

Manifesting a Blackboard Image Restore and Mosaic using Multifeature Registration Algorithm

Priyanka Virendrasinh Jadeja¹, Dr. Dhaval R. Bhojani²

¹*Department of Electronics and Communication Engineering, Darshan Institute of Engineering and Technology, Rajkot, jadejapriyanka111@gmail.com*

²*Department of Electronics and Communication Engineering, Darshan Institute of Engineering and Technology, Rajkot, diet.ec.hod@gmail.com*

Abstract— This paper focuses on stitching the Images of the Document[2] i.e. Document Image Mosaicing. The Image mosaicing is done in order to create a “larger field of view” as opposed to small “field of view” obtained by the camera lens. Also it improves the resolution of the picture. Document[3] image mosaicing (we take blackboard picture as a document here) is the mosaicing of documents (2D or planar mosaicing) so that they could be viewed together without any breaks in them, for example say a complicated derivation that occupied two to three pages of a pdf or paper can be fitted into one scene and viewed together for easier grasping of the idea. Also it could be mosaicing of the blackboard, the large theories or mathematical proofs can be mosaiced into one picture so that we have the complete idea of the theory or proof in ONLY ONE PAPER to be viewed together by readers. Also in this paper we demonstrate certain necessary restoration methods before we mosaic the blackboard image.

Keywords-document image mosaicing; blackboard image mosaicing; image restoration; SIFT; Feature based image registration

I. INTRODUCTION

Rather than observing images from each individual camera one large image containing all of the images' views of the scene activity is desirable when one, larger camera to capture this data is not available. This image which combines all of the images captured by the camera array is generally called a mosaic in the field of computer vision and is achieved by performing an inter-camera projection process to stitch the images together. Visually from an image analyst's perspective a mosaic is simpler to study than multiple images at different viewing angles and with redundant scene data. Moreover, for data exploitation or further processing of the image data a projected mosaic is typically desired over multiple, unaligned images containing overlapping pixel data of the same scene.

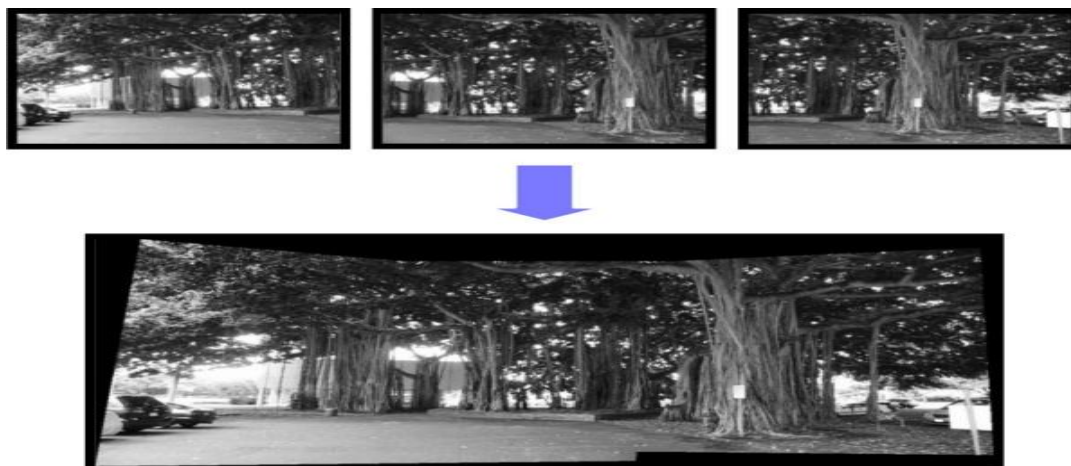


Figure 1. Split Images(Top) and its Mosaic(Bottom)

The Blackboard Image mosaicing comes under the heading of document image mosaic, where in we try to make a mosaic of the different portions of the blackboard pictures. The steps of image mosaicing are as under.

