CV Homework

Aqsa Shabbir
Department of Computer Engineering
Bilkent University
Ankara, Turkey
aqsa.shabbir@bilkent.edu.tr

Kousar Kousar
Department of Computer Engineering
Bilkent University
Ankara, Turkey
kousar.kousar@bilkent.edu.tr

***Abstract*—This report addresses two essential tasks in computer vision: image stitching and disparity map estimation, both widely applicable in areas like 3D reconstruction and panoramic imaging. For image stitching, we employ a pipeline that uses Scale-Invariant Feature Transform (SIFT) to detect and match feature points, followed by Random Sample Consensus (RANSAC) for robust homography estimation, aligning images accurately with minimal distortion. The blended panorama is refined with alpha blending to ensure seamless transitions across image boundaries. In the disparity map estimation task, we calculate pixel-wise disparities from stereo image pairs using a normalized correlation-based matching approach. Experimental results show that the stitching process produces coherent panoramas with smooth transitions, and disparity estimation achieves reliable depth mapping, closely matching ground truth data. This work demonstrates effective use of classical methods for accurate image alignment and depth estimation, providing insights for further enhancements in handling complex textures and low-contrast regions.**

I. Introduction

In recent years, the field of computer vision has advanced significantly, driven by increasing computational capabilities and the development of sophisticated algorithms for image analysis. Two essential techniques that have become fundamental in many computer vision applications are image stitching and disparity map estimation. These tasks underpin numerous practical applications, from panoramic photography and virtual reality to autonomous driving and 3D scene reconstruction. This report explores these techniques, implementing classical methods to create effective pipelines for both image stitching and depth estimation.

Image stitching is the process of aligning and blending multiple images to form a cohesive, larger composite image, typically a panorama. This task requires accurate detection and matching of corresponding feature points across images, precise estimation of transformation parameters to align these images, and smooth blending to eliminate visible seams. Stitching algorithms typically rely on a sequence of steps involving feature detection, feature matching, homography estimation, and blending. For feature detection and description, robust methods such as SIFT are often employed due to their ability to recognize keypoints that remain stable across transformations like scaling, rotation, and illumination changes. Once features are detected, feature matching methods identify corresponding points across image pairs. However, noise and inaccuracies in feature matching can lead to misalignment. To address this, homography estimation methods, often RANSAC, is employed to exclude outliers and achieve accurate image alignment. Finally, alpha blending is applied to ensure seamless transitions in overlapping areas, creating a visually cohesive panorama.

A study by Brown et al. [1] proposed an automatic panoramic image stitching method using multi-scale feature matching with SIFT descriptors to ensure high robustness across scenes with varying levels of detail and texture. Their work introduced a robust pipeline for aligning large sets of images. Depth estimation, on the other hand, is a technique used to extract depth information from pairs of stereo images. By calculating the disparity between corresponding pixels in the left and right images, it is possible to infer depth, as greater disparity generally indicates closer objects. Depth estimation has become critical in applications such as autonomous navigation, augmented reality, and robot vision, where an accurate understanding of the three-dimensional environment is essential. For this task, rectified stereo image pairs are typically used, simplifying the search for corresponding pixels to a single horizontal dimension. Classical methods for disparity estimation involve correlation-based matching, where pixel intensities are compared across small windows to find the best match.

The implementations of our homework are carried out using traditional computer vision techniques. For image stitching, we employ SIFT for feature detection and description, RANSAC for robust homography estimation, and alpha blending for smooth transitions. In the disparity map estimation task, we use normalized correlation-based matching with a predefined search range to compute dense disparity maps. Experimental results are evaluated for stitching and disparity estimation qualitatively by comparing generated maps with ground truth data.

The report is structured as follows: Section II presents Related Work. Section III discusses the methodologies, detailing each step in the image stitching and disparity estimation pipelines. Section IV presents and discusses the results. Finally, Section V concludes the report.

II. Related Work

In recent years, feature detection, feature matching, and model estimation advancements have significantly contributed to image stitching and disparity estimation tasks. This section reviews some studies related to image stitching and disparity map estimation techniques.

