

IMAGE MOSAICING USING FEATURE DETECTION ALGORITHMS

Under the supervision
of
Prof. U. C. Pati

Submitted by:
POOJA GHOSH
111EI0614
B.Tech 4th year
111ei0614@nitrkl.ac.in



Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Odisha, India- 769008

CERTIFICATE

This is to certify that the thesis entitled, “Image Mosaicing for a Panoramic View using Feature Detection Algorithm” submitted by **Pooja Ghosh** (Roll No. - 111EI0614) in partial fulfilment of the requirements for the award of Bachelor of Technology in Electronics and Communication Engineering at National Institute of Technology, Rourkela is an authentic work carried out by her under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/ Institute for the award of any Degree or Diploma.

Place: Rourkela

Prof. U. C. Pati

Date: 7-05-2015

Dept. of Electronics and Communication Engineering

National Institute of Technology Rourkela – 769008

ACKNOWLEDGEMENT

In this growing age of technology, the extent of correctness is a major point. This correctness and pin pointed guess is achieved only through hard work, experience and well-guided practice. As a new comer to this field, I required a lot of guidance that would not only help me in learning about this project but also help in improving my skills and increase my efficiency.

Hence, the most vital part is the presence of a guide whose knowledge and practical experience built my self-confidence and lent me a helping hand by which I was able to finish this project successfully. So, I am highly indebted to **Dr. Umesh C. Pati**, who acted as burning candle and enlightened me on this topic by providing all the necessary information and required resources till the completion of the project. I am also very thankful to **Ms. Achala Pandey** for her sincere guidance and help.

Last but not the least, my earnest thanks to all my friends, who have patiently extended me all the help needed for finishing this endeavour.

Pooja Ghosh

(111ei0614)

ABSTRACT

In most recent couple of decades, image processing specialists has been using image mosaicing as a testing field in real time applications. It has wide utilization in the 3D picture reproduction, field of satellite imaging, computer vision fields and a few therapeutic fields also. Movement recognition & tracking, mosaic-based localisation, resolution enhancement, generating substantial FOV, augmented reality, and so forth are also some of its application fields.

In this exploration work, feature based image mosaicing procedure has been proposed. There are five essential steps in feature based procedures: feature extraction, feature matching, transformation model estimation, image re-sampling and transformation, and image blending. The achievement of image mosaicing can be accounted by the feature identification algorithms such as Harris corner detector, SURF, FAST and FREAK. But each of these algorithms has their own particular impediments and preferences as indicated by the applications concerned.

The proposed strategy first compares the above mentioned four feature extraction algorithm on the basis of accuracy and computational time and determines FREAK to be the most optimum one and then utilizes this FREAK descriptor algorithm for feature detection. All the distinctive features detected in an image and the feature descriptors are shaped around the corners. Matching between the feature descriptors from both the images is done to achieve best closeness and all the features other than the ones with higher degree of resemblance are rejected. Now, the features with higher degree of resemblance are used for computing the transformation model and correspondingly, the warping of the image is done. The warping of the picture is done on a typical mosaic plane after estimation. The removal of the intensity seam in the neighbourhood of the boundary of the images and to modify the image grey levels at the

junction joint to obtain a smooth transition between the images is the final step. Alpha blending technique is utilized for the purpose of image blending

CONTENTS

CERTIFICATE.....	2
ACKNOWLEDGEMENT.....	3
ABSTRACT.....	4
LIST OF FIGURES.....	7
CHAPTER-1 INTRODUCTION.....	8
CHAPTER-2 STEPS INVOLVED IN IMAGE MOSAICING FOR A PANORAMIC VIEW.....	10
2.1. IMAGE ACQUISITION.....	12
2.2. FEATURE DETECTION AND EXTRACTION.....	12
2.2.1. Features.....	12
2.2.1.1. Points.....	13
2.2.1.2. Edges.....	13
2.2.1.3. Corners.....	14
2.2.1.4. Blobs.....	14
2.2.2. Feature Detection.....	15
2.2.2.1. Edge Detection.....	15
CANNY.....	15
SOBEL.....	16
2.2.2.2. Corner Detection.....	16
2.2.3. FEATURE MATCHING.....	16
2.3. IMAGE REGISTRATION.....	17
2.4. IMAGE WARPING.....	18
2.5. IMAGE BLENDING.....	21
AlphaBlending.....	21
CHAPTER – 3 COMPARISON OF FEATURE DETECTION ALGORITHMS FOR IMAGE MOSAICING.....	23
3.1. HARRIS CORNER DETECTION ALGORITHM.....	24
3.2. SURF (Speeded-Up Robust Features) ALGORITHM.....	25
3.3 FAST (Features from Accelerated Segment Test) ALGORITHM.....	26
3.4. FREAK (Fast Retina Key point) DESCRIPTOR.....	28
3.5. Parameters and procedures used for Comparison of Feature Detection Algorithms for Image Mosaicing.....	29
3.5.1. Basic Concept behind using the Above Algorithm is:.....	30

3.6. RESULTS.....	31
3.6.1. Tabulation for Comparison of Different Feature Detection Algorithms	31
3.6.2. Charts plotted for accuracy versus no. of images used.....	32
3.6.3. Charts plotted for time complexity versus no. of images used.....	33
3.6.3.1. Closer View of Runtime Graphs for FREAK, FAST, Harris	33
CHAPTER – 4 RESULTS.....	34
TEST CASE-1	35
TEST CASE -2.....	38
TEST CASE -3	40
TEST CASE -4.....	43
CHAPTER-5 CONCLUSION.....	45
CHAPTER – 6 REFERENCES.....	47

CHAPTER-1

INTRODUCTION

An image mosaic is a panoramic image created by collecting successive images that fit into the same scene. The understanding of geometric relationships or the relation between various image coordinate systems i.e., coordinate changes among the different images to be processed can be used for obtaining a mosaiced image. After the required transformations, - like affine, perspective and polynomial transformations-, are applied, the overlapping portion of transformed images is warped and blended, hence framing a single image of the same object that covers the entire visible area of the scene. So, a single substantial image is framed by converging two or more images inputs in image mosaicing. The output mosaiced scene is his single blended image that has a larger field of view (FOV).

Input images are taken and features are identified utilizing various feature detection technique. Feature correspondences between the images are found and then the projection of images is done in a plane and after that they are warped and blended together. The MATLAB software is utilized for actualising the entire strategy.

CHAPTER-2

STEPS INVOLVED IN IMAGE MOSAICING FOR A PANORAMIC VIEW

The steps involved in Image mosaicing are as follows:

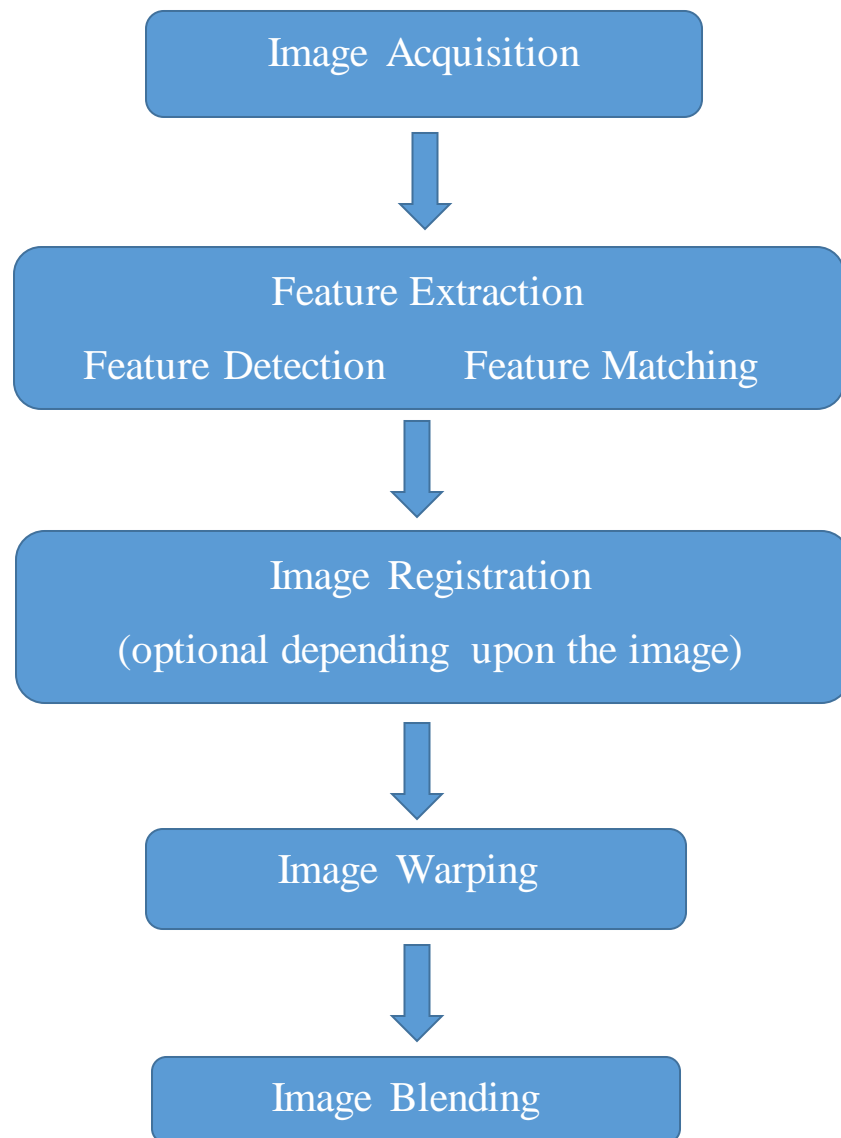


Figure 1. Steps in image mosaicing

2.1. IMAGE ACQUISITION

Images are taken as input in this part. It can be done utilizing imread function accessible in Matlab.

