# CE676 Laser Scanning and Photogrammetry

# Project Report

**Name: Tenzing Pema Thungon**

**Roll No.: 231030063**
**(Group5)**

**Email: tenzing23@iitk.ac.in**

Project Report CE676A

# "Quality Assessment of Aerial LiDAR Data for IIT Kanpur Campus: A Comprehensive Analysis"

Tenzing Pema Thungon
231030063
GEO-INFORMATICS
tenzing23@iitk.ac.in

## 1. Introduction

The Quality Assessment of Aerial LiDAR Data for IIT Kanpur Campus: A Comprehensive Analysis report presents a detailed evaluation of the accuracy, coverage, and distribution of LiDAR (Light Detection and Ranging) data collected for the IIT Kanpur campus. LiDAR technology offers precise and efficient means of capturing three-dimensional data of the Earth's surface, making it invaluable for various applications such as terrain mapping, urban planning, and environmental monitoring. In this study, we conducted a thorough assessment of the LiDAR data to ensure its reliability and suitability for use in research, planning, and decision-making processes. The assessment encompassed several key aspects, including vertical accuracy, horizontal accuracy, relative accuracy, data density, Nominal Pulse Spacing (NPS), identification of data voids, and spatial distribution. By examining these factors, we aimed to provide insights into the overall quality and integrity of the LiDAR dataset.

## 2. Objectives

- Determine the total number of LiDAR returns in the data.
- Calculate the maximum, minimum, and average overlap between the two flight lines.
- Measure the absolute vertical accuracy of the LiDAR data
- Calculate the absolute planimetric accuracy of the LiDAR data.
- Evaluate the relative accuracy within swath and overlap areas.
- Calculate the Nominal Pulse Spacing (NPS) and data density.
- Detect and analyze any data voids or gaps in the LiDAR data
- Determine if the LiDAR data exhibits uniform distribution in planimetry.

## 3. Methodology

### 3.1 Quantify LiDAR Returns

LiDAR returns refer to the number of laser pulses emitted by a LiDAR system and the subsequent number of returns detected by the system for each emitted pulse. Each laser pulse can result in multiple returns as it interacts with different objects and surfaces within its footprint. The number of returns provides valuable information about the characteristics of the terrain, vegetation, and structures within the LiDAR data. LiDAR returns help assess the density and richness of the data, providing insights into the complexity of the surveyed area. They are crucial for understanding the vertical structure of the landscape, including ground elevation, vegetation canopy, and building heights.

**LiDAR returns- How many returns are there in the data:**

**Case 1:**
To obtain the number of returns in the data, the following algorithm was implemented:

- The CloudCompare software was opened with the provided LAS data to get standard fields like intensity, return number, scan direction, etc.
- The software was used to show and modify the flight lines. After segmenting the data, target areas were chosen, and the file was saved as a CSV.

- To find out how many LiDAR returns there were for each pulse, the header information from the CSV file was read using MATLAB software.

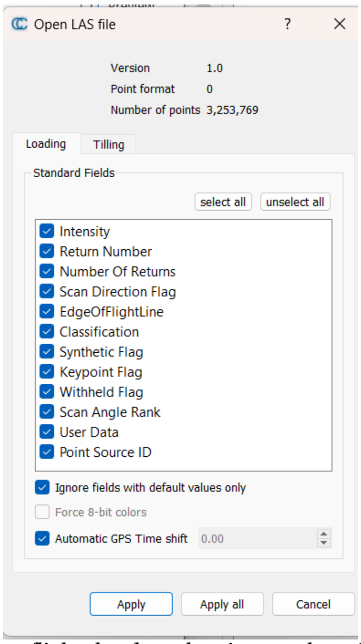



*Figure 1: flight1 las data showing number of points*

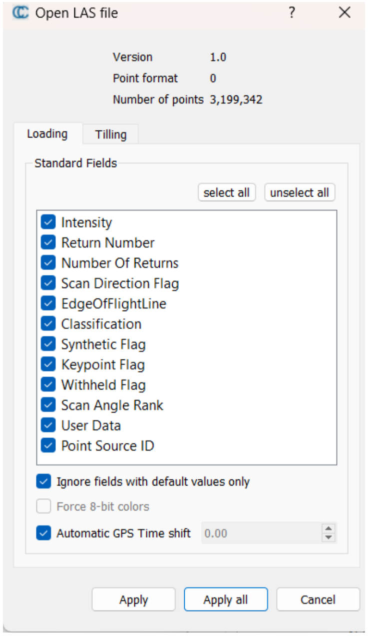



*Figure 2: flight2 las data 38 showing number of points*

- The number of returns per pulse was retrieved as an integer column.
- The extracted column contained unique integers that were identified, indicating that the LiDAR system detected several returns for each pulse.
- The maximum number of returns per pulse was determined, was found to be 3.

**Case 2:** QGIS tool "lasinfo"

To verify the LiDAR returns, the LAS file was opened in QGIS tool lasinfo, which provided various information, including the number of returns per pulse he file type, version, and intensity etc:

- The LAS file contained a total of 1-3 number of returns per pulse.

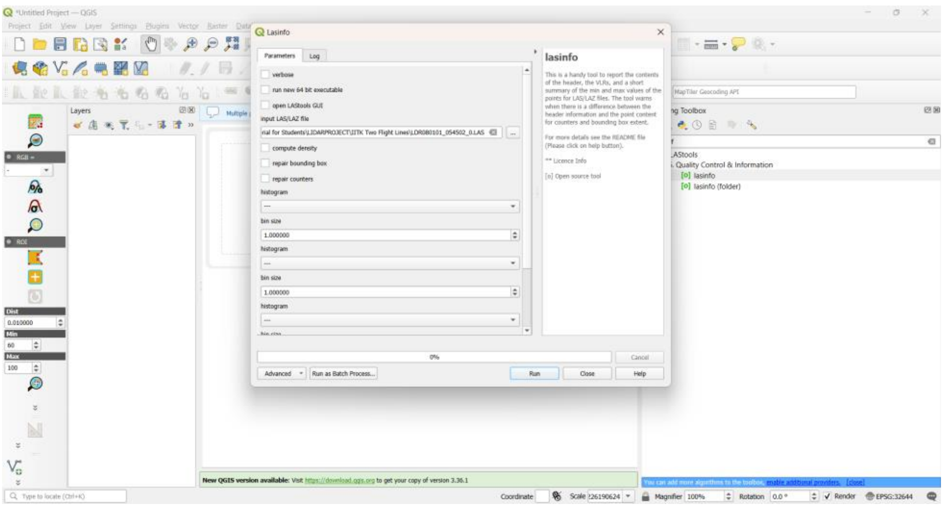
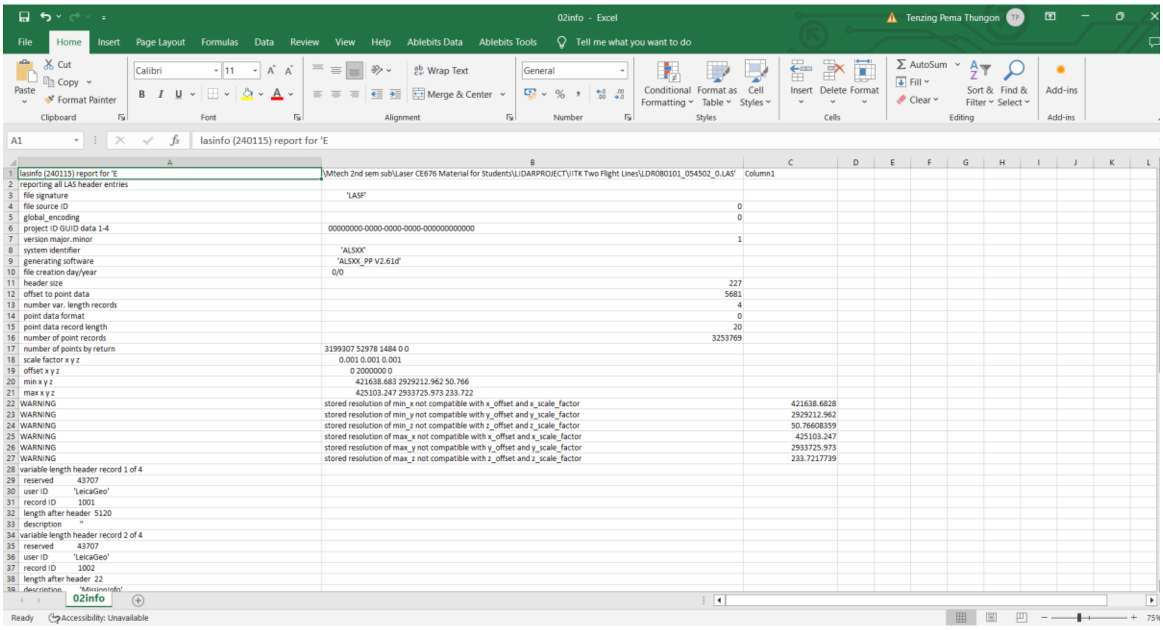



*Figure 3: Using lasinfo in QGIS to generate given LAS file data*



| | A | B | C |
|---|---|---|---|
| 1 | lasinfo (240115) report for 'E | \Mtech 2nd sem sub\Laser CE676 Material for Students\LIDARPROJECT\IITK Two Flight Lines\LDR080101_054502_0.LAS' | Column1 |
| 2 | reporting all LAS header entries | | |
| 3 | file signature | 'LASF' | |
| 4 | file source ID | 0 | |
| 5 | global_encoding | 0 | |
| 6 | project ID GUID data 1-4 | 00000000-0000-0000-0000-000000000000 | |
| 7 | version major.minor | 1 | |
| 8 | system identifier | 'ALSXX' | |
| 9 | generating software | 'ALSXX_PP V2.61d' | |
| 10 | file creation day/year | 0/0 | |
| 11 | header size | 227 | |
| 12 | offset to point data | 5681 | |
| 13 | number var. length records | 4 | |
| 14 | point data format | 0 | |
| 15 | point data record length | 20 | |
| 16 | number of point records | 3253769 | |
| 17 | number of points by return | 3199307 52978 1484 0 0 | |
| 18 | scale factor x y z | 0.001 0.001 0.001 | |
| 19 | offset x y z | 0 2000000 0 | |
| 20 | min x y z | 421638.683 2929212.962 50.766 | |
| 21 | max x y z | 425103.247 2933725.973 233.722 | |
| 22 | WARNING | stored resolution of min_x not compatible with x_offset and x_scale_factor | 421638.6828 |
| 23 | WARNING | stored resolution of min_y not compatible with y_offset and y_scale_factor | 2929212.962 |
| 24 | WARNING | stored resolution of min_z not compatible with z_offset and z_scale_factor | 50.76608359 |
| 25 | WARNING | stored resolution of max_x not compatible with x_offset and x_scale_factor | 425103.247 |
| 26 | WARNING | stored resolution of max_y not compatible with y_offset and y_scale_factor | 2933725.973 |
| 27 | WARNING | stored resolution of max_z not compatible with z_offset and z_scale_factor | 233.7217739 |
| 28 | variable length header record 1 of 4 | | |
| 29 | reserved 43707 | | |
| 30 | user ID 'LeicaGeo' | | |
| 31 | record ID 1001 | | |
| 32 | length after header 5120 | | |
| 33 | description '' | | |
| 34 | variable length header record 2 of 4 | | |
| 35 | reserved 43707 | | |
| 36 | user ID 'LeicaGeo' | | |
| 37 | record ID 1002 | | |
| 38 | length after header 22 | | |
| 39 | description 'MissionInfo' | | |

Figure 4: Info generated using QGIS lasinfo tool

The indication of 1-3 LiDAR returns per pulse suggests that the LiDAR system detected multiple returns for each emitted laser pulse. This indicates that the LiDAR data captures detailed information about the terrain and objects within the surveyed area, including ground surfaces, vegetation, and structures.

## 3.2 Evaluate LiDAR Overlap

LiDAR overlap refers to the extent to which two or more LiDAR flight lines cover the same area on the ground. It represents the redundancy in data acquisition and is crucial for ensuring the completeness and accuracy of the LiDAR dataset. Overlap is necessary to account for variations in terrain, vegetation, and other features that may affect the quality and reliability of the LiDAR data.

