

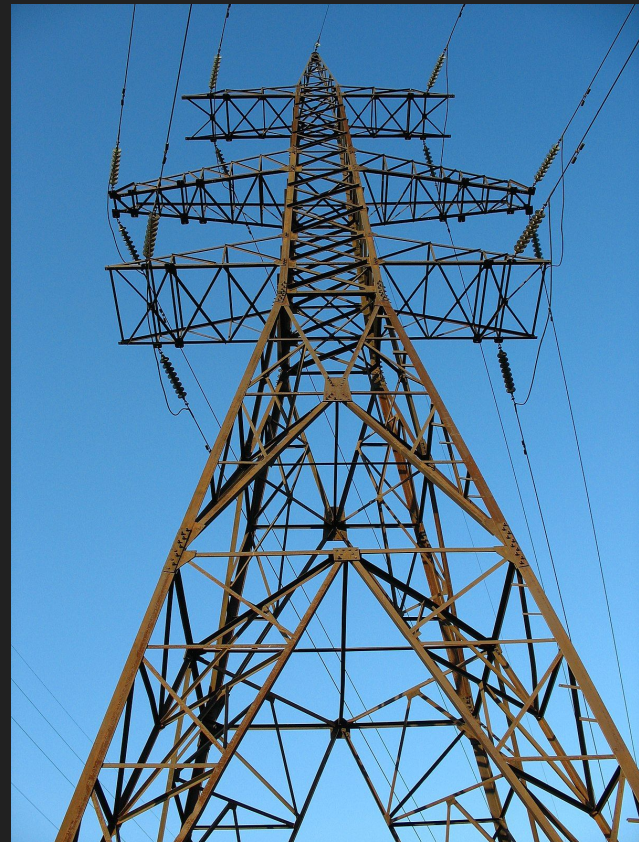
# Point Cloud Segmentation of Infrastructural Steel Elements

DISAL-SP 183

Dimitri Jacquemont

# Enhancing Infrastructure Inspection

- Current solutions **risky**, and **time consuming**
- Innovative approach using **UAV** equipped with **ToF camera**.
- **Real-time segmentation pipeline** capable of efficiently analyzing 3D point-cloud data, to **monitor the integrity of steel infrastructures**.



# State of the Art in Point Cloud Segmentation

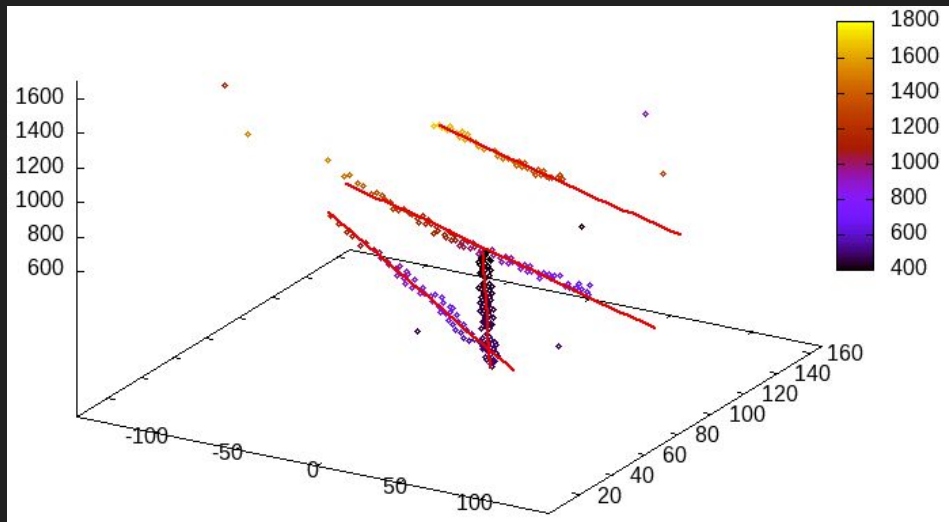
## Iterative Hough Transform for Line Detection in 3D Point Clouds

Dalitz, Christoph, Tilman Schramke, and Manuel Jeltsch, 2017  
([https://www.ipol.im/pub/art/2017/208/?utm\\_source=doi](https://www.ipol.im/pub/art/2017/208/?utm_source=doi))

