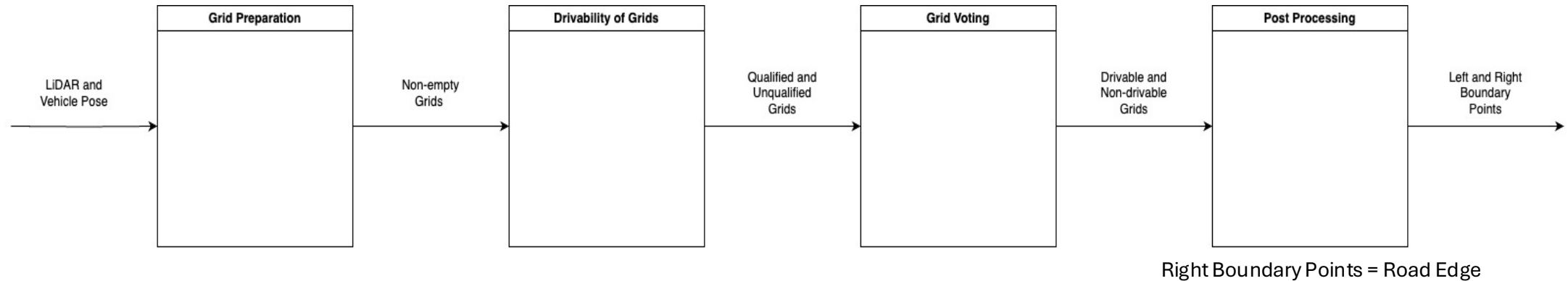# Boundary points from 2130 - 2160

Aneesh Batchu

Oct 5th, 2024

Goal: Convert 3D lidar points into the calculation of the road edge. The BIG PICTURE of the process is shown below
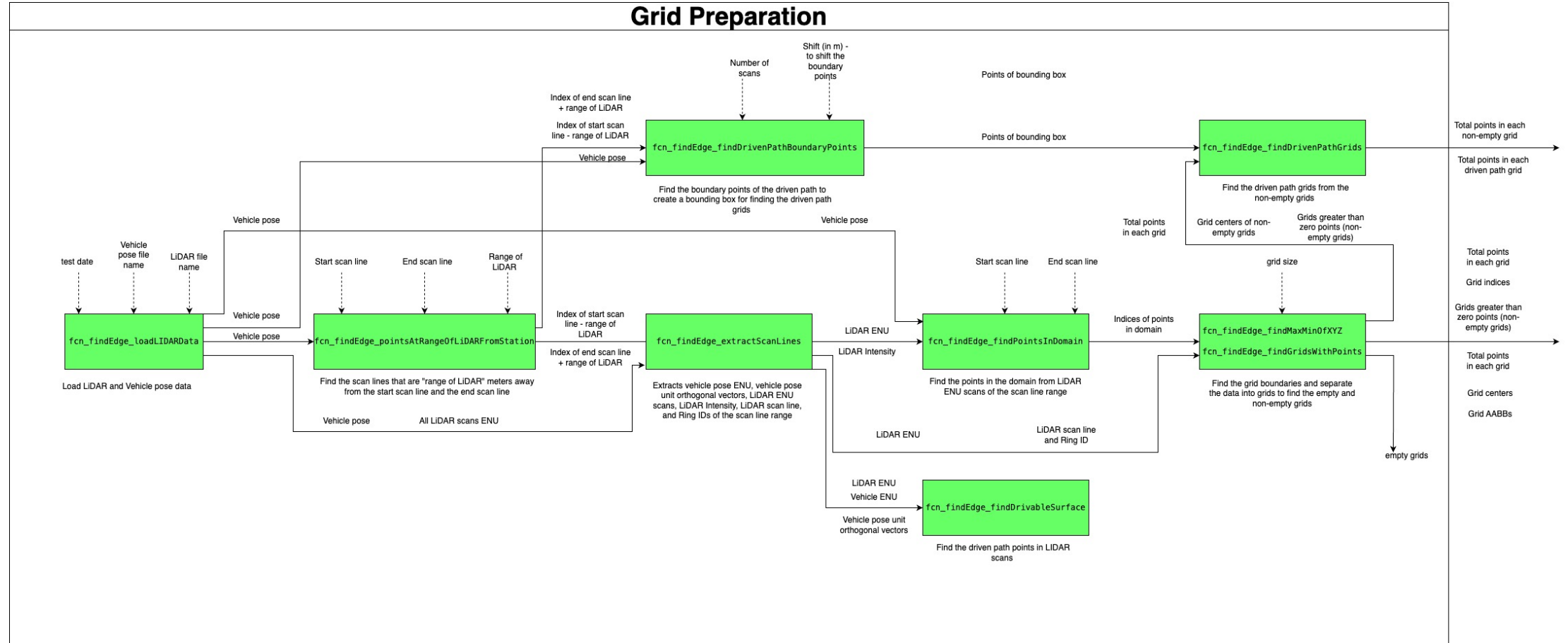
| LiDAR and Vehicle Pose → | **Grid Preparation** | Non-empty Grids → | **Drivability of Grids** | Qualified and Unqualified Grids → | **Grid Voting** | Drivable and Non-drivable Grids → | **Post Processing** | Left and Right Boundary Points → |

Right Boundary Points = Road Edge

- Road edge: The edge of the pavement and the vegetation next to the road.

Overview

Grid Preparation

fcn_findEdge_loadLiDARData
Load LiDAR and Vehicle pose data

fcn_findEdge_pointsAtRangeOfLiDARFromStation
Find the scan lines that are "range of LiDAR" meters away from the start scan line and the end scan line

fcn_findEdge_extractScanLines
Extracts vehicle pose ENU, vehicle pose unit orthogonal vectors, LiDAR ENU scans, LiDAR Intensity, LiDAR scan line, and Ring IDs of the scan line range

fcn_findEdge_findPointsInDomain
Find the points in the domain from LiDAR ENU scans of the scan line range

fcn_findEdge_findGridsWithPoints
Find the grid boundaries and separate the data into grids to find the empty and non-empty grids

fcn_findEdge_findDrivablePathBoundaryPoints
Find the boundary points of the driven path to create a bounding box for finding the driven path grids

fcn_findEdge_findDrivenPathGrids
Find the driven path grids from the non-empty grids

fcn_findEdge_findDrivableSurface
Find the driven path points in LiDAR scans

Drivability of Grids

Grid conditions

no. of points per non-empty grid >= suitable point density

fcn_findEdge_determineGridPointDensity
fcn_findEdge_classifyGridsBasedOnDensity
Determine the suitable "point density" for the analysis by comparing the point densities of grids with those of neighboring grids

no. of scan lines in each non-empty grid >= 1

fcn_findEdge_calcNumberOfGridScanLines
fcn_findEdge_classifyGridsBasedOnScanLines
Determine the number of LiDAR scan lines in each grid

minimum transverse span >= transverse_span_threshold

fcn_findEdge_determineTransverseSpanThreshold
fcn_findEdge_classifyGridsBasedOnTransverseSpan
Find the orthogonal distances of points in the remaining rings by projecting orthogonally from one ring.

fcn_findEdge_findDrivenPathGrids
Find the driven paths in the qualified grids

fcn_findEdge_classifyQualifiedGrids
If either of the conditions fail, the non-empty grids are classified as UNQUALIFIED. If all the conditions pass, the non-empty grids are classified as QUALIFIED.

Grid Voting

Grid conditions

standard deviation in z fit error < standard deviation threshold

fcn_findEdge_determineSTDInZError
fcn_findEdge_histogramSTDInZError
Find the standard deviations of the z-fit errors for all qualified grids, and compare these standard deviations with those of the grids in the driven path to determine the suitable standard deviation threshold

angle between unit normal vector and vertical < theta threshold

fcn_findEdge_determineAngleDeviation
fcn_findEdge_histogramAngleDeviation
Find the angles between the unit normal vectors of the fitted planes and vertical ([0 0 1]) for all qualified grids, and compare these angles with those of the grids in the driven path to determine the suitable theta threshold

z-intercept of the fitted plane < z-height threshold

Find the z-intercept of the fitted planes for all qualified grids, and compare these angles with those of the grids in the driven path to determine the suitable z-height threshold

fcn_findEdge_classifyGridsAsDrivable
If all conditions pass, DRIVABLE
If all conditions fail, NON_DRIVABLE
If only a few fail, UNCERTAIN

Post Processing

fcn_findEdge_prepGridCentersForBoundaryDetection
Prepare the grid centers of qualified and unqualified for boundary detection

fcn_findEdge_findBoundaryPoints
Find the boundary points of qualified and unqualified grids

fcn_findEdge_prepGridCentersForBoundaryDetection
Prepare the grid centers of drivable and non-drivable for boundary detection

fcn_findEdge_findBoundaryPoints
Find the boundary points of drivable and non-drivable grids

fcn_findEdge_findTrueBoundaryPoints
Find the true boundary points by removing the boundary points of qualified and unqualified from drivable and non-drivable boundary points

fcn_findEdge_findNearestBoundaryPoints
Find the nearest boundary points to the driven path

fcn_findEdge_seperateLeftRightBoundaries
Separate the left and right boundaries from the nearest boundaries

Grid Preparation

Grid Preparation

Number of scans

Shift (in m) - to shift the boundary points

Points of bounding box

Index of end scan line + range of LiDAR

Index of start scan line - range of LiDAR

Vehicle pose

fcn_findEdge_findDrivenPathBoundaryPoints

Points of bounding box

fcn_findEdge_findDrivenPathGrids

Total points in each non-empty grid

Total points in each driven path grid

Find the boundary points of the driven path to create a bounding box for finding the driven path grids

Find the driven path grids from the non-empty grids

Vehicle pose

Vehicle pose

Total points in each grid

Grid centers of non-empty grids

Grids greater than zero points (non-empty grids)

test date

Vehicle pose file name

LiDAR file name

Vehicle pose

Start scan line

End scan line

Range of LiDAR

Start scan line

End scan line

grid size

Total points in each grid

Grid indices

fcn_findEdge_loadLIDARData

Vehicle pose

Index of start scan line - range of LiDAR

Index of end scan line + range of LiDAR

fcn_findEdge_pointsAtRangeOfLiDARFromStation

fcn_findEdge_extractScanLines

LiDAR ENU

LiDAR Intensity

fcn_findEdge_findPointsInDomain

Indices of points in domain

fcn_findEdge_findMaxMinOfXYZ

fcn_findEdge_findGridsWithPoints

Grids greater than zero points (non-empty grids)

Total points in each grid

Load LiDAR and Vehicle pose data

Find the scan lines that are "range of LiDAR" meters away from the start scan line and the end scan line
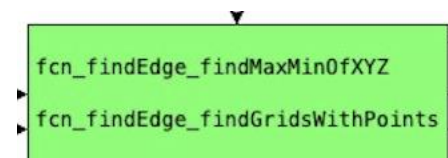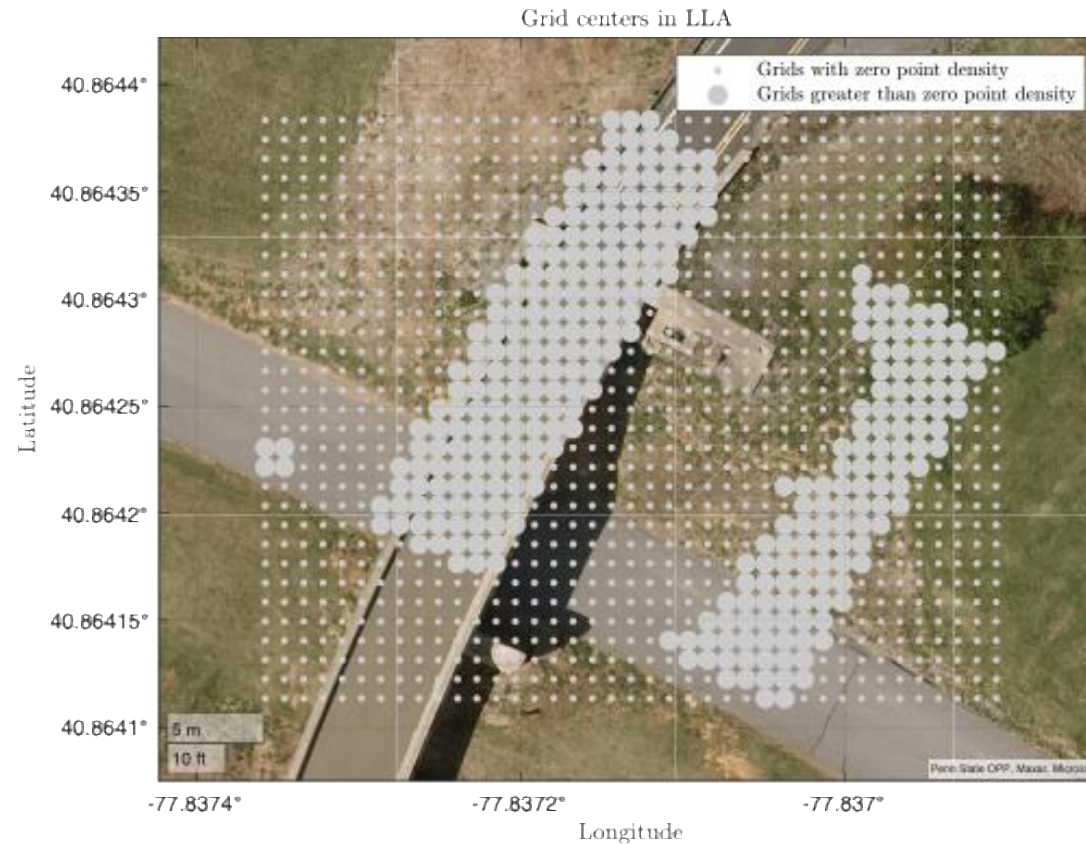
Extracts vehicle pose ENU, vehicle pose unit orthogonal vectors, LiDAR ENU scans, LiDAR Intensity, LiDAR scan line, and Ring IDs of the scan line range