**LiDAR overlap - What is the maximum, minimum, and average overlap in the two flight lines?**

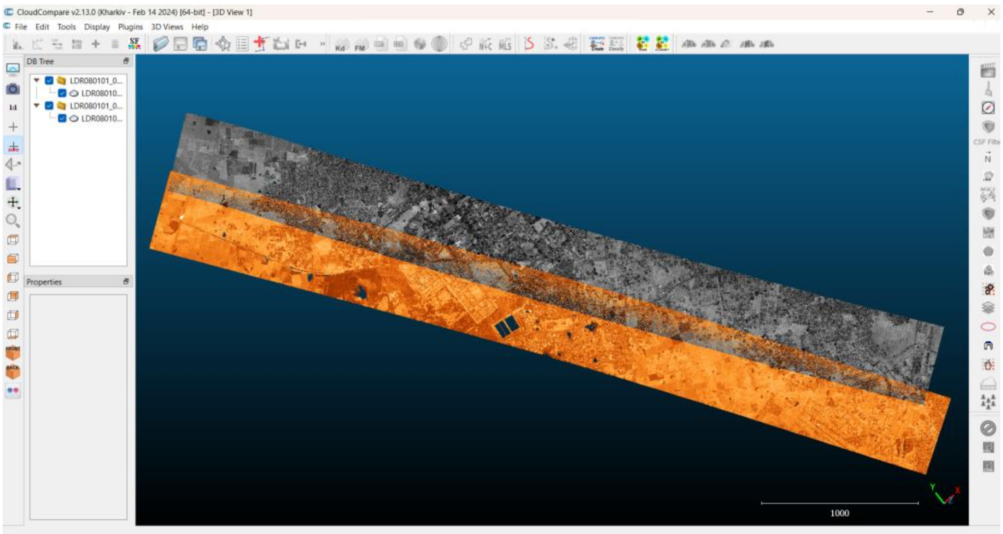



Figure 5: Showing the given LAS file in Cloudcompare

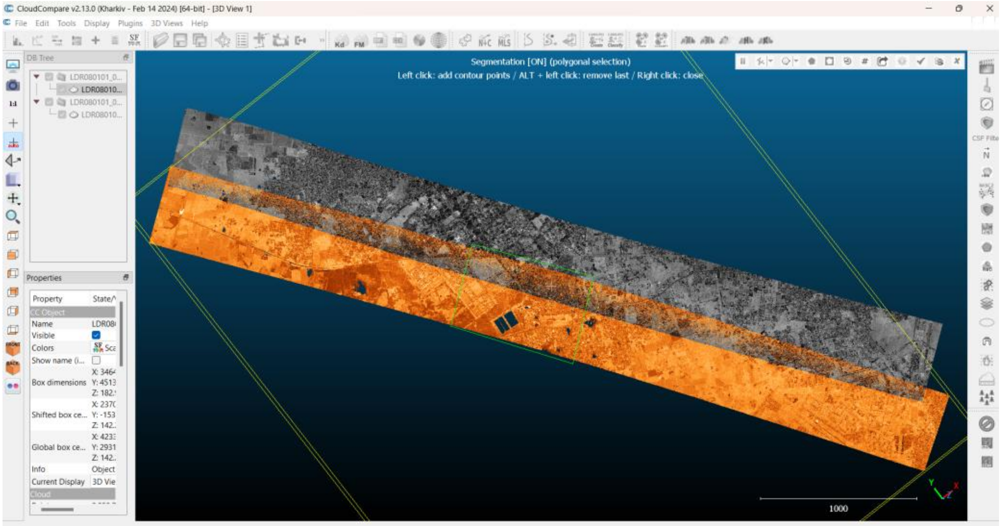



*Figure 6: img showing max overlaying area, being segmented in cloudcompare*

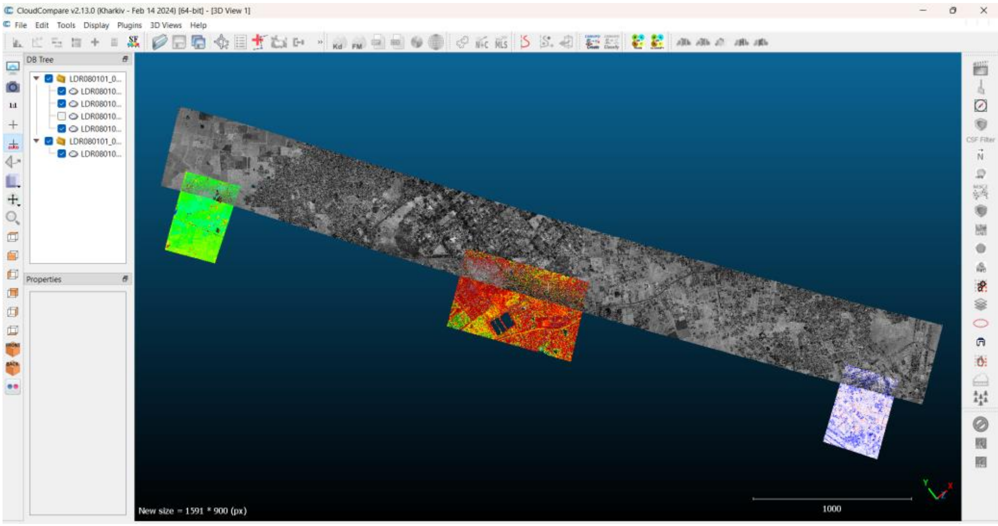



*Figure 7: test sites considered to compute Overlap*

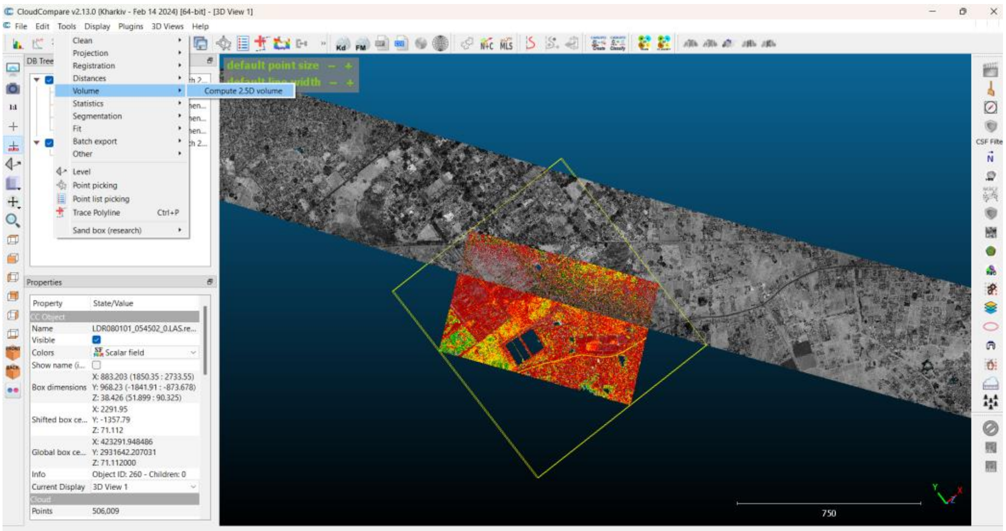



*Figure 8: Surface area computed using Volume tool*

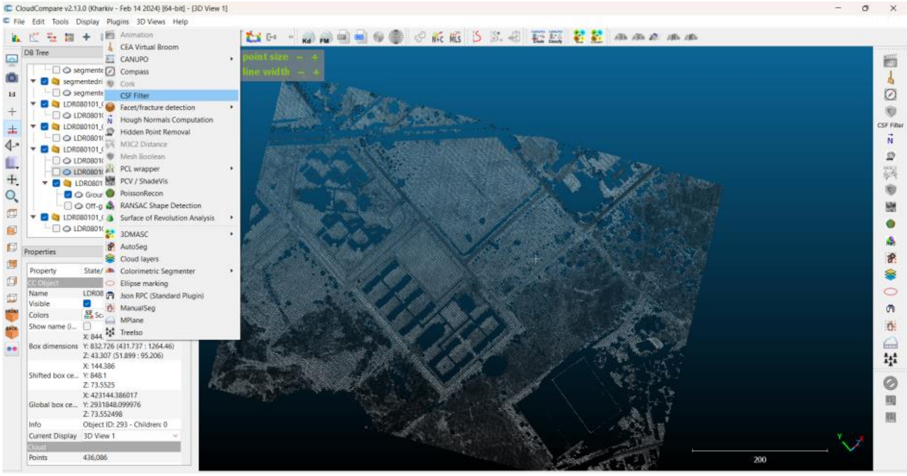



*Figure 9: Using csf to generate dem*

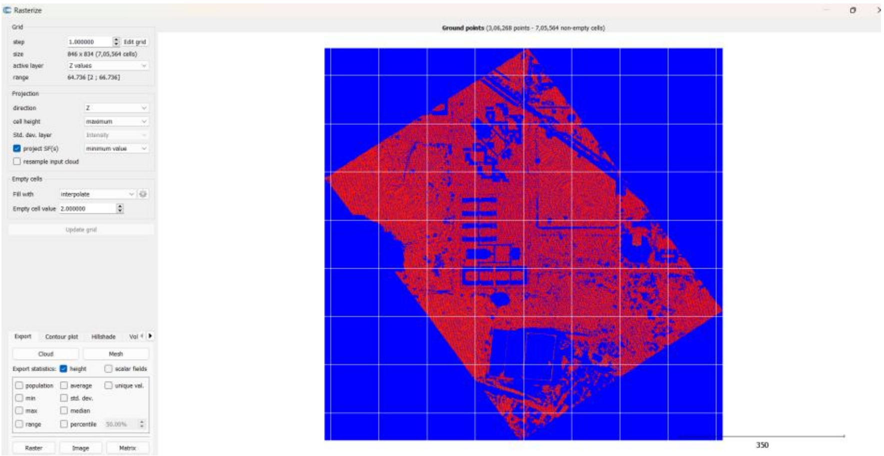



*Figure 10: Rasterize the ground points*

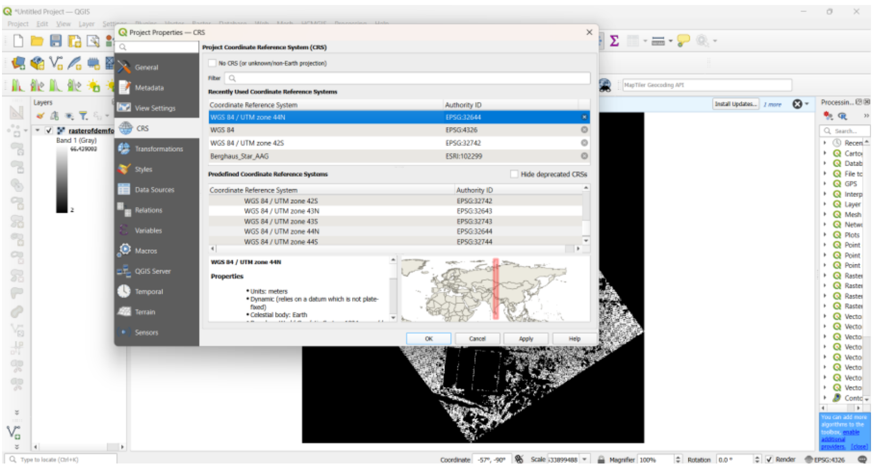



*Figure 11: give the CRS as WGS 84*

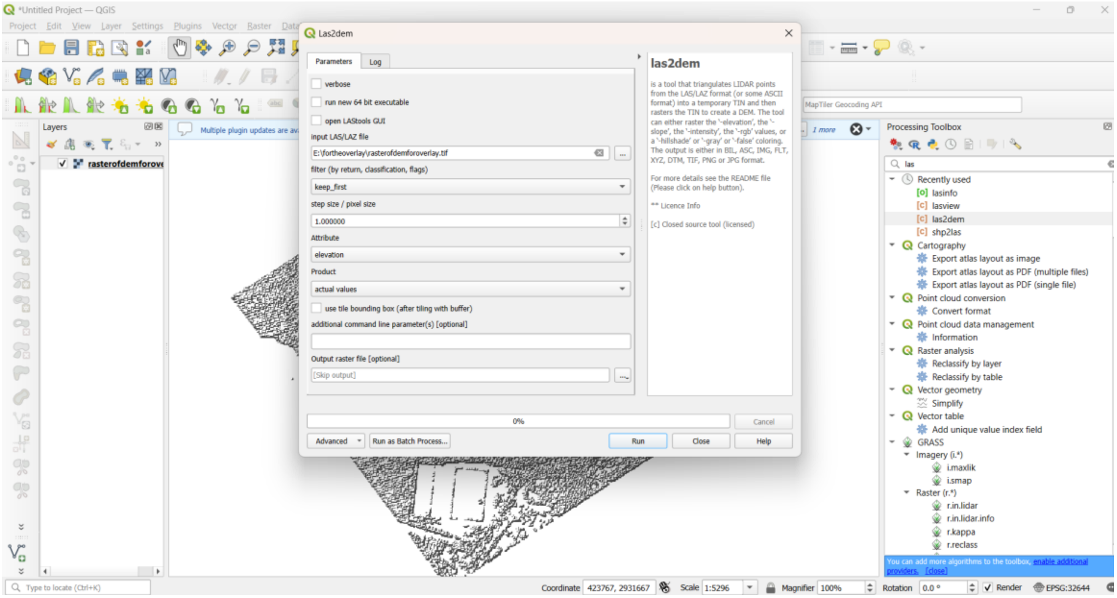


*Figure 12: Las2dem only for 1st return*

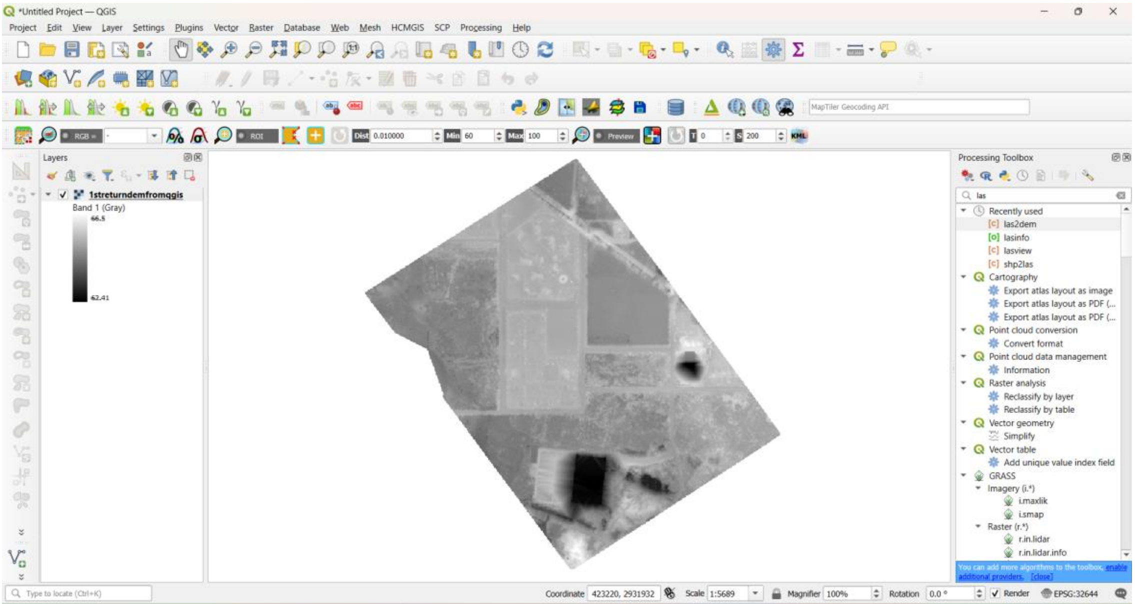


*Figure 13: Dem of 1st return generated using las2dem in Qgis*

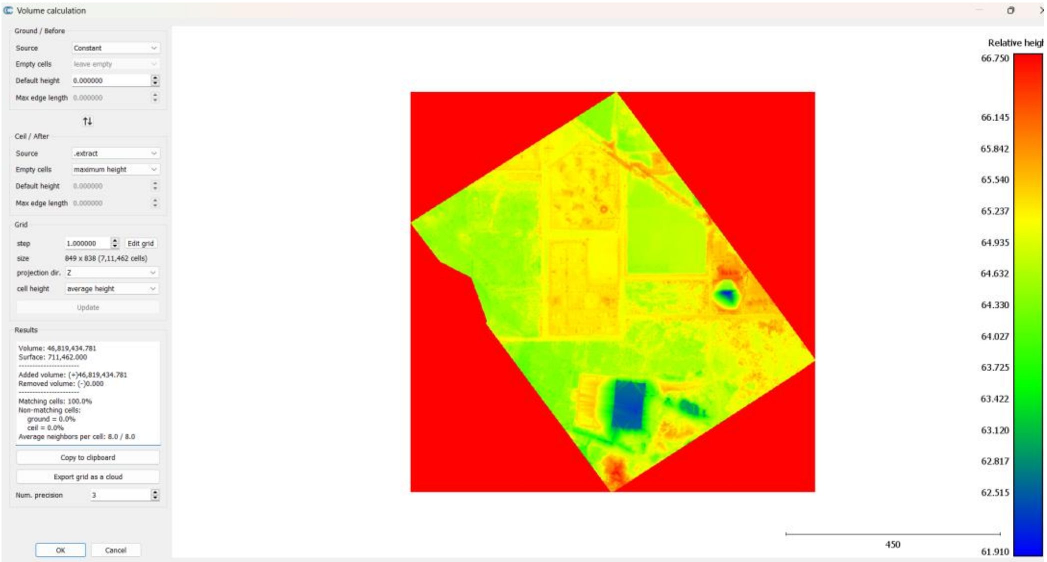



*Figure 14: Use las tool generate to find the surface area of 1st return*

**Procedure:**

- **Visualization and Test Area Selection:**
  - Visualize the LiDAR data for both flight lines and identify suitable test areas for segmentation.
  - Define the maximum and minimum extents of the test areas to establish the boundaries for overlap assessment.
- **Generation of Digital Elevation Models (DEMs):**
  - Utilize CloudCompare and QGIS software to generate Digital Elevation Models (DEMs) based on the LiDAR data.
  - Focus on extracting DEMs for the first return data, which represent the initial interaction of the LiDAR pulses with the terrain surface.
- **Identification of Overlapping and Non-Overlapping Areas:**
  - Segment the DEMs to delineate the overlapping and non-overlapping regions between the two flight lines.
  - Calculate the areas of the overlapping and non-overlapping regions to quantify their spatial extents.
- **Determination of Percentage Overlap:**
  - Find unique points within the overlap area for each flight line using MATLAB.
  - Compute the maximum, minimum, and average overlap percentages based on the percentages obtained from both flight lines.

Based on the analysis conducted using the provided MATLAB code, the following results were obtained:

- **Minimum Overlap:** 17.5903%
- **Maximum Overlap:** 19.2237%
- **Average Overlap:** 18.407%

These results indicate the percentage of overlap between the two flight lines over the specified test areas.

## 3.3 Assess Altimetric (vertical) Accuracy

Altimetric accuracy refers to the precision and reliability of elevation measurements obtained from LiDAR data. Absolute vertical accuracy specifically quantifies the agreement between LiDAR-derived elevation values and accurately measured ground control points (GCPs) in non-vegetated areas.

**Altimetric (vertical) accuracy – What is absolute vertical accuracy?**

**Case 1: Non vegetated Area**

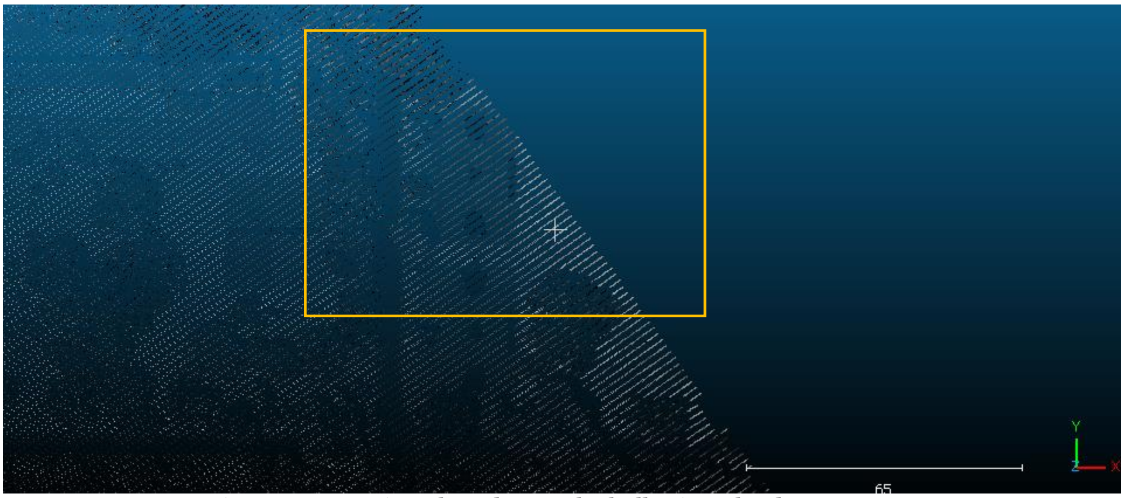


*Figure 15: Test site selected as Basketball using CloudCompare*

**Procedure for Determining Absolute Vertical Accuracy:**

1. Ground Truthing with GCPs:
   - Ground control points (GCPs) are established in the field using high-precision GNSS instruments to obtain accurate ground measurements.
   - These GCPs serve as reference points for validating the elevation values derived from LiDAR data.
2. TIN Interpolation:
   - The GCPs' easting and northing coordinates are used to generate a Triangulated Irregular Network (TIN), representing the bare earth surface.
   - TIN interpolation is employed to estimate elevation values for the LiDAR points at the same locations as the GCPs.
3. Calculation of Height Differences:
   - The LiDAR-interpolated elevations are compared with the measured GCP elevations to determine the height differences between the two datasets.
4. Compute Root Mean Square Error (RMSE):
   - Calculate the Root Mean Square Error (RMSE) as a measure of the overall discrepancy between the elevations obtained from the GCPs and the TIN.