- Method **detecting lines in 3D point clouds** using the Hough Transform

Other solution explored (previous SP)

- RANSAC
- DBSCAN
- GMM

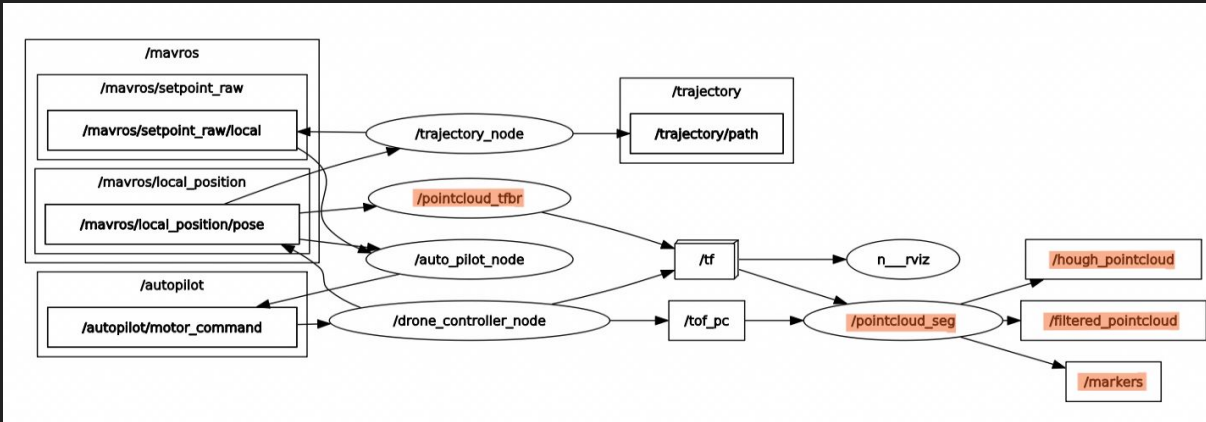


Algorithm output

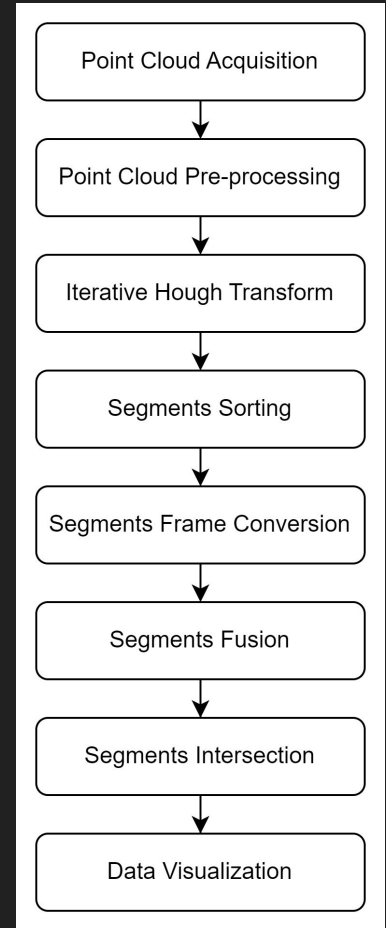
```
nbpoints=78, a=(52.889395,52.335678,460.717949), b=(0.564829, -0.539195, 0.624689)
nbpoints=70, a=(69.556691,28.172875,1018.697143), b=(0.126173, 0.064910, -0.989882)
nbpoints=47, a=(54.826033,9.888771,818.893617), b=(0.116360, 0.020515, -0.992995)
nbpoints=36, a=(75.600461,38.137831,1496.355556), b=(0.110137, 0.070930, -0.991382)
```

# Key Achievements

- **Pipeline** creation/**ROS node** integration
- **Adaptation** of the **C++ hough-3d-lines** library
- Experimental **testing** & performances **quantification**



