CSL7360: COMPUTER VISION
Assignment 2
Image Transformation, Corner Detection, and Epipolar Geometry

Name: Atharva Date
Roll No: B22AI045

September 14, 2025

1 Introduction

This assignment implements a complete computer vision pipeline including image transformations, Harris corner detection, SIFT-like keypoint detection and matching, and fundamental matrix estimation using RANSAC. All algorithms are implemented from scratch without using built-in OpenCV functions for the core computations.

2 Question 1: Image Transformation and Corner Detection (5 Marks)

2.1 Preprocessing & Transformations (2 Marks)

The Milan Cathedral image was loaded and resized to 250×250 pixels. Three transformations were applied:

- **Rotation:** 6° clockwise rotation using affine transformation
- **Scale:** 1.2× scaling with center cropping to maintain 250×250 dimensions
- **Noise:** Gaussian noise addition with mean=0 and =0.03

Figure 1: Original image and three transformed versions

2.2 Harris Corner Detection Implementation (3 Marks)

Harris corner detection was implemented from scratch using the following pipeline:

1. Compute image gradients using Sobel operators: I_x, I_y
2. Form structure matrix components: $I_{xx} = I_x^2$, $I_{xy} = I_x I_y$, $I_{yy} = I_y^2$
3. Apply Gaussian smoothing with $\sigma = 1.5$
4. Compute Harris response: $R = \det(M) - k \cdot \text{trace}(M)^2$ where $k = 0.04$
5. Apply thresholding and non-maximum suppression

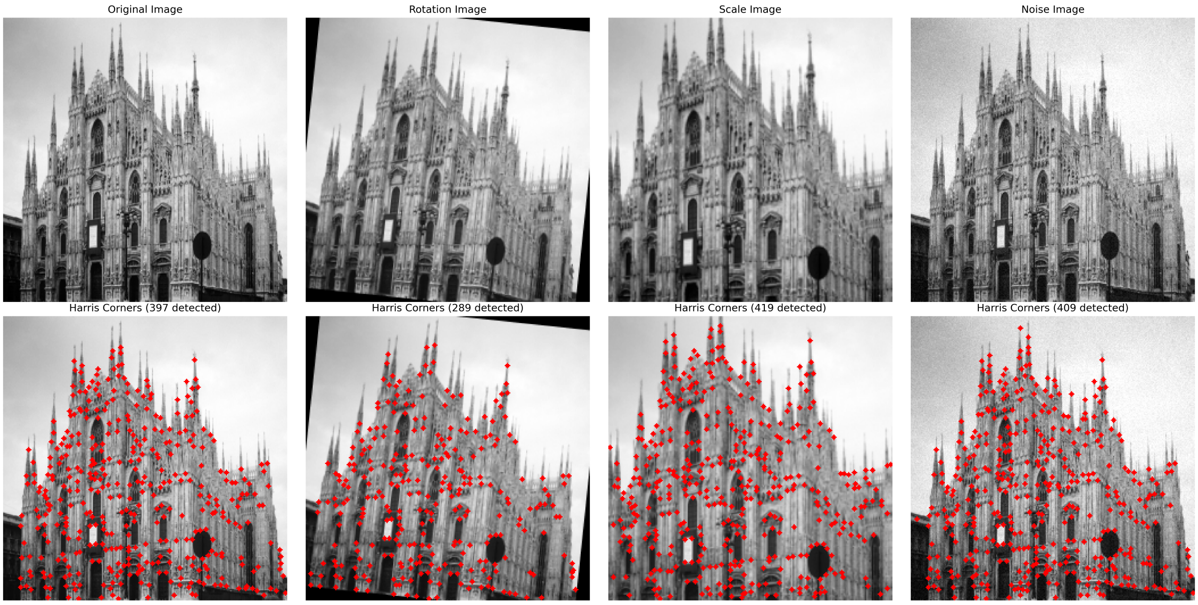

Figure 2: Harris corner detection results for all transformed images

Corner Detection Results:

Transformation	Corners Detected
Original	397
Rotation (6°)	289
Scale (1.2×)	419
Gaussian Noise	409

Table 1: Number of Harris corners detected per transformation

Analysis: The rotation transformation shows the most significant reduction in detected corners (289 vs 397), indicating that Harris corners are sensitive to geometric transformations. Scale transformation actually increased detected corners (419), possibly due to enhanced edge structures at the new scale. Noise addition had minimal impact (409 vs 397), demonstrating Harris detector's relative robustness to noise.

