A Comparative Analysis of Feature Detection and Matching Algorithms for Aerial Image Stitching

Mohd Ismail Jolhip[1], Jacey-Lynn Minoi[1,2], Terrin Lim[2]
[1]*Institute of Social Informatics and Technological Innovations, Universiti Malaysia Sarawak.*
[2]*Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak.*
ismail.jolhip@gmail.com

***Abstract*—Features detection and matching are the essential processes in image mosaicing and computer vision applications. Our work intend to find descriptors that are obtained by considering all interest/feature points and its locations on images, and then form a set of corresponding spatial relations based on the interest points between images. Hence in this paper, we will evaluate and present the performance of a few detector-descriptor-matcher approaches on raw aerial images for stitching image purposes. We have experimented on Canny Edge Detector, SIFT and SURF approaches to extract feature points. The extracted descriptors are then matched using FLANN based matcher. Finally, the RANSAC Homography is used to estimate the transformation model so stitching procedure could be applied in order to produce a mosaic aerial image. The results have shown that SURF approach outperforms the others in terms of its robustness of the method and higher speed in execution time.**

***Index Terms*—Image Stitching; Interest Points; Feature Detection; Feature Matching.**

I. INTRODUCTION

Feature detection, feature extraction and matching process are essential processes and at the base of many image processing and computer vision applications. Its applications could be used to align images for stitching, and for object recognition. There are numbers of feature detection and extraction algorithms that have been researched. The development of feature extraction tasks continues to grow exponentially to build sophisticated imaging applications. All these applications require the presence of robustness on the extracted feature points on the images. Hence, attempts to achieve highly reliable matching results from a pair of images is a challenge for the most of feature detection and matching algorithms.

The suitability of the feature extraction and stitching approaches depends on the types of the image. Generally, images are variant of scale, illumination, orientation, noise, transformation and blurring, hence extracting and determining the corresponding feature points on each image are challenging [4]. Various methods are researched and developed to robustly overcome all these variants. At presently, existing feature extraction approach has been compromised between the robustness and the execution time, but the fastest method with best results in all conditions has not been achieved.

Comparative studies have been done to evaluate the performance of the feature extraction and image matching algorithms to create a panoramic image [11] [13]. In our paper, we will be analysing and discussing the performance of a few methods employed on raw aerial images and these images were captured using drones over unconstrained environment. The aim of the paper is to evaluate the robustness and the efficiency of the cost of time on the different approaches on the images.

We proposed to study the commonly used method, Canny Edge detector, as the base comparison approach, and a couple of scale and rotation invariant extraction methods, i.e. the Scale Invariant Feature Transform (SIFT), and the Speeded-Up Robust Feature (SURF). SIFT is an efficient way to solve scale changes of images, and it has high robustness and location precision. SURF is a speed-up algorithm of SIFT.

Based on these feature extraction methods, we detailed the stitching by building the correspondences of a set of aerial images, establishing the corresponding points and then generate a panaromic image. We proposed the Fast Local Approximate Nearest Neighbors (FLANN) and Random Sample Consensus (RANSAC) techniques. The details of these extraction and fitting algorithms can be seen in the next section.

II. LITERATURE SURVEY

Given a set of images, a common approach for stitching typically consists of three steps: feature detection, feature extraction and image matching. First, images are selected, and keypoints, or salient points in images are detected. Second, the regions content are extracted and local descriptors are computed using feature extraction algorithms. Finally, the point correspondences are computed using image matching algorithms by overlapping regions between images [1-3] to perform a stitching task.

There are various image stitching frameworks that address the early feature extraction and matching algorithms. Most of the developed algorithms worked well under certain image conditions. Invariants present a typical problem in these algorithms for consistent, accurate and fast feature matching.

Some research focused on the use of feature extraction algorithms to automatically mosaic images by employing SIFT [4]. Researchers [5] presented a comparative study of using SIFT and SURF algorithms in image registration. The results presented that SIFT could detects more feature points while SURF perform faster than SIFT algorithms. Authors in [6] discussed the combination of feature detector-descriptor in indoor images and then compared the performance of the algorithms. [7] did a comparative analysis SIFT and the traditional photogrammetric feature extraction methods and matching metrics by experimenting tests on images acquired by drones.

