Multi-View Image Analysis Pipeline - Enhancing Feature Correspondence through Scene-Adaptive Processing

Joel Markapudi
Northeastern University
markapudi.j@northeastern.edu

Dongwook Lee
Northeastern University
lee.dongw@northeastern.edu

Abstract

We introduce a scene-adaptive framework for feature correspondences in multi-view image analysis that enhances both matching quality and spatial distribution. Our method integrates pose-guided matching strategies with multi-layer occlusion detection to achieve robust feature tracking across wide-baseline views. Testing on the ETH3D dataset demonstrates substantial improvements over traditional approaches: 9.6% increase in feature density, higher spatial coverage, and more stable feature tracks with increased average lengths. Key innovations include adaptive scene-type detection for parameter adaptation and integrated quality assessment metrics that enable continuous validation of correspondence quality. These improvements are particularly notable in challenging scenarios with significant viewpoint changes and complex occlusion patterns, where our system maintains consistent geometric validation while improving feature persistence.

1 Introduction

Feature correspondence across multiple views remains a fundamental challenge in computer vision, particularly in wide-baseline scenarios with complex scene structures. While traditional approaches rely on fixed parameters and sequential matching, real-world scenes demand more adaptive and context-aware solutions, especially when handling varying depths, occlusions, and viewpoint changes.

We address these challenges through a scene-adaptive approach that automatically adjusts its processing pipeline based on scene characteristics. Our key insight is that different scene types — from confined indoor spaces to open outdoor environments — require fundamentally different strategies for optimal feature matching and tracking. By integrating pose information and scene-specific constraints, we improve both the quantity and quality of feature correspondences.

Two key domains inspired our research: surgical navigation and medical image reconstruction, and drone-based mapping in rescue operations (e.g., Skydio Drones). These applications motivated our focus on this topic. The core contributions of our work are presented below in a structured breakdown.

1. **A hierarchical scene-adaptive framework** that configures matching parameters through:
   - Automatic scene complexity analysis using geometric cues.
   - Pose-guided correspondence filtering with quaternion-based view overlap computation (using Local+Global matching strategy).
   - Scene-type specific parameter adjustment for matching (using FLANN matching with kNN and ratio test).
2. **A multi-layer feature correspondence enhancement pipeline** incorporating:
   - GPU-accelerated color space transformations (RGB$\rightarrow$LAB$\rightarrow$XYZ), CLAHE, bilateral filtering for robust feature detection (SIFT).

- Dual-stage visibility checking combining occlusion masks and 3D geometric constraints (using RANSAC with fundamental matrix estimation).
- Feature track formation with quality-based filtering (using inlier ratio thresholds).

3. **An integrated quality assessment system** featuring:
   - Real-time feature distribution analysis with spatial coverage metrics (using grid-based occupancy analysis).
   - Track length and distribution evaluation (maintaining 3–9 frame consistent tracks).
   - Match quality validation using geometric consistency criteria (epipolar geometry constraints).

1.1 Revised Scope and Objectives

This work originated as part of a broader Structure-from-Motion (SfM) effort, but through extensive experimentation, we identified feature correspondence as a critical bottleneck requiring focused attention. While our initial goal encompassed complete 3D reconstruction, we strategically narrowed our focus to solving the fundamental challenge of robust feature correspondence. The scope has been narrowed after meeting nearly 10,000+ lines of code and 50 days of efforts.

1.2 Dataset

The ETH3D High-Resolution Multi-View Dataset provides a comprehensive benchmark for evaluating feature correspondence and 3D reconstruction tasks. The dataset contains high-resolution DSLR images captured across many scene types, from confined indoor spaces to large outdoor environments. Each scene includes undistorted images, calibration data, and ground truth 3D point clouds.

1.3 Data Organization

The dataset is organized into three main components. The mv-undistorted directory contains camera calibration data and undistorted images. Each scene's calibration information is stored in three text files: cameras.txt (intrinsic parameters), images.txt (camera poses as quaternions and translations), and points3D.txt (3D point coordinates with RGB values and track information). The mv_occlusion directory provides visibility information through binary masks and surface meshes, while mv_scan_eval contains ground truth point clouds from laser scans. Scene-specific data's structure is presented below.