   - The RMSE is computed as the square root of the sum of the squared height differences divided by the number of data points.
   - This provides a quantitative measure of the vertical discrepancy between the LiDAR data and the ground truth GCPs.
5. Vertical Accuracy Assessment:
   - The RMSE serves as an indicator of the absolute vertical accuracy of the LiDAR data in non-vegetated areas.

- A lower RMSE value indicates higher vertical accuracy, while a higher RMSE value suggests greater discrepancy between the LiDAR-derived elevations and the ground truth.

**Selection Standards for Test Sites:**

- Availability in the Reference Data:
    - Test sites were selected based on their presence in the given IITK LiDAR dataset, ensuring consistency and comparability with the reference data.
- Flat Ground Surface:
    - Selected sites were characterized by flat and even ground surfaces to ensure that the LiDAR data predominantly captured first return signals, providing accurate elevation measurements.
- Absence of Vegetation:
    - Vegetation-free areas were preferred to minimize interference from vegetation and ensure clear visibility of the ground surface, facilitating accurate ground truthing with GCPs.
- Open Sky Accessibility:
    - Test sites were chosen in locations where the sky was unobstructed, allowing for easy setup of GNSS receivers and unhindered reception of satellite signals during GCP data collection.
- Visibility in LiDAR Data and on Ground:
    - The selected test sites were clearly identifiable in both the LiDAR data and on the ground, ensuring accurate alignment between the LiDAR-derived elevations and the measured GCPs.

## Trimble R10 GNSS receiver:

Procedure for GCP Collection for Accuracy Testing: Using Trimble R10 GNSS receiver.

- Setting up the R10 GNSS Receiver:
    - Begin by marking the Control Point (CP) at the specified site with a visible marker, such as a bright paint dot, serving as the Ground Control Point (GCP) reference.
    - Mount the R10 receiver securely onto the bipod head and extend the bipod legs to position the instrument approximately 2 meters above the GCP.
    - Ensure the tripod is leveled by adjusting the legs until the bubble is centered within the circle, indicating a perfectly leveled instrument.
    - Power on the R10 receiver and initialize it to prepare for data collection.


*Figure 16: Front and back panel of R10 Receiver*

*Figure 17: img of the Test site for NVA*

- Trimble TSC7 Controller Setup:
    - The Trimble TSC7 controller is paired with the R10 GNSS receiver to facilitate data collection.
    - Initialize Trimble Access software on the TSC7 controller and create a new project group with a meaningful name, ensuring seamless communication with the RTK base station via the VPN network.
    - Configure the project settings, including the coordinate system (World Wide UTM), zone (44N), and local datum (WGS-1984 with 7 parameters), setting the project height distinct from the antenna height.
    - Define the survey style as "RTK" for precise static positioning and select "Rover" as the receiver role.
    - Specify survey type as "RTK" and input R10 receiver details, including antenna type, measured height, and total antenna height.
    - Confirm successful connection to the RTK base station through the VPN network, ensuring signal strength and data quality indicators are optimal.

- RTK Survey Execution:
    - Activate the "GNSS" function in the instrument settings and configure the R10 receiver's "Rover Mode" for customizable observation times.
    - Connect the controller to the rover and navigate to the "Measure" menu in Trimble Access.
    - Select "RTK" mode and initialize the survey by clicking on "Measure Points."
    - Assign names and codes to measured points, such as "GCP1" or "GCP2," using the "GCP" code for control points.
    - Set observation times based on location and satellite visibility, typically between 2-3 seconds per control point.
    - Capture all required point data using the Trimble TSC7 controller, ensuring accurate and comprehensive data collection for subsequent analysis.