Global Node Graph



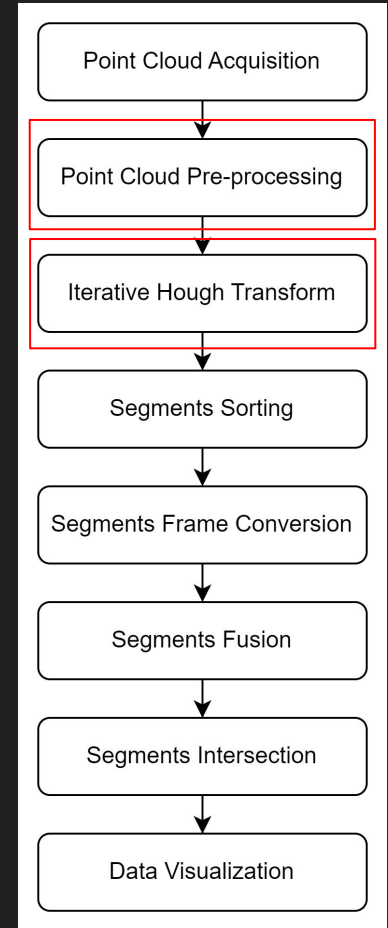
# Methodology & Implementation

## Point Cloud Pre-processing:

- Filtering thresholds within a specified window
- Voxel grid filter

## Iterative Hough Transform:

- Creation of a function running the Hough Transform
- Hough Library modified for real time (Hough Space initialization)



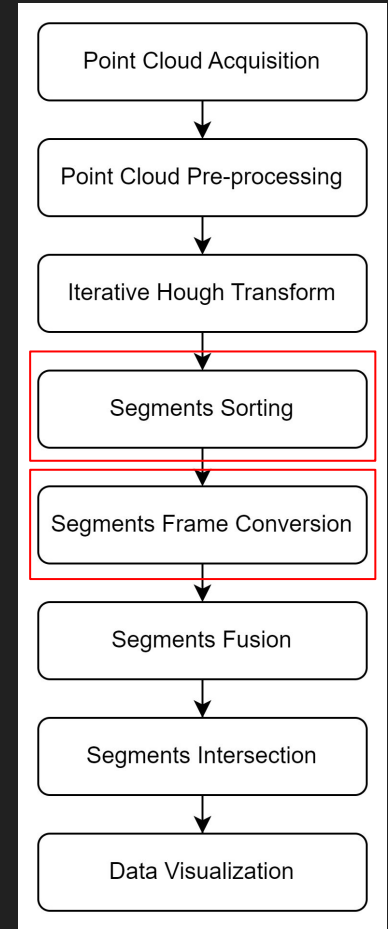
# Methodology & Implementation

## Segments Sorting:

- Number of points
- Beam radius
- Segment points density
- PCA

## Segments Frame Conversion:

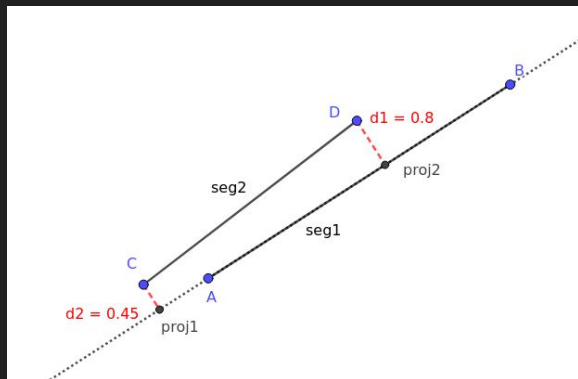
- Transformation from drone frame to world frame
- Segments' parameters transformation