II. IMAGE RESTORATION

Image Restoration is a field of Image Processing which deals with recovering an original and sharp image from a degraded image using a mathematical degradation and restoration model. It is the operation of taking a corrupted/noisy image and estimating the clean original image. Various ways Split Images can go wrong are as follows

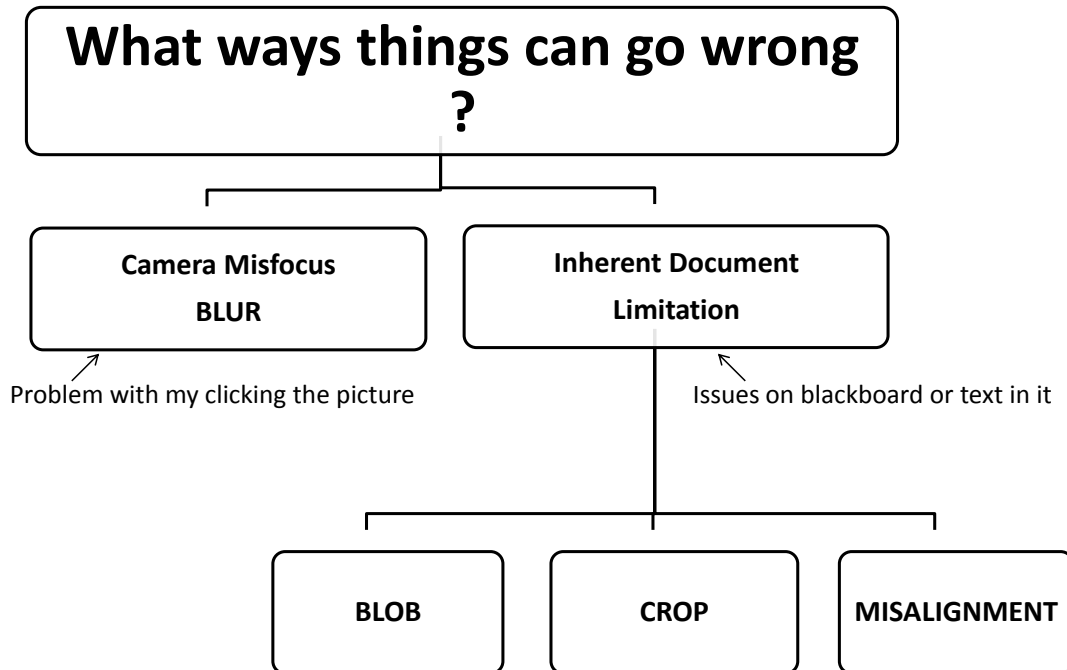


Figure2 Types of image Restoration needed in blackboard image mosaicing

2.1.Deconvolution (deblur)[7]

Blurring [18] is a form of bandwidth reduction of the image due to imperfect image formation process. It can be caused by relative motion between camera and original images. Normally, an image can be degraded using low-pass filters and its noise. This low-pass filter is used to blur/smooth the image using certain functions.

Three main deconvolution algorithms for camera blur are – (1) Blind deconvolution algorithm, (2) Lucy-Richardson Algorithm, (3) Wiener algorithm, We use here Lucy method as it gave us better results than other, The otherwise hit Blind deconvolution method produced unnecessary ringing effect, The result of this is demonstrated below.



Figure3 left – original image, right– blind deconvolution[8]



Figure4 left – original image, right– Lucy Richardson[10] deconvolution using same amount of blurring as in blind deconvolution method above.

2.2 Blob Removal

In the field of computer vision, blob detection refers to mathematical methods that are aimed at detecting regions in a digital image that differ in properties, such as brightness or color, compared to areas surrounding those regions.

2.3 Cropping the edge automatically

Cropping of the border of the blackboard was required as the part of preprocessing, so an algorithm which automatically cropped the rectangular border of blackboard was designed.

