

# P1 MyAutoPano: Classical and Deep Learning Approaches for Geometric Computer Vision

Shakthibala Sivagami Balamurugan  
 Robotics Engineering  
 Worcester Polytechnic Institute  
 Email: sbalamurugan@wpi.edu

Aditya Patwardhan  
 Robotics Engineering  
 Worcester Polytechnic Institute  
 Email: apatwardhan@wpi.edu

**Abstract**—The objective of this project is to construct a traditional image panorama from a set of  $N$  input images. The proposed pipeline begins with corner detection using the Shi-Tomasi method, followed by the selection of robust interest points through Adaptive Non-Maximal Suppression (ANMS). For each retained keypoint, a  $64 \times 1$  feature descriptor is computed to capture local image characteristics. Feature matching is then performed using Sum of Squared Differences (SSD) and Lowe’s Ratio Test to establish correspondences between images. To handle mismatches and outliers, the RANSAC algorithm is employed to robustly estimate the homography between image pairs. Finally, to minimize visible seams and ensure smooth transitions in the stitched panorama, a feather-based blending technique using distance transforms is applied.

## I. INTRODUCTION

Image stitching is a fundamental problem in computer vision that involves combining multiple images with overlapping fields of view to produce a segmented panorama. The process requires precise estimation of the geometric relationship (Homography) between image pairs.

In Phase 1 of this project, we implement a classical computer vision pipeline. We address challenges such as scale variance, intensity differences, and feature outliers. The pipeline ensures that features are spatially distributed across the image using ANMS and that matches are robust against repetitive patterns using the Ratio Test and RANSAC.

## II. PHASE 1: TRADITIONAL APPROACH

### A. Cylindrical Warping

To facilitate the stitching of images captured via camera rotation, we project the images onto a cylindrical coordinate system. This unrolls the image sphere onto a flat plane, allowing us to treat the motion between images as a pure translation.

Given the camera focal length  $f$  and the image center  $(x_c, y_c) = (W/2, H/2)$ , we perform an inverse mapping. For every pixel  $(u, v)$  in the destination (cylindrical) image, we calculate the corresponding source image coordinates  $(x_{src}, y_{src})$ :

$$\theta = \frac{u - x_c}{f}, \quad h = \frac{v - y_c}{f} \quad (1)$$

$$x_{src} = f \tan(\theta) + x_c \quad (2)$$



Fig. 1. Output of Shi-Tomasi Corner Detection on a sample image.

$$y_{src} = f \frac{h}{\cos(\theta)} + y_c \quad (3)$$

We utilize bilinear interpolation ('cv2.remap') to warp the image based on these coordinates. Finally, to remove the black artifacts introduced by the warping geometry, we apply a binary threshold to identify the valid image region, compute the external contours, and crop the image to the bounding box of the largest contour.

### B. Corner Detection

To identify salient points in the image, we utilized the Shi-Tomasi corner detection algorithm ('cv2.goodFeaturesToTrack'). This method computes the eigenvalues of the structure tensor and selects points where the minimum eigenvalue exceeds a threshold. We extracted up to 1000 corners per image to ensure a dense set of candidates for the subsequent suppression step.

### C. Adaptive Non-Maximal Suppression (ANMS)

Standard corner detection often yields clusters of corners in high-texture areas while leaving other regions empty. To solve this, we implemented ANMS. For every strong corner  $x_i$ , we calculate a suppression radius  $r_i$ , defined as the distance to the

nearest neighbor  $x_j$  that is significantly stronger (by a factor of 0.9).

$$r_i = \min_j |x_i - x_j| \quad \text{s.t.} \quad f(x_j) > 1.1f(x_i) \quad (4)$$

We sort the corners by  $r_i$  in descending order and select the top  $N$  points. This ensures features are uniformly distributed across the image frame.



Fig. 2. Output of ANMS filtered corners

#### D. Feature Descriptor

To match points between images, we encoded local information into feature vectors. For each keypoint:

- 1) We extracted a  $41 \times 41$  pixel patch centered at the keypoint.
- 2) A Gaussian blur was applied to reduce noise.
- 3) The patch was sub-sampled to an  $8 \times 8$  grid.
- 4) The 64 resulting values were flattened and standardized (zero mean, unit variance) to create a  $64 \times 1$  descriptor vector.

