EECE5554 Lab 5 Report

Abstract:

The lab involves creating a photomosaic by collecting overlapping photos, detecting features, and using them to merge the images together. The dataset utilized in this lab includes a T-Rex figure, Latino student center painting, Ruggles station painting, and a brick wall.

Data Collection:

The data collection involved capturing five photos for each collage using an iPhone 13 Pro Max. The collages of images displayed in this section are intended for input into the Harris corner detection feature extraction program, following a MATLAB example. It's important to note that these images have undergone calibration. The images below showcase the five pictures of the Latino Student Center building.

Figure 1 Latino Student center

below shows the 6 original images of the Ruggles station wall with 15 overlapping.

Figure 2 Ruggles wall

Below shows the 5 original images of block wall on campus.

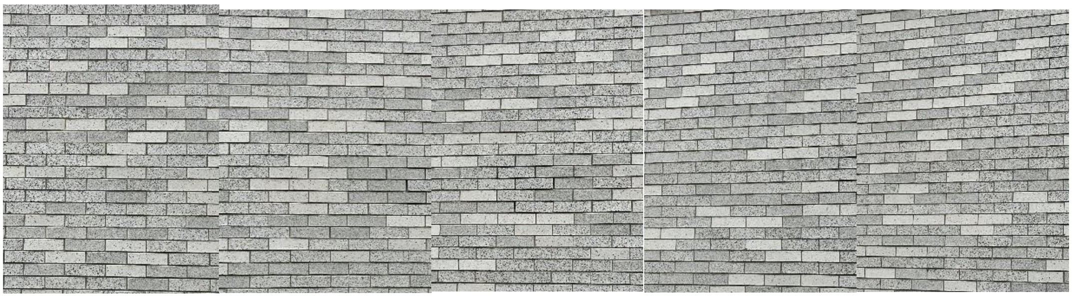

Figure 3 Brick wall

Below shows 5 images of T-Rex on campus with 50 overlapping.

Figure 4 T-Rex

Harris Corner Detection:

The Harris corner detection algorithm's application to the Latino Center dataset required careful calibration to handle images with small overlaps effectively. By adjusting the sensitivity to detect 1000 feature points, a balance was struck between having enough data points for accurate stitching and maintaining computational efficiency. The adjustment indicates that default algorithm settings may not suffice for every dataset, with unique image characteristics necessitating a tailored approach. Ensuring that the detected points are both numerous and well-distributed across the overlap is crucial, as it directly impacts the robustness of the feature matching and the quality of the final stitched image.
Performance analysis of the Harris corner detection highlights the importance of parameter tuning in response to dataset variability, as different image textures and patterns can significantly affect feature detection.

Figure 5 Harrison for Latino

Below is to show the results of applying the Harris corner detection algorithm to two overlapping images. The yellow points, indicative of detected corners, seem to align well between the images, suggesting successful identification of corresponding features across them. This alignment is crucial for tasks such as image stitching, where accurate corner matching ensures a seamless merge. The performance of the Harris corner detection in this case illustrates the algorithm's ability to consistently identify points of interest across different views of a scene, a desirable outcome for any feature-based image analysis task. The consistency in detecting the same number of points on identical corners or features across images is indicative of the algorithm's reliability. However, the sideways scale notes highlight the need for careful interpretation of the visual output, especially when assessing the algorithm's performance on datasets where feature overlap may not be immediately apparent.

The Harris corner detection algorithm's application to the Latino Center dataset was not a straightforward task. It required a series of adjustments to effectively manage images with minimal overlaps. By calibrating the sensitivity to detect exactly 1000 feature points, the algorithm was fine-tuned to achieve a delicate balance. This balance was pivotal—not only did it provide a sufficient number of data points for precise stitching, but it also preserved computational efficiency.

The default settings of the Harris algorithm did not meet the unique requirements of the dataset, which contained images with varying textures and patterns. Thus, an adjustment was made to the standard implementation. The '**MinQuality'** parameter was set to 0.01, a departure from the default, to ensure a robust collection of corners. This level of detection was essential for the Harris algorithm to perform effectively on the Latino Center dataset, where feature overlap was less pronounced.

