

# Feature Based Approaches for Homography Estimation

Samuel Venezia<sup>1</sup>, Sonya Coleman<sup>1</sup>, Dermot Kerr<sup>1</sup>, and John Fegan<sup>2</sup>

<sup>1</sup> School of Computing, Engineering and Intelligent Systems, Ulster University, Londonderry

<sup>2</sup> Metro Surveillance Group LTD, Cookstown

## Abstract

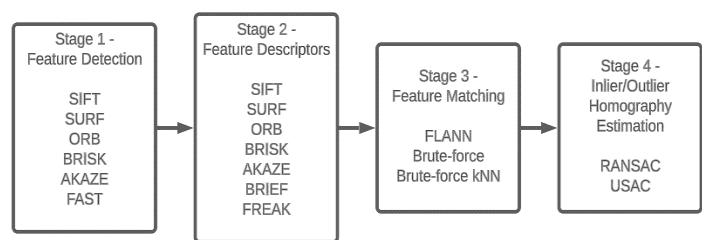
Image stitching is a method of producing a wider field of view by combining several overlapping images. With four main stages in the image stitching process, the algorithms used at each stage can have a dramatic impact on the success of stitching an image. For each stage, there are a wide range of algorithms to choose from and it can be a challenge to identify a stitching pipeline that will produce the best results. In this paper, we study the approaches involved in each of the four stages of image stitching. A real-world dataset is utilised to evaluate each algorithm, where images are transformed to different perspectives. The similarities of these images are compared to a warped perspective image obtained using the homographies provided by the dataset. The pipelines tested were limited to producing accurate results up to and including a 50° perspective change. Pipelines utilising BRISK's feature detector, FREAK, and Brute Force produced significant results. However, pipelines incorporating ORB, FAST, or BRIEF produce poor results when compared to other feature detection and feature description algorithms. Generally, the ratio test hindered the matched pairs process, although there were exceptions. Finally, the inlier/outlier detection algorithms, USAC and RANSAC, had similar performances with no definitive data to suggest that, in general, one outperforms the other.

**Keywords:** Image stitching, Perspective Warping, Feature Mapping, Homography estimation

## 1 Introduction

Computer Vision (CV) applications such as autonomous vehicles, security systems, and sports analytics benefit from an image with a wide field of view (FOV) and high resolution. However, obtaining an image with a wide FOV and high resolution can be a challenge. A wide-angle lens can replace a standard lens to fix this, however this distorts the image, reducing its quality. Moreover,

the image is still restricted to the original resolution of the capture device. Image stitching is a technique that produces panoramic images without the use of a wide-angle lens and can be done with either moving or static cameras. Merging overlapping images of a scene through image stitching provides a high-resolution image with a broad field of view without the distortions of wide-angle lenses. This provides a viable method for creating a wider FOV [Wang and Yang, 2020]. Feature-based approaches to image stitching include feature detection, feature description, feature matching, outlier removal, homography estimation, and transformation of pixels to the stitched image [Jakubović and Velagić, 2018]. This paper evaluates multiple algorithms across four stages of feature-based image stitching, as listed in Figure 1, determining the techniques that produce the highest quality results.



**Figure 1: Image Warping Pipeline and Algorithms**

## 1.1 Feature Detectors and Descriptors

Feature detectors search for distinctive visual components within an image. Figure 1's Stage 1 lists several feature detector algorithms. When matching features appear in multiple images of the same scene, they can be used to locate overlapping areas and stitch the images together. Feature descriptors represent the detected feature's local pixel neighbourhood. A unique description of a feature is provided by descriptors, enabling feature matching across different images. Stage 2 of Figure 1 outlines several feature descriptor algorithms.

A ubiquitous feature detection and descriptor approach is the Scale Invariant Feature Transform (SIFT) [Lowe, 2004]. The SIFT feature detector utilises an image pyramid representation to provide invariance to scale changes [Leutenegger et al., 2011]. SIFT identifies features by using the Difference of Gaussians (DoG) approach to detect gradient changes and the Hessian to reject edge points. The SIFT descriptor utilises a histogram of local and global gradients to describe the neighbourhood of pixels around each detected feature. While SIFT has been considered one of the benchmark approaches to feature detection and description it has a disadvantage in that it requires significant computation.

