Panorama Stitching: A Practical Implementation (CS 333 Lab Report)

Code Repository: https://github.com/notsanidhyak/Panaorama-Stitching

Samanway Maji
Computer Science and Engineering
202151136@iiitvadodara.ac.in

Riya
Computer Science and Engineering
202151132@iiitvadodara.ac.in

Sanidhya Kumar
Computer Science and Engineering
202151138@iiitvadodara.ac.in

***Abstract* - This report presents a pragmatic implementation of panorama stitching, leveraging existing techniques. The methodology includes transformations, homograph computation, SIFT feature extraction, RANSAC outlier handling, and image warping/blending. While not introducing novel approaches, the report emphasizes the effective utilization of well-established methods for creating seamless panoramas. It serves as a practical guide for panorama stitching implementation, making use of established techniques in the field.**

I. INTRODUCTION

Panorama stitching stands at the intersection of computer vision and image processing, revolutionizing the way we capture and perceive visual content. The essence of panorama stitching lies in seamlessly combining multiple images to create a panoramic view that extends beyond the limitations of a single photograph. This transformative technology has found applications in various domains, including photography, virtual reality, architectural visualization, and surveillance.

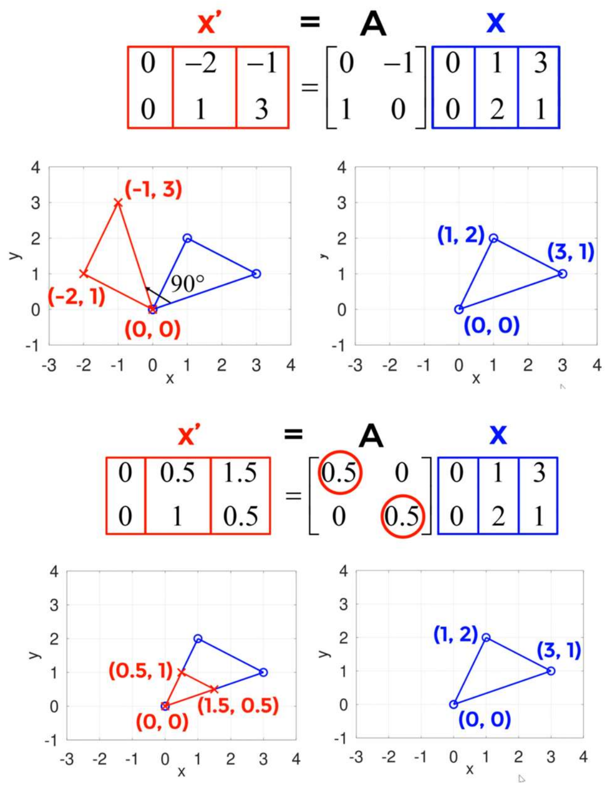
One of the key challenges addressed by panorama stitching is the distortion caused by differences in perspective, lighting conditions, and camera parameters across images. Geometric transformations and homograph computation play a crucial role in aligning these images accurately, ensuring a seamless transition between them. Feature extraction techniques, such as the widely used Scale-Invariant Feature Transform (SIFT), enhance the identification of distinctive features within images, contributing to robust stitching results.

Handling outliers is another critical aspect of panorama stitching. The Random Sample Consensus (RANSAC) algorithm is commonly employed to identify and eliminate erroneous matches, improving the accuracy and reliability of the stitching process. Moreover, the final steps involve image warping and blending, where the aligned images are seamlessly fused to create a visually harmonious panorama.

II. THEORY

A. Transformation of images:

Transformations of images refer to fundamental operations that modify the spatial characteristics of an image without changing its content. These transformations are essential in various fields, including computer vision, image processing, and computer graphics. The most common basic image transformations include translation, rotation, scaling, and flipping.



B. Homography:

A homography matrix is a transformation matrix that describes the geometric relationship between two images of the same scene taken from different perspectives or under different camera orientations. This matrix is particularly essential in the context of image stitching, where multiple images need to be aligned to create a seamless panorama.

