UNIVERSITÀ DEGLI STUDI DI TRIESTE

LAUREA MAGISTRALE IN

DATA SCIENCE AND
SCIENTIFIC COMPUTING

Comparative Evaluation of Multi-View Stereo 3D Reconstruction Techniques on Specialized Dataset

Supervisor:
Prof. Felice Andrea Pellegrino

Student:
Matteo Boi

Accademic Year 2022/2023

UNIVERSITÀ DEGLI STUDI DI TRIESTE

LAUREA MAGISTRALE IN

DATA SCIENCE AND
SCIENTIFIC COMPUTING

Valutazione Comparativa di Tecniche di Ricostruzione 3D Stereo Multi-Vista su Dataset Specializzato

Supervisore:
Prof. Felice Andrea Pellegrino

Laureando:
Matteo Boi

ANNO ACCADEMICO 2022/2023

What does it mean, to see? The plain man's answer (and Aristotle's, too) would be, to know what is where by looking. In other words, vision is the process of discovering from images what is present in the world, and where it is.

— David Marr

ABSTRACT

In the rapidly evolving field of computer vision, Multi-View Stereo (MVS) reconstruction remains a critical challenge, particularly in the context of 3D object reconstruction. This thesis presents a comparative study of three leading MVS reconstruction techniques—COLMAP, GBi-Net, and MVSFormer—applied to a novel dataset obtained via a UR10e collaborative robot (cobot) capturing images of various 3D printed objects. This work not only explores traditional and learning-based MVS methods but also introduces an innovative dataset acquisition approach that leverages the precision of cobots and the versatility of 3D printing technology.

Through rigorous experimentation and analysis, we evaluated the performance of each method in terms of depth map accuracy and the quality of the reconstructed point clouds. Our findings reveal significant distinctions among the techniques, with MVSFormer demonstrating superior performance in capturing detailed and smooth depth maps, while GBi-Net excelled in generating more complete but slightly less accurate reconstructions compared to COLMAP, which struggled with completeness yet provided precise estimations in textured areas. This study underscores the potential of transformer-based models in MVS reconstruction, particularly for industrial applications where accuracy and detail are paramount.

Additionally, this research introduces an exploration into enhancing MVS reconstruction through the integration of artificially textured patterns on objects, simulating laser projector outputs. This novel approach aimed to address the challenges posed by textureless surfaces, a common issue due to the objects' inherent dark and color uniformity. By implementing simulated laser patterns, the study sought to improve depth estimation accuracy and the overall quality of 3D reconstructions. The findings confirm the effectiveness of this strategy, suggesting that such texturing techniques, potentially applied via actual laser projectors, could significantly advance 3D reconstruction processes.

Furthermore, the thesis proposes a scalable and adaptable framework for acquiring training data, emphasizing the accessibility and cost-effectiveness of using 3D models for ground truth generation and 3D printed objects for image capture. This approach not only facilitates the generation of high-quality datasets tailored to specific MVS challenges but also opens avenues for further research and application in various fields requiring precise 3D reconstructions.

In conclusion, this research contributes to the ongoing dialogue in the computer vision community about the efficacy of different MVS methodologies and the development of more efficient and accessible data acquisition techniques. Our comparative analysis and dataset generation strategy offer valuable insights and tools to advance the state-of-the-art in 3D reconstruction technologies.

SOMMARIO

Nel campo della visione artificiale, la ricostruzione stereo multi-vista (MVS) rimane una sfida critica, in particolare nel contesto della ricostruzione di oggetti 3D. Questa tesi presenta uno studio comparativo di tre tecniche nella ricostruzione MVS: COLMAP, GBi-Net e MVSFormer. Queste sono applicate a un nuovo dataset ottenuto tramite un robot collaborativo UR10e di vari oggetti stampati in 3D. Questo lavoro esplora metodi MVS tradizionali e basati sull'apprendimento, introducendo inoltre un approccio innovativo all'acquisizione del dataset, sfruttando la precisione dei cobot e la versatilità della tecnologia di stampa 3D.

Attraverso analisi e sperimentazioni, abbiamo valutato le prestazioni di ciascun metodo in termini di accuratezza delle mappe di profondità e qualità delle nuvole di punti ricostruite. I risultati rivelano distinzioni significative tra le tecniche, con MVSFormer che dimostra una prestazione superiore nella cattura di mappe di profondità dettagliate e uniformi, mentre GBi-Net eccelle nella generazione di ricostruzioni più complete, sebbene leggermente meno accurate. COLMAP, invece, fatica con la completezza ma fornisce stime precise nelle aree in cui sono presenti textures. Questo studio sottolinea il potenziale dei modelli basati su transformers nella ricostruzione MVS, in particolare per applicazioni industriali dove l'accuratezza e il dettaglio sono fondamentali.

Inoltre, questo lavoro introduce un'esplorazione volta a migliorare la ricostruzione MVS attraverso l'integrazione di pattern testurizzati sugli oggetti, simulando l'utilizzo di proiettori laser. Questo approccio mira ad affrontare le sfide poste dalle superfici prive di texture, un problema causato dell'oscurità intrinseca degli oggetti e dall'uniformità di colore. Questo approccio cerca di migliorare l'accuratezza nella stima della profondità e la qualità delle ricostruzioni 3D. I risultati confermano l'efficacia della strategia, suggerendo che tali tecniche di testurizzazione, potenzialmente applicate tramite proiettori laser reali, potrebbero significativamente avanzare il processo di ricostruzione 3D.

La tesi propone un framework scalabile e adattabile per l'acquisizione di dati di addestramento, enfatizzando l'accessibilità e l'efficacia dei costi nell'uso di modelli 3D per la generazione di verità di fondo e oggetti stampati in 3D per la cattura delle immagini. Questo approccio facilita la generazione di dataset di alta qualità su misura per specifiche sfide MVS, e apre le porte per ulteriori ricerche e applicazioni.

In conclusione, questo lavoro contribuisce al dialogo in corso nella comunità scientifica riguardo l'efficacia delle diverse metodologie MVS e lo sviluppo di tecniche di acquisizione dati più efficienti e accessibili. La nostra analisi comparativa e la strategia di generazione del dataset offrono intuizioni e strumenti per avanzare lo stato dell'arte nelle tecnologie di ricostruzione 3D.

CONTENTS

LIST OF FIGURES

ACRONYMS

LIDAR Light Detection and Ranging
MVS Multi-View Stereo
CNNs Convolutional Neural Networks
RNNs Recurrent Neural Networks
FPNs Feature Pyramid Networks
RGB Red Green Blue
EPE End Point Error
SIFT Scale Invariant Feature Transform
tf-idf term frequency-inverse document frequency
SVD Singular Value Decomposition
PLA Polylactic Acid
RANSAC RANdom SAmple Consensus
DCNs Deformable Convolutional Networks
MLPs Multi-Layer Perceptrons
ETB Error Tolerance Bins
ViTs Vision Transformers
MAE Mean Absolute Error
RMSE Root Mean Squared Error
DEP Depth Error Percentage
DNNs Deep Neural Networks
NLP Natural Language Processing

1 INTRODUCTION

The task of creating 3D models from images is a critical challenge in photogrammetry and computer vision. The objective is to extract meaningful geometric and semantic details from photographic images. The importance of converting the physical world into 3D digital forms spans across various domains including, but not limited to, quality control monitoring, conservation of cultural heritage, enhancement of digital maps and navigation systems, immersive experiences in virtual tourism and gaming, advancements in mixed reality technologies, and the functionality of autonomous robotics.

Different uses necessitate specific requirements from a 3D modeling system; for instance, tasks focused on inspection demand precise geometry reconstructions for accurate measurements, whereas applications designed towards visualization prioritize the aesthetic and realistic portrayal of models over their geometric precision.

Numerous technologies enabling 3D modeling have emerged to meet these diverse requirements, each with its own set of advantages and limitations depending on the application context. Broadly, these methods are divided into active techniques (like Light Detection and Ranging (LIDAR) and radar) and passive techniques (involving camera use). Cameras, as passive devices, are noted for their energy efficiency and non-reliance on direct contact with physical objects. Additionally, the widespread integration of camera technology into consumer devices has dramatically reduced the cost of camera equipment in recent years. This trend has led to a surge in visual data, with almost everyone possessing a digital camera, thereby contributing to a vast repository of visual records of the world. This ever-growing collection, stored on personal devices and cloud services, offers a dynamic digital snapshot of our environment. Leveraging this extensive and varied visual data for 3D modeling unveils tremendous opportunities but also presents considerable challenges for image-based 3D reconstruction systems.

Fundamentally, an image-based 3D reconstruction system seeks to emulate the spatial perception of the human visual system to deduce 3D structures from 2D images captured by cameras. Humans intuitively construct a detailed 3D perception of their surroundings effortlessly; however, distilling this process into a computational algorithm proves to be a challenging task. Despite extensive research leading to a comprehensive understanding of many aspects of this field, fully grasping and translating this knowledge into practical, robust algorithms remains difficult to achieve. The conventional strategy involves breaking down the reconstruction process into smaller, manageable tasks. However, the inherent limitations in deducing accurate 3D models from 2D representations often

necessitate the integration of assumptions about the physical world into the reconstruction algorithms.

The main challenge involves the process of depth estimation, often referred to as stereo matching, where the goal is to find corresponding pixels across two or more images capturing the same scene. Due to issues like occlusions and variations in surface appearance when viewed from different angles, stereo matching poses a complex, ill-defined problem. A critical aspect of establishing pixel matches is the consistency in visual appearance across images, which depends on factors like the materials' properties, lighting conditions, and other specifics of the scene capture process.

To measure the degree of similarity between corresponding pixels, a photo-consistency metric is employed, calculating the matching cost through various methods such as the sum of absolute differences or the normalized cross-correlation, among others [55]. Stereo matching techniques are broadly divided into local methods, which compute and aggregate costs using pixel neighborhoods, and global methods, which formulate an energy function for the entire image to maintain spatial consistency.

Once correspondences are accurately established, it's possible to determine the depth for each pixel, facilitating the projection of these pixels into 3D space. However, surfaces with special characteristics, such as non-reflective or texture-poor areas, pose significant challenges to establishing unique solutions.

MVS techniques aim to create a comprehensive 3D model of a scene using several overlapping images taken from various viewpoints. These methods extend beyond basic stereo matching by incorporating numerous images, thus leveraging redundancy for improved cost aggregation and depth estimation. While classic stereo-matching techniques often work within narrow baselines, MVS algorithms excel in handling images from dramatically different viewpoints, overcoming the limitations associated with geometric and photometric distortions and occlusions.

Over years of dedicated research, MVS algorithms have evolved considerably. While they were initially categorized based on scene representation or reconstruction algorithms, modern approaches often combine elements from multiple categories.

This thesis takes a detailed look at MVS reconstruction, comparing traditional and modern 3D modeling methods. It's organized to guide readers from basic principles to more complex techniques, aiming to give a thorough overview of the topic.

Chapter 2 sets the stage, introducing the thesis' objectives and providing a bird's-eye view of the multi-view stereo framework. It delves into the complexities of depth estimation and the challenges inherent in stereo matching, laying the groundwork for the detailed exploration that follows. In addition, it makes a distinction between traditional methods and learning-based techniques. It offers an overview of depth map-based methods and established benchmark datasets like DTU Robot Image Data

Sets, Tanks and Temples, and BlendedMVS, setting a rigorous standard for comparison.

Chapter 3, describes the essential principles underpinning Computer Vision, covering digital imaging, camera models, and stereo vision. This chapter is crucial for understanding the technical challenges and solutions in 3D reconstruction that MVS methods need to face.

Chapter 4 introduces the custom MVS dataset developed for this study. It details the process of 3D printing, robotic setup, and image capturing, showcasing the preparation undertaken to ensure a robust evaluation platform for the MVS techniques under scrutiny.

Chapter 5 through Chapter 7 explore the chosen MVS methods: COLMAP [50], GBi-Net [41], and MVSFormer [6]. These chapters provide an in-depth analysis of the inner workings and key components of each model, prefaced by introductory sections that lay out the theoretical foundations necessary for understanding their mechanisms and functionalities.

Chapter 8 presents the results of the depth map analysis and 3D point cloud reconstruction, employing established metrics to quantify their performance. This rigorous evaluation shows the strengths and weaknesses of each method under various lighting conditions.

Chapter 9 introduces an innovative approach by augmenting the dataset with textures to simulate laser projector patterns. This chapter explores the impact of artificial texturing on the models' performance, particularly highlighting the enhancements in depth map generation and point cloud reconstruction.

The thesis ends in Chapter 10, where the conclusions are drawn from the comprehensive analysis conducted in the preceding chapters. It synthesizes the findings, reflecting on the implications for future research and the potential for practical applications of MVS techniques in industrial settings.

Part I

SCIENTIFIC BACKGROUND

2

THE MULTI-VIEW STEREO FRAMEWORK

Image-based 3D reconstruction transforms overlapping 2D image projections into a 3D scene structure. Achieving a comprehensive and precise scene reconstruction demands depth estimation for every pixel through stereo matching, where pixel correspondences across multiple images are identified. This is a complex task due to occlusions, variations in surface appearances across views, and non-Lambertian surfaces (i.e. those that do not uniformly scatter light in all directions, often resulting in shiny or reflective appearances that vary with the viewing angle). Methods to establish pixel correspondences range from local, using patches for direct disparity estimation, to global, formulating an energy function for spatial consistency.

The standard workflow for image-based 3D reconstruction involves feature extraction and matching for finding correspondences, followed by image registration and triangulation to estimate camera positions and initiate sparse reconstruction. The subsequent stage, dense 3D reconstruction, is crucial as it enhances the model's detail using the camera's calibrated settings.

MVS algorithms extend these principles to multiple images, enhancing the ability to reconstruct complex scenes by leveraging the redundancy of multiple viewpoints. Unlike traditional two-view approaches, MVS can handle significant variations in viewpoint, scale, and occlusion, improving depth accuracy and scene complexity handling. MVS algorithms use the known positions and orientations of cameras to guide the reconstruction process. By understanding where each camera is located and how it is oriented within the environment, the MVS system can triangulate the position of points in 3D space more accurately. This knowledge allows the algorithm to combine information from multiple images taken from different viewpoints, enhancing the ability to reconstruct the shape and structure of objects within the scene. In addition, multiple views help to fill in gaps and provide a more comprehensive understanding of the scene, especially in areas where individual views may lack detail or features. This results in scenes with a higher density of features that might provide more information for reconstruction.

MVS techniques vary based on the scene representation, including voxel-based, surface-evolution, feature-point-growing, and depth-map-based methods. Depth map fusion techniques, in particular, have gained popularity for large-scale, high-resolution applications due to their scalability and efficiency. Depth maps offer a 2.5D view per image, which can be merged into a unified model using 3D fusion. Point clouds provide a sparse representation, created by projecting image pixels into 3D space according to depth information. Volumetric methods utilize 3D grids to

describe geometry, either through occupancy functions or by encoding distance to the nearest surface. Triangular meshes offer another approach to model surfaces, also derivable by combining multiple depth maps into a cohesive structure.

Deep Neural Networks (DNNs) have become a cornerstone in computer vision, significantly enhancing applications like person re-identification, image and face alignment, object recognition, and stereo matching. Their ability to extract and aggregate features effectively addresses challenges in MVS tasks, such as occlusion, variable lighting, and textureless areas. Advances in large-scale 3D datasets have propelled deep learning MVS methods beyond traditional approaches. Deep learning MVS techniques generally fall into two categories based on scene representation: depth map-based and volumetric-based methods. Depth map-based approaches generate a 2.5D depth map for each view, leveraging geometric relationships between overlapping views before fusing these maps into a coherent 3D model. Conversely, volumetric methods predict 3D space occupancy directly from images, offering a global scene representation. The efficiency of depth map methods depends on the input images' quantity and resolution, whereas volumetric methods are constrained by the scene's size, making them less suitable for large outdoor areas. Consequently, many MVS networks opt to output depth maps, indirectly forming a unified 3D model.

Considering that all models utilized in this study adhere to this approach, our discussion will primarily concentrate on it.

2.1 TRADITIONAL METHODS

Traditional MVS techniques have laid the foundational frameworks upon which contemporary depth estimation technologies are built. These methods are characterized by their reliance on the principle of local smoothness and uniform depth across surfaces, with pioneering works [17, 19] highlighting the challenges of accurately capturing slanted surfaces due to the assumption of fronto-parallel planes. Such approaches often struggle with scalability and computational efficiency when applied to high-resolution datasets, prompting the development of alternative strategies like adaptive weighting, surface normals incorporation, and coarse-to-fine processing to address these limitations.

A significant advancement in the field was the introduction of the PatchMatch algorithm, which represented a departure from the computational intensity of traditional MVS methods. PatchMatch [4], by utilizing 3D-oriented planes for depth estimation, notably improved upon the efficiency and scalability of depth estimation processes. It facilitated the handling of slanted surfaces by avoiding the assumption of fronto-parallel bias, thus addressing a critical limitation of previous approaches. This method also enabled sub-pixel accuracy in depth estimations, a feature that was challenging to achieve with earlier techniques. Following its introduction,

further improvements and regularization techniques were introduced to PatchMatch, enhancing its accuracy and smoothness in depth estimations.

The development of Gipuma and COLMAP represents two of the most notable advancements in the field of MVS. Gipuma [20], introduced a highly parallelizable approach to depth estimation, significantly improving upon the computational efficiency of traditional methods. It leverages the parallelization power of modern GPUs to perform rapid depth estimations across datasets, making it a powerful tool for 3D reconstruction.

COLMAP [50], is an end-to-end image-based 3D reconstruction software that incorporates both feature extraction and matching, alongside robust MVS algorithms. It has become a benchmark in the field for its comprehensive approach to 3D modeling from images, offering tools for both sparse and dense reconstruction. COLMAP's success lies in its ability to automate the reconstruction process while providing high-quality results, integrating both geometric and photometric information to produce detailed 3D models. For these reasons, it has been selected as one of the models for analysis in this thesis.

2.2 LEARNING-BASED METHODS

Unlike traditional methods that focus on handcrafted features and photometric consistency, deep learning approaches utilize complex feature representations that merge photo consistency with scene context, offering robustness against challenges like textureless surfaces and occlusions. These methods not only improve matching accuracy by incorporating global scene understanding but also overcome ambiguities in sparse image data.

In depth estimation, supervised learning models aim to reduce the difference between estimated depths and ground truth, incorporating regularization for smoothness. Unsupervised techniques [10] rely on cues like consistency checks and semantic information instead of ground truth data for training [53].

2.2.1 *Depth Map-Based Methods*

Depth Map-based MVS Methods focus on reconstructing 3D models by estimating the depth map for each camera view. These methods utilize calibrated camera parameters to project 3D models onto color images and depth maps, employing classical stereo-matching techniques for depth estimation.

MVSNet [61] stands as the foundational end-to-end network for depth estimation, setting the standard for the field. Recent developments in MVS have primarily built upon and refined various aspects of this pipeline, aiming to enhance the accuracy and efficiency of depth estimation and 3D model reconstruction.

The typical process begins with feature extraction from input images, followed by constructing a cost volume using differential warping based

on camera parameters. The cost volume aggregates matching costs across all hypotheses of depth, with photometric consistency serving as the depth indicator. The next steps include regularizing the cost volume to mitigate noise from non-ideal conditions like non-Lambertian surfaces or occlusions and applying depth regression to pinpoint the most probable depth value for each pixel.

Feature Extraction: MVSNet implemented 2D Convolutional Neural Networks (CNNs) to extract deep image features, but subsequent developments refined the process by integrating techniques like spatial pyramid pooling [28] or Feature Pyramid Networks (FPNs) [36] covered in Section 6.1.3. More recent innovations have focused on incorporating attention mechanisms like self-attention layers to gather long-range contextual information [6].