2.2. FEATURE DETECTION AND EXTRACTION

One of the most essential and the most fundamental steps in mosaicing of images is the Feature Detection and Feature correspondence between the images. Feature correspondences between the images is needed to adjust and blend the images correctly. Below is a brief description of the instructions required for detecting and correlating the features between the two or more images.

2.2.1. Features

Any combination of data points or any portion of the image that can be considered as an identity point or a key point. For further processing, this point can be considered as a candidate point.

Another query that one needs to consider is, what type of points would provide better feature point arrangement. Those points where there is a little or a smooth variation in the pixel quality are not supposed to be considered as feature points because they won't be able to provide much information when finding out relationships among the images. In general, candidate points are those feature points where there is a substantial variation in pixel quality. The example of features is shown below in figure 2. The red marked rectangles in the image depicts the feature points.

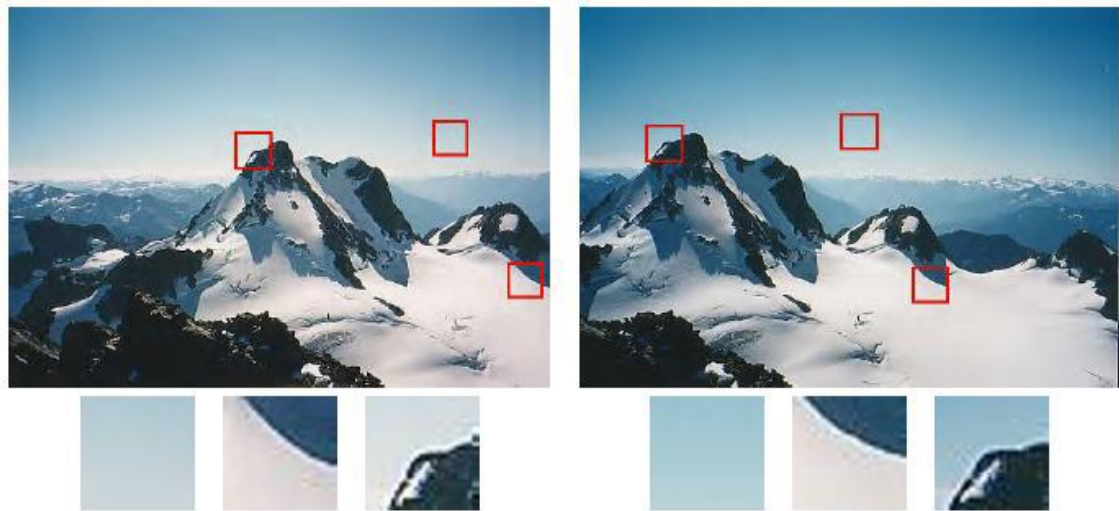


Figure 2. Representation of feature points

As depicted in the above Figure 2. [13], the edges or corners will be considered as good candidate points.

The type of features are briefly described below.

2.2.1.1. Points

In this, the features are considered to be point-type. But as there are large number of points in an image, so considering a point to be a feature will not be of much worth, because of the reason that time required for processing these many points will be quite high and henceforth, the memory requirement would also be large.

2.2.1.2. Edges

The boundary points of the images are considered to be edges. The junctions or intersections between pixel values that are different from their neighbourhood pixel values can be defined as edges. The gradient between the neighbourhood derivative pixel values is lower than the gradient between image pixels. For low level processing, these are considered to be good features. So, the one dimensional feature points is also known as Edges.

2.2.1.3. Corners

The two-dimensional feature points are considered as corners. The feature points in images which generate gradients of more prominent values in both the dimensions are said to be corners. Very rapid variations in pixel values is seen in corners. Here, the algorithms which were developed over time for detecting feature points using edge detection method is applied in both direction to find a corner. Sometimes, a few chosen corner points are not the traditionally chosen corner points, such as, small dark patches in white background. They are considered as corner points, even when they are not supposed to be.

2.2.1.4. Blobs

While point like features are provided by corners, the description of feature points provided by blobs is complementary and has region-type structures. In Blob detectors, firstly a middle point is taken which is considered to be a preferred point (generally, it is a nearby maxima of the administrator response) and then comparison is done between this middle point and all the other remaining points. Hence, as it works on a patch of features instead of a certain feature point, therefore, it is much more helpful in identifying even such interest points that are somewhat hard to be perceived by corner detection algorithms.

An image is shrunk into another process, when a smooth patch is found in the image and, then, conversion of this smooth patch into an interest point may be done. So, taking into consideration some of the correspondent properties between the images while processing them, a particular scale should be considered.

The corner detection algorithm includes the algorithm like LoG and DoG, in spite of the fact that they deal with blob detection.

Further, we would learn about how to consider and assess the feature points based on image mosaicing operations.

2.2.2. Feature Detection

Feature detection is instinctive yet an essential and a fundamental step in image mosaicing through which candidate points are identified for further processing. Only those points which can be effortlessly matched are preferred in feature detection algorithm. Many different methods have been developed so far for the purpose of feature detection. To sum things up, and to present the notion of how feature detection algorithms are utilized to detect features, some of the most widely used feature detectors are described below.

Before going into the brief description of feature description techniques, it is further emphasized. It should be noted that only those points are chosen as candidate points where the variation in pixel value is not only fairly high but also effortlessly detectable.

2.2.2.1. Edge Detection

CANNY

Canny edge detection chips away at the straightforward recipe of finding the edge using slope/gradient values. This method, which has been developed by CANNY, convolutes the image with a predefined gradient matrix. The magnitudes can be modified using adaptive techniques. In light of its notoriety and use in different fields, MATLAB has incorporated a predefined function for it.

SOBEL

While canny edge detector convolutes one patch over the image, Sobel does likewise using two patches; one in x-direction and another in y-direction to achieve a superior and quicker result. It can likewise be adjusted using adaptive filtering techniques. Because of its immense applications in image processing, MATLAB has additionally incorporated a predefined function for it.

2.2.2.2. Corner Detection

Different types of corner detection algorithm such as Harris, FAST, SURF, FREAK are explained in Chapter-3

2.2.3. FEATURE MATCHING

All the features that have been detected should be matched so as to confirm that features are from the corresponding locations from completely different images. As the arrangement of feature points may not be accurate, therefore, an exact coordinating needs to be done by the means of progressive incremental motion refinement, but that is tedious, and may degrade the performance.

When features tracking is done over larger image arrangements, it may result in larger variation in their appearances. In such case, with the help of an affine, projective or other movement model, comparison between the appearances must be done.

The descriptors of differential invariants are lethargic to moderations in arrangement by configuration, and hence, failed to do as expected.

2.3. IMAGE REGISTRATION

The process of superimposing or geometrically aligning two or more images of the same scene:

- From different viewpoints
- Taken at different times
- By different cameras

is known as image registration.

Here, two or more images geometrically aligned', generally taking into consideration only two at a time, i.e., the reference and target image.

It is the not only the fundamental basis, but also the most essential part of image mosaicing process. Multiple source images are registered with certain image by using an image transformation parameters which can be generated using similarity measurement. This similarity measurement can be obtained by collecting two or more images of the same scene but produced from different perspectives, different sensors and different times. In this process, multiple source images can be transformed into a single homogenous coordinate system for obtaining the best possible match on the pixel level. The source and target image $I_1(x, y)$ and $I_2(x, y)$ respectively, can be represented by two 2-D arrays. Image registration can be stated as follows:

$$I_1(x, y) = g(I_2(f(x, y)))$$

Where, f is a homography transform model estimated function for 2-D coordinates.