The commonly used feature detection, Canny Edge detection algorithm, was used in object detection, image segmentation, image mining and face recognition [8] [9]. The performance presented when using Canny Edge and compared with Sobel edge detection suggested that Canny edge detection could perform much better for accuracy of the edges detection and faster execution of cost of time, although it is computationally expensive [11].

In face recognition applications, Canny Edge and Sobel edge detection algorithms were also employed by [10] to extract edges of face in face images. Their results have shown that the detected edges is a lot more accurate to Sobel's.

III. Overview of Feature Detection Algorithms

A. Canny Edge Detector

Canny Edge Detector is a multi-stages operator used to detects interest points of edges in images with noise suppressed at the same time. The creator formalised the problem of edge detection following the list of criteria in order to improve the algorithm: low error rate that is led to accurate detection, localised edge points and single constraint response [12].

Canny Edge detection runs in 5 stages. First, Canny operator perform smoothing on an image to reduce noise by blurring the image using Gaussian filter. Here is an example of 5x5 Gaussian filter matrix with σ = 1.4 of a smoothing method.

$$B = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * A \tag{1}$$

Second, it finds the intensity (magnitude and direction) gradient of the image where the gradients of the image that has a large intensity will be marked as edge. Third, the operator then applies a non-maximum suppression to thin or erode the edges and only local maxima will be marked as the edges and the outcome is a binary image. The edge and gradient can be determined by using the following equation 2.

$$G = \sqrt{{G_x}^2 + {G_y}^2} \tag{2}$$

Fourth, the angle is determined:

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right) \tag{3}$$

Double thresholding algorithm is applied to determine the potential edges. Finally, hysteresis is used for edge tracking and to form the final and continuous edges by suppressing all unconnected edges with strong edges [13].

B. Scale Invariant Feature Transform – SIFT

The Scale Invariant Feature Transform (SIFT) was first presented by David G. Lowe in 1999 [14]. SIFT is proven to be invariant of rotation, scale, illumination and even a certain degree of changes in viewpoint. The SIFT descriptors' computation involves four main steps: scale-space extrema detection, feature point localisation, orientation assignment, and feature descriptors [15].

The first step of SIFT operator is to find potential interest points in image using Difference of Gaussian (DoG) scale-space filtering operators instead of Gaussian to improve the computation speed.

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - (G(x,y,\sigma) * I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{4}$$

The * is the convolution operator, $G(x,y,\sigma)$ is a Gaussian variable scale, $I(x,y)$ is the image $D(x,y,\sigma)$ is Difference of Gaussians with k times scale.

In the feature point localisation step, a low contrast local extrema (interest points) is rejected and edges due to noise are discarded [15]. While in the orientation step, the region content SIFT feature location is determined and an orientation histogram is formed from the gradient orientations of sample points within a region around the feature point. This shows that the local descriptors are robust to rotation. Multiple keypoints are created at the same region location and within the same scale, but with different directions and orientations. Final step, descriptor of feature point is computed and SIFT descriptors are measured from the orientation histograms. These descriptors are invariant of the orientation and illumination.

C. Speeded-Up Robust Features – SURF

Speeded-Up Robust Features (SURF) was introduced by Herbert Bay [16]. Their experiment showed that the performance of this method works well and was faster. Similar to SIFT, SURF algorithm also consists of two stages: feature point detection, and feature point description. SURF was formulated to ensure high speed in three steps of feature detections: feature point detection, feature point description, and feature point matching.