As an alternative the Speeded-Up Robust Features (SURF) approach [Bay et al., 2006] uses a different image representation technique to reduce the computation. Rather than using an image pyramid and applying DoG, SURF uses Haar wavelet-based image filters which are applied at different scales across the image. SURF also includes a feature descriptor utilising a Haar wavelet distribution.

KAZE is an alternative feature detector and descriptor which takes inspiration from SIFT and SURF [Leutenegger et al., 2011]. It utilises a similar process for detecting features as SIFT while using a homogeneous process for computing descriptors as SURF. A variant of KAZE, Accelerated-KAZE (AKAZE), reduces the time needed to detect the features within the pyramidal framework by implementing the Fast Explicit Diffusion framework [Alcantarilla et al., 2013].

The Features from Accelerated Segment Test (FAST) corner detector [Trajković and Hedley, 1998] is a fast but stable corner detection algorithm able to identify corners with high accuracy. The Binary Robust invariant scalable keypoints (BRISK) feature detector improves upon FAST [Leutenegger et al., 2011], providing the efficiency of FAST whilst being invariant to scale and rotation due to the use of a pyramidal structure similar to that of SIFT. The BRISK feature descriptor uses a deterministic sampling pattern to ensure uniform density around the keypoint (an especially distinctive feature often invariant to image transformations) and retrieve its direction to maintain rotational invariance [Leutenegger et al., 2011]. By using fewer sampling points than bitwise comparisons, the feature descriptor lowers the number of comparisons and complexity [Leutenegger et al., 2011].

The Binary Robust Independent Elementary Features (BRIEF) is an alternative feature descriptor with an emphasis on reducing the memory consumption of the algorithm [Calonder et al., 2010]. Descriptors are computed from images by directly comparing intensities of point pairs using intensity difference tests.

The improved variation of FAST, orientated FAST (oFAST), incorporated an orientation operator using the intensity centroid approach [Rosin, 1999]. This enabled more information detailing the keypoint's orientation to be utilised by the descriptor. The oFAST and Rotated Brief (ORB) algorithm is based on FAST and the BRIEF descriptor [Calonder et al., 2010]. Rotated BRIEF (rBRIEF) was proposed [Rublee et al., 2011] as a modified version of BRIEF, aiming to reduce its sensitivity to in-plane rotations.

The conclusion drawn from real-world data was that ORB had better performance than SIFT and sometimes SURF when compared [Rublee et al., 2011]. Fast Retina Keypoint (FREAK) is an alternative feature descriptor inspired by the human visual system [Alahi et al., 2012]. Like BRISK, FREAK utilises a retinal circular sampling grid. Compared to BRISK, FREAK has a higher density of points that are closer to the centre [Alahi et al., 2012].

## 1.2 Feature Matchers

The feature matching methods, seen in Stage 3 of Figure 1, utilise the descriptors generated from the 2<sup>nd</sup> stage feature descriptors as seen in Figure 1. With descriptors acting as a unique signature of each feature point, enabling points across both images to be compared and matched. The Fast Library for Approximate Nearest Neighbours

(FLANN) and Brute-force (BF) are feature matchers, both utilising Euclidean distance to measure the similarity between feature points on each image [Noble, 2016, Muja and Lowe, 2014]. The BF approach compares descriptors of all features in one image with those in another image [Noble, 2016]. While FLANN utilises approximations comparing few features by utilising the k-Nearest Neighbours (kNN) algorithm to produce pairs faster than BF but at the cost of reduced accuracy [Muja and Lowe, 2014]. Previous research has suggested utilising BF with the kNN ratio test [Jakubović and Velagić, 2018], reduces the number of possible matches in the event of multiple points competing for a match.