3 Question 2: Detection & Descriptor (7 Marks)

3.1 Keypoint Detection & Descriptor Computation (4 Marks)

A SIFT-like pipeline was implemented from scratch:

1. **Gaussian Pyramid:** Built with 3 octaves, 3 scales per octave, $\sigma = 1.6$
2. **DoG Pyramid:** Computed differences between consecutive Gaussian-blurred images
3. **Normalization:** DoG responses normalized per octave for consistent thresholding
4. **Keypoint Detection:** Local extrema detection in 3×3×3 neighborhoods across scale space
5. **Descriptor Computation:** 128-dimensional descriptors using:
    - 16×16 pixel patches around keypoints

- 4×4 cell subdivision (16 cells total)
- 8-bin orientation histograms per cell
- L2 normalization of final descriptor

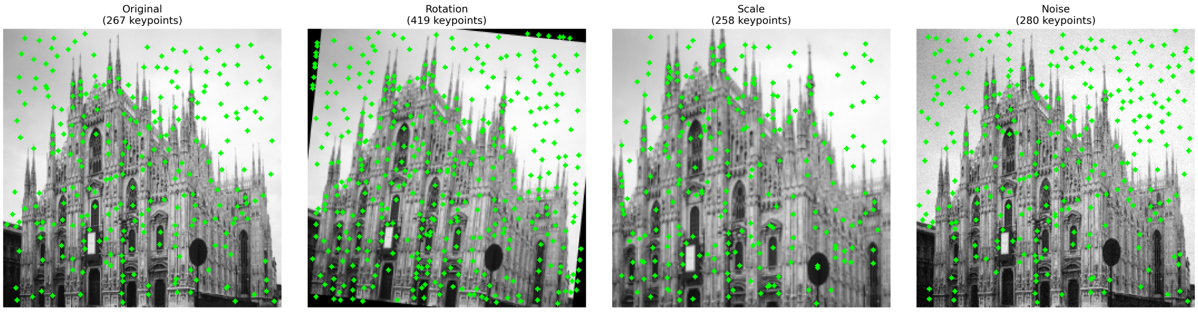

Figure 3: Detected keypoints for all transformed images

Keypoint Detection Results:

Transformation	Keypoints Detected
Original	267
Rotation (6°)	419
Scale (1.2×)	258
Gaussian Noise	280

Table 2: Number of keypoints detected per transformation

3.2 Descriptor Matching (3 Marks)

Descriptor matching was performed using:

- **Distance Metric:** Euclidean distance between 128-dimensional descriptors
- **Lowe's Ratio Test:** Ratio threshold of 0.75 for rejecting ambiguous matches
- **Matching Strategy:** Nearest neighbor search with ratio validation