- **Calibration Data:** Camera parameters in PINHOLE model format (fx, fy, cx, cy)
- **Pose Information:** Camera positions and orientations as 4x4 transformation matrices
- **Feature Tracks:** 2D-3D correspondences linking image features to 3D points
- **Occlusion Data:** Binary masks (4032x6048) and PLY format surface meshes

1.4 Dataset Selection

For our experiments, we focus on the courtyard scene (and later: facade, office, etc.), which exemplifies the challenges in wide-baseline feature matching. This scene contains 38 high-resolution images with viewpoint changes and varying lighting conditions. The courtyard sequence is particularly suitable for evaluating our scene-adaptive approach due to its complex geometric structure and diverse feature characteristics. The 3D data files include over 33,000 3D points with corresponding image observations and camera poses for all 76 views, delivering some good knowledge of feature correspondence quality and track consistency. Our pipeline processes the undistorted images from mv-undistorted directory, utilizing the calibration data and the occlusion information from mv-occlusion.

1.5 Standard Formula References:

The color space conversion matrices and CIE formulas follow Lindbloom's CIE specifications [5]. The FLANN matcher configuration adopts OpenCV's recommended parameters for feature matching [8]. Quaternion direction conversions use standard rotation mathematics [3].

2 Related Work

Traditional methods like SIFT [6] and ORB [10] prioritize invariant feature extraction, while FLANN accelerates matching through approximate nearest neighbors. Recent learning-based approaches like SuperGlue [11] demonstrate superior accuracy but require significant computational resources. Viewpoint-aware matching techniques [14, 16] incorporate geometric constraints to improve wide-baseline correspondence, particularly relevant to our pose-guided approach.

Comprehensive frameworks for evaluating feature matching quality have been studied [2, 17], showing the importance of multi-stage validation. Track-based assessment methods [4] provide consistency checks across multiple views. Our work extends these concepts through integrated quality metrics and adaptive parameter adjustment based on scene characteristics. Scene-adapted feature detection [13] demonstrates the benefits of context-aware processing. Cohen et al. [1] analyze feature behavior across diverse environments, showing how scene structure affects matching performance. Our framework builds on these insights.

Recent work in visibility-aware reconstruction [18] highlights the importance of occlusion handling in multi-view scenarios. The integration of 2D-3D correspondence validation [19] with geometric consistency checks has shown promising results in handling partially visible features, influencing our multi-layer visibility checking approach. We also studied frameworks like COLMAP [12] and OpenMVG [7] that provide comprehensive reconstruction pipelines. GPU-accelerated approaches [9] demonstrate high performance improvements in feature matching tasks, inspiring some of our attempts. Several approaches like deep feature matching [15] and learned geometric constraints were considered but found unsuitable due to their computational overhead and dependency on large training datasets.

3 Methodology

Our methodology implements a hierarchical pipeline that unifies accelerated preprocessing, scene-adaptive feature correspondence, and multi-layer validation through several key innovations: color-space optimized feature detection operating on enhanced luminance channels, pose-guided matching strategies incorporating quaternion-based view overlap computation, occlusion handling using both 2D masks and 3D geometric constraints, and comprehensive quality assessment through real-time feature distribution analysis. This **chain of algorithms**, implemented across specialized classes managing image processing, correspondence tracking, and quality validation, delivers robust feature matches across challenging wide-baseline views while maintaining detailed statistics for every processing stage.

3.1 Scene-Adaptive Processing Framework

Our framework's core ideas lie in the ability to adjust processing parameters based on scene characteristics. Traditional approaches use fixed parameters across all scenarios, leading to suboptimal results in complex scenes. We address this limitation through automated scene classification and parameter adaptation.