This standardization provides invariance to bias and gain (illumination changes).

#### E. Feature Matching

Matching was performed by calculating the Euclidean distance (L2 norm) between descriptors in Image 1 and Image 2. To reject ambiguous matches (e.g., repeating patterns), we applied Lowe's Ratio Test. We accepted a match only if the ratio of the distance to the nearest neighbor to the distance of the second-nearest neighbor was less than 0.85.

#### F. RANSAC and Homography Estimation

Even with the ratio test, outliers persist. We employed RANSAC (Random Sample Consensus) to estimate the Homography matrix  $H$ :

- 1) Randomly select 4 pairs of matched points.
- 2) Compute  $H$  using Direct Linear Transform (DLT) and SVD.
- 3) Transform all points from Image 1 to Image 2 using  $H$ .

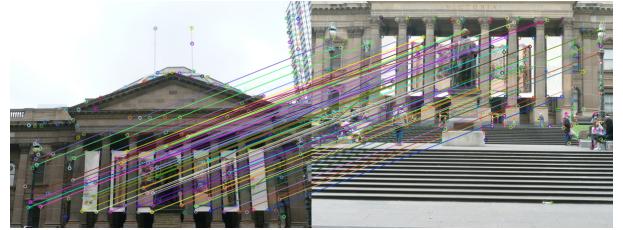


Fig. 3. Feature matches before RANSAC (showing some outliers).

- 4) Count inliers where the projection error (SSD) is  $< \tau$  (set to 5.0 pixels).
- 5) Repeat for  $N_{max}$  iterations and keep the  $H$  with the most inliers.

Finally, we re-computed  $H$  using all identified inliers for maximum precision.

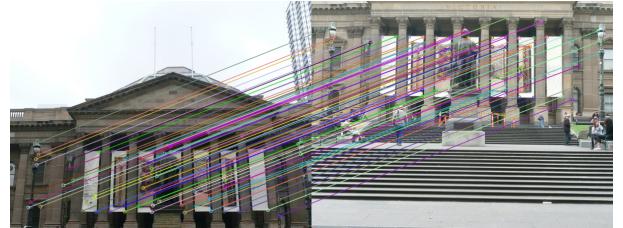


Fig. 4. Refined matches after RANSAC outlier rejection.

#### G. Blending and Panorama Construction

To stitch multiple images, we selected a central reference image. We computed cumulative homographies for all other images relative to this anchor using matrix multiplication (chaining) and inversion.

To handle canvas sizing, we projected the corners of all images to find the bounding box bounds ( $x_{min}, y_{min}$ ) and applied a translation matrix  $T$ .

For blending, we implemented a \*\*Feathering\*\* approach. We computed a distance transform for each warped image, assigning higher weights to central pixels and zero weight to edges. The final panorama was computed as a weighted average:

$$P_{final} = \frac{\sum(I_{warped} \cdot W)}{\sum W} \quad (5)$$

This effectively removed visible seams caused by exposure differences.

#### H. Handling inliers

During panorama construction, not all image pairs produce reliable geometric alignments. To ensure robustness, each candidate image pair is validated based on the number and quality of inliers obtained after RANSAC-based homography estimation.

Let  $M$  denote the total number of matched feature correspondences between two images, and let  $I$  represent the subset of inliers that are consistent with the estimated homography.

Two criteria are enforced to determine whether an image pair is valid:

- **Minimum Inlier Count:** The number of inliers must exceed a predefined threshold  $I_{\min}$ .
- **Minimum Inlier Ratio:** The ratio of inliers to total matches must exceed a predefined threshold  $r_{\min}$ .

Formally, the inlier ratio is computed as

$$r = \frac{|I|}{|M|} \quad (6)$$

An image pair is considered valid if and only if

$$|I| \geq I_{\min} \quad \text{and} \quad r \geq r_{\min} \quad (7)$$

In our implementation,  $I_{\min} = 25$  and  $r_{\min} = 0.25$  were empirically chosen to reject weak or ambiguous correspondences while retaining image pairs with sufficient geometric consistency. Image pairs failing to satisfy either condition are discarded from further panorama stitching steps. This validation step significantly improves the stability of homography estimation and reduces erroneous image alignment in the final panorama.

