Ipek Akkus - Assignment #3 Report:
3D Scene Reconstruction via Smartphone Camera Triangulation

1. Introduction

This report presents a complete 3D reconstruction pipeline developed as part of a computer vision assignment using stereo images captured from a smartphone. The project applies multi-view geometry, feature detection, epipolar geometry, and triangulation to recover a sparse 3D point cloud of a real-world scene.

2. Methodology

2.1 Image Capture

Two high-resolution images of a cluttered desk were captured using a handheld smartphone camera from slightly different angles to ensure ~60% scene overlap. Care was taken to avoid motion blur and maintain consistent lighting. The scene contained textured elements (books, fan, plants), which support reliable feature detection. As image 3 does not have the top of the image, and image 4 includes a chair as well, I selected the image 1 and 2 as my image pair of interest.

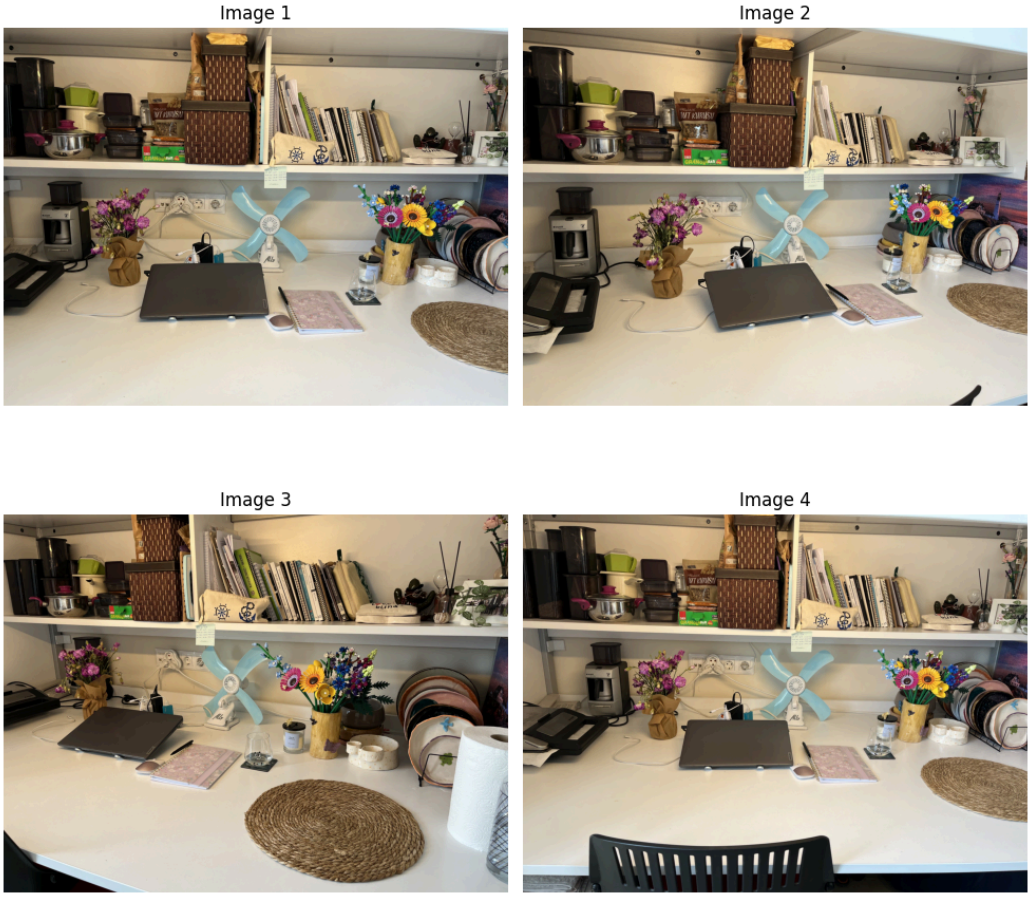

Figure 1. *Sample cluttered desk images*

2.2 Feature Detection and Matching

Keypoints were detected using two algorithms:

- SIFT (Scale-Invariant Feature Transform): robust to scale and rotation, provides high-quality descriptors.
- ORB (Oriented FAST and Rotated BRIEF): computationally efficient and suitable for real-time tasks.