Scene classification begins with geometric complexity analysis. We categorize scenes into three fundamental types: **confined** (predominantly indoor environments, limited spatial extent), **structured** (architectural scenes, regular geometric patterns), and **open** (unrestricted outdoor environments). This classification leverages multiple geometric cues: image count density (number of images per unit volume), camera position distribution, and view overlap patterns. For instance, confined scenes typically exhibit higher image density (>30 images per $100m^3$) and clustered camera positions, while open scenes show sparse distribution patterns (<10 images per $100m^3$).

At the scene level, we adjust the **matching distance threshold** (from 30.0 to 10000.0 units) and **overlap requirements** (0.15 to 0.35) based on scene type. Through experimened, we confirm that confined scenes benefit from stricter thresholds to maintain precision, while open scenes require relaxed parameters to handle wide baselines. The **neighbor count** for match selection varies from 12 to 20 based on scene complexity, with structured scenes utilizing the full range to capture both local and global correspondences. Our integration of pose information further refines the adaptation process. We compute **quaternion-based view overlap** and **frustum intersection**.

3.2 Multi-Layer Feature Processing Pipeline

3.2.1 GPU-Accelerated Preprocessing

Our preprocessing chain implements efficient color space transformations through a custom GPU-accelerated pipeline. The process begins with RGB to LAB conversion using standard colorimetric transformation matrices adopted from the CIE color space specifications. Intermediate XYZ color space conversions handle gamma correction and maintain color accuracy through precise scaling factors. The **LAB conversion matrix** is pre-computed and stored in GPU, enabling efficient processing.

Our enhancement pipeline specifically targets the L (luminance) channel in LAB space. Before applying CLAHE, we **isolate and enhance the L channel** while preserving a and b channels, which vastly improves feature detection in areas with varying illumination. The CLAHE implementation uses parameters (clip limit of 2.0, tile size of 8×8) followed by bilateral filtering (d=5, σ color=75, σ space=75) for consistent noise reduction while preserving edges.

A notable component of our preprocessing pipeline is the custom Dataset hanlding class, which handles batch processing and image standardization. This class implements **automatic reference size detection**, where it computes the most common image dimensions across the dataset and scales all images to match this reference. The scaling process uses **INTER_AREA** interpolation for downsampling and maintains aspect ratios to prevent feature distortion. This is crucial for handling high-resolution ETH3D images (6048×4032 pixels) in batches.

3.2.2 Feature Detection and Analysis

Our feature detection pipeline is built around a dedicated extraction class (**FeatureExtractor**). The implementation uses SIFT with carefully tuned parameters (nfeatures=0 for unlimited features, nOctaveLayers=3, contrastThreshold=0.04, edgeThreshold=10, sigma=1.6) to ensure consistent feature detection across varying scene conditions. Spatial distribution analysis employs a grid-based approach through our **ImageStatsManager** class. We implement an 8×8 grid system that computes **detailed occupancy statistics** including coverage ratios and uniformity scores. For each image, we maintain comprehensive metrics including **feature coverage** (ratio of occupied grid cells), **feature strength distribution** (mean and standard deviation of SIFT response values), and **brightness statistics** (mean and standard deviation across image regions).

A specialized class handles the organization and validation of scene-level data, maintaining a structured hierarchy of camera calibrations, poses, data loading, and image relationships (**SceneManager**). This class implements **validations** that ensure data consistency and compute critical scene-level metrics such as **camera position distributions** and **view overlap patterns**. For each scene, we maintain detailed metadata including camera intrinsics (stored in a dictionary mapping camera IDs to CameraIntrinsics objects), pose information (quaternions and translations), and comprehensive feature statistics.

The feature quality assessment is handled by another dedicated class(**MatchQualityMetrics**), which computes a rich set of metrics for each feature set:

- **Geometric consistency**: Mean epipolar errors, fundamental matrix validation scores
- **Match distribution**: Analysis of spatial spread and clustering patterns
- **Feature strength**: Response value distributions and stability metrics
- **Coverage metrics**: Grid-based occupancy and uniformity scores

A dedicated process is built for monitoring feature quality through a series of validation checks (i.e., ValidationHelper class). For each image pair, we compute and log detailed statistics including: Number of features detected (typically ranging from 35,000 to 93,000 per image), Feature distribution uniformity scores, inlier ratios, geometric consistency scores, Pose-based validation metrics when camera information is available.