2.4 Adjusting Misalignment automatically

Misalignment in image occurs due to imperfection in capturing an image. The image is shifted in the 2d plane i.e. rotated by certain amount automatically by our Preprocessing algorithm.

III. IMPLEMENTATION RESULTS OF IMAGE RESTORATION[6] ALGORITHM

Implementation of Image cropping, blur removal, alignment and blob removal is shown as under using our preprocessing (restoration) algorithm.

3.1. Cropping the edge automatically

Cropping of the border of the blackboard was required as the part of preprocessing, so an algorithm which automatically cropped the rectangular border of blackboard was designed.



Figure5. Image automatic cropping of border

3.2. Blur removal



Figure6 left top – original test image, right top – blurred image, bottom left – deblurring using Lucy Richardson Algorithm (LR), bottom right – deblurring using higher iterative value of LR.

3.3. Adjusting Misalignment automatically

Misalignment in image occurs due to imperfection in capturing an image. The image is shifted in the 2d plane i.e. rotated by certain amount automatically by our Preprocessing algorithm.



Figure7 left top – original test image, right top – edge detected, bottom left – original test image with reference edge, bottom right – aligned image

3.4. Blob Removal[11]

Misalignment in image occurs due to imperfection in capturing an image. The image is shifted in the 2d plane i.e. rotated by certain amount automatically by our Preprocessing algorithm.



Figure8 left top – original test image, right top – gray scale image, middle right - original minus opened image, middle left - opened image, bottom left – contrast adjust image, bottom right – background cleared image

An algorithm was created that which had all the preprocessing schemes mentioned above together. The picture taken as test is a blurred one which is also rotated by few degrees clockwise, the reason being the purpose of checking the algorithm for blurred, rotated, uncropped and blobbed image.

IV. STEPS OF FEATURE[1] BASED IMAGE MOSAICING

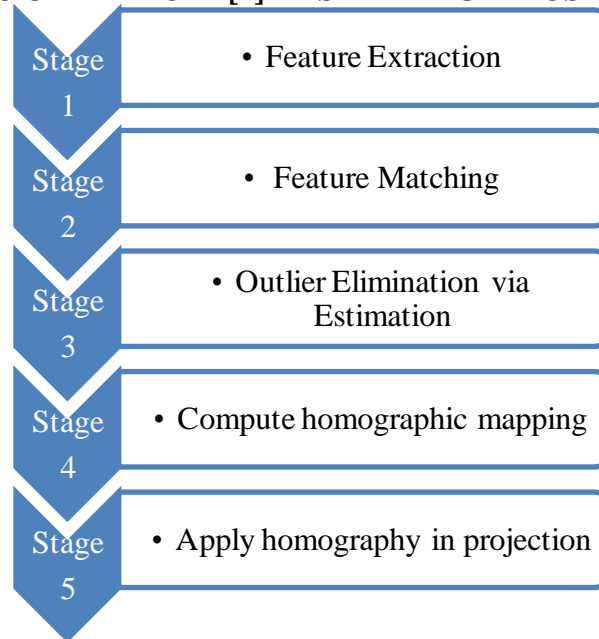


Figure9 Steps of Image Mosaicing

4.1. Feature Extraction

Features[4] are computed in this stage for each image contributing to the formation of the mosaic. Various types of features can be extracted from an image. However, feature types and algorithms used in mosaicing which are examined in this thesis include classical Harris corners, shape based connected component descriptors (CCD), the machine learning based FAST algorithm, and the scale-invariant feature transform (SIFT). In our project we used SIFT Algorithm. First, SIFT key points are computed based on local maxima and minima of pixel neighborhoods that are consistent across multiple difference of Gaussian layers, and key points are identified for various resolutions or octaves of an input image. Difference of Gaussian layers is calculated

4.2. Feature Matching

In the second stage of processing, features are matched between each pair of overlapping images. Feature matching techniques are Cross correlation[13] method and Nearest Neighbor method. We use Cross Correlation method here.