Cost volume creation: Key to this approach is the creation of a cost volume through the plane sweep method. The first step involves discretely sampling the depth range that the scene could span. Each depth hypothesis corresponds to a potential distance from the camera at which the actual point in the scene might exist. The plane sweep method involves sweeping virtual planes across the 3D space and measuring photometric consistency to gauge depth likelihood. It employs a differential warping technique that involves using two images for which the camera intrinsics ($\mathbf{K}_r$, $\mathbf{K}_s$) and extrinsics ($\mathbf{T}_r, \mathbf{T}_s$) are known. The intrinsics define the camera's internal parameters, while the extrinsics represent its position and orientation in space ($\mathbf{T}_i = [\mathbf{R}_i, \mathbf{t}_i]$), respectively. r indicates the reference camera and s the source camera. When hypothesizing a depth d for a point in the reference camera's frame along its principal axis n, a homography matrix $\mathbf{H}_s$ is calculated. This matrix maps the coordinates from a pixel location in the reference image (u_r) to its corresponding location in the source image (u_s), approximated as $u_s \sim \mathbf{H}_s(d)u_r$. The formula for $\mathbf{H}_s(d)$ is given by:

$$\mathbf{H}_s(d) = \mathbf{K}_s\mathbf{R}_s\left(\mathbf{I} - \frac{(\mathbf{t}_r - \mathbf{t}_s)\mathbf{n}^T}{d}\right)\mathbf{R}_r^T\mathbf{K}_r^{-1} \tag{1}$$

Differential warping connects pixel pairs that map to the same points in three-dimensional space, enabling the construction of a cost volume. The resulting cost volume is a 3D (or sometimes 4D when including features across multiple views) data structure that stores the "cost" associated with each pixel at different depth levels. The "cost" is a measure of how likely it is that a given pixel corresponds to a particular depth, based on the similarity of appearance across multiple images taken from different viewpoints.

Cost volume regularization: This process aims to mitigate the effects of noise and ambiguities, often arising from occlusions, repetitive patterns, or textureless areas, that can distort depth predictions. The regularization leverages spatial context and the inherent smoothness of real-world surfaces to produce more consistent and reliable depth maps. It involves applying filters or networks that enforce spatial continuity and penalize abrupt depth changes, which are unlikely in natural scenes, except at

object boundaries. This is achieved by integrating smoothness constraints that encourage neighboring pixels to have similar depth values, thereby smoothing out the depth field while preserving edge sharpness where significant color or intensity gradients indicate true object edges. In the context of deep learning-based MVS systems, like MVSNet, regularization typically employs 3D CNNs or U-Net architectures. A key feature of these regularization networks is their ability to operate on different scales, allowing them to capture both high-level scene context and fine detail.

Depth maps: The result of the regularization step is a refined cost volume, from which depth estimates are extracted with greater accuracy. Techniques such as the soft argmin operation are then applied to translate the probabilistic depth values into final, continuous depth estimates for each pixel. The result is a depth map that combines the initial, data-driven insights from the cost volume with the contextual intelligence imposed by regularization, yielding depth predictions that are both precise and aligned with the physical properties of the observed scene.

Depth Filtering: The filtering process leverages both photometric and geometric consistency checks to identify reliable depth estimates. Photometric consistency involves comparing the appearance of corresponding points across images; points that do not exhibit similar color or intensity are likely to be inaccuracies and are thus filtered out. Geometric consistency, on the other hand, involves evaluating the geometric alignment of points in 3D space when viewed from different angles. Points that do not maintain consistent geometric relations across views are also considered unreliable and are filtered out.

Depth Map Fusion: Following the filtering stage, depth map fusion integrates the refined depth maps from multiple viewpoints into a single, unified 3D model. Fusion algorithms typically employ a volumetric representation of the scene, such as a voxel grid, where each voxel's occupancy or emptiness is determined based on the aggregated depth information. The fusion process is guided by the principle of maximizing consensus among the depth maps. A common approach involves weighting the contribution of each depth map to the final model based on the estimated reliability of the depth information it provides. This ensures that more accurate and consistent depth estimates have a greater impact on the final reconstruction. The outcome of the depth filtering and fusion process is a high-quality, dense 3D reconstruction of the scene, synthesizing information from multiple perspectives into a cohesive model.

2.3 MVS BENCHMARKS

In this section, we introduce the most used Multi-View Stereo benchmarks. They play a crucial role in advancing the field of 3D reconstruction by providing standardized datasets and evaluation metrics that facilitate the comparison of different algorithms. These benchmarks are designed to test the ability of MVS methods to accurately reconstruct 3D models from multiple images, under varying conditions of image quality, resolution,

and geometry complexity. The selection of models discussed in this thesis (Chapter 5, Chapter 6, Chapter 7) achieves state-of-the-art performance on these benchmark.

2.3.1 *DTU Robot Image Data Sets*

The DTU dataset [1] is a comprehensive collection featuring 128 scenes captured in a controlled lab setting, utilizing a structured light scanner for its reference models. Each scene is documented from either 49 or 64 fixed camera angles, under seven distinct lighting setups, producing high-resolution Red Green Blue (RGB) images (1200 × 1600 pixels). This variety in objects and materials makes it an excellent resource for training and evaluating deep learning-based MVS algorithms in conditions that mimic real-world scenarios. The dataset's reference models, primarily point clouds, necessitate the conversion of these into depth maps for analysis, typically achieved through surface reconstruction techniques like Screened Poisson Surface Reconstruction [30]. Common practice divides the dataset into training, testing, and validation segments.

In terms of evaluation, the protocol involves comparing MVS-generated reconstructions against the scanned reference models to calculate point-wise reconstruction errors, both mean and median. This method assesses the alignment of the reconstructed model with the reference, employing accuracy and completeness as the primary metrics. Accuracy reflects the average proximity of reconstructed points to the nearest point on the reference model, while completeness measures how close each point in the reference model is to a point in the reconstructed model. Often, these metrics are expressed as a percentage of points falling within a specified error margin from their counterparts. An overall quality score, averaging the mean accuracy and mean completeness, provides a comprehensive indicator of the reconstruction's fidelity.

2.3.2 *Tanks and Temples*

Tanks and Temples [31] is a benchmark designed for evaluating MVS methods under real-world conditions, featuring high-resolution indoor and outdoor scenes captured with an industrial laser scanner. The dataset includes high-resolution videos, making it suitable for assessing video-based MVS approaches. Scenes are categorized into intermediate and advanced levels based on their scale, complexity, and various challenges they present, facilitating a structured evaluation of MVS algorithms' performance. Additionally, Tanks and Temples offers an online platform for comparing results.

For evaluation, the benchmark uses the F-score at a specific threshold to gauge performance. This score combines accuracy and precision into a single metric through the harmonic mean, which is particularly effective at highlighting poor performance by being more responsive to lower values in either accuracy or completeness compared to the arithmetic mean.

2.3.3 *BlendedMVS*

BlendedMVS is a state-of-the-art multi-view stereo dataset designed specifically for training deep learning algorithms. Acknowledging the high costs associated with scanning large-scale ground-truth models, BlendedMVS offers a synthetic alternative. The dataset's images and corresponding depth maps are generated from textured 3D mesh models using an innovative and cost-effective data creation process that incorporates variable lighting conditions. It features 113 carefully chosen 3D textured models spanning a variety of scenes including architecture, street views, sculptures, and small objects, with each scene containing between 20 and $1,000$ images. In total, the dataset comprises $17,818$ images captured from unstructured camera paths, all standardized to a resolution of 1536×2048 pixels. This extensive and meticulously crafted synthetic dataset is instrumental in enhancing the generalization capabilities of machine learning models trained on it.

BlendedMVS employs a combination of 2D depth map validation and 3D point cloud evaluation to quantitatively assess reconstruction accuracy. For depth maps, it measures accuracy using the End Point Error (EPE), which is the average L1 loss across all depth values. Additionally, it calculates the percentage of depth estimations that deviate from the true depth by more than 1 and 3 units, providing insight into the depth prediction's precision. In the 3D domain, point cloud evaluations focus on accuracy, completeness, and the F-score. These metrics together offer a comprehensive view of the model's performance in both 2D and 3D aspects of the reconstruction process.

2.3.4 *Others*

Other mentionable MVS benchmarks:

- **Middlebury Stereo Evaluation**: One of the first and most referenced benchmarks, Middlebury provides high-quality images in controlled light. It's great for testing how accurately algorithms can estimate depth in simpler scenes and offers dense, reliable ground truth data [48].
- **ETH3D Benchmark**: ETH3D includes indoor and outdoor scenes at both low and high resolutions. It stands out because it tests algorithms in challenging light conditions and areas with little texture, checking how well they can adapt [52, 51].
- **KITTI Vision Benchmark Suite**: Mainly for self-driving car research, KITTI also offers stereo images useful for MVS testing. It features real traffic scenes from moving vehicles, challenging MVS methods to deal with movement and blur [21, 40].

3

FUNDAMENTALS OF COMPUTER VISION

Building on the historical context of MVS provided in the previous section, this chapter delves into the fundamentals of Computer Vision, a field that forms the cornerstone of how machines interpret and understand the visual world. Computer Vision, at its core, is about enabling computers to replicate the human visual system's capabilities, thereby allowing them to identify patterns, objects, and scenes in images and videos.

The chapter first covers the basics of digital imaging, exploring the process through which images are represented and processed digitally. Then, it discusses local and global descriptors, pivotal in feature detection and description. Local descriptors, focusing on specific points or small regions within an image, enable tasks such as object recognition and matching across different views. Conversely, global descriptors provide a comprehensive representation of an entire image, useful for scene classification and retrieval.

The following section delves into the mathematical models that describe how a camera captures a 3D scene onto a 2D image, with a focus on the pinhole camera model and camera calibration. Their mathematical formulation is crucial for reconstructing 3D structures from 2D images, a fundamental aspect of computer vision.

Stereo Vision is explored as a method to infer depth information from two or more cameras' views, mimicking human binocular vision. This section covers the principles that allow for the extraction of 3D spatial information from stereo image pairs, setting the stage for more complex MVS systems. Triangulation is a core step in it, detailing the mathematical principles that enable the precise localization of 3D points by intersecting lines of sight from multiple images. These concepts are essential for accurate 3D mapping and navigation in many applications such as robotics and augmented reality.

3.1 DIGITAL IMAGING

Digital images are the representation of visual information in a format that can be stored and processed by a computer. At its core, an image is composed of pixels, the smallest unit of visual information, arranged in a grid. Each pixel contains data for color and intensity, typically represented in various color spaces such as RGB for color images or grayscale for black and white images. The resolution of an image, defined by its width and height in pixels, determines its detail level. High-resolution images contain more pixels, offering finer detail at the cost of increased storage and processing requirements.

Most challenges in computer vision can essentially be reduced to determining correspondences between raw input images (e.g., matrices of color values) and abstract representations (e.g., high-level semantic categories or low-dimensional numerical vectors). By linking raw images to these abstract concepts, we can make predictions about individual images (e.g., identifying the content of the scene), which facilitates establishing connections between multiple images (e.g., identifying images captured at the same location). In image-based 3D modeling, the goal is to deduce the geometric relationships among several images to reconstruct the three-dimensional structure of the scene being captured. A fundamental challenge is thus the robust and efficient geometric linking of multiple related images. Typically, this linking process is approached in two stages: first, independently describing the contents of all input images. Then, using these independent descriptions to identify similar images that portray the same scene structures.

Various methods exist in literature for describing the content of an image. These methods can be broadly categorized based on whether they describe content globally (e.g., "this image shows a dog") or locally (e.g., "this region/pixel in the image shows its nose"). Additionally, these methods vary in the specificity of their descriptions, i.e., whether the descriptions are category-level (e.g., statue, city) or instance-level (e.g. Statue of Liberty, New York City). In image-based 3D modeling, description methods from both ends of this spectrum are leveraged. Broadly speaking, local description methods allow for the precise reconstruction of geometric relationships between pairs of images at the instance level, whereas global description methods, though less specific and accurate, facilitate the efficient association and categorization of large image collections.

3.1.1 *Local description*

Local image description involves breaking down an image into locally distinctive features to characterize its content. These local features, such as points, edges, or blobs, are defined by the image content within their immediate vicinity. Ideally, these features should be distinctive and consistently recognizable across different images of the same object, ensuring their description remains consistent despite variations in lighting and perspective [39].

The focus often lies on sparsely detected point features, which facilitate a robust and efficient approach to geometric reconstruction challenges. These point features are identified at unique locations within an image, such as corners or intersections. To correlate points belonging to the same object across multiple images, one must extract and compare their geometric (position, orientation, size, etc.) and appearance attributes (for example, a descriptor vector detailing the colors within a small surrounding area of the point). This representation, however, is not inherently stable across different lighting or viewing conditions. To counteract this, the appearance descriptions of features are usually standardized based on their geometry,

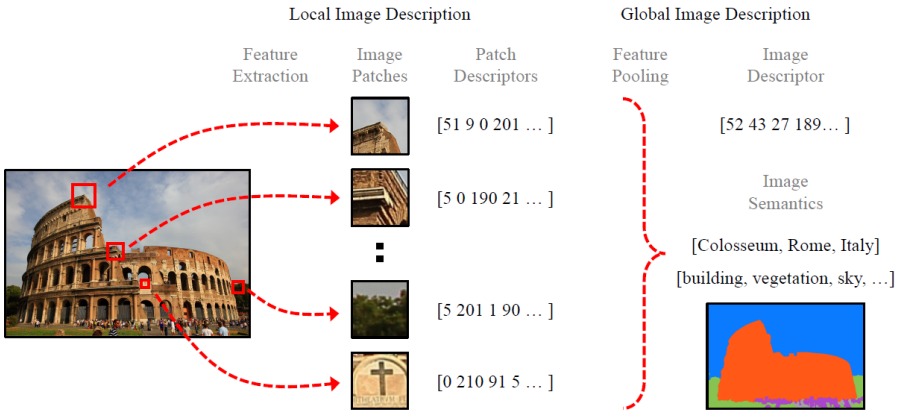


Figure 3.1: An example of the concept of local and global descriptors. Credits: [49].

with a preference for encoding color gradient information rather than absolute color values. Local image descriptions typically compile the content of an image into hundreds or thousands of local feature points, including their basic geometric and appearance attributes. Matching these local features across two or more images depicting the same scene allows for establishing pixel-level correspondences between image regions. As further discussed in the chapter, this pixel-level correspondence is crucial for determining the relative geometric relationships between images.

The Scale Invariant Feature Transform (SIFT) has been a leading method for hand-crafted local features for over two decades, celebrated for its robustness. Along with SIFT [37], its derivatives [5, 15], set a benchmark for robustness. The advent of deep learning models has introduced a new era for local feature detection, where features are learned directly from data rather than being manually designed [29, 3, 33]. These deep learning-based features have shown remarkable success in capturing intricate patterns and nuances that hand-crafted features might miss, leading to more accurate and versatile applications in image-based 3D modeling. This shift towards automatically learned features represents a significant evolution in the field, enabling more sophisticated and adaptive approaches to understanding and reconstructing complex scenes.

3.1.2 *Global description*

Global image description condenses an image's content into a singular representation, such as a high-level semantic label or a numerical vector. This method offers a more streamlined representation compared to local feature descriptions, facilitating the efficient organization and linking of vast image datasets. One common method for crafting a global image descriptor involves aggregating local feature information into a standardized histogram, drawing inspiration from the bag-of-words approach

widely utilized in natural language processing and information retrieval [25]. In this analogy, individual local image features act as "words," while the global descriptor vector forms a histogram reflecting the distribution of these "words" across a predefined visual vocabulary [42]. This vocabulary is typically developed through hierarchical clustering applied to a broad set of local image features. To identify images sharing similar content, the technique examines the similarity in their visual word distributions, often employing mechanisms like term frequency-inverse document frequency (tf-idf) scoring. This process can be scaled to accommodate large collections of images by using an inverted index that maps visual words to their occurrences in images. A key benefit of this approach is the seamless integration of local and global image descriptions, as the same local features are utilized in both constructs.

In recent years, novel methods that leverage data-driven learning have emerged for global image description. These methods avoid aggregating discrete local features in favor of employing end-to-end trainable CNNs that are adept at directly mapping images to a fixed-size vector within an embedding space or generating a semantic summary of the image [32, 46]. By conducting nearest neighbor searches within this embedding space or applying semantic analysis, these advanced models achieve efficient image association, showcasing the potential of directly learned representations in enhancing image categorization and retrieval processes.

3.2 CAMERA MODEL

The pinhole camera model simplifies complex optical systems to a single point through which light travels; this model captures the essence of perspective and depth perception in a two-dimensional image. The mathematical formulation of this process, involving focal length and the positioning of objects relative to the camera, illustrates the geometric principles of how images are projected onto a plane. The correct calibration of the camera is crucial because it determines the camera's internal and external parameters, allowing for the accurate alignment of 3D world coordinates with 2D image coordinates, thereby correcting lens distortions and ensuring precise depth estimation and 3D reconstructions from 2D images.

3.2.1 *Pinhole Camera*

The pinhole camera model is a simplified representation that describes how a camera captures an image. It assumes that there is a single point (the pinhole) through which light from the scene passes before projecting onto the camera sensor or film at the back of the camera. In the context of an ideal pinhole camera, these light rays converge at a single projection center $\mathbf{C} \in \mathbb{R}^3$ (the aperture). The projection process of a 3D scene point

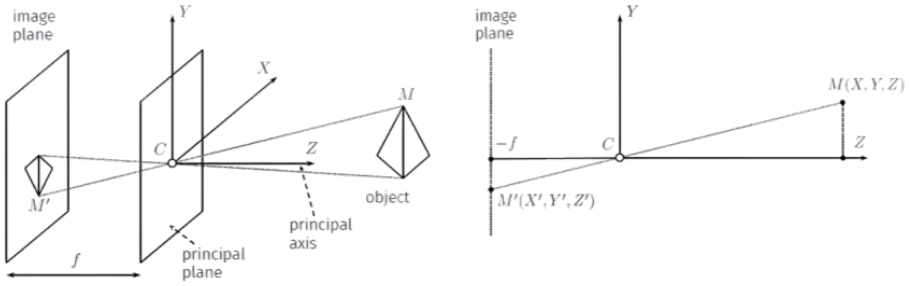


Figure 3.2: Perspective projection of an object through a pinhole camera.

$\mathbf{M} \in \mathbb{R}^3$ into the 2D imaging plane can be visualized in Figure 3.2 and can mathematically be expressed as follows:

$$[X', Y', Z']^T = \left[-\frac{f}{Z} \cdot X, -\frac{f}{Z} \cdot Y, -f\right]^T \tag{2}$$

Here, $-f/Z$ acts as the perspective scale factor, emphasizing the impact of depth (Z) on the scaling of image points. As the distance from the pinhole (Z) increases, the scale factor decreases, resulting in smaller projections of objects that are further away. This principle underlies the effect of perspective in the pinhole camera model, where objects closer to the camera appear larger than those farther away, despite their actual sizes.

Building on this foundational concept, we can further express the projection process as follows:

$$\mathbf{m} \simeq \lambda[u\ v\ 1]^T = \mathbf{PM} = \mathbf{K}[\mathbf{R}\ \mathbf{t}]\mathbf{M} = \mathbf{K}\left[\mathbf{R} \mid -\mathbf{R}^T\mathbf{C}\right]\mathbf{M} \tag{3}$$

In this formula, $\mathbf{P}$ denotes a 3×4 matrix of rank 3, which facilitates the mapping of the scene point $\mathbf{M}$ in homogeneous coordinates from the 3D projective space $\mathbb{P}^3$ to an observed point $\mathbf{m}$ in the 2D projective imaging plane $\mathbb{P}^2$. The 3×3 rotation matrix $\mathbf{R} \in SO(3)$ alongside the translation vector $\mathbf{t} \in \mathbb{R}^3$ define the Euclidean transformation from the world to the camera coordinate system, known as the extrinsic camera calibration parameters. Conversely, the intrinsic camera calibration parameters are captured within the upper triangular matrix $\mathbf{K}$, specified as:

$$\mathbf{K} = \begin{bmatrix} f & s & c_u \\ 0 & \alpha f & c_v \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

Within the intrinsic camera matrix $\mathbf{K}$, several parameters are critical for accurately modeling the camera's optical characteristics. The focal length (f) is pivotal, controlling the camera's field of view and influencing the perceived depth in the image. The aspect ratio factor (α) adjusts the image projection to match the sensor's aspect ratio, ensuring the geometric integrity of the captured scene. Although typically negligible in modern

cameras, the skew coefficient (s) is included to correct for any potential skewness in the pixel grid, ensuring the axes remain perpendicular. Finally, the optical center (c_u, c_v) identifies the central point in pixel coordinates where the camera's optical axis intersects the image plane. If not known, the correct estimation of those parameters is essential for a precise projection from 3D space to the 2D imaging plane, facilitating accurate depth estimation and 3D reconstruction from 2D images.