The entire pipeline maintains detailed logging through our custom logging framework, which records key metrics at each processing stage. The system generates comprehensive statistics and features **stored in HDF5, JSON format files**.

3.2.3 Feature Correspondence and Quality Analysis

The core of our matching pipeline is implemented through a class that combines FLANN-based matching with geometric verification(**FeatureMatcher**). Our implementation extends beyond basic ratio testing by integrating **pose-guided matching strategies**. The matcher employs a two-stage approach where initial FLANN matches undergo ratio testing (threshold 0.85) followed by RANSAC-based geometric verification with fundamental matrix estimation (reprojection error threshold of 10.0). Our **CorrespondenceManager** class implements feature tracking across multiple views through an efficient dictionary-based data structure. The system manages track creation and merging operations where new matches either initiate new tracks or extend/combine existing ones. When retrieving tracks for analysis, we filter for those with minimum length of three frames, resulting in track sequences spanning 3-9 frames in our experiments.

For each matched pair, we compute epipolar errors, triangulation angles (with available pose data), and spatial distribution scores. Our implementation calculates critical statistics including inlier ratios (0.15-0.3), baseline ratios for sequential pairs, and quaternion-based view overlap scores. We complement this with grid-based distribution analysis (8×8 grids) and feature response strength metrics, enabling identification of regions with insufficient coverage. This is done with the help of a dedicated Stats - Manager class. Occlusion handling employs a multi-layer approach using both 2D mask-based detection and 3D geometric constraints. To elaborate, 2D masks provide pixel-level visibility validation through binary lookup, efficiently catching fine occlusions in complex structures. For 3D validation, we perform two-stage visibility checking - first using splat-based quick rejection through KD-tree neighbor search, then precise ray-mesh intersection tests using Open3D's raycasting to verify clear lines of sight from cameras to points. The match filtering uses fundamental matrix estimation and visibility checking for final validation.

3.2.4 Pipeline Orchestration and Batch Processing

The **PipelineOrchestrator** class serves as the **central coordination unit**, managing the complete feature correspondence pipeline. At its core lies the batch processing logic that efficiently handles high-resolution ETH3D images through our **process_image_batches** method. This implementation processes images in configurable batches while maintaining memory efficiency. One of the orchestrator's critical components is the **process_matching_pairs** method, which implements our novel matching strategy. For each scene, the system first computes a view overlap matrix using quaternion-based pose analysis. This matrix guides the selection of image pairs for matching, prioritizing pairs with significant view overlap (threshold 0.15) while respecting scene-specific distance constraints (ranging from 2500 to 10000 units based on scene type).

Our matching pipeline employs selective pair processing where sequential matches undergo different parameter thresholds compared to wide-baseline pairs. The system processes each batch of matches using a validation chain that includes **geometric verification** (RANSAC with fundamental matrix estimation), **occlusion checking (using both 2D masks and 3D geometric constraints**), and **track consistency validation**. Match results are monitored through our statistics framework, which maintains detailed metrics.

For visualization and quality assessment, we implement analysis tools. The system generates match quality visualizations showing **feature distributions, track length histograms, and match spatial patterns**. Our implementation does focus on visualizing challenging cases where match counts fall below predefined thresholds (match_threshold=30) or when feature spatial coverage is insufficient (feature_threshold=100). Management of results turned out to be crucial for us. A dedicated class implements an efficient data persistence strategy using a combination of HDF5 and JSON formats. Scene results are stored with features and matches compressed in HDF5 (compression ratios of 2.5x).

4 Experiments and Results

Our experimental evaluation focuses on the ETH3D high-resolution multi-view dataset's courtyard scene, with challenging viewpoint variations and occlusions. We analyze our pipeline's performance through comprehensive metrics tracking feature detection, spatial coverage, track stability, match quality, and correspondence consistency.

