Objectives

- To combine multiple images into a single larger image (panorama).
- To utilize image processing techniques for feature detection and matching.
- To implement and understand homography matrix calculation and image warping.
- To address blending issues for seamless image integration.

Introduction

Image stitching is the process of combining multiple images to create a single larger image, often referred to as a panorama. This technique is widely used in applications such as panoramic photography, virtual tours, remote sensing, GIS, medical imaging, and art.

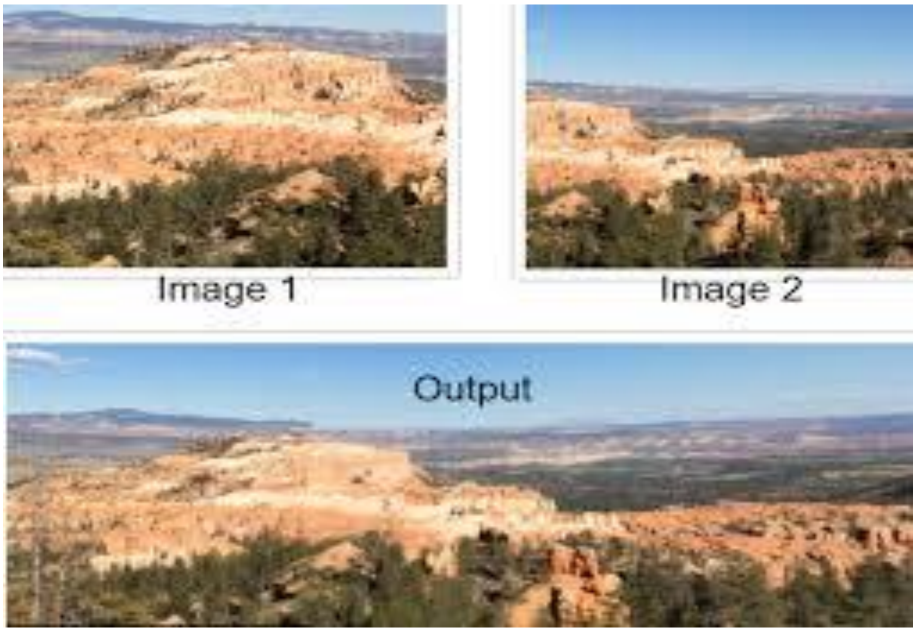

Fig-1: Input Image

Theory

Image stitching involves several key steps:

1. **Feature Detection**: Identifying key points in images using algorithms like SIFT (Scale-Invariant Feature Transform).

2. **Feature Matching**: Matching these key points between images using techniques like brute force matching.

3. **Homography Matrix Calculation**: Calculating a transformation matrix that maps points from one image to corresponding points in another.

4. **Image Warping**: Aligning the images based on the homography matrix.
5. **Blending**: Combining the images into a single seamless panorama.

Methodology

1. Feature Detection

Feature detection is a critical step in the image stitching process, as it involves identifying key points in the images that can be matched between the overlapping regions. The Scale-Invariant Feature Transform (SIFT) algorithm is commonly used for this purpose because it is robust to changes in scale, rotation, and illumination. The SIFT algorithm includes the following steps:

- **Keypoints Detection**: SIFT generates a series of images at different scales using a process called image pyramids. The algorithm applies Gaussian blurring and subsampling to the original image to create these pyramids. At each level of the pyramid, the Difference of Gaussians (DoG) is computed by subtracting adjacent levels of Gaussian blurred images. Keypoints are identified by comparing each pixel in the DoG images with its 26 neighbors (8 in the same scale and 18 in the neighboring scales). If the pixel is a local maximum or minimum, it is considered a keypoint candidate.

- **Feature Description**: Once keypoints are detected, each keypoint is described using a feature vector that captures its local appearance, orientation, and structural information. This is achieved through gradient histograms, which represent the distribution of gradient directions and magnitudes around the keypoint.

- **Gradient Calculation**: For each keypoint, the gradient magnitude and direction are computed within a local neighborhood (typically a 16x16 window). This helps in describing the local image structure around the keypoint.

- **Histogram Accumulation**: A histogram of gradient orientations is constructed where each bin accumulates gradient magnitudes weighted by their orientations. This histogram forms the keypoint's descriptor, making it robust to changes in illumination and small distortions.

- **Multiple Orientations**: If there are multiple peaks in the gradient orientation histogram that are within 80% of the highest peak, multiple orientations are assigned to the keypoint. This increases the robustness of the keypoint matching process by allowing each keypoint to be represented by multiple descriptors.