Unlike SIFT, which builds an image pyramids and filtering each of the layer by using DoG and taking the difference, SURF on the other hand creates a "stack" without 2:1 down sampling for higher levels in the pyramid. Generally, SURF involves three steps: establishing integral image, building scale-space image, and positioning feature points [17]. SURF detects feature points using Hessian Matrix approximation where the Hessian determinant is maximum gives the location of the feature point. For an image I with a given point P =(x,y), the Hessian matrix H = (P,σ) in P at scale σ is defined as:

$$H(P,\sigma) = \begin{bmatrix} L_{xx}(P,\sigma) & L_{xy}(P,\sigma) \\ L_{yx}(P,\sigma) & L_{yy}(P,\sigma) \end{bmatrix} \tag{5}$$

where $L_{xx}(P,\sigma)$ is a convolution of the image with second order derivatives of the Gaussian.

Using the integral images, SURF filters the "stack" using a box of filter of second-order Gaussian partial derivatives (Octave), and integral images allow the computation of rectangular box filters to be fast. The SURF scale space is built by maintaining the same image and changing the box filter size. The determinant of the Hessian matrix is denoted as:

$$Det(H_{approx}) = D_{xx}..D_{xy} - (0.9D_{yy})^2 \tag{6}$$

where 0.9 is a general weighting factor. The threshold value of the calculation's result is set, and the extreme points value has to be greater than threshold. All the pixels will be compared to its eight neighbour pixels of the same scale and nine pixels of the adjacent upper and lower scale, then we will obtain local maximum points which are marked as feature points.

The SURF descriptor is based on similar properties as in SIFT. First is to determine the orientation by adding Haar response in x and y directions. Then, construct a square region in the center of the feature point aligned to the selected orientation. The wavelet responses are invariant towards illumination and to ensure invariance to contrast, change the descriptor into unit vector. The focus principal of SURF is to speed-up matching step [18].

IV. Experiments

The aerial images data is drawn from our database of 200 images and each image is not preprocessed and they are taken over a period of time. The images were captured using 14.2 Megapixels digital camera Sony NEX-5 with 18-55mm lens and attached beneath a drone. We flew the eBee drone. For this experiment, we have pre-processed the aerial image by downscaling the size of the image from 4000 x 2658 pixel resolutions to 600 x 399 pixel resolutions. This stage is required as the proposed methods are unable to handle the different dimension and pixel resolution of the images. Four pairs of images (see Figure 1 (a-d)) are selected based on the scenery, illuminations, objects and colour variability. The size and resolution of the images are set as shown in Table 1 and each are stored in JPEGS format.

Table 1
The size and resolution of pair of aerial images

Image Pair	Dimension	Resolution
Pair 1	600 x 399	100
Pair 2	600 x 399	100
Pair 3	600 x 399	100
Pair 4	600 x 399	100

The experimental test employs the Open Source implementations of the Canny Edge detector, SIFT and SURF detector and descriptor algorithms that are available and widely used for researchers. All of the algorithms are written in three different image stitching programs and executed to compare the efficiencies of the different methods required in the stitching and the workflow process.

The experiments are carried out on a 2.30 GHz CPU and 4GB system memory in Windows environment computer.

(a) First pair of the aerial images

(b) Second pair of the aerial images

(c) Third pair of the aerial images

(d) Forth pair of the aerial images

Figure 1: Four pairs of aerial images to be used for evaluations

The detector-descriptor-matcher approach that we employed in this experiment follows the existing image extraction and matching workflow and process. The following experiments are organised in three main methods and the algorithms are tested on each pair of the images.

A. *Method 1*

The first method is implemented with a series of function modules to carry out the workflow. The steps of the workflow follow these orders.

a. Image acquisition

A pair of images is acquired from the existing images stored in a folder of the executed program directory.

b. Edge detection using Canny Edge Detector

Canny edge operator is used to detect the edge of images separately.

c. Features detection and description using SIFT and SURF

The edges computed by Canny operator are used to find the feature points and its descriptors based on the edges of the image.

d. Matching keypoints using FLANN

This method is a fast local approximate nearest neighbors (FLANN) calculation between two sets of feature points in the images. It matches keypoints found between images and eliminate any erroneous match keypoints.

e. RANSAC translation estimation

Given the features in two images of the same scene, the corresponding features of the two images is measured and the translation between images is estimated.

f. Image stitching

After the corresponding features have been found and the translation between images are measured, the two images are then stitched together accordingly. When the two images are translated, the dimension area of a new canvas is created and calculated by masking the two images and estimating the overlapping areas.

