Summer Mandatory Internship Report

AI-Powered Volume Estimation Pipeline

by:

Zeineb Tekaya

Internship at: Octomiro

Internship Mentor: Mr Montassar Khamassi

IIA 4,

National School of Applied Sciences and Technologies,

University of Carthage

November 5, 2024

Summer Mandatory Internship Report

AI-Powered Volume Estimation Pipeline

---

by:

Zeineb Tekaya

Internship at: Octomiro

Internship Mentor: Mr Montassar Khamassi

Internship Committee's Assessment:

Abstract

This report summarizes my summer 2024 internship experience at Octomiro, a startup company, specifically within their computer vision R&D team. The internship spanned from June 18th to September 9th.

My primary responsibilities involved researching, implementing, and benchmarking nine different depth-based techniques for estimating volume and generating 3D reconstructions from RGB-D images. This research culminated in the design of a structured workflow for a computer vision-powered API service based on the optimal method identified. Furthermore, I implemented a machine learning model with pre-trained image segmentation to enhance accuracy and efficiency. Finally, I developed and deployed the API service, ensuring its reliability through rigorous testing.

A key accomplishment of the internship was achieving a volume calculation method with **92.51%** accuracy. Through this experience, I significantly developed skills in communication, teamwork, problem-solving, time management, back-end development, code compliance, and unit testing. I learned a great deal about computer vision techniques and gained valuable experience in overcoming challenging technical problems.

Overall, this internship at Octomiro provided invaluable practical experience in applying computer vision principles to real-world applications. It significantly enhanced my technical skills and deepened my understanding of the software development lifecycle, contributing substantially to my career goals in the field of computer vision and machine learning.

Acknowledgements

I would like to express my sincere gratitude to Mr. Montassar for his invaluable guidance and unwavering support throughout this internship. His insightful ideas, constant encouragement, and positive attitude made the entire experience both productive and enjoyable. I am also deeply grateful to Mouhieddine Drissi for his exceptional technical expertise and unwavering assistance, particularly during the deployment of my API and workflow. His willingness to share his knowledge and patiently guide me through technical challenges was instrumental to my success. I also extend my heartfelt thanks to Mr. Mohammed Baccar for his infectious enthusiasm and for consistently brightening the workplace, and to Mr. Ikbel Naccache for his invaluable mentorship in improving my soft skills. His guidance on Management 3.0 and effective teamwork provided me with a fresh perspective on my work methodology. Furthermore, I am grateful to Mr. Anis Kacem for his insightful support and guidance, especially during the 3D reconstruction and volume estimation phases of my project. Finally, I want to thank the other interns who contributed to a positive and collaborative work environment, making this internship a truly memorable and enriching experience.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The field of computer vision, a rapidly evolving branch of artificial intelligence, holds immense potential to revolutionize various aspects of our lives [2]. Through advanced algorithms and techniques, particularly those rooted in deep learning [3], computer vision enables machines to "see" and interpret images, unlocking possibilities in diverse areas ranging from medical diagnosis [4] and autonomous vehicles [5] to industrial automation [6] and everyday consumer applications.

My internship focused on the exciting intersection of computer vision and 3D reconstruction, specifically within the R&D team (AI Octobots) at Octomiro. My primary responsibilities involved researching, implementing, and benchmarking nine different depth-based techniques for estimating volume and generating 3D reconstructions from RGB-D images, culminating in the development of a computer vision-powered API service. The OctoVolume project itself was particularly fascinating, demonstrating the ability to accurately calculate the volume of material within a container (with 92.51% accuracy) and generate its 3D reconstruction using only a depth-sensor enabled camera (such as a phone or specialized camera). This technology has potential applications in inventory management for factories, warehouses, and even small businesses, offering rapid results (within 10-15 seconds) and potentially streamlining operations.

My interest in computer vision stems from its transformative potential across diverse sectors. As a computer science student, I am eager to contribute to this burgeoning field and gain practical experience in applying theoretical knowledge to real-world problems. This internship aligned perfectly with my aspirations, offering the chance to delve into the daily life of an engineer within a dynamic startup environment. My primary objectives included acquiring hands-on experience in computer vision techniques, developing coding best practices and complying with industry standards, and learning how to create operational code for seamless integration by others.

Octomiro, a company focused on developing cutting-edge AI-powered solutions, offered an ideal platform for me to achieve these goals. The internship's focus on developing a practical application directly addressed my desire for real-world experience and aligned perfectly with my academic background and career aspirations.

This report documents my internship journey, starting with a comprehensive presentation of Octomiro and its mission in the following chapter. Subsequent chapters will delve into the specifics of my project, methodologies employed, results achieved, and key learning outcomes. This structure aims to provide a clear and insightful account of my internship experience and its contributions to both my personal and professional development.

Chapter 2

Overview of the Host Organization

2.1 Company Background and Mission:

Octomiro is a technology startup focused on revolutionizing enterprise resource planning (ERP) systems by integrating AI and computer vision. The company's core mission is to transform inventory management and enhance operational efficiency by providing businesses with unparalleled accuracy and automation through its innovative solutions. Octomiro firmly believes that computer vision holds the key to bridging the gap between ERP data and operational reality, empowering businesses to make informed decisions based on a precise and real-time understanding of their inventory.

