

# Enhancing Image Stitching Algorithms with SIFT Feature Detection

**Mingkai Wang**

School of computing, Neusoft Institute Guangdong, Foshan, China

jiangwanzheng@ldy.edu.rs

**Abstract.** Image stitching technology plays a significant role in the fields of computer vision and image processing, with applications ranging from panoramic photography to virtual reality (VR), augmented reality (AR), medical diagnostics, and autonomous vehicle technology. As technology advances, the demand for high-quality, real-time panoramic images provided by image stitching technology continues to grow. This study aims to implement an image stitching method based on the Scale-Invariant Feature Transform (SIFT) feature point detection algorithm, combined with the Random Sampling Consensus (RANSAC) algorithm and the calculation of the homography matrix to automatically stitch two images. This paper elaborates on the entire process of image feature extraction, feature matching, homography matrix calculation, and image fusion, and compares different fusion modes. The experimental results show that the method can achieve seamless image stitching in some cases, its performance in complex scenes such as crowds or traffic flows is average. This study provides new perspectives and methods for the application of image stitching technology.

**Keywords:** Image stitching, SIFT feature detection, RANSAC algorithm, homography matrix.

## 1. Introduction

Image stitching technology has shown its important application value in several fields in recent years. In virtual reality (VR) and augmented reality (AR), high-quality panoramic images can provide a more immersive experience, enabling users to perceive the virtual environment in a comprehensive way. Image stitching is a technique that combines several overlapping images (which may have been acquired at different times, from different viewpoints, or by different sensors) into one large, seamless, high-resolution image. When acquiring images of a scene with a wide field of view using an ordinary camera, the resolution of the image will be lower the larger the scene is because of the resolution of the camera. In the medical imaging field, accurate panoramic images can help doctors conduct more detailed diagnosis and surgery planning, and improve the quality of medical care [1]. In driverless technology, image stitching technology enables vehicles to fully perceive the surrounding environment and enhance driving safety. In addition, in cultural heritage protection and tourism, image stitching technology can reconstruct and display historical sites and scenic spots, providing a new way for cultural heritage protection and tourism promotion [2,3].

Image stitching technology has significant research value. Stitching technology involves many disciplines, which promotes the cross-fertilization of these fields. The development of image stitching technology promotes the innovation of algorithms, such as the optimization of feature point extraction,

image alignment, and fusion algorithms, and these technological advances are of great significance to the accuracy and efficiency of image processing. With the advancement of technology, stitching technology can be applied to more complex scenes, such as dynamic environments and image stitching under low-light conditions, and these challenges have driven the continuous improvement and breakthroughs of related technologies.

Currently, significant progress has been made in image stitching techniques. Researchers have achieved many results in feature point extraction (e.g., SIFT, SURF), image alignment (e.g., RANSAC), and image fusion (e.g., seamless cloning) [4,5]. Existing techniques still face challenges in dealing with some specific problems, such as real-time stitching in dynamic scenes, image quality enhancement in low-light environments, and computational efficiency when dealing with large-scale panoramic images. In addition, the existing methods have certain shortcomings in dealing with image distortion and fusion inhomogeneity.

This paper proposes a feature-point-matching-based image stitching method using the SIFT feature detection and RANSAC algorithm. By extracting and matching local features in images, combined with homography matrix calculation, automatic image stitching is achieved.

## 2. Related work

Image stitching technology plays an increasingly vital role in the fields of computer vision and image processing, with applications ranging from panoramic photography to VR, AR, medical diagnostics, and autonomous vehicles. As technology advances, the demand for real-time and high-quality image stitching technologies is growing. Researchers have explored a variety of methods to enhance the performance and effectiveness of image stitching.