2.4. IMAGE WARPING

For the purpose of blending images, distortion of the image is done significantly by digital manipulation of that image through image. It can be used for modifying the position of the image points by distorting and mapping the images from one frame to another frame while keeping the pixel value unchanged, so that they can be blended. This can be based mathematically on any function from part of the plane to the plane. The original image can be reconstructed provided the function is injective. But if the function is bijective in nature, then inverse transformation of any image can be done. For this last step, the range of warped image coordinates for each input image needs to be computed in order to determine the size of the output mosaiced image. The size of the output image can be easily achieved by mapping four corners of each source image forward and computing the minimum and the maximum values of x and y coordinates. Finally, calculation of x-offset and y-offset values specifying the offset of the reference image origin relative to the output panorama needs to be done. In the next step, the mapping the pixels from each input image to the plane defined by the reference image can be done using the inverse warping method described above. Here, an already existing code is used for warping the images to a generalized plane so that they could be blended properly. Taking the basic Cartesian co-ordinate systems, the co-ordinates transformation can be described as follows.

[It can be represented as $x' = x + t$ or

$$x' = [I \ t]x \dots\dots 1$$

I is the 2*2 identity matrix and,

(x, y, 1) is the homogenous 2D co-ordinate

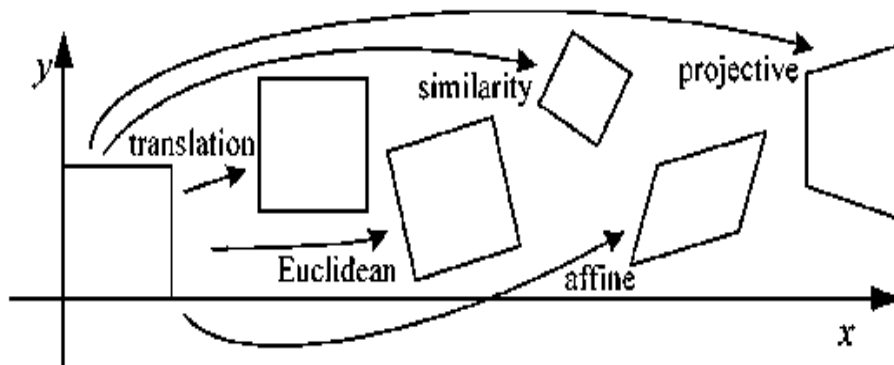


Figure 3 Basic set of 2D planar transformation

Scaled Rotation: It is also known ‘Similarity Transform’. It can be represented as:

$$x' = sRx + t.$$

where, x = original image matrix

s = an arbitrary scale factor

This expression can also be represented as:

$$x' = [sR \ t] x \sim = \begin{bmatrix} a & -b & t_x \\ b & a & t_y \end{bmatrix} \dots\dots\dots 2$$

Where, $[a^2 + b^2 = 1]$ is no longer a requirement. Now in similarity transformation, angle between lines are never altered.

Rotation and Translation: It is also known as ‘2D Euclidean transformation’. It is called so because there is preserving of Euclidean distances in this type of transformation. Here,

$$x' = Rx + t \dots\dots\dots 3 (I)$$

$$\text{or, } \mathbf{x}' = [I \quad t] \tilde{x} \quad \dots\dots\dots 3(2)$$

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \quad \dots\dots\dots 4$$

Where, R is an orthonormal rotation matrix whose determinant value is 1.

Affine: It can be represented as

$$\mathbf{x}' = A$$

where, A is an arbitrary 2*3 matrix.

$$\mathbf{x}' = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{bmatrix} \quad \dots\dots\dots 5$$

Parallel lines remain parallel under affine transformation.

Projective: It is also known as perspective transform, it can be operated on homogenous coordinates represented as \hat{x} and \tilde{x} as follows

$$\hat{X} = H \tilde{x} \quad \dots\dots\dots 6$$

Where, \sim denotes equality up to scale and H is an arbitrary 3*3 matrix.

H is a homogenous matrix which is defined up to a scale.

The homogenous coordinate obtained should be normalized to get inhomogeneous coordinate \mathbf{x}' represented as

$$\mathbf{x}' = \frac{h_{00}x + h_{01}y + h_{02}}{h_{20}x + h_{21}y + h_{22}}$$

and,

$$y' = \frac{h_{10}x + h_{11}y + h_{12}}{h_{20}x + h_{21}y + h_{22}} \dots\dots\dots 7$$

The orientation of straight lines is never altered in this type of transformations.






Name	Matrix	Number of d.o.f.	Preserves	Icon
Translation	$[\mathbf{I} \mathbf{t}]_{2 \times 3}$	2	Orientation + ...	
Rigid (Euclidean)	$[\mathbf{R} \mathbf{t}]_{2 \times 3}$	3	Lengths + ...	
Similarity	$[s\mathbf{R} \mathbf{t}]_{2 \times 3}$	4	Angles + ...	
Affine	$[\mathbf{A}]_{2 \times 3}$	6	Parallelism + ...	
Projective	$[\tilde{\mathbf{H}}]_{3 \times 3}$	8	Straight lines	

Table 1 Hierarchy of 2D coordinate transformations.

2.5. IMAGE BLENDING

The images are blended after warping the images to a common plane or it can be said that for obtaining the final output image with greater FOV than the existing images, images need to be mixed with each other after warping. And hence, in this way averaging of pixel values can be done.

AlphaBlending

This algorithm is a transition smoothing algorithm. The image boundaries are gradually degraded in decreasing order as well as increasing order instead of smoothening the image boundaries according to the equation given below,

$$I = I_1 * \alpha + I_2 * (1 - \alpha) \quad \dots\dots\dots 8$$

The value of the alpha depend upon the window size. Its value varies from 0 to 1 and vice versa and the increment or decrement respectively depends on the window size.

Figure 8. below shows the feather blending with different window sizes.

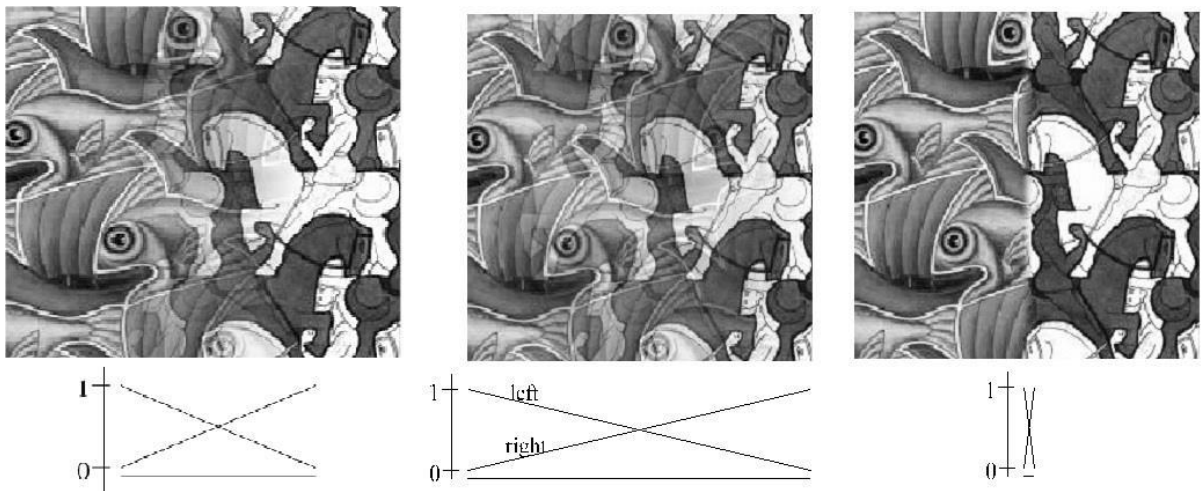


Figure 4 Alpha blending with varying window

CHAPTER – 3

COMPARISON OF FEATURE DETECTION ALGORITHMS FOR IMAGE MOSAICING

3.1. HARRIS CORNER DETECTION ALGORITHM

Harris corner detection algorithm is an algorithm which detects feature points by designing a local detecting window inside the image. The small amount of shifting of window in different direction can be determined by the average variation in the pixel intensity. The corner point is the centre point of the window. Hence, on shifting the window in any of the direction, a large variation in pixel intensity is seen. Hence, Harris corner detector detects the corner points. When the window is shifted, no change in pixel intensity is seen in any direction if a flat region appears. But when there is no change in pixel intensity along the edge direction then an edge region is detected. But when there is a significant change in pixel intensity in every direction, then a corner is said to be detected. Hence, a mathematical approach for determining whether the region found is flat, edge or corner is provided by Harris corner detection algorithm. More number of features are detected using this detection algorithm. Though, it is found to be scale variant, but it has a benefit when it comes to rotation, as it is invariant to rotation.

The change in pixel intensity for the shift [u, v] is given as below:

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u + y + v) - I(x, y)]^2 \quad \dots\dots 9$$

Where, $w(x, y)$ is a window function,

$I(x, y)$ is the intensity of the individual pixel, and

$I(x + u, y + v)$ is the pixel intensity after shift.