3.2.2 *Camera Calibration*

The camera's internal parameters (such as focal length, optical center, and lens distortion) and external parameters (position and orientation in space) are determined through the process known as camera calibration. This is crucial for accurate 3D reconstruction because it corrects for lens distortion and aligns the 3D world coordinates to the 2D image coordinates. By relating image points $\mathbf{m}$ to their corresponding world points $\mathbf{M}$, one can estimate the 12 parameters of the projection matrix $\mathbf{P}$ through a linear relationship obtained from Equation 3:

$$\mathbf{m} \simeq \mathbf{P}\mathbf{M} = \begin{bmatrix} \mathbf{P}_1^T \\ \mathbf{P}_2^T \\ \mathbf{P}_3^T \end{bmatrix} \mathbf{M} \tag{5}$$

Based on this equation, rearranging the terms with the Direct Linear Transform method allows for the elimination of the unknown scaling factor λ, leading to a homogeneous equation system:

$$\begin{cases} \mathbf{P}_3^T \mathbf{M} u = \mathbf{P}_1^T \mathbf{M} \\ \mathbf{P}_3^T \mathbf{M} v = \mathbf{P}_2^T \mathbf{M} \end{cases} \Rightarrow \begin{bmatrix} \mathbf{M}^T & 0 & -\mathbf{M}^T u \\ 0 & \mathbf{M}^T & -\mathbf{M}^T v \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix} = 0 \tag{6}$$

This system requires at least 6 pairs of 2D-3D correspondences and provides two constraints per pair for the 12 unknowns. The solution, found within the null space of $\mathbf{A}$, corresponds to the eigenvector associated with the smallest singular value in the Singular Value Decomposition (SVD) of $\mathbf{A}$. Decomposing the projection matrix $\mathbf{P}$ into intrinsic and extrinsic parameters is achieved through QR-decomposition, adjusting the normalization for correct scaling, given that $\mathbf{K}$ is upper triangular and $\mathbf{R}$ is orthonormal [18].

The previously mentioned projection model assumes an ideal pinhole camera, which doesn't account for real-world camera lens distortions. These distortions are often modeled with radial coefficients $k_1, k_2, \ldots$ in higher-order polynomial expressions:

$$\tilde{\mathbf{m}} = f_{\text{dist}}(\mathbf{m}) = \begin{bmatrix} u \\ v \end{bmatrix} (1 + k_1 r^2 + k_2 r^4 + \ldots) \tag{7}$$

where $r^2 = u^2 + v^2$, and $\tilde{\mathbf{m}}$ and $\mathbf{m}$ represent distorted and undistorted points, respectively. If lens distortion parameters are known beforehand, calibration uses undistorted points. Otherwise, calibration must also estimate these parameters, complicating the process [26, 44].

For effective image-based 3D modeling, selecting an appropriate camera model is crucial for achieving high-quality reconstructions. Depending on the scenario, intrinsic parameters might be determined in advance through detailed calibration routines, or each image may require independent calibration during reconstruction, limiting the feasibility of using complex distortion models due to the lack of sufficient data from a single image.

3.3 STEREO VISION

Stereo vision is a technique used to perceive depth by emulating the human visual system, which relies on two eyes to gauge the distance and depth of objects. In computer vision, stereo vision systems use two or more cameras to capture the same scene from slightly different angles. By analyzing the differences between these captured images, algorithms can estimate the depth of each point in the scene. This process is known as disparity calculation, where greater differences between corresponding points in the stereo images indicate closer objects and smaller differences suggest objects are further away. The first step involves identifying correspondences between images, a foundational concept introduced earlier. The following step is triangulation, which leverages these correspondences to accurately determine the 3D coordinates of scene points, effectively bridging the gap between two-dimensional image data and the three-dimensional structure of the scene.

3.3.1 *Triangulation*

In image-based 3D modeling, our aim is to reconstruct the 3D structure of a scene ($\mathbf{M}$) from its 2D image representations ($\mathbf{m}$). However, directly reversing the projection from 3D to 2D is not feasible. This challenge arises because the projection process inherently loses depth information; in a 2D image, all points along a viewing ray from the camera to the scene are collapsed into a single image point. Specifically, in the projective geometry framework, any two points $\mathbf{M}_1$ and $\mathbf{M}_2$ in 3D space that lie on the same viewing ray from the camera produce the same image point after projection, despite being at different distances from the camera. This creates an ambiguity in the scale, or depth, of the scene (λ), since multiple 3D points have equivalent projections onto the 2D plane, making it challenging to determine the exact position of $\mathbf{M}$ in three-dimensional space from its image $\mathbf{m}$ alone.

To deduce the depth λ, or the distance from the camera location $\mathbf{C}$ to the scene point $\mathbf{M}$ along the viewing ray y for a given pixel $\mathbf{m}$, one must utilize the camera's intrinsic and extrinsic parameters. With λ known, the

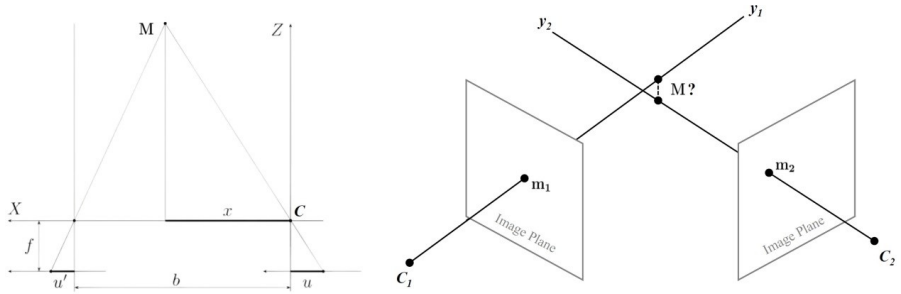


Figure 3.3: Illustration of Stereo Vision: process of computing disparity from two images captured from different viewpoints.

3D position of the point observed in the image, denoted as $\bar{\mathbf{M}}$, can be calculated as:

$$\bar{\mathbf{M}} = \lambda \mathbf{R}^{\top} \mathbf{K}^{-1} \mathbf{m} + \mathbf{C} \tag{8}$$

Lacking direct depth information (λ), the position can be inferred by intersecting viewing rays from the same scene point captured in images from different cameras. This method, known as triangulation, uses the rearranged projection equation in a direct linear transform format to estimate the intersection point:

$$\begin{cases} \mathbf{P}_3^{\top} \mathbf{M} u = \mathbf{P}_1^{\top} \mathbf{M} \\ \mathbf{P}_3^{\top} \mathbf{M} v = \mathbf{P}_2^{\top} \mathbf{M} \end{cases} \Rightarrow \begin{bmatrix} \mathbf{P}_3^{\top} \mathbf{u} - \mathbf{P}_1^{\top} \\ \mathbf{P}_3^{\top} \mathbf{v} - \mathbf{P}_2^{\top} \end{bmatrix} \mathbf{M} = 0 \tag{9}$$

This requires at least two observations from different images to form an over-constrained system with four linear equations for three unknown coordinates of the point. The extra equation accounts for potential inaccuracies in image measurements or camera calibration, as perfect intersections of viewing rays in 3D space are unlikely due to these errors. It's also worth noting that if the camera centers coincide, the equation system becomes singular, rendering all points along the parallel viewing rays as plausible solutions due to the geometric nature of the problem.

3.3.2 *Two-View Geometry*

In most real scenarios, camera calibration and 3D scene structures are not known a priori and must be recovered simultaneously. Two-View Geometry explores the challenge of deducing the 3D structure of a scene from 2D image data without prior knowledge of camera calibrations. This section introduces homography and epipolar geometry as frameworks for understanding the geometric relationship between two camera views under various motion conditions, bypassing the direct recovery of the scene's Euclidean structure. Homography applies to scenarios of either pure camera rotation or any motion capturing a flat scene, while epipolar

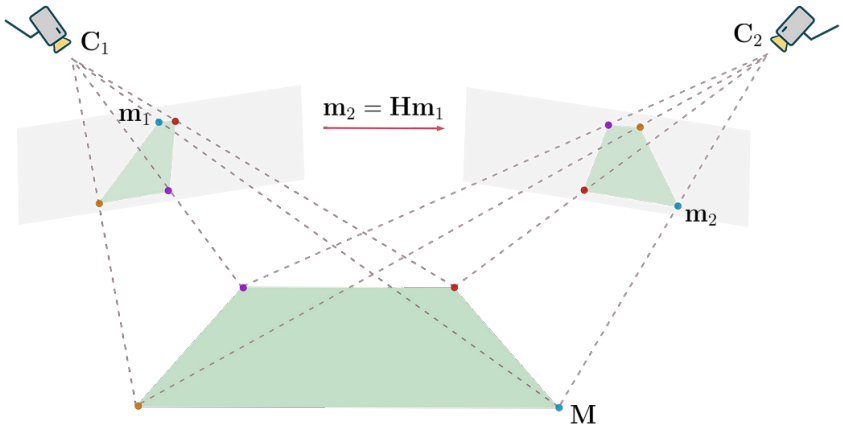


Figure 3.4: A 3D point **M** being projected onto the image planes of two cameras, $\mathbf{C}_1$ and $\mathbf{C}_2$, resulting in points m_1 and m_2, respectively. The homography matrix **H** facilitates the transformation of $\mathbf{m}_1$ in the image plane of $\mathbf{C}_1$ to $\mathbf{m}_2$ in the image plane of $\mathbf{C}_2$. Credits: https://tinyurl.com/4hr5d6km

geometry covers the two-view geometry for a moving camera observing any scene, regardless of the camera's intrinsic details being known.

3.3.2.1 *Homography*

A homography is a 2D projective transformation that maps points $\mathbf{m}_1$ on one plane to $\mathbf{m}_2$ on another through the homography matrix **H**, as shown in Figure 3.4.

$$\mathbf{m}_2 \simeq \lambda_2 \begin{bmatrix} u_2 & v_2 & 1 \end{bmatrix}^T = \mathbf{H}\mathbf{m}_1 = \begin{bmatrix} \mathbf{H}_1^T \\ \mathbf{H}_2^T \\ \mathbf{H}_3^T \end{bmatrix} \mathbf{m}_1 = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \mathbf{m}_1 \quad (10)$$

H possesses eight degrees of freedom due to the projective ambiguity λ. Therefore, at least four correspondences across two views are necessary to estimate it by solving the following homogeneous system of equations:

$$\begin{cases} \mathbf{H}_3^T \mathbf{m}_1 u_2 = \mathbf{H}_1^T \mathbf{m}_1 \\ \mathbf{H}_3^T \mathbf{m}_1 v_2 = \mathbf{H}_2^T \mathbf{m}_1 \end{cases} \Rightarrow \begin{bmatrix} \mathbf{m}_1^T & 0 & -\mathbf{m}_1^T u_2 \\ 0 & \mathbf{m}_1^T & -\mathbf{m}_1^T v_2 \end{bmatrix} \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \mathbf{H}_3 \end{bmatrix} = 0 \quad (11)$$

Assuming the extrinsic parameters for the first camera are set to $R_1 = I$ and $T_1 = 0$ for simplicity, the homography matrix **H** simplifies to:

$$\mathbf{H} = \mathbf{K}_2 \left(\mathbf{R}_2 - \frac{\mathbf{T}_2 \mathbf{n}^T}{d} \right) \mathbf{K}_1^{-1} \quad (12)$$

Here, **n** represents the unit normal vector to the scene plane, and d is the distance from the scene plane to the first camera's projection center. The

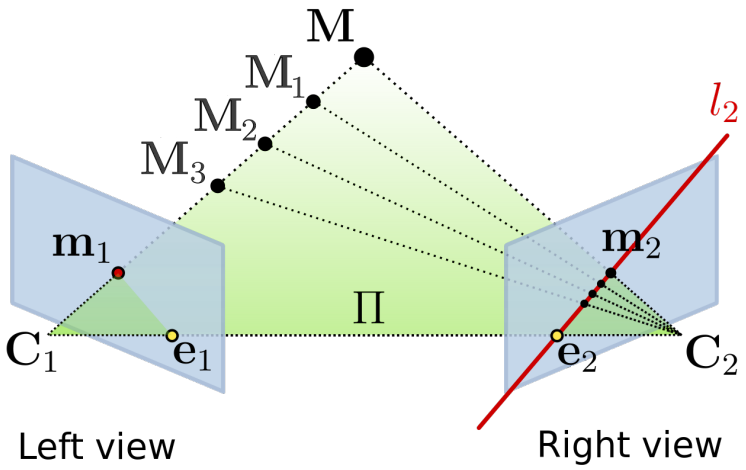


Figure 3.5: Visualization of fundamental concepts in Epipolar Geometry: the epipolar line l_2, epipoles e_1, e_2, and the relationship between corresponding points m_1, m_2 across two different camera views. Credits: https://tinyurl.com/v3bw4tmv

decomposition highlights a common challenge in 3D reconstruction: the inability of homography to determine the scene's scale and the specifics of camera motion due to scale ambiguity, a fundamental issue when lacking prior scene knowledge.

3.3.2.2 *Epipolar Geometry*

Another fundamental concept in stereo vision is epipolar geometry. It describes the geometric relationship between two views of a scene captured from different perspectives, i.e. by cameras with distinct projection centers, $\mathbf{C}_1$ and $\mathbf{C}_2$. Epipolar geometry is pivotal when the camera motion is not limited to rotation, or the scene is not strictly planar, accommodating more general movements and structures. The foundation of epipolar geometry is the epipole, defined as the image projection of one camera's center in the other camera's image plane, resulting in $\mathbf{e}_1 = \mathbf{P}_1\mathbf{C}_2$ and $\mathbf{e}_2 = \mathbf{P}_2\mathbf{C}_1$.

Given an observation $\mathbf{m}_1$ of a 3D point $\mathbf{M}$ in the first image, we can delineate the epipolar line l_1 as passing through both the epipole $\mathbf{e}_1$ and the observation $\mathbf{m}_1$. This line, when extended to intersect with the scene, defines an epipolar plane. The concept of epipolar lines and planes is reciprocal between the two images. The intersection of the epipolar plane with the second camera's image plane produces a corresponding epipolar line $l_2 = \mathbf{P}_2\Pi$, upon which any 3D point $\mathbf{M}$ on the epipolar plane Π must project in both images. The epipolar constraint encapsulates this geometric relationship, stipulating that the second image's observation $\mathbf{m}_2$ of the point $\mathbf{M}$ must reside on the epipolar line l_2. This tells us that any point $\mathbf{M}$ within the scene, when projected into the two images, must align with the designated epipolar lines derived from the cameras' relative

positions and orientations. This epipolar geometry framework thus serves as a cornerstone for establishing correspondences across images in two-view geometry, aiding in the accurate reconstruction of the 3D scene from two-dimensional observations.

Part II

DATASET

4

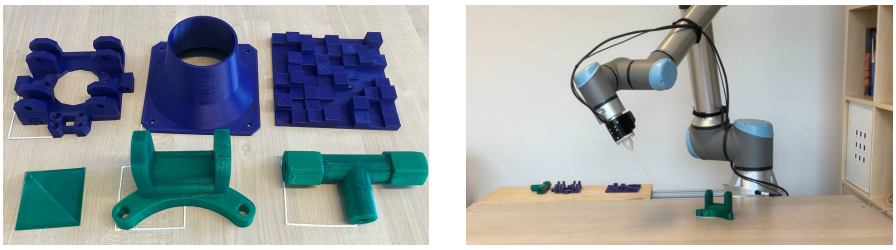
CUSTOM MULTI-VIEW STEREO DATASET

Over the past decades, progress MVS technology has significantly benefited from the introduction of benchmark MVS datasets, listed in Section 2.3. These datasets have played a pivotal role in providing researchers with standardized frameworks to evaluate and refine their algorithms, leading to substantial advancements in the field. Among the notable datasets are the Middlebury MVS set, DTU dataset, Tanks and Temples, ETH3D, and BlendedMVS, each contributing to the progression of MVS technology in unique ways. These benchmarks encompass a wide array of scenes, from indoor objects to complex outdoor environments, under various conditions, offering diverse challenges for MVS algorithms to tackle. However, it's noteworthy that the creation of these datasets often involves complex and particularly expensive techniques for obtaining ground truth data, such as laser scanning, which can be a limiting factor for their expansiveness and diversity.

In contrast to these traditional methodologies, we introduce a novel approach to generate ground truth data that leverages the precision of 3D printing technology. By using 3D printed objects as subjects in our image datasets, for which we have the exact 3D mesh models, we can derive highly accurate rendered depth maps and 3D point clouds. This method not only bypasses the need for expensive and sophisticated scanning equipment but also allows for the creation of highly controlled and varied datasets that can mimic real-world complexity in a scalable and cost-effective manner. This approach is advantageous in scenarios requiring the estimation of depth or point clouds of small-sized objects, such as in industrial environments, where precision and accuracy are essential. Additionally, this approach enables the training of models on datasets that are specifically constructed and tailored for certain tasks or objects, enhancing the applicability and effectiveness of MVS technologies in targeted scenarios.

4.1 3D PRINTING PROCESS

To create our dataset we employed the XYZ da Vinci Jr. 1.0 3-in-1 3D printer. The printer's build volume of 150x150x150 mm suits our requirement to focus on smaller objects. A total of six unique objects were selected for printing, based on criteria that ensure a diverse representation of shapes, sizes, and complexities from simple to more complex ones. The objects were printed using Polylactic Acid (PLA) filament, chosen for its environmental friendliness and printing properties. Each print was closely monitored to ensure quality and consistency. After printing, each object underwent minimal post-processing to remove any support material and



(a) The 3D printed objects of study. (b) The image capturing setup.

Figure 4.1: Visualization of the laboratory setup for the image capturing process.

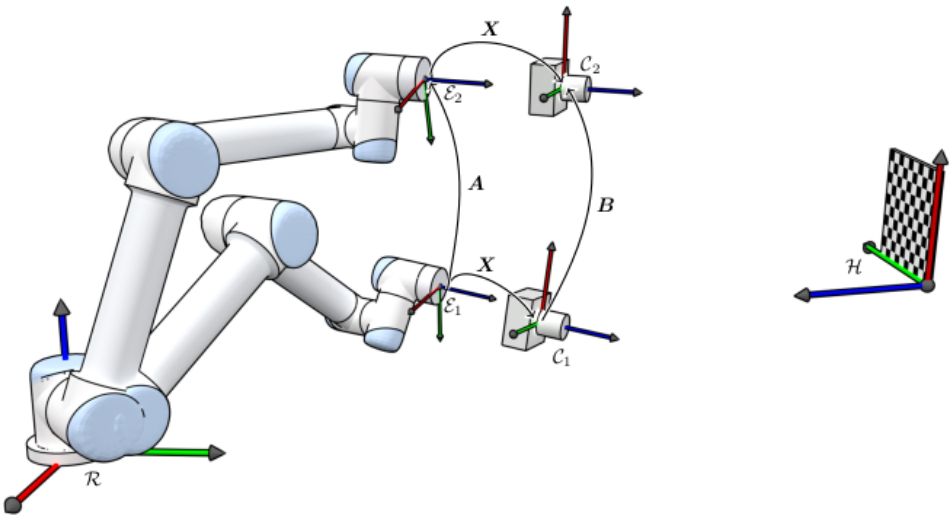


Figure 4.2: Visualization of the problem of determining transformation X in a Eye-in-Hand Configuration. A camera is mounted on the robot hand and the goal is to find the transformation X from the robot hand frame $\mathcal{E}$ to the camera frame $\mathcal{C}$. Credits: https://tinyurl.com/2p8htzw

to smooth the surface, ensuring that the physical models closely match their digital counterparts.

