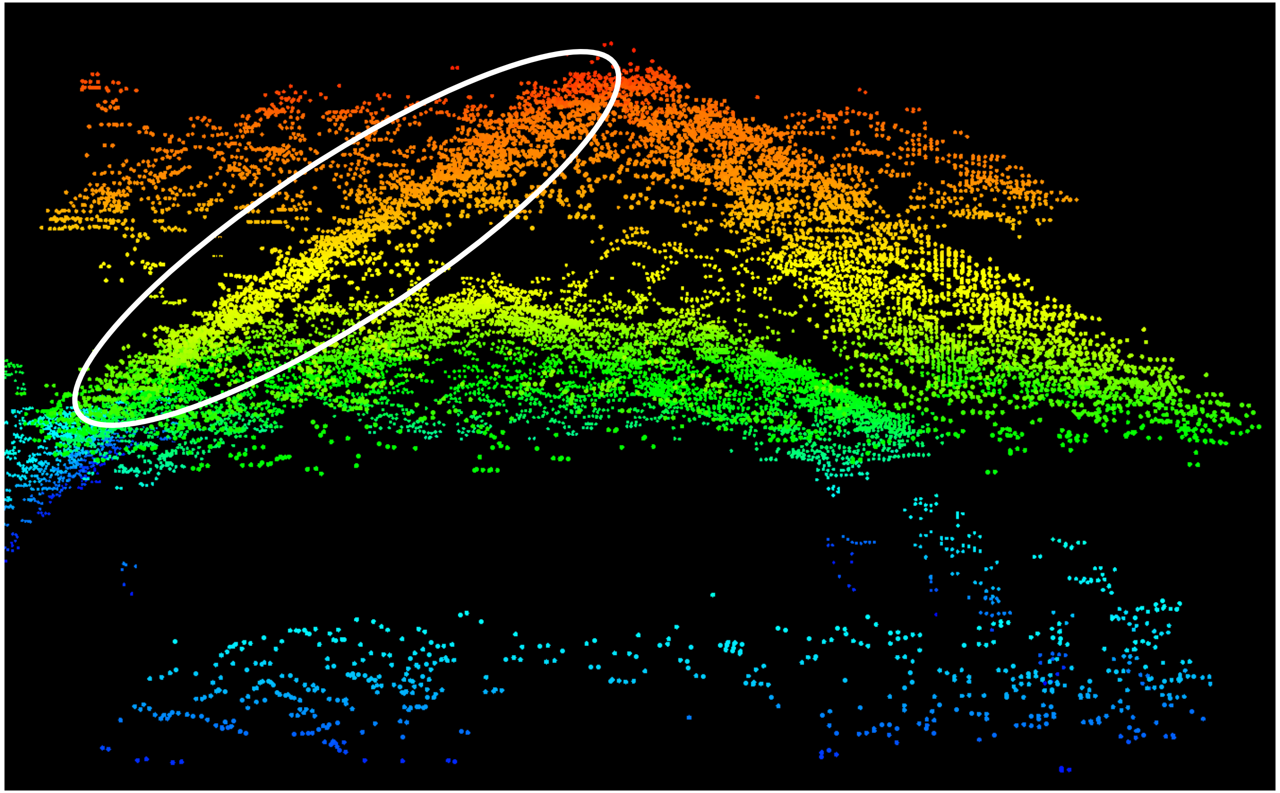
NTH001_030



Lawrence's optimization. Good

Modeling covariance manually

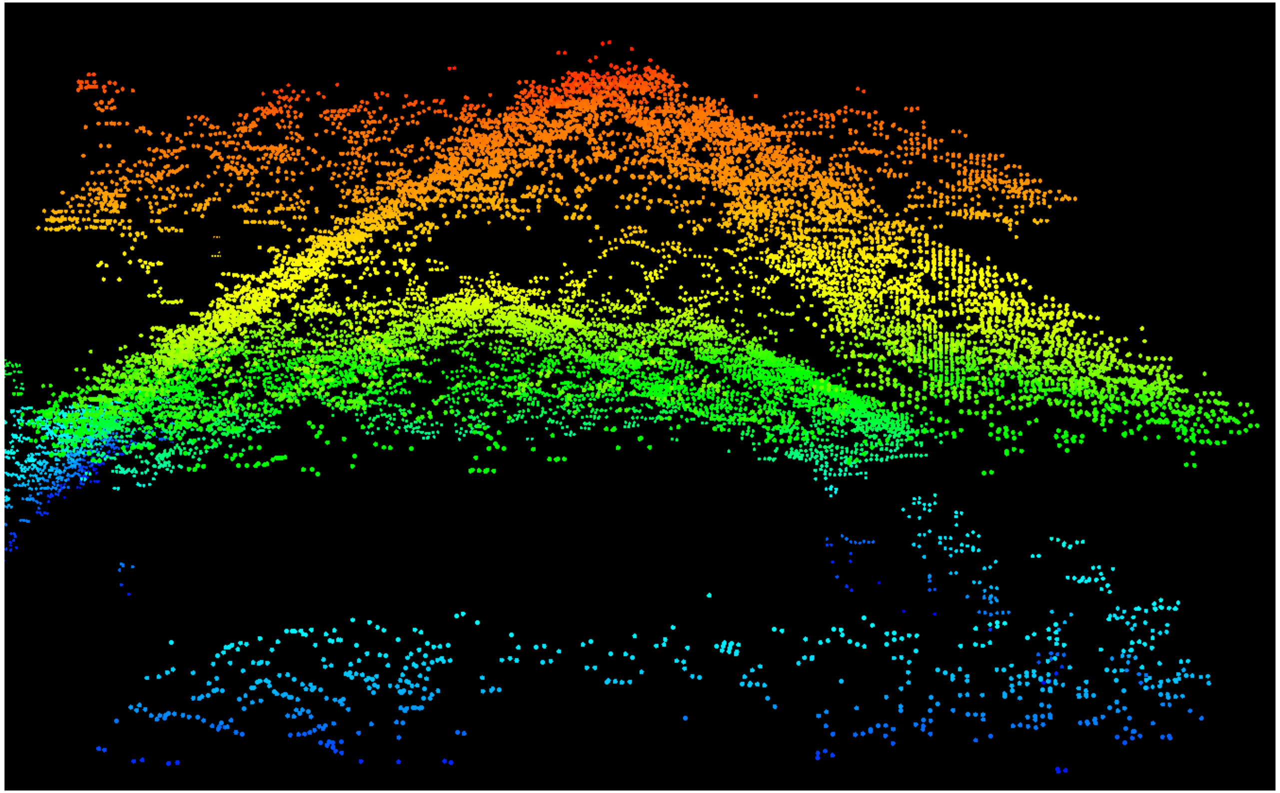
NTH001_030



Lawrence's optimization. Good

Modeling covariance manually

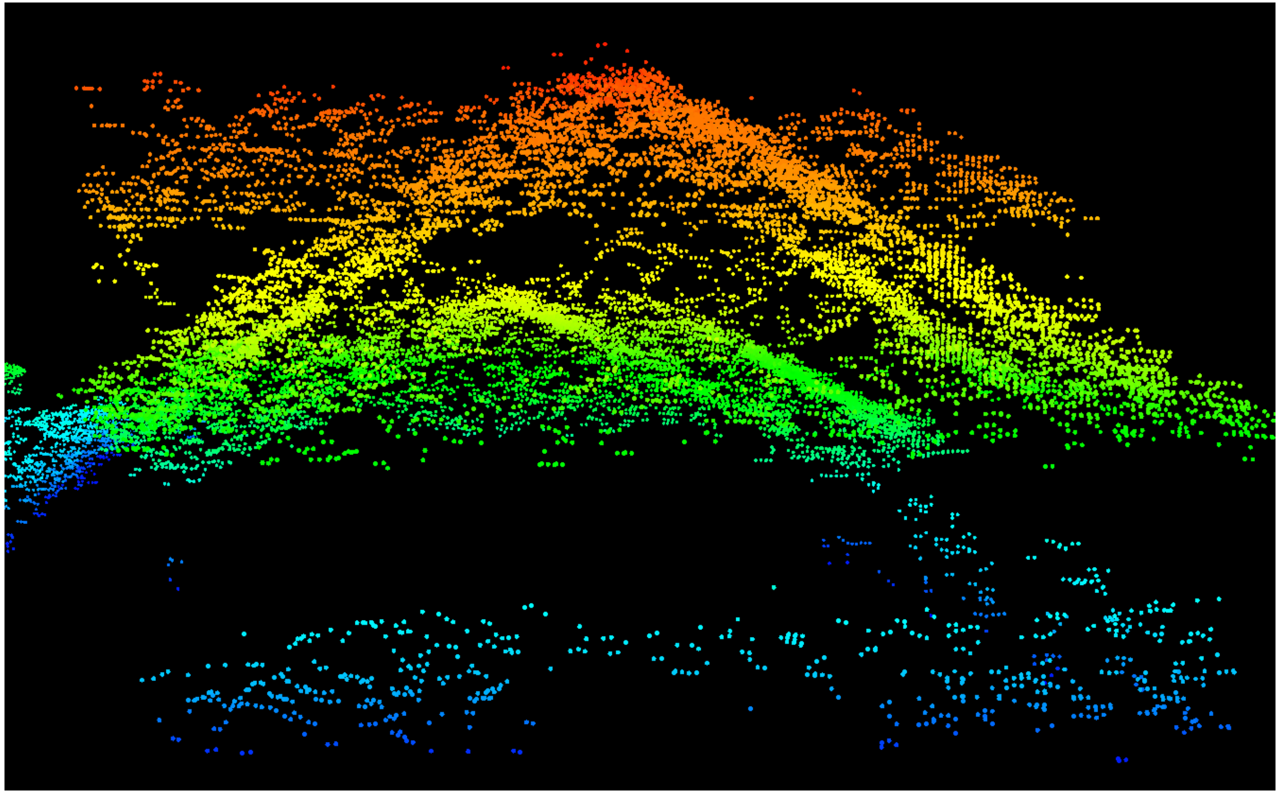
NTH001_030



Constant covariance pose graph. Good

Modeling covariance manually

NTH001_030



Modeled covariance pose graph. Bad

Optimizing model parameters to some metric