2.2 Core Values and Foundations:

Octomiro's operations are guided by the principles of precision, reliability, and integrity. The company embraces a transparent and collaborative approach, striving to tailor solutions that meet each client's specific needs. Born out of a desire to address persistent limitations in traditional ERP systems, Octomiro's foundation lies in a passion for innovation and a commitment to providing practical solutions to real-world challenges faced by businesses.

2.3 Products and Services:

Octomiro offers a suite of AI-powered solutions designed to enhance inventory management and streamline operational efficiency. These include:

- **OctoPrecision:** Provides a precise and real-time view of inventory, ensuring accurate stock management and informed decision-making.

- **OctoGuard:** Acts as a virtual sentinel, leveraging AI to instantly detect irregularities and non-conformities in products, ensuring compliance with established standards.
- **OctoXact:** An object-counting module capable of accurately quantifying inventory items.
- **OctoNorm:** A module dedicated to identifying deviations from established standards, enabling proactive quality control.
- **OctomiroApp:** A smartphone application providing seamless access to inventory data and insights.
- **OctoHub:** An intelligent vision box installed in transition areas for real-time monitoring and anomaly detection.
- **OctoPipes:** ERP connectors facilitating seamless integration of Octomiro's solutions with existing systems.

2.4 Target Market and Approach:

Octomiro's solutions are targeted towards businesses across various industries seeking to optimize their operations, enhance inventory accuracy, and streamline their logistics. The company takes a consultative approach, working closely with clients to:

- Identify challenges in their ERP processes.
- Develop personalized solutions leveraging computer vision and machine learning algorithms.
- Prepare for ERP onboarding by integrating Octomiro's solutions seamlessly.
- Implement solutions in phases, accompanied by rigorous testing to ensure optimal performance.

2.5 Team and Expertise:

Octomiro's diverse team comprises experts in ERP systems, AI research, software development, and business consulting. This interdisciplinary approach allows the company to offer comprehensive and tailored solutions to its clients. The team's expertise ensures: Unparalleled knowledge of ERP systems to optimize integration with AI solutions, continuous innovation through the exploration of the latest advancements in AI research, robust and functional software development to deliver high-quality solutions and personalized support through experienced consultants who guide clients in utilizing the solutions effectively.

Chapter 3

Specifications/Targeted Objectives

3.1 Project Context and Problem Statement:

The primary focus of my internship project was to develop OctoVolume, a robust and efficient solution for estimating the volume of materials within a container using RGB and depth data. This project aimed to address the need for automated and accurate volume measurement in various industries, including manufacturing, logistics, and inventory management, where manual methods are often time-consuming, prone to errors, and potentially unsafe. The project posed several key challenges:

3.1.1 Segmentation of material and container:

A ccurately separating the material of interest from the container and background in the RGB-D image data was crucial for isolating the volume to be measured. This involved finding a segmentation model that was object-agnostic and could generalize to different types of materials without extensive training.

3.1.2 Point cloud creation and merging:

Converting segmented image data into point clouds representing the material and the container ground and effectively combining them to represent the entire volume posed a technical challenge.

3.1.3 Volume estimation techniques:

Researching, implementing, and benchmarking various techniques for calculating the volume of a point cloud, an area with limited readily available resources, was a central aspect of the project.

3.1.4 3D Reconstruction:

Generating an accurate 3D mesh from the point cloud data required investigating and experimenting with different algorithms to find the most suitable approach for this application.

3.2 Project Objectives and Scope:

The primary objective of this project was to develop a prototype system capable of estimating the volume of various materials contained in a container with the highest achievable accuracy. The initial focus was on experimenting with gravel and sand as test materials, examining the impact of different material shapes on the accuracy of volume estimation.

3.2.1 Scope of Work:

My responsibilities encompassed the entire project lifecycle, from initial research and development to implementation, benchmarking, and deployment. Specific tasks included:

1. Implementing and evaluating different segmentation models.
2. Developing algorithms for point cloud creation and merging.
3. Researching, implementing, and benchmarking nine different volume estimation techniques.
4. Exploring and implementing 3D reconstruction methods.
5. Selecting the most effective techniques and designing a clear workflow.
6. Developing and deploying a RESTful API endpoint for accessing the volume estimation functionality.

3.2.2 Tasks/Deliverables:

1. **Literature Review and Benchmarking of Existing Methods:** Survey existing techniques for segmentation, point cloud processing, volume estimation, and 3D reconstruction.
2. **Implementation of Segmentation Models:** Implement and evaluate different object-agnostic segmentation models for separating materials from containers.

3. **Point Cloud Generation and Processing:** Develop algorithms to create and merge point clouds representing the material and container from segmented images.

4. **Volume Estimation Implementation and Evaluation:** Implement and benchmark at least nine different volume estimation techniques for point cloud data.

5. **3D Reconstruction Implementation:** Implement and evaluate suitable 3D reconstruction algorithms for visualizing the material within the container.

6. **Workflow Design and API Development:** Design a clear and efficient workflow based on the best-performing techniques and develop a RESTful API endpoint for accessing the volume estimation functionality.

7. **Testing and Validation:** Conduct rigorous testing to evaluate the accuracy and performance of the final solution across different materials and container shapes.

8. **Documentation and Reporting:** Document the project's methodology, results, and findings in a comprehensive internship report.

3.3 Expected Outcomes and Deliverables:

3.3.1 Expected Outcomes:

1. Achieving a volume estimation accuracy of at least 90% for various materials and container shapes.

2. Developing a robust and efficient API service for easy integration with other systems.

3. Gaining insights into the effectiveness of different segmentation and volume estimation techniques for this specific application.