### 1.3 Outlier Removal for Homography Estimation

Homography is a 2-D perspective transformation that aims to map the pixels in a source image to a destination image and as such has a direct application in image stitching. Algorithms, such as Random Sample Consensus (RANSAC), Universal RANSAC (USAC), and Progressive Sample Consensus (PROSAC), are applied in Stage 4 as seen in Figure 1 for the identification of inlying matched pairs and the removal of outliers [Raguram et al., 2013]. RANSAC has been commonly employed for classifying between inliers and outliers [Caparas, 2020, Jakubović and Velagić, 2018, Tong et al., 2021]. USAC, an alternative to RANSAC, produced results that are more in line with the ground truth when compared to RANSAC and PROSAC, a substitute algorithm that favours speed over accuracy [Raguram et al., 2013]. Inliers are used to estimate homography, which generates a matrix for mapping an image to a different perspective [Sharma and Jain, 2020].

### 1.4 State-of-the-Art Approaches

AKAZE is considered a state-of-the-art feature detector and descriptor algorithm, outperforming SIFT, SURF, ORB, and BRISK using a variety of datasets [Sharma and Jain, 2020, Tong et al., 2021]. While FREAK is a state-of-the-art feature descriptor, outperforming the feature descriptors SIFT, SURF, and BRISK [Alahi et al., 2012]. Meanwhile, within the matching stage, BF is state-of-the-art, producing more accurate descriptor pairings than FLANN, while incorporating kNN increases BF's effectiveness [Noble, 2016, Caparas, 2020]. USAC is a state-of-the-art algorithm for removing outlying matched pairs, as it produced more accurate results compared with RANSAC and PROSAC [Raguram et al., 2013].

Multiple standard lens cameras with overlapping FOVs can be used for applications like sports analytics to create a single-perspective view and enhance resolution through image stitching. Fast image stitching is better achieved with feature-based methods instead of end-to-end deep learning approaches. The latter requires re-running the entire stitching process for each frame, due to its architecture [Yi et al., 2016].

## 2 Methodology

Our overall approach is based on defining an image stitching pipeline with multiple algorithms available for each stage. Our aim is to compare algorithms for each stage of image stitching to establish the most effective pipeline. Figure 1 shows this pipeline, with each stage having multiple algorithms. By utilising the feature detectors of Stage 1 such as SIFT, ORB, BRISK, AKAZE, and FAST, we can determine the most effective algorithm for detecting keypoints in each image. Due to licensing restrictions, testing SURF will not be possible, so it will be left out of the method and results. Comparing corner detectors, such as FAST with its variants (ORB and BRISK) to pyramidal-based algorithms like SIFT or AKAZE, will determine the better feature detector.

The feature descriptors of Stage 2 are compared against each other to determine the best method for creating descriptors given the keypoints identified in Stage 1. The feature descriptors of SIFT, ORB, BRISK, and AKAZE are compared alongside the stand-alone feature descriptors of BRIEF and FREAK. For example, one pipeline would

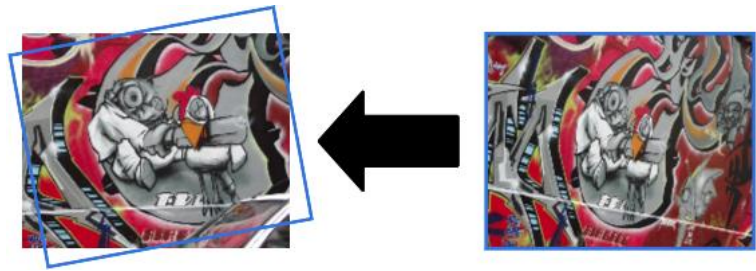
use the SIFT feature detector and FREAK descriptor, while another pipeline would use the SIFT feature detector and descriptor.

Stage 3 produces pairs of corresponding keypoints that are within the overlapping regions of the images to be stitched. Because FLANN performed poorly in the past, it was excluded from the data. The algorithms of Stage 3 comprise of BF and BF kNN. These two algorithms are compared against each other, to determine what influence the matching algorithm has on the success of a pipeline. When implementing BF kNN, the threshold is set to 0.75, testing the ratio of distance, enabling additional flexibility when determining the keypoints for matching in each image.