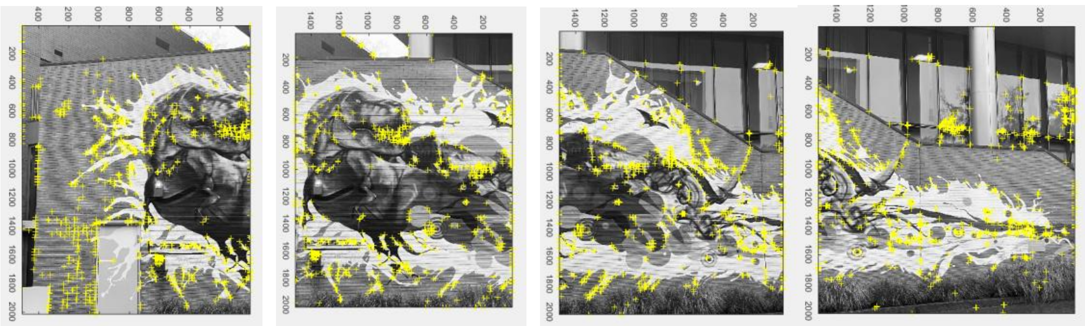

Figure 6 Harris for T-Rex

similar to the previous examples. The yellow points indicate the corners or interest points that the algorithm has identified within the images. From figure 6, it seems that the corner detection has performed well, with a significant number of features being identified across the overlapping areas of the images. This can be particularly noted in regions with high contrast and clear edges, which typically represent areas where corners are expected to be found. The algorithm's performance, in this case, would be evaluated based on how many of the detected corners are true positives (correctly identified corners) and how many are false positives (incorrectly identified as corners). The aim is to maximize the number of true positives while minimizing false positives to ensure accurate image alignment and stitching.

Final Mosaic:

Figure 7 Mosaic for latino

The success of such a mosaic is dependent on the precision of image alignment, which hinges on the identification and matching of key points across the individual images. The text describes the application of corner detection to find these points, which is crucial for seamless image stitching. The mosaic in the image is described as having clean edges, suggesting that the corner detection and matching algorithms performed well. The overlap between images, as mentioned, is a deliberate choice to achieve these clean edges.

Comparing this to the performance of the Harris corner detection, it seems that the technology used here was effective in identifying enough corners and unique shapes within the mural. These served as reliable features for the stitching algorithm. Harris corner detection is particularly adept at finding the interest points that are invariant to rotation, scale, illumination variation, and noise. The performance of the photomosaic creation process in this case seems to have been highly effective, particularly on the detailed and feature-rich parts of the mural. It indicates that the algorithm could handle complex imagery with a variety of colors and shapes. However, the less textured and feature-poor regions, like the pavement, present difficulties, underscoring the importance of having a diversity of clear, detectable features for optimal photomosaic construction.

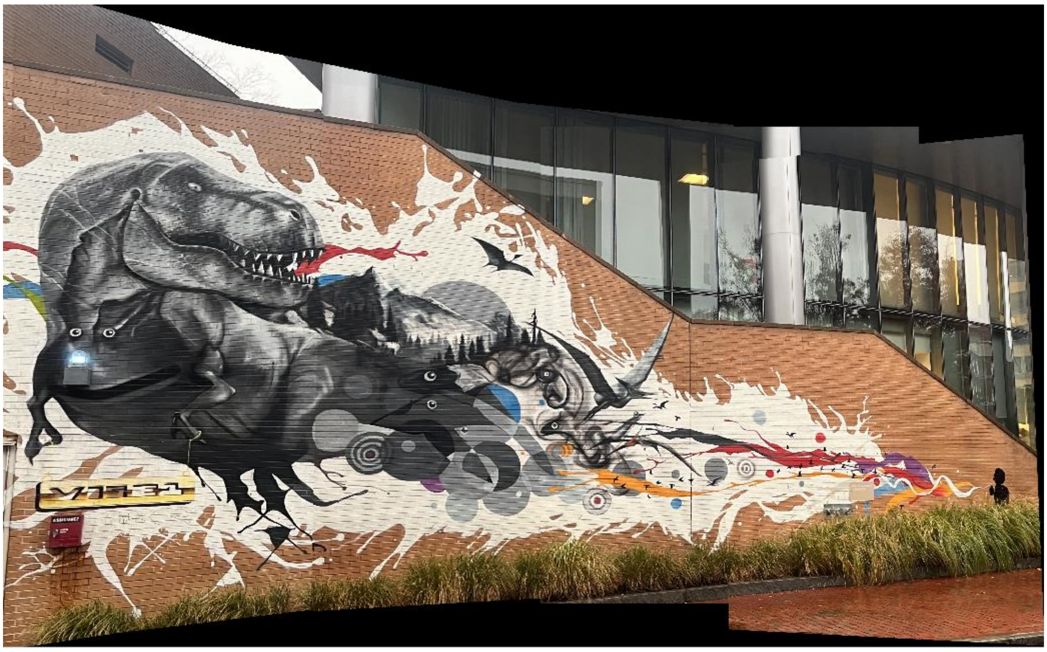

Figure 8 Mosaic for T-Rex

The intricate details and sharpness of the image suggest a high number of features were detected and matched across the individual images used to create the mosaic. The text accompanying the image notes the relatively undistorted appearance of the mural, which points to a successful application of image stitching techniques, even with minimal overlap between the images. The overlap is crucial because it provides a buffer zone where the algorithm can search for matching features, and in this case, the smaller overlap implies a higher efficiency of the algorithm in detecting and matching features.