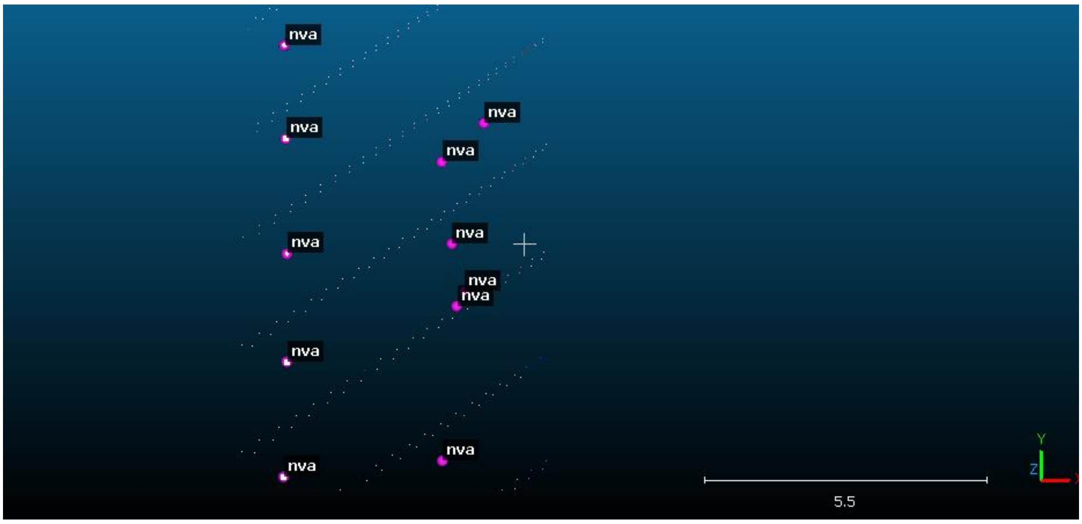


*Figure 18: overlating GCPs on LiDAR data*

## Formulas and functions used in the provided MATLAB code:

- Delaunay Triangulation:
  - Function: delaunayTriangulation
  - Purpose: Generates a Delaunay triangulation from a set of points.
- Scattered Interpolant Creation:
  - Function: scatteredInterpolant
  - Purpose: Creates a scattered interpolant for interpolating values at scattered points in 2D or 3D.
- Data Filtering:
  - Logical Indexing: ~isnan()
  - Purpose: Filters out NaN values and corresponding data.

- Root Mean Square Error (RMSE) Computation:
  - Formula: RMSE

$$\text{Vertical Accuracy} = \sqrt{\frac{\sum_{i=1}^{N}(Z_i - \text{Average Height})^2}{N}}$$

Vertical accuracy in 95% confidence interval:

$$Accuracy_z = 1.96 * RMSE_Z$$

- where $N$ is the total number of points in the patch, $Zi$ represents the elevation of each point, and **Average Height** is the mean height calculated earlier.
  - Function: sqrt, mean
  - Purpose: Computes the RMSE as a measure of vertical accuracy.
- Visualization:
  - Plotting Functions: scatter, triplot, surf
  - Purpose: Visualizes the GCPs, LiDAR points, TIN, and interpolated surface.
- Data Loading:
  - Function: readmatrix
  - Purpose: Reads data from CSV files into MATLAB matrices.
- Convex Hull Check:
  - Function: inpolygon
  - Purpose: Checks whether points are inside the convex hull of a polygon.

**Data Analysis and Comparison:**

- Similar analyses were conducted for each test site, allowing for a comprehensive evaluation of the LiDAR data's vertical accuracy across various terrains.

**Analysis Using MATLAB: TIN Interpolation of the collected GCP's:**

- After GCP data collection, TIN interpolation is performed using MATLAB to generate a triangulated irregular network (TIN) representing the bare earth surface.
- Easting and northing coordinates of the GCPs are used to interpolate elevations corresponding to LiDAR data points.
- The interpolated surface TIN is generated, depicting elevations ranging from **65.7m to 65.82m**, facilitating further analysis and assessment of vertical accuracy.

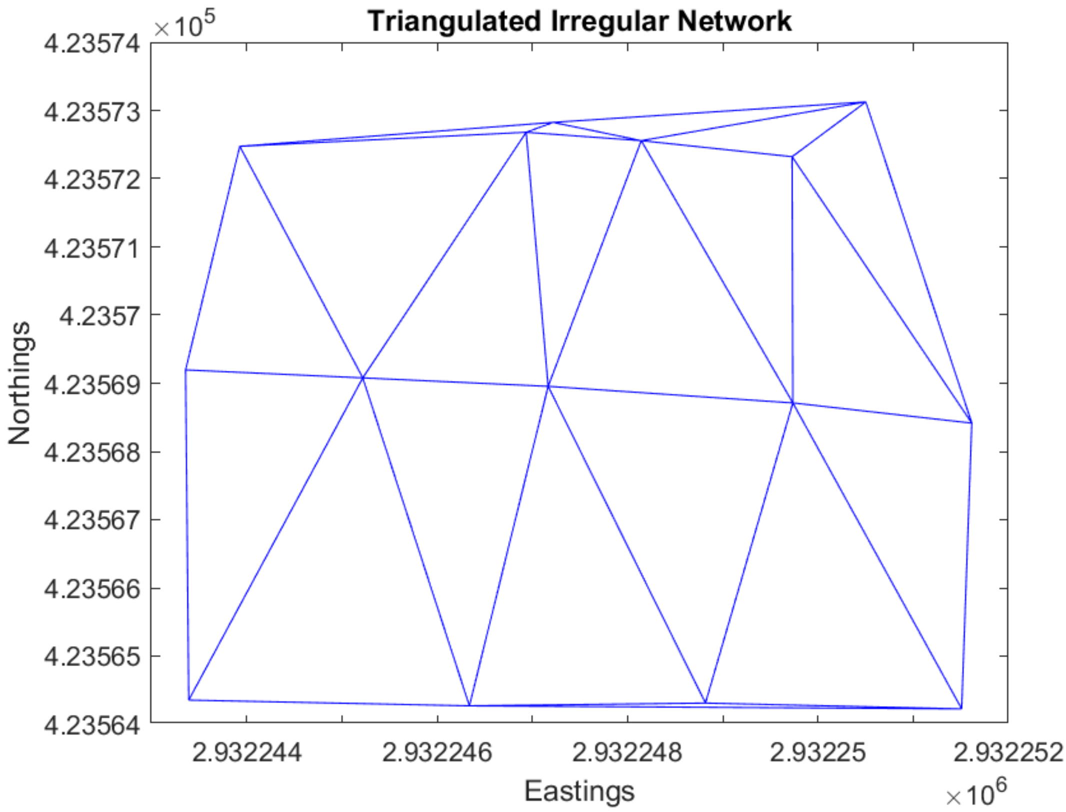



*Figure 19: TIN generated uisng MATLAB for the GCP,s*

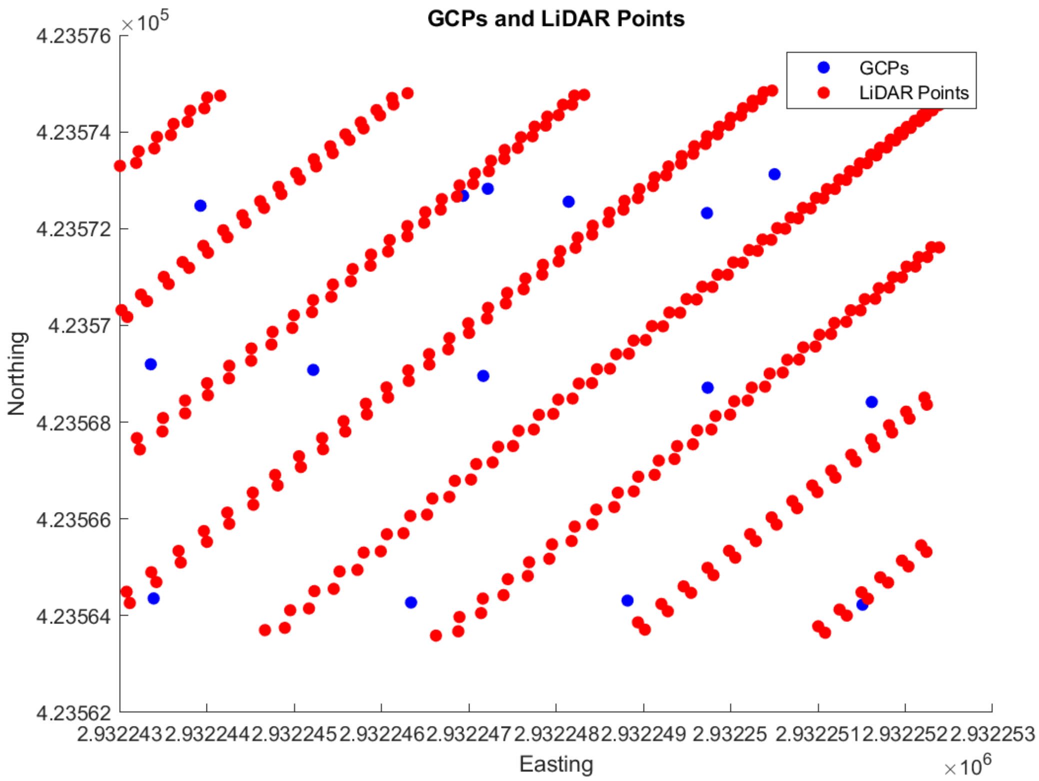
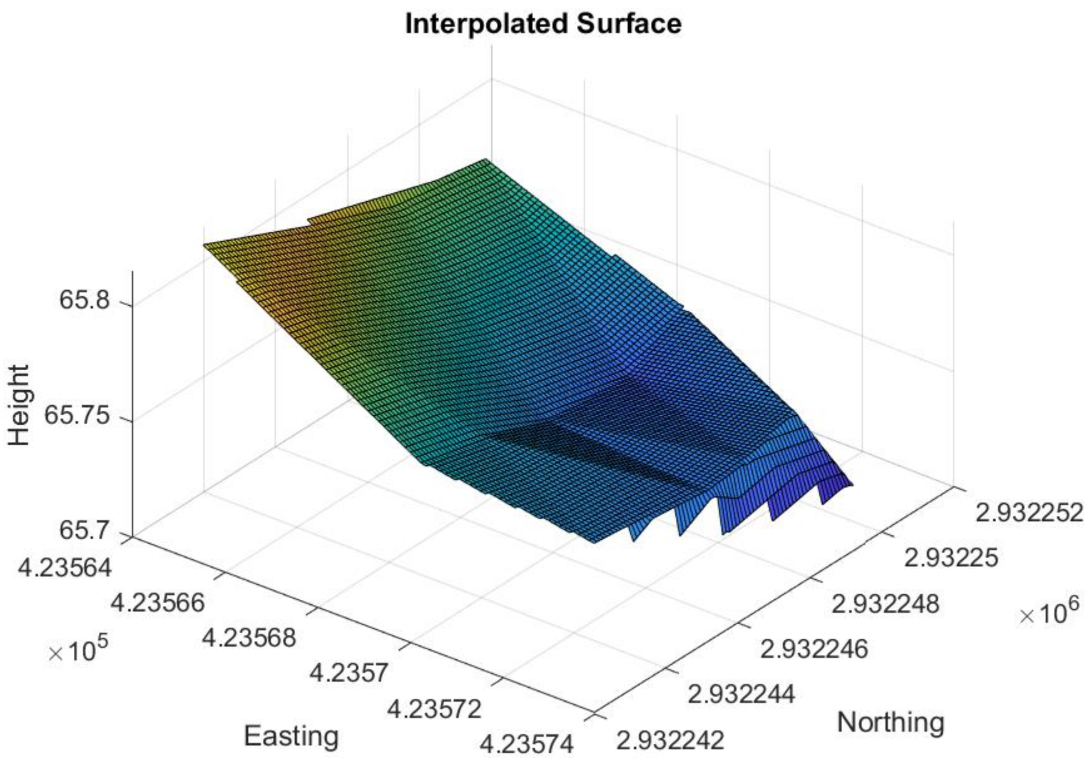



Figure 20: Showing both the GCP's and the Lidar Points and visualising the shift



## Case 2: Vegetated vertical accuracy

In areas with dense vegetation, accurately measuring the ground surface elevation using LiDAR data becomes challenging due to the presence of vegetation canopy. Vegetation obstructs LiDAR signals, resulting in inaccuracies in the elevation measurements. Therefore, assessing the vertical accuracy of LiDAR data in vegetated areas is crucial for understanding and mitigating these potential errors.