Stage 4 assesses two outlier elimination algorithms. More specifically, a comparison between RANSAC and USAC is being made. The resulting inlying matched pairs will be utilised to estimate the homography.

Due to the focus on perspective transformation, the Oxford Affine Covariant Regions Viewpoint Graffiti (OACRVG) dataset [Mikolajczyk and Schmid, 2005] is utilised to assess the performance of the different algorithms which make up the stitching pipeline. OACRVG is a standard test dataset utilised in multiple image stitching experiments focusing on perspective warping [Calonder et al., 2010, Leutenegger et al., 2011]. The dataset contains six images, each of size 800x640 pixels and includes homographies to map the first image with the other five images, treating image one as a pair with each subsequent image [Mikolajczyk and Schmid, 2005]. Additionally, there is a viewing angle change of 40° between images one and three, with an increase of 10° of change for each subsequent image [Mikolajczyk and Schmid, 2004].

Using four of the best inlier matched pairs from Stage 4 of the pipeline, the homography is estimated for each image in the OACRVG dataset being mapped onto image one. Estimating the homography requires the eight degrees of freedom afforded by the four matched pairs to compute the homography. The two images are mapped onto a 2D plane, to enable a common viewpoint between the images. Once the homography is estimated, every image is warped onto image one through a perspective transformation, as illustrated in Figure 2. The homographies provided by the OACRVG dataset were used to create the warped images, which can be seen in Figure 3. We aim to compare the pipelines mentioned and shown in Figure 1 to identify the ones that produce warped images of comparable quality to those produced by the homographies available in the dataset. Processing pipelines and algorithms are implemented using OpenCV 4.6.0 [Bradski, 2000] and Numpy 1.24.2 [Van Der Walt et al., 2011].



**Figure 2: Perspective Transformation of Image 3 to 1.**

Image #	Actual Image	Warped Image
2		
3		
4		
5		
6		

**Figure 3: Actual and Expected Warped Images, mapping images 2, 3, 4, 5 and 6 onto image 1 generated using the inverted homographies provided by the OACRVG dataset.**

### 3 Results





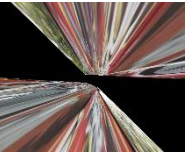
The Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Feature Similarity Index (FSIM) have previously been utilised to measure performance and concluded that while each method produced consistent results throughout, SSIM and FSIM are normalised and easier to interpret when compared with the absolute errors of MSE and PSNR [Sara et al., 2019]. PSNR and FSIM were chosen here to evaluate the warped images utilising the Image Similarity Measures toolbox [Müller et al., 2020].

Table 1 shows the mean FSIM and PSNR results for each pipeline on each image. FSIM is normalised between 0 and 1 where the higher the value, the more similar the images are [Sharma and Jain, 2020]. FSIM distinguishes complex parts of an image by detecting changes in light and gradient [Sara et al., 2019]. PSNR is the ratio calculated between the highest signal power and the power of the distorted noise [Sara et al., 2019]. Seven pipelines are among the top ten FSIM and PSNR from Table 1. Four of the seven pipelines that performed the best used BRISK, SIFT, and AKAZE detectors and descriptors, and three used BRISK and AKAZE detectors with a FREAK descriptor. Five of the pipelines employ BF, while two incorporate BF kNN as a matcher. RANSAC was utilised in three of the seven pipelines while USAC was utilised in four of the pipelines. While the best performing pipeline utilised USAC, RANSAC performed admirably, taking 2nd and 3rd place in Table 1 (ranked by PSNR).

