Palm Tree Mesh Creation

Mahmoud Sunbul

June-July 2025

1 Introduction

This documentation describes the process and evaluation of creating a detailed 3D mesh of a palm tree in `.obj` format. Such meshes are important for applications including simulation for harvest path planning and tree health monitoring, such as detecting pests that may reduce date quality or damage the tree. The primary objective is to determine an efficient and effective method for generating high-quality palm tree meshes, which can be used for dataset creation, algorithm training, and research in agricultural robotics and inspection.

Each team member explored a different approach to mesh generation. The approach described here uses a moving camera to capture and reconstruct the tree's geometry. This report covers the tools, pipeline, parameter selection, experimental methodology, results, and recommendations.

2 Pipeline Overview

The mesh creation process consists of several stages: data capture, processing, intermediate visualization, and mesh refinement.

Pipeline Flowchart

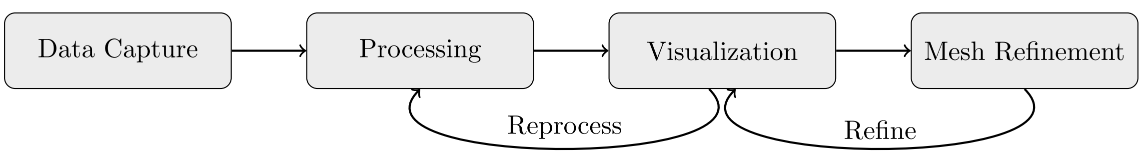


Figure 1: Pipeline workflow for palm tree mesh creation

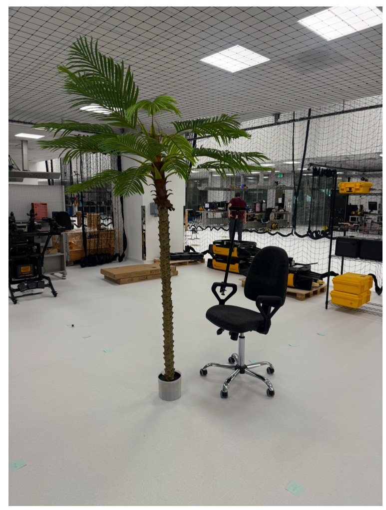
3 Tools and Technologies

Two main devices were used: the ZED stereo camera and the iPhone 16 Pro Max, each with their respective software (ZED SDK and a third-party iPhone LiDAR app (3d Scanner App)). The ZED camera enables programmable integration and precise adjustment of internal capture parameters, while the iPhone app provides rapid mesh creation using LiDAR and photogrammetry, with real-time preview. Note: Each of them used different technology, sensors, and algorithms; therefore, this should be considered when examining the results.

Camera/Device Pictures

Figure 2: ZED_Camera

(a)

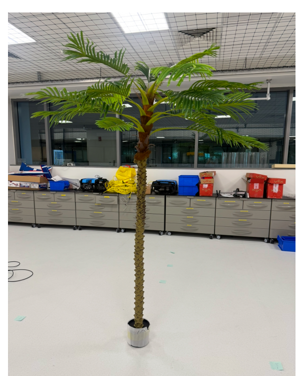


(b)

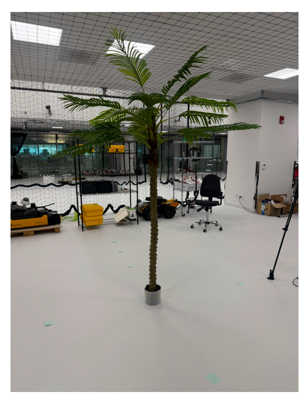
Figure 3: Camera Mounted on Wheelchair

(a)

(b)



(c)



(d)

Figure 4: Palm tree image for visual reference

Parameter	ZED Camera	iPhone App
Confidence Threshold	✓	✓
Range Adjustment	✓	✓
Resolution Choice	✓	✓
Frame Rate	✓	
Manual Exposure Control	✓	
Sensitivity to Motion	High	Low
Live Preview		✓
Export Formats (.obj, etc.)	✓	✓
Processing Customization	✓	✓

Table 1: Main controllable and influencing parameters for each data capture method.

4 Parameters

Note: *Environmental lighting is an external factor affecting both methods but on different scales.*

4.1 Parameters for iPhone App

The iPhone LiDAR scanning app offers several modes and parameter controls both before and after recording. Some settings must be chosen before capture and cannot be changed post-recording.