The algorithm for Harris corner detection is given as:

1. Autocorrelation matrix M for each pixel (x, y) in the image is calculated as follows:

$$M = \sum_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad \dots\dots 10$$

2. There is Gaussian filtering for each pixel of image is generated using discrete two-dimensional zero-mean Gaussian function as:

$$Gauss = \exp(-u^2 + v^2) / 2\delta^2 \quad \dots\dots\dots 11$$

3. The corners for each pixel (x, y) is calculated and R is given as follows:

$$R = \det(M) - k * \text{trace}(M)^2 \quad \dots\dots\dots 12$$

4. We choose a local maximum point. The feature points whose pixel values are corresponding with the local maximum interest point are considered in Harris corner detection method

5. The detection of corner points is done after setting the threshold value T

3.2. SURF (Speeded-Up Robust Features) ALGORITHM

SURF (Speeded-Up Robust Features) is a quick and a robust algorithm which was developed by Bay [14] for nearby, closeness invariant representation and correlation. The SURF methodology can be partitioned into three fundamental steps. To start with, the first step is, choosing key feature points such as edges, corners, blobs, and T-intersections at distinctive regions in the image. Second step is, feature vector is used to depict the surrounding neighbourhood of each feature point. This descriptor must be a unique one. Also, at the same

time, it ought to be robust to error identification, noise, photometric and geometric deformations. Last step is, the descriptor feature vectors are coordinated among the different accessible images. A Fast-Hessian Detector is used for finding feature points taking into consideration the close estimation of the Hessian lattice of a given picture point. Before the shaping of feature point descriptor is done from the wavelet responses in a certain surrounding to the point, an introduction task needs to be done. This can be done by using the responses to Haar wavelets. This is the reason why a circular region is developed around the detected feature points when SURF algorithm is used.

The fundamental point of interest of the SURF methodology is its fast computation, which empowers numerous ongoing applications, for example, image mosaicing, tracking and object recognition. It has speeded-up the SIFT's location transform as well as has counteracted nature of the recognized feature points from degrading. The principle focus is laid on speeding-up the matching step.

3.3 FAST (Features from Accelerated Segment Test) ALGORITHM

FAST stands for Features from Accelerated Segment Test. In this, a machine learning approach is used which is adaptive in processing. Here, only a few points inside the range are identified and processed and the point that fall outside the scope of interest are rejected.

The FAST strategy was presented by Rosten Drummond [15] for recognizing interest points in an image. The purpose behind the work of FAST algorithm was to add such an interest point indicator in the list of other interest point detector that can be utilized continuously real time frame rate applications.

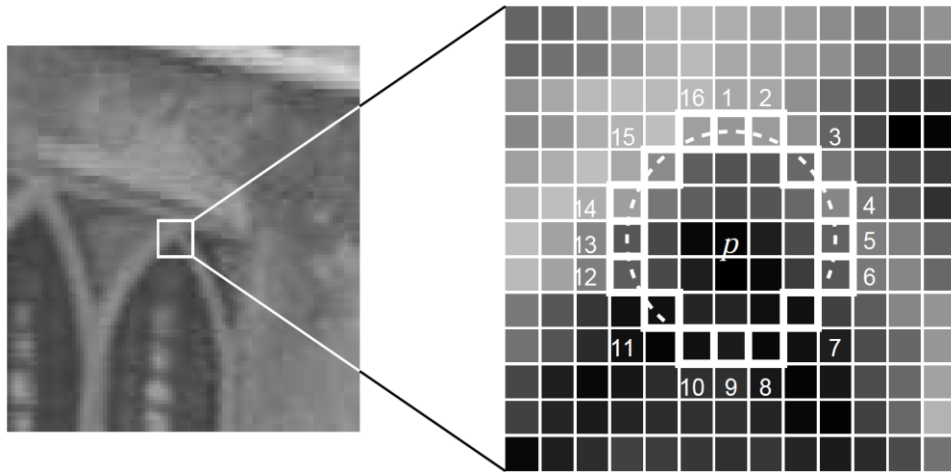


Figure 5 Image showing the interest point under test and the 16 pixels on the circle [1]

The algorithm is explained below:

1. A pixel “p” is selected in the image. The intensity of this pixel is assumed to be I_p . This is the pixel under test, i.e., it needs to be confirmed whether it is an interest/feature point or not. (Refer to figure 2)
2. A threshold intensity value T is set, (assume its value to be around 20% of the pixel under test).
3. A circle of 16 pixels surrounding the pixel p is considered whose pixel intensity is assumed to be I_p . (This is a Bresenham circle [2] of radii 3.)
4. For the pixel to be distinguished as an interest point, “N” adjacent pixels out of the 16 need to be either above or beneath I_p by the quality T . (The creators have utilized $N = 12$ as a part of the first form of the algorithm)
5. The algorithm can be made fast by comparing the intensity of pixels 1, 5, 9 and 13 of the circle with I_p first. As apparent from the above figure, no less than three of the above mentioned four pixels ought to fulfil the rule in step 4, so that the interest point will exist.
6. The pixel “p” will not be considered as an interest point (corner) in the event that no less than three of the above mentioned four pixel values - I_1, I_5, I_9, I_{13} are neither above nor underneath $I_p + T$. hence, in such situations, pixel “p” will be rejected from becoming a corner

point. Else if at least three of the pixels are above or beneath $I_p + T$, then check for each of the 16 pixels and check if 12 contiguous pixels fall under the criterion.

7. The process is repeated for all the pixels in the image.

There are a couple of limits to the algorithm. To start with, for $N < 12$, the algorithm does not work extremely well in all cases in light of the fact that when $N < 12$ the quantity of interest points identified are high. Second, the order in which the 16 pixels are queried decides the speed of the algorithm

3.4. FREAK (Fast Retina Key point) DESCRIPTOR

FREAK is a binary descriptor that improves the sampling pattern and method of pair selection used by BRISK. At the locations around the feature point 43 weighted Gaussians are evaluate using FREAK but the pattern formed by these Gaussians is biologically inspired by the retinal pattern in the eye. There is overlapping of the pixels that are being averaged (see Figure 6), and hence, they are much more concentrated near the key point than at the farther side. This results in a more accurate depiction of the key point as examination will demonstrate.

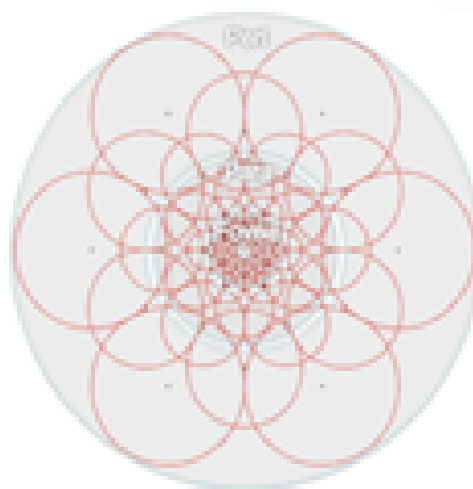


Figure 6. Freak Sampling Pattern[2]

A cascade is used in the actual FREAK algorithm for comparing these pairs, which puts forward the most important 64 bits for speeding up the matching process.

The algorithm used is as follows:

1. For obtaining the local areas of interest from the images, the FAST key point detector is applied separately on all the image inputs.
2. Then, the descriptors, i.e., feature vector for all of these key points that are present in the reference images are obtained using the FREAK key point descriptor, which describes each key point with a 64 bit descriptor.

3.5. Parameters and procedures used for Comparison of Feature Detection Algorithms for Image Mosaicing

Two main parameters used for comparison of different feature detection algorithms in image mosaicing are:

- Accuracy and
- computational time(run time)

The procedure used is as follows:

1. Read an input image (original image) and save it in variable a.
2. Then, distort the original image and save it in another variable b. This can be done by either resizing the image or by rotating the image or by changing pixel intensity of the image, etc. But care should be taken whatever distortion is done should be same when using for each of the four algorithms.

3. Detection of feature points is done using different algorithms separately and out of these, N strongest features are detected in each case which is followed by extraction of features.
4. Then, feature matching is done for the extracted features in each case
5. Accuracy(%) can be given as:

$$\frac{1}{n} \sum_{i=1}^n \frac{I_m}{I_{ex}} * 100$$

where, I_m =total no. of matched features

I_{ex} =total no. of extracted features from original image

n = total no. of images used

6. Lesser the computational time, better is the performance of the algorithm. This can be calculated by using the run and time option provided in MATLAB whenever the program is run. On clicking the run and time option, profiler window opens up that includes all the timing data for each line executed in the code and also the total execution time. So, from there we can note down the total execution time and time for only extraction.