B. Method 2

The workflow of the second method consists of steps that execute in the following order.

a. Image acquisition

The pair of images is acquired from the existing images stored in a folder of the executed program directory.

b. Feature detection and extraction using SIFT

SIFT features are detected and extracted from the acquired images and each feature gives an output in the form of matrix. And the properties of the feature points are as follows: 1) Coordinates of the keypoint. 2) Size of the keypoint. 3) Orientation angle of the keypoint. 4) The response by which the strongest keypoints have been selected and 5) Octave, from which pyramid layer the keypoint has been extracted.

c. Feature matching using FLANN

The extracted SIFT features from images then matched using the FLANN algorithm, and the procedure is as described in Method 1 step 3.

d. RANSAC translation estimation

The translation between the images is estimated using the RANSAC iteration method (steps as described in Method 1 step 4).

e. Image stitching

After the corresponding features have been found and the translation between images are estimated, the two images are stitched together. The two images are translated, the dimension area of a new canvas is created and calculated by masking the two images and estimating the overlapping areas.

C. Method 3

The third method is implemented using similar environment as in Method 1 and Method 2. The steps of the workflow are in the following order.

a. Image acquisition

Images are acquired from the existing images database stored in a folder of the executed program directory.

b. Feature detection and extraction using SURF

The SURF features are detected and extracted on the acquired images at distinctive location and the neighborhood of each keypoint is represented by a feature vector. The processes involve in SURF detection and description are as follows: 1) Establishing integral image to accelerate convolution between original images and box filters in the process of feature detection. 2) Building a scale space to locate the feature points in the image and 3) Detecting and obtaining extreme or local maximum point of the images by using the fast Hessian matrix on each level of the image scale space.

c. Matching keypoints using FLANN

The extracted SURF features are then matched using the FLANN algorithm as described in Method 1 step 3.

d. RANSAC translation estimation

The translation between images is estimated using the RANSAC iteration algorithm.

e. Image stitching

Once the corresponding features have been measured and the translation between images are estimated and then the two images are stitched together. When the two images are translated, the dimension area of a new canvas is created and calculated by masking the two images and estimating the overlapping areas.

V. EXPERIMENTAL RESULTS

For the purpose of this paper, three parameters were examined for a comparative analysis of the selected feature descriptors for image stitching based on the: (a) number of feature points detected, (b) number of good matches (accuracy), and (c) processing time.

The first part of the experiment shows the numbers of feature points detected for each method. Table 2 shows the number of feature points detected from the combination of detector and descriptor. It shows that the combination of CANNY-SURF could detect more number of feature points than the other combinations of the employed methods. All of the detected feature points in images are later filtered using the FLANN based algorithm with aim to eliminate any mismatch feature points in both images to order to obtain potential true feature points.

Table 2
The number of detected feature points from the detector-descriptor.

	Pair 1		Pair 2		Pair 3		Pair 4	
	img1	img2	img1	img2	img1	img2	img1	img2
SIFT-SIFT	3095	3702	3735	3095	2988	1923	2161	1951
SURF-SURF	2766	2820	2769	2766	2531	2474	2388	2487
CANNY-SIFT	3000	3572	3414	3000	2326	1319	1816	1577
CANNY-SURF	4426	4501	4658	4426	3951	2773	4108	4018

Table 3 shows the execution time or cost of time of processing feature extraction methods. We have found that the SURF algorithm is a better feature extraction method when used as detector and descriptor than combining it with Canny method. Given that SURF is the fastest method, hence the combination of CANNY-SURF as detector and descriptor also would give a good performance in term of speed in feature detection and description.