If we have a decent registration metric,

$$\text{Accurate covariance model} \implies \text{Accurate global registration} \implies \text{Metric locally optimized}$$

Optimizing model parameters to some metric

If we have a decent registration metric,

$$\text{Accurate covariance model} \implies \text{Accurate global registration} \implies \text{Metric locally optimized}$$

? ? ?

$$\text{Metric optimized} \implies \text{Accurate global registration} \implies \text{Accurate covariance model}$$

If so, this should work.

Optimizing model parameters to some metric

If we have a decent registration metric,

$$\text{Accurate covariance model} \implies \text{Accurate global registration} \implies \text{Metric locally optimized}$$

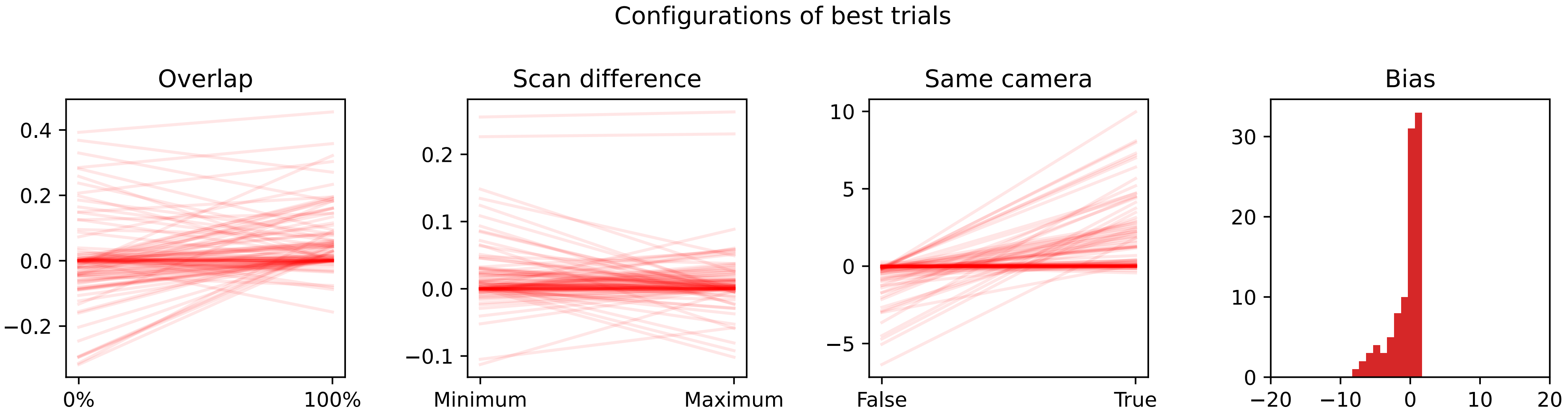
$$\overset{???}{\text{Metric optimized} \implies \text{Accurate global registration} \implies \text{Accurate covariance model}}$$