3.5.1. Basic Concept behind using the Above Algorithm is:

Even when we distort an image, the best feature points of an image should remain almost the same. So, when feature matching is done between the original image and the distorted image, the more is the no. of matching features out of the no. of extracted features, the more is the accuracy.

So, it can be said that accuracy is a relative term that also depends on the no. of extracted features. Hence, accuracy is defined as the percentage of matched features to the extracted features.

3.6. RESULTS

3.6.1. Tabulation for Comparison of Different Feature Detection Algorithms

No. of Images	Algorithm	Run Time (sec)	Extracted Features (original) (distorted)		Matched Features (original) (distorted)		Accuracy (%)
1	FREAK	0.002	45	109	35	35	77.7
	FAST	0.003	136	148	73	73	53.6
	SURF	0.057	38	70	14	14	26.3
	Harris	0.008	48	140	8	8	16.7
4	FREAK	0.003	288	508	154	154	49.35
	FAST	0.004	556	591	241	241	42.51
	SURF	0.215	369	520	131	131	36.07
	Harris	0.024	249	895	30	30	15.73

Table 2 Comparison of Different Feature Detection Algorithms

3.6.2. Charts plotted for accuracy versus no. of images used

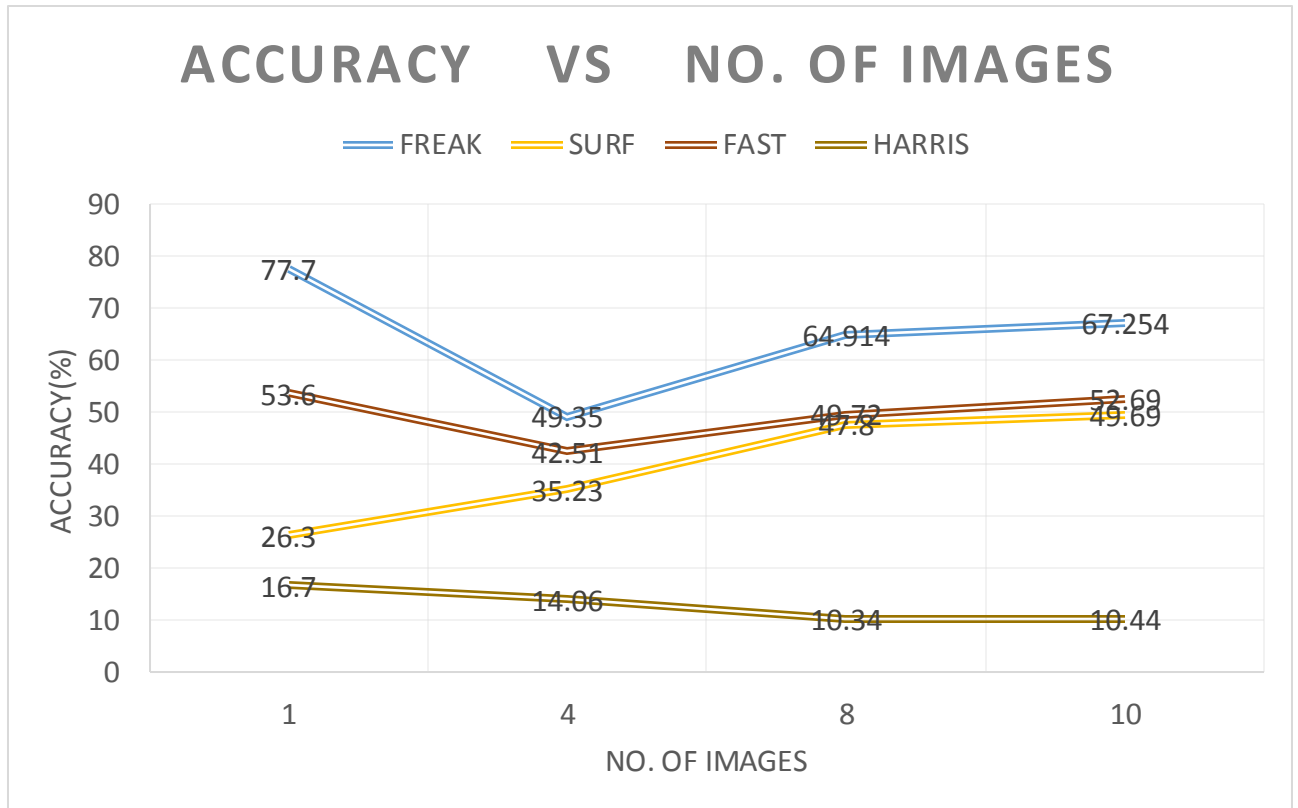


Figure 7 accuracy versus no. of images used plot

3.6.3. Charts plotted for time complexity versus no. of images used

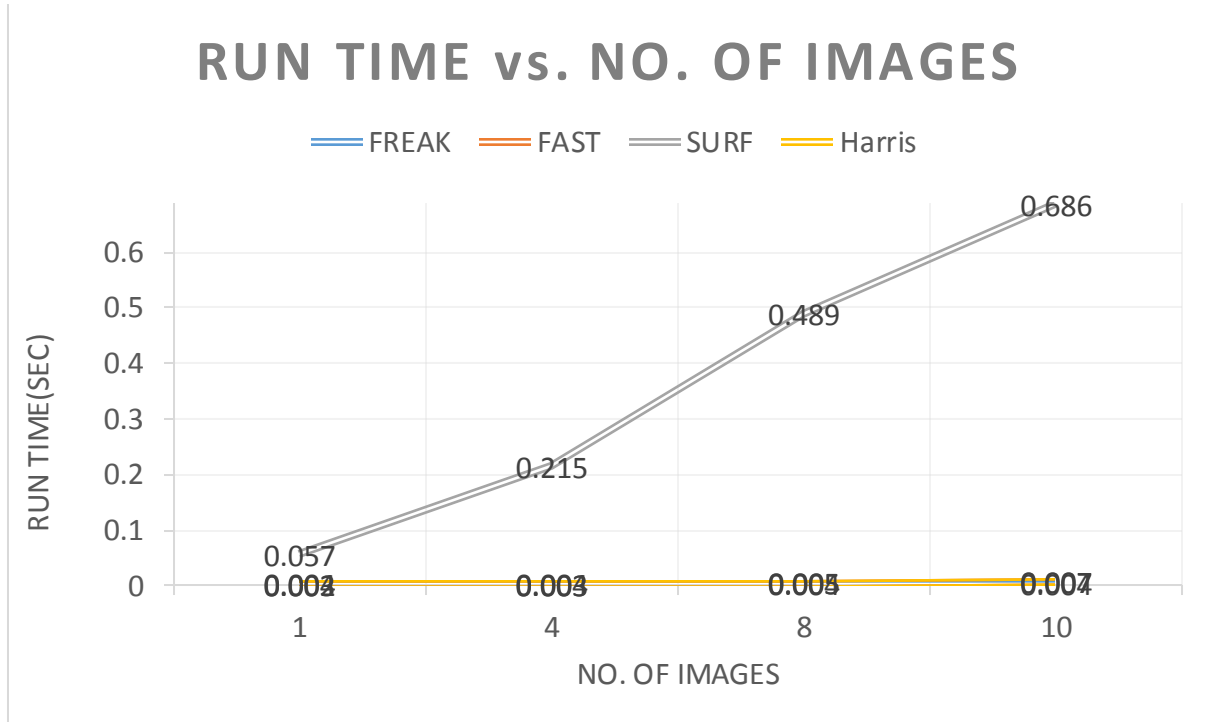


Figure 8 time complexity versus no. of images used plot

3.6.3.1. Closer View of Runtime Graphs for FREAK, FAST, Harris

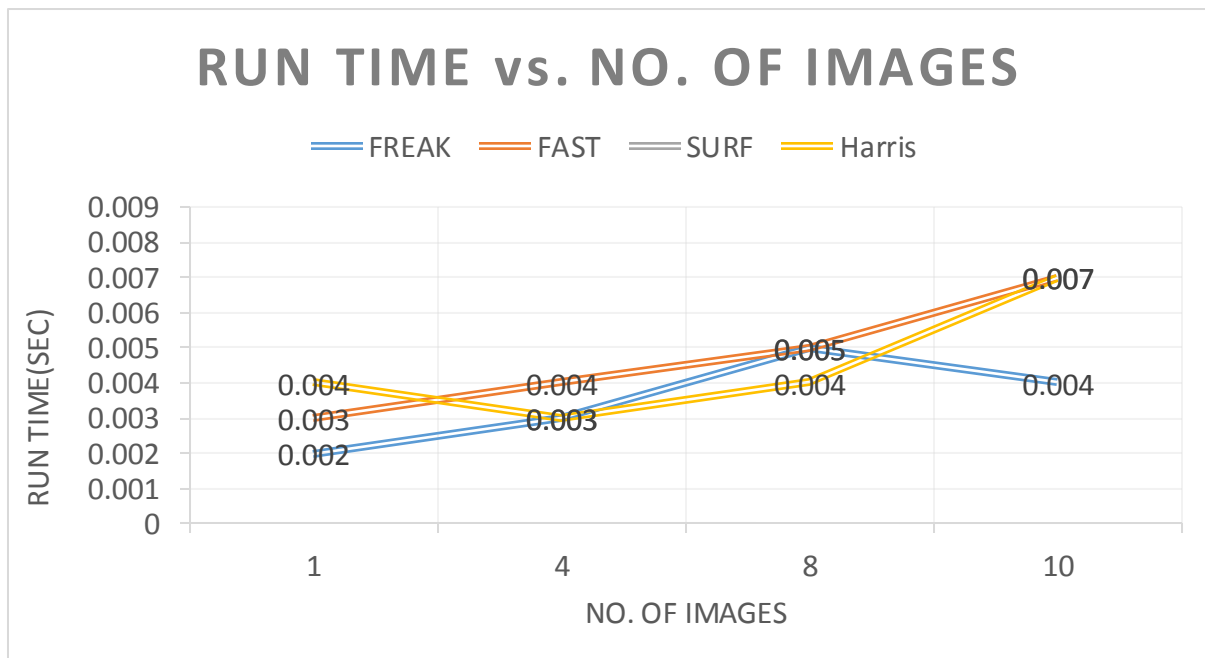


Figure 9 closer view of time complexity versus no. of images used

CHAPTER - 4

RESULTS

TEST CASE-1



Input image-1(ref)

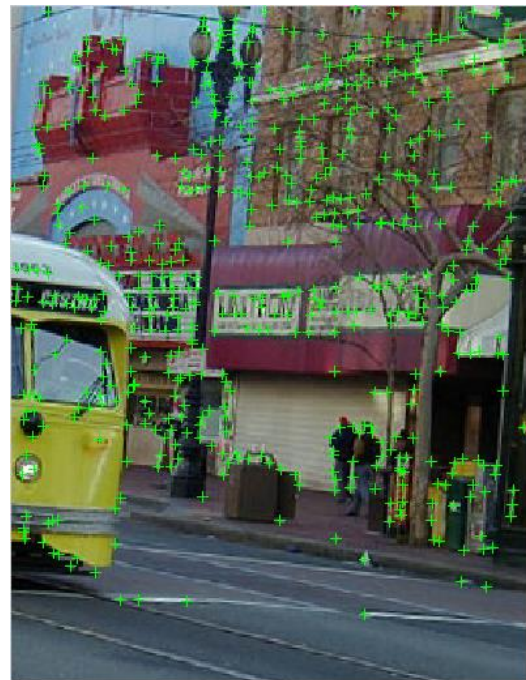


input image-2(target)

After using FREAK:



ref image

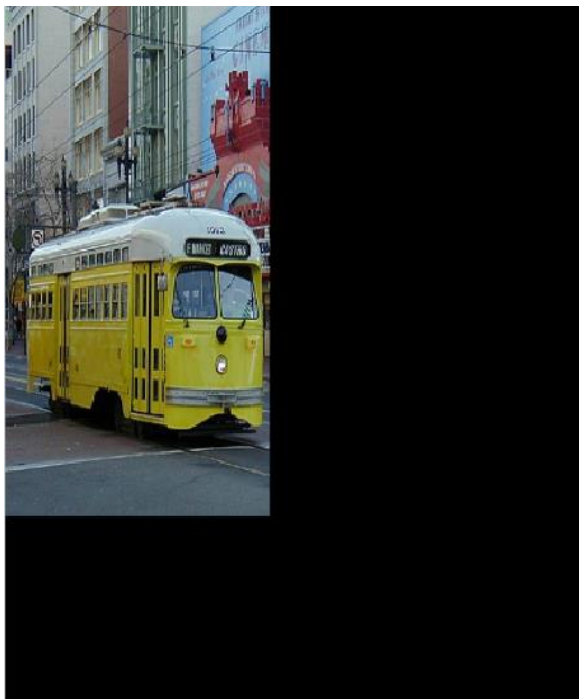


target image

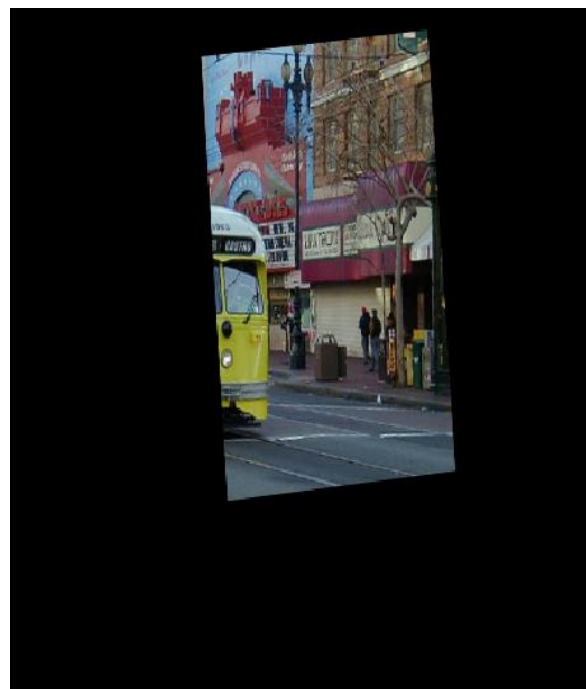
After FEATURE MATCHING:



after WARPING



ref image onto mosaic plane



target image onto mosaic plane

BLENDING



Final mosaiced output

TEST CASE -2

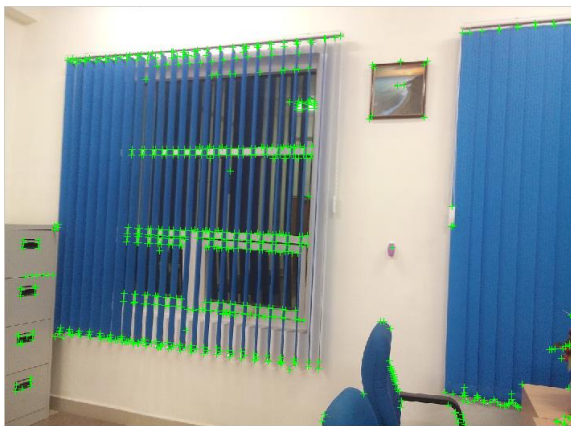


Input image-1(ref)



Input image-2(target)

After using FREAK:

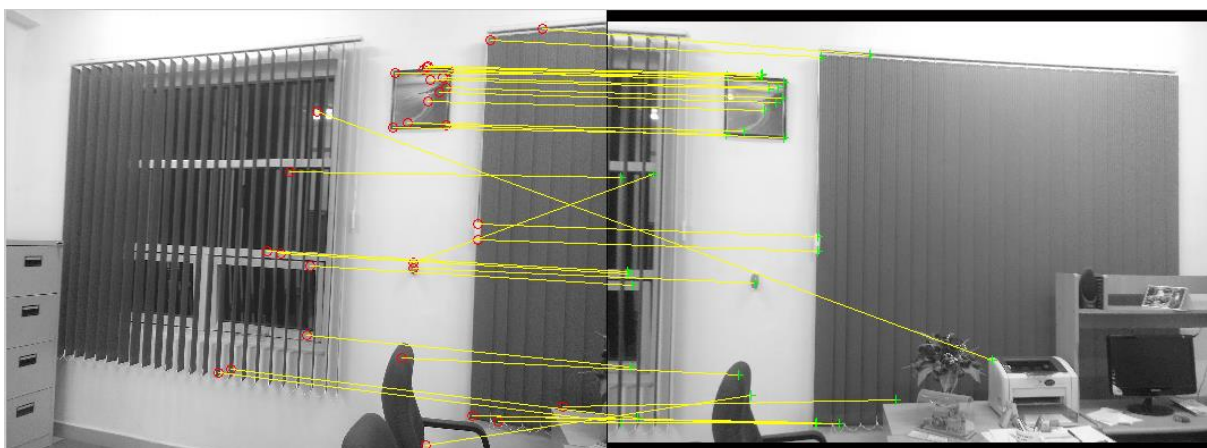


ref image

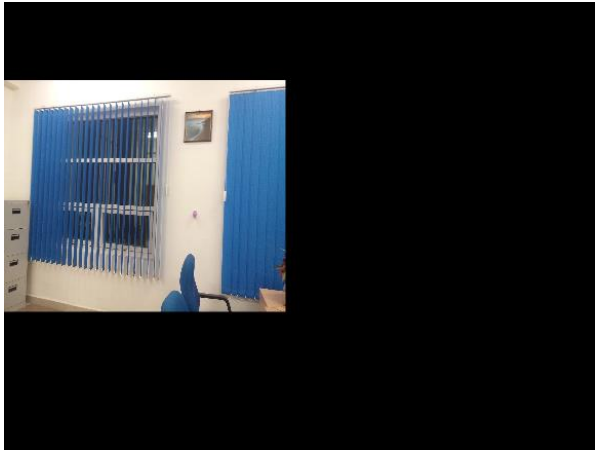


target image

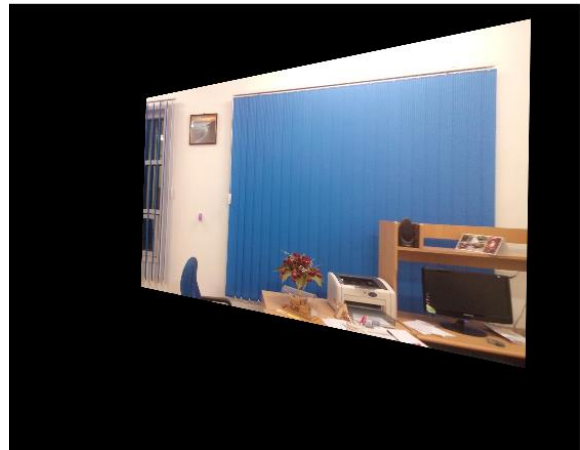
After FEATURE MATCHING



After WARPING



ref image onto mosaic plane



target image onto mosaic plane

After BLENDING



Final mosaiced output

TEST CASE -3



Input image-1(ref)



Input image-2(target)

After using FREAK:

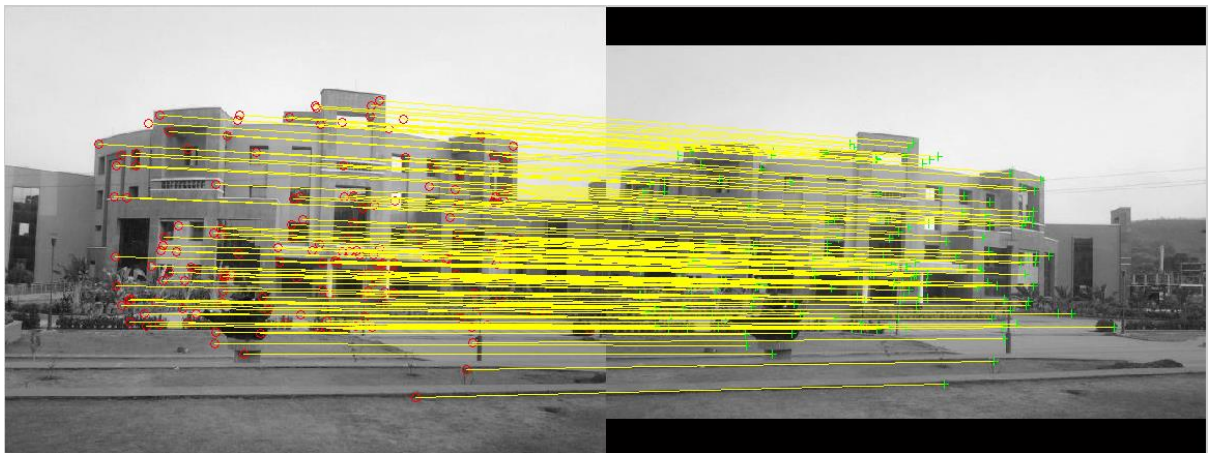


ref image



target image

After FEATURE MATCHING



After WARPING



ref image onto mosaic plane



target image onto mosaic plane

After BLENDING



Final mosaiced output

TEST CASE -4

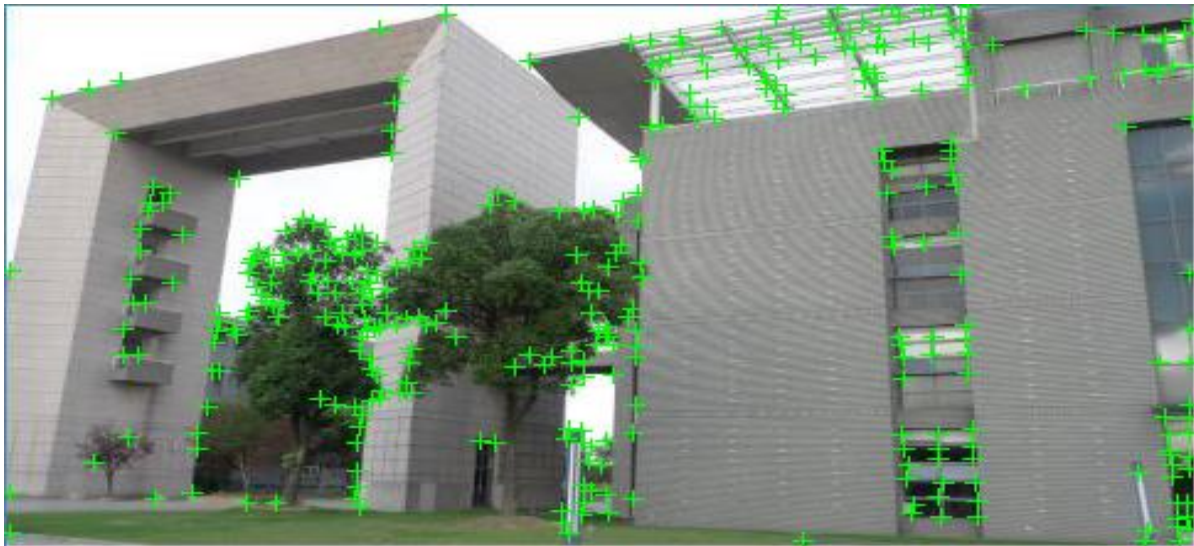


Input image-1(ref)

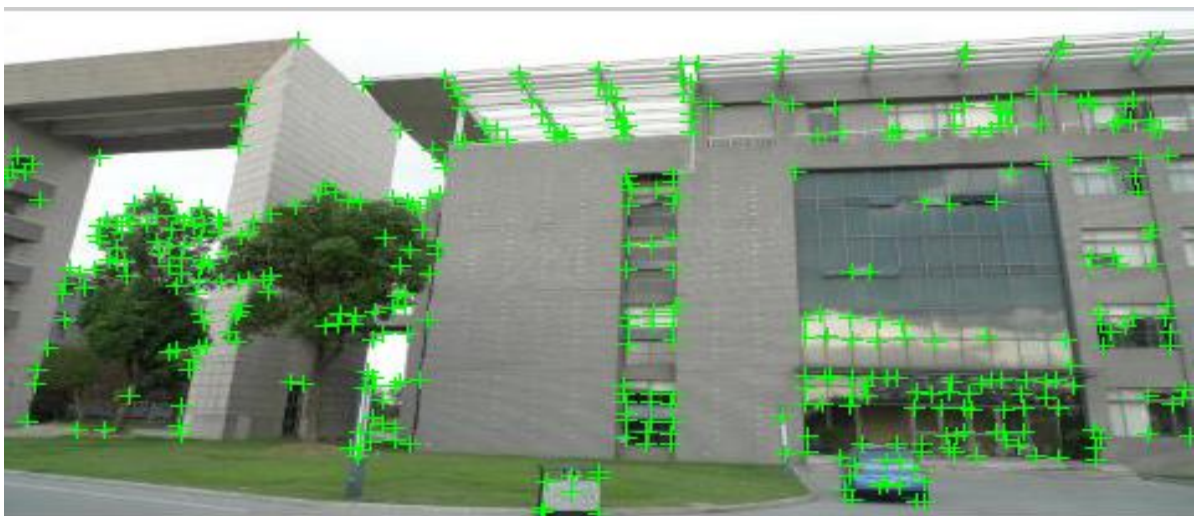


Input image-2(target)

After using FREAK:

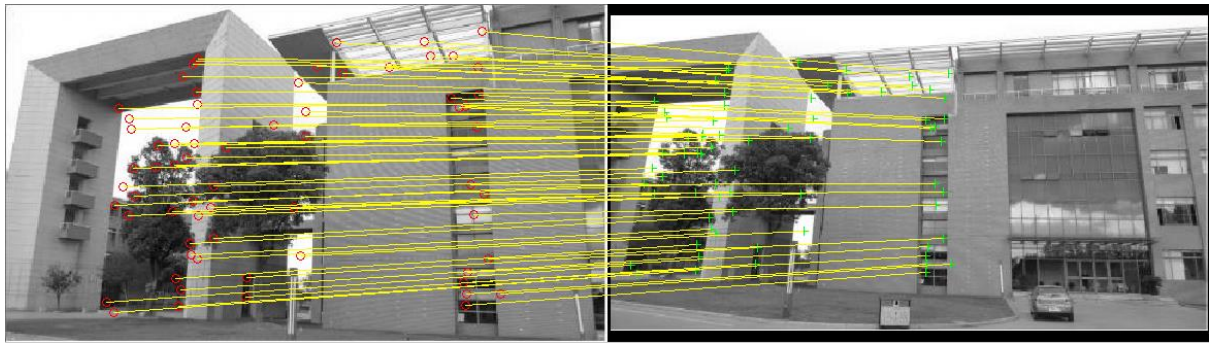


ref image

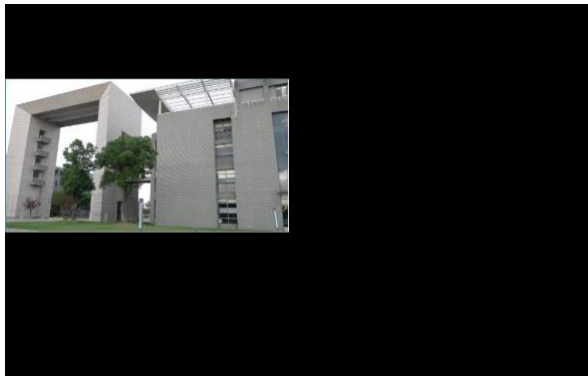


target image

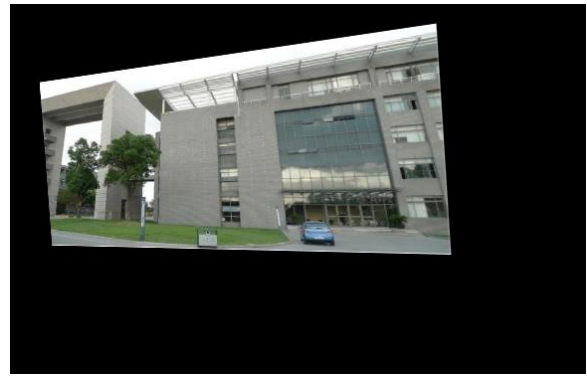
After FEATURE MATCHING



After WARPING

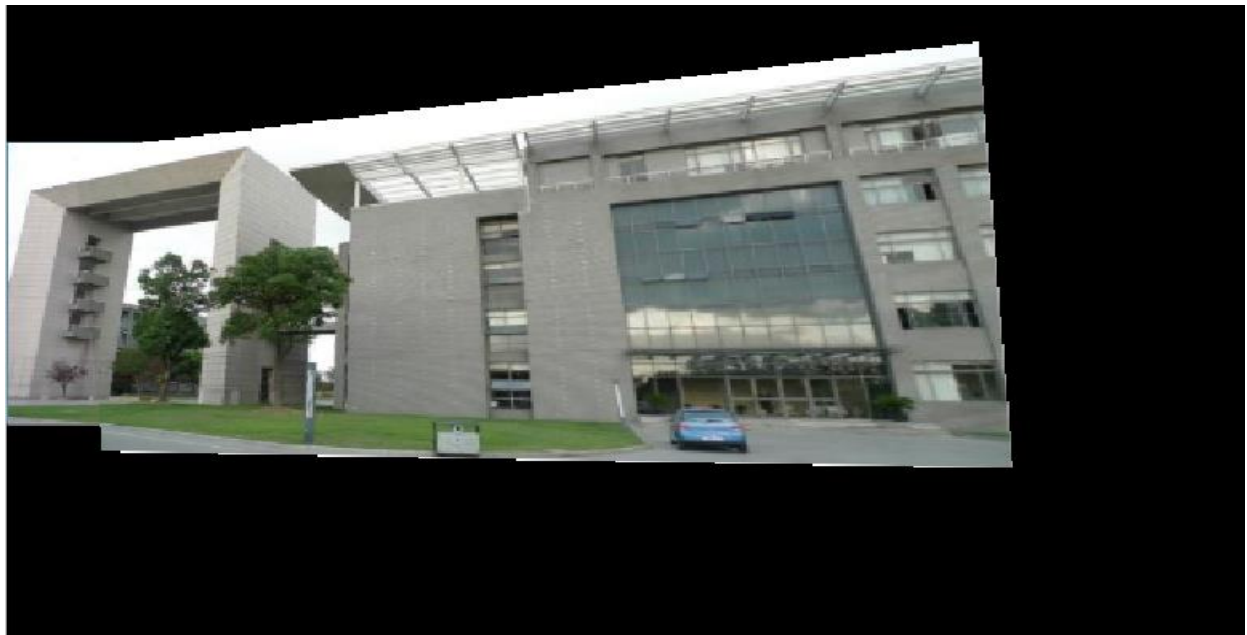


ref image onto mosaic plane



target image onto mosaic plane

After BLENDING



Final mosaiced output

CHAPTER-5

CONCLUSION

The method of merging of the images which has some overlapping area to form a single image with larger field of view is known as Image mosaicing. Yes, of course, image mosaicing can also be done for images which do not have any overlapping area, but here in this project we have only considered the case of images with overlapping area.

From both the graphs it can be concluded that FREAK has the highest accuracy with an optimum speed. FAST also has a good accuracy and speed but both are less as compared to FREAK. SURF has mediocre accuracy but is slowest as compared to other algorithm. Harris has a poor accuracy as compared to others but has good computational speed because of the simple algorithm implied.

While comparing different feature detection algorithm care should be taken for the image dataset used. The accuracy results may vary from image, so, same set of images should be used for each algorithm and such images should be chosen in the dataset that have larger no. of interest points. So, after comparing and seeing the results, we have used FREAK feature detection algorithm for extracting feature points from images. The common overlapping area between the two images is matched using the interest/feature points.

The estimation of transformation model is done. After Estimating the transformation model, the transformation of image is done with respect to the reference image and then a common mosaiced frame is used for warping the image onto it.

Image blending using Alpha blending method has been implemented to obtain the final mosaiced image output.

The final mosaiced image output is achieved after performing the image blending which can be implemented using the Alpha blending method.

CHAPTER – 6

REFERENCES

- [1] E. Rosten and T. Drummond (2006). Machine Learning for High-speed Corner Detection. Proceedings of the 9th European Conference on Computer Vision - Volume Part I (pp. 430-443). Berlin, Heidelberg, Springer-Verlag.
- [2] A. Alahi, R. Ortiz and P. Vandergheynst, "FREAK: Fast Retina Keypoint," *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
- [3] N. et al., editors, Handbook of Mathematical Models in Computer Vision, pages 273–292, Springer, 2005.
- [4] D. K. Jain, G. Saxena, V. K. Singh, "Image mosaicing using corner technique", Proceedings of International Conference on Communication System and Network Technologies, pp. 79-84, 2012.
- [5] Satellite and aerial Image mosaicing – a comparative insight- Samy Ait-Aoudia, Ramdane Mahiou, Hamza Djebli, El – Hachemi Guerrou, *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
- [6] M. Brown, D.G. Lowe, Recognising panoramas, in proceedings of the Ninth IEEE International Conference on Computer Vision, Vol. 2, pp. 1218-1225, 2003.
- [7] M. Su, W. Hwang, K. Cheng, "Analysis on multi resolution mosaic images", *IEEE transactions on Image Process.*, Vol. 13, No. 07, pp. 952–959, 2004.
- [8] Hemlata Joshi, Mr. KhomLal Sinha, "A Survey on Image Mosaicing Techniques", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 2, February2013*.
- [9] N. Gracias, M. Mahoor, S. Negahdaripour, A. Gleason, "Fast image blending using watersheds and graph cuts", Proceedings of British Machine Vision Conference on Image and Computer Vision, Vol. 27, No. 5, pp. 597–607, 2009.

- [10] Debabrata Ghose, Sangho Park, Naima Kaabouch and William Semke, "Quantitative evaluation of Image Mosaicing in multiple scene categories," *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
- [10] Image Alignment and Stitching by Richard Szelisky, Springer, 2011.
- [12]"The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix". *International Journal of Computer Vision*, Vol. 24, No. 3, pp. 271–300,
- [13] Shishir Maheshwari "image mosaicing using Graph cut method",M.tech Thesis,NIT Rourkela, 2014.
- [14] H. Bay, A. Ess, T. Tuytelaars and . L. Van Gool, "SURF : Speeded Up Robust Features," *International journal of Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008.
- [15] Hemlata Joshi and Mr. KhomLal Sinha, "A Survey on Image Mosaicing Techniques," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 2, no. 2, February 2013.
- [16] Prof. Mayur Dhait, Miss. Rashmi S. Ghavghave, "Image Mosaicing Using Feature Detection Algorithm", *International Journal of Informative & Futuristic Research*, Vol-1 Issue -8, April 2014.
- [17] http://en.wikipedia.org/wiki/Midpoint_circle_algorithm.