Briefly, the planar homography relates the transformation between two planes (up to a scale factor):

$$s\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \mathbf{H}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

The homography matrix is a 3x3 matrix but with 8 Degrees of freedom (degrees of freedom) as it is estimated up to a scale. It is generally normalized with $h_{33}=1$ or
$\mathbf{h^2_{11}+h^2_{12}+h^2_{13}+h^2_{21}+h^2_{22}+h^2_{23}+h^2_{31}+h^2_{32}+h^2_{33}=1.}$

The following examples show different kinds of transformation but all relate a transformation between two planes.

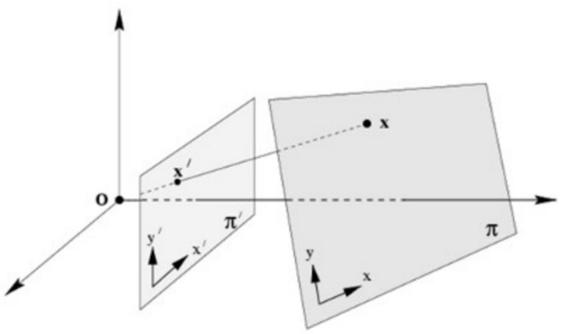


Fig: A planar surface and the image plane

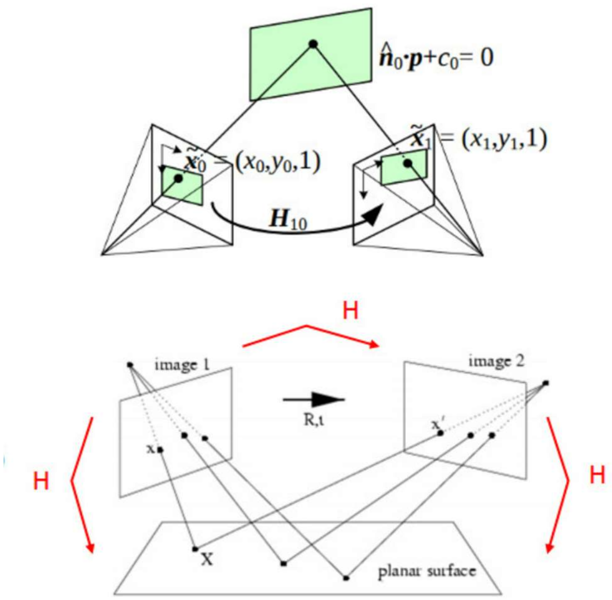


Fig: a planar surface viewed by two camera positions

C. SIFT:

Scale-Invariant Feature Transform (SIFT) is a popular method for extracting distinctive and invariant features from images. SIFT features are widely used in computer vision tasks such as object recognition, image stitching, and matching. The SIFT feature extraction process involves several key steps:

Scale-space Extrema Detection:

SIFT operates on multiple scales to detect features at different levels of detail. It uses a Gaussian pyramid to create a series of blurred images. At each scale level, the Difference of Gaussians (DoG) is computed by subtracting adjacent blurred images. Local extrema in the DoG represent potential keypoint locations.

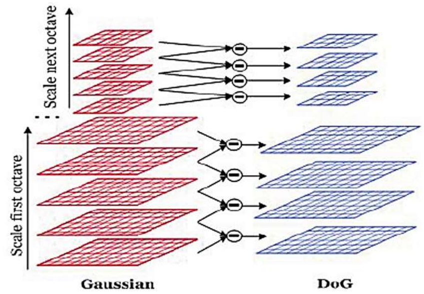


Fig. Construction of the Difference of Gaussian (DoG) scale space

Keypoint Localization:

Once potential keypoints are identified, the algorithm refines their locations for accuracy. It fits a 3D quadratic function to the nearby data points in the DoG space, allowing for sub-pixel localization of keypoints. Keypoints with low contrast or poorly localized are discarded.