If so, this should work.

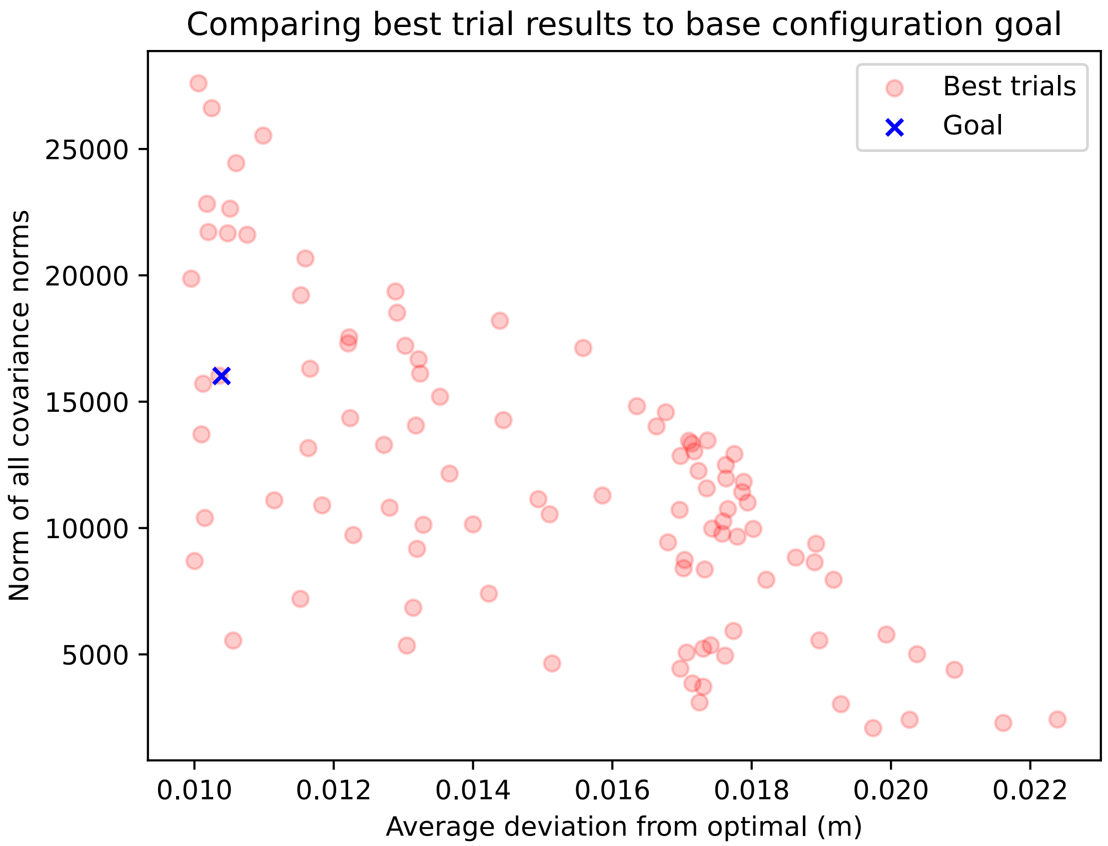
Choosing a metric is hard. I used a weighted combination of Lawrence's `ncc_nxn.py` and graph sparsity to test.