Both detectors were used with descriptor-based matching:

- SIFT with a Brute-Force matcher and Lowe's ratio test
- ORB with Hamming distance and cross-check

Matched keypoints were visualized and filtered to remove weak or repetitive matches.

Figure 2a. *SIFT keypoint match visualizations*

Figure 2b. *ORB keypoint match visualizations*

2.3 Camera Calibration

Intrinsic parameters were estimated using OpenCV's checkerboard calibration process. A printed 9x6 checkerboard was captured from multiple angles and used to compute the camera matrix and distortion coefficients. One of them is shown as a sample in Figure 3.

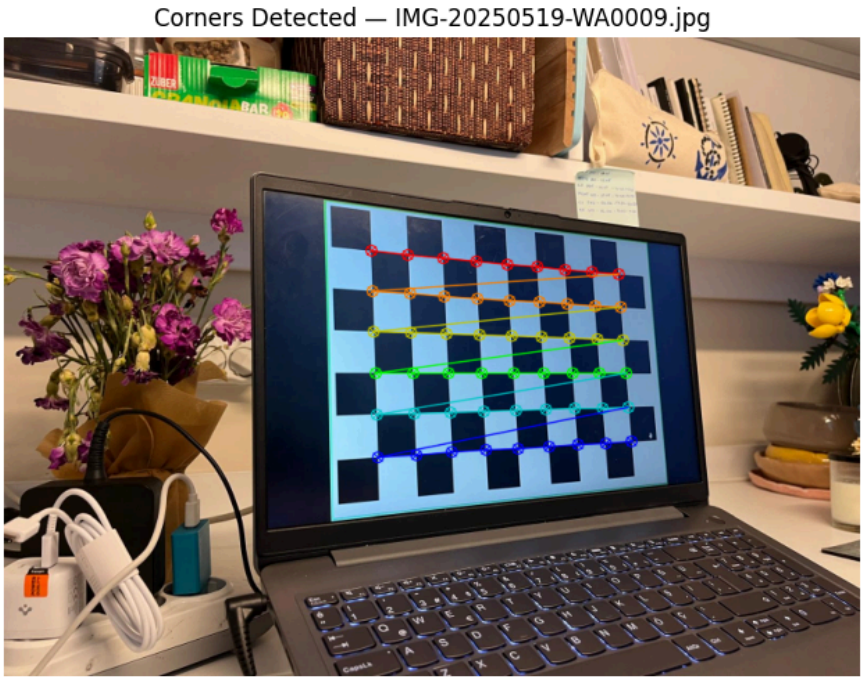

Figure 3. *Detected Corners of Checkerboard*

2.4 Fundamental Matrix Estimation

The fundamental matrix was estimated using the 8-point algorithm and RANSAC to handle outliers. This matrix defines the epipolar geometry between the image pair. Epipolar lines were plotted to validate the consistency of the recovered geometry.