Find the points in the domain from LiDAR ENU scans of the scan line range

Find the grid boundaries and separate the data into grids to find the empty and non-empty grids

Grid centers

Grid AABBs

Vehicle pose

All LiDAR scans ENU

LiDAR ENU

LiDAR scan line and Ring ID

LiDAR ENU

Vehicle ENU

fcn_findEdge_findDrivableSurface

Vehicle pose unit orthogonal vectors

empty grids

Find the driven path points in LiDAR scans

Load LiDAR and Vehicle Pose Data

🔤	LIDAR_file_stri...	'Velodyne_LiDAR...
{}	LiDAR_Scan_E...	*2463x1 cell*
🔤	test_date_string	'2024_06_28'
🔤	vehicle_pose_s...	'VehiclePose_ENU...
▦	VehiclePose	*2463x6 double*

fcn_findEdge_loadLIDARData

LiDAR scan lines [2130 - 2160] are plotted on a geo plot. The RHS plot is the zoomed-in plot of the LHS
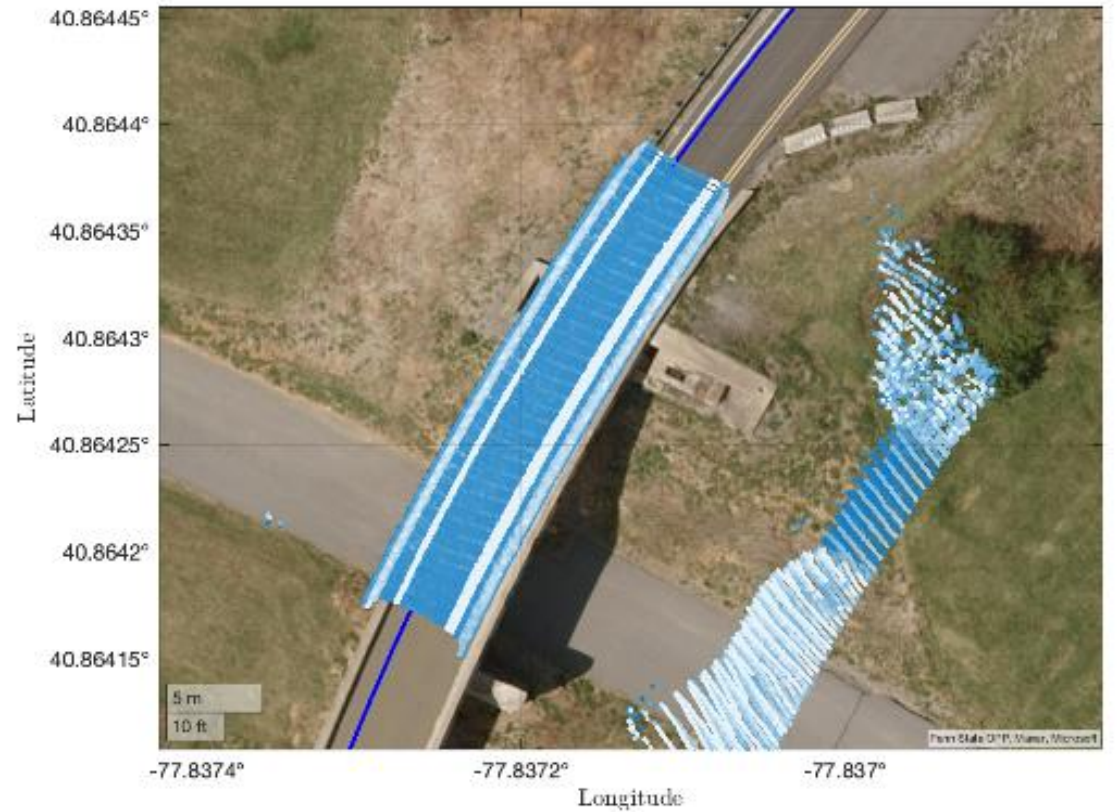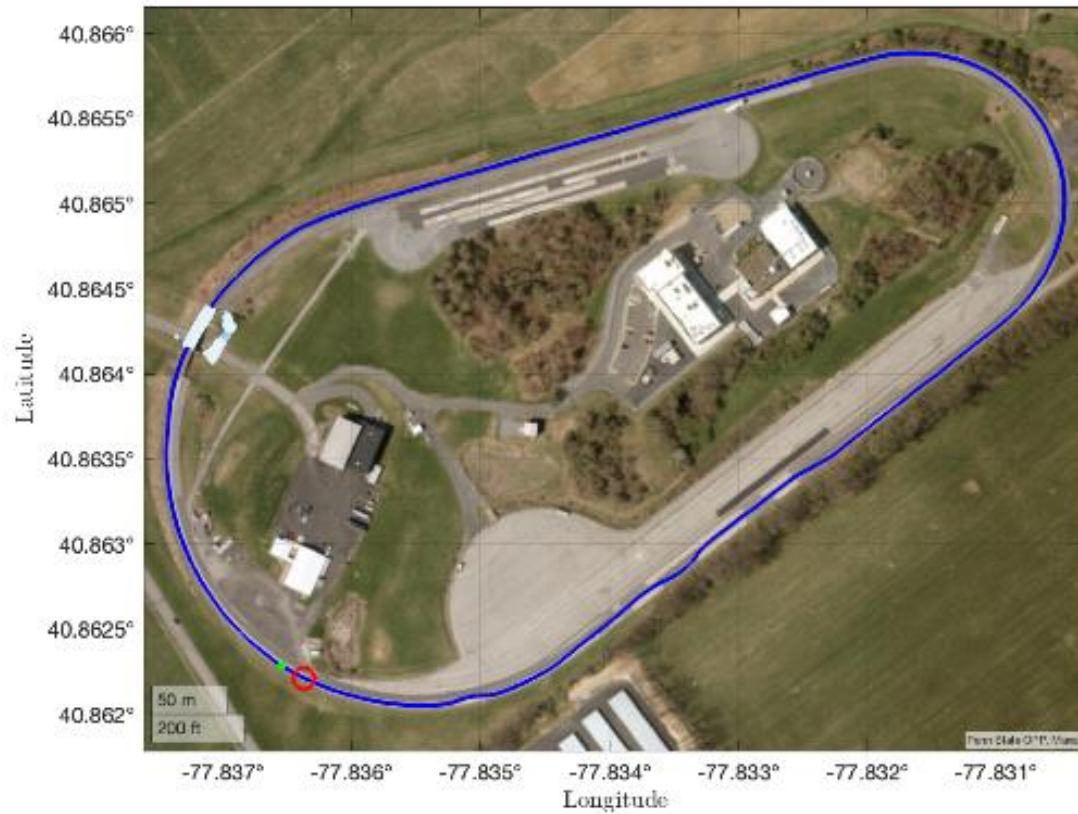
fcn_findEdge_loadLIDARData

The scan lines that are "range of LiDAR" meters away from the start scan line and the end scan line are computed

The index of the scan line that is 100 meters before the start scan line is 2016.
The index of the scan line that is 100 meters after the end scan line is 2284.

fcn_findEdge_pointsAtRangeOfLiDARFromStation
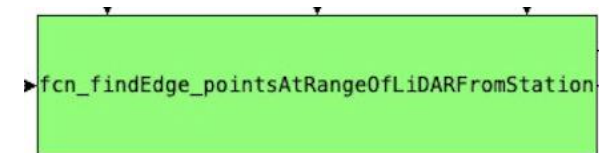
The gaps in the LiDAR scans (2130-2160) (LHS) indicate that insufficient data was captured. To address this, the indices of the scan lines that are 100 meters (the range of LiDAR) before the start scan line (2130) and 100 meters after the end scan line (2160) were determined by computing the distances of the scan lines from each station.
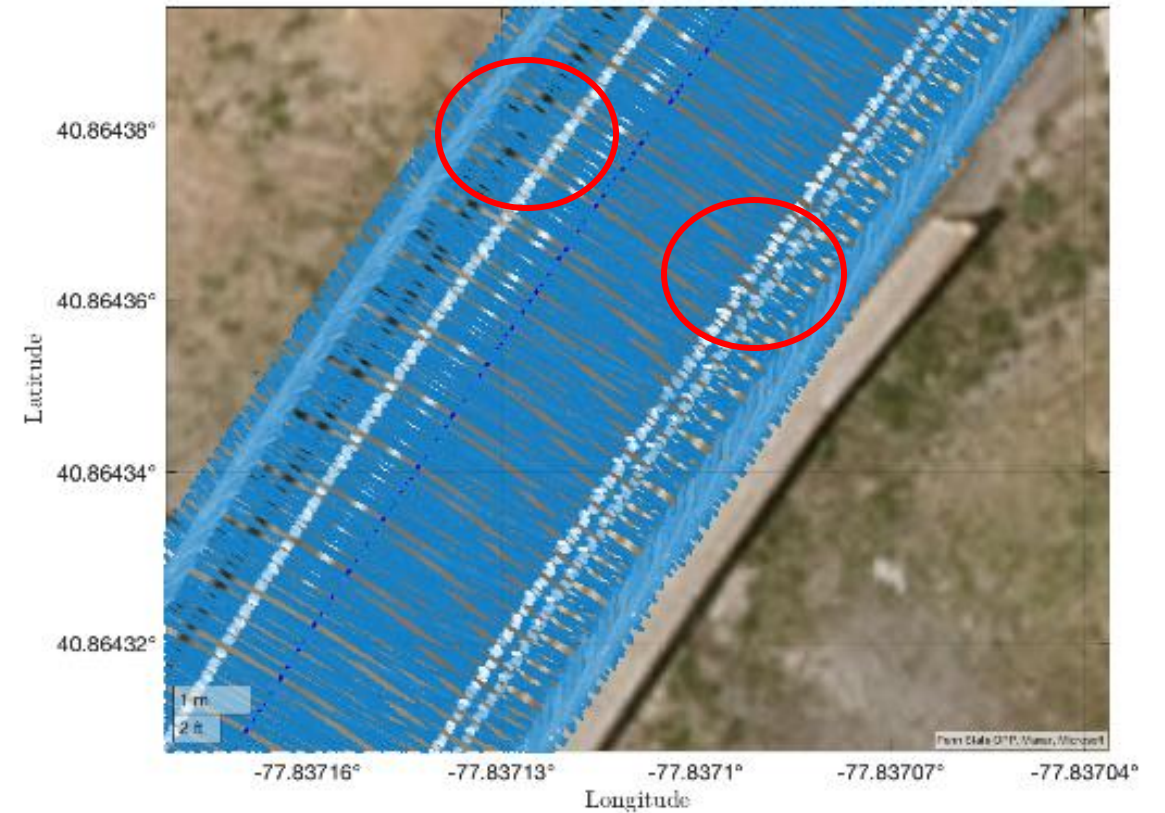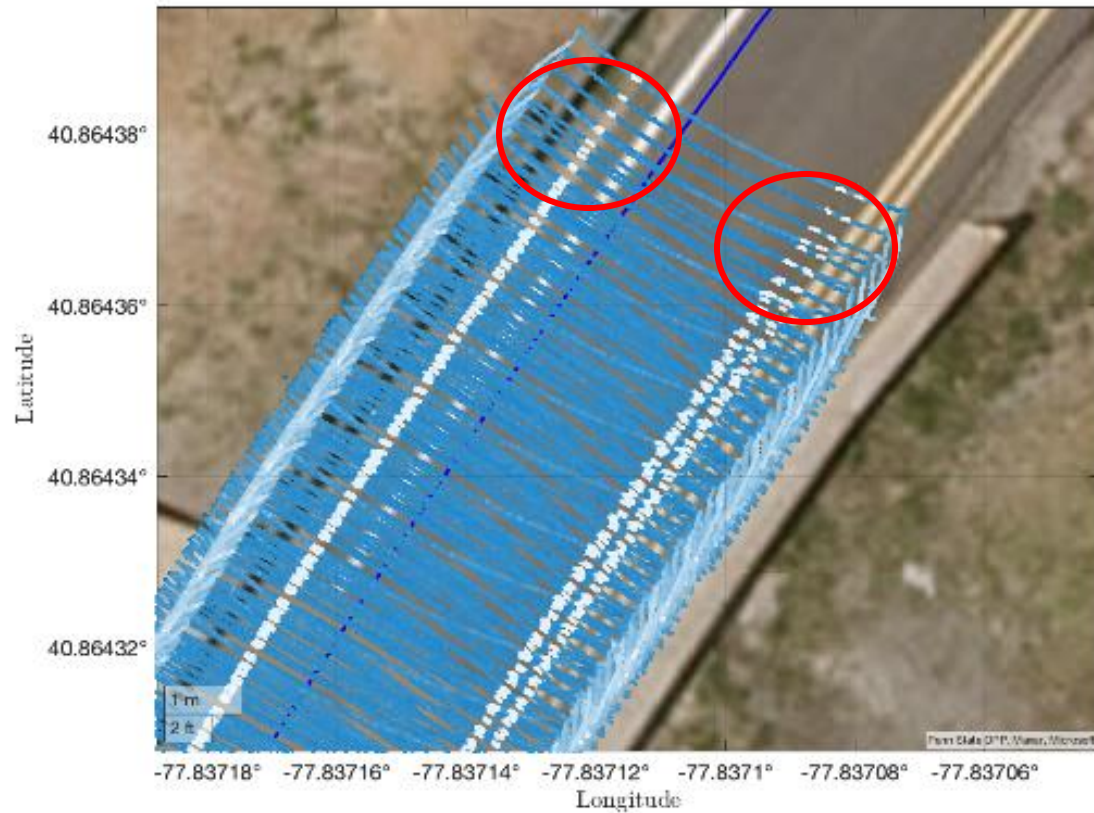
`fcn_findEdge_pointsAtRangeOfLiDARFromStation`