The ORB algorithm is known for its speed and patent-free limitations and is commonly used in real-time image stitching tasks. Providing an efficient solution for image stitching, ORB combines FAST feature detection with BRIEF feature description. For example, ORB features are often used in combination with brute force matching or FLANN matching to achieve fast and accurate feature matching. To further improve the quality of stitching, some researchers have begun to explore the use of convolutional neural networks (CNNs) to learn more complex feature descriptions and matching relationships. The successful application of convolutional neural networks in image recognition and processing has made them a promising research direction in the field of image stitching. Self-attentive mechanisms (e.g., deformers) have also been used to optimize image stitching tasks. These methods focus on the critical parts of the image and improve the accuracy and visual effect of stitching.

In the field of 3D reconstruction, researchers have explored methods that combine multi-view geometry with image stitching to achieve higher-quality image synthesis. For example, using structured light methods for 3D reconstruction followed by image stitching can generate more realistic panoramic images. To address the issue of stitching artifacts caused by exposure inconsistencies, some studies have focused on image fusion and exposure compensation techniques. These studies adjust the brightness and contrast of the images, improving the visual continuity of the stitching area.

## 3. Method

In this study, a comprehensive method for image stitching is implemented using feature-based techniques. The process involves several stages: feature detection and description, feature matching, homography estimation, and image blending. The key steps are outlined below.

### 3.1. Feature detection and description

The feature detection is done using SIFT algorithm. SIFT is used to identify the key points in each image and calculate their corresponding descriptors [6]. The detectAndDescribe function utilizes the SIFT detector to extract key points and descriptors from the left and right images.

*3.1.1. SIFT Scale invariance:* SIFT is able to detect feature points at different scales and can effectively deal with scaling changes in images.

- **Rotation invariance:** The orientation of SIFT feature points is calculated based on the local gradient distribution, which makes it robust to image rotation.
- **Light invariance:** SIFT reduces the effect of light changes on feature point detection by operating on the image gradient.
- **Feature Descriptors:** The feature descriptors generated by SIFT are 128-dimensional vectors, which contain rich local image information and are used for more accurate feature matching.

3.1.2. *SIFT and ORB* While the ORB algorithm is computationally more efficient, SIFT performs better in terms of accuracy when dealing with complex images [4,6]. Here two algorithms, SIFT and ORB, are compared in Table 1.

**Table 1.** Comparison of SIFT and ORB algorithms

Characterization	SIFT	ORB
Feature Detection	By finding the extreme point (DoG)	Using the FAST algorithm
Computational Efficiency	High computational complexity and slow speed	Computationally efficient for real-time applications
Rotation Invariance	Provides good rotation invariance	Has rotation invariance, but may not be as good as SIFT under extreme conditions

### 3.2. Feature matching

Feature matching is achieved by comparing descriptors from both images. The matchKeyPoint function employs a nearest-neighbor approach to identify the best and second-best matches for each descriptor. The ratio test is applied to filter out weak matches, retaining only those that are consistently reliable. The function outputs well-matched keypoints between the images.

In feature matching, we find the best match by comparing the feature descriptors in image A and image B. The distance between the two feature descriptors is the Euclidean Distance. The most commonly used distance metric is Euclidean Distance, which represents the similarity between two feature descriptors.

Given two descriptors  $f_i$  and  $f_j$ , The Euclidean distance is calculated as:

$$d(f_i, f_j) = \text{sqrt} \left( \sum_{k=1}^n (f_{i,k} - f_{j,k})^2 \right) \quad (1)$$

### 3.3. Homography estimation

The homography matrix H is computed using the RANSAC algorithm to robustly estimate the transformation between the two images. The fitHomMat function performs random sampling to estimate H and identifies inliers that are consistent with the estimated homography. The matrix HHH facilitates the alignment of the images by mapping points from one image to the other.

### 3.4. Image blending

Image blending is conducted to create a seamless panorama by merging overlapping regions of the images. Two blending modes are implemented: linear blending and linear blending with a constant width transition. The warp function applies the computed homography to align the right image with the left image and then blends the images using the selected blending mode.

**Linear Blending:** This mode transitions the overlapping regions gradually based on their alpha mask. The linearBlending function computes the alpha values to smoothly blend the images in the overlapping area. Table 2 shows the implementation process of image stitching.