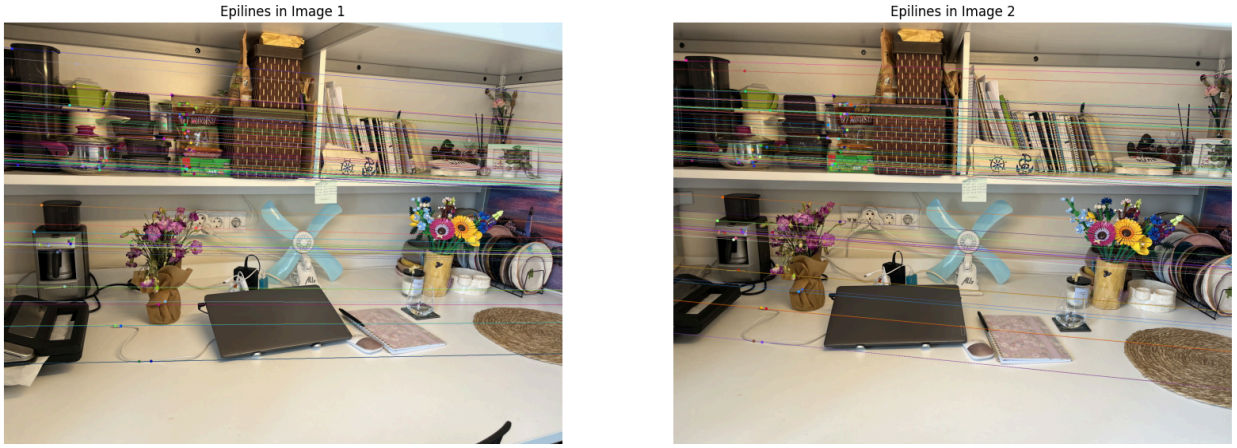

Figure 4. *Epipolar lines on both images*

2.5 Triangulation and 3D Reconstruction

Using the filtered inlier correspondences and camera matrices, 3D points were triangulated. The points were converted from homogeneous to Euclidean coordinates. A scale factor was applied to spread the reconstruction. Colors were assigned based on image intensity using the *plasma* colormap.

Outliers were removed using bounding box filtering. Camera poses were also visualized for spatial context, which are seem to be correct.