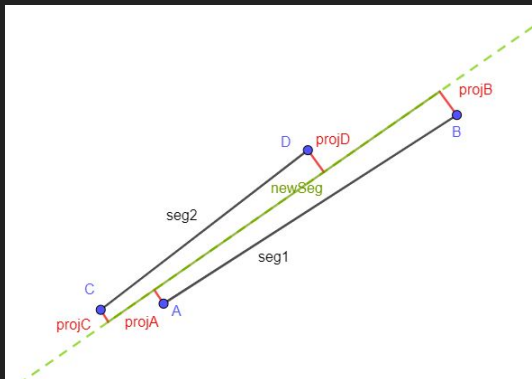
# Methodology & Implementation

## Segments Fusion:

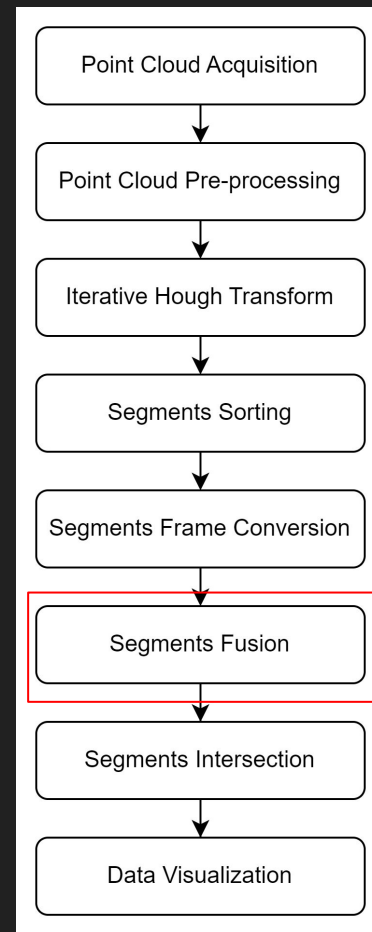
- Similarity detection
- Weighted fusion



Segment Similarity Detection



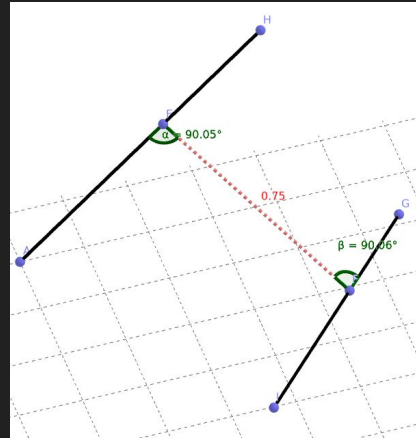
Segment Weighted Fusion



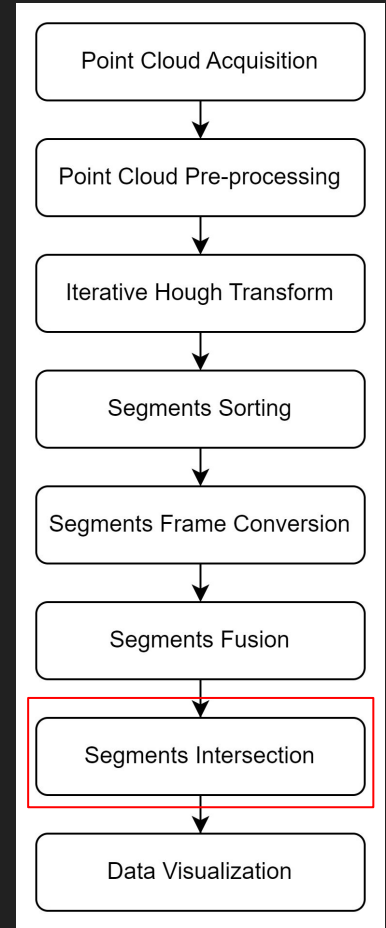
# Methodology & Implementation

## Segments Intersection:

- Shortest distance between 2 segments
- Solving linear system to find intersection point