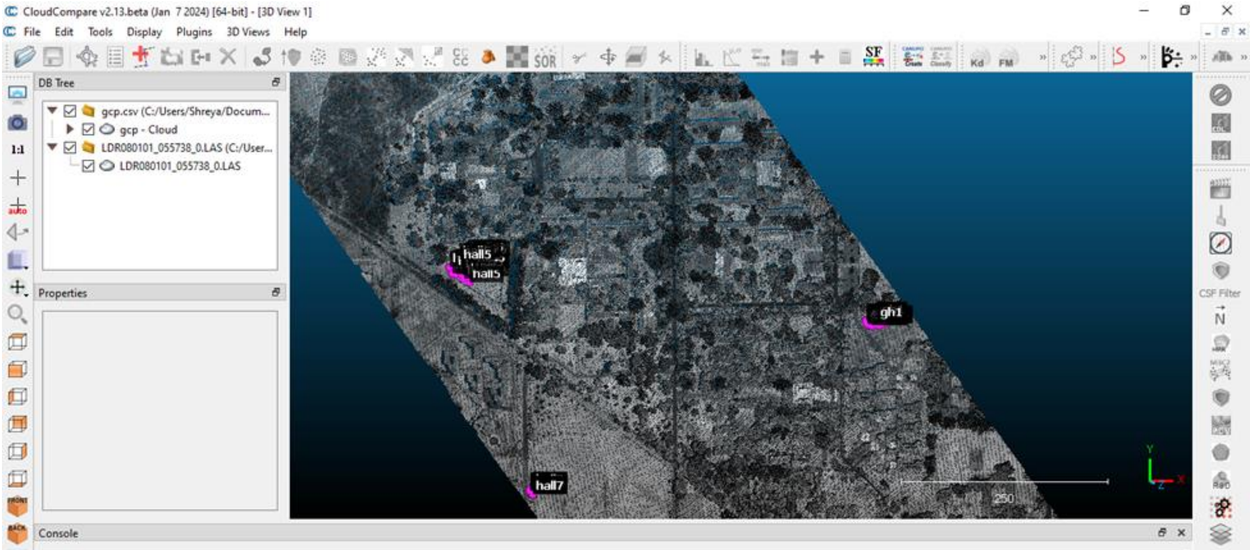



*Figure 21: GPC's Data overlayed on the Lidar data in CloudCompare*

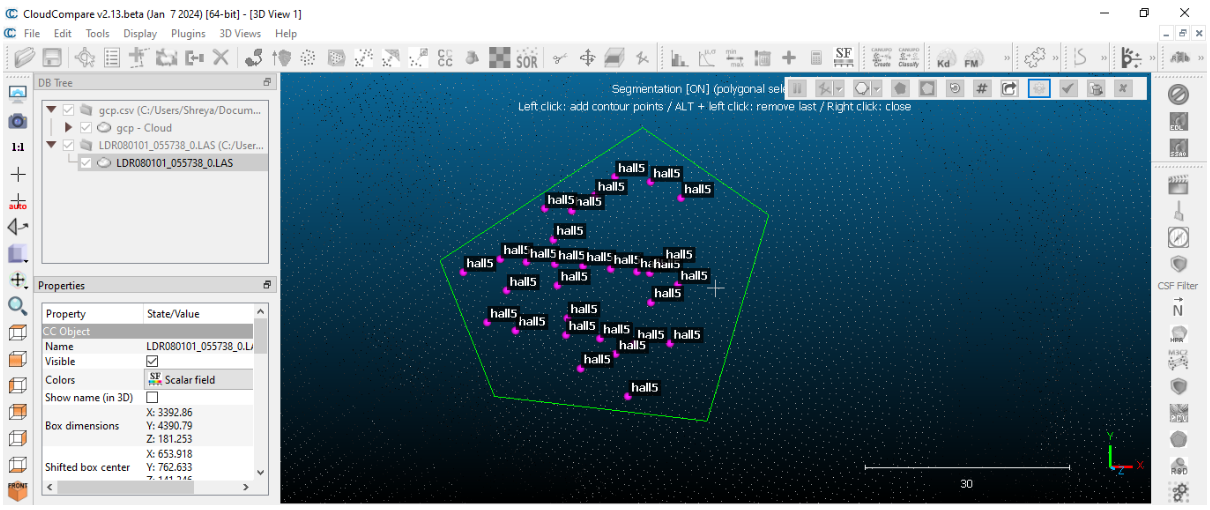



*Figure 22: segmented Lidar Data and GCP's*

**Procedure:**

Data Collection:

Ground Control Points (GCPs) were established in areas with grass or small vegetation, representing moderately vegetated regions.

30 GCPs were collected per data site to ensure a representative sample.

Data Segmentation:

The LiDAR data, along with the corresponding GCPs, were segmented according to the designated test sites.

The segmented data were saved in CSV format for further processing using MATLAB.

Algorithm Overview:

The MATLAB code was designed to quantify the vegetated vertical accuracy by comparing LiDAR-derived elevations with GCP measurements.

Algorithm Steps:

Step 1: Read LiDAR data and GCP data from CSV files.
Step 2: Extract elevation values from the third column of both datasets.
Step 3: Calculate the differences between LiDAR and GCP elevations to assess vertical accuracy.
Step 4: Sort the differences in descending order to identify outliers.
Step 5: Calculate the 95th percentile of the sorted differences.
Step 6: Display the calculated 95th percentile as the vegetated vertical accuracy.

The obtained results for both Vegetated Vertical Accuracy (VVA) and Root Mean Square Error (RMSE) provide insights into the quality and accuracy of the LiDAR data collected for the specified area.

## Results obtained in Case 1 (VVA) and Case 2 (NVA):

**Vegetated Vertical Accuracy (VVA) (95th percentile): 3.605.**

- This value represents the 95th percentile of the differences between LiDAR-derived elevations and ground control point (GCP) measurements in areas with vegetation. The VVA indicates the vertical accuracy of the LiDAR data in vegetated areas, specifically highlighting the extent of discrepancies between LiDAR-derived elevations and actual ground elevations. A higher value for VVA suggests larger discrepancies between LiDAR data and ground truth measurements in vegetated areas.

**Non-Vegetated Vertical Accuracy (NVA) (95th percentile): 0.40745.**

- This value represents the RMSE of the height differences between LiDAR-interpolated elevations and measured GCP elevations in non-vegetated areas. The RMSE quantifies the vertical accuracy of the LiDAR data in non-vegetated areas, providing a measure of the average discrepancy between LiDAR-derived elevations and ground truth measurements. A lower RMSE value indicates better agreement between LiDAR data and ground truth measurements in non-vegetated areas.

## 3.4 Analyze Absolute Planimetric (horizontal) Accuracy

Assessing planimetric accuracy in LiDAR data involves verifying the precision of horizontal measurements, typically achieved through ground control points or identifying discrepancies in redundant data. However, challenges arise due to the limited sampling characteristics of airborne LiDAR data, where point spacing often exceeds the laser ground spot diameter. Nevertheless, roof corners of buildings are often suitable ground control points for planimetric accuracy verification.

### The method:

- Identification of Possible Corner Data:
  - Segmentation of potential corner data from the LiDAR dataset.
  - Calculation of the average northing and easting values for each identified corner.
- Determination of Horizontal Accuracy (HA):
  - Loading of LiDAR data and ground control point (GCP) data from CSV files.
  - Extraction of northing and easting coordinates from both datasets.
  - Calculation of differences between corresponding points in the LiDAR and GCP datasets for both northing and easting coordinates.
  - Squaring of the differences to compute squared errors.
  - Computation of mean squared errors (MSE) for both northing and easting coordinates.
  - Calculation of root mean squared errors (RMSE) by taking the square root of the MSE.
  - Estimation of the 95th percentile confidence interval using the confidence multiplier.
  - Displaying of RMSE values for both northing and easting coordinates.
  - Displaying of estimated 95th percentile confidence intervals for both northing and easting coordinates.

Formulas and Functions Used:

- Read Matrix: The readmatrix function is used to load both the LiDAR data and GCP data from CSV files into MATLAB as matrices. This function handles reading data from files without a header and extracts the data as separate matrices.
- Difference Calculations: The differences between corresponding points in the LiDAR and GCP datasets are calculated for both northing and easting coordinates. This involves subtracting GCP values from LiDAR values.
- Squared Errors: The squared errors are calculated by squaring the differences obtained in the previous step. This is done for both northing and easting coordinates.
- Mean Squared Error (MSE): The mean squared errors are calculated by taking the mean of the squared errors for each coordinate (northing and easting).
- Root Mean Square Error (RMSE): The RMSE is calculated for both northing and easting coordinates by taking the square root of the mean squared errors.
- 95th Percentile Confidence Interval: The confidence interval is estimated using the confidence multiplier (1.96 for a 95% confidence level) and the RMSE values.

RMSE stands for Root Mean Square Error. It is a measure of the average magnitude of the errors between predicted or estimated values and the actual observed values. RMSE is widely used in various fields, including statistics, engineering, and data analysis, to assess the accuracy of models or measurements.

Mathematically, RMSE is calculated as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- $n$ is the number of observations.
- $yi$ is the observed (actual) value.
- $y^\wedge i$ is the predicted (estimated) value.
- The term $(yi - y^\wedge i)^2$ represents the squared difference between each observed and predicted value.
- The sum of squared differences is averaged over all observations.
- Finally, the square root of the average squared differences is taken to obtain the RMSE.

Analysis of the values as per ASPRS standard:

| Vertical Accuracy Class | Absolute Accuracy: $RMSE_z$ Non-Vegetated (cm) | Absolute Accuracy: NVA at 95% Confidence Level (cm) | Recommended Minimum NPD[8] (pls/m²) | Recommended Maximum NPS[8] (m) |
|---|---|---|---|---|
| 1-cm | 1.0 | 2.0 | ≥20 | ≤0.22 |
| 2.5-cm | 2.5 | 4.9 | 16 | 0.25 |
| 5-cm | 5.0 | 9.8 | 8 | 0.35 |
| 10-cm | 10.0 | 19.6 | 2 | 0.71 |
| 15-cm | 15.0 | 29.4 | 1 | 1.0 |
| 20-cm | 20.0 | 39.2 | 0.5 | 1.4 |
| 33.3-cm | 33.3 | 65.3 | 0.25 | 2.0 |
| 66.7-cm | 66.7 | 130.7 | 0.1 | 3.2 |
| 100-cm | 100.0 | 196.0 | 0.05 | 4.5 |
| 333.3-cm | 333.3 | 653.3 | 0.01 | 10.0 |

*Figure 23: ASPRS standard*

## Results Interpretation:

- **RMSE for Northing and Easting**: The RMSE values for **northing (0.44) and easting (0.77)** provide a measure of the standard deviation of the errors in the LiDAR data compared to the GCP data. Lower RMSE values indicate better accuracy.
- As per the ASPRS standards, as the obtained RMSE for NVA is 46.7cm, it falls under the highlighted category of fig (21).
- **Potential Discrepancies:** The differences observed between the LiDAR and GCP data may be influenced by various factors, including changes in terrain, vegetation growth, urban development, and technological advancements over the past 15 years. These factors could contribute to discrepancies between the two datasets, impacting the calculated RMSE and confidence intervals.

## 3.5 Examine the relative accuracy

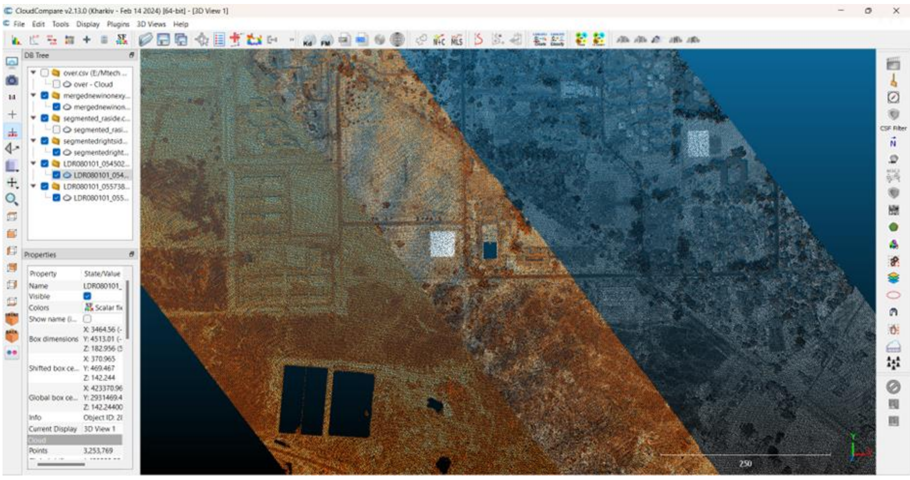


*Figure 24: Overlaying area, non-overlaying test sides considered for relative accuracy*

**Procedure:**

a. Use flat areas which are horizontal (say rooftop or road surface)
b. Select a patch in the above
c. Determine the average height of all the points of the patch
d. Compute the standard deviation of other data about this mean
e. This is vertical relative accuracy of data

**Average Height of the Selected Patch (Average Elevation)**:

- The average height of all points within the selected patch is calculated by taking the mean of the elevation (Z-coordinate) values of the points in the patch.
- This value represents the central tendency or the typical elevation of the patch.

**Relative Vertical Accuracy:**

- The relative vertical accuracy of the data in the patch is computed as the standard deviation of the heights of the points in the patch from the mean height (average elevation).
- In other words, it measures the spread or variability of the elevation values around the mean elevation.
- A lower relative vertical accuracy value indicates that the elevation values of the points in the patch are more closely clustered around the mean, implying higher accuracy

Formula: $\text{Average Height} = \frac{\sum_{i=1}^{N} Z_i}{N}$

where $N$ is the total number of points in the patch, and $Zi$ represents the elevation of each point.

$$\text{Vertical Accuracy} = \sqrt{\frac{\sum_{i=1}^{N}(Z_i - \text{Average Height})^2}{N}}$$

Where $N$ is the total number of points in the patch, $Zi$ represents the elevation of each point, and **Average Height** is the mean height calculated earlier.