4.2 ROBOTIC SETUP AND IMAGE CAPTURING

The robotic setup for capturing the images incorporates the UR10e cobot (collaborative robot) from Universal Robots. Given its precision, it is suited for automated image acquisition tasks. The UR10e has a low repeatability error of ± 0.05 mm, ensuring that every image captured is from the exact position and orientation required, a critical factor for high-quality MVS datasets. On the robot's end-effector (hand) a camera (eye) is mounted for image capturing.

4.2.1 *Hand-Eye Calibration*

Before the actual image-capturing phase, hand-eye calibration is a critical process that addresses the challenge of determining the spatial relationship

between the robot's hand and the mounted camera. This relationship is important for precise interaction with the environment, such as object manipulation, navigation, and detailed 3D reconstruction, where knowing the camera's position and orientation relative to a fixed point in space is essential.

The core of hand-eye calibration is encapsulated in the equation $AX = XB$, which describes the homogeneous transformations that link the robot's hand (the manipulator), the eye (camera), and the world coordinate system. In this equation, A and B denote known transformation matrices. Specifically, A captures the movement of the robot's hand between two distinct positions within the robot's base coordinate system. On the other hand, B is concerned with capturing the shift in the camera's position, but from the viewpoint of the camera's own coordinate system, effectively describing how the camera's orientation and position change relative to the robot itself. The matrix X stands as the sought-after transformation matrix that accurately defines the camera's position and orientation in relation to the robot's hand. Successfully determining X enables precise prediction of the camera's viewpoint within the robot's base coordinate framework, as illustrated in Figure 4.2.

The equation $AX = XB$ defines a set of linear transformations, although finding a solution within the special orthogonal group $SO(3)$ introduces challenges, making the equation appear non-linear in practice. For accurate estimation, multiple movements and observations are necessary to collect sufficient data. Various algorithms exist for solving this problem, with the most common approaches involving either closed-form solutions, which offer a straightforward but sometimes less accurate answer, or iterative optimization techniques, which can provide higher accuracy at the cost of increased computational complexity. The standard procedure is to take pictures of a checkerboard using the camera. The pose of the checkerboard pattern in the camera frame C can be found using OpenCV, a widely-used computer vision library. The transformation X can be found using the method described in [43].

In the context of our custom dataset creation, hand-eye calibration is indispensable for ensuring that each image captured by the camera can be precisely associated with a specific robot pose. This precision enables the generation of a dataset where the ground truth position and orientation of the camera are known with high accuracy. This allows us to automate the image capture process, systematically covering a comprehensive range of viewpoints around each 3D-printed object.

4.3 IMAGE RENDERING AND GROUND TRUTH GENERATION

The output of the image capturing process is a collection of images alongside precisely known camera poses relative to the system's origin, which is placed at the corner of each object, as shown in Figure 4.3. The subsequent phase involves generating ground truth depth maps and point clouds,

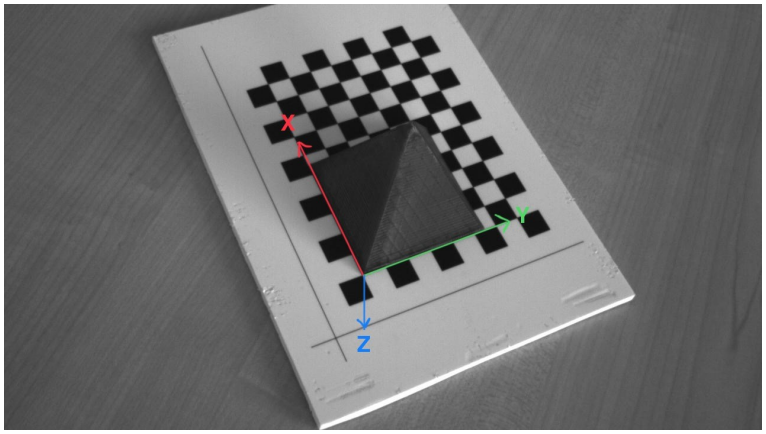
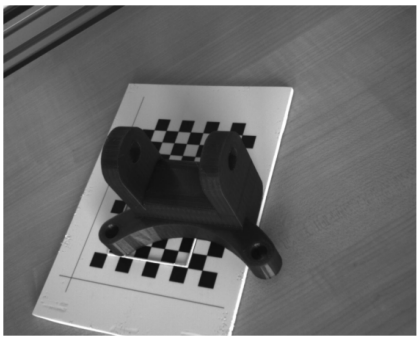
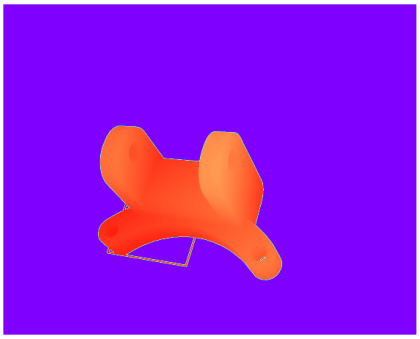


Figure 4.3: Visualization of the coordinate system adopted.

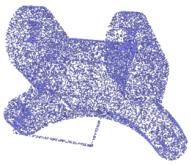


(a) The undistorted version of the original image.

(b) A photorealistic rendered image of the object.



(c) The depth map derived from the rendered image.



(d) The point cloud representation of the object.

Figure 4.4: Comprehensive visualization of the rendering process.

necessitating a rendering tool capable of providing both. For this purpose, we selected PyTorch3D.

PyTorch3D [47] is a Python library developed by the Facebook AI research team, contributing to the suite of tools available for 3D computer vision tasks. It is positioned alongside other significant projects such as TensorFlow Graphics by Google and Kaolin by NVIDIA, as one of the initial libraries to implement differentiable rendering. The library is designed to be compatible with PyTorch, as well as with C++ and CUDA, which allows for its application in real-time scenarios.

Incorporating the limitations posed by PyTorch3D's handling of camera models, it's critical to address the issue of lens distortion which PyTorch3D does not inherently account for. PyTorch3D assumes that images are captured with a pinhole camera model which implies no lens distortion. However, real-world cameras, particularly those used in the image capturing process for 3D reconstruction tasks, often introduce radial and tangential distortion due to the curvature and alignment of the lens elements. To mitigate this discrepancy and ensure the accuracy of the rendered images and depth maps, images are pre-processed using OpenCV, which offers comprehensive support for image distortion correction. Before feeding the images into the rendering pipeline of PyTorch3D, we employ OpenCV's undistortion functions, which adjust the images based on the known distortion coefficients obtained during the camera calibration phase.

To integrate the captured images and their associated camera poses into the PyTorch3D rendering pipeline, a crucial step involves the conversion of coordinate systems. Camera poses obtained from the robotic system are defined in its own coordinate space, which differs from the coordinate system used by PyTorch3D. The transformation process involves applying a series of rotations and translations to the camera poses to align them with the PyTorch3D coordinate system. This alignment ensures that the spatial relationships between the camera and the objects in the scene are accurately maintained in the rendered outputs, Figure 4.4b.

As an integral feature of the rendering process, the renderer class stores the distance of each pixel from the camera within the rendered images. By storing the depth information for each pixel, the renderer provides a detailed map that represents the distance of scene elements from the perspective of the camera's principal plane. These depth maps serve as our ground truth, offering a benchmark against which we can compare the accuracy and effectiveness of various models in replicating the true spatial configuration of the scene. An example is shown in Figure 4.4c.

Additionally, PyTorch3D extends its utility by enabling the generation of point clouds from the 3D meshes, as visualized in Figure 4.4d. This capability further enriches our dataset by providing a versatile form of ground truth data. Point clouds, representing sets of vertices in a three-dimensional coordinate system, offer a different perspective compared to depth maps, capturing the geometry and structure of objects in space. By leveraging this feature, we gain access to a comprehensive set of metrics for evaluating model performance, allowing for a complete assessment of

how well different models can reconstruct the objects within the captured scenes.

Part III

METHODS

5

COLMAP

Among the choice of available models, COLMAP [50] distinguishes itself as a model of particular interest, primarily due to its enduring status as a state-of-the-art framework in the field of MVS. This chapter is dedicated to exploring COLMAP, motivated by its comprehensive approach to image-based 3D modeling and its proven efficacy in handling diverse and complex datasets.

COLMAP encapsulates a robust set of algorithms that address the entire pipeline of 3D reconstruction, from initial image matching to dense point cloud generation. Its versatility is further highlighted by its ability to seamlessly transition between structured and unstructured environments, showcasing its adaptability to a wide range of scenarios. The decision to incorporate COLMAP into our comparative study stems from its innovative use of geometric constraints and optimization techniques, which significantly enhance the accuracy and completeness of the reconstructed scenes.

We proceed now to present the full pipeline for 3D reconstruction from a collection of unstructured images $I = \{I_i | i = 1, \dots, N_I\}$, where each image I is captured with a perspective camera and can be visualized as a color matrix. COLMAP follows a sequential process incorporating an incremental sparse reconstruction phase. This strategy is adopted due to its enhanced robustness in unstructured environments [2], despite the existence of hierarchical [22] and global methodologies [13] as alternatives.

The first phase is the search for correspondences, initiating with feature extraction and matching, and proceeding to geometric verification. This process constructs a scene graph that lays the groundwork for subsequent reconstruction stages. Initially, the system seeds the model through a carefully chosen two-view reconstruction, then progressively integrates new images. This stage involves triangulating scene points, filtering out outliers, and refining the overall reconstruction via bundle adjustment.

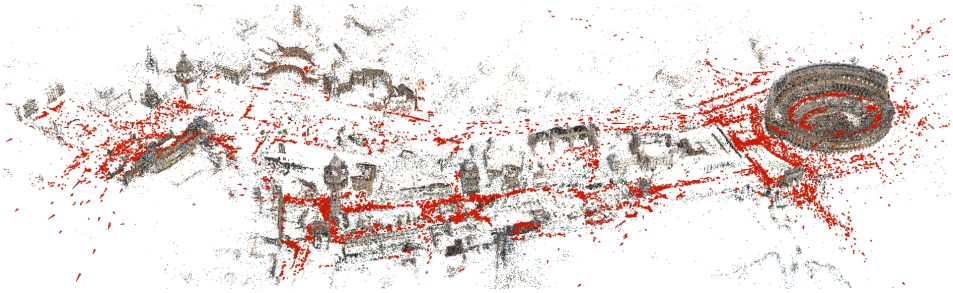


Figure 5.1: Sparse models of central Rome using 21K photos produced by COLMAP's SfM pipeline. Credits: https://colmap.github.io/

Figure 5.2: Dense models of several landmarks produced by COLMAP's MVS pipeline. Credits: https://colmap.github.io/

The culmination of this pipeline is the dense reconstruction stage, which leverages the sparse scene model and camera calibrations to craft a more detailed scene representation, potentially materializing as a dense point cloud or a textured surface mesh.

5.1 CORRESPONDENCE SEARCH

Beginning with an unorganized set of images I, the initial step in a 3D reconstruction pipeline involves establishing the two-view geometry for overlapping image pairs. This process results in a scene graph where images and scene points form nodes, interconnected through bidirectional edges indicating scene overlap, each annotated with a model of the two-view geometry, and undirected edges marking visibility links between images and scene points.

- **Feature Extraction**: For each image I_i within the set I, local image features $F_i = \{g_j, d_j\}$ are detected, encompassing feature geometry g_j (like position, orientation, scale) and appearance descriptors d_j (e.g., gradient histograms). These features are designed to be invariant to changes in lighting and viewpoint [44, 37], enabling unique identification of object features across different images, as shown in Figure 5.3.

- **Feature Matching**: This stage identifies pairs of input images viewing the same scene segment by utilizing F_i for appearance-based comparison. A naive approach would compare all feature pairs across all image combinations, a computationally intense process. Instead, a more efficient method employs hierarchical feature indexing to rapidly identify visually similar image pairs and establish feature correspondences, significantly reducing computational load.

- **Geometric Verification**: Potentially overlapping image pairs identified in the matching stage undergo geometric verification to confirm that matched features correspond to the same physical point in the scene. This verification involves estimating transformations based on two-view geometry, utilizing homographies for static or planar scenes, and epipolar geometry for general motion scenarios, with robust techniques like RANdom SAmple Consensus (RANSAC) helping to filter out incorrect matches.

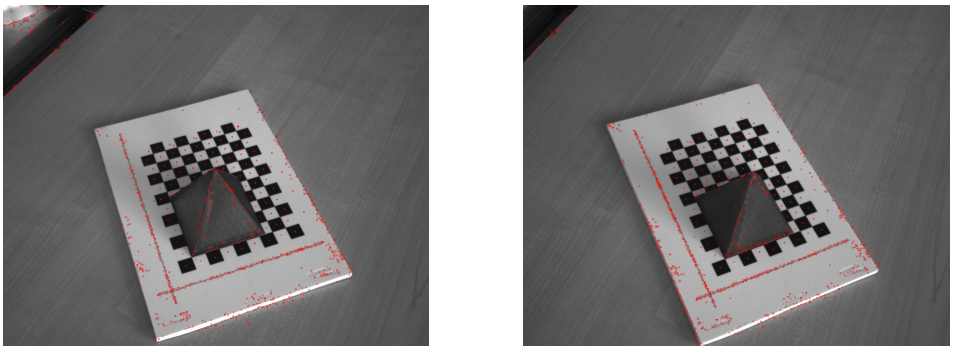


Figure 5.3: Possible representative feature points depicted as red dots in two example images obtained by COLMAP.

The outcome of these stages is a set of geometrically verified image pairs, alongside their established inlier correspondences and the defined geometric relationship, effectively setting the stage for transforming this data into a scene graph.

5.2 SPARSE RECONSTRUCTION

The sparse reconstruction process begins with the scene graph, derived from the correspondence search phase. This graph feeds into the reconstruction stage, yielding both intrinsic and extrinsic calibrations for each camera ($\mathbf{P}_c = \mathbf{K}_c[\mathbf{R}_c \mid \mathbf{t}_c]$) and a set of triangulated 3D points $M = \{\mathbf{M}_k \in \mathbb{R}^3 \mid k = 1 \dots N_M\}$. If the camera poses are not provided, however, this process starts without any prior knowledge of the scene's scale, orientation, or location, leading to an inherently arbitrary reference frame.

The primary challenge in linking these two-view geometries across overlapping images in the graph is the ambiguity of scale in each reconstruction and the error accumulation from imperfect pair reconstructions. There are two principal methods to address this: incremental and global reconstruction. Incremental reconstruction builds the scene starting from a two-view base, progressively integrating new images, performing triangulation, filtering, and refining through bundle adjustment to mitigate error accumulation. Conversely, global methods optimize camera motion across all views simultaneously, first establishing global orientations and then positions, concluding with a comprehensive bundle adjustment for refinement.

Incremental reconstruction is chosen for its robustness, especially with uncalibrated images and outlier management in two-view reconstructions. It starts by selecting an image pair to seed the model. The initial choice is crucial due to the potential inability to reconstruct the scene from a poor start. The algorithm then extends by registering new images against this foundation, calibrating camera positions, and progressively enhancing the scene model.

Triangulation extends the model by including new points observed in the newly added images, thereby enriching scene representation. This interdependence between image registration and triangulation necessitates a cyclical approach to ensure accuracy. Bundle adjustment is applied to refine 3D structures and camera parameters simultaneously. It minimizes the reprojection error, which is the discrepancy between the observed positions of points in images and their predicted positions based on the 3D model and camera parameters. Starting with initial estimates, it projects 3D points back into images, calculates the discrepancies, and iteratively adjusts the model to better fit the data. This optimization process continues until achieving minimal error and is essential for improving the accuracy and integrity of the 3D model constructed from multiple images, despite being computationally demanding.

Outlier management remains essential throughout, as even post-bundle adjustment, mismatches and inaccuracies persist. Filtering leverages multiple views to eliminate these errors, ensuring a coherent and accurate reconstruction. This approach emphasizes the balance between computational efficiency and reconstruction fidelity, highlighting the need for meticulous process management to achieve accurate and robust 3D models. An example of the output of this reconstruction can be seen in Figure 5.1.

5.3 DENSE RECONSTRUCTION

Transitioning from the preliminary sparse reconstruction, which primarily relies on a limited selection of distinct image features to determine structure and motion, the dense reconstruction phase aims to significantly enhance the scene's representation. Initially sparse and providing only a basic approximation of the real world, the objective is to transform this into a more detailed and comprehensive representation, such as a dense point cloud or a textured surface mesh.

Dense modeling fundamentally seeks to reconstruct a scene surface that aligns with the reprojected appearance across all images. The approaches differ in their problem parameterization: some operate within image space, deducing per-pixel depth and normals, while others approach the task directly within scene space, optimizing the surface itself. Each strategy offers distinct advantages for certain scenarios. Image space parameterization is inherently limited by the image resolution, ensuring each pixel correlates to a scene point, thereby facilitating scalability. Conversely, scene space parameterization can potentially reduce redundant computations and enhance multi-view occlusion handling but lacks inherent scalability without prior scene occupancy knowledge.

COLMAP adopts a hybrid dense modeling pipeline tailored for unstructured image sets. It segments the problem into three key phases: independently recovering dense depth maps for each image, merging these into a consistent point cloud, and finally generating a surface mesh and applying texture. This approach, balancing between image and scene space parameterizations, optimizes for both efficiency and robustness and

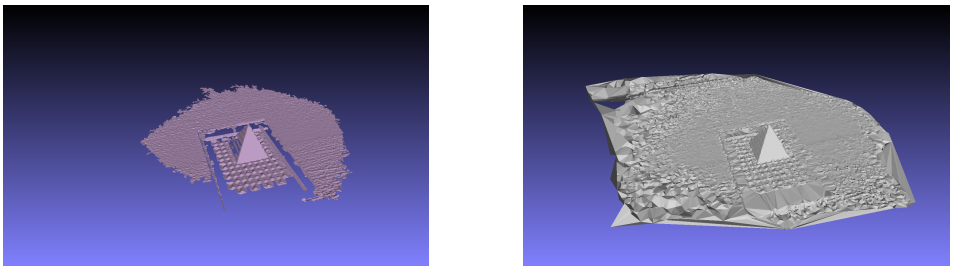


Figure 5.4: Final output of COLMAP reconstruction process adopting Poisson Surface Reconstruction (left) and Delaunay Triangulation (right).

performs very well on various benchmark datasets [50]. Various results of this process can be visualized in Figure 5.2.

5.4 RECONSTRUCTION

COLMAP employs various 3D reconstruction techniques, including Poisson Surface Reconstruction and Delaunay-based fusion methods like Delaunay Triangulation for dense point cloud and mesh generation. An example is shown in Figure 5.4.

Poisson Surface Reconstruction is a method for reconstructing surfaces from oriented point clouds. The technique works by solving a Poisson equation, which seeks to interpolate a smooth surface through the input points. The input is a set of points with associated normal vectors, which are usually obtained from the dense point cloud generated in the earlier stages of the COLMAP pipeline. Then, it generates an implicit surface model by assuming the gradient of the indicator function of the reconstructed surface aligns with the normals of the input points. This results in a scalar field whose iso-surface (surface of equal value) represents the reconstructed model.

Delaunay Triangulation is used to reconstruct surfaces by creating a mesh that satisfies the Delaunay condition (no point in a dataset should lie within the circumcircle of any triangle in the triangulation). The process begins with the generation of a mesh. Then, surface extraction algorithms identify the relevant subset of triangles that best represent the surface of the object or scene being reconstructed. This step often involves removing extraneous triangles that do not contribute to the main surface, based on criteria like normal orientation and proximity.

6 GENERALIZED BINARY SEARCH NETWORK

DNNs, through their remarkable capability to extract and aggregate informative features, have demonstrated superior performance across a variety of applications, including many computer vision tasks. A few examples are image and face alignment [56], object recognition[45], and stereo matching [57]. In MVS settings, deep learning has revolutionized depth estimation tasks by significantly improving performance and innovation. This is particularly true in addressing challenges such as occlusion, lighting variations, and textureless regions, which are often difficult for traditional methods to handle.

