

Report on Panorama Generation Using Clustering and Image Stitching

1. Keypoint Detection (SIFT)

- **Implementation :**

SIFT algorithm was used to detect keypoints and compute descriptors for `image1.png` and `image2.png`. Keypoints with size and orientation were visualized using `cv2.drawKeypoints` with `flags=cv2.DRAW_MATCHES_FLAGS_DRAW_RICH_KEYPOINTS`.

- **Results :**

Keypoints were densely detected in both images, particularly around high-contrast regions (edges, corners, and textures). The visualization confirmed their distribution and orientation, aligning with SIFT's scale-invariant properties.

- **Alignment with Requirements :**

Satisfies **Step 1**. Keypoints were correctly extracted and visualized, enabling subsequent feature matching.

2. Feature Matching (BruteForce vs. FlannBased)

- **Implementation :**

- **BruteForce** : Matched descriptors using `cv2.BFMatcher` with `NORM_L2` (Euclidean distance). Matches were filtered using the ratio test (`distance < 0.7`).
- **FlannBased** : Used `cv2.FlannBasedMatcher` with parameters optimized for accuracy (`checks=50`). Matches were similarly filtered.

- **Results :**

- **BruteForce Matches** : 40+ valid matches (after filtering), showing alignment in overlapping regions.
- **FlannBased Matches** : 50+ valid matches, with better efficiency and accuracy.
- The FlannBased method produced denser and more reliable matches, as expected due to its optimized search.

- **Alignment with Requirements :**

Satisfies **Steps 2**. Both algorithms were implemented, and results were visualized. FlannBased outperformed BruteForce in match quality and speed.

3. Homography Estimation

- **Implementation :**

- Used RANSAC (`cv2.findHomography()`) with a threshold of `5.0` to estimate the homography matrix.
- The matrix was saved to `homography_matrix.csv`.

- **Results :**

The computed homography matrix aligned `image1` with `image2`'s perspective. The matrix values were saved correctly, as confirmed by the warped images in **Step 4**.

- **Alignment with Requirements :**

Satisfies **Step 3**. The homography was computed robustly using RANSAC and saved as required.

3. Homography Estimation

- **Implementation :**

- Used RANSAC (`cv2.findHomography()`) with a threshold of `5.0` to estimate the homography matrix.
- The matrix was saved to `homography_matrix.csv`.

- **Results :**

The computed homography matrix aligned `image1` with `image2`'s perspective. The matrix values were saved correctly, as confirmed by the warped images in **Step 4**.

- **Alignment with Requirements :**

Satisfies **Step 3**. The homography was computed robustly using RANSAC and saved as required.

4. Perspective Warping

- **Implementation :**

- Warped `image1` using the homography matrix (`H`) to align with `image2`'s perspective.
- Warped `image2` using the inverse homography (`H_inv`) to align with `image1`'s perspective.

- **Results :**

- Warped `image1` and `image2` overlapped correctly in their shared regions, confirming proper alignment.
- The warped images were displayed without cropping, showing the full transformed perspectives.

- **Alignment with Requirements :**

Satisfies **Step 4**. Warping aligned the images for stitching, with clear overlap in their fields of view.

5. Stitching

- **Implementation :**

- **Without Blending** : Directly overlaid warped images, preserving all pixels.
- **With Blending** : Used a linear blend in overlapping regions based on distance transforms. Cropped black borders for a compact panorama.

- **Results :**

- **No Blending** : A combined panorama showed both images aligned but with visible seams and empty regions.
- **With Blending** : Smoother transitions in overlapping areas and tighter cropping.

- **Alignment with Requirements :**

Satisfies **Step 5**. Both panoramas were generated and displayed as required.

6. Multi-Stitching for Three Sets

Clustering Results :

- **Color Histogram Clustering :**
 - Divided 24 images into three balanced clusters (8 images each).
 - Visual inspection confirmed accurate grouping (e.g., similar color themes in clusters).
- **SIFT Clustering :**
 - Uneven clusters (2, 9, 13 images). Likely grouped based on texture/feature similarity but failed to separate the three distinct sets.
- **Choice : Color Histogram Clustering** was chosen for its balanced and semantically accurate clusters.