A. Image Stitching

Image stitching relies heavily on robust feature detection, matching, and seamless blending to create a coherent panoramic image from multiple overlapping views. Traditional methods often use the SIFT for feature detection, as its robustness to scale, rotation, and illumination changes makes it suitable for aligning diverse image sets [2]. However, improvements like PCA-SIFT [3] and CSIFT [4] add layers in robustness. PCA-SIFT achieves compactness and efficiency by reducing dimensionality with Principal Component Analysis, while CSIFT introduces color invariance to handle illumination changes, which is crucial in complex environments. To improve feature matching reliability, researchers have also enhanced RANSAC for model estimation. For instance, Shi et al. [5] propose pre-filtering steps to increase the inlier ratio, making transformation calculations more accurate and reducing computation time. Advanced stitching approaches, like multi-homography-based method presented by Wang et al. [6] address discontinuities by dividing images into blocks and computing local transformations.

B. Disparity Map Estimation

In stereo vision, accurate disparity map estimation is essential for depth perception, as it relies on high-precision pixel correspondence between stereo images. Traditional approaches often use correlation-based matching, yet achieving sub-pixel accuracy remains challenging, especially under varying lighting conditions. Psarakis et al. [7] introduced an Enhanced Normalized Cross-Correlation (ENCC) method that achieves sub-pixel accuracy by minimizing "pixel-locking" and enhancing resilience to photometric distortions. Advancements in disparity estimation contribute to more robust and computationally efficient stereo vision systems, making them capable of handling complex real-world variations in lighting and geometry.

III. Methodology

The methodology for this homework encompasses two main tasks: image stitching and disparity map estimation. In this section, we discuss these tasks in detail.

A. Image Stitching

The image stitching process is implemented as a multi-step pipeline involving feature detection, feature matching, homography estimation, and blending for seamless image integration. The details of image stitching is shown in Algorithm 1.

1) Feature Detection using SIFT: We employ the SIFT to detect distinctive keypoints in both images. SIFT is particularly effective because it detects key points that are invariant to scale and orientation, making it ideal for identifying corresponding points even in images with different perspectives or lighting conditions. The SIFT algorithm extracts descriptors for each keypoint, which are vectors representing local image patterns. These descriptors are later used for feature matching.

Algorithm 1 Image Stitching Algorithm

```
def MATCH_KEYPOINTS(desc1, desc2)
    return matches
def HOMOGRAPHY_RANSAC(src_pts, dst_pts)
    for number of epochs:
        homography using random point
        track highest inlier counts
    return homography
def BLEND_IMG(warped_img, img2, offset)
    blend overlapping region
    return blended img
def STITCH_IMGS(img1, img2)
    detect keypoints and desc
    src_pt, dst_pt ← MATCH_KEYPOINTS
    H ← HOMOGRAPHY_RANSAC(src_pt, dst_pt)
    stitched_img ← BLEND_IMG(warped_img, img2)
    return stitched_img
def MAIN()
    img1, img2 ← LOAD_IMG(img_path1, img_path2)
    stitched_img ← STITCH_IMG(img1, img2)
    return stitched_img
```

2) Feature Matching: To find corresponding keypoints between images, we compute the Euclidean distance between each pair of SIFT descriptors from the two images. Matches are initially proposed based on the nearest-neighbor distance. We use a ratio test to filter out unreliable matches. For each descriptor in the first image, the distance to the closest descriptor in the second image is compared to the distance to the second-closest descriptor. If the ratio of these distances is below a threshold (which is 0.75), the match is considered reliable and retained for further processing.

3) Homography Estimation with RANSAC: The homography matrix, which represents the geometric transformation between two images, is estimated using RANSAC. RANSAC is a robust fitting method that iteratively selects random subsets of matched keypoints, calculates a potential homography, and counts the inliers that fit this model within a defined threshold. This process is repeated for several iterations to maximize the number of inliers, thereby identifying the homography that best aligns the two images while ignoring outlier matches.

4) Image Warping and Blending: Using the estimated homography, one of the images is warped to align with the other. A warp function is implemented to transform image coordinates based on the computed homography matrix. To ensure a smooth transition between overlapping areas, alpha blending is applied. The overlapping regions are blended by gradually changing the opacity of pixels from one image to the other, resulting in a seamless integration. The final stitched image is then rendered with smooth boundaries between the two images.

B. Disparity Map Estimation

The disparity map estimation task uses a normalized correlation-based matching approach to compute the disparity between corresponding pixels in a stereo pair of images. This approach estimates depth by calculating pixel shifts between the left and right images in rectified stereo pairs. The details are explained in Algorithm 2.