4.3. Outlier Elimination Via Estimation

In the third stage of mosaicing for feature based systems, false matches are detected through an estimation process. The names of Method are as under RANSAC[12] and BAYSAC, We use RANSAC here i.e. RANdom SAMpling Consensus.

4.4. Compute Homographic Mapping

In this stage, the final mapping is computed which will relate coordinates of two overlapping images captured of a common scene. The input to this stage is estimated feature match inliers between two images. A homography is the name of a matrix capable of projectively mapping points in one image to those in another.

4.5. Applying Homography in Projection

To complete the process of mosaicing computed homographies are used to transform the set of individual images captured of a common scene, projecting them as one, final complete image. This

single image containing all of the imaged portions of a single scene is called the mosaic, and this stage of processing is called perspective projection.

V.

VI. IMPLEMENTATION RESULTS OF MOSAICING USING SIFT[5]

In Implementation results below show the split images first of the blackboard, underneath which the image registration is, the blue lines are key points matched using SIFT Algorithm.

5.1. Mosaic Example 1

To complete the process of mosaicing computed homographies are used to transform the set of individual images captured of a common scene, projecting them as one, final complete image. This single image containing all of the imaged portions of a single scene is called the mosaic, and this stage of processing is called perspective projection.



Figure10. Split Images

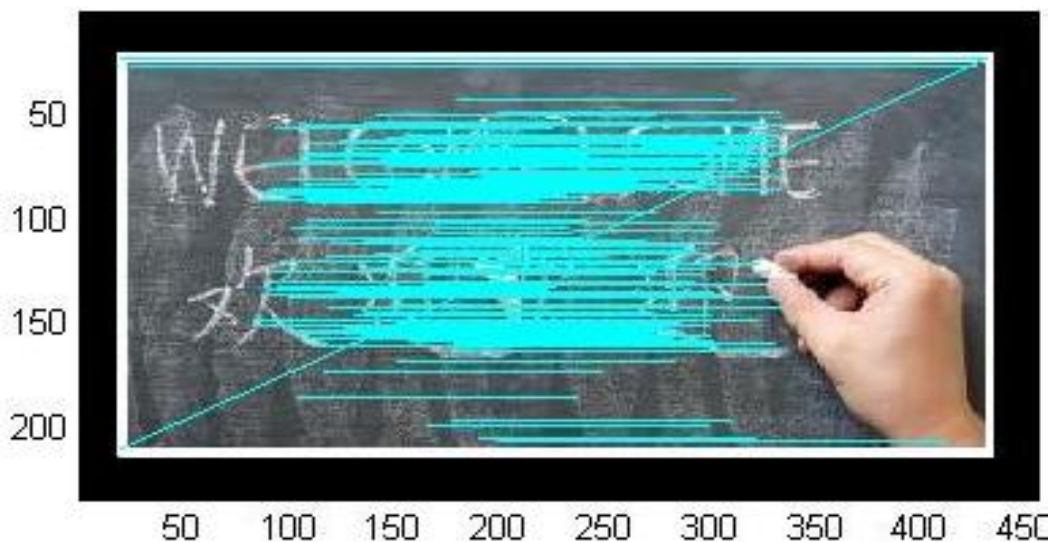


Figure11. KeyPoint Matching



Figure12. Final Mosaiced Image Using our Algorithm

322 keypoints found.
344 keypoints found.
Found 202 matches.
Elapsed time is 14.216829 seconds.