#	Pipeline	FSIM ↑	PSNR ↑	#	Pipeline	FSIM	PSNR
1	BRISK – FREAK – BF – USAC	<b>0.604</b>	<b>93.605</b>	24	BRISK - BRIEF - BF - USAC	0.472	87.681
2	SIFT – SIFT – BF kNN – RANSAC	<b>0.581</b>	<b>93.326</b>	25	SIFT – FREAK – BF kNN – USAC	0.430	87.086
3	BRISK – FREAK – BF – RANSAC	<b>0.596</b>	<b>93.018</b>	26	SIFT – FREAK – BF – USAC	0.451	87.083
4	AKAZE – AKAZE – BF – USAC	<b>0.560</b>	<b>92.224</b>	26	ORB – BRIEF – BF – USAC	0.445	86.972
5	BRISK – BRISK – BF – RANSAC	<b>0.563</b>	<b>91.749</b>	28	SIFT – FREAK – BF kNN – RANSAC	0.426	86.970
6	AKAZE – FREAK – BF – RANSAC	0.512	<b>91.685</b>	29	BRISK – BRIEF – BF – RANSAC	0.421	86.392
7	SIFT – SIFT – BF kNN – USAC	<b>0.571</b>	<b>91.637</b>	30	AKAZE – BRIEF – BF – RANSAC	0.406	86.279
8	AKAZE – FREAK – BF – USAC	<b>0.556</b>	<b>91.194</b>	31	ORB – BRIEF – BF – RANSAC	0.416	85.945
9	BRISK – FREAK – BF kNN – RANSAC	0.532	<b>90.956</b>	32	SIFT – BRIEF – BF – USAC	0.382	85.405
10	AKAZE – AKAZE – BF – RANSAC	0.519	<b>90.891</b>	33	SIFT – BRIEF – BF – RANSAC	0.384	85.056
11	BRISK – FREAK – BF kNN – USAC	<b>0.547</b>	90.643	34	FAST – FREAK – BF – RANSAC	0.259	82.350
12	SIFT – SIFT – BF – USAC	<b>0.559</b>	90.640	35	BRISK – BRIEF – BF kNN – RANSAC	0.300	81.930
13	AKAZE – FREAK – BF kNN – RANSAC	0.498	90.232	36	FAST – BRIEF – BF – RANSAC	0.288	81.742
14	SIFT – SIFT – BF – RANSAC	<b>0.539</b>	90.208	37	FAST – FREAK – BF kNN – RANSAC	0.267	81.691
15	BRISK – BRISK – BF kNN – USAC	0.515	90.096	38	AKAZE – BRIEF – BF kNN – USAC	0.286	81.515
16	AKAZE – FREAK – BF kNN – USAC	0.504	89.717	39	BRISK – BRIEF – BF kNN – USAC	0.293	81.251
17	ORB – ORB – BF – USAC	0.481	89.459	40	SIFT – BRIEF – BF kNN – RANSAC	0.279	81.146
18	SIFT – FREAK – BF – RANSAC	0.489	88.821	41	AKAZE – BRIEF – BF kNN – RANSAC	0.276	81.085
19	ORB – FREAK – BF – USAC	0.504	88.745	42	FAST – FREAK – BF – USAC	0.264	81.037
20	ORB – ORB – BF – RANSAC	0.467	88.182	43	FAST – FREAK – BF kNN – USAC	0.259	80.427
21	BRISK – BF kNN – RANSAC	0.456	88.165	44	FAST – BRIEF – BF kNN – USAC	0.252	80.296
22	ORB – FREAK – BF – RANSAC	0.485	88.046	45	SIFT – BRIEF – BF kNN – USAC	0.254	80.188
23	AKAZE – BRIEF – BF – USAC	0.464	87.910	46	FAST – BRIEF – BF kNN – RANSAC	0.228	79.720

Table 1: Combined FSIM and PSNR mean results of the pipelines ranked by PSNR metric.

As shown in Table 2, the best pipeline from Table 1 was unable to produce a warped image to beyond image 4 (50° rotation), suggesting that, even though the pipeline scored the highest, there could potentially be a restriction in their ability to handle significant perspective transformations. In accordance with the conclusion from [Ruble et al., 2011], the simplicity of FAST hampers its performance. FAST is sub-par, producing the worst results, as it could not determine the orientation of a keypoint. ORB was the least successful out of the comprehensive feature detector and descriptors, with only two warped images generated using BF. This supports the conclusions of [Alahi et al., 2012], while disagreeing with [Ruble et al., 2011], with SIFT outperforming or at least performing at the same level as ORB. The pipelines incorporating BRISK for both feature detection and descriptors performed at the same level as ones that incorporated SIFT and AKAZE for both stages, with BRISK – BRISK – BF – RANSAC, falling within the top five of Table 1. This is in line with the conclusions of [Leutenegger et al., 2011], with BRISK as a feature descriptor performing competitively against SIFT. The BRISK feature detector approach outperforms other FAST corner detection-based algorithms in terms of PSNR and FSIM scores due to the scaling invariance BRISK provides. Four pipelines that use this version of FAST are among the top ten for FSIM and PSNR. This implies that BRISK's feature detector is the most skilled at generating superior keypoints, resulting in a broader image viewpoint.