The pioneering work of MVSNet [61] introduced a novel depth prediction methodology and the concept of constructing 3D cost volumes from 2D features, employing 3D CNNs or Recurrent Neural Networks (RNNs) for depth estimation. This method, however, is not without its challenges, primarily the extensive memory demands of 3D cost volumes. Attempts to mitigate this through reduced resolution or progressive resolution enhancement have met with limited success, often at the expense of accuracy [11, 23, 58].

GBi-Net [41] tries to reduce memory consumption while retaining state-of-the-art performances on benchmarks like DTU and Tanks & Temples. It innovates by employing a generalized binary search approach within its CNN framework, significantly optimizing the efficiency of depth map generation and fusion without compromising depth prediction accuracy. This approach focuses on optimizing the depth hypothesis space, representing a major step towards achieving high efficiency and accuracy in 3D scene reconstruction.

6.1 CONVOLUTIONAL NEURAL NETWORKS

This section provides a concise overview of CNNs, beginning with the fundamentals of generic neural networks and advancing towards specific variants of CNNs, such as Deformable Convolutional Networks (DCNs) and FPNs. CNNs excel due to their ability to automatically and hierarchically extract features from raw images, learn spatial hierarchies of features, and efficiently manage the vast amount of data involved in image processing through shared weights and pooling operations. Their layered structure allows CNNs to capture both low-level features (like edges and textures) in early layers and complex, high-level features (like objects' parts) in deeper layers. This ability to learn feature representations directly from data, without the need for manual feature engineering, is a key factor in their success across a wide range of computer vision tasks.

6.1.1 *Deep Feedforward Networks*

Deep Feedforward Networks, or Multi-Layer Perceptrons (MLPs), aim to approximate functions by learning the mapping from input to output. Their structure and learning capabilities are particularly suited for tasks where the objective is to categorize inputs into predefined classes (classification) or predict continuous values (regression). These networks are characterized by a unidirectional flow of data through a direct acyclic graph structure. The architecture of neural networks is depicted as a series of interconnected functions forming a sequential chain. The complexity of a network's architecture, or its depth, is defined by the number of layers it contains, with current trends leaning towards more intricate designs. Training these networks involves adjusting parameters to closely mirror the target function, relying on a collection of training data for guidance. Gradient descent is commonly utilized for this learning process.

6.1.2 *Convolutional Neural Networks*

CNNs have emerged as one of the most effective network types in deep learning, distinguished by their unique use of convolutions over standard matrix multiplications in at least one of their layers. Pioneered nearly three decades ago by the development of LeNet-5 [34], CNNs showcased their potential by recognizing handwritten zip codes, a groundbreaking achievement for its time. This early success hinted at CNNs' applicability beyond synthetic datasets to real-world problems. The real exploration and adoption of CNNs surged with the advent of parallel computing on GPUs around 2010, leading to significant advancements. A pivotal moment came with the introduction of AlexNet [32], which significantly outperformed competitors in the ImageNet classification challenge, propelling the exploration and utilization of CNNs across various computer vision tasks, frequently surpassing traditional methods. Despite the lack of comprehensive theoretical explanations for CNNs' success, their effectiveness can be attributed to three core principles:

1. **Sparse Interactions**: By employing small kernels (ranging from 1x1 to at most 9x9), CNNs reduce the number of parameters required, leading to faster training and enhanced network efficiency. This contrasts with standard neural networks that connect every input to each output, resulting in a dense interaction model.

2. **Parameter Sharing**: CNNs utilize the same kernel across the entire input image, dramatically decreasing the number of parameters needed and, consequently, the memory requirements. This shared kernel approach ensures that all parts of the image are processed uniformly.

3. **Equivariant Representation**: This property ensures that shifting the input by a certain amount leads to an equivalent shift in the

output, facilitating the detection of features like edges regardless of their position in the image. This characteristic is particularly useful for tasks requiring consistent feature detection across varied image locations.

CNNs have proven exceptionally successful in fields like natural language processing, image and video recognition, and image segmentation.

3D Convolutional Neural Networks (3D CNNs) extend the principles of traditional 2D CNNs to analyze volumetric data, allowing them to capture spatial hierarchies in three dimensions. This capability makes 3D CNNs particularly effective for tasks such as medical imaging analysis, video recognition, and 3D shape recognition. Unlike 2D CNNs, which operate on images (2D data) by applying filters to capture width and height dimensions, 3D CNNs apply filters across width, height, and depth, enabling the extraction of features with spatial context in all three axes. This depth dimension can represent time in video sequences or actual spatial depth in 3D scans, allowing for the analysis of temporal sequences or volumetric shapes.

6.1.3 *Deformable Convolutional Networks*

DCNs [14] represent a significant advancement in the field of computer vision, particularly for tasks that require an understanding of complex and variable shapes within images. Unlike traditional CNNs that apply fixed, rigid filters across an image, DCNs introduce flexibility in the convolution operation by allowing the spatial sampling locations to adapt based on the input feature. This adaptability enables DCNs to better model geometric transformations and deformations in the visual data, making them particularly effective for tasks like object detection, semantic segmentation, and pose estimation where the objects of interest may vary in shape, size, or orientation. By learning the optimal displacement of sampling points for each convolution operation, DCNs can capture more relevant features and provide a better understanding of the image content. This approach has not only improved the performance on standard benchmarks but also opened new avenues for research and applications in areas where the precise modeling of object variability is crucial.

6.1.4 *Feature Pyramid Networks*

A specialized type of CNN adopted in the streamline of GBi-Net are FPNs [36]. They are a deep learning architecture designed for scaling the detection of objects across various sizes within an image. They address a common challenge in image recognition tasks, where objects of interest vary significantly in size, making it difficult for a single-resolution feature map to effectively capture all relevant details. FPNs tackle this issue by creating a pyramid of feature maps at multiple levels of resolution, enabling the network to detect objects at different scales.

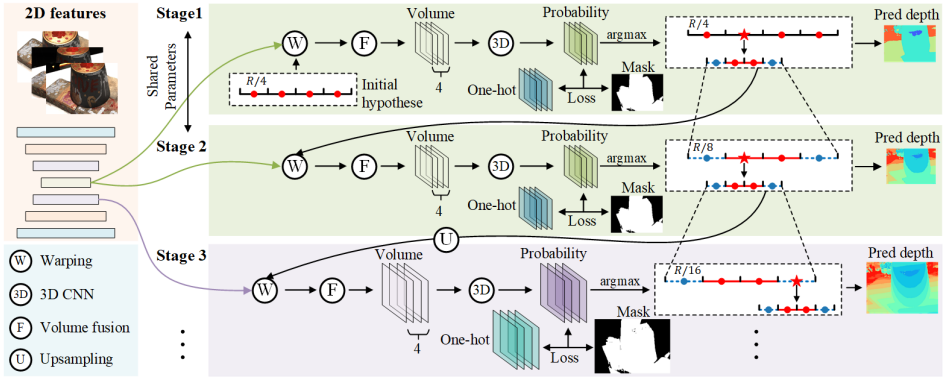


Figure 6.1: The multi-stage framework of GBi-Net. Credits: [41]

At its core, an FPNs takes a single-scale input and produces multiple feature maps, each of a different scale, through a bottom-up pathway. Then, a top-down pathway upsamples the coarse, high-level feature maps from higher pyramid levels, and merges them with the corresponding bottom-up maps using lateral connections. This process enriches the semantic value of the lower-level, high-resolution feature maps, making them more effective for detecting smaller objects while retaining the ability to recognize larger objects through the higher-level, lower-resolution maps.

FPNs are particularly useful in object detection and segmentation frameworks, where they significantly enhance the model's performance across a broad spectrum of object sizes. By leveraging the inherent multi-scale, pyramidal hierarchy of deep convolutional networks, FPNs efficiently encode multi-scale, semantic features in a single, unified architecture without the need for processing the image at multiple scales during inference.

6.2 GBI-NET

The GBi-Net architecture combines a 2D CNN for extracting visual features from images with a binary search network that iteratively estimates depth. It operates across multiple search stages, each beginning with the creation of 3D cost volumes through differential warping of feature maps between reference and source images. These volumes are then processed by 3D CNNs to predict depth labels. The full pipeline of the model can be seen in Figure 6.1

6.2.1 *Feature Extraction*

GBi-Net employs an FPNs for analyzing a set of images, which includes one reference image along with several source images. The FPNs generates feature maps at four different levels of detail. To further refine these feature maps, each level is processed through DCNs layers. These DCNs are designed to better capture the varied and dynamic aspects of scenes, making them highly effective for MVS applications, where understanding complex scene geometry is crucial.

6.2.2 *Cost Volume Regularization*

GBi-Net constructs detailed 3D representations of scene depth, called cost volumes, for each possible depth value being considered. This is done by shifting and rotating the features extracted from a reference image to match those of each source image, a process guided by the known camera angles and positions. The creation of these cost volumes is a crucial step and involves comparing and combining features from both the reference and source images. This comparison generates a dense, pixel-level 3D map that reflects how well the features of the source images align with those of the reference image across different depths.

To synthesize these individual cost volumes into a single, coherent volume, GBi-Net uses a weighted merging strategy. Each source image's contribution is determined by a specific weight matrix, emphasizing the importance of certain images over others based on their alignment accuracy.

This combined cost volume is then refined using a 3D UNet, a process known as regularization. The 3D UNet simplifies the cost volume by reducing its complexity down to a single channel, which essentially turns it into a map indicating the probability of each pixel being at a certain depth. Applying a Softmax function to this probability map allows the model to determine the most likely depth for each pixel. This streamlined approach facilitates both the calculation of the model's training loss and the final depth estimation, making GBi-Net an efficient solution for MVS tasks.

6.2.3 *Binary Search for MVS*

The binary search strategy is introduced for constructing 3D cost volumes, significantly reducing memory requirements without compromising efficiency. Traditional MVS techniques tend to densely sample potential depth values for every pixel, a process that consumes a significant amount of memory. Even though some cascade-based methods [23] aim to mitigate this by employing a stepwise refinement strategy from coarser to finer depth estimates, they still sample depth densely at each step, limiting model efficiency.

To enhance this process, the Bi-Net model introduces an innovative approach by segmenting the entire range of depth into two halves at each stage of depth estimation. It uses the midpoint of these halves as representative depth values (hypotheses). This binary search technique simplifies the process of building the cost volume and improves the efficiency of determining the correct depth labels. With each iteration, the method halves the search range by focusing on the half that's likely to contain the actual depth, significantly reducing the computational complexity and the memory footprint of the 3D cost volume.

This streamlined binary search approach leads to a marked reduction in the memory demands of the model, with the bulk of the memory require-

ment now coming from the 2D feature extraction phase rather than the construction and processing of large cost volumes. Experimental evidence confirms that Bi-Net not only conserves memory but also maintains, or even enhances, the accuracy of depth predictions. This makes it a highly efficient solution for MVS applications, effectively overcoming common challenges such as excessive memory use and errors in depth classification.

6.2.4 *Generalized Binary Search for MVS*

The Generalized Binary Search strategy enhances the binary search technique, specifically tackling the challenges of error accumulation and optimization inefficiencies. This refined method introduces three key improvements:

1. **Error Tolerance Bins (**ETB): After selecting the most relevant depth range (or bin) from a search stage, the strategy further divides this bin into two new ones for the next stage and introduces additional, smaller bins on both sides. These extra bins are designed to accommodate minor prediction errors, thereby expanding the search range slightly beyond the strict binary division. This flexibility is key for refining depth predictions and reducing the potential for error accumulation over successive stages.

2. **Gradient-Masked Optimization**: This optimization technique adopts a targeted approach to training, using depth maps as ground truth to generate precise labels. It cleverly addresses the challenge of handling depth values that fall outside the expected range by introducing a mask for each search stage. This mask ensures that only the gradient information from pixels within the predicted depth range is used for updating the model's parameters. This focused update mechanism significantly improves the specificity and efficiency of the training process.

3. **Efficient Gradient Updating**: Diverging from traditional models that accumulate gradients over multiple stages before updating, this approach opts for immediate parameter updates at the end of each stage. This strategy is more memory-efficient and facilitates faster model convergence, ensuring that the model remains responsive and accurate without the lag associated with cumulative gradient updates.

These mechanisms collectively refine the binary search strategy, ensuring high efficiency in memory usage and depth prediction accuracy without compromising on model performance.

7

MVSFORMER

As mentioned in Chapter 2, feature extraction, cost volume construction, and cost volume regularization are the three fundamental steps for learning-based MVS methods. As for GBi-Net (Section 6.2), FPNs are commonly used for feature extraction, enhancing multi-scale image features. Other advances include deformable convolutions [58, 60] and attention mechanisms [62, 65] to refine FPNs, yet challenges persist in generalizing across complex scenes. Cost volume regularization aims to enhance MVS performance by smoothing feature correlations, particularly in challenging scenarios like non-Lambertian surfaces or occlusions [61]. Despite their effectiveness, these methods still struggle with ambiguous matchings like reflections and texture-less areas due to their limited receptive fields and overly abstract high-level features.

MVSFormer [6] introduces a pioneering approach to MVS by leveraging Vision Transformers (ViTs). With their long-range attention capabilities, ViTs offer a promising alternative [16, 8], potentially providing a comprehensive global understanding and effective feature matching for MVS. ViTs' patch-wise feature encoding aligns well with the 1D feature matching challenges of MVS. Despite their potential, there's a notable gap in research on leveraging pre-trained ViTs for enhancing MVS.

MVSFormer integrates hierarchical ViT Twins [12] as its backbone, benefiting from efficient attention mechanisms and pyramid architecture for high-resolution training and improved outcomes. It also introduces a multi-scale training strategy tailored for MVS, addressing the challenge of training on low-resolution images while testing on high-resolution datasets. This strategy, combined with a novel approach to depth prediction, merges the strengths of regression and classification methods for more reliable depth estimation and superior point cloud results.

7.1 TRANSFORMERS

Vanilla transformers, introduced by [54] in their seminal paper "Attention is All You Need," have revolutionized the field of Natural Language Processing (NLP) and have subsequently made significant impacts in various domains, including computer vision. The core innovation of the transformer architecture lies in its attention mechanism, which allows the model to focus on different parts of the input data at different times, enabling it to capture complex relationships within the data. This mechanism is particularly adept at handling sequential data, making transformers highly effective for tasks such as translation, text summarization, and more recently, for image-related tasks.

Transformers consist of an encoder and a decoder, each comprising multiple layers of self-attention and position-wise fully connected layers. The self-attention mechanism allows the model to weigh the importance of different input elements irrespective of their positions in the sequence. This is a departure from previous sequence modeling approaches, such as RNNs and CNNs, which process data in a fixed order or rely on the spatial proximity of data points.

In the context of computer vision, and specifically in MVS depth estimation and 3D reconstruction, transformers have been adapted to handle image data, leading to the development of ViTs and their derivatives like MVSFormer. These models leverage the transformer's ability to capture long-range dependencies and its scalability to large datasets, offering significant advantages over traditional and convolution-based methods in terms of accuracy and efficiency. The adaptation usually involves reshaping image data into a sequence of patches, which are then processed by the transformer model to capture complex spatial hierarchies and relationships between different parts of the image, facilitating effective depth estimation and 3D reconstruction from multiple views.

7.1.1 *Vision Trasformers*

ViTs represent a shift in how deep learning models perceive and process visual data. Introduced by [16], ViTs approach image analysis by dividing an image into a series of fixed-size patches, treating each patch as a token similar to how words are treated in text processing. These patches are then linearly embedded, supplemented with positional encodings to retain spatial information, and fed into a standard transformer encoder structure. The model employs self-attention mechanisms across these patches, enabling it to dynamically focus on the most informative parts of the image throughout different layers of the architecture. This approach allows ViTs to capture complex patterns and relationships within the visual data. Unlike traditional convolutional networks, ViTs are not inherently biased towards local features, enabling them to leverage global context more effectively. This characteristic, combined with their scalability and ability to learn from large datasets, has positioned ViTs as a potent alternative to CNNs in many computer vision applications. Indeed, ViTs set new benchmarks across various vision tasks, including image classification [16], object detection [35], and segmentation [64], and have shown promise in specialized areas like optical flow [27] and point cloud processing [24]. Unlike CNNs, ViTs can model long-range dependencies more effectively but require substantially more data to achieve similar generalization, mainly due to their lack of inherent inductive biases such as translational invariance and locality.

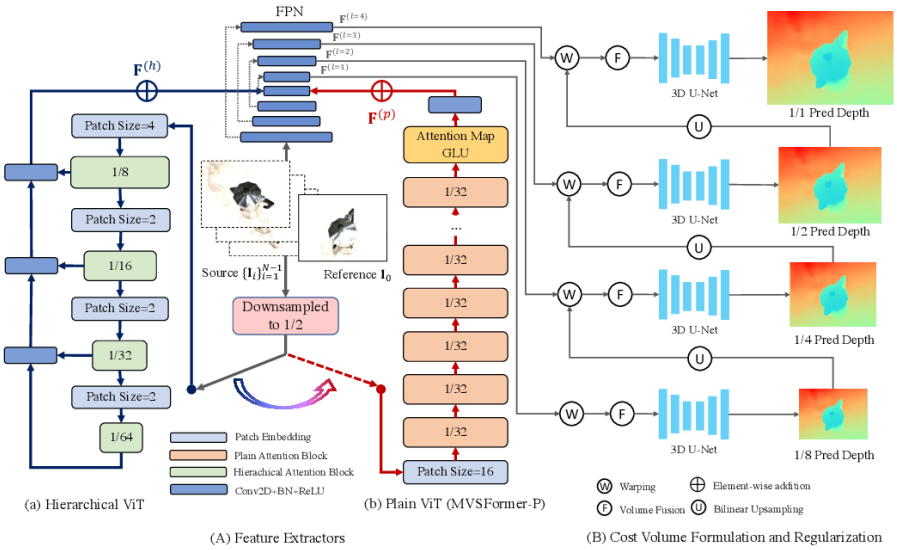


Figure 7.1: The overview of MVSFormer. (A) Feature extractors of hierarchical ViT (a) and plain ViT (b). Inputs for ViTs are downsampled to the 1/2 resolution. (B) Multi-scale cost volume formulation and regularization. *Warping*: warping source features with upsampled depth hypotheses for cost volumes. *Volume Fusion*: fusing cost volumes from all source views with respective visibility. Credits: [6].

7.2 MVSFORMER

The MVSFormer architecture processes a set of N images from different viewpoints, including a reference image I_0 and $N-1$ source images, alongside their camera intrinsics and extrinsics. MVSFormer utilizes either a hierarchical ViT known as Twins [12] or a standard ViT model DINO [9], each enhanced with unique training strategies. Considering we exclusively utilized the original MVSFormer, our discussion will focus solely on that model, omitting the DINO variant. The architecture, shown in Figure 7.1, employs multi-stage cost volume formulation and regularization to calculate the probabilities for various depth hypotheses, refining depth estimates from coarse to fine. Optimization is achieved using cross-entropy loss, with depth expectations determined during the inference phase.

7.2.1 *Feature Extraction*

In the MVSFormer framework, an FPN is employed as the primary feature extractor, augmented by pre-trained ViTs to capture global feature correlations, while the FPN focuses on detailed features. To manage computational and memory demands, images are first downsampled before being processed by the ViT, with the absolute position encoding resized to match the image scale through bicubic interpolation. The outputs from ViTs are then integrated with the top-level FPN encoder features, allowing the extraction of multi-scale features from the FPN decoder, which com-

bines the strengths of both ViT and CNN architectures for improved cost volume reliability.

The standard configuration of MVSFormer utilizes the Twins model as its core architecture, which is specifically optimized for handling images of varying resolutions. The Twins model tackles the need for efficient attention mechanisms and robust handling of positional information across different image scales. Furthermore, this setup is enhanced by integrating an extra FPN. This additional FPN plays a role in both encoding and upscaling features extracted at multiple scales. This integration enables the Twins model to adapt to and process images at various resolutions effectively, allowing for straightforward fine-tuning of the pre-trained model with minimal finetuning required to accommodate different image sizes.