5.2. Mosaic Example 2

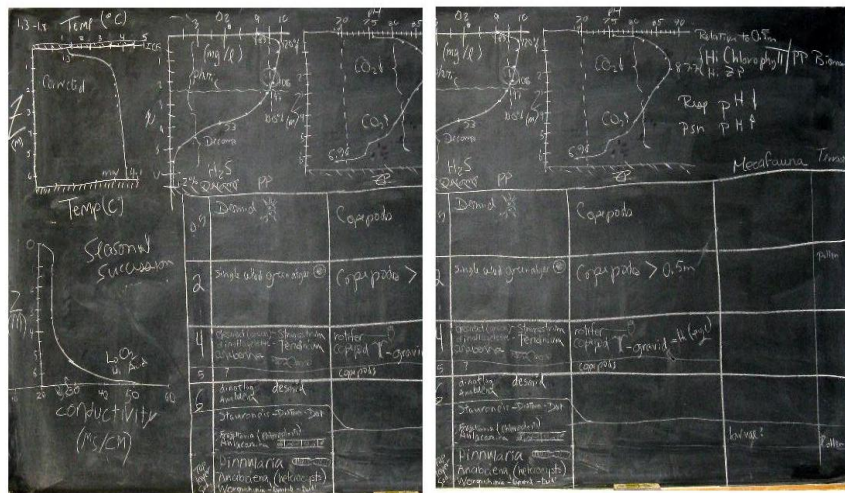


Figure13. Split Images

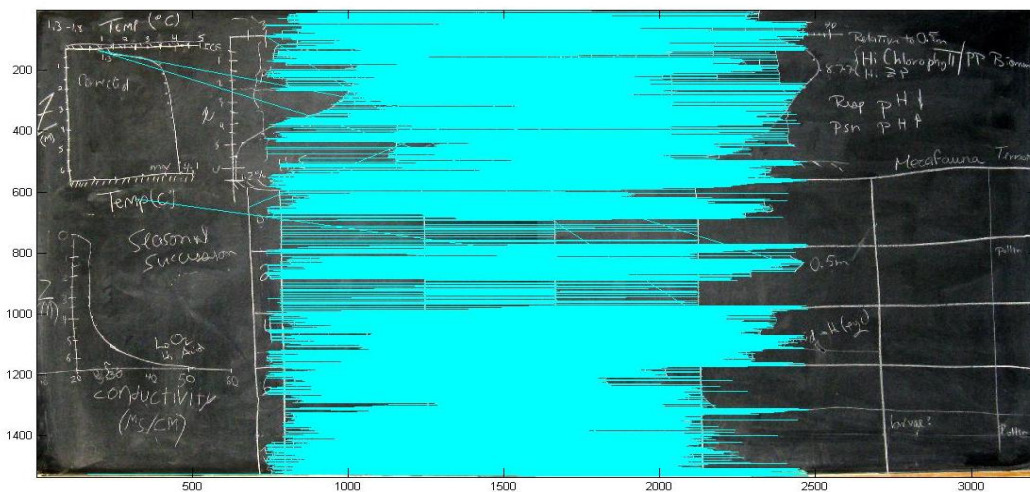


Figure14 KeyPoint Matching

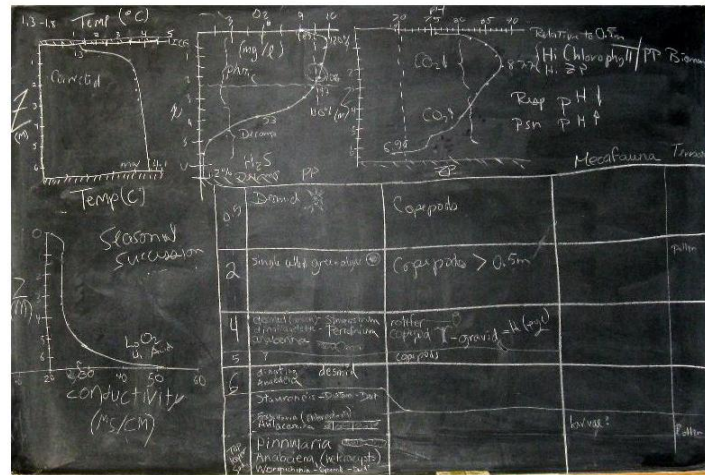


Figure.15 Final Mosaiced Image Using our Algorithm

5.3. Mosaic Example 3