4.1.1 Pre-Recording Options (Modes and Controls)

- **Mode of Recording:**
  - **LiDAR Advanced**
  - **LiDAR**
  - **Point Cloud**
  - **Photos**
  - **TrueDepth**
  - **RoomPlane**
- **Controllable Parameters (*LiDAR Advanced* mode):**
  - **Confidence:** Low, Medium, High (set before scanning)
  - **Range:** 0.3 m to 6.0 m (set before scanning)
  - **Resolution:** 5 mm to 50 mm (set before scanning)
  - **Masking:** Enable/disable masking of face/object (set before scanning)
- **Note:** These options cannot be changed after recording; for LiDAR mode, more post-processing is available (see below).

4.1.2 Post-Capture Process Parameters (LiDAR / LiDAR Advanced)

Parameter	LiDAR	LiDAR Advanced	Increment	Range / Options
Voxel Size	✓		1 mm	3 mm – 21 mm
Max Range	✓		0.1 m	1.0 m – 6.0 m
Smoothing	✓	✓	1x	Off, 1x – 8x
Simplify	✓	✓	1%	Off, 8% – 95%

Table 2: Post-capture parameters: controllable options for LiDAR and LiDAR Advanced modes in the iPhone app.

Note: *With LiDAR Advanced mode, you must select confidence, range, resolution, and masking before recording. In LiDAR mode, you can adjust voxel size, max range, smoothing, and simplify after recording. This distinction affects workflow and flexibility.*

4.2 Parameters for ZED Camera

The ZED Camera offers extensive flexibility in parameter selection at each stage of data acquisition and processing. Users can either utilize the official ZED applications, which provide a straightforward graphical interface, or work directly with the ZED SDK libraries for greater control and integration with custom workflows.

While the ZED app is designed for ease of use, the SDK and libraries allow for advanced configuration and fine-tuning of nearly all processing parameters. This enables integration with other software tools and adaptation to a wide range of project requirements.

Due to the wide variety of available parameters—which would require several pages to describe in full—readers are encouraged to consult the official ZED documentation (see References) for comprehensive details.

A summary of the primary parameters used in this work is provided in the "Processing Settings" tables for ZED Camera Example 1 and Example 2.

5 Experimental Methodology

Two main data collection setups were used:

1. **ZED Camera on Wheelchair:** The camera was attached to a wheelchair (see Figure 3) to minimize vibration and enable smooth rotation around the palm tree at a fixed distance of approximately 2 meters.

2. **iPhone 16 Pro Max:** The iPhone was used with a LiDAR-based app. Different modes (LiDAR Advanced, LiDAR) were evaluated.

Each method was employed to collect two samples, resulting in a total of four mesh outputs. The comparisons were conducted using both visual inspection—evaluating surface detail, completeness, and realism—and quantitative analysis, which included the number of vertices, triangles, and file size.
Note: *To ensure an accurate comparison, the vertex count should include only the mesh itself after filtering out the background. Additionally, since different technologies and algorithms were utilized, these factors should be carefully considered during the analysis.*

6 Results and Examples

Each sample below includes method, key parameters, original data characteristics, mesh statistics, and visualization placeholders.

Example 1: ZED Camera Sample 1 (Wheelchair Setup)

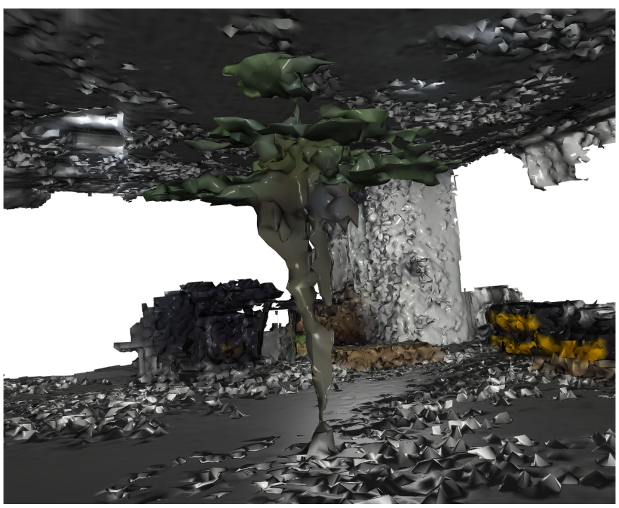
Processing Settings for ZED Camera Sample 1