Extract the vehicle pose ENU, vehicle pose unit orthogonal vectors, LiDAR ENU scans, LiDAR Intensity, LiDAR scan line, and Ring IDs of the scan line range ([745 1111])

LIDAR_ENU	2841669x3 double
LIDAR_scanLin...	2841669x2 double
LIDAR_intensity	2841669x1 double
VehiclePose_E...	2841669x3 double
VehiclePose_U...	2841669x3 double

fcn_findEdge_extractScanLines

The points in the domain are selected by extending the orthogonal vector of the change in the XY position of the vehicle by 22 m (six times lane width). All the points within these boundaries (red points) are considered to be in the domain.


```
fcn_findEdge_findPointsInDomain
```

LiDAR data is segmented into grids to classify them into those with zero points and those with more than zero points (non-empty grids). The grids with zero points (empty grids) are then dropped from the further analysis.

Grid centers in LLA

fcn_findEdge_findMaxMinOfXYZ

fcn_findEdge_findGridsWithPoints

The boundary points (LHS) of the driven path are computed to create a bounding box for finding the driven path grids. The RHS plot has the driven path grids

Drivability of Grids

Drivability of Grids

Grid size Point density Transverse span threshold

Learned

Grid conditions

Total points in each non-empty grid

Total points in each driven path grid

no. of points per non-empty grid >= suitable point density

`fcn_findEdge_determineGridPointDensity`

`fcn_findEdge_classifyGridsBasedOnDensity`

Determine the suitable "point density" for the analysis by comparing the point densities of driven grids with those of neighboring grids

Total points in each grid

Points of bounding box

`fcn_findEdge_findDrivenPathGrids`

Find the driven paths in the qualified grids

Current grid numbers of driven path

Grid centers of driven path grids

Total points in each grid

Grid indices

no. of scan lines in each non-empty grid >= 1

`fcn_findEdge_calcNumberOfGridScanLines`

`fcn_findEdge_classifyGridsBasedOnScanLines`

Determine the number of LiDAR scan lines in each grid

Original qualified grids Current qualified grids

Total points in each grid

Grid centers

Grid AABBs

Non-empty grids greater than required point density

Non-empty grids with more than one scan line

Grid centers of qualified grids

Current qualified grids

Original qualified grids

minimum transverse span >= transverse_span_threshold

`fcn_findEdge_determineTransverseSpanThreshold`

`fcn_findEdge_classifyGridsBasedOnTransverseSpan`

Find the orthogonal distances of points in the remaining rings by projecting orthogonally from one ring.

Non-empty grids with more than transverse span threshold

`fcn_findEdge_classifyQualifiedGrids`

If either of the conditions fail, the non-empty grids are classified as UNQUALIFIED. If all the conditions pass, the non-empty grids are classified as QUALIFIED.

Grid centers of qualified grids

Unqualified grids

Grids greater than zero points (non-empty grids) Grid centers

Grid Condition: Determine the point density (points per grid) by comparing the driven path densities with those of neighboring grids. This comparison has been done for multiple datasets with different grid sizes. In the current analysis, the grid size is chosen as 1 m. The RHS plot depicts the grids with the required point density and those with insufficient point density.

fcn_findEdge_determineGridPointDensity

fcn_findEdge_classifyGridsBasedOnDensity

Determining suitable point density

chosen point density = 222

The orange and blue bins correspond to the point densities of the driven path and all the grids, respectively.

Grid Centers in LLA

Grids with low point density
Grids with required density

For plane fitting in the next steps, it was determined that each grid should contain at least 20 points when the grid size is 0.3 m (0.3 × 0.3). Therefore, this criterion has been extrapolated for other grid sizes as well.

The point density is calculated as:

$$point\ density = floor(20 * \frac{grid\ size^2}{0.3^2})$$

Grid Condition: Determine the number of scan lines in non-empty grids. Grids containing only one scan line cannot be used to fit a plane, as the angle deviation in these grids is high and they are automatically classified as uncertain grids (neither drivable nor non-drivable). To reduce the number of uncertain grids, these grids are identified and dropped from further analysis. Fitting a plane and angle deviation are discussed in the following slides.

Grids with one scan line
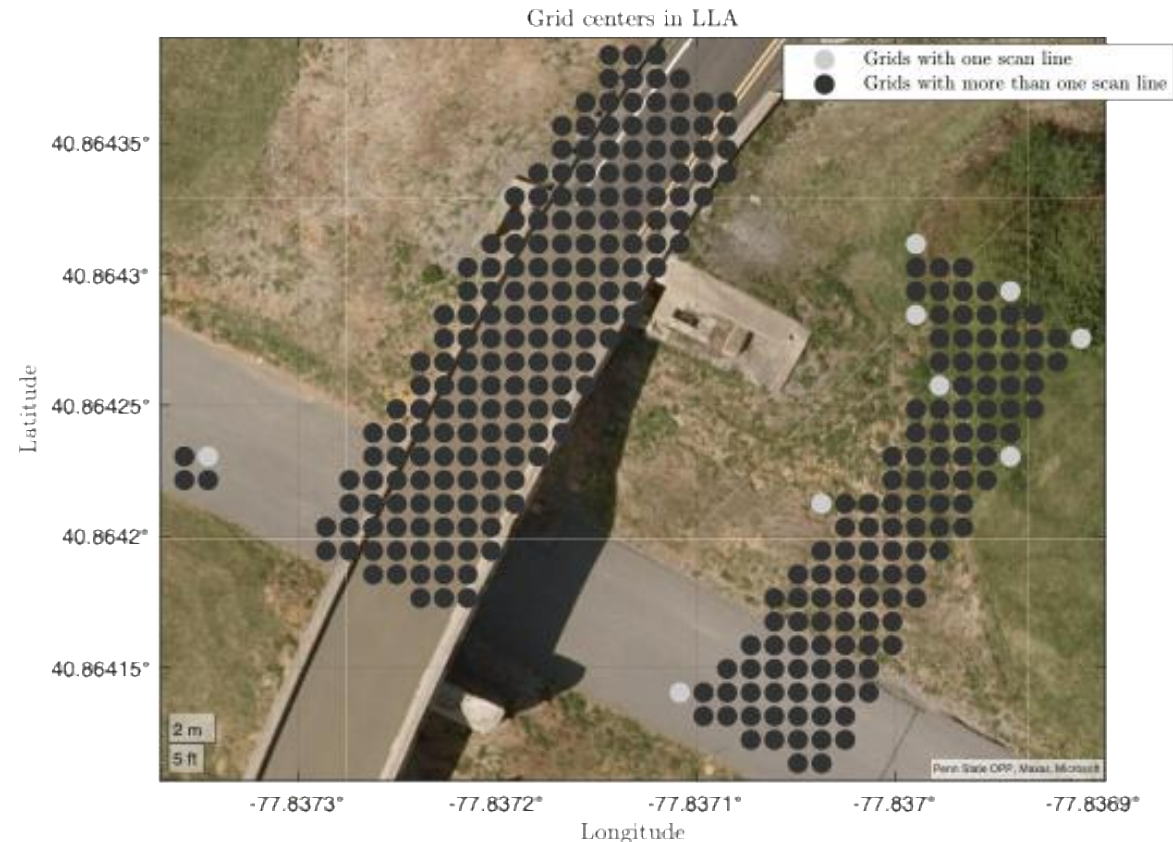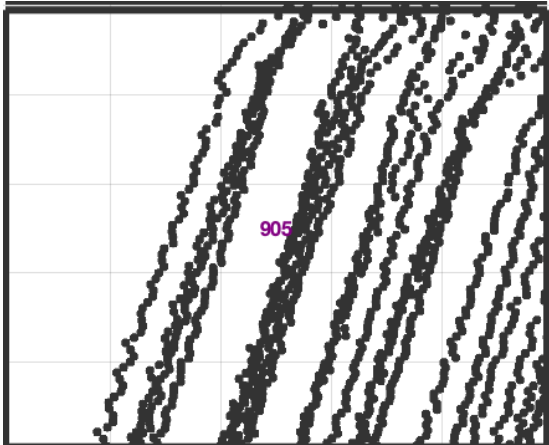
Grids with more than one scan line

Grid centers in LLA


```
fcn_findEdge_calcNumberOfGridScanLines

fcn_findEdge_classifyGridsBasedOnScanLines
```

Grid Condition: Sometimes, grids containing more than one scan line may still be classified as uncertain (as discussed in later slides) due to the overlapping of scan lines within the grid. It has been found that the maximum span distance of each grid should be at least 0.15 meters, based on an analysis of various locations and grid sizes.