Relative vertical accuracy of the data in the non-overlaying area of flight line 1 is: 3.7683 cm
Relative vertical accuracy of the data in the non-overlaying area of flight line 2 is: 8.0516 cm
Relative vertical accuracy of the data in the overlaying area is: 19.1502 cm

## 3.6 Analyze the data density and NPS.

### Data Density:

Data density refers to the density or concentration of LiDAR points within a specific area, typically measured as the number of points per unit area. It provides insights into the spatial resolution or level of detail captured by the LiDAR dataset.

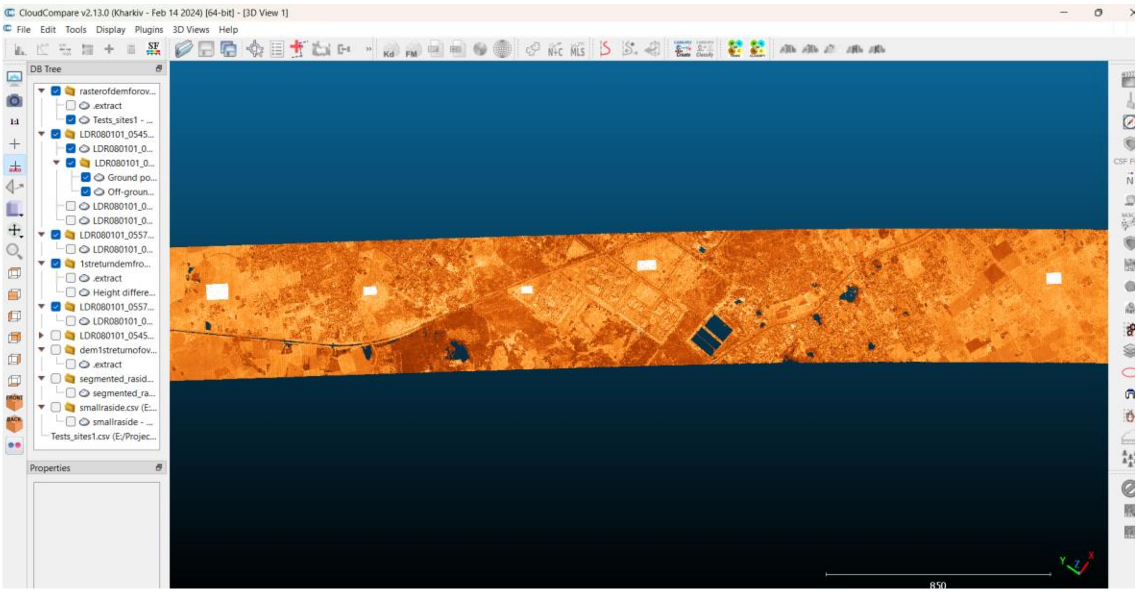



*Figure 25: Various test-sites segmented using CloudCompare*

**Procedure for Data Density Calculation:**

1. Locate rooftops or other flat surfaces where only **first return data are generated** and find area of such patches and total number of points for this.
2. Using a segmentation tool in **CloudCompare**, flat areas such as rooftops, roads, and ground were identified and segmented from the LiDAR point cloud data. This segmentation process separates the LiDAR points into different clusters or segments based on their characteristics, such as elevation and surface properties.
3. The segmented flat areas were combined into a single dataset, possibly by exporting the segmented areas as individual **CSV files** and then merging them into one CSV file containing all the flat areas.
4. **Create Delaunay Triangulation:** Create a 2D Delaunay triangulation of points from the LiDAR dataset to form Thiessen polygons.
5. **Calculate Thiessen Polygon Areas:** Determine the area of the Thiessen polygons corresponding to each point. This area represents the inverse of the density of LiDAR points around that point.

6. Iterate over each point and find its nearest neighbors (excluding itself) using the k-nearest neighbors **(KNN) algorithm.**
7. **Sort Areas:** Sort the calculated areas in descending order.
8. **Determine Threshold:** Determine the threshold for the top 5% of areas, indicating the data density.
9. **Identify Data Density:** Identify the data density by selecting the areas exceeding the top 5% threshold.
10. **Display Results:** Display the calculated data density, providing insights into the point spacing characteristics of the LiDAR data.

To calculate data density, we utilize the concept of **Thiessen polygons**, also known as **Voronoi polygons.** Thiessen polygons are a partitioning of a plane into regions based on proximity to a set of input points. Each polygon represents an area in the plane that is closer to a specific input point than to any other point in the set. The formula used to calculate the area of a Thiessen polygon for a given point involves determining the distance to its neighboring points and using these distances to compute the polygon's area. This process is repeated for each point in the dataset. By computing the area of each polygon, we can indirectly assess the density of LiDAR points around each point in the dataset.

Thiessen polygons, we first create a Delaunay triangulation based on the LiDAR data points. This triangulation divides the plane into a set of triangles based on the LiDAR point locations. Then, for each LiDAR point, we find its nearest neighbors using a specified number of nearest neighbors (in this case, 4).

The formula to calculate the area of a Thiessen polygon for a given LiDAR point $PiPi$ involves the following steps:

- Find the $kk$ nearest neighbors of point $PiPi$ within the Delaunay triangulation.
- Construct a polygon using the $kk$ nearest neighbors and point $PiPi$.
- Calculate the area of this polygon using the formula for the area of a polygon.
- Repeat steps 1-3 for each LiDAR point in the dataset.

The notation used in the formula is as follows:

- $Pi$: The $i$ith LiDAR point for which the Thiessen polygon area is being calculated.
- $k$: The number of nearest neighbors to consider when constructing the Thiessen polygon.
- $A(Pi)$: The area of the Thiessen polygon associated with LiDAR point $PiPi$.
- $Nk\ (Pi)$: The $kk$ nearest neighbors of point $PiPi$ within the Delaunay triangulation.
- $dij$: The Euclidean distance between points $PiPi$ and $PjPj$.
- $n$: The total number of LiDAR points in the dataset.

The formula to calculate the area of the Thiessen polygon $A(Pi)$ for a given LiDAR point $Pi$ is as follows:

$$A(Pi) = \frac{1}{2}\sum{}^{k}{}_{j} = dij \cdot di(j+1) \cdot \sin(\theta ij)$$

where:

- $\theta ij$: The angle formed between segments $PiPj$ and $PiPj+1PiPj+1$.

Once we have calculated the area of the Thiessen polygons for all LiDAR points, we sort the areas in descending order and determine the top 5% threshold. The data density is then calculated as the area corresponding to this threshold.

**The result obtained for data density is 0.7433 square units**. This value indicates the average spacing between points in the LiDAR dataset. A lower data density value suggests that the LiDAR points are more densely spaced, meaning there are more points per unit area. This can be beneficial for applications requiring detailed terrain modeling and accurate feature extraction. However, it's essential to consider the project requirements and balance the benefits of higher density with associated costs and processing considerations.

## NPS (Nominal Pulse Spacing):

The Nominal Pulse Spacing (NPS) represents the average spacing between points collected by a single LiDAR instrument within a single swath, specifically considering only the first return data. It provides an indication of the grid spacing of the collected points, which is essential for understanding the resolution of the LiDAR data. The NPS assessment typically focuses on the center 90% of each swath, excluding any unacceptable data voids.

Procedure for NPS Calculation:

- Load LiDAR Data: Load the LiDAR data from the CSV file for each flight line, skipping the first row which usually contains headers.
- Calculate Distances between neighboring points: Calculate the Euclidean distance between neighboring points in the LiDAR dataset. This involves computing the distance between each point and its adjacent point in the dataset.
- Compute Average Distance: Compute the average distance between neighboring points using a moving average approach to smooth out any irregularities. This involves averaging the distances calculated in the previous step over a specified window size. The moving average helps to smooth out variations in point density.
- Sort Distances: The computed average distances are sorted in descending order to identify the longest distances between neighboring points.
- Determine Threshold: The top 5% threshold of the sorted average distances is determined. This threshold represents the longest distances between neighboring points, indicating areas of lower point density.
- Identify NPS: The NPS is identified by selecting the average distances that exceed the top 5% threshold. This provides an estimate of the typical spacing between LiDAR points in the dataset.
- Display Results: Display the calculated NPS for each flight line, providing insights into the point spacing characteristics of the LiDAR data.

$$\mathbf{NPS} = \frac{\mathbf{1}}{\sqrt{\textbf{Data Density}}}$$

Where:

- NPS is the Nominal Pulse Spacing,
- Data Density represents the number of LiDAR points per unit area ($pts/m^2$ or $pts/km^2$).

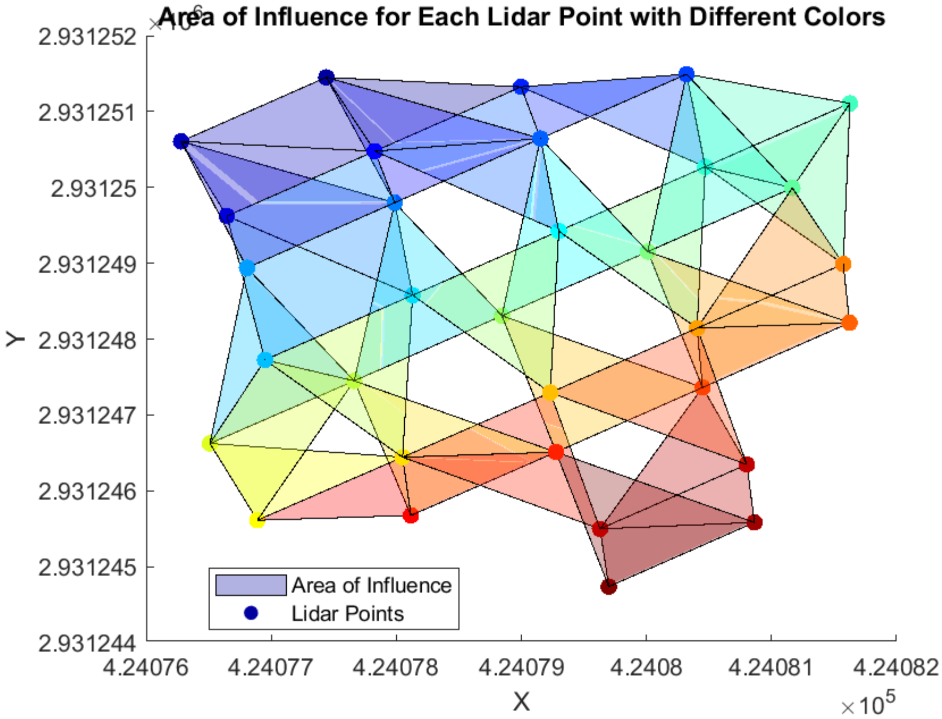
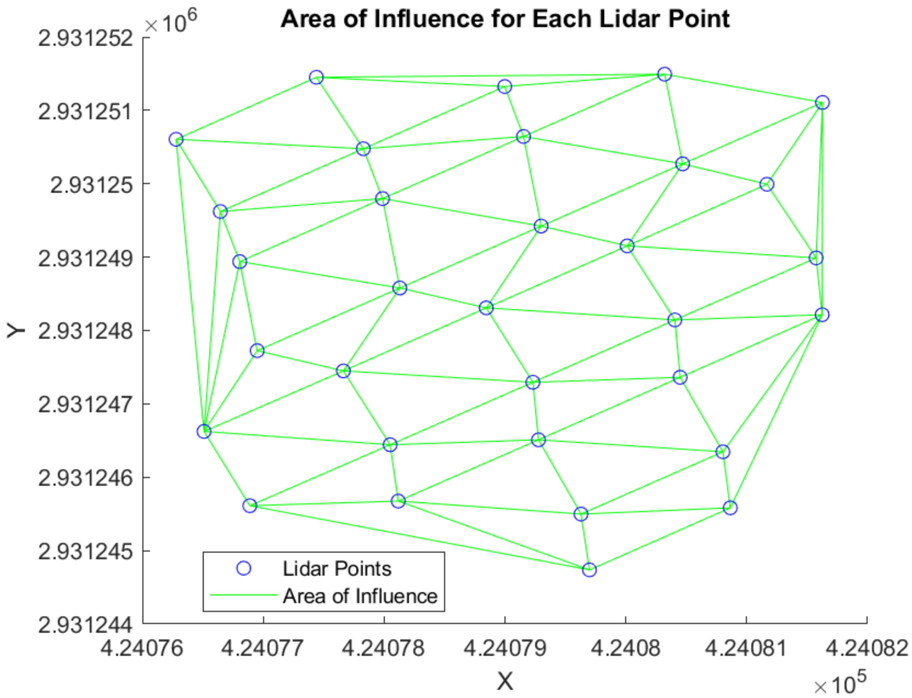



*Figure 26: img of Thiessen polygon shown in different colour generated using the lidar data*




- Flight Line 1 likely has a denser point cloud with points closer together, indicating higher resolution data.
- Flight Line 2 has a slightly higher NPS, suggesting a slightly sparser point cloud with points positioned slightly further apart compared to Flight Line 1. This may result in slightly lower resolution data compared to Flight Line 1.