**Table 2.** Implementation process of image stitching

---

Read two images <code>img1</code> and <code>img2</code> .
Check if the images are successfully loaded. If not, print an error message
Call the <code>detectAndDescribe()</code> function to extract keypoints and feature descriptors from the two images:
Input: <code>img1</code> and <code>img2</code>
Output: <code>kps1</code> and <code>features1</code> (keypoints and feature descriptors from image 1), <code>kps2</code> and <code>features2</code> (keypoints and feature descriptors from image 2)
Call the <code>drawKeypoints()</code> function to draw keypoints on the images and save the result.
Input: <code>img1</code> , <code>img2</code> , <code>kps1</code> , <code>kps2</code>
Output: A visualization result with keypoints drawn on the images
Call the <code>matchKeyPoint()</code> function to compute matching points between the two images.
Input: <code>kps1</code> , <code>kps2</code> , <code>features1</code> , <code>features2</code> , <code>ratio</code>
Output: Matching keypoints ( <code>goodMatches</code> )
Call the <code>drawMatches()</code> function to draw the matching points and save the result.
Input: <code>img1</code> , <code>img2</code> , <code>goodMatches_ps</code>
Output: A visualization result of the matching points
Call the <code>fitHomMat()</code> function to compute the homography matrix <code>HHH</code> using the RANSAC algorithm and identify inliers.
Input: <code>goodMatches_ps</code> , <code>nIter</code> , <code>th</code>
Output: Best homography matrix <code>HHH</code> and inlier matching points ( <code>save_Inlier_ps</code> )
Print the calculated homography matrix ( <code>H</code> ) and the number of inliers.
Call the <code>drawMatches()</code> function again to draw the inlier matching points and save the result.
Input: <code>img1</code> , <code>img2</code> , <code>save_Inlier_ps</code>
Output: A visualization result of the inlier matching points
Call the <code>warp()</code> function to stitch the two images together using the computed homography matrix.
Input: <code>img1</code> , <code>img2</code> , <code>H</code> , <code>blending_mode</code>
Output: The stitched image ( <code>stitch_img</code> )
Adjust the window size and display the stitched image.
Save the stitched image result and wait for the user to close the window.

---

## 4. Experiment

### 4.1. Experiment setup

The platform configuration parameters used in this report are the compiler and version PyCharm 2024, using OpenCV in the Python language as a framework, with Python 3.11.9

The growing demand for artificial intelligence in computer vision has been devoted to the study of feature matching in computer vision and panoramic mosaics of VR images. Such algorithms for feature detection require easy and efficient performance, and feature matching needs to be done with a high degree of accuracy. The main data comes from the cell phone camera, which can capture the object by changing the angle of view. The main advantages of the cell phone are its convenience, low cost, and also the capability to handle camera distortion, large parallax, and brightness variations. The following figures (Figure 1, figure 2 and figure 3) show the selected dataset.



**Figure 1.** Sunset 001 (left) and 002 (right) (Photo/Picture credit : Original)



**Figure 2.** Beach 001 (left) and 002 (right) (Photo/Picture credit : Original)



**Figure 3.** Sea 001 (left) and 002 (right) (Photo/Picture credit : Original)

#### 4.2. Results and discussion

Calculate the number of inlier points used by the SIFT algorithm and ORB algorithm for three different sets of image stitching, and the results are shown in the table 3.