Segment Intersection

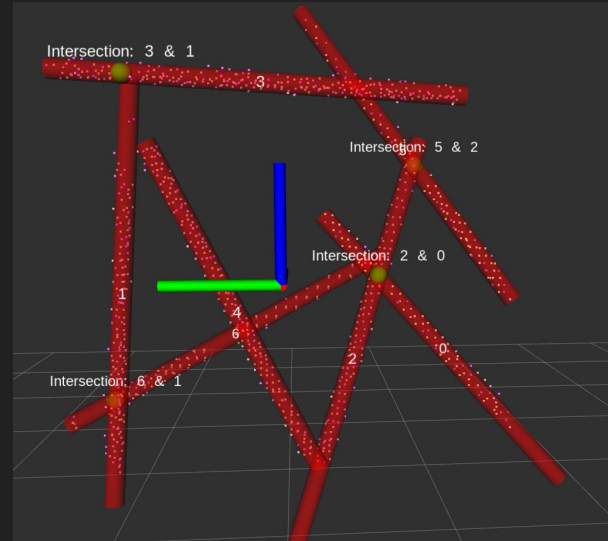




# Testing & Results

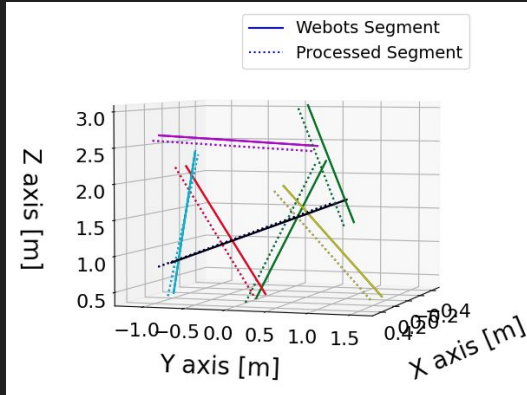


Simulation Environment



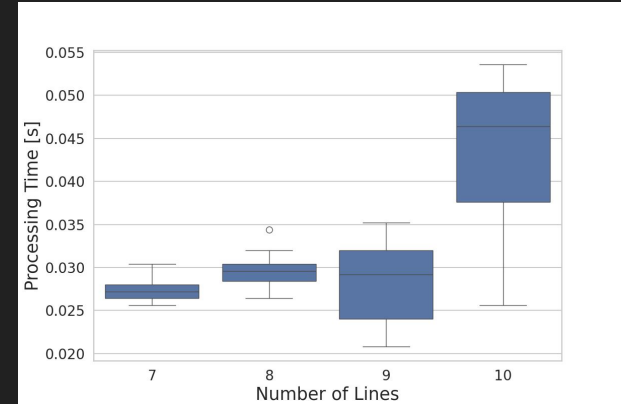
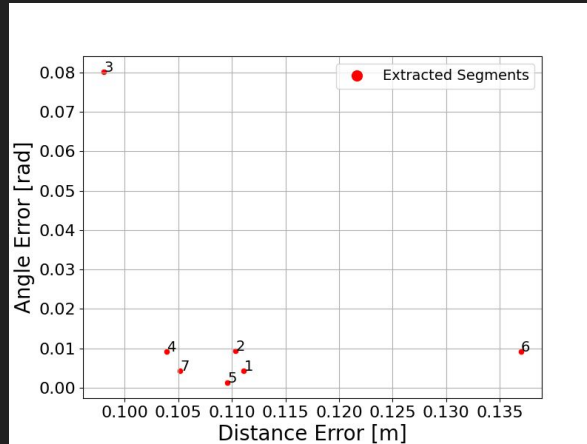
Extracted beams

# Testing & Results



Segment Comparison

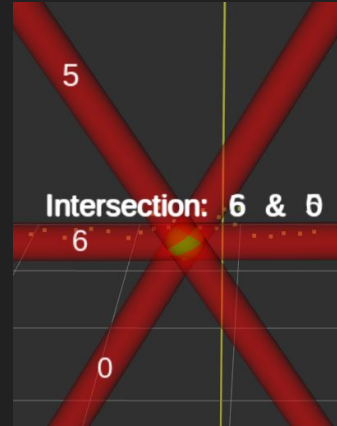
Distance VS Angle Error



Processing Time Distribution by Number of Lines

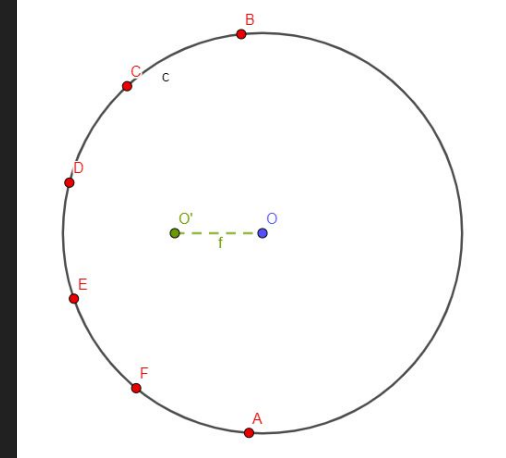
# Future Directions and Improvements

- Intersections with flat angles
- Intersection fusion
- Beam diversity
- Inexact drone pose
- Multiple radius segment detection
- Ground truth offset
- Segment division upon intersection



Intersection Example

Ground Truth Offset



# Q&A