Table 3
Experimental result of detection time to detect feature points

	Pair 1	Pair 2	Pair 3	Pair 4
SIFT-SIFT	1.9706	1.9136	1.7093	1.6208
SURF-SURF	0.6287	0.6077	0.5278	0.6448
CANNY-SIFT	1.856	1.8723	1.5985	1.4547
CANNY-SURF	0.7849	0.8298	0.6803	0.728

Table 4 shows the number of identified matching features and correspondences found between the two images. The results have shown that the extracted feature points from

CANNY-SIFT and CANNY-SURF would produce incorrect corresponding points. Although the number of feature points detected is high, the accuracy of the matching is lower.

Table 4
Number of correct matches found during image matching

	Matches Found			
	Pair 1	Pair 2	Pair 3	Pair 4
SIFT-SIFT	130	137	62	74
SURF-SURF	264	356	31	56
CANNY-SIFT	44	10	5	6
CANNY-SURF	372	85	2244	2

Table 5 shows the execution time of image matching. It has been identified that the feature points from the SURF detectors and descriptors would perform faster than the other combination of feature extraction.

Table 5
Cost of time of image matching.

	Matching time (seconds)			
	Pair 1	Pair 2	Pair 3	Pair 4
SIFT-SIFT	7.141	7.446	5.760	5.238
SURF-SURF	4.425	4.376	4.152	4.186
CANNY-SIFT	7.432	7.811	5.252	4.497
CANNY-SURF	7.919	8.315	7.357	7.602

VI. Discussion

The conducted experiments have shown that the results of the image stitching differ to different feature extraction methods.

Figure 2 (a) shows the detected features using SIFT and Figure 2 (b) shows the result of the feature detection and extraction using SURF on the same image. It can be seen that the SIFT algorithm is able to detect and extract more feature points than the SURF algorithm. This was as expected based on the literature and also the type of image that contains high amount of texture and elements. From our observation, we notice that the SIFT algorithm is also robust to detect and extract features that may not be visible to the naked eyes and yet may contain information for matching. These features may be the corners, edges and high contrast points that are considered as good and feasibly features. Although SURF algorithm is found to be a lot more robust than SIFT algorithm, the extracted features are lesser in number and more accurate for image matching and stitching.

The combinations of both SIFT and SURF with Canny algorithms has given an unpredicted result in the number of the features extracted. As SIFT extracted more feature points than the SURF algorithm, the combination of Canny-SURF has outperformed Canny-SIFT algorithms.

(a) (b)

Figure 2: (a) shows the detected features using SIFT and (b) the detected features using SURF on the same image.

The FLANN based matcher algorithm is used for image matching. We have found that the SURF algorithm alone is better as feature extraction method compared to the Canny-SURF method. We have also identified that the high number of extracted features may not necessarily give high accuracy in feature matching.

The execution time or cost of time for feature detection and matching for SURF outperform Canny-SIFT method. This is due to the integral image description in SURF to boost the speed of detection and description, and at the same time reducing the complexity of the convolution operation computational. The robustness of the detected SURF features lead to the speed of the SURF features matching.

Figure 3: Feature corresponding points between images using FLANN based matcher and SIFT methods.

Figure 4: The resulted stitched image using SUR, FLANN based matcher, and RANSAC estimation model.

VII. Conclusion and Future Work

In this paper presents the comparative analysis of three popular feature detection and description techniques namely Canny Edge, SIFT and SURF methods. The FLANN based matcher algorithm is used to search for matching features on the images and RANSAC method is used to reduce the erroneous mismatched features to improve the stitching process.

The analysis of the methods measures the effectiveness, accuracy and computational time of the methods. The stitching pipeline has been successfully implemented and executed on the aerial images.

Based on the detected feature points, we have found that the combination of Canny-SURF method could detect more features than the other proposed combination. The SURF outperforms other combination on both feature detection and matching in term of the execution time. In feature matching, we have identified that the SURF features are more "robust" for matching purposes and the SURF method has better overall performance.

The results from the experiment suggest that each of the method produce different stitching output. For future works, we will analyse the performance of the methods on different metrics, such as type-1 and type-2 error, balanced F-score and average number of obtained features per image etc.. To conclude, stitching aerial images would require high accuracy and speed for image matching, hence SURF method is proposed to be used.