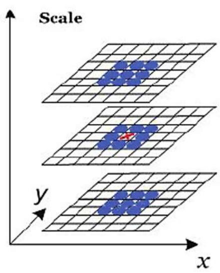


Fig: The pixel marked x is compared with the neighboring pixels (in green) and is selected as a keypoint or interest point if it is the highest or lowest among the neighbors

Orientation Assignment:

SIFT computes the orientation of each keypoint to ensure invariance to image rotation. A gradient histogram is generated for the region around each keypoint, and the dominant orientation is determined. The keypoint descriptor will be aligned with this orientation

Keypoint Descriptor Generation:

A local image patch around each keypoint is sampled based on the orientation calculated in the previous step. This patch is divided into smaller regions, and for each region, a histogram of gradient orientations is computed. These histograms are concatenated to form the final feature vector, known as the SIFT descriptor.

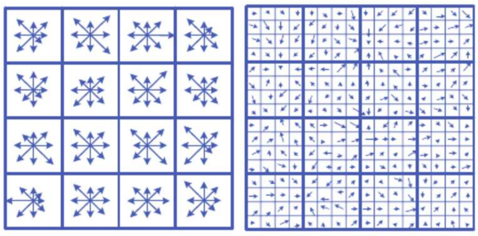


Fig: a) 4x4 descriptor (128 vector values) b)16x16 patch gradients

Descriptor Matching:

The SIFT descriptors of keypoints in two different images are compared to find matching keypoints. A common method is to use the Euclidean distance between the descriptors. Keypoints with the lowest distance are considered potential matches.

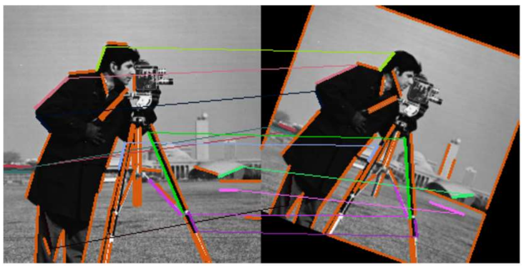
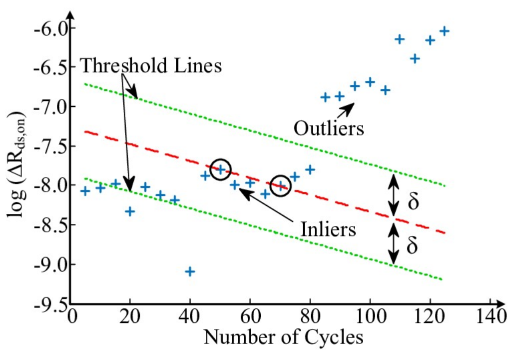


Fig: A sample of descriptor matching

Outlier Rejection using RANSAC:

To improve robustness against outliers, a technique such as Random Sample Consensus (RANSAC) is often employed. RANSAC helps in rejecting incorrect matches by iteratively fitting models to subsets of the data and identifying the most consistent set of matches.

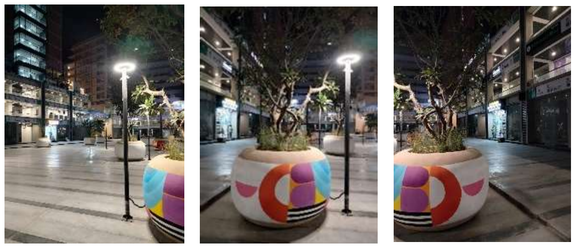


By following these steps, SIFT feature extraction provides a set of distinctive and invariant keypoints that can be used for various computer vision applications, such as image stitching, object recognition, and 3D reconstruction. The strength of SIFT lies in its ability to capture local image structures that are robust to changes in scale, rotation, and illumination.

III. SEQUENTIAL ELUCIDATION

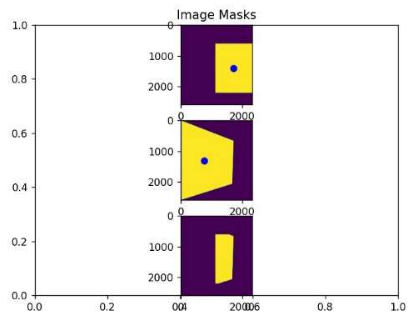
A step-by-step demonstration of the overall process of image stitching has been done on 3 images.