3.3.2 Deliverables:

1. A functional prototype system capable of estimating volume from RGB-D data.

2. A documented RESTful API for accessing the volume estimation functionality.

3. A comprehensive internship report detailing the project's methodology, results, and conclusions.

4. Source code for the developed algorithms and API service.

5. A presentation summarizing the project's findings and recommendations.

3.3.3 Evaluation of Success:

The project's success will be measured based on:

- Achieved volume estimation accuracy compared to ground truth measurements.
- Performance and efficiency of the API service (response time, resource utilization).
- The quality and completeness of the documentation and report.
- The effectiveness of the presented solution in addressing the initial problem statement.

Chapter 4

Internship Log

The following Gantt chart provides a visual representation of the internship project's timeline and progress. The project was divided into four main phases:

1. Research and Benchmarking
2. Development and Implementation
3. Deployment and Documentation
4. Testing and Validation

As illustrated in the chart, the project was successfully completed within the internship timeframe, achieving the intended objective of developing a functional and accurate volume estimation system. The Gantt chart provides a clear overview of the project's lifecycle, highlighting the various tasks undertaken, their dependencies, and their completion status.

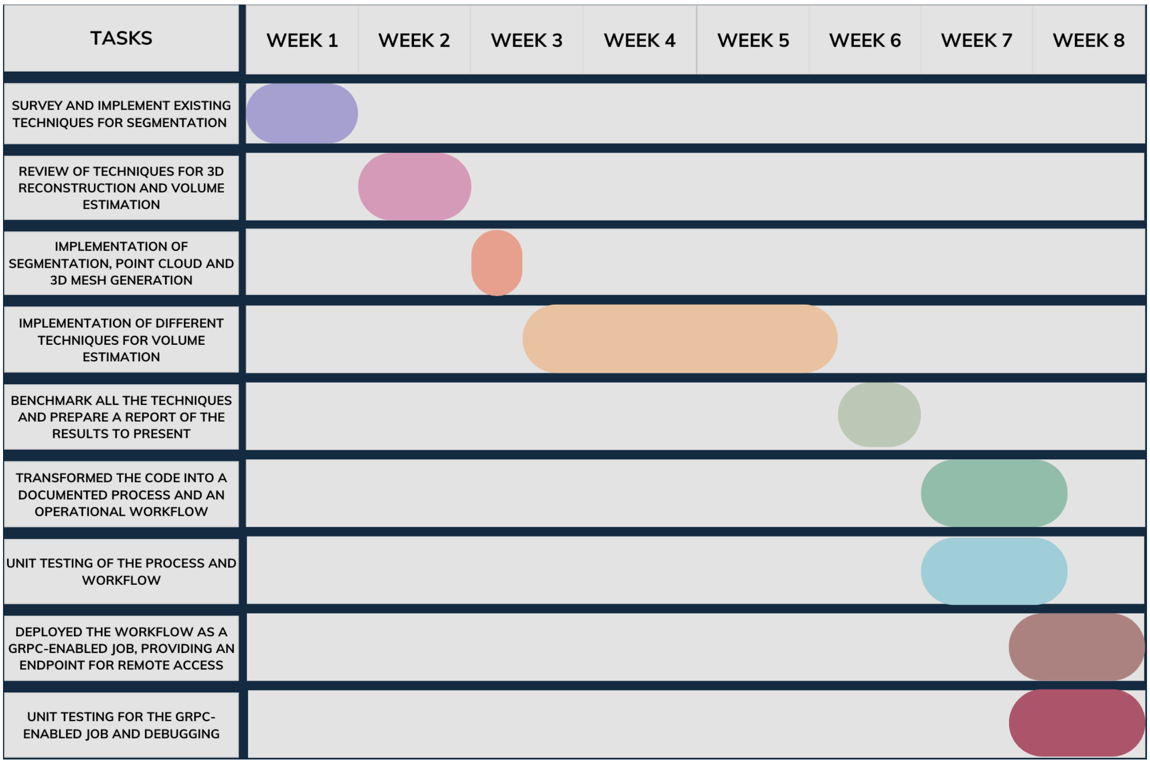

Figure 4.1: Gantt chart of the internship project

Chapter 5

Work Completed

5.1 Introduction

This chapter details the work accomplished during my internship, focusing on developing a robust and efficient system for estimating the volume of materials within containers and generating their 3D reconstructions from RGB-D data. The project's core objectives included achieving accurate volume estimations using object-agnostic segmentation techniques for identifying the material and container boundaries, designing a reliable and performant process, and ultimately creating a deployable API endpoint for accessing this functionality.

The chapter is structured as follows: Section 2 describes the methodologies and tools employed for research, implementation, and development. Section 3 delves into the implementation details of the software and workflow. Section 4 presents the results of the experiments and analysis of the chosen techniques. Finally, Section 5 discusses the challenges encountered and the solutions implemented to overcome them.

5.2 Methodologies and Tools

5.2.1 Research Phase:

- **Segmentation Model Research:** I conducted a literature review to identify suitable object-agnostic segmentation models capable of accurately separating the material from the container in RGB images. The research focused on models that demonstrated high accuracy, generalization capabilities, and efficiency. This included an in-depth investigation of Segment Anything Model (SAM) developed by Meta [1] and its mobile-optimized variant [7].