Maximum span distance of this grid < 0.15m
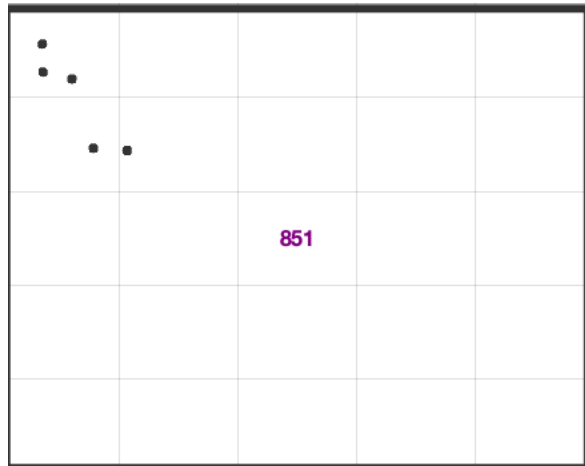
Maximum span distance of this grid > 0.15m
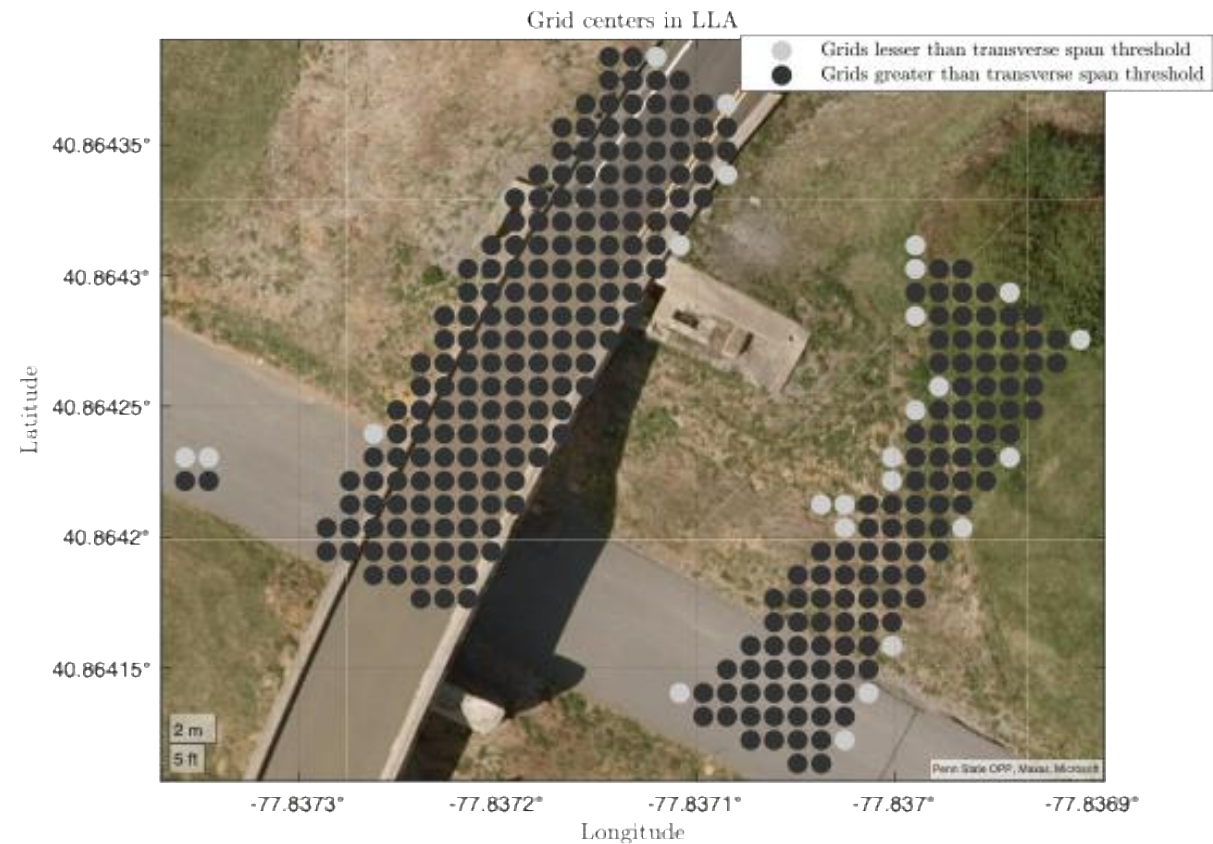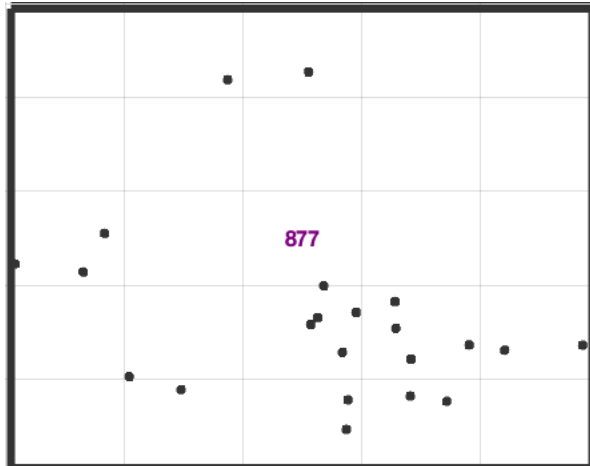
Grid centers in LLA


```
fcn_findEdge_determineTransverseSpanThreshold
fcn_findEdge_classifyGridsBasedOnTransverseSpan
```

Grids that contain sufficient point density, have more than one scan line, and meet the maximum span distance are classified as qualified grids. The remaining grids are classified as unqualified and are dropped from further analysis.

Grid centers in LLA

Legend:
- Grids with one scan line
- Grids with more than one scan line
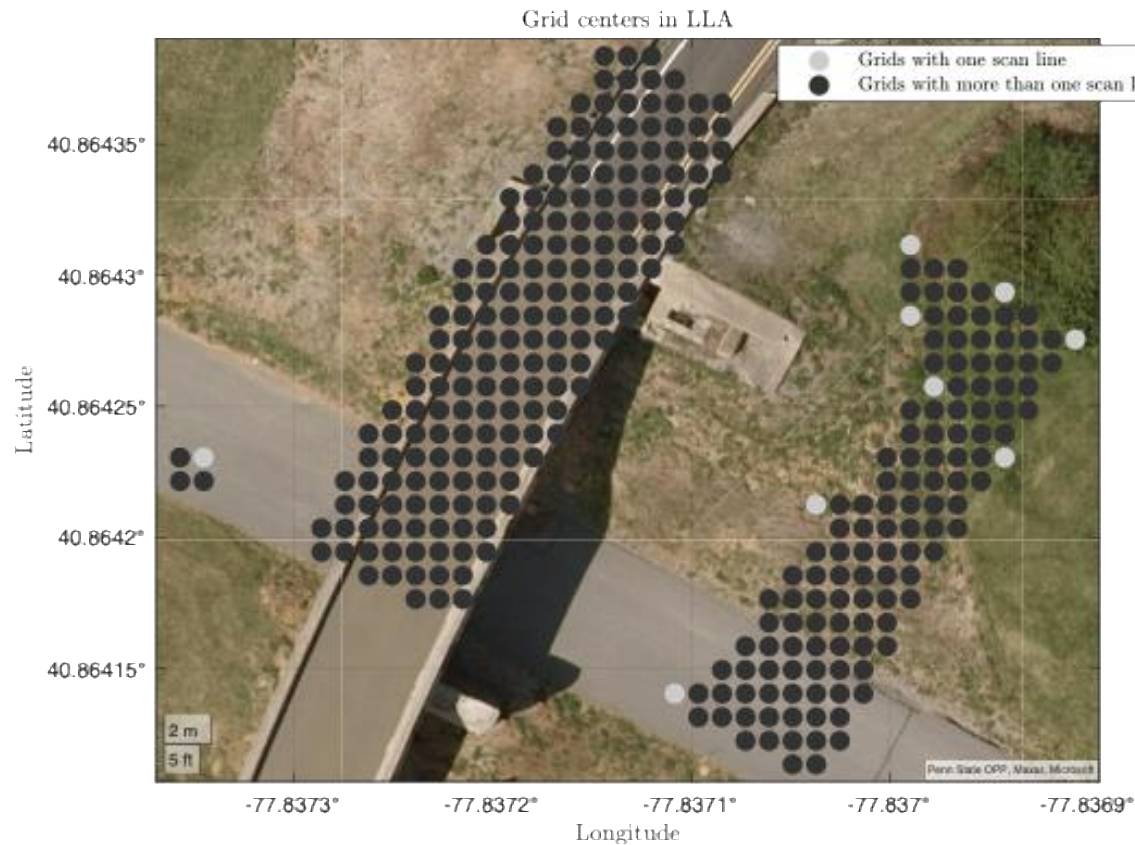
fcn_findEdge_classifyQualifiedGrids

The failed conditions of unqualified grids are highlighted with different color circles.

Grid centers in LLA

Legend:
- Unqualified grids
- Qualified grids
- Grids with low point density
- Grids with one scan line
- transverse span threshold fail
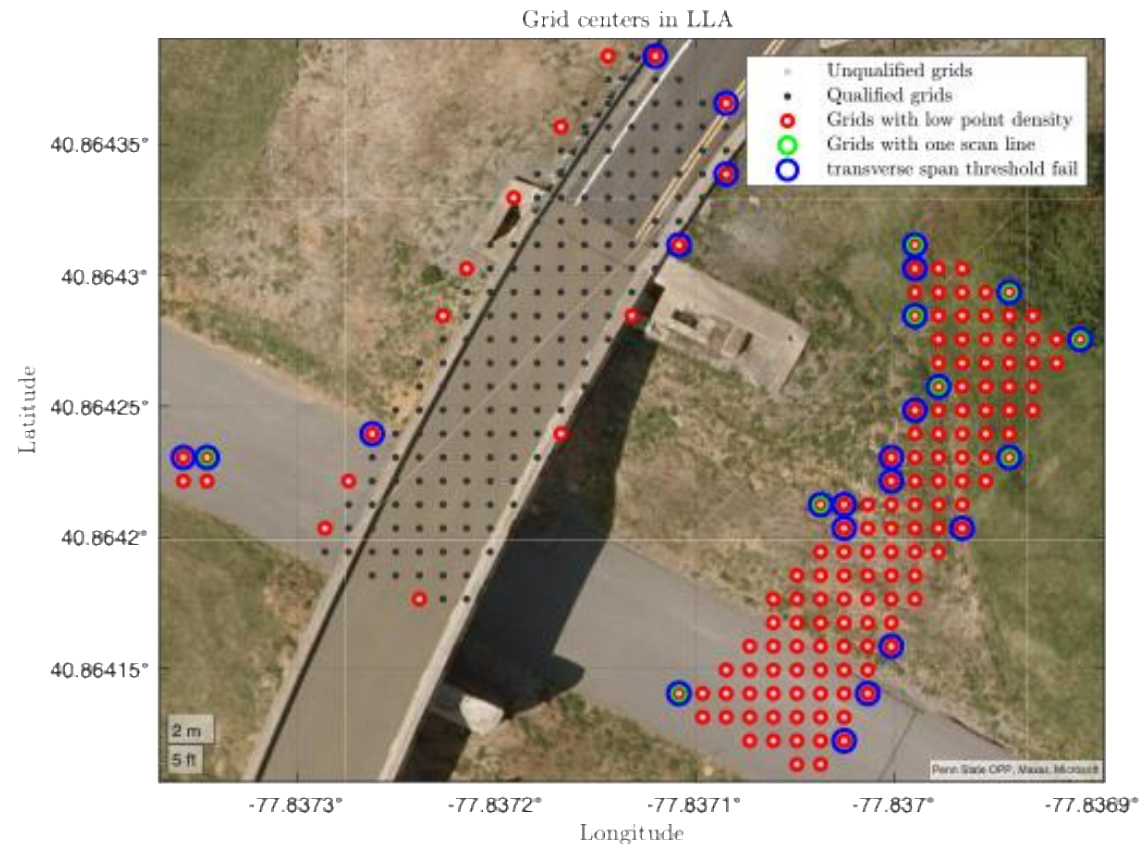
`fcn_findEdge_classifyQualifiedGrids`

Recalculate the driven path grids from the qualified grids for further analysis.