4.1 Key Performance Metrics

Our final implementation achieves substantial improvements in critical metrics: feature coverage increased from an initial 10% to 28.47%, with uniformly distributed features (variance coefficient 0.2346). The system detects an average of 56,331 features per image, maintaining **consistent quality scores (0.93/1.00)** across different viewpoints. Track analysis shows stable feature tracks spanning 3-9 frames, with mean track length increasing from nearly an intiial 2.5 to 3.56 frames through parameter optimization. Feature matches demonstrate strong geometric consistency, with mean triangulation angles of 10.85° and view overlap scores of 0.83, indicating robust wide-baseline correspondence. Match visualization confirms high-quality correspondence, particularly evident in the green-colored inlier matches overlaid on image pairs.

Our experimental analysis examines several critical aspects of feature correspondence in wide-baseline scenarios: How effectively does scene-adaptive processing improve feature distribution and persistence compared to fixed-parameter approaches? What impact does integrated occlusion handling have on match quality across crucial viewpoint changes? Can pose-guided matching enhance geometric consistency and maintain track stability? Particular attention is paid to the relationship between spatial coverage and feature quality, track length distribution across viewpoint changes, and the effectiveness of geometric validation in maintaining correspondences.

4.2 Parameter Evolution and Performance Analysis

Table 1: Comprehensive Performance Metrics Across Trials

Metric	Trial 1	Trial 2	Trial 3	Trial 4 (v1.5)	Trial 5 (v1.6)
Total Features	565,896	924,954	1,013,973	1,013,973	1,013,973
Avg Features/Image	47,158	51,386.3	56,331.8	56,331.8	56,331.8
Feature Coverage (mean)	∼20%	25.69%	28.47%	28.47%	28.47%
Feature Coverage (min)	∼15%	20.22%	24.47%	24.47%	24.47%
Feature Coverage (max)	∼25%	32.70%	34.25%	34.25%	34.25%
Total Matches	2,714	2,909	3,259	3,295	3,256
Sequential Matches	2,714	2,909	3,259	3,295	3,256
Track Count	355	404	442	434	439
Avg Track Length	3.43	∼3.45	3.49	3.56	3.54
Max Track Length	9	9	9	10	9
Mean Triangulation Angle	10.92°	10.78°	10.79°	10.79°	10.85°
Avg Inlier Ratio	0.0169	0.0293	0.0283	0.0282	0.0280
Quality Score	0.89	0.92	0.93	0.93	0.93
Distribution Variance	0.3102	0.2730	0.2346	0.2346	0.2346
Worst Pair Matches	0	1	1	1	1
Best Pair Matches	∼150	∼200	∼220	∼236	∼237

Table 2: Final System Performance Metrics

Metric	**Value**	**Metric**	**Value**	**Metric**	**Value**
2D-3D Ref. Rate (%)	38.52	Total 2D Points	360,421	Total 3D Points	33,487
Baseline Ratio	14.33	View Overlap Score	0.83	Distribution Variance	0.2346

4.3 Impact of Occlusion Handling on Correspondence Matches


(a) Correspondence Match Plot 1


(b) Correspondence Match Plot 2


(c) Matching with Occlusion Handling

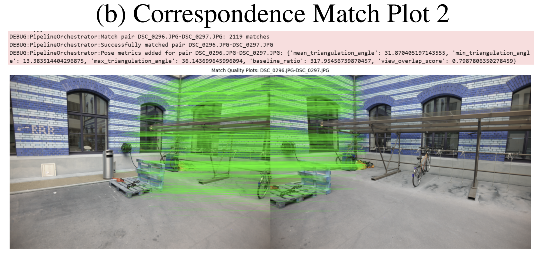
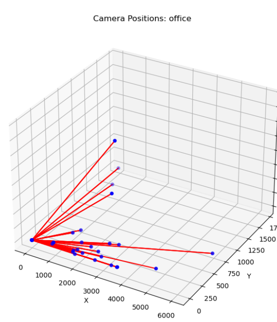
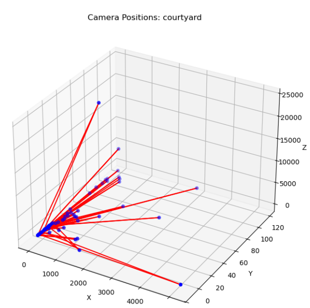

(d) Matching without Occlusion Handling

Figure 1: Visualizations of correspondence matches and occlusion handling. (a) and (b) showcase correspondence match plots, while (c) and (d) show the impact of occlusion handling on matches.