Optimizing model parameters to some metric

(A mix of pre-registration features and post-registration features)

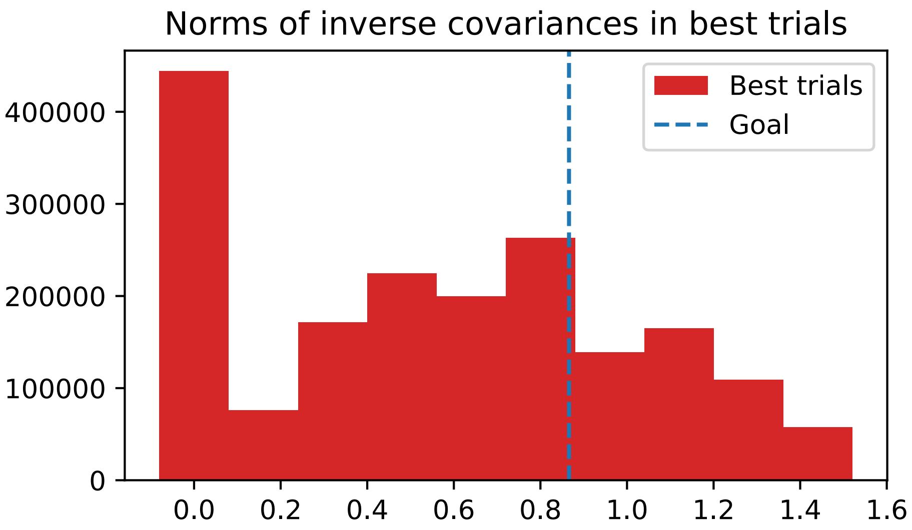


Optimizing model parameters to some metric



Optimizing model parameters to some metric

Sparse! We can avoid lots of registrations



Optimizing model parameters to some metric

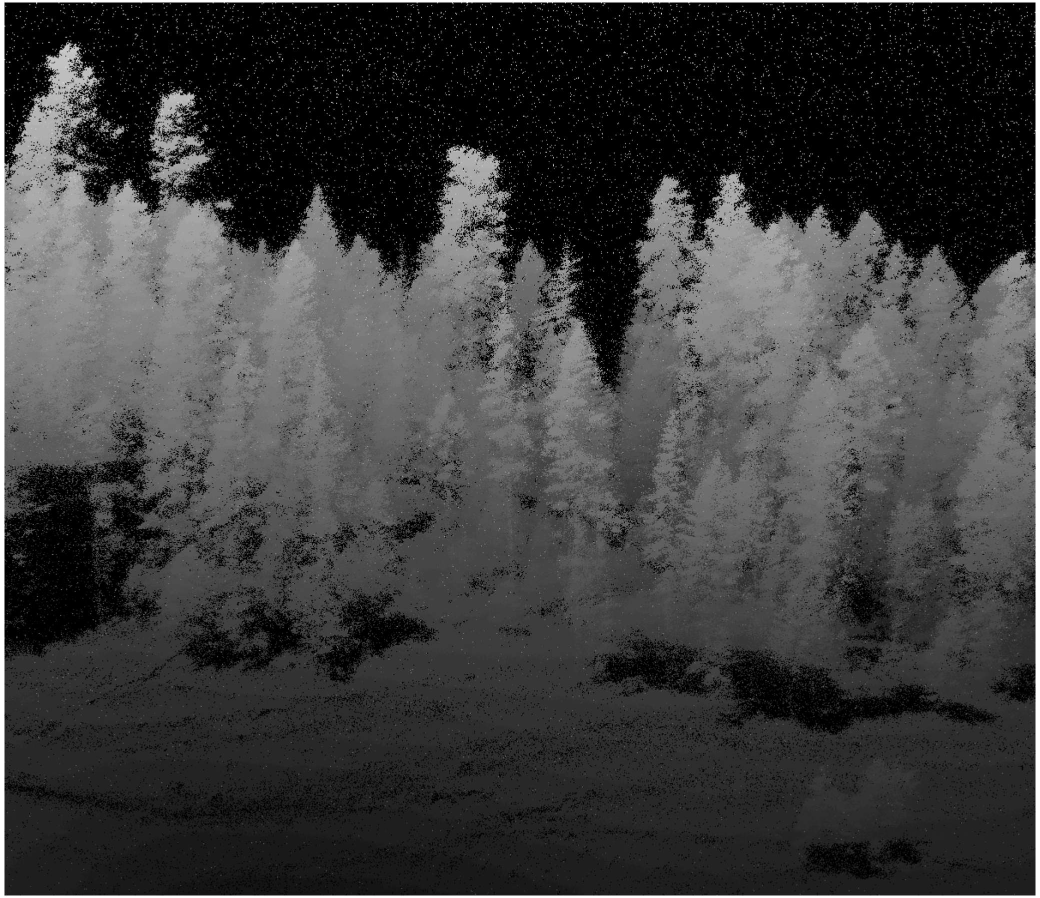
Future steps:

- Train/validation split
- Better pre- and/or post-registration features
- Better registration metric

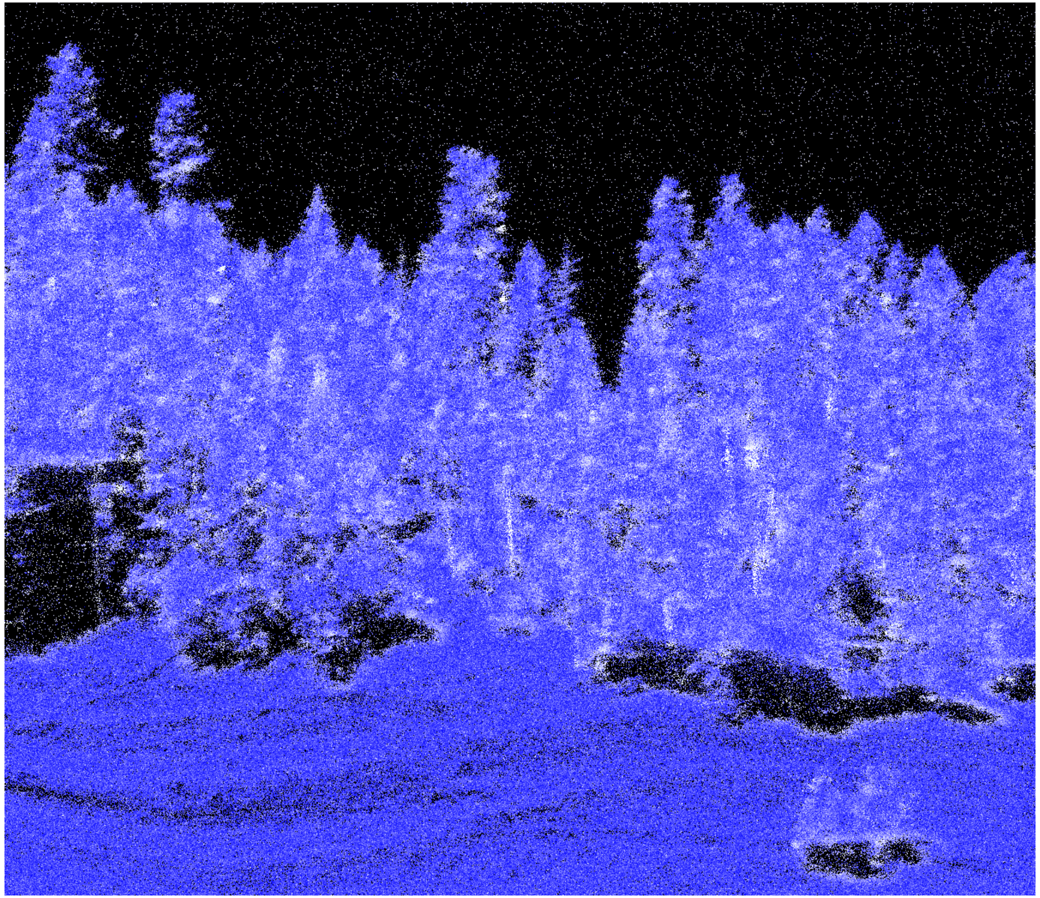
Ideas for registration metrics

Metric ideas	Advantages
Maximize average density per point (k-NN estimation)	Relatively fast as-is (still not fast enough)
Voxelize and minimize non-zero voxels	GPU parallelizable?
Maximize "sharpness" around highly linear regions (power lines, tree trunks, etc.)	Similar to what humans do