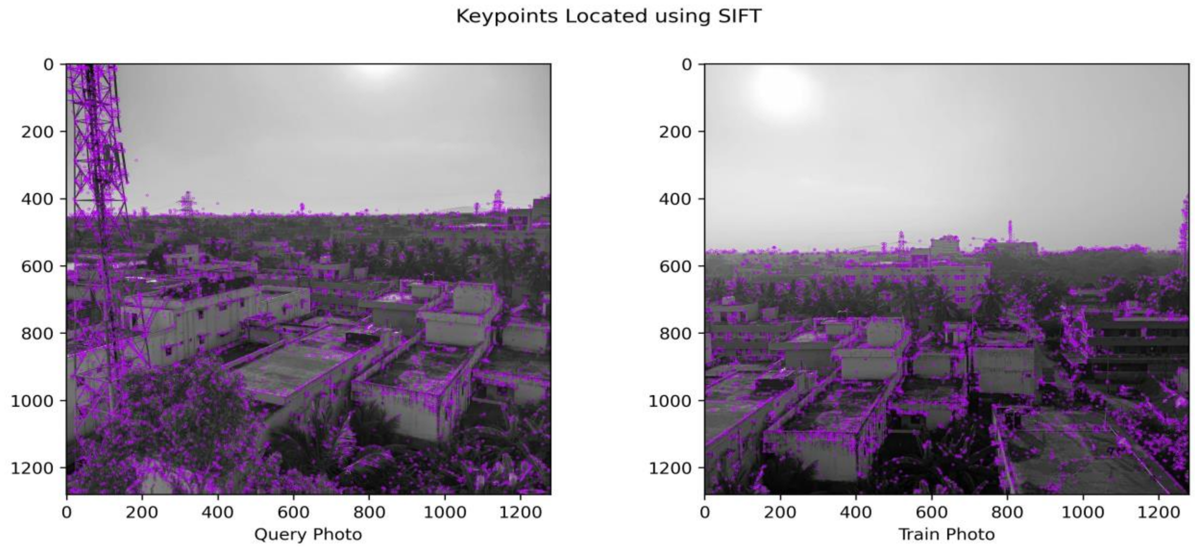

Fig-2: Key points

2. Feature Matching

Feature matching involves finding corresponding keypoints between the overlapping images. This is achieved through the following steps:

- **Brute Force Matching**: Each feature descriptor in one image is compared to every feature descriptor in the other image. This exhaustive search ensures that all possible matches are considered.

- **Distance Calculation**: For each feature descriptor, the Euclidean distance to every other descriptor in the second image is calculated. The Euclidean distance measures the similarity between descriptors; smaller distances indicate more similar descriptors.

- **Best Match Identification**: The best match for each feature descriptor is identified as the one with the smallest Euclidean distance. This step finds the closest corresponding keypoints between the images.

- **Threshold Application**: A threshold is applied to filter out matches that have distances exceeding a certain value. This helps eliminate poor matches and reduces the likelihood of incorrect correspondences.

- **Sorting Matches**: The matches are sorted based on their distance or similarity score. This sorted list can be used for further processing, such as homography estimation.

Fig-3: Feature Mapping

3. Homography Matrix Calculation

The homography matrix describes the geometric relationship between corresponding points in the two images. It is a 3x3 matrix that defines how points in one image can be transformed to align with points in the other image. The calculation of the homography matrix involves the following:

- **RANSAC Algorithm**: The Random Sample Consensus (RANSAC) algorithm is used to estimate the homography matrix. RANSAC is an iterative algorithm that is robust to outliers and can handle mismatches between corresponding points. It repeatedly selects random subsets of correspondences, computes the homography matrix for each subset, and evaluates its accuracy by counting the number of inliers (points that fit the estimated transformation well).