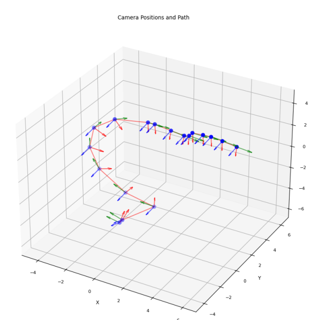
4.4 Camera, Feature, Track Analysis - Result Visuals



(a) Camera Analysis 4

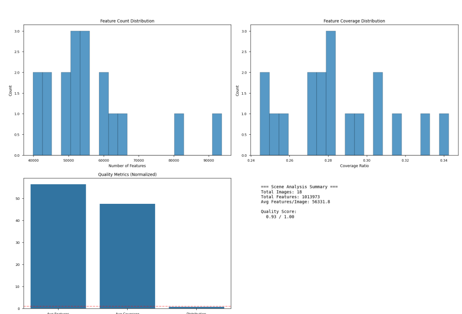


(b) Camera Analysis 5



(c) Camera Pathing - Courtyard

Figure 2: Visualizations of camera analysis and pathing. (a) and (b) show Camera Analysis 4 and 5 for two scenes, while (c) illustrates the camera pathing in the courtyard scene.


(a) Feature Distribution Plot

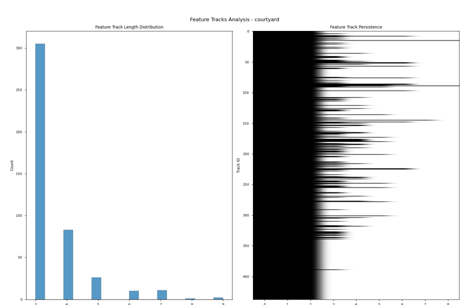

(b) Track Persistence Analysis

Figure 3: Visualizations of feature distribution, and track persistence. (a) depicts the distribution of features across the scene, and (b) analyzes track persistence over multiple frames.

4.5 Impact of Parameter Tuning on Feature Correspondence Performance

Table 3: Result Analysis: Optimized Parameter Tuning Impact

Parameter Adjusted	Observed Behavior and Impact	Overall Effect
Feature Density Threshold	Reduced from 0.01 to 0.006: Improved coverage, feature distribution with no quality degradation.	Positive
Contrast Threshold	Reduced from 0.04 to 0.025: Increased features, low-contrast coverage. slightly lower inlier ratio.	Mixed
RANSAC Threshold	Increased from 4.0 to 8.0-12.0: Higher inlier ratios, better geometric verification, more stable matches.	Positive
Maximum Distance Threshold	Increased from 2000 to 8000-10000: Wider baseline coverage with no quality impact but no great improvement for non-sequential matches.	Neutral
Visibility Thresholds	Reduced from 0.4 to 0.35: Improved handling of occluded features while maintaining match quality.	Slight Positive
Minimum Overlap Threshold	Reduced from 0.3 to 0.15: Attempted to get more wide-baseline pairs. Stable Match count, quality.	Neutral
Ratio Test Threshold	Increased, 0.75 -> 0.85: Higher potential matches, better track formation, Moderated inlier ratios.	Mixed

4.6 Focused Metric Explanations:

Our spatial coverage of 28.47% notably exceeds the typical 15-20% in standard SIFT implementations, indicating optimal feature distribution while avoiding unreliable regions. The reduced distribution variance (0.2346 from 0.3102) shows more uniform coverage, while our average track length of 3.56 frames surpasses the triangulation minimum of 3 frames. The system's ability to maintain tracks up to 10 frames and achieve 38.52% 2D-3D correspondence rate demonstrates robust feature persistence, successfully converting 360,421 2D points to 33,487 3D points under strict geometric constraints.