Grid centers in LLA
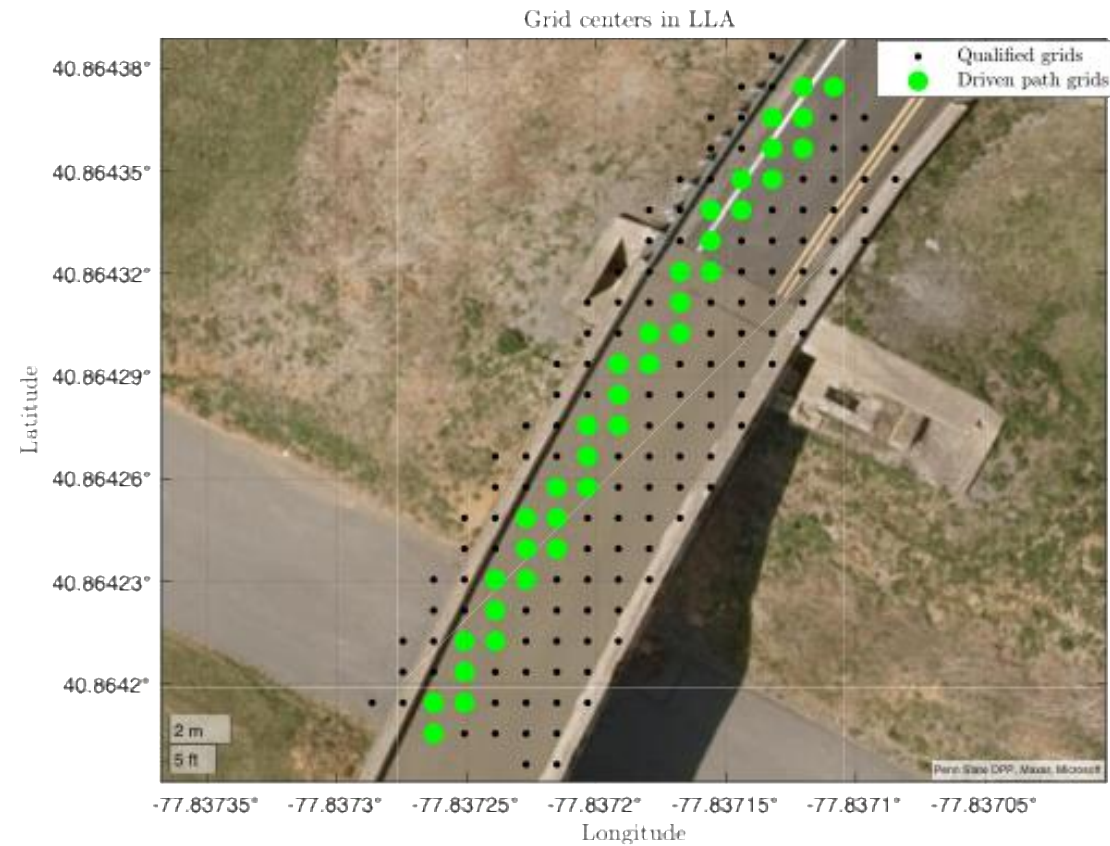
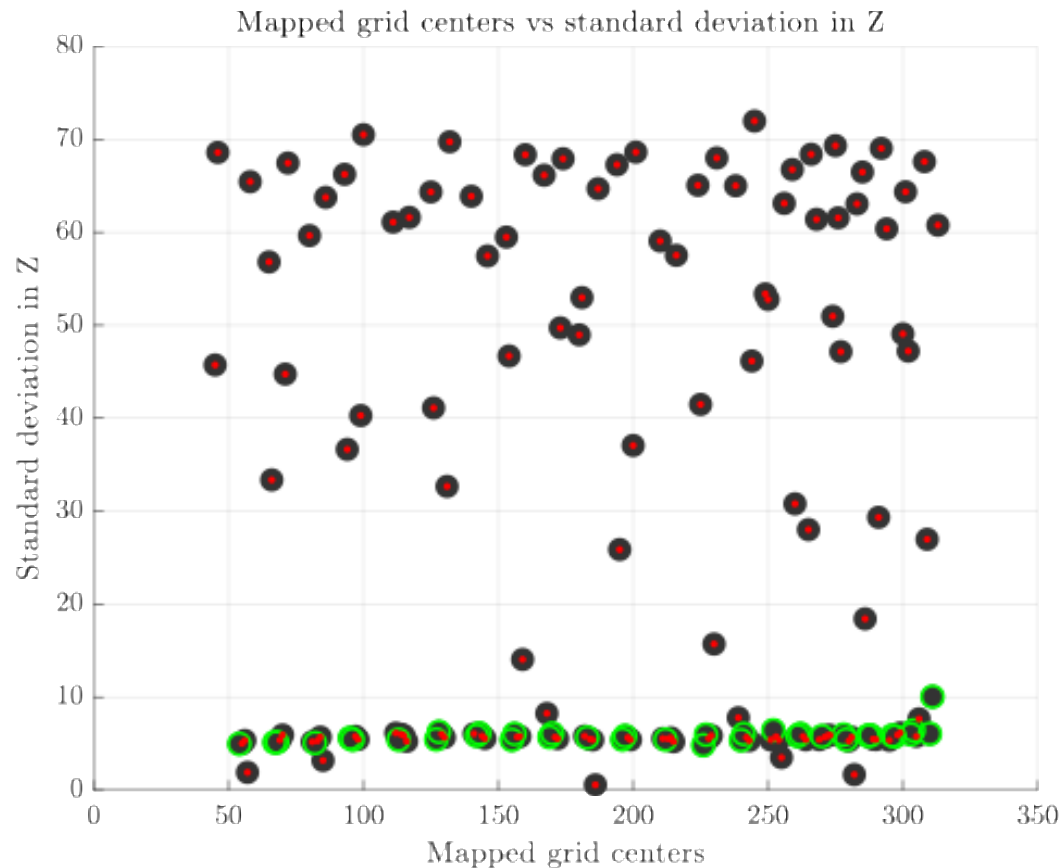fcn_findEdge_findDrivenPathGrids

Grid Voting
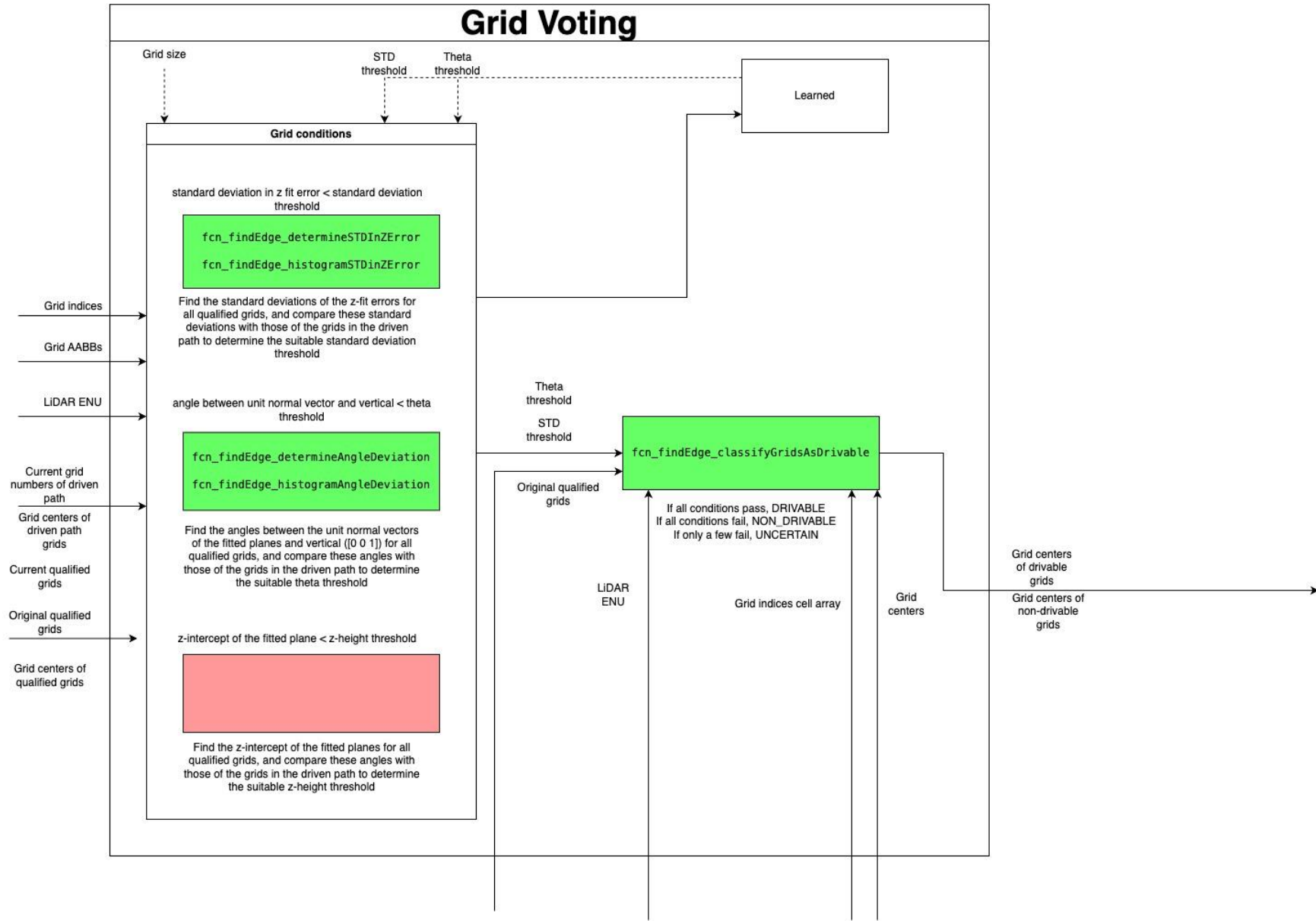
Plane fitting of a driven path grid (standard deviation in z-error = 0.02 m and angle deviation = 5.03 degrees) and a grid with a standard deviation of 0.12 m, with an angle deviation (the angle between the green and unit vector of [0 0 1])) of 3.2 degrees.

Plane fitting of a driven path grid

Plane fitting XYZ Plane fitting X vs Z Plane fitting Y vs Z
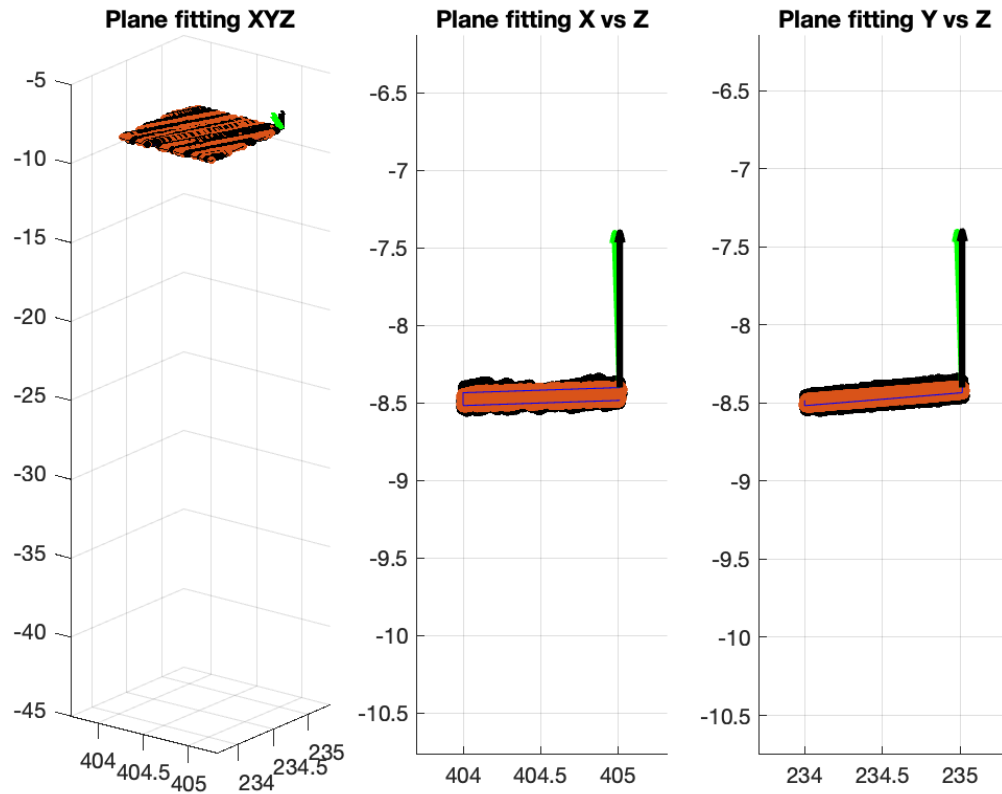
Plane fitting of failed standard deviation grid

Plane fitting XYZ Plane fitting X vs Z Plane fitting Y vs Z
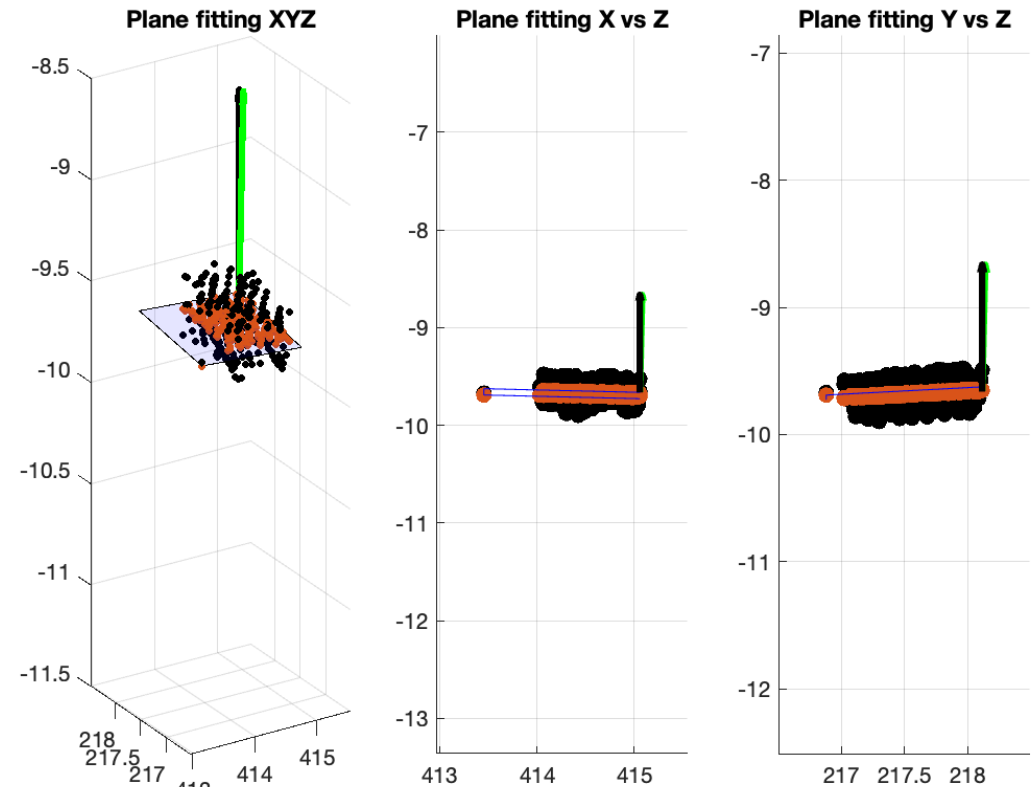
fcn_findEdge_determineSTDInZError

fcn_findEdge_histogramSTDInZError

Grid Condition: Find the standard deviations of the z-errors for all qualified grids by fitting a plane and compare these standard deviations with those of the grids in the driven path to determine a suitable standard deviation threshold. This comparison has been done for multiple datasets with the same grid size.