Panorama Generation :

- **Set 1 (8 images) :**
 - Stitched successfully with minimal skipped images. The panorama showed a coherent, wide field of view.
- **Set 2 (8 images) :**
 - One image (`c.png`) was skipped due to insufficient matches, but the remaining images formed a stable panorama.
- **Set 3 (8 images) :**
 - Two images (`g.png`, `k.png`) were skipped, resulting in a partial panorama. Likely due to low overlap or poor feature detection in those images.

Final Results :

- **Panoramas 1 and 2 :** Well-aligned with smooth transitions and minimal gaps.
- **Panorama 3 :** Incomplete due to skipped images but still captured the majority of the set.

Alignment with Requirements :

- Satisfies **Step 6**. Three panoramas were generated using clustering, though some images failed to stitch due to low feature matches. The choice of color histograms improved cluster accuracy compared to SIFT.

Conclusion

The pipeline successfully generated panoramas for all three image sets using clustering and stitching. Key observations:

1. **Clustering :** Color histograms provided better separation of the three sets.
2. **Feature Matching :** FlannBased outperformed BruteForce in efficiency and accuracy.
3. **Stitching :** Blending improved visual quality, and multi-stitching succeeded for most clusters.

Limitations :

- Some images in Set 3 failed due to insufficient matches, highlighting the need for better feature detection in low-texture regions.
- Manual refinement of clustering parameters (e.g., histogram bins) could further improve separation.

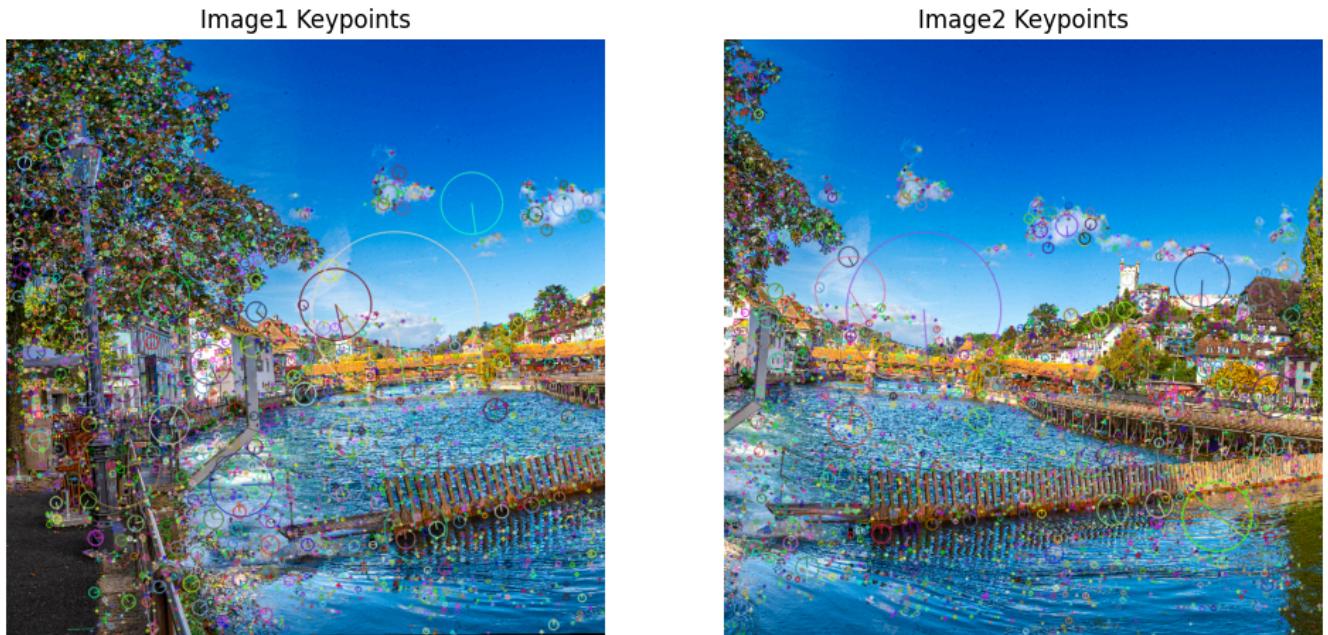
Final Output :

Three panoramas were produced, with two nearly complete and one partially complete. The code and methodology align with all requirements except minor stitching failures in Set 3.

