Panorama Stitching Using Keypoint Descriptors

İlbey GÜLMEZ - 2210765024

AIN433 Assignment 2

Introduction

This report presents the approach and methodology for panorama image stitching using keypoint descriptor methods on SURF and ORB. Panorama stitching combines multiple overlapping images to create a single panorama, often achieved through keypoint detection, matching, and transformation operations. This project divides the process into several stages: feature extraction, feature matching, homography calculation, image transformation, and comparison against ground truth images.

1 Feature Extraction

Feature extraction is the first step in stitching images together. Key points are detected in each sub-image, providing a basis for identifying overlaps and aligning images. Two feature extraction methods Were used:

- **SURF**: This method detects keypoints that are rotation- and scale-invariant, making it well-suited for complex scenes with significant variance in scale and rotation. It leverages Haar wavelets and integral images to detect features quickly while maintaining robustness.
- **ORB**: ORB is designed for efficient performance on mobile devices and systems with limited computational resources. It combines FAST keypoint detection with the BRIEF descriptor, offering a computationally lightweight but less accurate option compared to SURF.

The `extract_and_match_features` function initializes these methods based on user selection. It then detects keypoints and computes descriptors, which capture the unique characteristics of each keypoint for matching in subsequent stages. But for some of the panoramas, both algorithms failed to extract keypoints.

2 Feature Matching

After extracting features, key points are matched across image pairs to identify areas of overlap. Matching is implemented using two approaches:

- **FLANN-Based Matching (for SURF)**: This method uses an approximate nearest-neighbor search to identify matches quickly, leveraging the KD-Tree for high-dimensional data like SURF descriptors.
- **Brute-Force Matching (for ORB)**: Matches are determined based on Hamming distance, suitable for ORB's binary descriptors. Brute-force matching checks every possible match, making it exhaustive but often slower.

For SURF, A ratio test is applied to filter high-quality matches and eliminate false positives. An optional visualization displays the matched points, allowing for visual validation of the feature-matching accuracy. Key points are stored as arrays for transformation in the following steps. For the combination of panoramas especially in the last step, both approaches were unable to find matches, causing SSIM and PSNR to decrease dramatically.

3 Finding Homography (RANSAC)

The homography matrix, which defines the spatial transformation between images, is calculated using the RANSAC (Random Sample Consensus) algorithm. RANSAC iteratively estimates the homography matrix by selecting random subsets of matches and optimizing the model based on minimizing re-projection error. The process includes:

- **Homography Calculation**: For each iteration, the homography is calculated using four randomly selected key points from the matches.
- **Inlier Selection**: Using the homography matrix, inliers are identified as points with low re-projection error, ensuring the selected points align accurately under transformation.
- **Optimal Homography Matrix**: After multiple iterations, the homography matrix with the highest number of inliers is chosen as the final model.

The `ransac` function iterates over potential matrices, ultimately returning the homography matrix with the best possible alignment, ensuring minimal distortion and maximal accuracy for stitching.

4 Merging by Transformation

Using the homography matrix, the images are warped and aligned into a cohesive panorama. This section includes:

- **Warping Sub-images**: The left image is transformed first, followed by the right, aligning coordinates and ensuring consistent overlap.
- **Blending**: Gaussian blending is applied in overlapping areas, using a gradient to merge the pixels smoothly and avoid visible seams between images.
- **Black Border Removal**: Any remaining black borders, which can occur from image transformation, are removed to produce a cleaner final panorama.

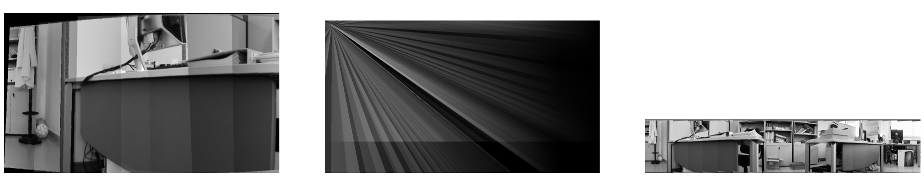
The `stitch_images` function performs these transformations, combining pairs of images until a single panoramic image is formed. This recursive merge-sort-like approach lets the stitching of multiple images in pairs.

5 Comparison with Ground Truth

To evaluate the quality of the stitched images, the generated panoramas are compared to ground truth images using two metrics:

- **PSNR (Peak Signal-to-Noise Ratio)**: A measure of signal fidelity, calculated by comparing pixel values between the constructed panorama and the ground truth. Higher PSNR values indicate closer similarity.
- **SSIM (Structural Similarity Index)**: A perceptual metric that assesses structural similarity, with values closer to 1 representing higher similarity to the ground truth.

6 Results Display



(a) SURF Panorama (b) ORB Panorama (c) Ground Truth

Figure 1: Comparison of SURF, ORB, and Ground Truth Panoramas

7 Results Summary

7.1 Performance Plots and Metrics

- **Feature Matching Visualizations**: Plots show key points and matching lines for each pair of sub-images, verifying correct matches.
- **Constructed Panorama vs. Ground Truth**: Each group displays the final panorama alongside its ground truth, allowing visual assessment of alignment accuracy.
- **Comparison Table**: PSNR and SSIM metrics, along with runtime for each method, are compiled into a table, showing the performance differences between SURF and ORB.

7.2 Observations

- **SURF vs. ORB**: Even though both algorithms didn't produce any meaningful result, SURF typically achieved higher PSNR and SSIM scores than ORB. However, its runtime is often longer due to the computational complexity of its descriptor calculations.
- **Performance Comments**: SURF's robustness to scale and rotation provides superior alignment in complex scenes, while ORB demonstrates efficiency for simpler images with lower accuracy requirements. This contrast reflects the trade-off between computational cost and accuracy.

8 Conclusion

In this assignment, an image-stitching system using SURF and ORB descriptors to construct panoramas from sub-images was developed. By extracting and matching features, calculating homography with RANSAC, and performing transformations, the solution couldn't generate panoramas comparable to ground truth images.

The experiments demonstrate that:

- **SURF** offers relatively better results but at a computational cost, making it ideal for high-quality stitching tasks where resources are not a limitation.
- **ORB** provides a faster, more efficient alternative, well-suited for real-time applications with limited resources, but at the expense of accurate panoramas.

Overall, the assignment highlights the importance of selecting the right feature descriptor based on the requirements of the task. SURF's performance in aligning images with significant scale and rotation differences suggests it is preferable for complex panorama stitching, while ORB may be preferred for simpler scenes or constrained environments. The results didn't align with expectations but still showed that balancing quality and speed is critical in practical applications of computer vision.

Future improvements could include:

- Exploring other blending techniques, such as multi-band blending, to further reduce visible seams in overlapping areas.
- Utilizing additional post-processing methods, like edge smoothing, to improve the visual quality of the final panorama.
- Integrating adaptive thresholding for matching in ORB to improve its reliability in complex scenes.

This assignment not only reinforces the foundational concepts of feature-based image alignment but also showcases practical challenges and trade-offs inherent in computer vision.