Algorithm 2 Disparity Map Estimation

def CALC_DISP(*img1*, *img2*, *win*, *search*)
    For each (y, x) in *img1*:
        For each u in $[x - search, x + search]$:
            Error between patches at (x, y) and (u, y)
    return *disp*
def PROC_IMGS(*path1*, *path2*, *scale*, *win*)
    load imgs from *path1* and *path2*
    disp_l $\leftarrow$ CALC_DISP(*img1*, *img2*, *win*)
    disp_r $\leftarrow$ CALC_DISP(*img2*, *img1*, *win*)
    return *disp_l*, *disp_r*
def MAIN()
    disp_l, *disp_r* $\leftarrow$ PROC_IMGS(*path1*, *path2*)
    return *disp_l* and *disp_r*

1) Image Padding: To prevent boundary issues during patch extraction, both input images are padded based on the specified window size. Padding ensures that the sliding window approach can be applied uniformly across the entire image, including the edges.

2) Patch Generation and Normalization: For each pixel in the left image, a square patch of pixels centered at that pixel is extracted. This patch serves as a template that will be matched against patches in the corresponding row of the right image within a defined search range. Both the template patch (from the left image) and candidate patches (from the right image) are normalized by subtracting their respective means and dividing by their norms. This normalization mitigates the effects of lighting differences between the two images.

3) Pixel Matching using Sum of Squared Errors (SSE): The search range is defined as a maximum disparity, atmost 65 pixels. Within this range, for each pixel in the left image, we calculate the sum of squared errors (SSE) between the normalized template patch and each candidate patch in the right image. The disparity for each pixel is determined by finding the position in the right image that yields the minimum SSE, indicating the best match. The disparity is then stored in a disparity map, where each pixel value represents the horizontal displacement between matching points in the left and right images.

4) Disparity Map Visualization: The computed disparity values are scaled and mapped to a grayscale image for visualization, with brighter values indicating closer objects and darker values indicating greater distances. This disparity map provides a dense representation of depth information across the scene.

IV. Results & Discussion

The results and discussion for Image Stitching and Disparity Map Estimation are presented in this section.

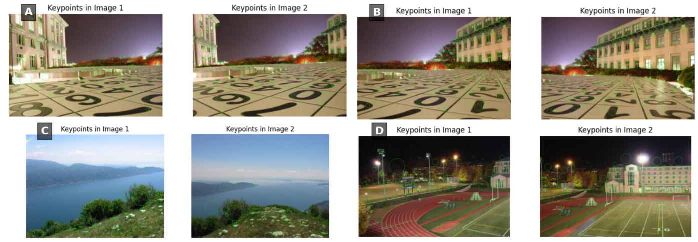

Fig. 1. Keypoint Detection

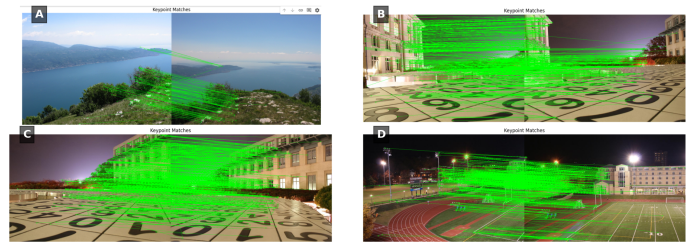

Fig. 2. Keypoints Matching

A. Image Stitching

1) Keypoint Detection and Matching: The algorithm detects distinct keypoints in each image using SIFT as shown in figure 1. The matching identifies corresponding keypoints across images shown in figure 2 based on descriptor similarity. The images show successful alignment in many parts suggesting accurate feature matching.

2) Homography Estimation with RANSAC: Homography matrices are estimated using RANSAC as shown in figure 3 to filter out outliers, represented in red. This approach ensures that the algorithm focuses on accurate matches, reducing the impact of outliers.

3) Image Stitching: We have employed alpha blending for smooth transitioning between images. Although blending reduces visible seams, differences in lighting and color around the transition areas are present in some images which is caused due to the differences in contrast where alpha blending alone cannot fully harmonize the images. The implementation of the warping function effectively maps image pixels according to the calculated homography matrix, aligning key regions accurately. However, minor distortions and misalignments are observed in some stitched images, particularly around the edges which occurs when scenes have depth or perspective differences. The final stitched images are shown in figure 4 , 5 , 6 and 7.