## 3.7 Identify Data Voids

Data voids refer to areas within a LiDAR dataset where there is a lack of sufficient data points, which could adversely affect the quality and accuracy of the derived products. These voids are typically undesirable as they can lead to inaccuracies or gaps in the representation of the terrain surface. Natural factors contributing to data voids may include water bodies, where LiDAR signals are absorbed or reflected differently, resulting in fewer data points. Additionally, low reflectance areas, especially in the near-infrared (NIR) region, may also contribute to voids as the LiDAR system may struggle to detect and capture returns from such surfaces effectively. Operational challenges such as high winds, extreme weather conditions, or navigation errors during data collection can also lead to unintended gaps in the data. Moreover, intentional post-processing procedures aimed at removing man-made structures or dense vegetation may result in voids in the dataset, especially in urban or vegetated areas. Ideally, LiDAR point data should exhibit a uniform spatial distribution without significant clustering or widely spaced high-density profiles. While LiDAR instruments do not generate regularly gridded points, the data collection should resemble a regular lattice of points, maintaining consistent point density throughout the dataset. Exceptions to data voids include specific scenarios where voids are deemed acceptable. For example, voids may be expected in water bodies due to signal absorption, or in areas with low reflectance where LiDAR signals struggle to penetrate dense vegetation. Additionally, areas covered by the overlap of two swaths may exhibit voids due to the nature of LiDAR data acquisition. Ensuring adequate point density and minimizing data voids is essential for accurate terrain representation and reliable geospatial analysis using LiDAR data. Therefore, assessing and addressing data voids are crucial steps in quality assurance and data validation processes for LiDAR datasets.

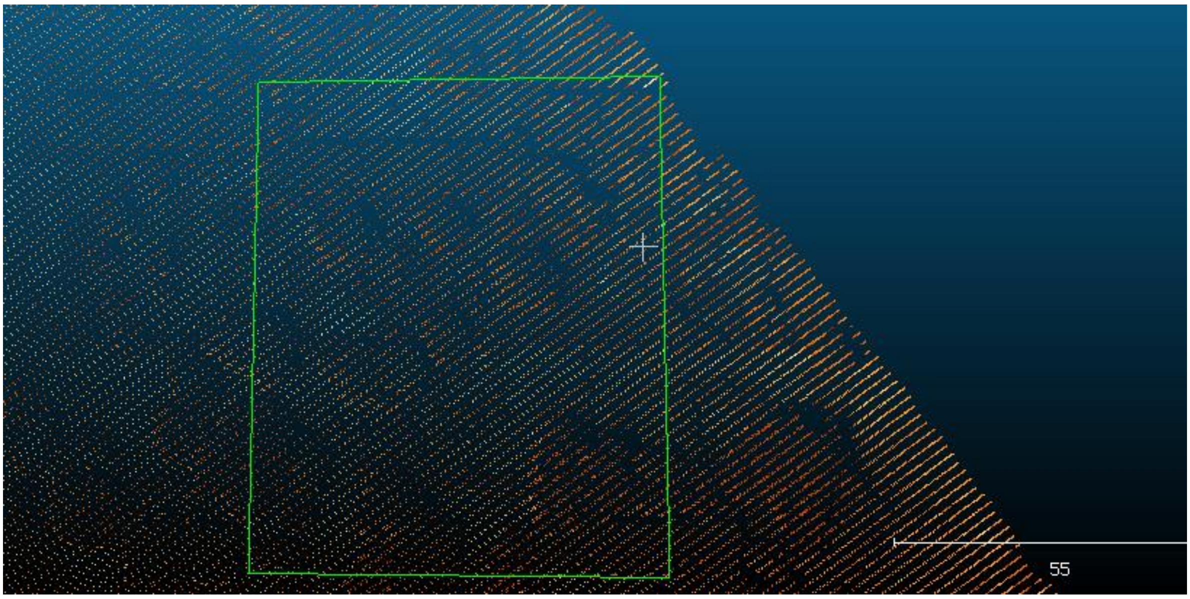



*Figure 27: Segmenting area with data voids using CloudCompare*

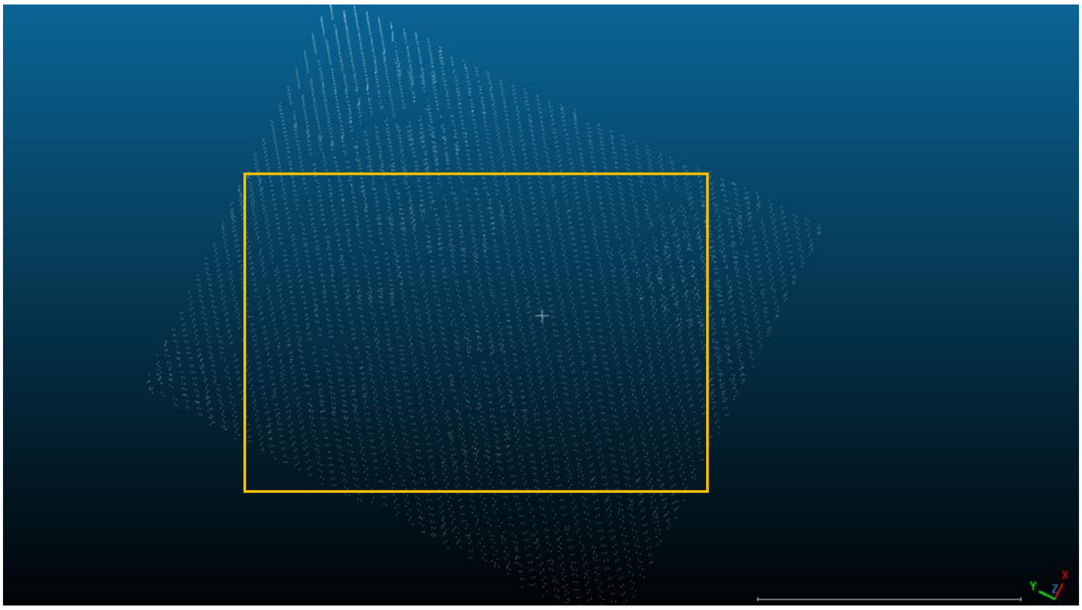


*Figure 28: Zoomed-in image of the test site 1 containing data-voids*

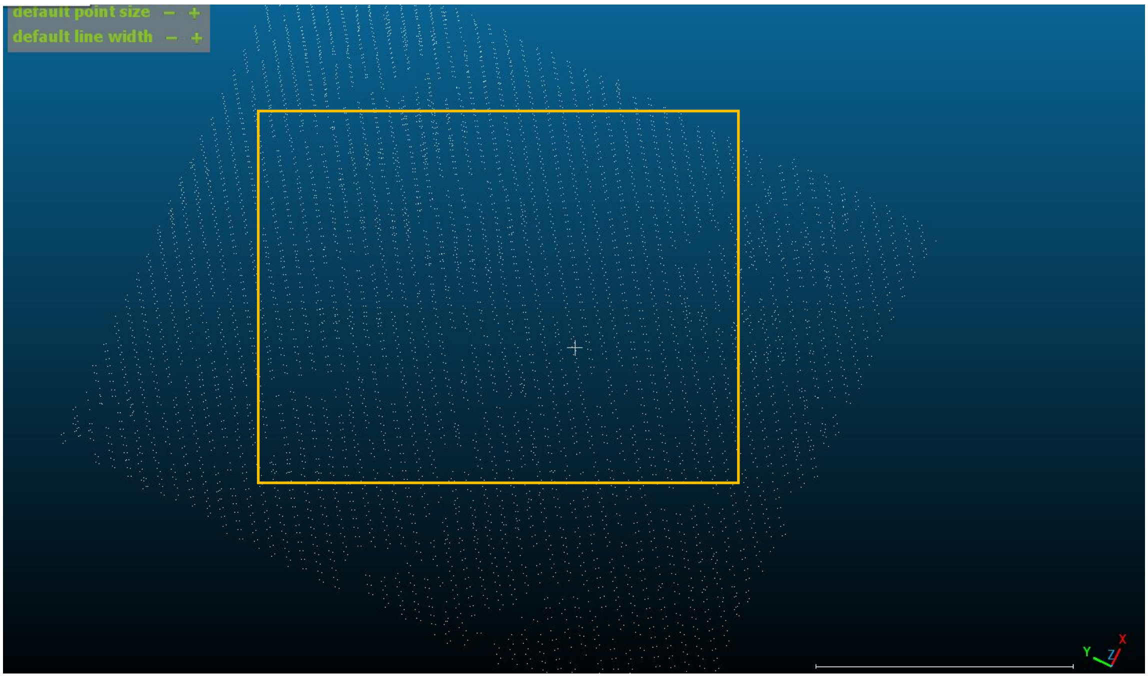



*Figure 29: When removing the noise in data using SOR tool in CloudComparethe voids are more visible*

To determine the presence of data voids and ensure data coverage is adequate, the following steps were performed:

- **Establishing Grid Cells**: A grid of 1m by 1m was established to partition the study area. Each grid cell represents a spatial unit for assessing the point density.
- **Point Density Criterion**: A criterion was set that each grid cell should contain a minimum of 10 LiDAR points per square meter. This criterion ensures sufficient point coverage for accurate representation of the terrain.

- **Counting Points in Grids**: The LiDAR points were then examined to determine the number of points falling within each grid cell. This count was performed to assess whether each grid met the density criterion.
- **Calculating Percentage of Filled Grids**: The total number of filled grid cells meeting the density criterion was divided by the total number of grid cells in the study area. This calculation yielded the percentage of grids with sufficient point coverage.
- **Threshold Evaluation**: If the percentage of filled grids exceeded a specified threshold (e.g., 90%), it indicates that the LiDAR data adequately covers the area with the required point density. Conversely, if the percentage falls below the threshold, it suggests the presence of significant data voids.

$$\textbf{Percentage of grids meeting the criterion} = \frac{\textbf{Total number of grids}}{\textbf{Number of grids meeting the criterion}} \textbf{x100\%}$$

This calculation is performed using the **histcounts2** function to count the number of points in each grid cell and comparing it against the predefined density criterion. The percentage of grids meeting the criterion is then determined based on the count of filled grids.

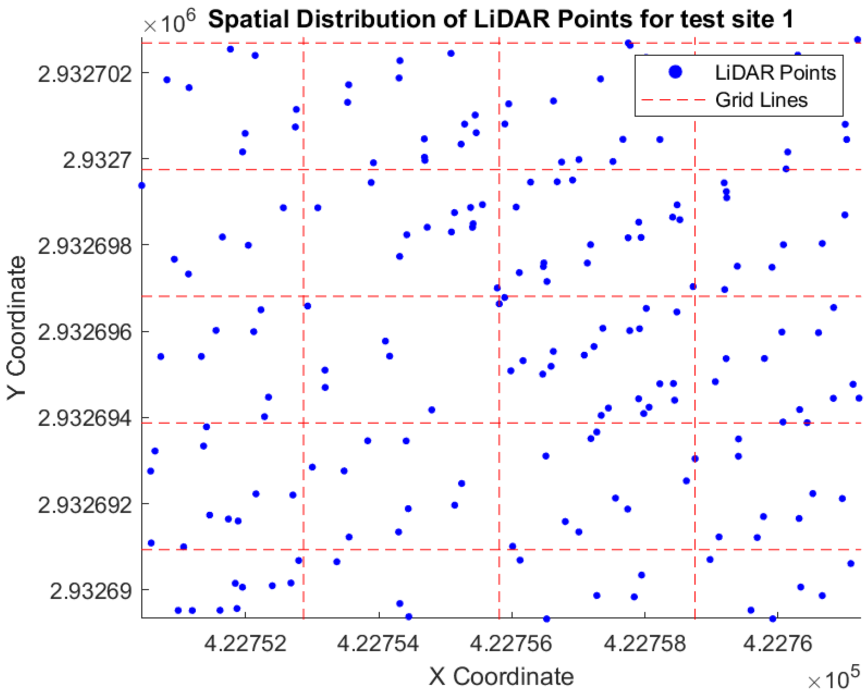
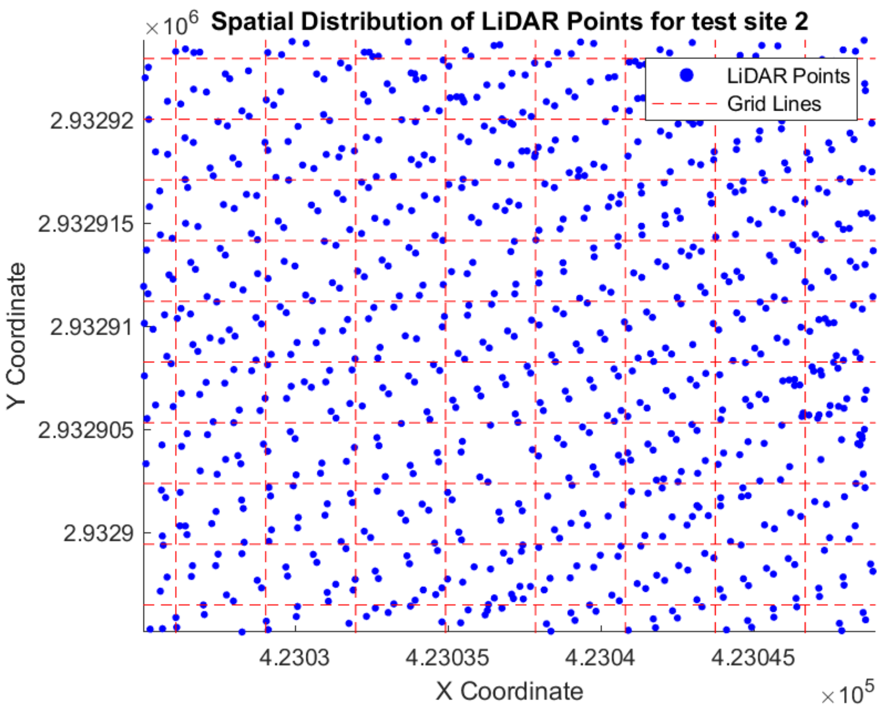