7.2.2 *Efficient Multi-scale Training*

ViTs struggle with certain limitations due to their design, notably lacking translation invariance and locality. This means they don't naturally adjust to changes in image position or scale, which poses a challenge for handling the varied and high-resolution images required for MVS tasks. On the other hand, CNNs can better manage these variations thanks to their dynamic kernels and ability to perform random cropping, inherently adjusting to different input sizes. This flexibility is vital in MVS tasks to prevent the model from overfitting to a specific resolution and failing to generalize to higher resolutions.

To address this challenge in MVSFormer, the approach involves training the model on images of different scales, known as multi-scale training. This strategy requires keeping image sizes consistent within each training batch but allows for adjusting the number of images per batch based on their resolution to use memory more efficiently. By using gradient accumulation, MVSFormer can work with large batches of data, even when limited by memory. This involves breaking down a large batch into smaller subsets, processing each separately, and then combining their gradients to update the model. The training process randomly pairs each training instance with a resolution and batch size that accommodates the image size, ensuring that larger images are processed in smaller groups to keep memory usage in check. This technique not only improves the efficiency of training but also helps the model learn more effectively, reduces training variability, and enhances the effectiveness of normalization techniques like BatchNorm, leading to better performance.

7.2.3 *Correlation Volume Construction*

The creation of the multi-stage cost volume in the MVSFormer involves a systematic process that begins with setting up a series of inverse depth ranges for each stage of the multi-stage cost volume. To construct the correlation volume, features from source views are warped to align with

the reference view. This is done by transforming a 2D pixel in the reference image using the camera intrinsics and extrinsics of both reference and source views, as well as a depth hypothesis, to obtain its corresponding position in the source image.

To evaluate the similarity between the reference image features and those of the warped source images, the model segments features into groups by their channel distribution. It then assesses the similarity within each group between the reference and warped source features. By aggregating these similarity measures across all groups, a comprehensive cost volume is created, encapsulating the degree of feature match across different views.

Moreover, to refine this matching process, a 2D CNN is used to learn pixel-wise weight visibility for each source view based on the entropy of normalized correlations. These source feature correlations are then fused with their visibility weights to form the input for the 3D U-Net cost volume regularization. The regularization process outputs a pixel-wise 3D cost volume for each stage, ensuring that the depth estimation is as precise as possible at every stage of the process.

7.2.4 *Temperature-based Depth Prediction*

The MVSFormer model uses the 3D U-Net's output to create a probability volume for depth prediction by applying a softmax function. Traditional MVS methods either regress depth values directly (referred to as REG) or classify each pixel into discrete depth intervals (referred to as CLA).

The temperature-based method aims to combine the advantages of both approaches to improve depth map accuracy and reliability. In this method, a temperature parameter is introduced to modify the softmax operation applied to the cost volume, which is derived from feature correlations between different views. By adjusting the temperature, the method can control the sharpness of the probability distribution over depth values. A higher temperature leads to a softer distribution (mimicking classification), while a lower temperature results in a sharper distribution (mimicking regression).

During inference, instead of selecting the depth value with the highest probability (as in CLA) or computing the expected depth value across all hypotheses (as in REG), the temperature-based method calculates the expected depth using the modified softmax distribution. This allows the method to achieve smoother depth predictions and better handle uncertainties in depth estimation. This temperature adjustment enhances the model's performance across different stages of resolution without the need for retraining. It leverages the advantages of both depth estimation strategies, offering a flexible and effective solution for various datasets.

Part IV

RESULTS

8

RESULTS

In this chapter, we present a comprehensive evaluation of the produced depth map and reconstructed point clouds. We begin by detailing the assessment of depth maps, utilizing well-established evaluation metrics to quantify their accuracy, and proceed to describe the results obtained with the different models.

Following the depth map evaluation, we shift our focus to the reconstructed point clouds, which represent the final result of the MVS pipeline. We discuss the post-processing steps applied to the point clouds, aimed at enhancing their quality and usability. Then, we evaluate these point clouds against ground truth data using metrics such as accuracy, completeness, overall score, and percentage of spurious points, offering insights into the quality of the 3D reconstructions. These metrics serve to highlight the fidelity of the models to the actual scene geometry, as well as their robustness across various scene complexities.

Throughout this chapter, we aim to provide a transparent and detailed account of our results, underpinned by a rigorous evaluation framework.

8.1 DEPTH MAPS

As mentioned in Chapter 2, a depth map is an image that contains information relating to the distance of the surfaces of scene objects from a viewpoint. In simpler terms, it represents how far away each pixel in an image is from the camera that captured it. The values in a depth map measure the distance between the camera and the points in the scene for each pixel. Depth map estimation is a key component in 3D reconstruction, responsible for recovering the depth information that is lost during the projection of three-dimensional points onto the two-dimensional image plane. As outlined, this process begins by extracting distinctive features across multiple images and establishing correlations between various viewpoints. Following this, depth maps are estimated for each image. This estimation is grounded in the geometric and photometric consistencies derived from the initial feature correlations, ensuring that the depth values accurately reflect the spatial dimensions of the scene. The final step is the fusion of these individual depth maps into a unified 3D model. Consequently, depth maps are a key component in 3D reconstruction, serving as a critical measure for evaluating the performance of different methods.

8.1.1 *Evaluation Metrics for Depth Maps*

To quantitatively assess the performance of depth map-based 3D reconstruction methods, several error metrics are employed. These metrics allow

a comprehensive evaluation by highlighting different aspects of accuracy and reliability. We decided to adopt three different measures: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Depth Error Percentage (DEP).

Mean Absolute Error (MAE): measures the average magnitude of the errors in a set of predictions, without considering their direction. It's calculated as the average absolute difference between each predicted depth value and the corresponding ground truth value across all pixels. Mathematically, it's defined as:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\frac{|d_i - \hat{d}_i|}{d_i} \tag{13}$$

where N is the total number of pixels, d_i is the ground truth depth for pixel i, and $\hat{d}_i$ is the predicted depth for pixel i.

Root Mean Squared Error (RMSE): provides a measure of the magnitude of the error. By squaring the errors before averaging, RMSE gives a higher weight to larger errors. This makes it particularly sensitive to outliers. The RMSE is calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(d_i - \hat{d}_i)^2} \tag{14}$$

where the symbols represent the same quantities as in Equation 13.

Depth Error Percentage (DEP): quantifies the proportion of pixels in a depth map that exhibit an error surpassing certain predefined thresholds. This metric provides an intuitive measure of the depth estimation accuracy by highlighting the percentage of pixels that fail to meet specific accuracy criteria. The Depth Error Percentage for a threshold δ is defined as:

$$\text{DEP}(\delta) = \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\left(|d_i - \hat{d}_i| > \delta\right) \times 100\% \tag{15}$$

where N is the total number of pixels, d_i represents the ground truth depth for pixel i, $\hat{d}_i$ is the predicted depth for pixel i, and $\mathbb{1}$ is the indicator function, which equals 1 if the condition $|d_i - \hat{d}_i| > \delta$ is true (i.e., the absolute error is greater than the threshold δ) and 0 otherwise. The result is multiplied by 100 to express the error percentage. Chosen values for δ are $[1, 2, 3, 4]$, reflecting the percentage of pixels with an error greater than 1, 2, 3, and 4 millimeters from the ground truth, respectively. This metric is particularly useful for assessing the robustness of depth estimation methods against significant errors, providing a clear picture of the proportion of estimations that deviate markedly from the ground truth.

8.1.2 *Results*

This section presents and compares the results obtained from the three adopted methods: COLMAP, GBi-Net, and MVSFormer, highlighting their respective strengths and limitations.

In the analysis of depth maps generated by COLMAP for the six objects under study, as depicted in the second column of Figure 8.1, the depth maps exhibit a notably tidy appearance, underscoring COLMAP's precision in capturing the contours and the precise regularization process. However, a critical observation is the discontinuity within the depth maps. Specifically, these maps are characterized by the presence of pixels for which depth values have not been estimated, leading to gaps in the depth information. This phenomenon arises due to several factors inherent to the image-based 3D reconstruction process, in particular in our case it is mainly due to textureless areas and low lighting. Since depth estimation relies heavily on finding correspondences between different views, textureless surfaces can lead to insufficient data for accurately estimating depth. Poor lighting conditions and shadows can significantly affect the visibility of features in images, leading to areas where the algorithm struggles to find reliable data, resulting in gaps. Quantitatively, the percentage of pixels not estimated across all objects on average is around 4%. This metric highlights an area for improvement in COLMAP's depth estimation process, emphasizing the need for strategies that can enhance the continuity of depth information and minimize the incidence of unestimated pixels.

GBi-Net depth maps, shown in the third column of Figure 8.1, exhibit a notable improvement over COLMAP in terms of continuity, effectively eliminating gaps and presenting a more unified depth estimation across the scene. However, this improvement comes with its own set of challenges. GBi-Net depth maps sometimes display areas where depth values exhibit substantial variance from adjacent areas, resulting in less smooth depth maps. This variance can introduce abrupt changes in depth perception, affecting the visual coherence of the reconstructed scene. Additionally, while COLMAP has demonstrated a capacity for more complete depth estimation, particularly in regions of the image captured in fewer views such as corners, GBi-Net's estimations in these areas tend to lack accuracy.

The depth maps generated by MVSFormer are the smoothest among all the evaluated methods, producing depth maps that are both continuous and regular. This ensures that the depth transitions across the scene are visually coherent, making MVSFormer's outputs the most aesthetically pleasing and technically proficient of the group. The depth maps are uniform, with minimal abrupt changes in depth values, thereby offering a more refined and detailed representation of the scene's geometry. However, this achievement in smoothness and continuity comes with its trade-offs. MVSFormer sacrifices even more completeness than GBi-Net, particularly in areas of the scene that are less represented in the dataset, such as regions only visible in a limited number of views.

Table 8.1: Comparison of MAE and RMSE for COLMAP, GBi-Net, and MVSFormer under different lighting conditions.

(a) Artificial Light

Method	Mean Absolute Error	Root Mean Squared Error
COLMAP	0.3144	0.2596
GBi-Net	0.0172	0.0189
MVSFormer	**0.0076**	**0.0086**

(b) Natural Light

Method	Mean Absolute Error	Root Mean Squared Error
COLMAP	0.2084	0.2125
GBi-Net	0.0128	0.0157
MVSFormer	**0.0064**	**0.0078**

Despite the challenges and limitations associated with GBi-Net and MVSFormer, it's noteworthy that both methods demonstrated the capability to reconstruct areas without textures or with reflections, where COLMAP faced difficulties. This is illustrated in the depth maps shown in the fourth row of Figure 8.1, where COLMAP struggles to reconstruct the area affected by the neon light's reflection on the object. This achievement underscores the significant progress made by learning-based approaches in addressing some of the inherent challenges of 3D reconstruction, particularly in dealing with surfaces that lack distinctive features.

The comparative analysis presented in Table 8.1, Table 8.2 confirms what was visually apparent from the depth map images: MVSFormer outperforms both COLMAP and GBi-Net under both artificial and natural lighting conditions. This superior performance of MVSFormer aligns with expectations, highlighting its robustness and efficacy in depth map reconstruction across varying lighting scenarios.

A significant portion of the error associated with COLMAP's results can be attributed to areas that were not reconstructed, indicating that while COLMAP is capable of producing high-quality reconstructions, its performance is somewhat hindered by incomplete coverage of the scene. This aspect underscores the importance of considering the completeness of the reconstruction when evaluating performance.

During our testing, it was observed that GBi-Net has the potential to achieve better results than those presented, suggesting that with more fine-tuning, its performance could be significantly enhanced. However, GBi-Net's performance was found to be highly dependent on the preprocessing steps of the dataset, which involves using COLMAP's sparse reconstruction to obtain crucial parameters such as maximum and minimum depth values and depth ranges that are then fed to the model. This dependency contrasts

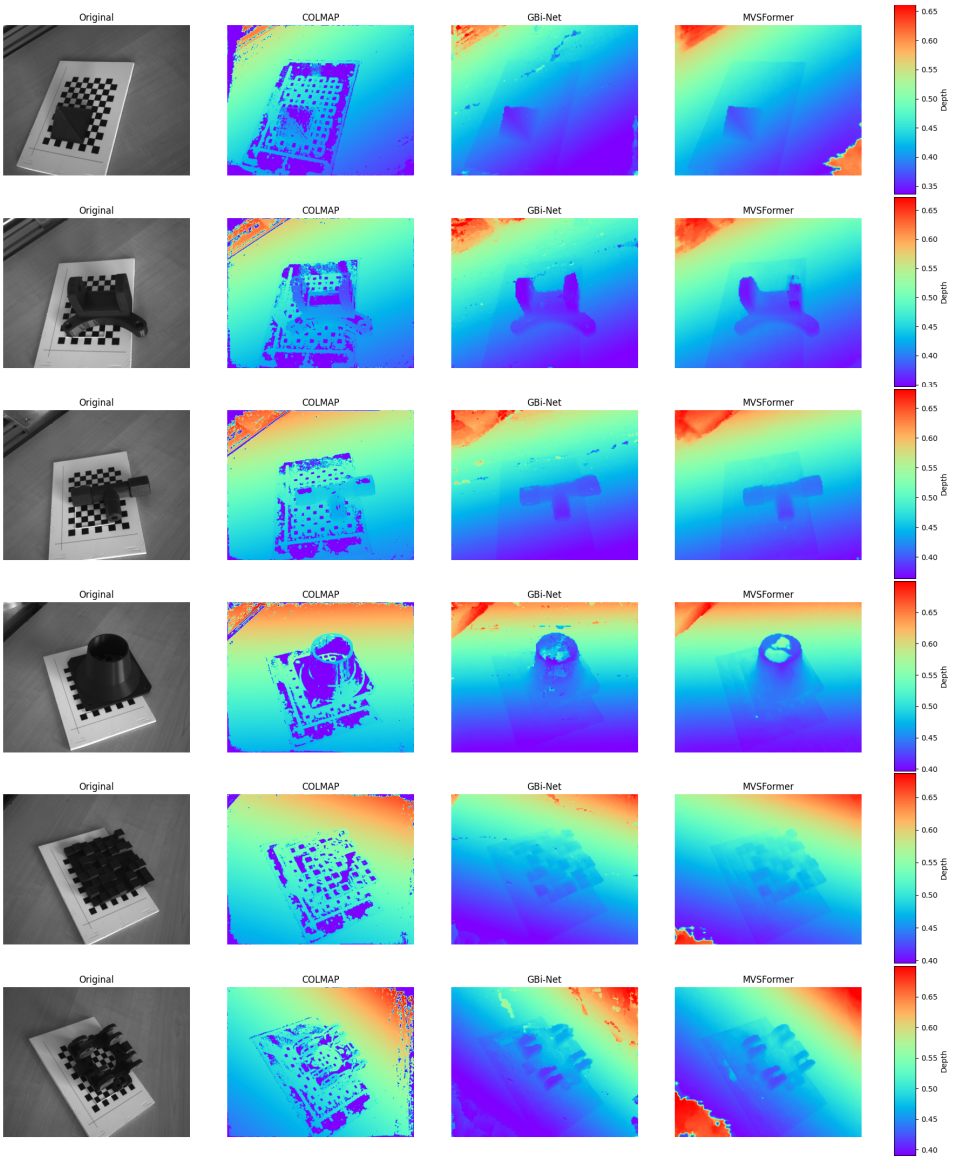


Figure 8.1: Comparative visualization of depth map reconstructions across the six objects. For each object, the first image represents the original scene, followed by the depth maps generated by COLMAP, GBi-Net, and MVSFormer, respectively. The depth is expressed in meters.

Table 8.2: Comparison of the percentage of pixels with an error larger than a specified thresholds, for COLMAP, GBi-Net, and MVSFormer under different lighting conditions.

(a) Artificial Light

Method	1mm (%)	2mm (%)	3mm (%)	4mm (%)
COLMAP	57.11	44.70	40.61	38.93
GBi-Net	55.79	38.59	30.10	25.17
MVSFormer	**50.12**	**38.38**	**25.00**	**19.24**

(b) Natural Light

Method	1mm (%)	2mm (%)	3mm (%)	4mm (%)
COLMAP	47.58	33.79	30.09	28.47
GBi-Net	51.47	30.71	23.79	19.90
MVSFormer	**43.70**	**25.81**	**19.08**	**15.29**

with MVSFormer, which did not exhibit the same level of sensitivity to the preprocessing steps, pointing to a more versatile and robust approach to depth estimation.

Furthermore, all models demonstrated improved performance under natural light compared to artificial lighting. This improvement is likely due to the brighter images and reduced reflections on surfaces that are characteristic of natural lighting conditions, as opposed to the more challenging lighting environments created by artificial neon light. The enhanced performance under natural lighting conditions suggests that lighting plays a critical role in the effectiveness of 3D reconstruction methods and could be a valuable avenue for future research. Specifically, investigating additional preprocessing steps to further optimize captured images for 3D reconstruction under various lighting conditions could constitute a significant area of study, potentially leading to advancements in the field and more accurate depth map reconstructions across a broader range of scenarios.

8.2 3D POINT CLOUDS

Section 8.2 delves into the evaluation of reconstructed point clouds, the final result of the 3D reconstruction process. This phase of analysis focuses on transforming the depth information obtained from COLMAP, GBi-Net, and MVSFormer into dense, geometrically accurate point clouds that represent the spatial structure of the scanned objects.

In the process of transforming depth information into a unified and geometrically accurate point cloud, COLMAP utilizes a multi-step approach. Initially, depth maps undergo a filtering phase, where outliers and erroneous depth estimates are identified and eliminated through consistency

checks across multiple views, ensuring the reliability of depth data for fusion. The core of COLMAP's methodology lies in its stereo fusion process, where the depth maps are merged into a single point cloud. This step involves projecting depth information into three-dimensional space using the camera's intrinsic and extrinsic parameters, subsequently combining these projections into a cohesive model. Throughout this process, special attention is paid to the geometric consistency and confidence levels of the depth measurements, with a weighting system applied to optimize the fusion based on accuracy and viewing angles. Furthermore, COLMAP refines the point cloud using visibility information to enhance the model's fidelity, ensuring that only points with consistent visibility across views are retained.

As for COLMAP, GBi-Net integrates both photometric and geometric consistencies for depth map filtration. Geometric consistency evaluates depth consistency across multiple views. Photometric consistency, on the other hand, utilizes probability volumes to assess the quality of depth hypothesis matching. These volumes, constructed in the cost volume regularization phase, represent the classification probabilities for depth hypotheses, or in other terms, the likelihood of each depth hypothesis for each pixel, quantifying how probable it is that a specific depth value is correct for a given pixel based on the multiple views. Given that the method encompasses K stages, yielding K probability volumes, the photometric consistency for each pixel is determined by averaging the maximum probabilities across K' stages, with the formula:

$$Ph(p) = \frac{1}{K'} \sum_{k=1}^{K'} \max\{P_k(j,p) | j = 1, ..., D\}$$

Here, $Ph(p)$ denotes the photometric consistency of pixel p, and D is the number of depth hypotheses. This calculation averages the highest classification probabilities across the first K' (e.g., 6 out of 8) stages, adjusting the resolutions of probability volumes to the highest stage's resolution for this computation. Depths with consistency scores below a certain threshold are discarded. This approach effectively filters outliers by leveraging photometric consistency maps to enhance depth map quality and subsequent point cloud reconstruction.

MVSFormer adopts the standard Gipuma algorithm for fusion. Gipuma [20] employs a strategy focused on patch-based processing, by dividing images into manageable patches for efficiency purposes. The core algorithm is very similar to COLMAP, whereas the hypotheses are evaluated based on how well the textures and colors match (photometric consistency) and whether the depth values adhere to the scene's expected geometry (geometric consistency), utilizing camera calibration data to ensure accuracy. While COLMAP's fusion strategy is designed to be comprehensive, prioritizing accuracy and the elimination of outliers, Gipuma focuses on a

(a) Reconstructed. (b) Plane removal. (c) Clustered.

Figure 8.2: Visualization of point cloud processing steps: The first image (Reconstructed) shows an example of a reconstructed point cloud. The second image (Plane removal) illustrates the point cloud after the removal of the major plane, to focus on the objects of interest. The final image (Clustered) displays the result of clustering, highlighting the largest cluster in red and other clusters in blue.

highly parallelizable, patch-based approach that tries to balance accuracy and computational efficiency.