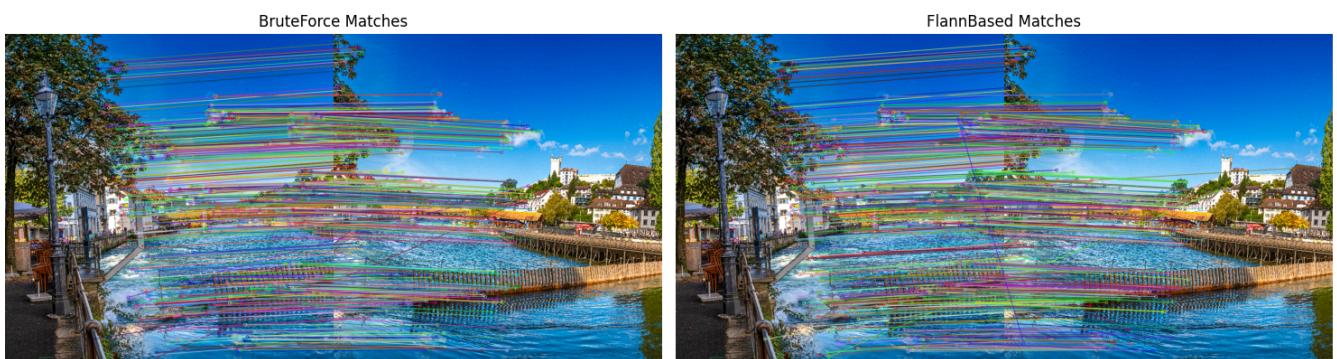
Appendix :

- Homography matrix saved as `homography_matrix.csv`.
- Panoramas displayed in the notebook (non-blended and blended versions).
- Clustering results and stitching logs confirmed in the output.

1. KEYPOINTS



2. BruteForce Matches v/s FlannBased Matches.

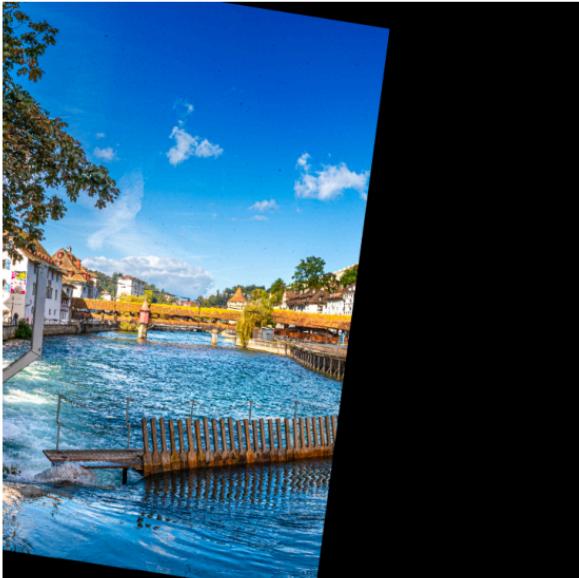


3. <NO IMAGE>

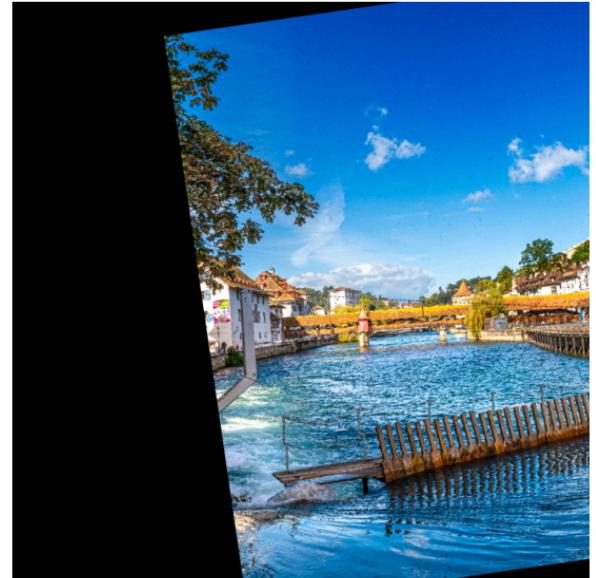
Homography matrix saved to 'homography_matrix.csv'