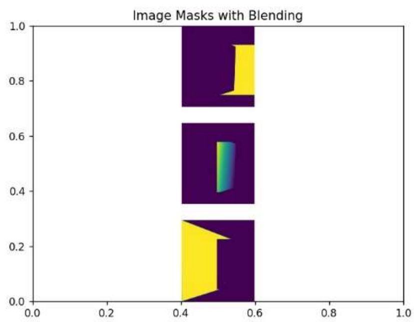
Original Images:



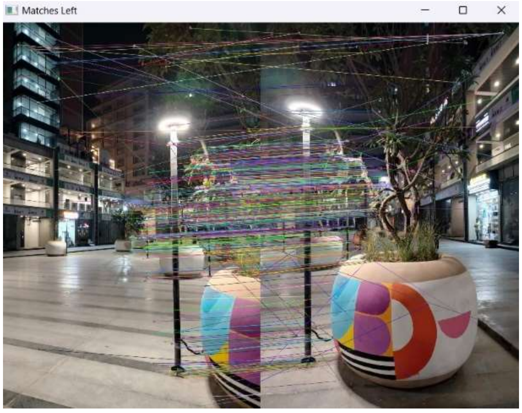
From the images, it can be assumed what the combination of all the three will look like.

Following this, the image stitching process starts. Initially the left and the center images are taken and matching is done between them, by first find the region of intersection, and then linear blending is done.

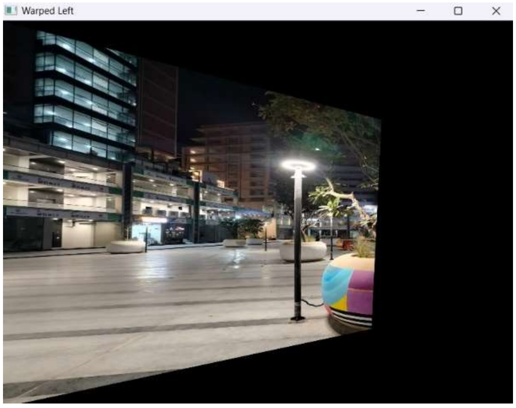




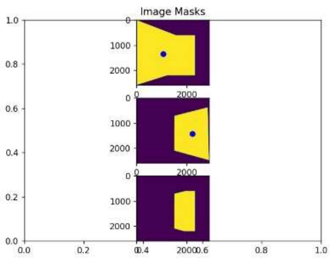
Following, the feature matching is done between the two images. It is evident, how a line joins to regions of the image like each other.

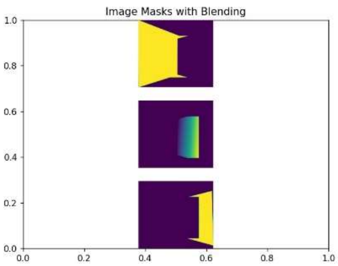


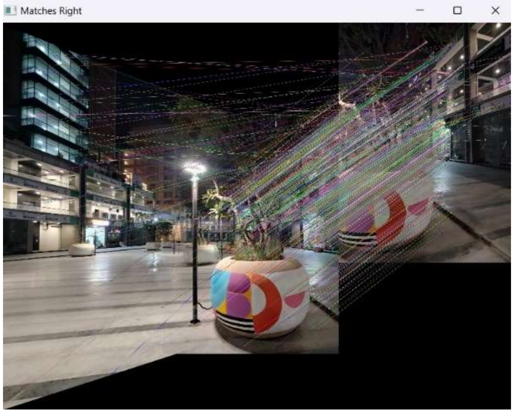
The left and center image gets stitched, and here is the output:



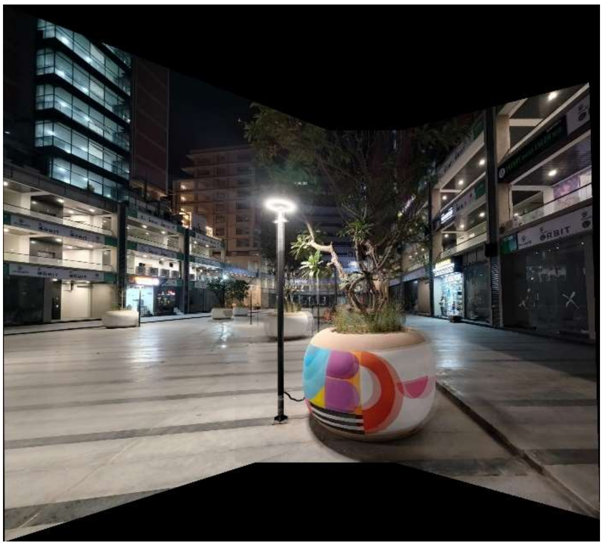
Now this serves as one image, which is to be stitched to the right side of the image. The steps remain the same.







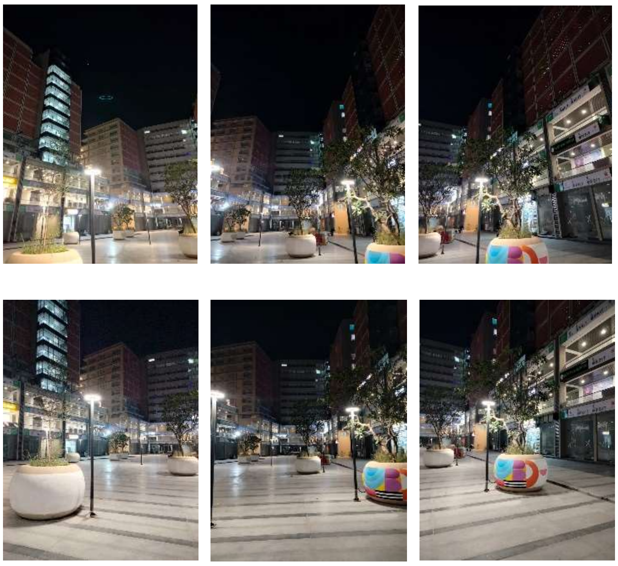
After the feature matching, the stitched image is ready, and here is the output:



The stitching has been done quite perfectly, although there are hazy portions which might be due to imperfect feature matching.

To further delve into the functionality, we have performed stitching of 6 images.

Input Image:



Output Image:

The six images have been stitched properly and the output has been generated which is quite accurate, if the stretching and squishing are considered.

IV. CONCLUSION

SIFT, or Scale-Invariant Feature Transform, emerges as a cornerstone in the intricate process of panorama stitching, demonstrating its multifaceted utility through various stages. Primarily, SIFT's role in feature matching stands out, as it adeptly identifies and matches key points across overlapping images, even accommodating variations in scale, rotation, or perspective. Beyond feature matching, SIFT significantly contributes to the subsequent alignment phase, where its prowess ensures the precise positioning of images. This alignment, crucial for spatial coherence, lays the foundation for the seamless blending of images, a pivotal step in crafting panoramic views. Notably, SIFT's inherent scale invariance and resilience to transformations amplify its effectiveness, proving particularly adept at stitching images taken from disparate angles or distances. In essence, SIFT's comprehensive capabilities bolster the robustness and reliability of the entire panorama stitching process, making it a linchpin in the synthesis of visually compelling panoramic views from diverse image sources.

V. REFERENCES

A. Singh, Aishwarya. "SIFT Algorithm: How to Use SIFT for Image Matching in Python (Updated 2023)." Analytics Vidhya, August 4, 2023. *https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/.*

B. More effective image matching with scale invariant feature transform. Accessed December 14, 2023. *https://www.researchgate.net/publication/268204633_More_effective_image_matching_with_Scale_Invariant_Feature_Transform.*

C. More effective image matching with scale invariant feature transform. Accessed December 14, 2023. *https://www.researchgate.net/publication/268204633_More_effective_image_matching_with_Scale_Invariant_Feature_Transform.*

D. Text generated by ChatGPT, December 1st, 2023, OpenAI, *https://chat.openai.com/chat. Edited for style and content.*

VIDEO DEMO LINK

https://youtu.be/HE8IjYGCNCs?si=QqnK3HYOjHHssRSY