ACKNOWLEDGMENT

This research work was funded and supported by the Fundamental Research Grant Scheme (FRGS/ICT01(01)/1284/2015(01)) and the International Fund for Agricultural Development (IFAD) Grant with grant number IFAD L18403 I03 00.

REFERENCES

[1] Pranoti Kale and K.R.Singh, *A Technical Analysis of Image Stitching Algorithm*, International Journal of Computer Science and Information Technologies, vol. 6, 2015.

[2] Shikra Arya, *A Review on Image Stitching and its Different Methods,*International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5, issue 5, May 2015.

[3] Pranoti Kale and K.R.Singh, *A Technical Analysis of Image Stitching Algorithm Using Different Corner Detection Method,* International Journal of Innovative Research in Computer and Communication Engineering, vol. 3, issue 4, April 2015.

[4] Matthew Brown and David G. Lowe, *Automatic Panoramic Image Stitching using Invariant Features*, International Journal of Computer Vision, vol. 74, pp. 59-73, August 2007.

[5] P.M. Panchal, S.R. Panchal and S.K. Shah, *A Comparison of SIFT and SURF*, International Journal of Innovative Research in Computer and Communication Engineering, vol. 1, issue 2, pp. 323-327, April 2013.

[6] A.M.M. Madbouly, M. Wafy and Mostafa-Sami M. Mostafa, *Performance Assessment of Feature Detector-Descriptor Combination*, International Journal of Computer Science Issues, vol. 12, Issue 5, pp. 87-94, September 2015.

[7] Lingua, A., Marenchino, D. and Nex, F., *Performance analysis of the SIFT operator for automatic feature extraction and matching in photogrammetric applications*. Sensors, vol. 9, Issues 5, pp.3745-3766, May 2009.

[8] Mathew George and C. Lakshmi, *Object Detection Using the Canny Edge Detector*, International Journal of Science and Research, vol 2, No. 5, pp. 213-215, May 2013.

[9] A.D Chitra and P. Ponmuthuramalingam, *An Approach for Canny Edge Detection Algorithm on Face Recognition,* International Journal of Science and Research, vol. 4, issue 11, pp. 24-27, November 2015.

[10] S. Vijayarani and M. Vinupriya, *Performance Analysis of Canny and Sobel Edge Detection Algorithms in Image Mining*, International Journal of Innovative Research in Computer and Communication Engineering, vol. 1, issue 8, pp. 1760-1767, October 2013.

[11] Raman Maini and Himanshu Aggarwal, *Study and Comparison of Various Image Edge Detection Techniques*, International Journal of Image Processing, vol. 3, issue 1, pp.1-12, February 2009.

[12] R. Karthik, A. AnnisFathima, V. Vaidehi, *Panoramic View Creation Using Invariant Momentsand SURF Features,* IEEE International Conference on Recent Trends in Information Technology, pp. 376-382, 2013.

[13] Zhitao Xiao, Ming Yu, Chengming Guo and Hongmei Tang, *Analysis and comparison on image feature detectors,* 3rd International Symposium on Electromagnetic Compatibility, 2002, pp. 651-656.

[14] A. Aichert, *Feature extraction techniques*, Online, Available: http://home.in.tum.de/~aichert/featurepaper.pdf, January 2008.

[15] Lowe D.G., *Object recognition from local scale invariant features,* Proceedings of the 7th IEEE International Conference on Computer Vision, IEEE vol. 2, pp. 1150-1157, September 1999.

[16] Lowe D.G., *Distinctive image features from scale-invariant keypoints,* International Journal of Computer Vision, vol. 60, pp. 91-110, November 2004.

[17] H. Bay, A. Ess, T. Tuytelaars, and L. J. V. Gool, *Speeded-up Robust Features (SURF)*, Computer Vision Image Understanding, vol. 100, pp. 346 – 359, June 2008.

[18] Yang Zhan Long and Guo bao-long, *Image Mosaic Based on SIFT*, International Conference on Intelligent Information Hiding and Multimedia Signal Processing(IIHMSP), pp. 1422-1425, 2008.