Image #	2	3	4	5	6
Result					
PSNR	105.263	101.760	100.116	80.386	80.501
FSIM	0.868355	0.822749	0.800981	0.24983	0.276183
Pipeline	BRISK – FREAK – BF – USAC				

**Table 2: Comparison of the best method from Table 1 for each of the images based on the PSNR metric.**

As most of the pipelines involving SIFT and BRISK outperformed their AKAZE counterparts, these findings contrast with [Tong et al., 2021, Sharma and Jain, 2020]. Even though AKAZE yielded higher PSNR and FSIM scores compared to SIFT when BF and RANSAC were used in Stages 3 and 4, it was generally less effective than SIFT or BRISK-based pipelines, indicating that this is not the norm. When comparing BRIEF and FREAK, the dedicated feature descriptors, FREAK, produced higher PSNR and FSIM scores. FREAK is utilised in five of the top ten PSNR mean results, and also utilised in four of the FSIM results. BRIEF's poor performances can be traced to its inadequate performance with in-plane rotations. This result concurs with [Alahi et al., 2012], finding that FREAK outperformed the feature descriptors of BRISK, SIFT, AKAZE, and ORB.

Within the top ten FSIM and PSNR results in Table 1, seven utilised BF, compared with three using BF kNN. Finding fewer but more precisely matched pairs can reduce the possibility that a homography matrix is estimated. However, if there are over four matched pairs, but they are not matched correctly, the homography will be incorrectly calculated, producing an inaccurately warped image. This is not the case for all instances of kNN, with SIFT – SIFT – BF kNN – RANSAC producing a higher PSNR and FSIM score than its BF counterpart, with the former placing 2nd; while the latter performed worse placing 14th in Table 1. This suggests that kNN is preferable when enough features, with corresponding descriptors are detected.

Producing the highest similarity of FSIM and PSNR, BRISK – FREAK – BF – USAC suggests USAC minimised the number of matched pair outliers for estimating the homography, while its RANSAC equivalent pipeline placed 3rd with a slight decrease in overall performance. However, some pipelines utilising RANSAC outperformed their USAC counterparts, as shown with SIFT - SIFT - BF kNN – RANSAC and SIFT - SIFT - BF

kNN – RANSAC, with PSNR scores of 93.326 and 91.637, respectively. This disparity suggests neither algorithm outperforms the other, suggesting that they are of equal standing. The key factors affecting the performances of a warped image are the feature detectors, descriptors, and matching algorithms used in Stages 1, 2, and 3. Thus, it is recommended that pipelines with different algorithms be employed.

## 4 Conclusion

This paper compared the performances of image warping pipelines with 4 stages. The pipelines that used BRISK or SIFT in Stage 1 showed the best results for various perspectives using the OACRVG dataset. Using SIFT, BRISK, and AKAZE in Stages 1 and 2 yielded high PSNR and FSIM scores. Meanwhile, pipelines utilising FREAK perform similarly to or outperformed the other feature descriptors. Concluding that FREAK is a state-of-the-art feature descriptor, supporting the conclusion of [Alahi et al., 2012]. Pipelines with a ratio test (kNN) performed worse than matchers without, although exceptions exist. Furthermore, there is no definitive data to suggest that there is a significant performance increase in USAC compared to RANSAC that can be fully explained by the outlier removal approach, to the contrary of [Raguram et al., 2013]. Notwithstanding the accomplishments of the pipelines, the findings show they can only handle a maximum perspective change of 50°. This could be due to factors such as the dataset's resolution with a smaller overlapping region affecting the ability to warp the image. We intend to develop this research further by applying it to sports analytics. Further testing of the pipelines should involve the use of multiple datasets with different resolutions and environments, such as sports fields. Finally, DL approaches producing a homography matrix may provide an alternative solution to the pipelines discussed in this paper.