*Figure 30:*



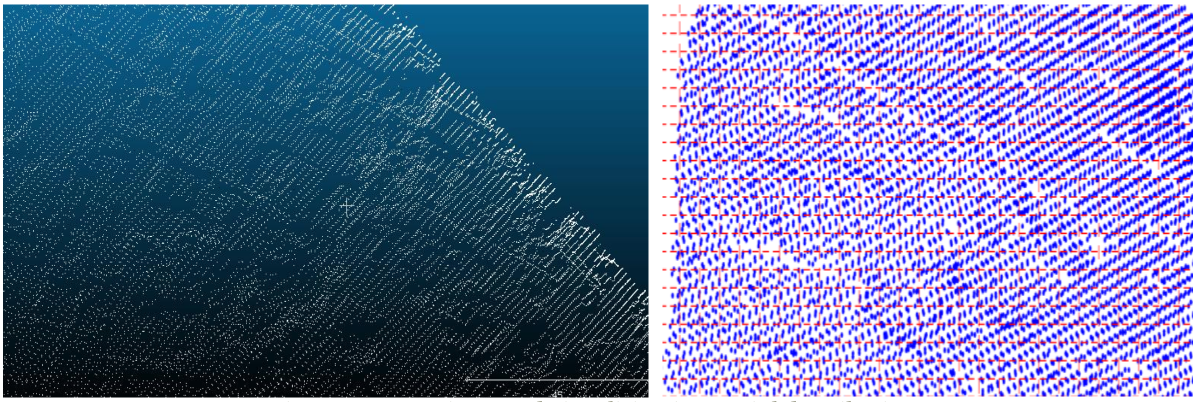



Figure 31: Test site-2 and visualising its spatial destribution

The results obtained for each test site indicate that the percentage of filled grids exceeded the acceptable threshold of 90%, indicating that the data voids in all test sites are within acceptable limits. This suggests that the LiDAR data adequately covers the study areas with the required point density, ensuring accurate representation of the terrain surface.

## 3.8 Assess Spatial Distribution

Spatial distribution refers to the arrangement or pattern of data points across a given area or region. It involves assessing whether the data points are evenly spread out or if they exhibit clustering or gaps. In the context of LiDAR data, spatial distribution is critical for understanding the coverage and quality of the surveyed terrain.

**Procedure:**

Data file Extraction using CloudCompare

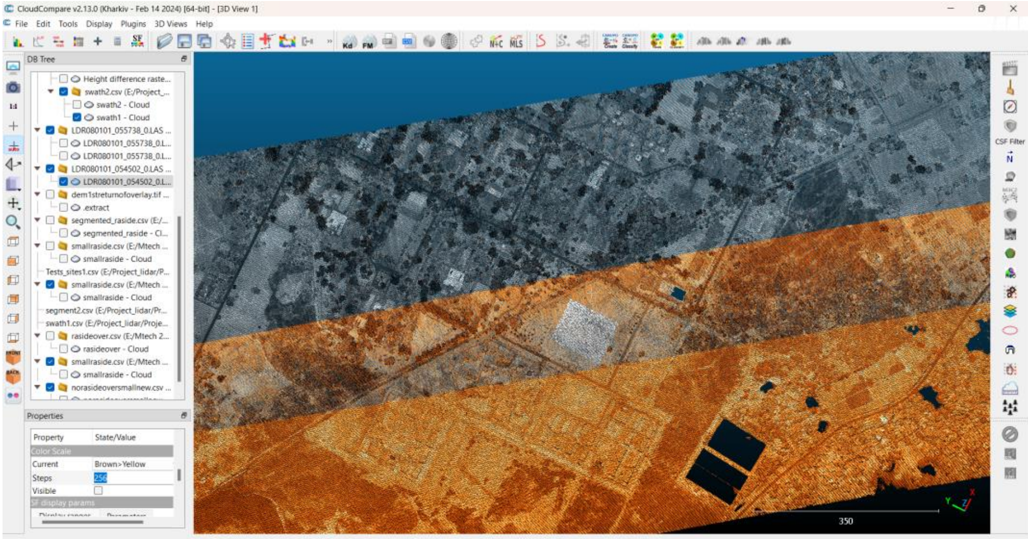


*Figure 32: White colour patch chosen as test-site*

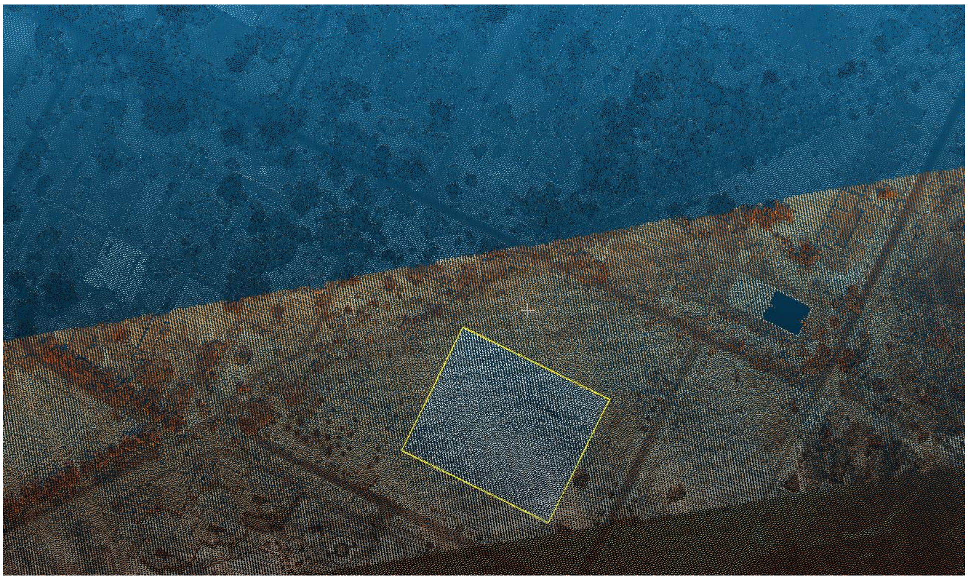


*Figure 33: Zoomed-in image of the test-site1*

1. **Uniform Distribution:** In an ideal scenario, the LiDAR data points should be uniformly distributed across the surveyed area. This means that the points are evenly spread out without any significant clustering or gaps, providing comprehensive coverage of the terrain.
2. **First Return Data:** When assessing spatial distribution, only the first return data is typically considered. The first return represents the laser pulse's initial interaction with the surface, providing valuable information about the topography and features of the area.
3. **Central 90% of the Swath**: LiDAR systems often capture data in swaths or strips as they scan the terrain. When evaluating spatial distribution, it's common to focus on the central portion of these swaths, typically the central 90%. This ensures that the analysis is based on the most representative data captured by the LiDAR sensor.
4. **Grid of Size 2*NPS:** To analyze spatial distribution, a grid is overlaid on the surveyed area, with each grid cell having dimensions of 2 times the Nominal Pulse Spacing (NPS). The NPS represents the average distance between adjacent LiDAR points and serves as a measure of point density.
5. **90% Grids Should Be Filled:** The objective is for at least 90% of the grid cells to contain at least one LiDAR point. This criterion ensures that the data coverage is sufficiently dense and comprehensive, minimizing gaps and ensuring accurate representation of the terrain.

In the scenario where we assessed the spatial distribution for flight line 1 using the following method:

- **Grid-based Analysis**: We divided the surveyed area into a grid with each grid cell having dimensions of 2 times the Nominal Pulse Spacing (NPS). This grid-based approach allowed us to systematically analyze the spatial distribution of LiDAR points.
- **Percentage of Filled Grids**: We calculated the percentage of grid cells that contained at least one LiDAR point. The objective was for at least 90% of the grid cells to be filled, indicating a uniformly distributed spatial pattern.
- **Assessment Result**: Based on the analysis of the filled grid cells, we determined whether the spatial distribution for flight line 1 was uniform or non-uniform.

The result indicates that **93.9815%** of the grid cells are filled with LiDAR points for flight line 1. This percentage is well above the threshold of 90%, indicating that the spatial distribution of LiDAR points in flight line 1 is uniform. Uniform spatial distribution is desirable as it ensures adequate coverage and sampling of the area, allowing for accurate analysis and interpretation of the data.

# 4. Results:

The analysis revealed valuable insights into the quality and characteristics of the aerial LiDAR data for the IIT Kanpur campus:

- Vertical Accuracy Assessment:
  - Vegetated Vertical Accuracy (VVA) was found to be 3.605, indicating the 95th percentile of differences between LiDAR-derived elevations and ground control point (GCP) measurements in vegetated areas.
  - Non-Vegetated Vertical Accuracy (NVA) was determined to be 0.40745, representing the RMSE of height differences between LiDAR-interpolated elevations and measured GCP elevations in non-vegetated areas.
- Horizontal Accuracy Evaluation:
  - Root Mean Square Errors (RMSE) for northing and easting coordinates were calculated as 0.44 and 0.77, respectively, providing measures of the standard deviation of errors in the LiDAR data compared to GCP data.
- Relative Accuracy Analysis:
  - Relative vertical accuracy values were determined for different test sites, with the non-overlaying area of flight line 1 exhibiting a relative vertical accuracy of 3.7683 cm.
- Data Density and NPS Assessment:
  - The data density was computed as 0.7433 square units, indicating the average spacing between points in the LiDAR dataset.
  - Nominal Pulse Spacing (NPS) analysis revealed variations between flight lines, with Flight Line 1 exhibiting denser point clouds compared to Flight Line 2.
- Identification of Data Voids:
  - Data voids within the LiDAR dataset were assessed, and it was determined that the percentage of filled grids exceeded the acceptable threshold of 90% in all test sites.
- Spatial Distribution Evaluation:
  - Spatial distribution analysis indicated that 93.9815% of grid cells were filled with LiDAR points for Flight Line 1, demonstrating a uniform distribution pattern.
- LiDAR Overlap Assessment:
  - The LiDAR overlap assessment provides crucial insights into the degree of coverage redundancy between the two flight lines, aiding in the evaluation of data consistency and completeness.
  - Minimum Overlap: 17.5903%
  - Maximum Overlap: 19.2237%
  - Average Overlap: 18.407%

# 5. Conclusion:

The comprehensive analysis of the aerial LiDAR data for IIT Kanpur campus yielded promising results regarding its quality and accuracy. The assessment of vertical and horizontal accuracy, relative accuracy, data density, NPS, identification of data voids, and spatial distribution provided valuable insights into the reliability and suitability of the dataset for various applications. Overall, the LiDAR data exhibited satisfactory vertical and horizontal accuracy, with minor variations observed between vegetated and non-vegetated areas. The relative accuracy analysis highlighted differences in accuracy across different test sites, indicating the importance of considering local conditions and terrain characteristics.
In conclusion, the quality assessment of the aerial LiDAR data for IIT Kanpur campus indicates its suitability for a wide range of applications, including terrain modeling, urban planning, and environmental monitoring. However, ongoing monitoring and periodic updates may be necessary to maintain data integrity and accuracy over time.

# 6. References

Jie Shan, C. K. (2008). Topographic Laser Ranging and Scanning . *Taylor & Francis Group, LLC.*

B. L. Lohani, "Airborne Altimetric LiDAR: Principle, Data Collection, Processing and Applications," IIT Kanpur