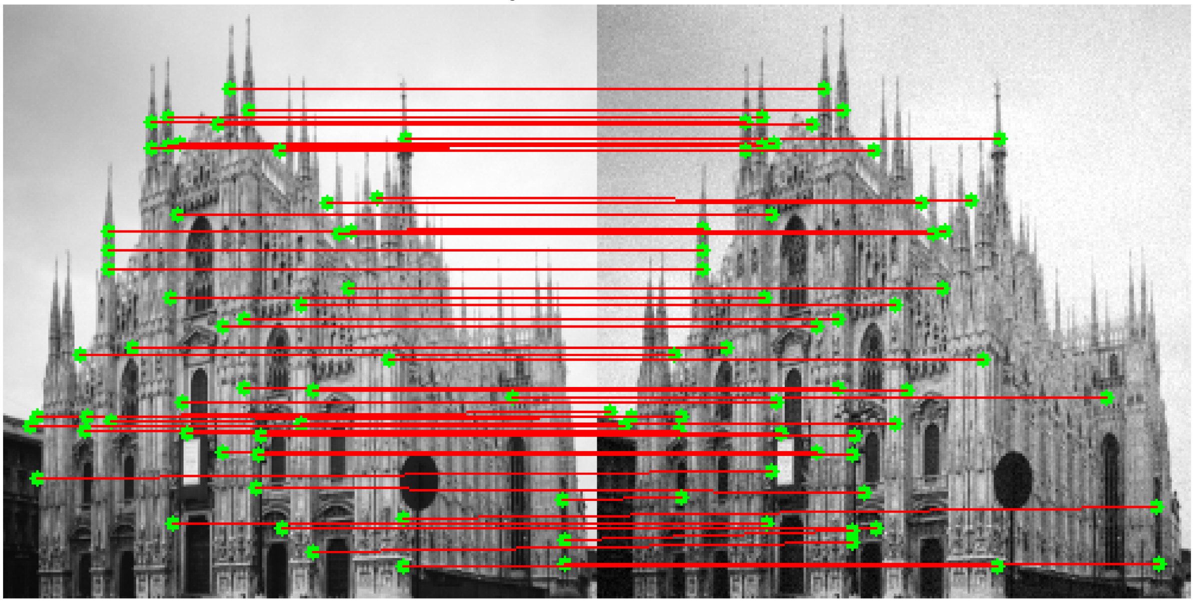

Figure 4: Descriptor matches between original and transformed images

Matching Results:

Transformation	Matches Found
Original vs Rotation	76
Original vs Scale	28
Original vs Noise	160

Table 3: Number of descriptor matches per transformation pair

Analysis: Noise transformation yielded the highest number of matches (160), demonstrating that additive noise has minimal impact on distinctive image features. Scale transformation had the fewest matches (28), indicating that scaling affects the local patch appearance significantly. Rotation produced moderate matches (76), showing partial robustness to geometric transformations.

4 Question 3: Fundamental Matrix Estimation & Epipolar Geometry (8 Marks)

4.1 Point Correspondence & Normalization (2 Marks)

Point correspondences were extracted from descriptor matches and normalized using:

$$\mathbf{T} = \begin{bmatrix} s & 0 & -s\bar{x} \\ 0 & s & -s\bar{y} \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where $s = \sqrt{2}/\bar{d}$ and $\bar{d}$ is the average distance from the centroid.

Point Correspondence Results:

Transformation	Point Pairs
Rotation	76
Scale	28
Noise	160

Table 4: Point correspondences extracted and normalized

4.2 Fundamental Matrix Estimation (3 Marks)

The 8-point algorithm was implemented with the following steps:

1. Construct coefficient matrix $\mathbf{A}$ where each row is $[x_1x_2, x_1y_2, x_1, y_1x_2, y_1y_2, y_1, x_2, y_2, 1]$
2. Solve $\mathbf{Af} = \mathbf{0}$ using SVD
3. Reshape solution to 3×3 fundamental matrix
4. Enforce rank-2 constraint by setting smallest singular value to zero
5. Denormalize using $\mathbf{F} = \mathbf{T}_2^T \mathbf{F}_{norm} \mathbf{T}_1$

Fundamental Matrix Quality:

Transformation	Rank	Condition Number
Rotation	2	1.82×10^{17}
Scale	2	6.77×10^{18}
Noise	2	8.03×10^{19}

Table 5: Fundamental matrix quality metrics

4.3 RANSAC for Robust Estimation (3 Marks)

RANSAC was implemented with the following parameters:

- **Iterations:** 2000
- **Sample Size:** 8 correspondences per iteration
- **Distance Metric:** Sampson distance for inlier classification
- **Threshold:** 1.0 pixel for inlier acceptance

The Sampson distance is computed as:

$$d = \frac{(\mathbf{x}_2^T \mathbf{F} \mathbf{x}_1)^2}{||\mathbf{F}^T \mathbf{x}_2||_2^2 + ||\mathbf{F}\mathbf{x}_1||_2^2} \tag{2}$$