Parameter	Value
DEPTH_MODE	`NEURAL_PLUS`
COORDINATE_UNITS	`MILLIMETER`
DEPTH_STABILIZATION	50
DEPTH_MIN	600 mm
DEPTH_MAX	2500 mm
SPATIAL_RESOLUTION	`HIGH`
SPATIAL_RANGE	`SHORT`
MAX_MEMORY_MB	8192
SAVE_TEXTURE	True
USE_CHUNK_ONLY	False
REVERSE_VERTEX_ORDER	False
MAP_TYPE	`MESH`
STABILITY_COUNTER	0 (auto)
FILTER_LEVEL	`LOW`
CONFIDENCE_THRESHOLD	77

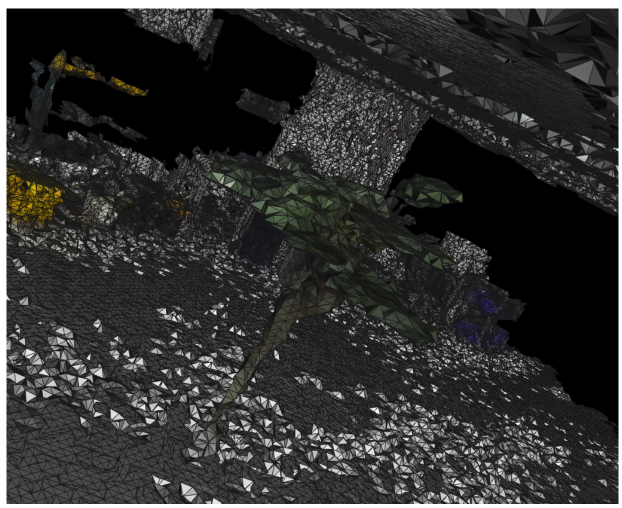
Output Data for ZED Camera Sample 1

Metric	Value
Vertices	**257,558**
Triangles	**509,856**
Has Vertex Colors	True
Has Texture	False
Video Quality	HD1080, 30 FPS
Video Size	10.0 GB
Mesh (.obj) File Size	35.7 MB
Total number of Frames	4706

Results



(a) Without Viewing Adjustment



(b) With Viewing Adjustment

Figure 5: ZED Camera Result 1

Example 2: ZED Camera Sample 2 (Wheelchair Setup)

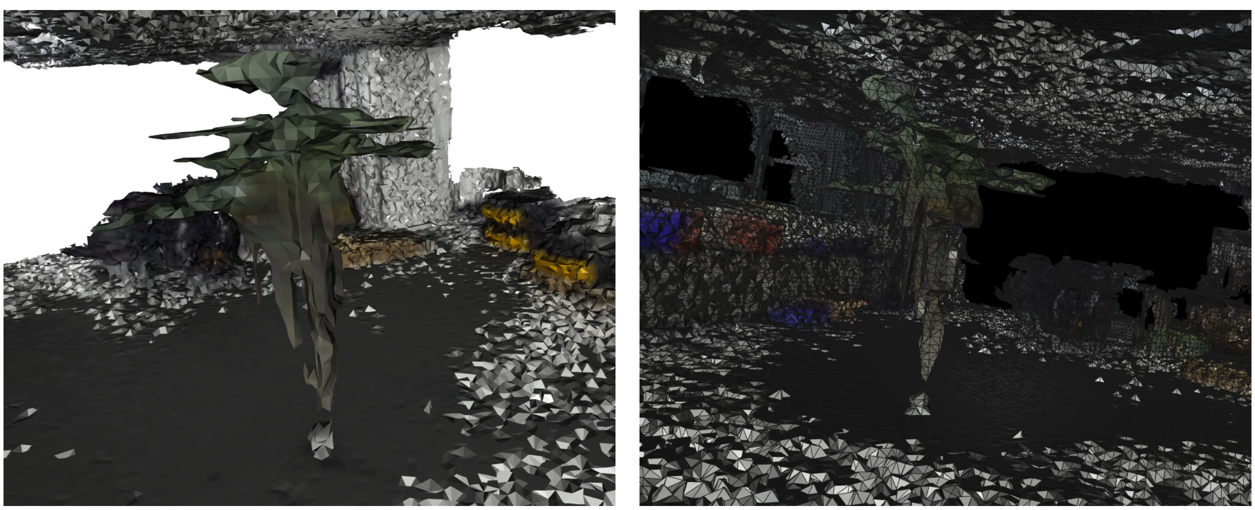
Processing Settings for ZED Camera Sample 2