- **Volume Estimation Techniques Research:** I researched various techniques for calculating the volume of a point cloud, including an enhanced slicing-based approach that approximates the volume by calculating the cumulative area of cross-sectional slices [8], voxel-based approaches, and algorithms specifically designed for irregular shapes like convex hull and Monte-Carlo.

- **3D Reconstruction Research:** I explored different methods for generating 3D meshes from point clouds, including Poisson Surface Reconstruction [9], Screened Poisson Surface Reconstruction [10], SDF-StyleGAN [11], MeshAnything [12], and Blender's API.

5.2.2 Implementation and Benchmarking:

Performance Benchmarking:

A systematic benchmarking process was employed to evaluate the performance of the selected segmentation models MobileSAM and SAM using different prompting techniques. This involved:

- Creating a diverse dataset of RGB-D images of various materials and container shapes captured using the Intel RealSense D435i depth camera.

- Evaluating the models' segmentation performance on the dataset using various prompting methods, including providing point prompts, bounding boxes, and textual descriptions.

- Measuring key performance metrics for each prompting approach:
  - **Inference speed:** This metric reflects the time taken by the model to process an image and generate the segmentation mask for the given prompt.
  - **GPU consumption:** This assessment monitored the amount of GPU memory used during inference for different prompting methods.

- Analyzing the benchmark results to identify the prompting technique that offered the best balance of speed, accuracy, and efficient GPU utilization for this specific application and the more balanced segmentation model out of the two.

A similar benchmarking approach was used to evaluate the performance of the different volume estimation techniques. This involved:

- Generating point cloud data from the segmented images produced by the selected segmentation model.

- Applying each volume estimation technique to the point cloud data and comparing the estimated volume against the manually calculated ground truth volume.
- Calculating performance metrics such as Mean Absolute Error (MAE) and accuracy to assess the accuracy and consistency of each technique. MAE represents the average absolute difference between the estimated and ground truth volumes, while accuracy reflects the percentage of estimations within a certain tolerance range of the ground truth.

Volume Estimation and 3D Reconstruction Implementation:

Various techniques for volume estimation and 3D reconstruction from point cloud data were implemented using Python and relevant libraries. This involved:

- Implementing algorithms for point cloud creation from depth images using the realsense2 library, which provides tools for working with Intel RealSense cameras.
- Exploring and implementing different techniques for point cloud processing, including filtering, downsampling, and surface normal estimation, using the Open3D library.
- Implementing different algorithms for volume estimation, including the slicing-based method, voxel-based methods, Monte-Carlo and convex hull calculations, utilizing Open3D and NumPy for numerical operations.
- Implementing different approaches for 3D reconstruction, focusing on Poisson surface reconstruction due to its efficiency and ability to handle noisy point cloud data, using Open3D.

Code Refinement and Documentation:

The code for each implementation was thoroughly reviewed and refined to ensure clarity, maintainability, and adherence to coding standards. This involved

- Applying consistent formatting and naming conventions to improve code readability.
- Adding comments to explain complex logic and document design decisions.
- Organizing the code into modular functions and classes to improve reusability and maintainability.
- Encapsulating the entire volume estimation process within a well-structured Python class to promote code organization and facilitate integration into the workflow.

Workflow Development with Octopipes:

The chosen volume estimation and 3D reconstruction techniques were integrated into a cohesive workflow using the Octopipes library, which facilitated the definition and execution of multi-step processes.

Custom handlers were developed for each step in the workflow to manage data input, processing, and output.

The complete workflow was thoroughly tested using various unit tests to ensure its reliability and accuracy with the pytest library.

A job was created for the volume estimation workflow using a job scheduler, enabling automated execution and facilitating the creation of a gRPC API endpoint for remote access.

The necessary changes were made to the Protocol Buffer (protobuf) file to define the input and output data structures for the gRPC API.

The job and the API endpoint were extensively tested to ensure their proper functioning and to address any identified bugs or issues.

5.3 Implementation and Development

5.3.1 Software Architecture:

The core functionality of the volume estimation and 3D reconstruction system was implemented within a modular Python class named VolumeEstimator. This class encapsulates the methods for loading RGB-D data, performing segmentation using the chosen model, processing the point cloud, estimating the volume, and generating the 3D reconstruction. The class interacts with external libraries like Torch for segmentation, Open3D for point cloud processing and 3D reconstruction, and NumPy for numerical operations.

5.3.2 Module Interactions:

The VolumeEstimator class serves as the central component of the system. Its methods are called in a specific sequence within the Octopipes workflow to perform the end-to-end volume estimation process. The workflow is defined using Octopipes' pipeline construction features, and custom handlers are created for each step to manage data input, processing, and output.

5.3.3 Design Decisions:

- **Segmentation Model and Prompting:** MobileSAM was selected for segmentation due to its balance between accuracy and computational efficiency. Additionally, point prompting was chosen as the preferred method for interacting with the model. This decision was made based on the benchmarking results and considerations for user experience, as point prompts provide a more intuitive and practical interaction method for users of the mobile app that will host OctoVolume.