8.2.1 *Postprocessing*

In the post-processing phase of our reconstructed point clouds, a procedure is implemented to ensure their readiness for evaluation against the ground truth point clouds. This phase is important, given that the ground truth datasets contain solely the objects of interest, missing any external elements present within the experimental setup, such as the laboratory table or the checkerboard. Consequently, the reconstructed point clouds necessitate a refinement process to enable a fair and accurate comparison.

The initial step in this refinement process involves the removal of the laboratory table from the reconstructed point cloud. This is achieved through the application of a plane segmentation technique using the 'segment_plane' function available in Open3D. This function identifies and separates the planar surface corresponding to the table, allowing for its exclusion from the point cloud, thereby isolating the objects of interest. Figure 8.2b illustrates this plane removal process, showcasing the before and after states of the point cloud.

Subsequently, the point cloud undergoes a cleaning process to eliminate outliers and spurious points, which may result from the table's removal or arise from inaccuracies in the reconstruction process. To address this, a clustering algorithm, specifically HDBSCAN [38], is applied to the point cloud. This method clusters the points based on their density, facilitating the identification and retention of the primary cluster, which represents the object of interest. Figure 8.2c provides a visual comparison of the points labeled as main cluster (red) and spurious (blue). By focusing on the cluster with the highest number of points, we ensure the exclusion of outliers and erroneous reconstructions, streamlining the point cloud for further analysis. It's important to highlight that the cleanliness of the

reconstructed point cloud, especially in terms of the number of spurious points after the removal of the plane, is part of our evaluation.

The final step in the post-processing phase involves aligning the refined reconstructed point cloud with the ground truth point cloud. This alignment is necessary because the two point clouds exist in distinct coordinate systems. Specifically, the ground truth point cloud is positioned within the Pytorch3D coordinate system, where its origin is centered on the object, while the reconstructed point cloud operates within the reference camera coordinate system. All necessary transformations between these systems are well-defined, guaranteeing that we can accurately align the two point clouds.

8.2.2 *Evaluation Metrics*

In assessing the quality of 3D reconstructions, it is crucial to employ a set of evaluation metrics that enable a comprehensive analysis of the reconstructed models. For our study, we adopt the evaluation metrics utilized by the well-known DTU dataset, a standard benchmark in the field of 3D reconstruction [1], and adopted by various other datasets.

Accuracy: measures the closeness of the reconstructed points to the actual object surfaces. It is calculated as the average distance from each point in the reconstructed model to the nearest point on the ground truth surface. A lower average distance indicates a reconstruction that closely approximates the real object, highlighting the precision of the method.

Completeness: Completeness evaluates how well the reconstructed model captures the entirety of the object's surface. It is determined by the average distance from points on the ground truth surface to the nearest point in the reconstructed model. A low value signifies that the majority of the object's surface has been successfully captured in the reconstruction.

Overall Quality: This metric combines accuracy and completeness to provide a holistic assessment of the reconstruction's quality. It balances the precision of the reconstruction with its ability to capture the entire object, offering a comprehensive view of the method's effectiveness.

Number of Spurious Points: While not directly part of the DTU metrics, the number of spurious points remaining in the reconstructed model after the post-processing step of outlier filtering is an important aspect of our evaluation. A lower number of spurious points indicates a cleaner, more accurate reconstruction, reflecting positively on the method's ability to generate high-quality point clouds.

8.2.3 *Results*

In Table 8.3 we present the results for point cloud reconstruction metrics. Under both lighting conditions, MVSFormer demonstrates superior accuracy, confirming its effectiveness as observed in the depth map results. This suggests that MVSFormer's approach to depth estimation and point cloud

Table 8.3: Comparison of point cloud reconstruction metrics across COLMAP, GBi-Net, and MVSFormer under varying lighting conditions, with lower values indicating better performance for all metrics.

(a) Artificial Light

Method	Accuracy	Completeness	Overall	Spurious Points (%)
COLMAP	2.39	4.52	3.45	13.97
GBi-Net	1.27	**4.02**	**2.65**	43.15
MVSFormer	**1.04**	5.46	3.25	**7.16**

(b) Natural Light

Method	Accuracy	Completeness	Overall	Spurious Points (%)
COLMAP	2.31	4.67	3.49	10.32
GBi-Net	1.15	**3.26**	**2.21**	16.81
MVSFormer	**1.04**	4.49	2.77	**8.26**

reconstruction is particularly robust, capable of generating precise depth information across different lighting scenarios.

However, when examining completeness, a measure of how well the reconstructed point cloud captures the entirety of the object's surface, GBi-Net shows a tendency to produce more complete depth maps, contrary to MVSFormer, which, despite its high accuracy, often results in less complete reconstructions. This discrepancy highlights the impact of the different fusion techniques employed by each method, with GBi-Net potentially benefiting from its approach to integrating depth information across multiple stages.

The post-processing step, important for refining the point clouds and enhancing their quality, introduces another layer of variability to the results. The reliance on a clustering algorithm to identify and retain the most relevant cluster can occasionally lead to inaccuracies, necessitating a repetition of the process in some instances. This aspect underscores the importance of the spurious points metric in our comparison, offering insight into the presence of erroneous or irrelevant points within the reconstructed point clouds. Here, MVSFormer excels once again, consistently producing fewer spurious points compared to the other methods. This can be attributed to two primary factors: firstly, the likelihood of a more strict threshold being applied to classify points as accurate 3D representations, and secondly, the smoothness and regularity of the depth maps produced. This efficiency in filtering out irrelevant data points further attests to MVSFormer's robustness in generating high-quality 3D reconstructions.

Overall, the consistency between the depth map and point cloud reconstruction results reinforces MVSFormer's leading position in terms of accuracy and the effective handling of spurious points. Nonetheless,

the variations observed in completeness and the challenges encountered during post-processing highlight the complex interplay of factors influencing 3D reconstruction outcomes. These include the specific fusion and clustering techniques employed, which significantly affect the final quality of the reconstructions.

9 TEXTURES AUGMENTATION

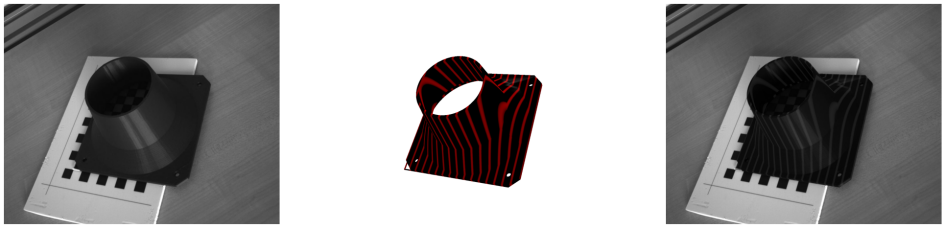
In the final results chapter of this thesis, we explore a new approach in our comparative study, focusing on the models' performance when faced with images augmented with a texture of red lines, mimicking the pattern projected by a laser projector. This experimental approach addresses a notable challenge encountered throughout our analysis: the struggle of reconstructing models with textureless regions. Due to the nature of our dataset, it was necessary to find a solution that would improve the accuracy of depth estimation and 3D reconstruction of our objects, which are predominantly dark and without any color variations.

The introduction of a simulated laser projector pattern onto the objects serves as a way to add artificial texture to these otherwise feature-sparse images. This method is rooted in the understanding that MVS techniques significantly benefit from distinct visual features to accurately triangulate points in space and generate detailed depth maps [26, 59]. Textureless surfaces, lacking these critical visual cues, often result in ambiguous depth estimations and subsequently, incomplete or inaccurate point clouds.

By artificially enhancing the texture of the objects with a consistent pattern, we aim to provide each model with additional data points for feature matching across multiple views. This chapter examines how such augmentation influences the depth map generation and the overall quality of the reconstructed point clouds. Specifically, we assess whether the introduction of these red lines can effectively mitigate the issues related to textureless surfaces and lead to improvements in the models' performance metrics.

Notably, COLMAP demonstrates a marked performance improvement, benefiting greatly from the addition of artificial textures which mitigates its previous limitations with untextured surfaces. Conversely, GBi-Net and MVSFormer show more modest accuracy enhancements, underscoring their inherent robustness in depth estimation across varied textural scenarios. However, an advancement for these models was observed in the substantial reduction of erroneously estimated pixels, indicating that while their performance in accurately estimated areas remained stable, the frequency of significant errors in challenging regions was greatly diminished. This outcome confirms the limitations of traditional methods in textureless areas but also proves the efficacy of incorporating artificial textures for enhancing 3D reconstruction accuracy. At the same time, it highlights the robustness of learning-based approaches in practical, real-world applications.

In fact, this approach was designed to explore whether employing such artificial texturing could serve as a viable strategy in practical applications, potentially through the use of actual laser projectors. The results from



(a) Original. (b) Rendered. (c) Combined.

Figure 9.1: A three-part illustration showcasing the original image, the rendered 3D mesh with the wave pattern texture applied, and the final composite image combining the textured object with the original scene.

this experiment have indeed confirmed the effectiveness of this approach. The enhanced texture led to noticeable improvements in the models' performance, particularly in areas that previously posed challenges due to their textureless nature. This success not only validates the concept but also suggests that implementing such texturing techniques in real-world scenarios, using physical laser projectors, could significantly benefit 3D reconstruction processes.

9.1 TEXTURE AUGMENTATION PROCESS

In the process of adding simulated laser patterns to our dataset, we utilized Blender, a comprehensive open-source 3D creation suite, to project a wave pattern onto the 3D models. Blender allows the generation of a texture file based on the specific object, that can then be integrated into other rendering platforms, such as PyTorch3D, for further processing.

To apply the newly created texture to the 3D models and generate the augmented images for our dataset, we turned to PyTorch3D. It enabled us to render the 3D mesh with the added wave pattern texture, positioning it identically to the original image captures. This step ensured that the textured model aligned perfectly with the original scene, maintaining the integrity of the spatial relationships and camera perspectives established during the initial dataset creation in Chapter 4.

Following the rendering, we created a mask of the object to isolate it from its surroundings. This mask was used to overlay the textured object onto the original image. By doing so, we effectively simulated the presence of the laser-projected pattern on the object within the natural environment of the original dataset images. This methodical approach allowed us to create a set of images that closely resemble what would be observed if a real laser projector were used to enhance the textures of the objects before capturing their images for 3D reconstruction. The process is illustrated in Figure 9.1.

Table 9.1: Comparison of Mean Absolute Error and Root Mean Squared Error for COLMAP, GBi-Net, and MVSFormer under different lighting conditions in the case of augmented images.

(a) Artificial Light

Method	Mean Absolute Error	Root Mean Squared Error
COLMAP	0.0268	0.0629
GBi-Net	0.0165	0.0187
MVSFormer	**0.0115**	**0.0130**

(b) Natural Light

Method	Mean Absolute Error	Root Mean Squared Error
COLMAP	0.0227	0.0581
GBi-Net	0.0169	0.0198
MVSFormer	**0.0059**	**0.0094**

9.2 DEPTH MAPS

The inclusion of textured patterns on objects within our dataset confirmed our hypothesis, yielding improvements in the performance of the MVS techniques. Notably, this effect is especially present in the performance boost observed for COLMAP upon transitioning from original to textured object scenarios. The positive impact of this texturing approach is further validated by the data presented in Table 9.1 and Table 9.2, which adopt the same metrics introduced in Section 8.1.1.

Under artificial lighting, COLMAP's mean absolute error significantly decreased from 0.3144 to 0.0268, and its root mean squared error dropped from 0.2596 to 0.0629. A similar trend is observed under natural lighting, where mean absolute error improved from 0.2084 to 0.0227, and RMSE from 0.2125 to 0.0581. These enhancements are impressive, especially when contrasted with the changes noted for GBi-Net and MVSFormer, which, while also showing improvement, do not exhibit as dramatic a reduction in errors as COLMAP does.

Moreover, before the addition of textures, COLMAP exhibits relatively high percentages of pixels with errors exceeding the thresholds, particularly under artificial light, where on average 57.11% of pixels have an error greater than 1mm, and even under natural light, this figure stands at 47.58%. However, with the introduction of the wave pattern textures, these percentages drop dramatically to 6.35% under artificial light and to 6.00% under natural light for the same 1mm threshold.

The significant enhancement in COLMAP's performance can be attributed to its challenge with textureless surfaces. By integrating simulated laser patterns, which effectively introduce artificial textures onto the ob-

Table 9.2: Comparison of the percentage of pixels with an error larger than a specified thresholds, for COLMAP, GBi-Net, and MVSFormer under different lighting conditions in the case of augmented images.

(a) Artificial Light

Method	1mm (%)	2mm (%)	3mm (%)	4mm (%)
COLMAP	**6.35**	**4.74**	**4.29**	**4.07**
GBi-Net	28.75	20.74	19.06	18.42
MVSFormer	18.40	16.46	15.79	15.30

(b) Natural Light

Method	1mm (%)	2mm (%)	3mm (%)	4mm (%)
COLMAP	**6.00**	**4.40**	**4.00**	**3.82**
GBi-Net	26.76	19.95	18.38	17.79
MVSFormer	12.55	10.35	9.75	9.31

jects, COLMAP's ability to triangulate points and estimate depth more accurately is markedly improved. This suggests that the previously observed large errors in COLMAP's performance are predominantly due to its inability to deal with textureless regions.

On the other hand, GBi-Net and MVSFormer exhibited only marginal or no improvements in accuracy with the introduction of textured patterns, indicating their inherent robustness in depth estimation regardless of texture presence or absence. This observation suggests that these models are well-equipped to handle textureless surfaces, maintaining a consistent level of accuracy even without the aid of artificial textures. However, an improvement for both models can be noticed in the reduction of erroneously estimated pixels. This implies that while the accuracy in high-confidence areas remains relatively unchanged, the inclusion of textures diminishes the instances of large errors in areas where depth estimation was previously uncertain or inaccurate.

9.3 POINT CLOUDS

Incorporating textured patterns into our dataset has markedly enhanced point cloud reconstruction, with notable advancements showcased in Table 9.3. The addition of artificial textures yielded benefits across all evaluated metrics, not only spotlighting COLMAP's significant performance enhancements but also highlighting the diminished uncertainty in depth estimations for challenging pixels by the learning-based approaches.

Under artificial light, the accuracy of COLMAP improved from 2.39 to 0.93, while its completeness metric saw a dramatic reduction from 4.52 to 2.68, indicating a more detailed and comprehensive capture of

Table 9.3: Comparison of point cloud reconstruction metrics across COLMAP, GBi-Net, and MVSFormer under varying lighting conditions, in the case of augmented images.

(a) Artificial Light

Method	Accuracy	Completeness	Overall	Spurious Points (%)
COLMAP	0.93	2.68	1.81	8.13
GBi-Net	**0.57**	3.01	1.79	9.02
MVSFormer	**0.57**	**2.6**	**1.59**	**4.98**

(b) Natural Light

Method	Accuracy	Completeness	Overall	Spurious Points (%)
COLMAP	1.06	2.83	1.94	10.95
GBi-Net	**0.58**	**2.77**	**1.68**	**7.2**
MVSFormer	0.59	3.22	1.9	15.53

the object's geometry. GBi-Net and MVSFormer achieved even greater accuracy improvements, both reaching a notable accuracy of 0.57, and MVSFormer excelled in completeness, improving from 5.46 to 2.6. Another significant outcome from the introduction of textures is the reduction in the percentage of spurious points, for all three models in both lighting scenarios. These results highlight the effectiveness of adding textures in addressing the challenges posed by textureless regions, and especially mitigating the estimation uncertainty for difficult pixels.

The introduction of textured patterns also yielded notable improvements under natural lighting conditions, with all methods demonstrating enhanced accuracy and completeness. However, it's worth noting that in the case of natural light, there was a slight increase in the percentage of spurious points for MVSFormer. This uptick likely stems from the postprocessing steps, such as plane removal and clustering. While these steps are essential for refining the final point cloud, they also introduce additional opportunities for errors to influence the results.

The results underscore the inherent robustness of learning-based approaches like GBi-Net and MVSFormer in depth estimation and 3D reconstruction, even in the face of challenging textureless surfaces. While traditional methods like COLMAP still show notable improvements with the addition of textures, learning-based models are increasingly demonstrating their resilience, maintaining high performance levels whether textures are present or not. However, the reduction in erroneous estimated pixels for these models, particularly in areas of uncertain depth estimation, still emphasizes the efficacy of preprocessing and image augmentation steps.

In summary, the addition of simulated laser textures has proven to be a successful strategy for enhancing the quality of 3D reconstructions, especially for objects lacking natural textures. This experiment not only validates the hypothesis that artificial texturing can improve reconstruction outcomes but also highlights the evolving capabilities of learning-based MVS techniques in providing robust solutions for 3D modeling challenges.

Part V

FINAL REMARKS

10 CONCLUSIONS

Throughout this thesis, we conducted a detailed evaluation of three MVS reconstruction methods: COLMAP, GBi-Net, and MVSFormer.

At the beginning of our work, we delved into a search of the literature to identify the state-of-the-art methods in MVS reconstruction. Given the proliferation of learning-based techniques in recent years, understanding the current landscape was instrumental in laying the groundwork for our study, allowing us to navigate through the myriad of methods available. An observation from our literature review was the research community's growing inclination towards developing methods such as monocular depth estimation and wide landscape reconstruction. These advancements, while impressive, often prioritize efficiency over accuracy to accommodate the vast spatial scales and dynamic environments they are designed for. However, such a compromise is unsuitable for industrial applications, where precision must be prioritized. The selection of models mentioned above was driven by the desire to prioritize this type of application. At the same time, we tried to cover a broad spectrum of approaches, ranging from traditional techniques to those based on CNNs and, finally, to innovative methods employing transformers. Furthermore, these methods have demonstrated state-of-the-art results on the DTU dataset, a benchmark that closely mirrors the complexities and demands of industrial scenarios.

From a comparative study, our work progressed to become a complete pipeline that can be implemented and customized by a diverse array of users, including companies and researchers, particularly for specialized tasks such as those encountered in industrial settings. One key feature is the use of pretrained models. This eliminates the need for the time-consuming and resource-intensive steps of gathering data and training the model from scratch. For cases where additional training is necessary, we've proposed a cost-effective solution that involves using 3D models to create ground truth point clouds and 3D printed objects for the training images. This method avoids the expense of laser scanning equipment traditionally used to obtain ground truths.

In assembling our dataset, we aimed to cover a wide spectrum of object shapes and complexities. Our goal was to create a complete dataset that could challenge the MVS methods across various scenarios. By selecting objects with diverse characteristics, from simple geometric shapes to more intricate designs, we ensured our dataset would provide a comprehensive test bed. This approach allowed us to assess how well each reconstruction method performs not just in ideal conditions, but also when faced with real-world challenges.

Section 8.1 of the thesis present a detailed evaluation of depth map results and the analysis of reconstructed 3D point clouds, under different

lighting conditions. Our findings show that MVSFormer generally offers the best accuracy and does a good job at minimizing spurious points, consistent with what we saw in the depth map images. Despite its accuracy, we noticed that GBi-Net and even COLMAP in some cases could provide more complete depth maps. This is primarily due to the different ways each method handles the fusion of information during reconstruction. We also found that all models showed improved performance under natural light due to brighter images and reduced reflections. The primary limitations observed include COLMAP's tendency to generate depth maps with incomplete areas, failing to estimate depth in certain sections, which results in holes. Meanwhile, GBi-Net's effectiveness was significantly influenced by the preprocessing requirements, particularly its reliance on COLMAP's sparse reconstruction to derive essential parameters. On the other hand, MVSFormer demonstrated a more adaptable and resilient approach, showing less sensitivity to the specifics of preprocessing steps.

In Section 8.2, we delve into the process of evaluating reconstructed 3D point clouds, the most important step in comparing the effectiveness of the methods. The postprocessing step plays a crucial role in refining the point clouds. This includes the removal of extraneous objects, such as the laboratory table used during image capture, and the application of clustering algorithms designed to identify and eliminate outlier points. To facilitate a comprehensive comparison, we adopted a series of metrics widely used in the field: accuracy, completeness, and overall quality. As an additional evaluation metric, we included the percentage of spurious points before the clustering removal, to understand how well each model can filter out erroneous estimations.