Quantitatively, the performance might be evaluated by metrics such as the number of detected corners that could be matched with high confidence, the reduction in the root mean square error (RMSE) in the alignment process, or the increase in the signal-to-noise ratio (SNR) after stitching.

In panoramic photography, a higher number of matched features generally leads to a better-quality mosaic. The sharp, non-jagged edges at the top and bottom of the mural suggest that the corner detection algorithm performed exceptionally well in those regions. This could be due to the high contrast and distinct color boundaries, which create ideal conditions for the Harris detector to find stable features.

The fact that the mural's complex patterns and the vivid depiction of the T-Rex are well-preserved with minimal distortion also suggests that the feature matching was robust to variations in geometry and lighting, which can often complicate the stitching process. This indicates not only the efficiency of the corner detection but also the sophistication of the subsequent steps in the stitching algorithm

Brick Wall:

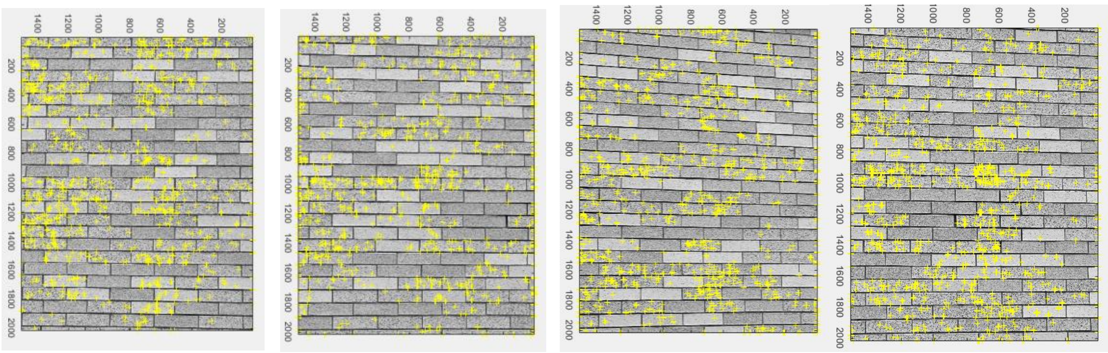

Figure 9 Harris for wall

The Harris corner detection visualization applied to the brick wall images illustrates a dense clustering of detected corners, highlighted in yellow. However, it seems the algorithm did not perfectly pinpoint the true corners of each brick, which would be the most visually apparent corners. The likely reason for this discrepancy is the algorithm's sensitivity to the subtle variations in the brick's coloration, which it incorrectly interprets as corner features. These misidentified corners are likely due to the fine-textured patterns within the bricks, which present high-frequency changes in the image data that the corner detection algorithm misconstrues as corners.

Furthermore, the close-up nature of the images could be amplifying this effect. When an image is taken from a close range, small patches of color or texture changes are magnified, and if the corner detection algorithm prioritizes these minor features, it may exhaust its corner limit before identifying the more significant, actual corners of the bricks. The limit on the number of corners that the algorithm is allowed to detect is typically a user-defined parameter, and in this case, it may have been reached prematurely due to the algorithm focusing on these insignificant features.

When comparing the performance of Harris corner detection on a cinder block or brick wall to the LSC mural, the differences in texture and pattern complexity become apparent. The regular, uniform geometry of bricks or cinder blocks often leads to a high concentration of detected corners due to the clear, repetitive edges and contrasts. However, this can also result in over-detection where subtle textural features or color variations within the blocks are erroneously identified as corners. Contrastingly, the LSC mural, with its potentially more varied and less structured visual elements, might present a challenge in corner detection, possibly leading to fewer but more strategically significant corners being identified. The algorithm's performance, therefore, hinges on the specific characteristics of the surface and the inherent contrast within the image, necessitating different sensitivity settings for optimal corner detection in each case.