- **The homography matrix is a 3x3 matrix and can be represented as:**

$$H= \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

❖ RANSAC algorithm is used to find the value of homography matrix. It is an iterative algorithm commonly used in. RANSAC is often used when dealing with correspondences between points in two images that may include outliers or mismatches

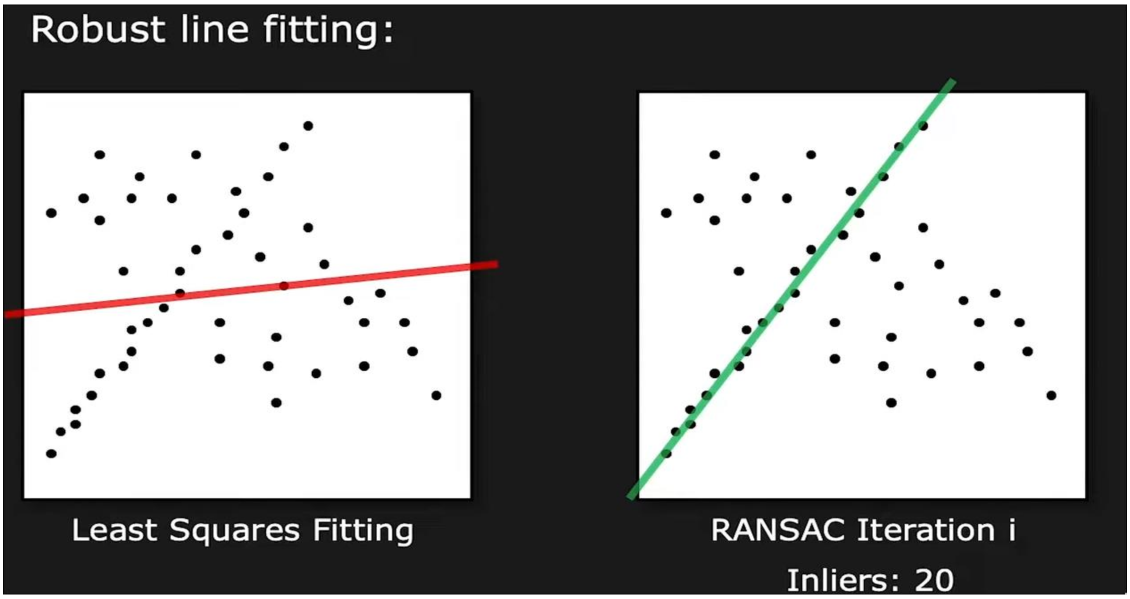

Fig-4: RANSA apply

4. Image Warpping

Image warping is the process of transforming the perspective of one image to align with another using the homography matrix. This step ensures that the images align correctly for stitching:

- **Applying Homography Matrix**: The homography matrix HHH is applied to each point in the source image to compute the corresponding points in the destination image. This transformation aligns the features (like edges, corners, and other keypoints) of one image with the corresponding features in the other image.

- **Perspective Transformation**: The transformation involves shifting, rotating, and scaling the source image to match the perspective of the destination image. This ensures that the overlapping regions align properly, facilitating seamless stitching.

5. Blending

Blending addresses the issue of differences in contrast and brightness between the images. This step ensures a seamless transition between the stitched images:

- **Weighted Blending**: To solve the issue of varying brightness and contrast, a weighted blending technique is used. This technique involves creating a linear gradient mask that smoothly transitions the intensity values from one image to another.
- **Creating Gradient Mask**: A linear gradient mask is created and duplicated across the three color channels (BGR). This mask is applied to the overlapping regions of the images to blend them together.
- **Blending Operation**: The blending operation involves element-wise multiplication of each image by its corresponding mask. This ensures a smooth transition between the images, reducing visible seams and making the final panorama appear seamless.

FIg-5: Before blending

FIg-6: After Blending

By following these steps, the image stitching process can effectively combine multiple images into a single, larger panoramic image, with smooth transitions and accurate alignment of overlapping

Input:

FIg-7: Left Image

FIg-8: Right Image

Output:

Fig-9: Output Image

Discussion

The image stitching process was successfully implemented, demonstrating the use of advanced image processing techniques. Key challenges included ensuring accurate feature detection and matching, as well as handling variations in image brightness and contrast. The use of SIFT for feature detection and RANSAC for homography calculation proved effective in creating seamless panoramas.

Conclusion

The project successfully combined multiple images into a single panorama using feature detection, matching, homography calculation, image warping, and blending techniques. The methodology provided a thorough understanding of the image stitching process and its applications. Future improvements could include optimizing the algorithm for speed and handling more complex stitching scenarios.

References

1. IEEE Xplore, “Automatic Panoramic Image Stitching using Invariant Features,” International Journal of Computer Vision, Jan. 2006.

2. Farhadi Ali et al., “Every picture tells a story: Stitching multiple images,” European Conference on Computer Vision, Springer Berlin Heidelberg, 2010.

3. O. Vinyals, A. Toshev, S. Bengio et al., "Show and tell: Lessons learned from the 2015 MSCOCO Detecting feature using SIFT and stitching image," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 4, 2017.