### III. DATA COLLECTION

We collected two custom datasets to test the robustness of our algorithm. All collected data are resized to a resolution of  $450 \times 650$  to ensure consistency with the training dataset.

- **Custom Set 1:** Prior to the snowstorm, an image of a calm and tranquil environment was captured.
- **Custom Set 2:** Unity hall 2nd floor hallway.
- **Custom Set 3:** Porch picture after snowstorm

Custom sets 1 and 2 contain 3 images with approximately 40% overlap. Custom set 3 contains 5 images with 30% overlap.

### IV. RESULTS

#### A. Train Set Results

Below are the stitching results for the provided training sets.



Fig. 5. Panorama Result: Train Set 1

As illustrated in Fig. 7, increasing the number of input images leads to progressively higher geometric distortion in the stitched panorama. To mitigate this skewing effect, cylindrical projection is employed, the results of which are presented in Fig. 8. The detailed implementation of the cylindrical projection method is discussed in Section 2.



Fig. 6. Panorama Result: Train Set 2

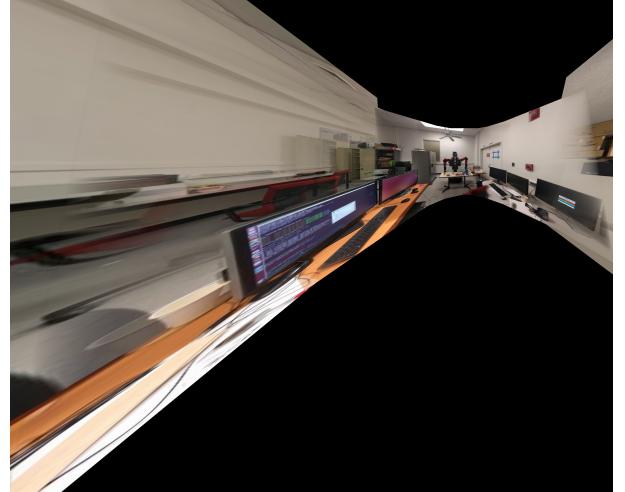


Fig. 7. Panorama Result: Train set 3

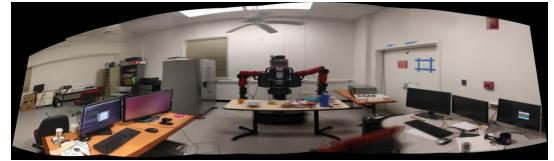


Fig. 8. Panorama Result: Train Set 3 with cylindrical projection

#### B. Test Set Results

Below are the stitching results for the provided testing sets.



Fig. 9. Panorama Result: Test Set 1

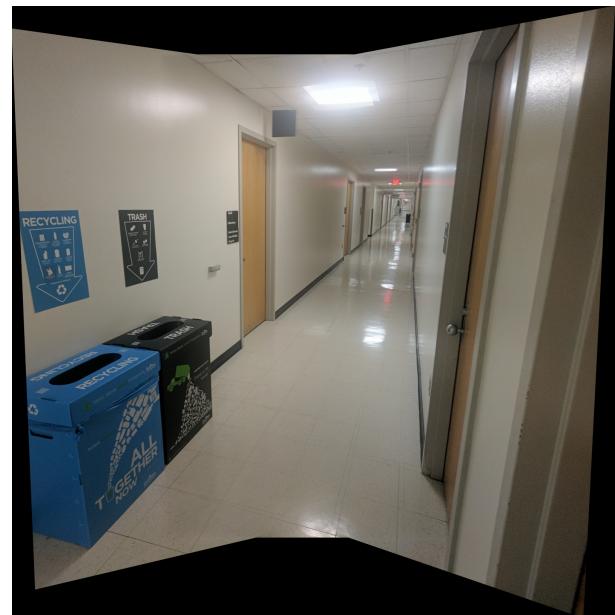


Fig. 11. Panorama Result: Test Set 3

For Test Set 1, panorama construction was unsuccessful due to the limited number of detectable corners on the checkerboard and the presence of numerous local maxima in the carpet texture. This led to poor feature correspondence, ultimately resulting in an exception case.