The orange and blue bins correspond to the standard deviations of the driven path and all the grids, respectively.

Initially, the standard deviation threshold is selected by choosing 3-5 standard deviations above the mean of the standard deviations of the driven path grids. Later, the threshold is slightly adjusted manually to find the most suitable value.
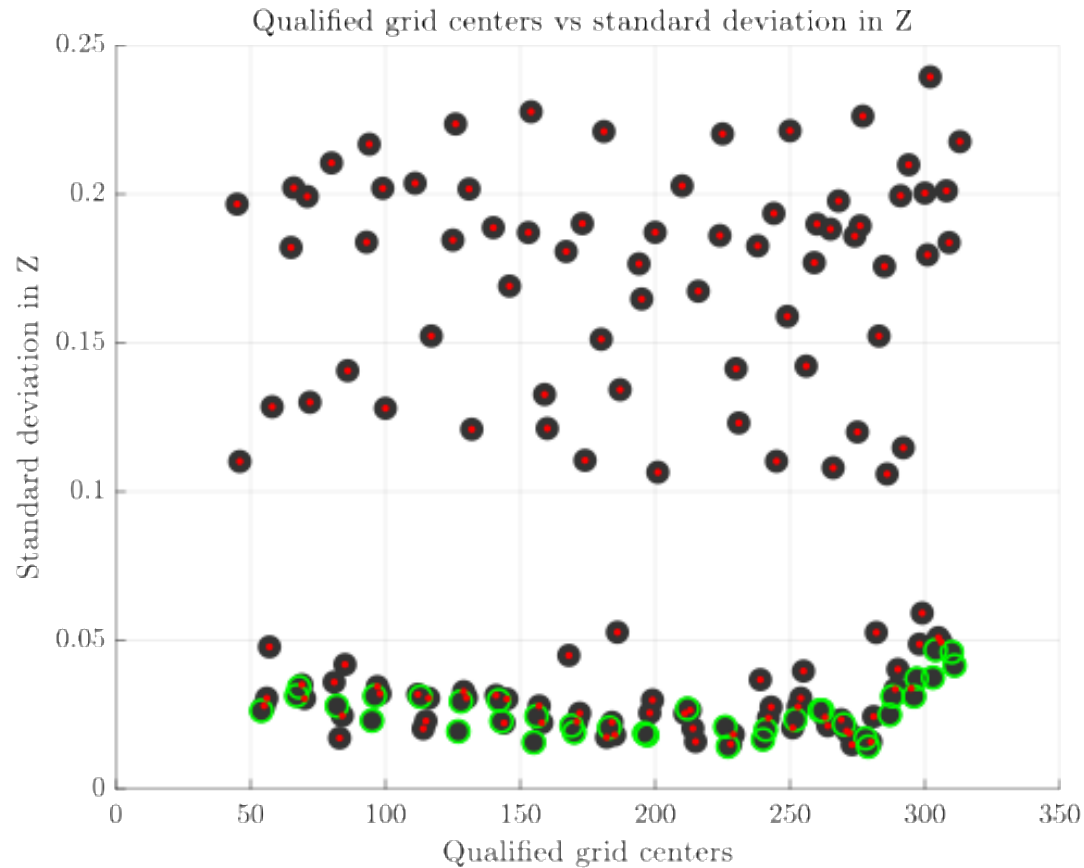
The standard deviations enclosed in green circles are the standard deviations of the driven path
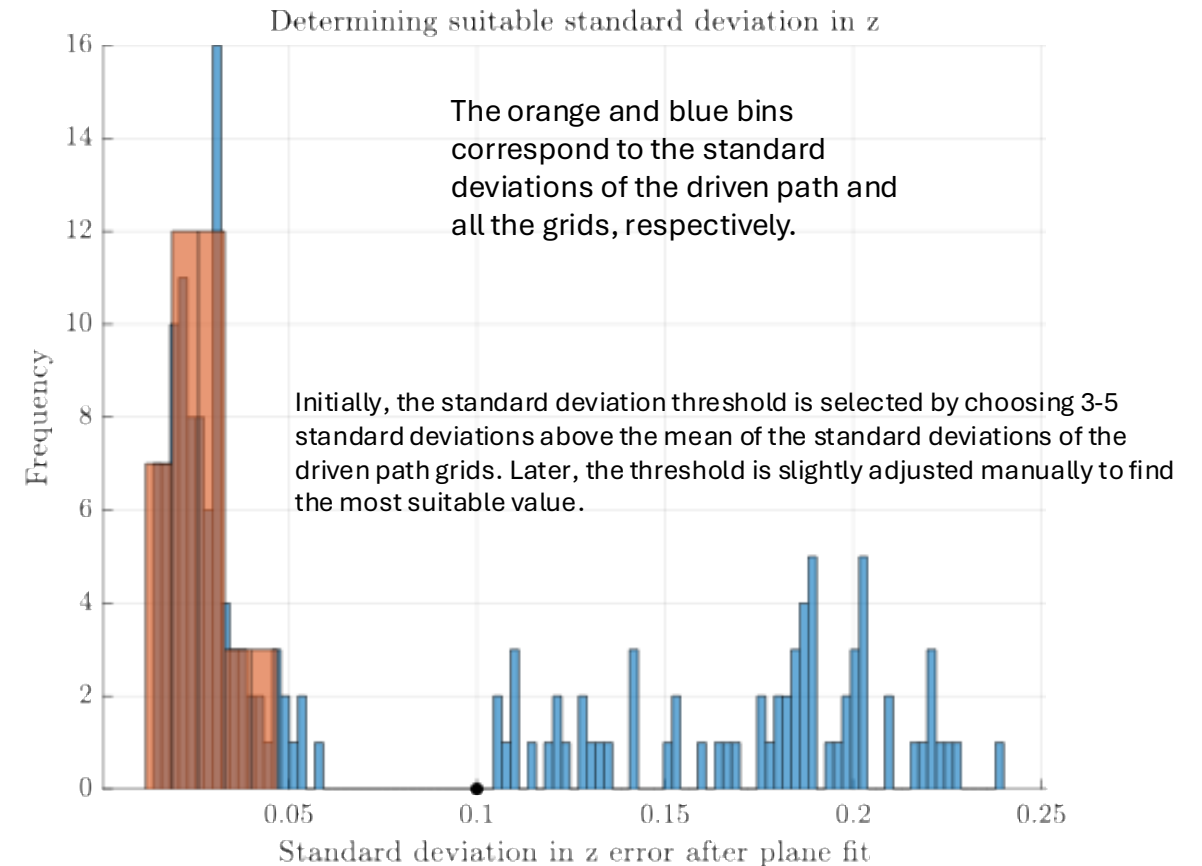
fcn_findEdge_determineSTDInZError

fcn_findEdge_histogramSTDinZError

Plane fitting of a driven path grid (standard deviation in z-error = 0.02 m and angle deviation = 5.03 degrees) and a grid with a standard deviation of 0.07 m, with an angle deviation of 10.29 degrees (the angle between the green vector and the unit vector [0 0 1]).
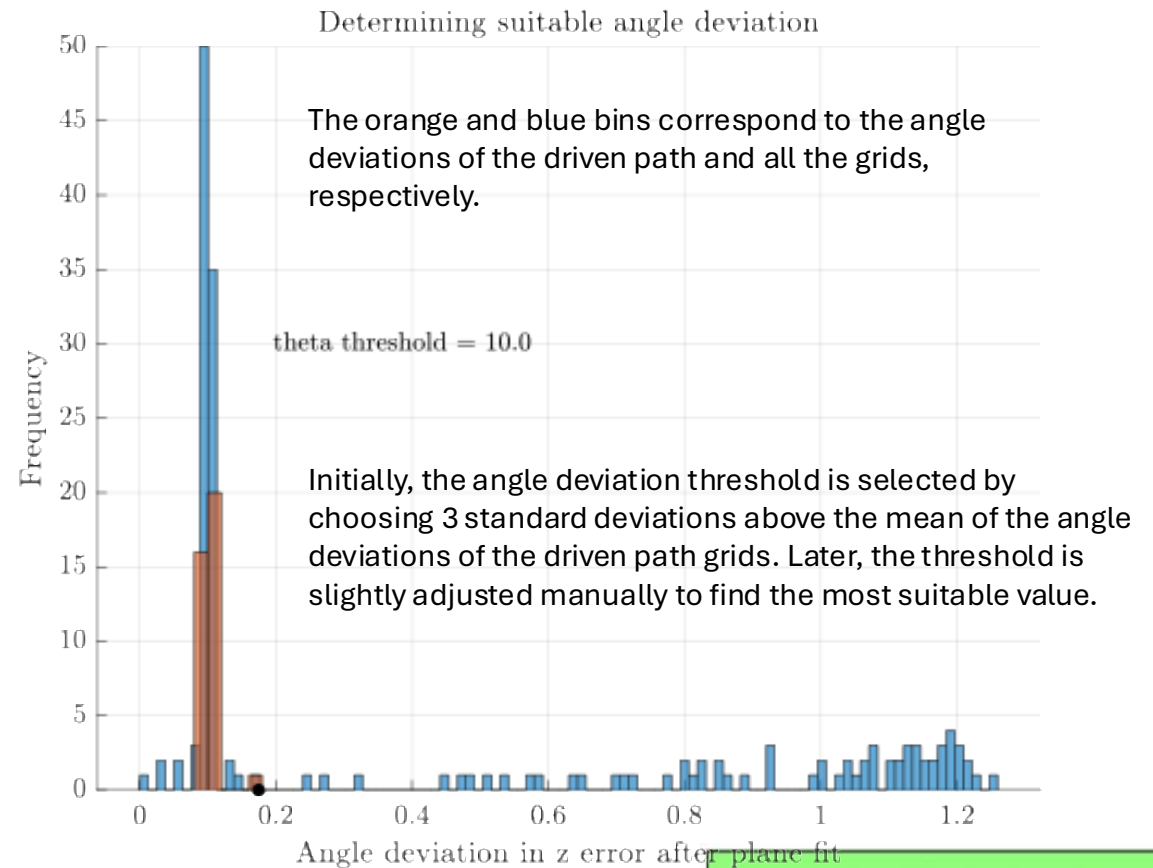