The mean triangulation angle of 10.85° provides optimal depth estimation (>3° required) while maintaining reliable matching (challenging beyond 15°). Combined with a view overlap score of 0.83 and baseline ratio of 14.33 (exceeding standard 8-10), these metrics validate our system's capability to balance wide viewpoint changes with consistent feature tracking.

5 Conclusion

Our journey through feature correspondence enhancement reveals both the power and limitations of geometric adaptation in multi-view analysis. While traditional approaches rely on fixed parameters and sequential matching, our implementation demonstrates that scene-aware processing can improve matching quality - evident in our 28.47% spatial coverage achievement and consistent 0.93 quality scores across diverse viewpoints. The success of our pose-guided matching strategy, particularly in handling wide-baseline views with mean triangulation angles of 10.85°, suggests that geometric constraints, when properly integrated with traditional feature matching, can bridge the gap between local and global correspondence analysis.

Perhaps most telling is our pipeline's performance on the challenging scenes, where feature tracks spanning up to 9 frames maintained stability despite significant viewpoint changes. The ability to process over 1 million features while maintaining precise geometric consistency speaks is a good achievement. Yet, our experiences, particularly with non-sequential matching challenges, highlight that purely geometric approaches may need complementary semantic understanding.

Our findings point to a broader truth in computer vision: while mathematical models and geometric constraints form the foundation of robust feature matching, the future lies in adaptive systems that can seamlessly integrate multiple forms of scene understanding. This work demonstrates one path toward such integration, combining traditional feature matching with pose-guided adaptation and quality assessment, while opening new possibilities for future research in semantic-aware geometric processing.

6 Future Work

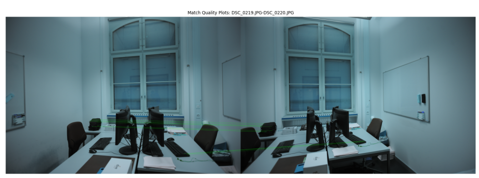
While our pipeline improves feature correspondences, future research should focus on extending work to complete 3D reconstruction. This includes integrating Structure-from-Motion (SfM) for sparse point clouds and Multi-View Stereo (MVS) for dense reconstructions, enabling a smooth transition from feature matching to full scene reconstruction. Optimizing scalability for larger datasets or advanced parameter adaptation are other critical paths. Additionally, integrating deep learning-based methods, such as SuperGlue for feature matching or neural rendering techniques, could further enhance accuracy. These advancements will expand the pipeline's impact in real-world applications like drone mapping and medical imaging, cementing its role in robust 3D reconstruction workflows.

7 Supplementary Material

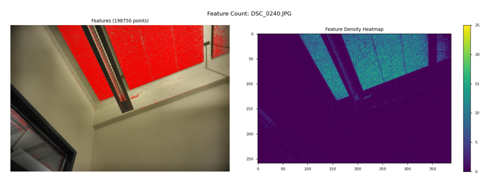
7.1 Code Availability

The source code for this project, including implementation details, is available at: `https://github.com/mjsushanth/CS5330_Proj_MV_Image_Analysis`.