B. Disparity Map Estimation

The disparity maps generated for the ”cloth” and ”plastic” images illustrate the ability of the implemented stereo-matching algorithm to capture depth information. The results demonstrate the structural alignment of the objects, with

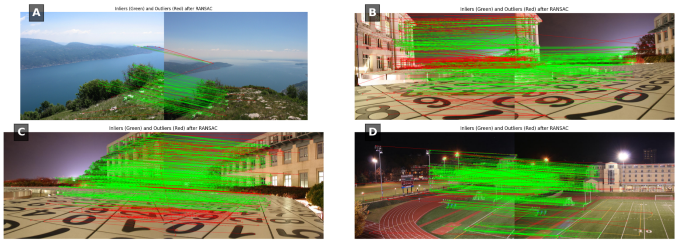

Fig. 3. RANSAC Implementation

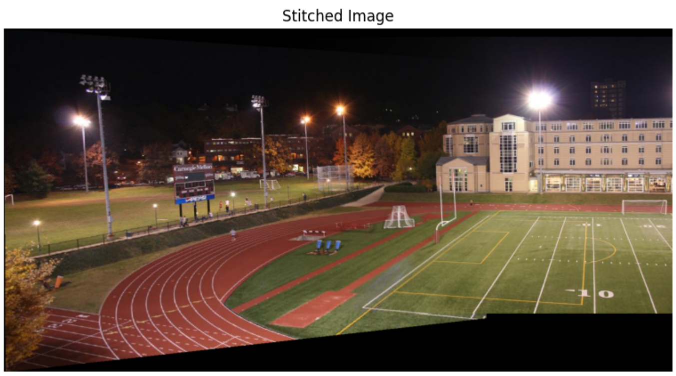

Fig. 4. Result for Image Stitching Pair 1

brighter regions indicating larger disparity values (closer objects) and darker regions showing smaller disparities (further objects). In textured regions, the disparity maps appear clearer, revealing depth variations more accurately. However, in areas with less texture or homogeneous surfaces, the algorithm introduces significant noise, resulting in artifacts across these regions.

In the ”cloth” disparity map 9, areas with distinct features yield reasonably accurate depth information, while smoother regions show some degree of noise and misalignment. The ”plastic” disparity map 8, on the other hand, is densely textured and exhibits high noise due to repetitive patterns, leading to more visible artifacts.