## Acknowledgements

This work was funded by a DfE CAST scholarship in collaboration with Metro Surveillance Group LTD.

## References

- [Alahi et al., 2012] Alahi, A., Ortiz, R., and Vandergheynst, P. (2012). FREAK: Fast retina keypoint.
- [Alcantarilla et al., 2013] Alcantarilla, P. F., Nuevo, J., & Bartoli, A. (2013). Fast explicit diffusion for accelerated features in nonlinear scale spaces.
- [Bay et al., 2006] Bay, H., Tuytelaars, T., and Van Gool, L. (2006). SURF: Speeded up robust features. Volume 3951 LNCS.
- [Bradski, 2000] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.
- [Calonder et al., 2010] Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). BRIEF: Binary robust independent elementary features. Volume 6314 LNCS.
- [Caparas, 2020] Caparas, A. (2020). Feature-based Automatic Image Stitching Using SIFT, KNN and RANSAC. International Journal of Advanced Trends in Computer Science and Engineering, 9(1.1 S I).
- [Jakubović and Velagić, 2018] Jakubović, A., and Velagić, J. (2018). Image feature matching and object detection using brute-force matchers. Volume 2018-September.
- [Leutenegger et al., 2011] Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). BRISK: Binary Robust invariant scalable keypoints.

- [Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2).
- [Mikolajczyk and Schmid, 2004] Mikolajczyk, K., and Schmid, C. (2004). Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1).
- [Mikolajczyk and Schmid, 2005] Mikolajczyk, K., and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10).
- [Muja and Lowe, 2014] Muja, M., and Lowe, D. G. (2014). Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11).
- [Müller et al., 2020] Müller, M. U., Ekhtiari, N., Almeida, R. M., and Rieke, C. (2020). Super-Resolution of Multispectral Satellite Images Using Convolutional Neural Networks. volume 5.
- [Noble, 2016] Noble, F. K. (2016). Comparison of OpenCV's feature detectors and feature matchers. In 2016 23rd International Conference on Mechatronics and Machine Vision in Practice (M2VIP), pages 1-6. IEEE.
- [Raguram et al., 2013] Raguram, R., Chum, O., Pollefeys, M., Matas, J., and Frahm, J. (2013). USAC: A Universal Framework for Random Sample Consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022-2038.
- [Rosin, 1999] Rosin, P. L. (1999). Measuring Corner Properties. *Computer Vision and Image Understanding*, 73(2).
- [Rublee et al., 2011] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF.
- [Sara et al., 2019] Sara, U., Akter, M., and Uddin, M. S. (2019). Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *Journal of Computer and Communications*, 07(03).
- [Sharma and Jain, 2020] Sharma, S. K., and Jain, K. (2020). Image Stitching using AKAZE Features. *Journal of the Indian Society of Remote Sensing*, 48(10).
- [Tong et al., 2021] Tong, C., Jianfeng, G., Xueli, X., and Jianxiang, X. (2021). High-precision Image Mosaic Algorithm Based on Adaptive Homography Transform. Volume 2021-July.
- [Trajković and Hedley, 1998] Trajković, M., and Hedley, M. (1998). Fast corner detection. *Image and Vision Computing*, 16[2], 75-87. ID:271526
- [Van Der Walt et al., 2011] Van Der Walt, S., Colbert, S. C., and Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. *Computing in Science and Engineering*, 13(2):22-30.
- [Wang and Yang, 2020] Wang, Z., and Yang, Z. (2020). Review on image-stitching techniques. *Multimedia Systems*, 26(4).
- [Yi et al., 2016] Yi, K. M., Trulls, E., Lepetit, V., and Fua, P. (2016). LIFT: Learned invariant feature transform. Volume 9910 LNCS.