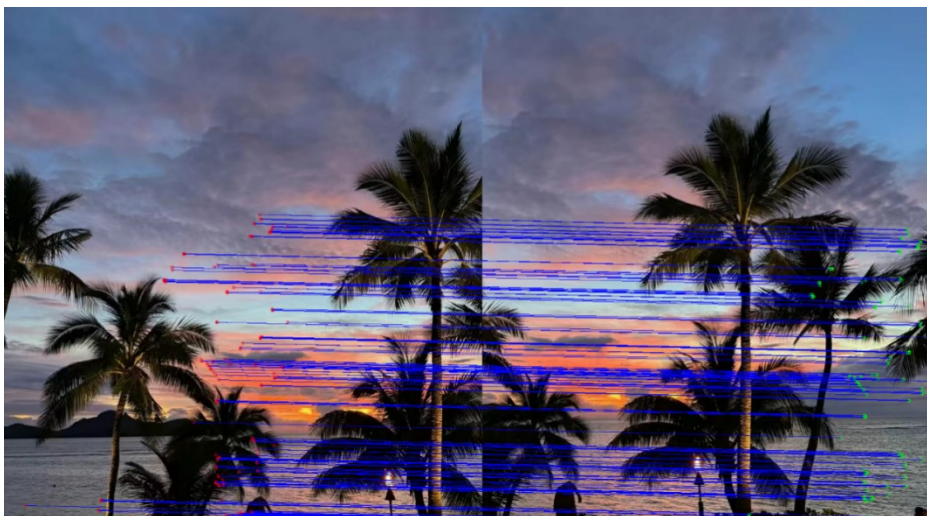
**Table 3.** Compares the demonstration images at the splicing and fusion time

	SIFT	ORB
Sunset	317	66
Beach	1743	224
Sea	2542	225

According to the table 3, it is known that the number of inlier points for both the SIFT algorithm is greater than the ORB algorithm. Although the ORB algorithm is faster in computation, SIFT is more suitable for scenarios that require high accuracy and processing of complex images [7,8]. The second set of images below visualizes the difference in the number of inlier points between the two algorithms (Figure 4 and figure 5).



**Figure 4.** Feature pint matching with SIFT (Photo/Picture credit : Original)



**Figure 5.** Feature pint matching with RD (Photo/Picture credit : Original)

In the spliced region of an image, the transition should be smooth, and there should be no noticeable abrupt changes in color, brightness, or texture. A good stitching algorithm should preserve the edge details in the image. If the edges of the spliced image are blurred or distortion occurs, the quality of the splicing is poor. It can be seen from the splicing results in Figures 6 and Figure 7 that the SIFT results have better edge details and excessive naturalness than the ORB results



**Figure 6.** Results f SIFT Sea Splicing (Photo/Picture credit : Original)



**Figure 7.** Results f RD Sea Splicing (Photo/Picture credit : Original)

## 5. Conclusion

This study presents an integrated approach to image stitching using SIFT and RANSAC algorithms. The main contributions of this research include efficient extraction of feature points, robust matching techniques and seamless image blending. The algorithms show significant improvement in stitching quality, especially in preserving edge details and reducing artifacts, compared to traditional methods such as the ORB algorithm. The results show that SIFT consistently identifies more outliers than the ORB algorithm, providing higher accuracy in complex scenes. The findings suggest that SIFT is

particularly advantageous in applications requiring high fidelity and detail, such as medical imaging and panoramic photography, where accurate feature matching is critical. Furthermore, this study acknowledges the limitations of existing algorithms when applied in dynamic environments and low-light conditions, where performance may degrade. As such, there is considerable potential for enhancing image stitching capabilities in real-time applications by exploring advanced feature extraction techniques and incorporating machine learning methods. Future work should focus on addressing these challenges, potentially leading to more resilient and adaptive image stitching solutions that can operate effectively across diverse scenarios and conditions. This research not only contributes to the field of computer vision but also lays the groundwork for future advancements in image processing technologies.

## References

- [1] Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91-110.
- [2] Mikolajczyk, K., & Schmid, C. (2005). A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10), 1615-1630. doi:10.1109/TPAMI.2005.188
- [3] Hartley, R. I., & Zisserman, A. (2003). Multiple View Geometry in Computer Vision. *Cambridge University Press*.
- [4] Fischler, M.A., & Bolles, R.J. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381-395.
- [5] Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding*, 110(3), 346-359.
- [6] Kaiser, P., & Schmid, C. (2011). Efficient SIFT matching with a multi-layer approach. *Image and Vision Computing*, 29(9), 645-654.
- [7] Brown, M.S., & Lowe, D.G. (2007). Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision*, 74(1), 59-73.
- [8] Zhang, Z. (1999). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1330-1334.