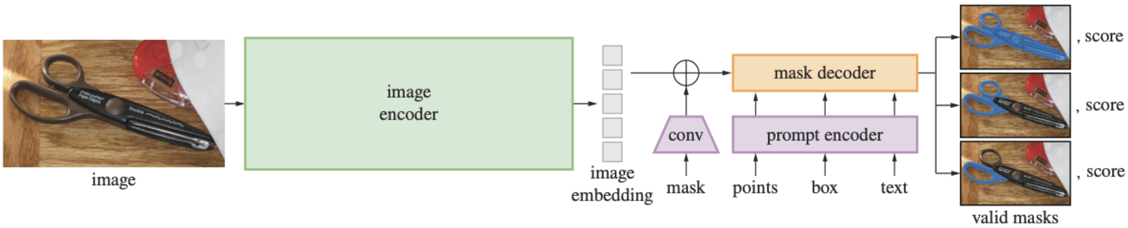

Figure 5.1: SAM overview. A heavyweight image encoder outputs an image embedding that can then be efficiently queried by a variety of input prompts to produce object masks at amortized real-time speed.[1]

- Opting for Poisson surface reconstruction due to its speed and robustness in handling noisy point cloud data.
- Implementing a modular class structure to improve code organization, maintainability, and reusability.
- Using Octopipes to define and manage the workflow for flexibility and ease of debugging.

5.3.4 Version Control and Code Management:

Git was used as the version control system to track code changes, facilitate collaboration, and manage different versions of the software. A consistent branching strategy was followed to ensure code stability and manage new feature development.

5.4 Results and Analysis

5.4.1 Dataset and Experimental Setup:

The volume estimation and 3D reconstruction techniques were evaluated using a dataset consisting of four RGB-D images: three depicting sand in different shapes and configurations within a container (Figure 5.3), and one depicting gravel (Figure 5.4). The ground truth volumes for each material configuration were calculated manually to serve as reference values for evaluating the accuracy of the algorithms. Data acquisition was performed using an Intel RealSense D435i depth camera (Figure 5.2), providing synchronized RGB and depth information.

Figure 5.2: Intel Realsense D435i

5.4.2 Segmentation Model Selection:

Both the standard Segment Anything Model (SAM) and its mobile-optimized version (MobileSAM) were implemented and evaluated for their segmentation performance. Both models produced satisfactory segmentation results for separating the material from the container in the RGB images. However, MobileSAM was selected for further experiments due to its significantly lower computational resource requirements (3GB GPU memory usage compared to 9GB for the standard SAM), which aligned better with the project's goal of creating an efficient and deployable solution.

5.4.3 Point Cloud Processing and 3D Reconstruction:

Following segmentation, the masked depth data was extracted to create point cloud representations of the material and the container ground. The point clouds were then processed to generate a complete representation of the material pile (Figure 5.5). This involved transposing and generating new points to recreate the bottom surface of the material, merging the point clouds, and interpolating new points to fill in gaps and create a smoother surface. Poisson surface reconstruction was chosen for generating the 3D mesh due to its lower computational cost and relatively fast execution time compared to other methods like MeshAnything (Figure 5.6).

Figure 5.3: RGB images of sand in three different configurations (from left to right): evenly distributed, with two mounds, with a single mound.

Figure 5.4: RGB image of the gravel

Figure 5.5: Illustrating point cloud processing steps for sand with two mounds(from left to right): Top surface of the sand point cloud, entire sand pile point cloud.

5.4.4 Volume Estimation Techniques Evaluation:

Nine different volume estimation techniques were implemented and evaluated using the processed point cloud data:

1. Voxelization with merged mesh of the material's bottom and top surfaces.
2. Voxelization of the entire point cloud of the material.
3. Convex hull on the point cloud.
4. Convex hull on the merged mesh.
5. Dynamic slicing method.
6. Monte Carlo on the merged mesh with oriented bounding box (OBB).
7. Monte Carlo on the merged mesh with axis-aligned bounding box (AABB).
8. Monte Carlo on the point cloud with OBB and AABB.
9. Monte Carlo on the point cloud with AABB.

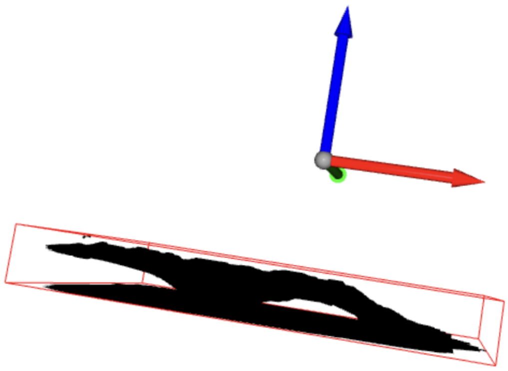

Figure 5.6: Merged meshes of the bottom and the top surface of the sand.

Figure 5.7: (Left) 3D reconstruction of the sand pile with RGB texture mapped onto the point cloud. (Right) Original RGB image of the sand in configuration with two mounds. The 3D reconstruction accurately captures the shape and texture of the sand, particularly the distinct mounds and the variations in color across the pile.

The accuracy of the nine volume estimation techniques was assessed using a dataset of four RGB-D images, capturing sand in three distinct configurations and gravel. The manually measured volumes for each material arrangement, serving as ground truth references, are presented in Table 5.1. The estimated volume obtained using each technique is displayed in Table 5.2, alongside the Mean Absolute Error (MAE), calculated as the average absolute difference between the estimated and reference volumes. A higher accuracy corresponds to a lower MAE. Additionally, Table 5.3 presents the accuracy of each technique, defined as the percentage of estimated volumes within a specified tolerance of the reference volume.