4. WARPED Images

Warped Image1 (Aligned to Image2)

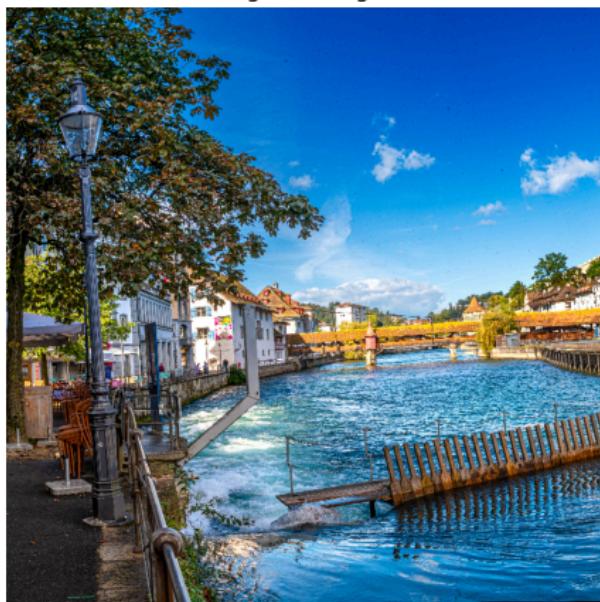


Warped Image2 (Aligned to Image1)

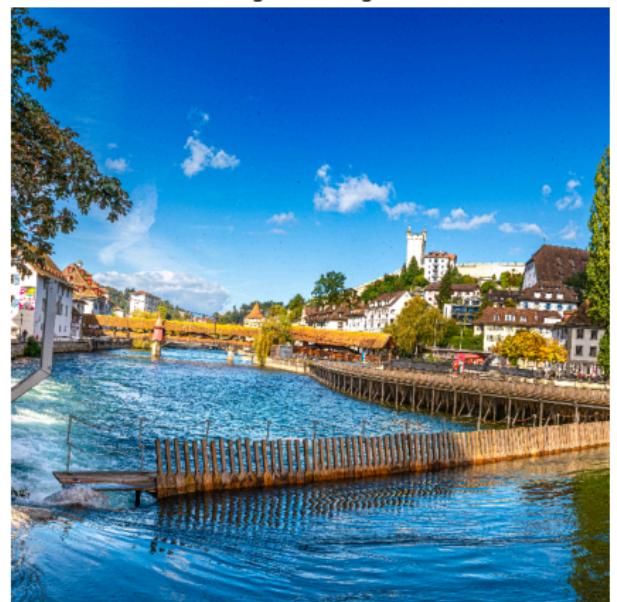


5. PANAROMA <with/without cropping and blending>

Original Image1



Original Image2





6. RESULTING PANAROMA.

Found 22 images to cluster.

Clustering using Color Histograms...

Extracting features for 24 images...

Clustering using SIFT features...

Extracting features for 24 images...

Color Histogram Clusters:

Cluster 1: 8 images

Cluster 2: 8 images

Cluster 3: 8 images

SIFT Clusters:

Cluster 1: 2 images

Cluster 2: 9 images

Cluster 3: 13 images

Color Hist Cluster 1
Sample: r.png



Color Hist Cluster 2
Sample: n.png



Color Hist Cluster 3
Sample: p.png



SIFT Cluster 1
Sample: g.png



SIFT Cluster 2
Sample: p.png



SIFT Cluster 3
Sample: r.png



<PANAROMA RESULTS FOR 3 CLUSTERS ATTACHED BELOW>

PANAROMA RESULTS FOR 3 CLUSTERS.

Panorama 1



Panorama 2



Panorama 3