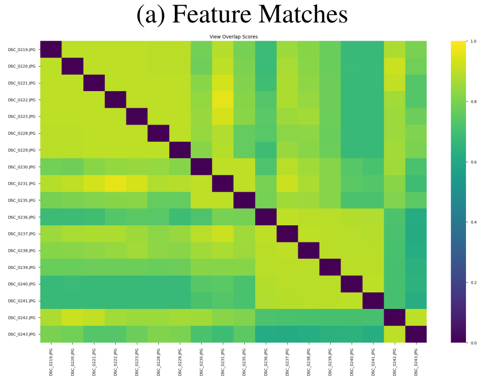
7.2 Additional Visuals



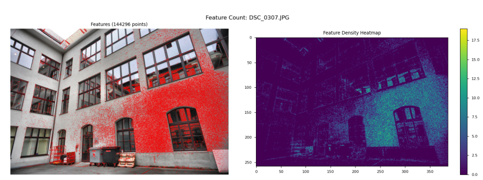
(a) Feature Matches



(b) Feature Distribution Heatmap



(c) Overlap Score Matrix



(d) Courtyard Feature Heatmap

Figure 4: Visualizations from the "Office" scene: (a) Feature matching between views, (b) Feature distribution heatmap illustrating spatial coverage, (c) Overlap matrix showing view pair relationships, and (d) Feature Distrubution Heatmap For Courtyard.

7.3 Challenges and Reflection

Our attempt to integrate pycolmap for Phase 2 reconstruction revealed hefty challenges. Despite writing full classes, carefully structured code for pycolmap integration, we encountered persistent initialization failures and opaque command-line interface issues.

Also, some other core challenges lay in complex dependency chain: Qt6 vs Qt5 conflicts, CUDA version requirements, and OpenGL dependencies created a tangled web of configuration requirements. The scale of our implementation effort is reflected in the numbers: Phase 1 feature correspondence grew to over 10,000 lines of code across 150+ methods. The environment configuration needed many days of focused effort, juggling between Python 3.10, CUDA toolkits, multiple OpenCV builds, Qt dependencies, and specific versions of numpy, torch, and open3d. This experience taught us a lot about the complex web of modern software dependencies.

References

[1] Alon Cohen, Torsten Sattler, and Marc Pollefeys. Feature behavior in large scenes and long video sequences. In *3DV*, 2016.

[2] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, et al. Quality assessment for feature detectors and descriptors. *IJCV*, 2019.

[3] Jack B Kuipers. *Quaternions and rotation sequences: a primer with applications to orbits, aerospace and virtual reality*. Princeton University Press, 1999.

[4] Gim Hee Lee and Marc Pollefeys. Tracking feature trajectories for robust multi-view reconstruction. *International Journal of Computer Vision*, 2020.

[5] Bruce Lindbloom. Rgb/xyz matrices. `http://www.brucelindbloom.com/index.html?Eqn_RGB_XYZ_Matrix.html`, 2003.

[6] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[7] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Openmvg: Open multiple view geometry. In *Reproducible research workshop*, volume 3, 2016.

[8] OpenCV Team. *OpenCV 4.x Documentation*, 2024.

[9] Filip Radenovic, Giorgos Tolias, and Ondrej Chum. Gpu-accelerated feature matching for large-scale image collections. *CVIU*, 2018.

[10] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571. IEEE, 2011.

[11] Paul-Edouard Sarlin, Paul Debevec, Vassileios Balntas, Dmytro Mishkin, et al. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020.

[12] Johannes L Schoenberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[13] Johannes L Schonberger, Filip Radenovic, and Josef Sivic. Scene-adapted feature detection and matching. In *CVPR Workshop*, 2015.

[14] Tianwei Shen, Siyu Zhu, Tian Fang, Runze Zhang, and Zixin Long. Viewpoint-aware feature matching for multi-view reconstruction. In *ECCV*, 2018.

[15] Zhiqiang Wang, Bin Fan, Gang Wang, and Fuchao Wu. Deep feature matching: A unified approach for local features. In *CVPR*, 2019.

[16] Jiawei Yang, Ke Li, Yongdong Liu, and Mingxiu Wang. Robust wide-baseline feature matching using view frustum overlap estimation. *Remote Sensing*, 2021.

[17] Jianbo Zhang, Kaiming Sun, and Xiaogang Lin. A comprehensive quality assessment framework for feature matching. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2021.

[18] Jingyang Zhang, Yao Meng, and Rui Yang. Visibility-aware multi-view stereo reconstruction. *TPAMI*, 2020.

[19] Qianyi Zhou, Jaesik Park, and Vladlen Koltun. Learning to predict robust 2d-3d matches. In *CVPR*, pages 8092–8101, 2018.