The results of the volume estimation experiments, summarized in Tables 5.2 and 5.3, reveal that three techniques consistently exhibit strong performance: Voxelization of the merged mesh, Convex Hull on the merged mesh, and Voxelization of the entire point cloud. To further illustrate these findings, 5.8 visually depict the MAE and accuracy values for each technique across the test cases.

Analyzing these results, it's evident that voxelization-based techniques, particularly

Table 5.1: Reference volumes of the materials used for evaluation.

Material	Evenly distributed sand (1)	Sand with two mounds (2)	Sand with a single mound (3)	Gravel (4)
Reference Volume (cm^3)	750	800	850	1100

Table 5.2: Volume Estimation Results: Estimated Volume (cm^3) and Mean Absolute Error (MAE).

Technique	1	2	3	4	MAE (cm^3)
Voxelization of 2 meshes	749	746	827	1122	**76.5**
Convex Hull on the entire point cloud	590	723	758	947	183
Convex Hull on merged mesh	652	860	927	1170	**35.25**
Dynamic slicing to rectangles on merged mesh	579	620	717	1120	178.5
Voxelization of the entire point cloud	757	785	813	1116	**69.75**
Monte-Carlo on top surface mesh & OBB	661	815	867	717	181
Monte-Carlo on top surface mesh & AABB	535	682	626	960	236.75
Monte-Carlo on merged mesh & AABB	524	561	645	830	297.5
Monte-Carlo on merged mesh & OBB	404	583	299	858	401.5

"Voxelization of the entire point cloud" demonstrate the highest overall accuracy (92.51%) and the second-lowest MAE (69.75 cm^3). This technique involves voxelizing the entire bounding box encompassing the point cloud of the material pile. Each voxel is then analyzed to determine whether it contains points representing the material. By summing the volume of all voxels containing material points, the technique provides a robust estimation of the total material volume. This robust performance can be attributed to the method's ability to closely approximate the shape of the material pile by dividing it into fine-grained volumetric units (voxels). The high resolution achieved by this approach allows for a more accurate representation of complex geometries, leading to more precise volume estimations.

The Voxelization of the merged mesh technique also performs admirably (91.66% accuracy), indicating that separately processing the top and bottom surfaces of the material before voxelization can effectively capture the overall shape and yield accurate volume estimates. Similarly, the "Convex Hull on the merged mesh" technique demonstrates competitive performance (91.18% accuracy), likely because it creates a tight-fitting geometric hull around the merged point cloud, providing a reasonable approximation of the material's volume.

Table 5.3: Accuracy (%) of volume estimation techniques across four test cases.

Technique	1	2	3	4	Avg Accuracy
Voxelization of 2 meshes	88,11%	87,76%	97,29%	93,5%	**91,66%**
Convex Hull on the entire point cloud	69,41%	85,05%	89,17%	78,91%	80,64%
Convex Hull on merged mesh	76,70%	98,83%	91,69%	97,5%	**91,18%**
Dynamic slicing to rectangles on merged mesh	68,11%	72,94%	84,35%	93,33%	79,68%
Voxelization of the entire point cloud	89,05%	92,35%	95,64%	93%	**92,51%**
Monte-Carlo on top surface mesh & OBB	77,76%	95,88%	98,03%	59,75%	82,85%
Monte-Carlo on top surface mesh & AABB	62,94%	80,23%	73,64%	80%	74,20%
Monte-Carlo on merged mesh & AABB	61,64%	66%	75,88%	69,16%	68,17%
Monte-Carlo on merged mesh & OBB	47,52%	68,58%	35,17%	71,5%	55,69%

While the other techniques, including dynamic slicing and Monte Carlo methods, produced reasonably accurate results for certain material configurations, they exhibited greater variations in performance across the test cases. These inconsistencies might stem from limitations in their ability to accurately capture intricate surface details, especially when dealing with irregular shapes or variations in material density.

The analysis of these results confirms the effectiveness of voxelization as a reliable and robust technique for volume estimation, particularly when applied to the entire point cloud representation of the material.

5.4.5 Selection of Optimal Technique and Process Refinement:

Based on the comprehensive benchmarking results (Tables 5.2 and 5.3, and Figures 5.8), "Voxelization of the entire point cloud" was selected as the optimal technique for volume estimation. While other methods, including "Voxelization of the merged mesh" and "Convex Hull on merged mesh", demonstrated strong performance in some cases, "Voxelization of the entire point cloud" achieved the highest average accuracy (92.51%) across all tested material configurations and the second-lowest average MAE (69.75 cm^3), indicating its

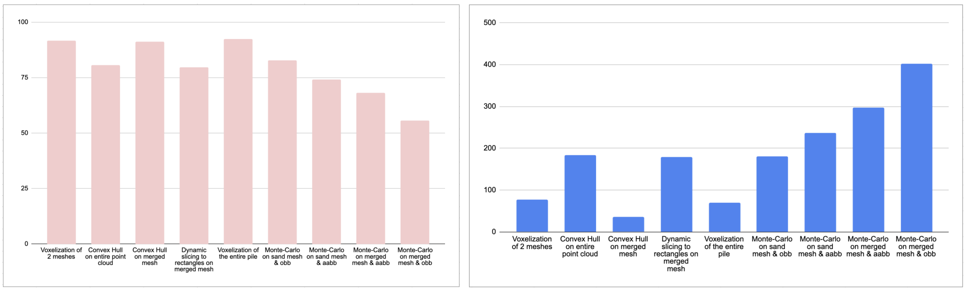

Figure 5.8: Performance of volume estimation techniques: (from left to right) Average Accuracy and Mean Absolute Error (MAE).

robust and consistent performance.