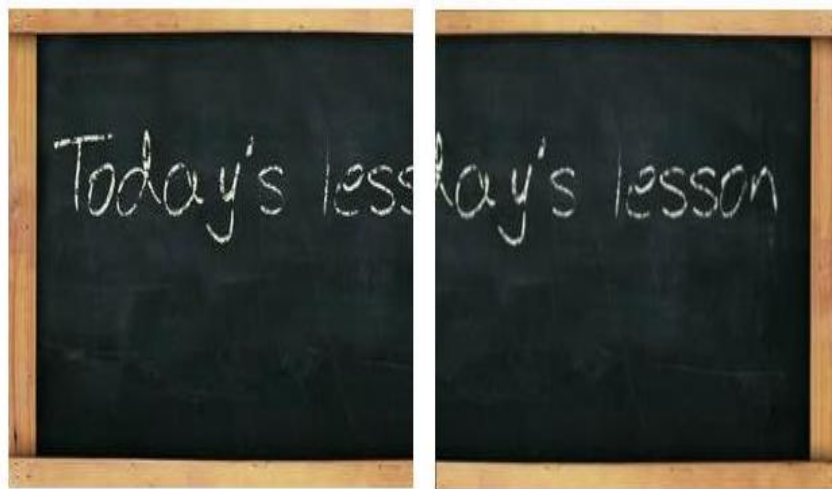


Figure.16 Split Images

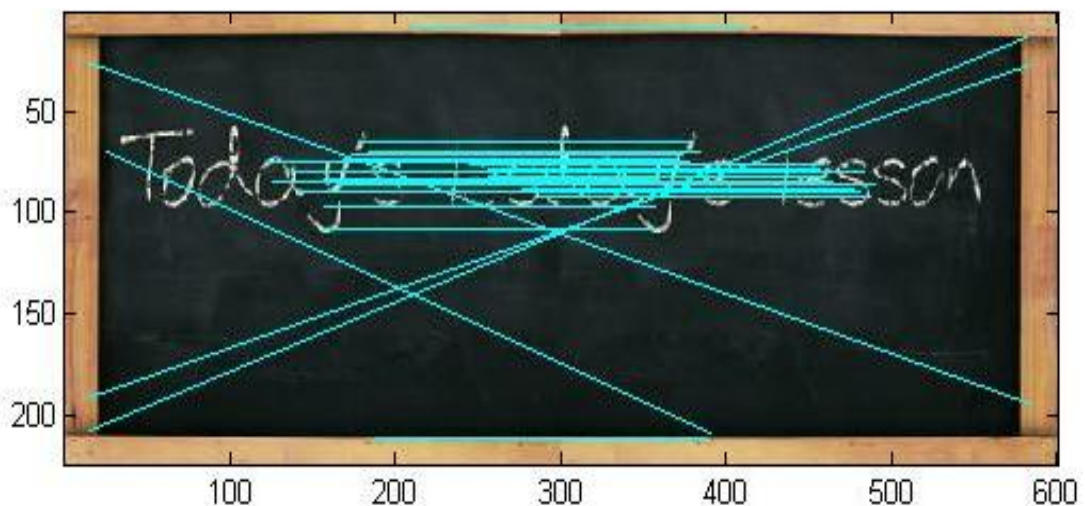


Figure.17 KeyPoint Matching



Figure18. Final Mosaiced Image Using our Algorithm

81 keypoints found.

73 keypoints found.

Found 9 matches.

Elapsed time is 6.598503 seconds.