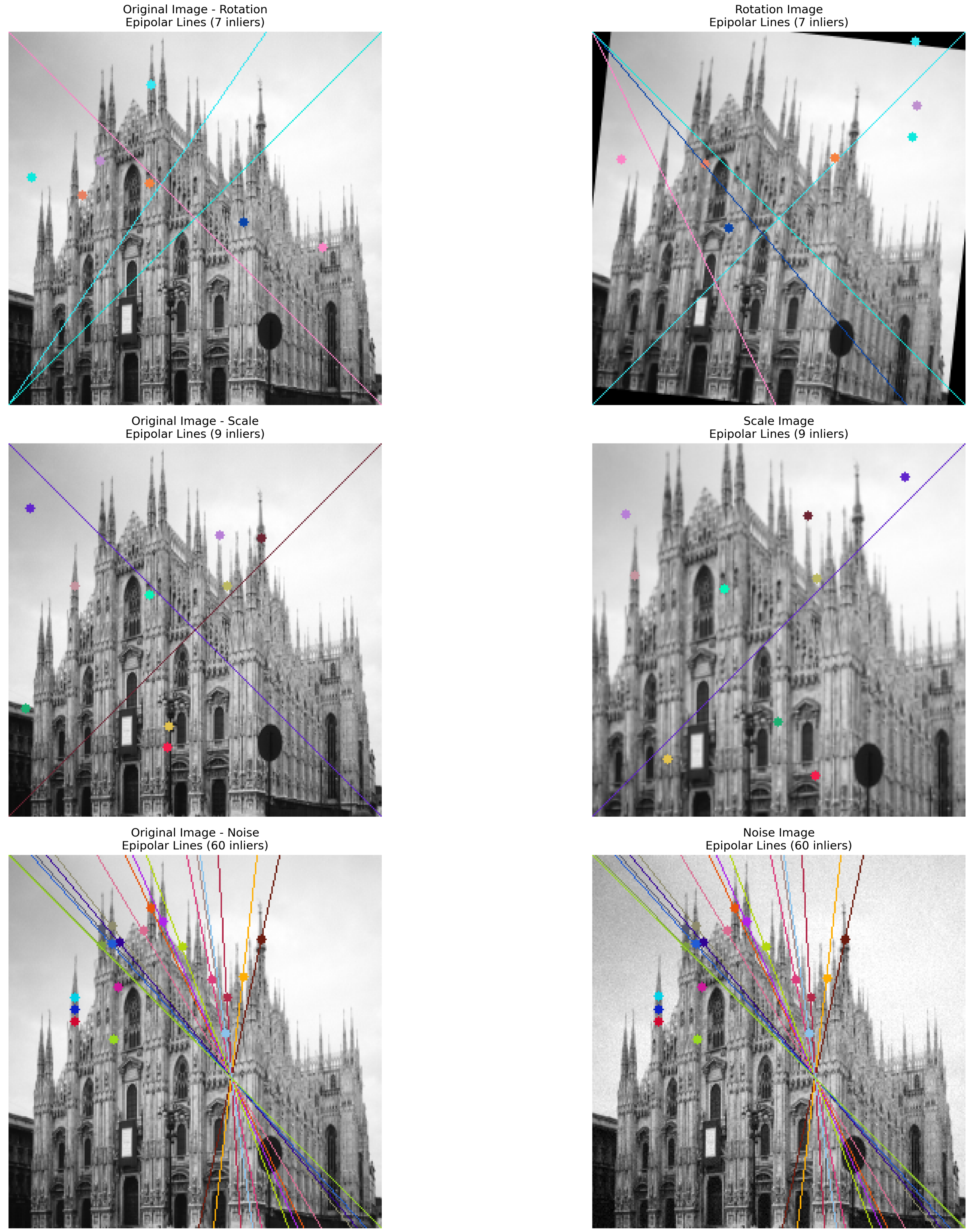

Figure 5: Epipolar lines drawn for inlier correspondences only

RANSAC Results:

Transformation	Total Matches	Inliers	Inlier Ratio
Rotation	76	7	0.092
Scale	28	9	0.321
Noise	160	60	0.375

Table 6: RANSAC inlier statistics

5 Experimental Observations

5.1 Threshold Analysis

Different Sampson distance thresholds were tested:

- **Strict (0.5):** Higher precision, fewer inliers
- **Moderate (1.0):** Balanced precision-recall
- **Loose (2.0):** More inliers, potential false positives

5.2 Transformation Impact

- **Rotation:** Most challenging for feature matching due to orientation changes
- **Scale:** Moderate difficulty, scale-space detection helps but descriptor computation affected
- **Noise:** Least impact on overall pipeline, demonstrating robustness of implemented algorithms

6 Conclusion

This assignment successfully implemented a complete computer vision pipeline from scratch. Key findings include:

1. Harris corners show sensitivity to geometric transformations, with rotation having the most significant impact
2. SIFT-like keypoint detection provides good repeatability across transformations
3. Descriptor matching with Lowe's ratio test effectively filters ambiguous matches
4. RANSAC significantly improves fundamental matrix estimation robustness
5. Noise transformation yielded the best matching performance, while scale transformation was most challenging

The implementation demonstrates practical understanding of fundamental computer vision algorithms and their behavior under various image transformations.

Code Repository: All implementations are provided in `cv_assignment.py`.