As previously described, this technique involves the following steps:

1. **Bounding Box Generation:** The bounding box of the point cloud representing the material pile is determined.
2. **Voxelization:** A 3D grid with evenly spaced volumetric units (voxels) is created, encompassing the entire bounding box.
3. **Voxel Occupancy Check:** For each voxel in the grid, the algorithm checks whether it contains any points from the material's point cloud.
4. **Volume Calculation:** The volume is calculated by summing the volume of all the voxels identified as containing material points.

The fine-grained resolution provided by voxelization enables this method to accurately represent the complex and irregular shapes of the materials, leading to more precise volume estimations.

To illustrate the complete workflow for volume estimation and 3D reconstruction, a simplified flowchart is presented in Figure 5.9. This flowchart depicts the key stages of the process, highlighting the data transformations and the core operations involved at each step.

The entire workflow, encompassing the stages illustrated in Figure 5.9, was integrated into a modular and well-structured Python class. This class contains methods for each processing step, from loading the RGB-D image to performing segmentation, point cloud processing, volume estimation, 3D reconstruction, and texture mapping. Extensive use of docstrings and comments within the code ensures readability, maintainability, and ease of understanding for future development or modification. The modular design facilitates the adaptation of the workflow for different applications or materials by allowing specific components to be easily modified or replaced without affecting the overall structure.

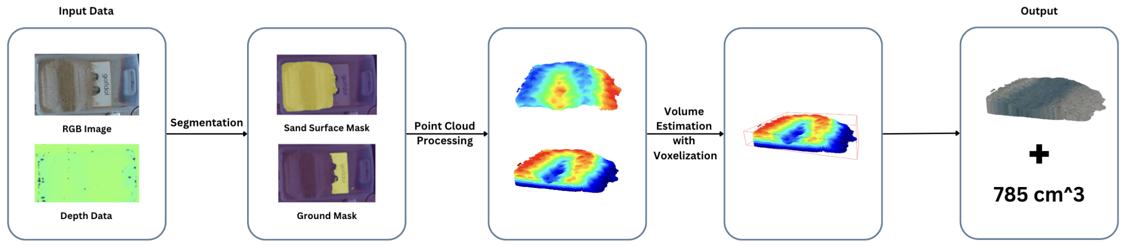

Figure 5.9: Flowchart of the volume estimation and 3D reconstruction process.

5.5 Challenges and Solutions

The development and implementation of this volume estimation and 3D reconstruction system presented several notable challenges. These challenges ranged from technical difficulties inherent to the specific problem domain to issues related to deploying a robust and scalable solution using gRPC. This section elaborates on these challenges, the approaches employed to overcome them, and the valuable insights gained throughout this process.

5.5.1 Limited Bibliography and Process Optimization:

One of the most significant challenges encountered was the scarcity of readily available literature on volume estimation using RGB-D images. This required extensive research and exploration to identify suitable methodologies and adapt them to this specific application.

The process of enhancing and optimizing the pipeline for higher accuracy also presented a considerable hurdle. Specifically, denoising the point clouds and effectively removing outliers were crucial to improving the fidelity of the volume estimations. Experimentation with various denoising algorithms, outlier removal techniques, and filtering parameters was necessary to find an optimal balance between noise reduction and the preservation of critical geometric features.

Additionally, determining the most suitable voxel size for the chosen technique ("Voxelization of the entire point cloud") was crucial for achieving accurate volume estimates. Both static and dynamic voxel sizing strategies were explored, involving testing different values to determine the configuration that consistently produced the most accurate results across the test cases.

5.5.2 RGB Color Interpolation for 3D Reconstruction:

Mapping the RGB color information from the original 2D image onto the reconstructed 3D material pile was another significant challenge. The interpolation of color values from the 2D image to the 3D point cloud, while maintaining visual fidelity and ensuring accurate alignment with the reconstructed geometry, required meticulous implementation and optimization. This involved a deep understanding of projective geometry, careful handling of color data, and experimenting with different interpolation techniques to achieve a realistic and visually pleasing textured 3D model.

5.5.3 gRPC Implementation and Data Handling:

Developing the gRPC API and managing data efficiently for deployment posed additional challenges. My limited prior experience with gRPC necessitated extensive research and familiarization with the framework's concepts, protocols, and implementation details. This involved studying gRPC documentation, consulting online resources, and experimenting with example implementations to gain a strong understanding of how to create efficient and scalable gRPC services. Moreover, specific challenges arose when:

- **Adding messages for intrinsics and depth data to the Protobuf file:** Defining appropriate data structures and ensuring consistent encoding and decoding of these specialized data types for communication between the client and server.
- **Writing methods to handle data from the S3 bucket and pass it to the workflow:** Effectively retrieving the data from the cloud storage, transforming it into the format required by the volume estimation workflow, and seamlessly integrating this data retrieval process within the gRPC service.

I sought assistance from Mouhieddine, the startup's CTO, who provided valuable guidance and orientation during these phases. His expertise in gRPC and distributed systems was instrumental in successfully navigating these complex technical challenges.