5.4. Mosaic Example 4

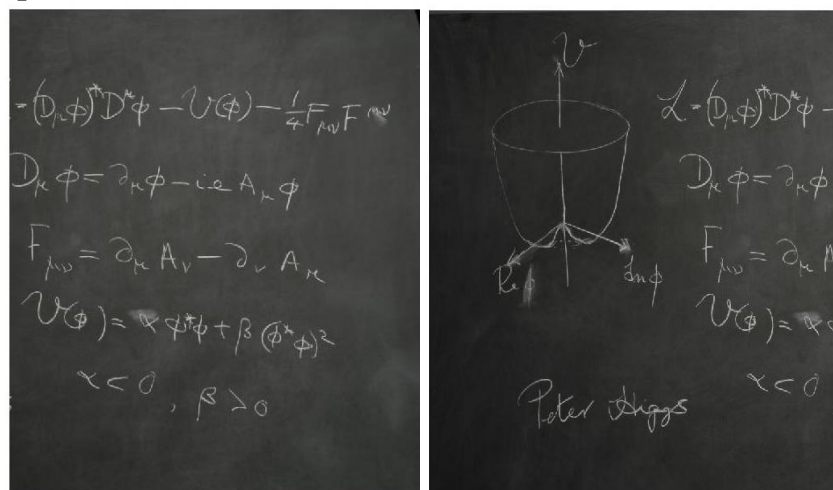


Figure.19 Split Images

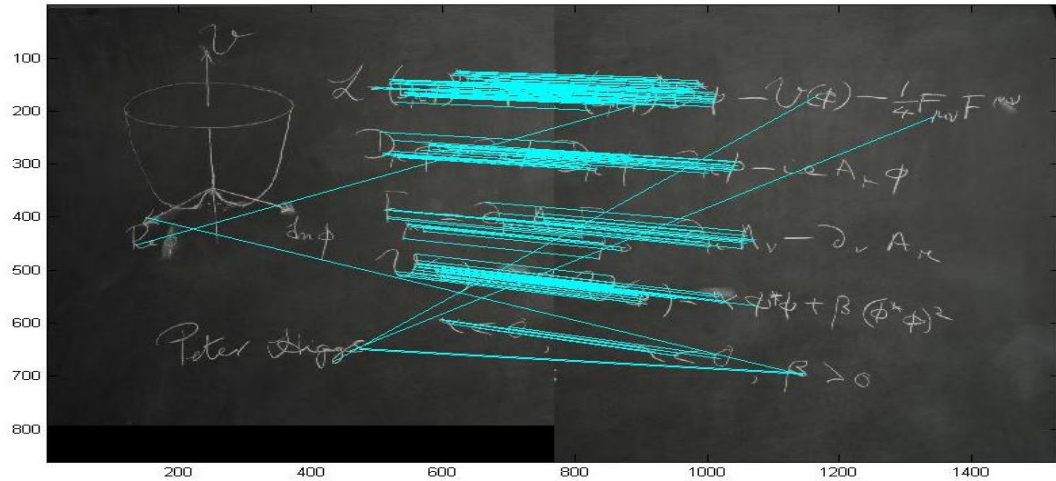


Figure.20 KeyPoint Matching

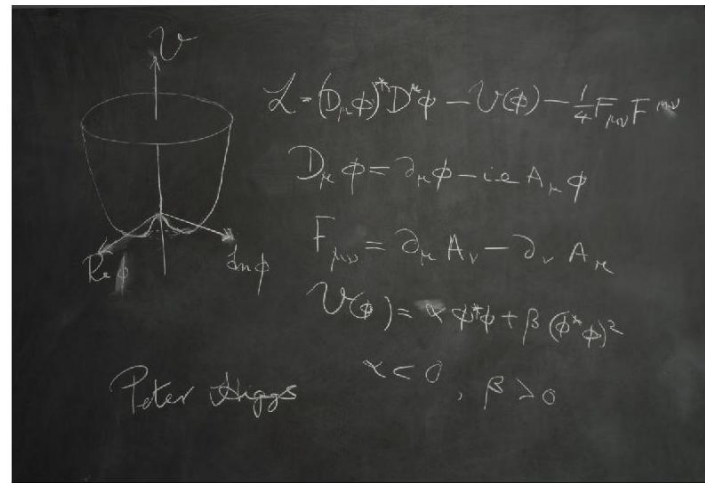


Figure21. Final Mosaiced Image Using our Algorithm

779 keypoints found.