Plane fitting of a driven path grid

Plane fitting XYZ

Plane fitting X vs Z

Plane fitting Y vs Z

Plane fitting of a failed angle deviation grid

Plane fitting XYZ

Plane fitting X vs Z

Plane fitting Y vs Z
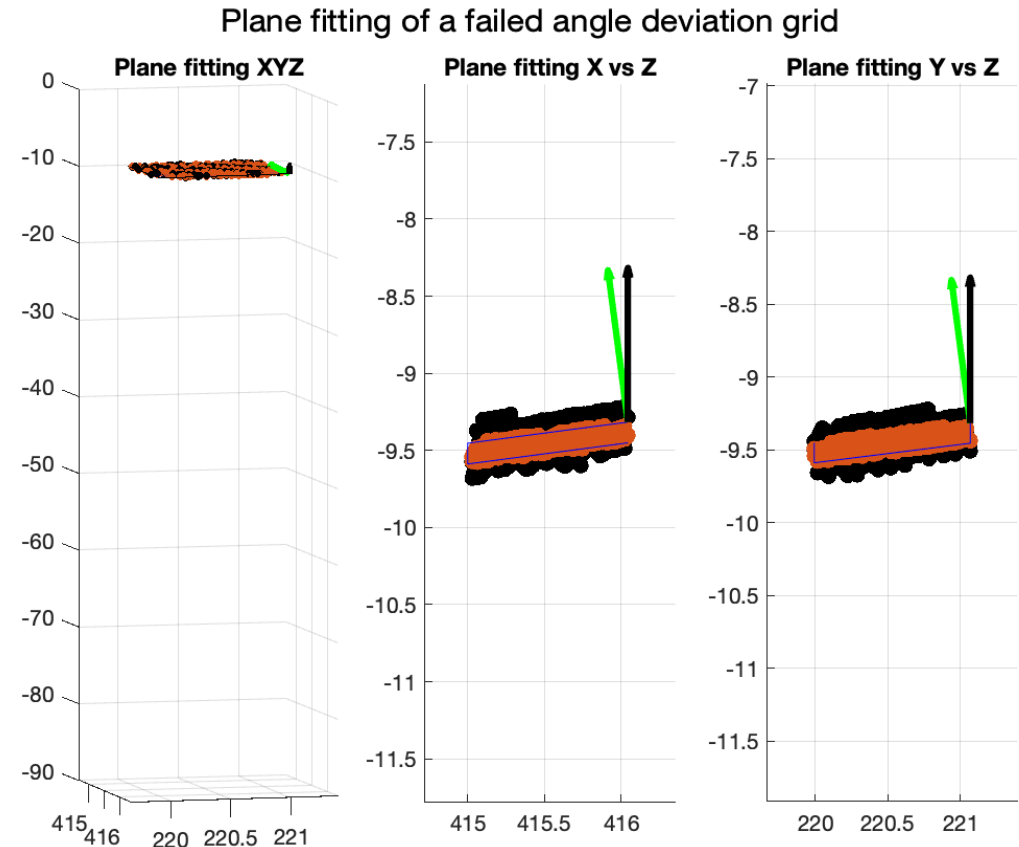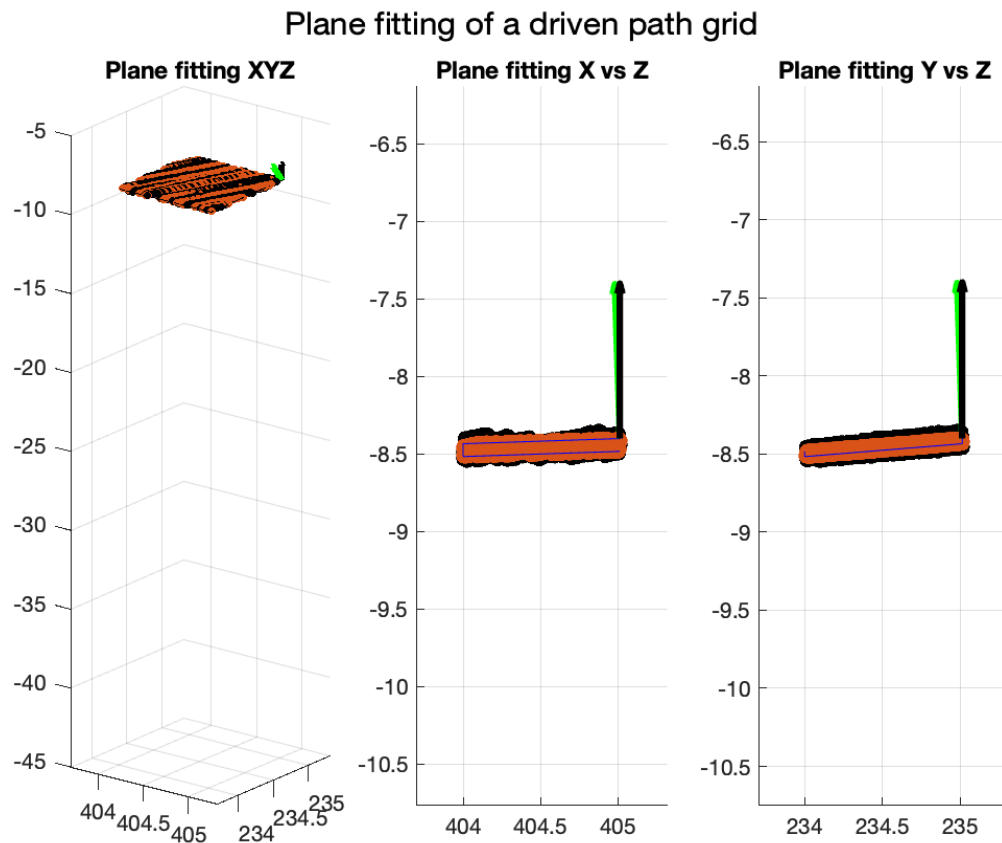
fcn_findEdge_determineAngleDeviation

fcn_findEdge_histogramAngleDeviation

Grid Condition: Find the angles between the unit normal vectors of the fitted planes and the vertical vector ([0 0 1]) for qualified grids and compare these angles with those of the grids in the driven path to determine a suitable theta threshold. This comparison has been conducted for multiple datasets with the same grid size.

The angle deviations enclosed in green circles are the angle deviations of the driven path.

The orange and blue bins correspond to the angle deviations of the driven path and all the grids, respectively.

theta threshold = 10.0

Initially, the angle deviation threshold is selected by choosing 3 standard deviations above the mean of the angle deviations of the driven path grids. Later, the threshold is slightly adjusted manually to find the most suitable value.

```
fcn_findEdge_determineAngleDeviation
fcn_findEdge_histogramAngleDeviation
```

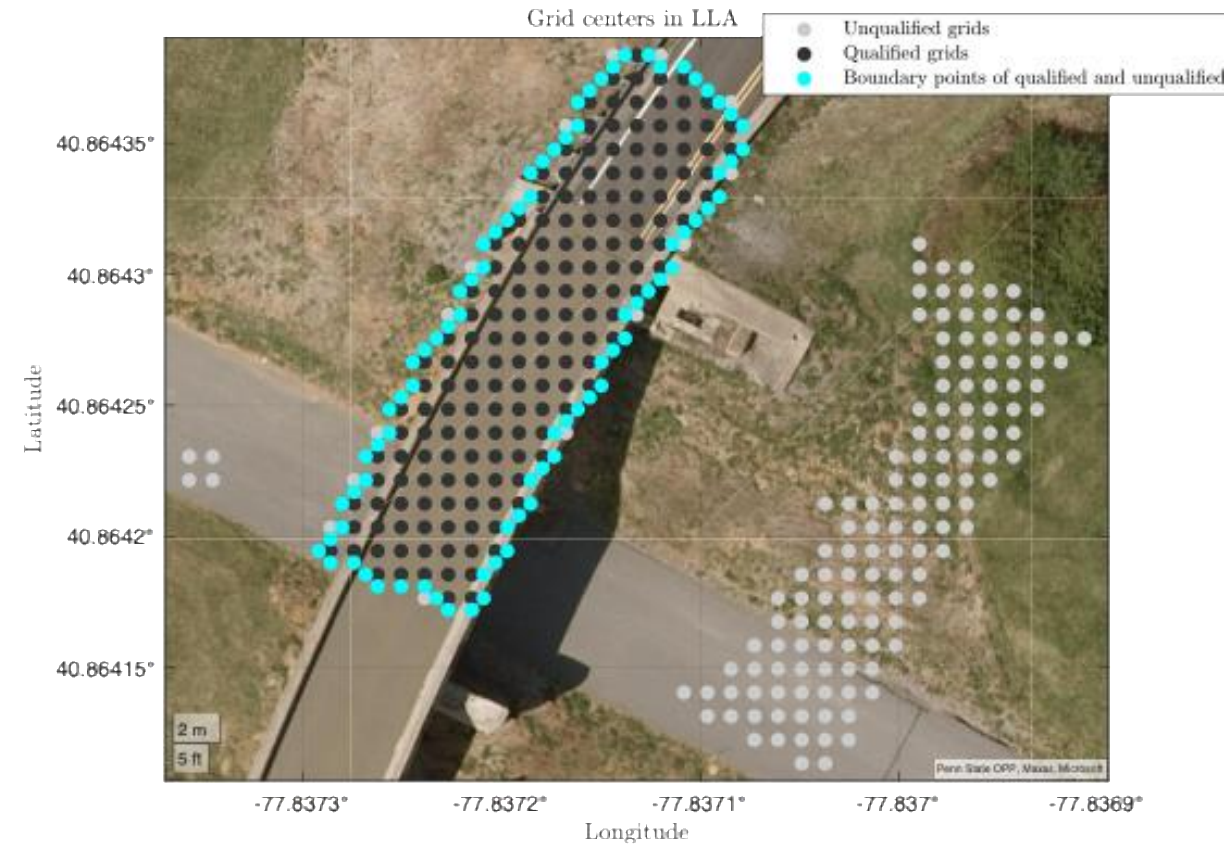# Voting: The qualified grids are classified as drivable, non-drivable, and uncertain grids.
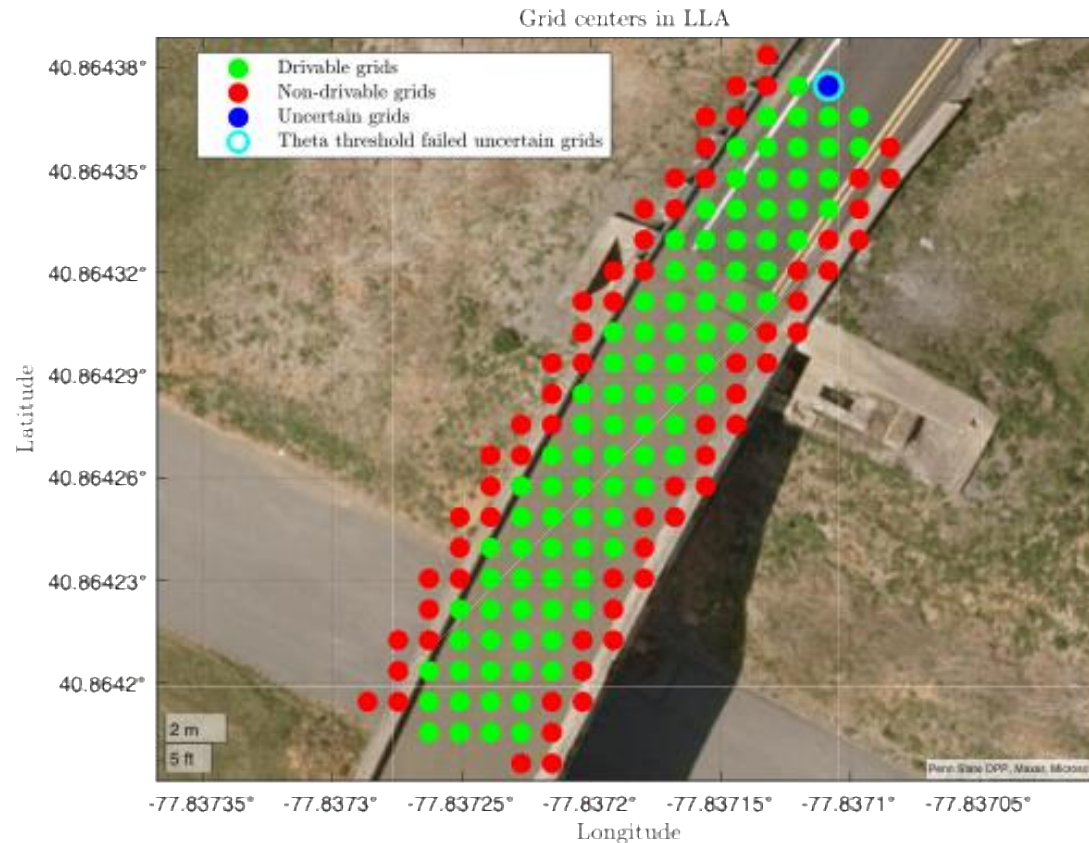
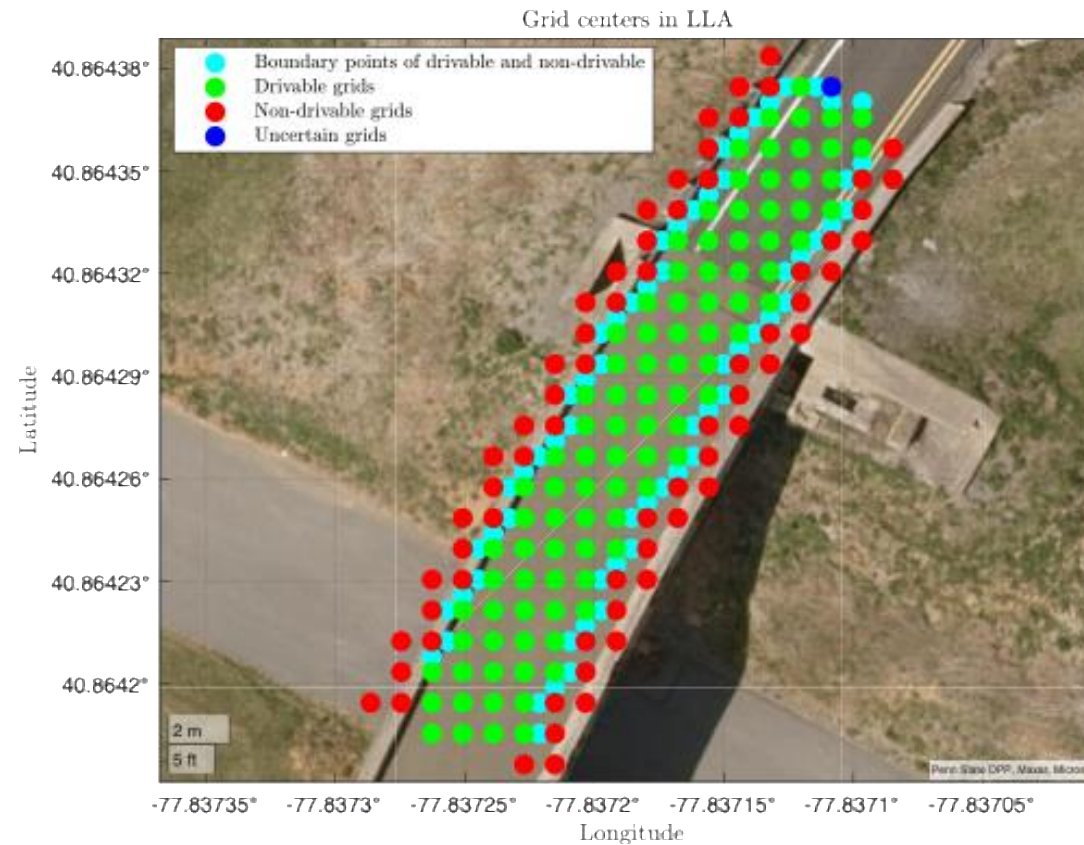