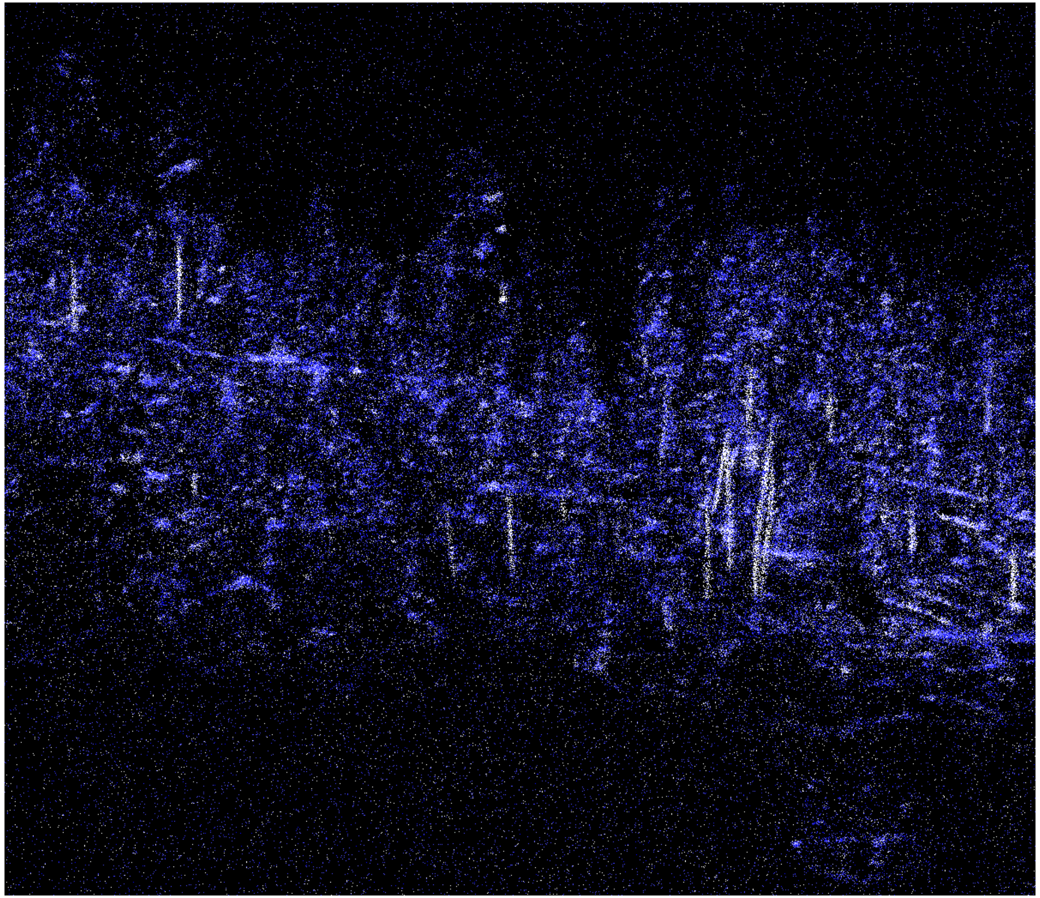
Potential "sharpness" registration metric



Potential "sharpness" registration metric



Potential "sharpness" registration metric



Summary

- Pose graph optimization is a fast, flexible, and interpretable way to combine pairwise registrations
- Accurately modeling covariances (uncertainty) of pairwise registrations is difficult
- Fast registration metric probably essential to create accurate covariance model