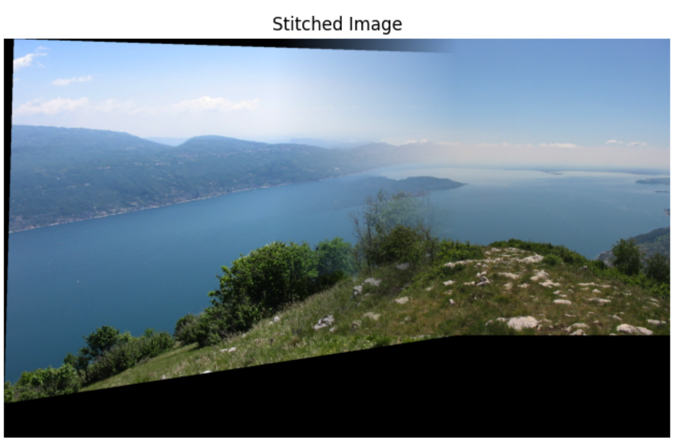

Fig. 5. Result for Image Stitching Pair 2

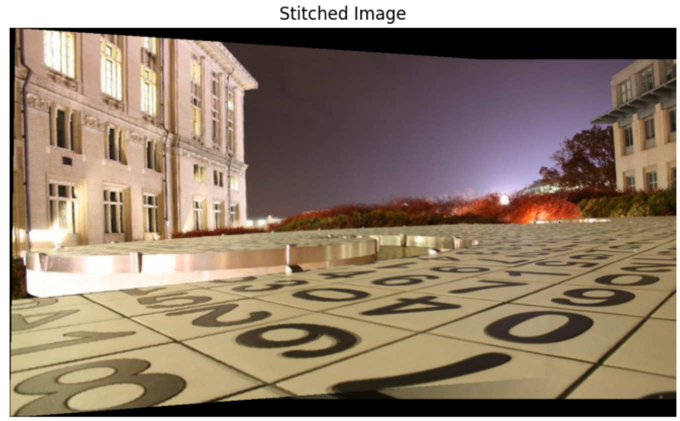

Fig. 6. Result for Image Stitching Pair 3

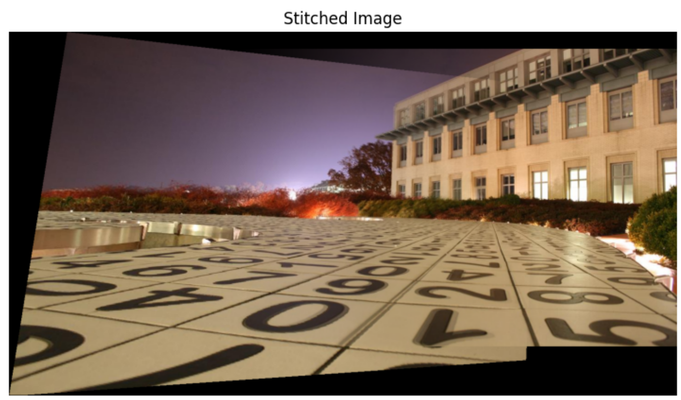

Fig. 7. Result for Image Stitching Pair 4

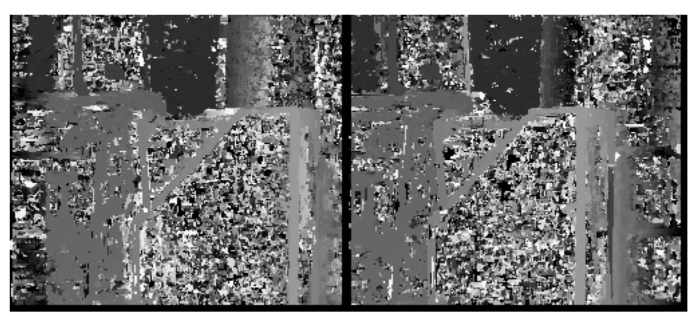

Fig. 8. Results for Disparity Plastic

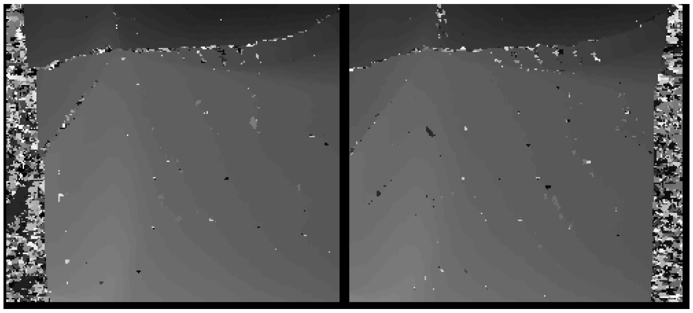

Fig. 9. Results for Disparity Cloth

V. Conclusion

The image stitching algorithm effectively combines multiple images into cohesive panoramas, successfully aligning and blending overlapping regions. While the implementation demonstrates strong alignment and blending capabilities, the limitations identified highlight opportunities for improvement in handling perspective variations, lighting discrepancies, and border artifacts. Furthermore, the implemented stereo-matching algorithm successfully generates disparity maps, providing a visual representation of depth for structured scenes. The results demonstrate that the algorithm performs well in capturing depth variations in textured areas with distinct patterns. However, the approach exhibits limitations in handling homogeneous and repetitive textures, resulting in noise and artifacts in the disparity maps.

References

[1] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International journal of computer vision*, vol. 74, pp. 59–73, 2007.

[2] A. Vedaldi, "An open implementation of the sift detector and descriptor," *UCLA CSD*, 2007.

[3] Y. Ke and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2, pp. II–II, IEEE, 2004.

[4] A. E. Abdel-Hakim and A. A. Farag, "Csift: A sift descriptor with color invariant characteristics," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2, pp. 1978–1983, Ieee, 2006.

[5] G. Shi, X. Xu, and Y. Dai, "Sift feature point matching based on improved ransac algorithm," in *2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, vol. 1, pp. 474–477, IEEE, 2013.

[6] W. Tengfeng, "Seamless stitching of panoramic image based on multiple homography matrix," in *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, pp. 2403–2407, IEEE, 2018.

[7] E. Z. Psarakis and G. D. Evangelidis, "An enhanced correlation-based method for stereo correspondence with subpixel accuracy," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 1, pp. 907–912, IEEE, 2005.