`fcn_findEdge_classifyGridsAsDrivable`

Post Processing
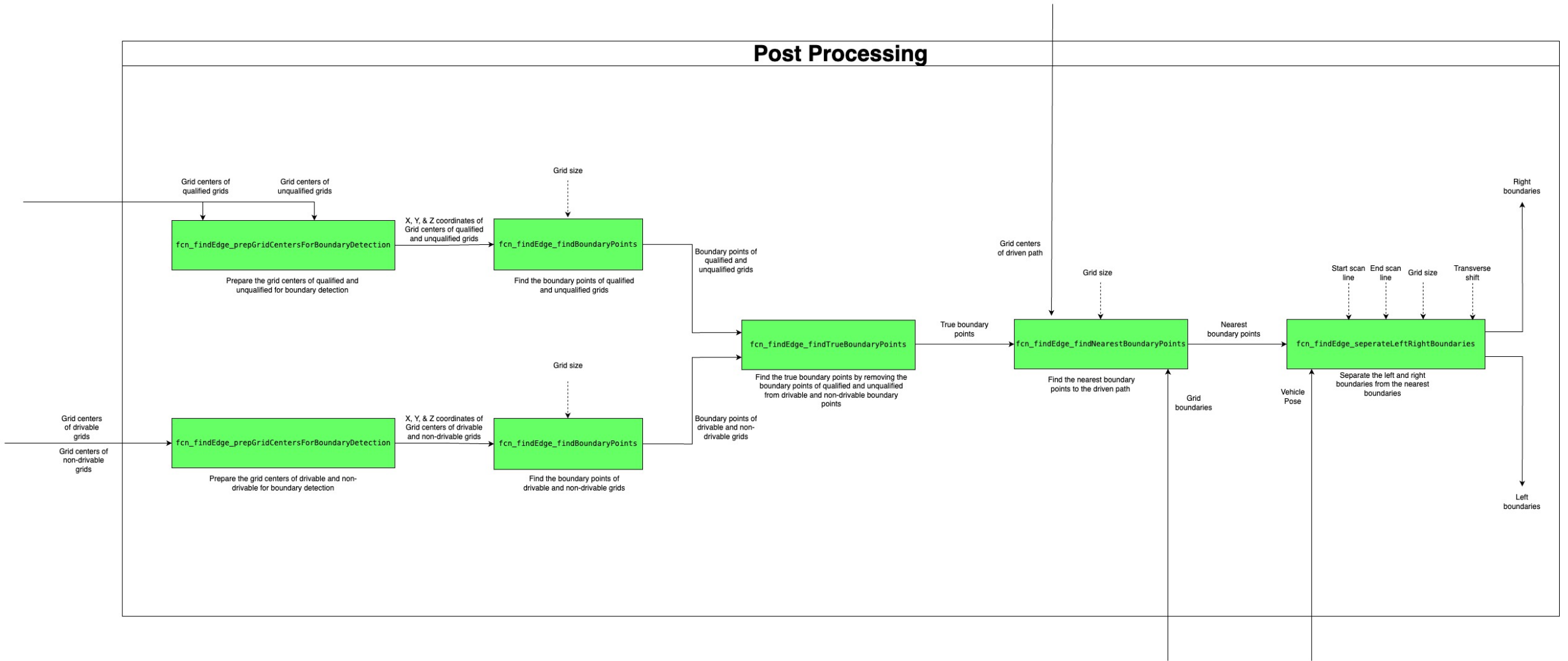
The boundary points of qualified and unqualified grids

fcn_findEdge_prepGridCentersForBoundaryDetection

fcn_findEdge_findBoundaryPoints
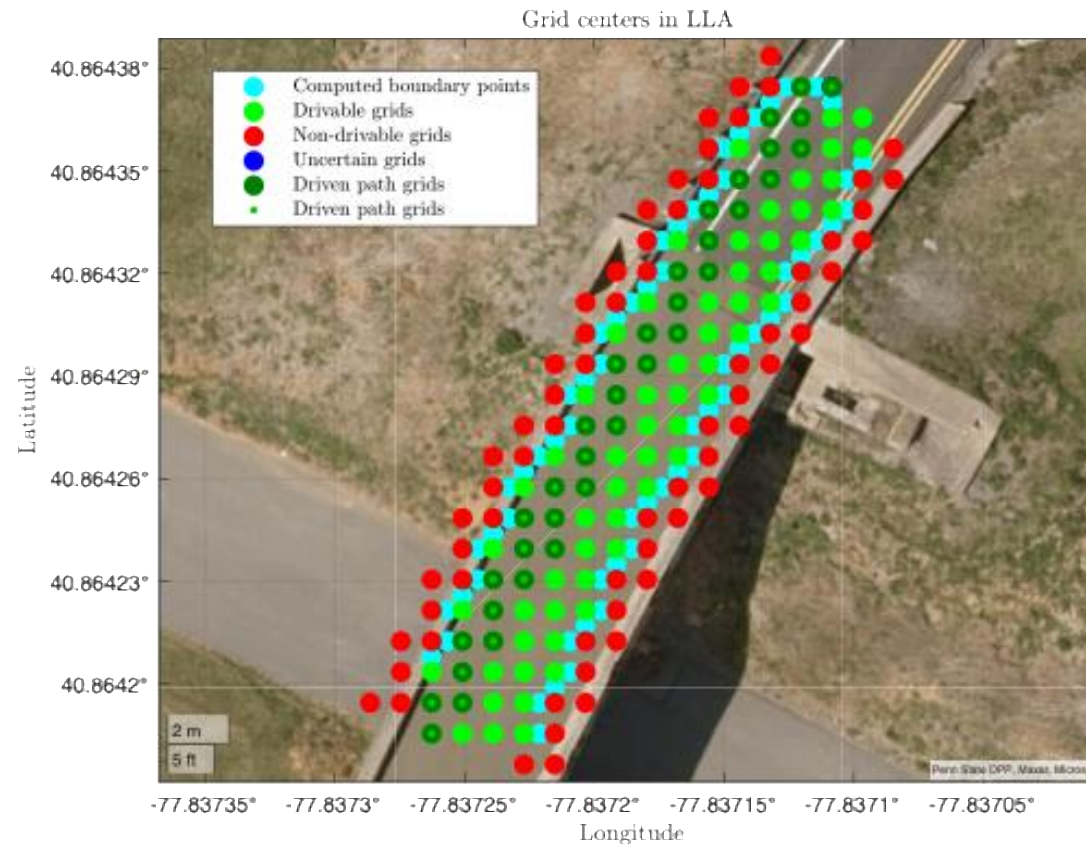
The boundary points of drivable and non-drivable grids

Grid centers in LLA

`fcn_findEdge_prepGridCentersForBoundaryDetection`
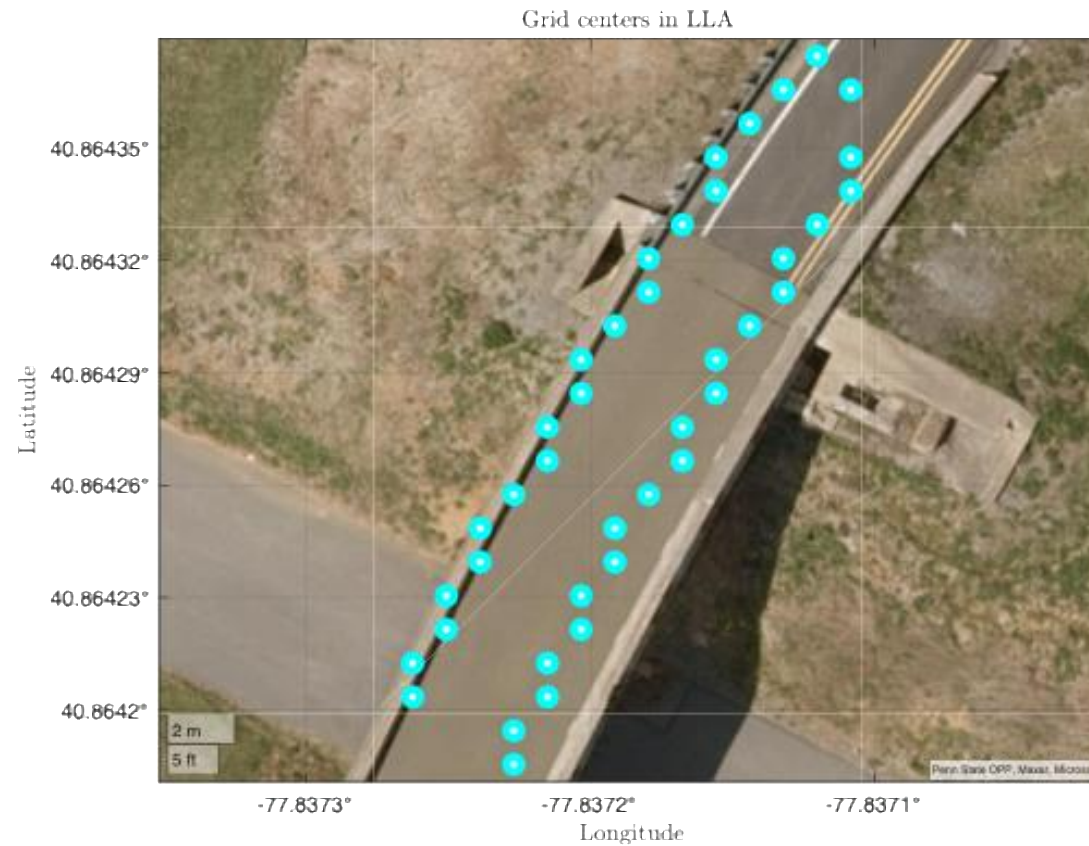
`fcn_findEdge_findBoundaryPoints`

The boundary points of qualified and unqualified are removed from the boundary points of drivable and non-drivable to find the true boundary points

fcn_findEdge_findTrueBoundaryPoints

The nearest boundary points are computed from the true boundary points

fcn_findEdge_findNearestBoundaryPoints

The nearest boundary points are separated into left and right boundary points. The right boundary points separate the pavement from the vegetation