852 keypoints found.

Found 218 matches.

Elapsed time is 16.926492 seconds.

The analysis from above results is as under in form of a table below

Table 1. Comparison of keypoints found, keypoints matched and time elapsed to calculate the mosaic between different images

Test image number	1	2	3	4
Keypoints found in split image 1	322	15058	81	799
Keypoints found in split image 2	344	14094	73	852
Keypoints matched in both the split images	202	7044	9	218
Time elapsed by	14.21 sec	128.70 sec	6.59 sec	16.92 sec

algorithm to Mosaic two split images				
---	--	--	--	--

It can be deduced from the table that as the keypoints found decreases the keypoints matched decreases and eventually the time elapsed to calculate the mosaic also decreases.

VII. CONCLUSION

While thinking about the Image Mosaicing One must think about the quality of results and the complexity of the algorithm. Keeping this thing in mind it is intended here to develop a preprocessing algorithm for split images which will crop, deblur, remove blob and align the split images if needed and then comes the SIFT algorithm for feature extraction feature matching and blending.

VIII. FUTURE WORK

The future work may extend the results obtained in this dissertation, which includes mosaicing a 3D image.

REFERENCES

- [1] Ballard, Bret Stephen, "Feature Based Image Mosaicing Using Region Of Interest For Wide Area Surveillance Camera Arrays With Known Camera Ordering" in University of Dayton, Dayton, Ohio, May 2011.
- [2] Ardian Philip Whichello and Hong Yan, "Document Image Mosaicing", Department of electrical Engineering, University of Sydney, Australia.
- [3] Jian Liang, Daniel DeMenthon, David Doemann, "Camera-based document image mosaicing" Language and Media Processing, published 18th international conference on pattern recognition in IEEE, University of Maryland, 2006.
- [4] Satya Prakash Mallick, "Feature based image mosaicing", San Diego.
- [5] Jyoti Joglekar, Shirish s. Gedam, "Matching with SIFT Features – A probabilistic approach", in IAPRS journal, Vol. XXXVIII, Part 3B, Mumbai, India, Sept, 2010.
- [6] Jotirmoy Banerjee, Anoop M. Namboodiri, C V Jawahar, "Contextual restoration of severely degraded images", IIT Hyderabad, India.
- [7] Anant Levin, Yair Weiss, Fredo Durand, William T. Freeman, "Understanding and evaluating blind deconvolution algorithm", Hebrew university.
- [8] D. Srinivasa Rao, K Selvani Deepthi, K Moni Sushma Deep, "Application of blind deconvolution algorithm for image registration", in International Journal of Engineering science and technology (IJEST), Vol. 3, No. 3, Vishakhapatnam, India, March 2011.
- [9] Anant Levin, Rob Fergus, Fredo Durand, William T Freeman, "Deconvolution using natural image priors", Massachusetts.
- [10] Swati Sharma, Shipra Sharma, Rajesh Mehra, "Restoration using modified Lucy Richardson algorithm in the presence of gaussian and motion blur", in Advance in electronics and electrical engineering, ISSN 2231-1297, Vol.3, number 8, pp. 1063-1070, Punjab, India.
- [11] Stack exchange website, <http://dsp.stackexchange.com/questions/1932/what-are-the-best-algorithms-for-document-image-thresholding-in-this-example>
- [12] Konstantinos G. Derpanis, "Overview of RANSAC algorithm", may 13, California.
- [13] Tomas Petricek, Tomas Svoboda, "Matching by normalized cross correlation reimplementation, comparison to invariant features" Czech Technical University, Sept9,2010.