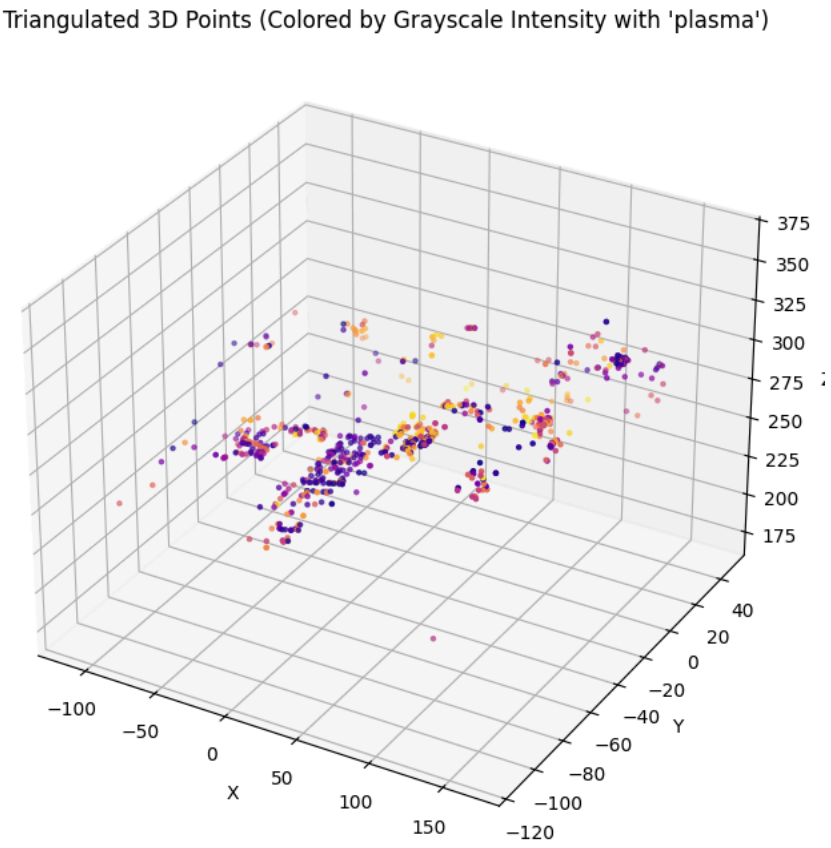

Figure 5. *Triangulated 3D point cloud*

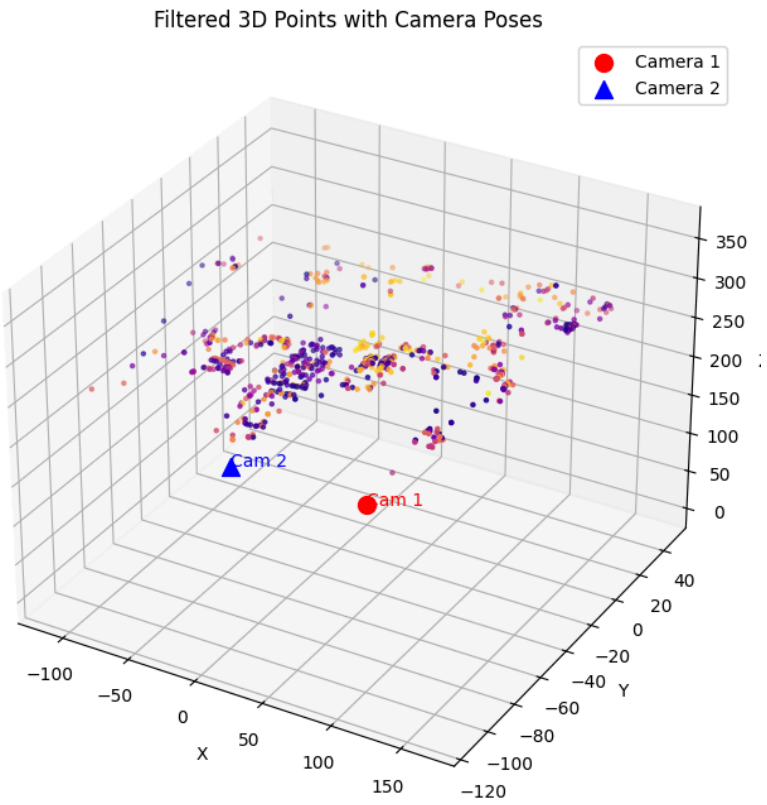

Figure 6. *Triangulated 3D Point Cloud - Filtered & with Camera Poses*

3. Results and Discussion

The results are provided in the Table 1.

Table 1. Detector Comparison: SIFT vs. ORB

Metric	SIFT	ORB
Keypoints Detected	8172/9024	1000/1000
Matches Found	899	366
Inliers (RANSAC)	773	191
Inlier Ratio	0.86	0.52
Match Ratio	0.11	0.37
Reprojection Error	0.79	0.82
Runtime	3.84 sec	0.13 sec

SIFT produced more robust and geometrically accurate matches, while ORB was significantly faster but less reliable. SIFT is preferred for offline reconstruction, whereas ORB fits real-time use cases.

3.2 Challenges and Solutions

- **Lighting Variability**: Uneven lighting conditions caused low contrast in some image areas, which impacted feature detection. Histogram equalization improved contrast and restored reliable detection. This was especially useful for ORB, which is more sensitive to illumination.
- **Keypoint Clutter**: SIFT returned thousands of keypoints, which overwhelmed epipolar plots and slowed matching. By filtering matches and limiting epipolar lines to a subset (e.g., 30 lines), visual clarity was significantly improved.
- **Scale Ambiguity**: The cv2.recoverPose() function estimates camera pose up to a scale. Without real-world depth information, a manual scale factor was introduced to meaningfully spread 3D points. Though arbitrary, it made the spatial relationships between cameras and points interpretable.
- **3D Outliers**: Initial 3D plots included many outliers due to mismatches. Applying bounding box thresholds helped clean the visualization while preserving core structure. This filtering made the final point cloud clearer and more structured.

3.3 Accuracy Analysis and Improvement Suggestions

- The fundamental matrix was visually validated through epipolar lines and reprojection errors.
- Triangulation errors were mostly due to image noise and imperfect matching.
- Potential improvements:
    - Incorporate reprojection error filtering before triangulation
    - Densify point cloud using dense stereo or multiview stereo techniques
    - Use structured light or depth sensors for ground truth comparison

Since no ground truth 3D structure was available, reconstruction quality was evaluated using reprojection error and epipolar alignment as proxies. These metrics helped ensure geometric consistency. In practice, stereo systems can further improve accuracy by integrating additional views or depth sensors.

There's a clear trade-off between **precision and speed**: SIFT yields higher-quality matches and better spatial consistency but is computationally heavier, taking 3–4 seconds per pair. ORB is much faster (~0.1 seconds) and more efficient, but often produces noisier reconstructions due to lower descriptor precision.

4. Conclusion

The project successfully demonstrates a full stereo vision pipeline for reconstructing 3D geometry from smartphone images. It includes calibration, feature detection, matching, epipolar geometry estimation, triangulation, and visualization with camera pose context. The comparison of SIFT and ORB also provides insights into the trade-offs between speed and accuracy in real-world computer vision applications.