Parameter	Value
DEPTH_MODE	`NEURAL_PLUS`
COORDINATE_UNITS	`MILLIMETER`
DEPTH_STABILIZATION	50
DEPTH_MIN	600 mm
DEPTH_MAX	2500 mm
SPATIAL_RESOLUTION	`HIGH`
SPATIAL_RANGE	`SHORT`
MAX_MEMORY_MB	8192
SAVE_TEXTURE	True
USE_CHUNK_ONLY	False
REVERSE_VERTEX_ORDER	False
MAP_TYPE	`MESH`
STABILITY_COUNTER	0 (auto)
FILTER_LEVEL	`LOW`
CONFIDENCE_THRESHOLD	77

Output Data for ZED Camera Sample 2

Metric	Value
Vertices	**356,144**
Triangles	**705,715**
Has Vertex Colors	True
Has Texture	False
Video Quality	HD720, 60 FPS
Video Size	6.1 GB
Mesh (.obj) File Size	49.6MB
Total number of Frames	7672

Results



(a) Without Viewing Adjustment

(b) With Viewing Adjustment

Figure 6: ZED Camera Result 2

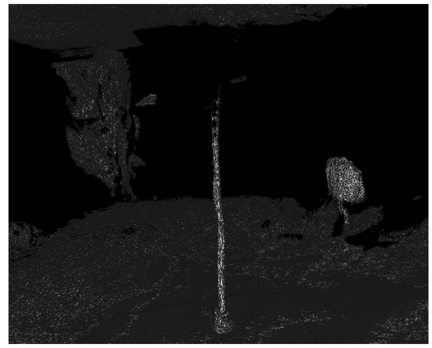
Example 3: iPhone Sample 1

Mode: LiDAR mode

Post Process	
Parameter	**Value**
Voxel Size	3 mm
Max Range	2 m
Smoothing	Off
Simplify	Off

Data Output	
Metric	**Value**
Original Data	961 MB scan, 465 images.
Vertices	**1,894,653**
Triangles	**2,072,418**
Mesh (.obj) File Size	200 MB `.obj` file.

Note: *The area captured covers mostly around the tree, so not much of the background is present unlike ZED. This is taken into account: most of the vertices are for the mesh.*



(a) Mesh viewed on computer

(b) Mesh viewed in iPhone app

(c) Close-up in iPhone app

Figure 7: iPhone LiDAR Mesh Sample 1: (a) Computer, (b) iPhone app, (c) iPhone app close-up

Example 4: iPhone Sample 2

Parameters

Parameter	**Value**
Mode	LiDAR Advanced mode
Confidence	Low
Range	1.5 m
Resolution	5 mm
Smoothing	Off
Simplify	Off

Data Output

Metric	**Value**
Original Data	411 MB, 316 images.
Vertices	**531,408**
Triangles	**840,284**
Mesh (.obj) File Size	60.6 MB `.obj` file.

Note: (this only shows the vertices)
The app exports the faces in a different file and they need to be added on.

(a) Mesh viewed on computer

(b) Mesh viewed in iPhone app

(c) Close-up in iPhone app

Figure 8: iPhone LiDAR Mesh Sample 2: (a) Computer, (b) iPhone app, (c) iPhone app close-up

7 Discussion, Limitations, and Reflection

The ZED camera provides flexibility in parameter adjustment and is suitable for integration with robotics platforms. However, it is sensitive to movement and vibration during capture, and requires careful setup. In practice, it produced lower-quality meshes, but is well-suited for navigation and general scene understanding rather than fine detail capture. The iPhone approach is less sensitive to hand motion, offers a real-time preview, and is easier to use for rapid data collection. Visual and quantitative comparison shows that the iPhone method produces meshes with higher surface detail and vertex count. Both methods face challenges in accurately capturing thin structures such as palm leaves; lowering confidence thresholds can help, but may introduce noise, especially at the base. Environmental lighting is a critical external factor for both methods. Overall, the choice of method depends on setup complexity, portability, processing time, and data requirements.

8 Next Steps (Recommended)

To further improve mesh quality, especially for difficult structures like palm leaves, it is recommended to:

- Test alternative apps or post-processing techniques for mesh refinement.
- Tune capture and processing parameters to optimize detail and reduce noise.
- Apply the best mesh creation approach to a real-world setting, as all experiments were performed under lab conditions (verification stage).

9 App Version and API Documentation

- iPhone LiDAR App: `https://3dscannerapp.com/`
- Stereolabs ZED SDK: `https://www.stereolabs.com/docs`
- Open3D Library: `https://www.open3d.org/docs/release/`

Acknowledgments

This work was completed as part of the first phase of the SSI program. I would like to sincerely thank every member of the labs for their continuous support and collaboration throughout this project. Special thanks to Prof. Shinkyu Park for providing this opportunity.

For any further inquiries, please contact me at: `MahmoudSunbulee@gmail.com`