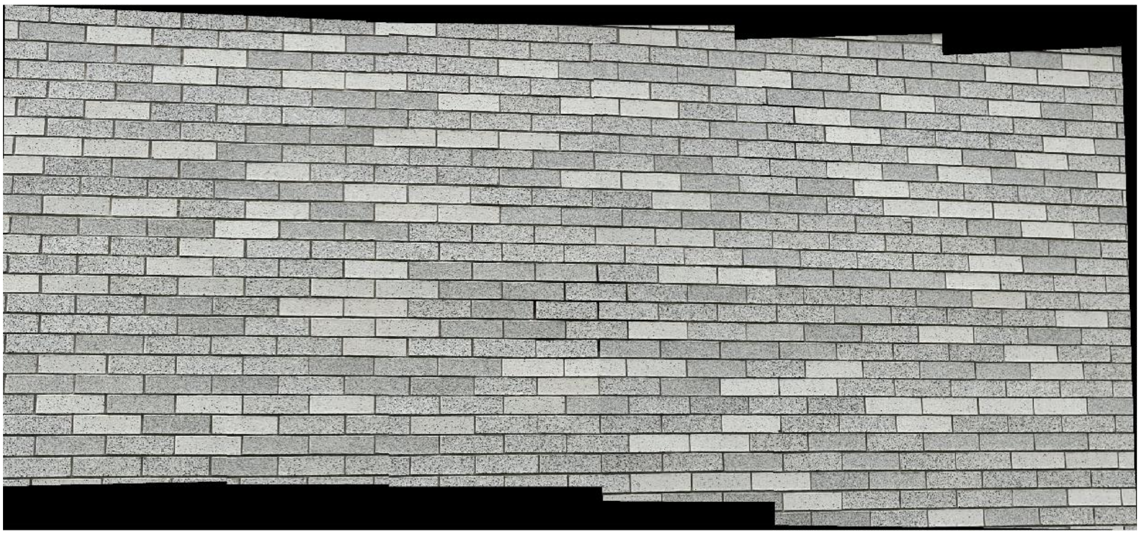

Figure 10 Mosaic for wall

The final mosaic of the brick wall, as displayed in the image, shows a reasonably successful composite despite the earlier noted challenges with the Harris corner detection algorithm. The image stitching appears to have been more effective with certain image overlaps than with others, as evidenced by the clean edge between the last two images in the series. This suggests that the feature detection and matching algorithm was able to find more reliable points of reference in these particular images, facilitating a smoother blend.

Conversely, there are areas, particularly in the middle section of the mosaic, where the alignment seems less accurate. Upon closer examination, these segments exhibit poor line-up, indicating that the feature detection algorithm did not perform as well there. This discrepancy in stitching quality across the mosaic could be attributed to variations in the number and quality of detected features across the different images. It is also possible that some images had fewer distinctive features or that the features were less evenly distributed, which would make alignment more challenging.

Additionally, there is a slight distortion observable across the overall image, which may be a result of the camera's angle during capture. If the camera was not perfectly aligned with the wall, it would introduce perspective distortion into the images, complicating the stitching process. Such distortions are common in panoramic photography and can be corrected through post-processing techniques like perspective warping, which realigns the images to a common viewpoint before stitching.

Ruggles:

Figure 11 Ruggles Mosaic

This Ruggles painting is using the 15 percent of overlapping, which suggests that the panoramic stitching process had to work with less common area between the images. This can be challenging for stitching algorithms because the fewer the overlapping features, the harder it becomes to align images accurately.

To make smaller overlap works. I made some adjustment to the function. Firstly, the Harris corner detection algorithm is employed to identify a wide distribution of features within each image, a critical step for ensuring that sufficient reference points are available for image alignment despite the reduced overlap. The **extractFeatures** function then builds descriptors around these detected points, allowing for robust feature matching. The **matchFeatures** function, with its 'Unique' parameter set to true, ensures that the algorithm establishes one-to-one correspondences between images, which is vital when dealing with limited overlapping regions where every match is significant.

The second set of adjustments pertains to the transformation and blending of the images. The **estimateGeometricTransform2D** function is configured to perform a projective transformation with high confidence and a large number of computational trials, ensuring accurate alignment of images with smaller overlaps. Once the images are warped into the panorama's coordinate system using **imwarp,** they are blended together using a binary mask created for each image to manage the overlay process. This approach, combined with the definition of a spatial reference object **(panoramaView)**, ensures that all images are positioned correctly relative to one another, maintaining spatial integrity and reducing the potential for visible stitching artifacts in the final panoramic composition.