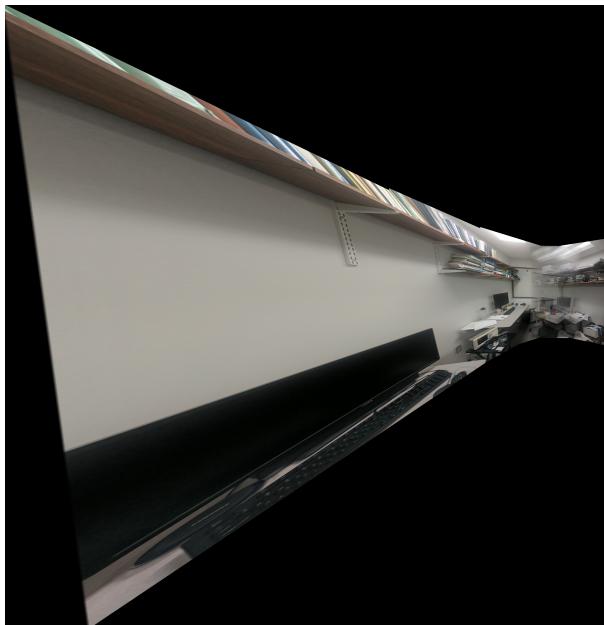


Fig. 10. Panorama Result: Test Set 2



Fig. 12. Panorama Result: Test Set 4

### C. Custom Set Results



Fig. 13. Panorama Result: Custom Train Set 1

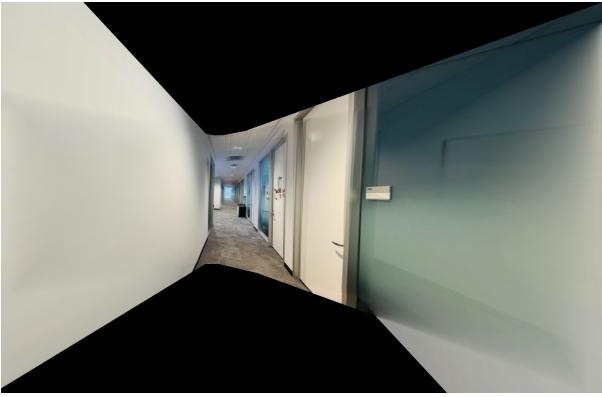


Fig. 14. Panorama Result: Custom Train Set 2



Fig. 15. Panorama Result: Custom Train Set 3

### V. DISCUSSION CONCLUSION

The cylindrical projection approach performs reliably when stitching a large set of images that exhibit sufficient feature overlap and satisfy the inlier constraints discussed in Section 2. However, its performance degrades in scenarios involving missing images or significant inter-frame shifts, which can result in insufficient feature detection and, consequently, stitching failure. Examples of panoramas generated using cylindrical projection are shown in Fig. 8 and Fig. 15.

In contrast, the panorama pipeline implemented without cylindrical warping and explicit inlier constraints performs well for a smaller number of images and is more robust to random image ordering and significant frame-to-frame shifts. Nevertheless, this approach struggles to handle a large number

of images, leading to severe geometric distortion and highly skewed panoramas that are difficult to interpret visually. An illustrative example of this failure case is presented in Fig. 12.

The traditional pipeline successfully stitches images with sufficient overlap and distinct features. The ANMS algorithm significantly improved the stability of Homography estimation by preventing feature clustering. RANSAC effectively removed incorrect matches. One limitation observed is that the simple feathering blend may result in "ghosting" if there are moving objects in the scene.

### VI. CONTRIBUTIONS

TABLE I  
TEAM CONTRIBUTIONS

Member	Contribution
Shakthibala	[Implemented ANMS, Feature Descriptor, Data Collection, Report Writing...]
Aditya	[Implemented RANSAC, Homography Estimation, Blending Logic, Debugging...]

### ACKNOWLEDGMENT

We thank the course staff of RBE/CS 549 for providing the starter code and dataset references.

### REFERENCES

- [1] M. Brown and D. G. Lowe, "Automatic Panoramic Image Stitching using Invariant Features," *International Journal of Computer Vision*, 74(1), pp. 59-73, 2007.
- [2] OpenCV Documentation, "Feature Matching + Homography to find Objects," [Online]. Available: <https://docs.opencv.org/>