Through this evaluative lens, MVSFormer consistently emerges as the most accurate, maintaining its precision across both artificial and natural lighting scenarios. Conversely, GBi-Net is observed to yield more complete point clouds, underscoring the varying impacts that different fusion techniques can have on the reconstruction outcome. In addition, MVSFormer is the most efficient in handling spurious points, proficiency attributed to the inherent smoothness and regularity of its depth maps.

Finally, all models showed improved performance under natural light, likely because the images are brighter and have fewer reflections. This observation opens up possibilities for future studies, such as exploring additional preprocessing steps to enhance image quality for 3D reconstruction tasks. Our work lays down a pipeline that simplifies the use of MVS techniques, making advanced 3D reconstruction more accessible and adaptable for a wide range of applications.

In the concluding chapter of the thesis, we delve into a new comparative study focused on how different models fare when analyzing images augmented with red line textures, simulating laser projector patterns. This experiment specifically tackles the issue of reconstructing models from textureless regions, which presents a challenge due to the objects' dark, uniform color. By introducing simulated laser patterns, we aim to enhance object textures, aiding in more accurate depth estimation and

3D reconstruction. This approach leverages MVS techniques' dependency on distinctive visual features for accurate space triangulation and depth map generation. The addition of artificial textures is posited to mitigate depth estimation ambiguities caused by textureless surfaces, potentially improving model performance metrics. This chapter evaluates the impact of this textural enhancement on both depth map generation and overall quality of reconstructed point clouds, investigating whether such artificial texturing could be practically implemented with real laser projectors.

The experiments demonstrated significant improvements in the performance of all models, particularly under artificial lighting conditions. The introduction of simulated laser patterns effectively enhanced the models' ability to accurately reconstruct point clouds from images with previously challenging textureless surfaces. Notably, COLMAP showed substantial improvement, benefiting greatly from the added textures. This suggests that integrating artificial texturing could be a viable solution for enhancing 3D reconstruction processes, potentially employing real laser projectors in practical applications.

10.1 FUTURE WORK

The exploration and findings presented in this thesis lay a solid foundation for future advancements in MVS reconstruction. While the evaluated methods, COLMAP, GBi-Net, and MVSFormer, demonstrate notable capabilities, the pursuit of more accurate and robust 3D reconstruction techniques suggests various paths for future research and development. These include:

Fine-Tuning of Parameters: After a preliminary comparison between the models, a fine-tuning step of their operational parameters could significantly impact the performance of these methods. Future work could involve a detailed analysis to identify the optimal parameter settings for each method, tailored to different reconstruction scenarios. This could enhance accuracy, completeness, and efficiency across a broader range of conditions.

Additional Training for Specific Tasks: Customizing the models through additional training sessions tailored to the specific requirements of the task at hand presents a promising direction. This becomes feasible thanks to the facility with which new data and their corresponding ground truths can be acquired through the approach we have proposed. By feeding the models data that closely resemble the target application scenario, such as industrial environments, the models can learn to navigate the unique challenges of these settings more effectively.

Exploring New Models: The MVS research landscape is continuously evolving, with new models like MVSFormer++ [7] setting new benchmarks on standard datasets such as DTU. Investigating these newer models and their methodologies could bring further improvements in depth estimation and point cloud reconstruction quality.

Incorporation of Sparse Depth Data: The addition of sparse depth data as an input to the models, as suggested by approaches like Region-Aware MVSNet [63], holds particular promise for scenarios with consistent environments, such as industrial settings. This strategy could provide a significant boost in reconstruction accuracy by leveraging prior depth information to guide the reconstruction process.

Combating Textureless Regions: One of the enduring challenges in MVS reconstruction is accurately capturing areas with little to no texture. Future research could explore innovative solutions to this problem, such as the strategic addition of textures through the use of laser projectors. By projecting known patterns onto textureless surfaces, models could gain additional reference points for depth estimation, potentially overcoming one of the major hurdles in accurate 3D reconstruction.

In summary, the path forward in enhancing MVS reconstruction techniques is multi-faceted, encompassing parameter optimization, model customization, exploration of cutting-edge approaches, and innovative solutions to longstanding challenges. As the field progresses, these future directions not only promise to refine the accuracy and applicability of MVS methods but also to expand their potential in transforming our ability to replicate the physical world in digital form accurately.

ACKNOWLEDGEMENTS

I stand at the completion of this thesis with a heart full of gratitude. This journey, filled with challenges and achievements, has been made possible by the unwavering support and encouragement from many. First and foremost, I extend my deepest appreciation to my supervisor. Your expertise and insights have not only shaped this thesis but have also inspired me to push the boundaries of my understanding and capabilities. Also, I would like to extend my sincere thanks to the large body of incredible researchers on the second floor of the C3 building, for some of the most fun lunches of the last months.

To Marta, my compass and my anchor, words fall short of expressing my gratitude. Your unwavering support, your belief in my potential, and your encouragement have been my strength. This achievement is as much yours as it is mine. Without you by my side, this goal might have remained beyond my reach. Thank you for being my constant source of motivation and love. Alongside Marta, I need to thank Malachia. I will never understand how such a small creature can make you feel that deeply loved and appreciated. Malachia, thank you for the countless moments of happiness and for being our furry little cheerleader.

My family, my foundation, deserves my heartfelt thanks. You have supported me in ways I could never have imagined, offering unwavering belief in my aspirations, even at times when it was underserved. Your sacrifices and unconditional love have shaped me, and I am eternally grateful for everything you have done for me.

Lorenzo, your friendship and readiness to lend a hand have meant the world to me. Remaining one of the last survivors of our group, your presence and support have been a constant reminder of the strength found in enduring friendships. Thank you for being there, always.
And Federico, my long-lasting roommate. Your companionship during our shared living experiences has added invaluable memories to my journey. Your support and patience have been a cherished part of my life.

To Giovanni and Greta, I owe a debt of gratitude for your generosity and hospitality. Living rent-free in your apartment and not meeting my demise at your hands is a testament to your kindness and patience.

Panfilo, Reekee, Virginia, Elena, Denise, and the whole Thursday Pizza group, you have been true friends. The moments we've shared have been some of the most beautiful in these years of life in Trieste. Your friendship

has been a source of joy and a cherished escape.

Jack and Cerchi, despite the distances that separate us, the memories we share are a testament to a friendship that distance cannot dim. You remain some of my closest friends, and I am grateful for the bond that we have maintained over the years.

To Richard, and my friends from Sardinia - Raffaele, Diego, and Patrizio - despite the physical distances between us, our friendship has withstood the test of time. The moments we manage to come together and catch up are highlights that I treasure. Thank you for being an enduring part of my life.

To all of you, thank you from the bottom of my heart. This thesis is not just a reflection of my work but a testament to the incredible support I have been blessed with.

BIBLIOGRAPHY

[1] Henrik Aanæs et al. 'Large-Scale Data for Multiple-View Stereopsis'. In: *International Journal of Computer Vision* 120 (Nov. 2016). DOI: 10.1007/s11263-016-0902-9.

[2] Sameer Agarwal et al. 'Building Rome in a day'. In: *2009 IEEE 12th International Conference on Computer Vision*. 2009, pp. 72–79. DOI: 10.1109/ICCV.2009.5459148.

[3] Vassileios Balntas et al. 'Learning local feature descriptors with triplets and shallow convolutional neural networks'. In: Jan. 2016, pp. 119.1–119.11. DOI: 10.5244/C.30.119.

[4] Connelly Barnes et al. 'PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing'. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 28.3 (Aug. 2009).

[5] Andrei Bursuc, Giorgos Tolias and Hervé Jégou. 'Kernel Local Descriptors with Implicit Rotation Matching'. In: *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. ICMR '15. Shanghai, China: Association for Computing Machinery, 2015, pp. 595–598. ISBN: 9781450332743. DOI: 10.1145/2671188.2749379. URL: https://doi.org/10.1145/2671188.2749379.

[6] Chenjie Cao, Xinlin Ren and Yanwei Fu. *MVSFormer: Multi-View Stereo by Learning Robust Image Features and Temperature-based Depth*. 2022. arXiv: 2208.02541 [cs.CV].

[7] Chenjie Cao, Xinlin Ren and Yanwei Fu. *MVSFormer++: Revealing the Devil in Transformer's Details for Multi-View Stereo*. 2024. arXiv: 2401.11673 [cs.CV].

[8] Mathilde Caron et al. *Emerging Properties in Self-Supervised Vision Transformers*. 2021. arXiv: 2104.14294 [cs.CV].

[9] Mathilde Caron et al. *Emerging Properties in Self-Supervised Vision Transformers*. 2021. arXiv: 2104.14294 [cs.CV].

[10] Di Chang et al. *RC-MVSNet: Unsupervised Multi-View Stereo with Neural Rendering*. 2022. arXiv: 2203.03949 [cs.CV].

[11] Shuo Cheng et al. *Deep Stereo using Adaptive Thin Volume Representation with Uncertainty Awareness*. 2020. arXiv: 1911.12012 [cs.CV].

[12] Xiangxiang Chu et al. *Twins: Revisiting the Design of Spatial Attention in Vision Transformers*. 2021. arXiv: 2104.13840 [cs.CV].

[13] David Crandall et al. 'Discrete-continuous optimization for large-scale structure from motion'. In: *CVPR 2011*. 2011, pp. 3001–3008. DOI: 10.1109/CVPR.2011.5995626.

[14] Jifeng Dai et al. *Deformable Convolutional Networks*. 2017. arXiv: 1703.06211 [cs.CV].

[15] Jingming Dong and Stefano Soatto. 'Domain-Size Pooling in Local Descriptors: DSP-SIFT'. In: *CoRR* abs/1412.8556 (2014). arXiv: 1412.8556. URL: http://arxiv.org/abs/1412.8556.

[16] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV].

[17] O.D. Faugeras and M. Hebert. 'The Representation, Recognition, and Locating of 3-D Objects'. In: *The International Journal of Robotics Research* 5.3 (1986), pp. 27–52. DOI: 10.1177/027836498600500302. eprint: https://doi.org/10.1177/027836498600500302. URL: https://doi.org/10.1177/027836498600500302.

[18] Olivier Faugeras. 'Three-dimensional computer vision: a geometric viewpoint'. In: *MIT press* (Jan. 1993).

[19] Andrea Fusiello, Emanuele Trucco and Alessandro Verri. 'A Compact Algorithm for Rectification of Stereo Pairs'. In: 12 (Oct. 2000). DOI: 10.1007/s001380050120.

[20] Silvano Galliani, Katrin Lasinger and Konrad Schindler. 'Massively Parallel Multiview Stereopsis by Surface Normal Diffusion'. en. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. Piscataway, NJ: IEEE, 2015, pp. 873–881. ISBN: 978-1-4673-8390-5. DOI: 10.1109/ICCV.2015.106.

[21] A Geiger et al. 'Vision meets robotics: The KITTI dataset'. In: *The International Journal of Robotics Research* 32.11 (Aug. 2013), pp. 1231–1237. DOI: 10.1177/0278364913491297. URL: https://doi.org/10.1177%2F0278364913491297.

[22] Riccardo Gherardi, Michela Farenzena and Andrea Fusiello. 'Improving the efficiency of hierarchical structure-and-motion'. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2010, pp. 1594–1600. DOI: 10.1109/CVPR.2010.5539782.

[23] Xiaodong Gu et al. *Cascade Cost Volume for High-Resolution Multi-View Stereo and Stereo Matching*. 2020. arXiv: 1912.06378 [cs.CV].

[24] Meng-Hao Guo et al. 'PCT: Point cloud transformer'. In: *Computational Visual Media* 7.2 (Apr. 2021), pp. 187–199. ISSN: 2096-0662. DOI: 10.1007/s41095-021-0229-5. URL: http://dx.doi.org/10.1007/s41095-021-0229-5.

[25] Zellig S. Harris. 'Distributional Structure'. In: *Papers on Syntax*. Ed. by Henry Hiż. Dordrecht: Springer Netherlands, 1981, pp. 3–22. ISBN: 978-94-009-8467-7. DOI: 10.1007/978-94-009-8467-7_1. URL: https://doi.org/10.1007/978-94-009-8467-7_1.

[26] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. 2nd ed. New York, NY, USA: Cambridge University Press, 2003. ISBN: 0521540518.

[27] Zhaoyang Huang et al. *FlowFormer: A Transformer Architecture for Optical Flow*. 2022. arXiv: 2203.16194 [cs.CV].

[28] Sunghoon Im et al. *DPSNet: End-to-end Deep Plane Sweep Stereo*. 2019. arXiv: 1905.00538 [cs.CV].

[29] He Kaiming et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[30] Michael Kazhdan and Hugues Hoppe. 'Screened poisson surface reconstruction'. In: *ACM Trans. Graph.* 32.3 (2013). ISSN: 0730-0301. DOI: 10.1145/2487228.2487237. URL: https://doi.org/10.1145/2487228.2487237.

[31] Arno Knapitsch et al. 'Tanks and temples: benchmarking large-scale scene reconstruction'. In: *ACM Trans. Graph.* 36.4 (2017). ISSN: 0730-0301. DOI: 10.1145/3072959.3073599. URL: https://doi.org/10.1145/3072959.3073599.

[32] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton. 'ImageNet Classification with Deep Convolutional Neural Networks'. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[33] Moo Yi Kwang et al. *LIFT: Learned Invariant Feature Transform*. 2016. arXiv: 1603.09114 [cs.CV].

[34] Y. LeCun et al. 'Backpropagation Applied to Handwritten Zip Code Recognition'. In: *Neural Computation* 1 (1989), pp. 541–551.

[35] Yanghao Li et al. *Exploring Plain Vision Transformer Backbones for Object Detection*. 2022. arXiv: 2203.16527 [cs.CV].

[36] Tsung-Yi Lin et al. 'Feature Pyramid Networks for Object Detection'. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 936–944. DOI: 10.1109/CVPR.2017.106.

[37] D.G. Lowe. 'Object recognition from local scale-invariant features'. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2. 1999, 1150–1157 vol.2. DOI: 10.1109/ICCV.1999.790410.

[38] Claudia Malzer and Marcus Baum. 'A Hybrid Approach To Hierarchical Density-based Cluster Selection'. In: *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, Sept. 2020. DOI: 10.1109/mfi49285.2020.9235263. URL: http://dx.doi.org/10.1109/MFI49285.2020.9235263.

[39] J.C. McGlone et al. *Manual of photogrammetry / editor, J. Chris McGlone ; associate editors, Edward M. Mikhail, James Bethel ; technical editor, Roy Mullen.* eng. 5th ed. American Society for Photogrammetry and Remote Sensing, 2004. ISBN: 1570830711.

[40] Moritz Menze and Andreas Geiger. 'Object scene flow for autonomous vehicles'. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3061–3070. DOI: 10.1109/CVPR.2015.7298925.

[41] Zhenxing Mi, Di Chang and Dan Xu. 'Generalized Binary Search Network for Highly-Efficient Multi-View Stereo'. In: *CVPR*. 2022.

[42] D. Nister and H. Stewenius. 'Scalable Recognition with a Vocabulary Tree'. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. 2006, pp. 2161–2168. DOI: 10.1109/CVPR.2006.264.

[43] Frank C Park and Bryan J Martin. 'Robot sensor calibration: solving AX= XB on the Euclidean group'. In: *IEEE Transactions on Robotics and Automation* 10.5 (1994), pp. 717–721.

[44] American Society of Photogrammetry et al. *Manual of Photogrammetry*. American Society of Photogrammetry. American Society of Photogrammetry, 1980. ISBN: 9780937294017. URL: https://books.google.it/books?id=1MoYAQAAIAAJ.

[45] Shaohua Qi et al. 'Review of multi-view 3D object recognition methods based on deep learning'. In: *Displays* 69 (2021), p. 102053. ISSN: 0141-9382. DOI: https://doi.org/10.1016/j.displa.2021.102053. URL: https://www.sciencedirect.com/science/article/pii/S0141938221000639.

[46] Filip Radenović, Giorgos Tolias and Ondrej Chum. 'CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples'. In: vol. 9905. Oct. 2016, pp. 3–20. ISBN: 978-3-319-46447-3. DOI: 10.1007/978-3-319-46448-0_1.

[47] Nikhila Ravi et al. *Accelerating 3D Deep Learning with PyTorch3D*. 2020. arXiv: 2007.08501 [cs.CV].

[48] D. Scharstein, R. Szeliski and R. Zabih. 'A taxonomy and evaluation of dense two-frame stereo correspondence algorithms'. In: *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001)*. 2001, pp. 131–140. DOI: 10.1109/SMBV.2001.988771.

[49] Johannes L. Schönberger. 'Robust Methods for Accurate and Efficient 3D Modeling from Unstructured Imagery'. en. Doctoral Thesis. Zurich: ETH Zurich, 2018. DOI: 10.3929/ethz-b-000295763.

[50] Johannes Lutz Schönberger and Jan-Michael Frahm. 'Structure-from-Motion Revisited'. In: (2016).

[51] Thomas Schöps, Torsten Sattler and Marc Pollefeys. 'BAD SLAM: Bundle Adjusted Direct RGB-D SLAM'. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[52] Thomas Schöps et al. 'A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos'. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[53] Elisavet Stathopoulou and Fabio Remondino. 'A survey on conventional and learning-based methods for multi-view stereo'. In: *The Photogrammetric Record* 38 (Aug. 2023). DOI: 10.1111/phor.12456.

[54] Ashish Vaswani et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL].

[55] P. Viola and W.M. Wells. 'Alignment by maximization of mutual information'. In: *Proceedings of IEEE International Conference on Computer Vision*. 1995, pp. 16–23. DOI: 10.1109/ICCV.1995.466930.

[56] Chen Wang et al. 'Self-Supervised deep homography estimation with invertibility constraints'. In: *Pattern Recognition Letters* 128 (2019), pp. 355–360. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2019.09.021. URL: https://www.sciencedirect.com/science/article/pii/S0167865519302673.

[57] Chen Wang et al. 'Self-Supervised Multiscale Adversarial Regression Network for Stereo Disparity Estimation'. In: *IEEE Transactions on Cybernetics* 51.10 (2021), pp. 4770–4783. DOI: 10.1109/TCYB.2020.2999492.

[58] Fangjinhua Wang et al. *PatchmatchNet: Learned Multi-View Patchmatch Stereo*. 2020. arXiv: 2012.01411 [cs.CV].

[59] Xiang Wang et al. 'Multi-view stereo in the Deep Learning Era: A Comprehensive Review'. In: *Displays* 70 (Oct. 2021), p. 102102. DOI: 10.1016/j.displa.2021.102102.

[60] Zizhuang Wei et al. *AA-RMVSNet: Adaptive Aggregation Recurrent Multi-view Stereo Network*. 2021. arXiv: 2108.03824 [cs.CV].

[61] Yao Yao et al. 'MVSNet: Depth Inference for Unstructured Multi-view Stereo'. In: *arXiv e-prints*, arXiv:1804.02505 (Apr. 2018), arXiv:1804.02505. DOI: 10.48550/arXiv.1804.02505. arXiv: 1804.02505 [cs.CV].

[62] Hongwei Yi et al. *Pyramid Multi-view Stereo Net with Self-adaptive View Aggregation*. 2020. arXiv: 1912.03001 [cs.CV].

[63] Yisu Zhang, Jianke Zhu and Lixiang Lin. *Multi-View Stereo Representation Revisit: Region-Aware MVSNet*. 2023. arXiv: 2304.13614 [cs.CV].

[64] Sixiao Zheng et al. *Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers*. 2021. arXiv: 2012.15840 [cs.CV].

[65] Jie Zhu et al. *Multi-View Stereo with Transformer*. 2021. arXiv: 2112.00336 [cs.CV].

COLOPHON

Comparative Evaluation of Multi-View Stereo 3D Reconstruction Techniques on Specialized Dataset,

Data Science and Scientific Computing, Università degli Studi di Trieste

Final Version as of 24th March 2024 (.