5.5.4 Reflections on Challenges and Learning Outcomes:

These challenges reinforced the importance of being adaptable and resourceful in tackling complex problems. This experience taught me to be comfortable exploring unfamiliar technologies, diligently researching and understanding new concepts, and leveraging available resources effectively. By facing these difficulties head-on, I gained invaluable experience in:

- Conducting thorough research to address gaps in existing literature.
- Experimenting with various techniques to optimize processes for accuracy and efficiency.
- Collaborating with experts to navigate complex technical hurdles.

- Persevering in challenging situations and learning through practical implementation.

Moreover, the project emphasized the importance of developing code with a focus on reusability and future maintainability. This understanding was applied while:

- Creating generic handlers within the Octopipes workflow, ensuring they can be adapted for future projects or modified easily for different material types or applications.
- Writing general-purpose methods for handling depth data and camera intrinsics from the S3 bucket, making this functionality readily available for other tasks or projects that might require similar data access and processing.

Through these challenges and the solutions devised, I not only successfully developed and deployed a functional volume estimation and 3D reconstruction system but also gained a deeper understanding of computer vision, 3D processing, and distributed systems, enhancing my skillset and solidifying good software development practices.

Chapter 6

Conclusion

This internship at Octomiro, focused on developing OctoVolume, a robust and efficient volume estimation system using RGB-D data, has proven to be an invaluable learning experience. I was fully immersed in the development cycle of a real-world project within a supportive and dynamic environment alongside the AI Octobots team.

The successful implementation of the system, achieving an average accuracy of 92.51% using the "Voxelization of the entire point cloud" technique, showcases the viability of this approach for practical volume measurement applications. This project significantly enhanced my technical skill set, equipping me with valuable proficiency in computer vision algorithms, 3D point cloud processing, and the effective use of Python libraries such as Open3D and OpenCV.

Overcoming various technical challenges during the internship, particularly those related to point cloud denoising, color interpolation for 3D reconstruction, and developing the gRPC API, strengthened my analytical and problem-solving abilities. These hurdles also highlighted the importance of continuous learning and resourcefulness, inspiring me to confidently explore unfamiliar technologies like gRPC and refine my understanding of core concepts through hands-on implementation.

Working collaboratively with the R&D team, particularly benefiting from the guidance of CTO Mouhieddine on backend development aspects, further solidified the value of clear communication, effective teamwork, and knowledge sharing within a professional environment. This experience not only provided practical insights into the workings of a dynamic tech startup but also confirmed my passion for the field, laying a strong foundation for a future career in computer vision and software development.

The developed OctoVolume prototype holds substantial potential for enhancing Octomiro's product offerings and tackling real-world problems across various sectors where precise volume measurements are essential. Further refining and validating the system with diverse, real-world data, potentially exploring advanced point cloud processing and alternative 3D reconstruction techniques, would further enhance its robustness and generalizability.

I am deeply grateful to the entire Octomiro team for their exceptional mentorship, shared knowledge, and the encouraging environment they cultivated, allowing me to gain a wealth of practical skills and strengthen my passion for computer vision. In particular, I would like to express my sincere gratitude to Mr. Montassar, head of the R&D team and COO of Octomiro, for providing me with the freedom and resources to explore any idea that arose, encouraging me at every step, and demonstrating unwavering patience and trust in my capabilities. His belief in me, particularly allowing me to tackle the workflow and backend aspects independently, provided invaluable opportunities to learn and grow throughout the internship. As I continue my journey, I am eager to expand my knowledge into areas such as system design and backend engineering to become a well-rounded and adaptable software developer.

Bibliography

[1] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., *Segment anything*, *arXiv preprint arXiv:2304.02643* (2023).

[2] R. Szeliski, *Computer Vision: Algorithms and Applications.* Springer Science & Business Media, 2010.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT Press, 2016.

[4] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, *A survey on deep learning in medical image analysis*, *Medical image analysis* **42** (2017) 60–88.

[5] J. Janai, F. Güney, A. Behl, and A. Geiger, *Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art*, *Foundations and Trends in Computer Vision* **12** (2020), no. 1-3 1–308.

[6] J. Kirschke, S. Hörmann, and M. Hensel, *Challenges and opportunities of artificial intelligence for industrial automation*, *Procedia CIRP* **72** (2018) 1076–1081.

[7] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., *Segment anything model (sam) on mobile devices*, *Meta AI Blog Post* (2023).

[8] X. Meng, T. Wang, D. Cheng, W. Su, P. Yao, X. Ma, and M. He, *Enhanced point cloud slicing method for volume calculation of large irregular bodies: Validation in open-pit mining*, *Remote Sensing* **15** (2023), no. 20 5006.

[9] M. Kazhdan, M. Bolitho, and H. Hoppe, *Poisson surface reconstruction*, *ACM Transactions on Graphics (TOG)* **22** (2006), no. 3 61–70.

[10] M. Kazhdan and M. Bolitho, *Screened poisson surface reconstruction*, *ACM Transactions on Graphics (TOG)* **32** (2013), no. 3 1–13.

[11] X. Zheng, P. Xu, Q. Zhang, C. Zhou, M. Wu, Z. Fang, J. Ma, J. Huang, and P. Torr, *Sdf-stylegan: Implicit sdf-based stylegan for 3d shape generation*, *arXiv preprint arXiv:2206.12055* (2022).

[12] L. Gao, J. Wu, S. Tang, S. Hu, W. Xue, A. Zeng, X. Zhou, R. Wang, C. Sun, and T. Fang, *Meshanything: Artist-created mesh generation with autoregressive transformers*, *arXiv preprint arXiv:2406.10163* (2024).