ENHANCEMENTS FOR DIGITAL IMAGING OF GUSSET PLATE CONNECTIONS: FISHEYE AND IMAGE STITCHING

Final Report

SPR 304-581

ENHANCEMENTS FOR DIGITAL IMAGING OF GUSSET PLATE CONNECTIONS: FISHEYE AND IMAGE STITCHING

Final Report

SPR 304-581

by

O. Tugrul Turan, Ph.D.
and
Christopher Higgins, Ph.D., P.E.

for

Oregon Department of Transportation
Bridge Engineering Section
4040 Fairview Industrial Drive
Salem OR 97302-1142

and

Federal Highway Administration
400 Seventh Street, SW
Washington, DC 20590-0003

September 2011

Technical Report Documentation Page

1. Report No. FHWA-OR-RD-12-03	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Enhancements for Digital Imaging ff Gusset Plate Connections: Fisheye and Image Stitching		5. Report Date September 2011 6. Performing Organization Code
7. Author(s) O. Tugrul Turan and Christopher Higgins		8. Performing Organization Report No.
9. Performing Organization Name and Address Oregon State University Department of Civil and Construction Engineering 220 Owen Hall Corvallis, OR 97331		10. Work Unit No. (TRAIS) 11. Contract or Grant No. SPR 304-581
12. Sponsoring Agency Name and Address Oregon Department of Transportation Bridge Engineering Unit and Federal Highway Administration 4040 Fairview Industrial Drive 400 Seventh Street, SW Salem, OR 97302-1142 Washington, DC 20590-0003		13. Type of Report and Period Covered Final Report 14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This report describes techniques to convert fisheye digital images of gusset plates to perspective images (defish). It also describes methods to stitch together partial images of a gusset plate into a composite of the complete gusset plate. The converted images can be used in rectification and metrification procedures to allow geometric measurements of the connection plates from the processed images. Software applications were written using Matlab to perform the image transformations.

17. Key Words gusset plate connections, bridge inspection, image processing		18. Distribution Statement Copies available from NTIS, and online at http://www.oregon.gov/ODOT/TD/TP_RES/	
19. Security Classification (of this report) Unclassified	20. Security Classification (of this page) Unclassified	21. No. of Pages 50	22. Price

Technical Report Form DOT F 1700.7 (8-72) Reproduction of completed page authorized Printed on recycled paper

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS					APPROXIMATE CONVERSIONS FROM SI UNITS				
Symbol	When You Know	Multiply By	To Find	Symbol	Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH					**LENGTH**		
in	inches	25.4	millimeters	mm	mm	millimeters	0.039	inches	in
ft	feet	0.305	meters	m	m	meters	3.28	feet	ft
yd	yards	0.914	meters	m	m	meters	1.09	yards	yd
mi	miles	1.61	kilometers	km	km	kilometers	0.621	miles	mi
		AREA					**AREA**		
in^2	square inches	645.2	millimeters squared	mm^2	mm^2	millimeters squared	0.0016	square inches	in^2
ft^2	square feet	0.093	meters squared	m^2	m^2	meters squared	10.764	square feet	ft^2
yd^2	square yards	0.836	meters squared	m^2	m^2	meters squared	1.196	square yards	yd^2
ac	acres	0.405	hectares	ha	ha	hectares	2.47	acres	ac
mi^2	square miles	2.59	kilometers squared	km^2	km^2	kilometers squared	0.386	square miles	mi^2
		VOLUME					**VOLUME**		
fl oz	fluid ounces	29.57	milliliters	ml	ml	milliliters	0.034	fluid ounces	fl oz
gal	gallons	3.785	liters	L	L	liters	0.264	gallons	gal
ft^3	cubic feet	0.028	meters cubed	m^3	m^3	meters cubed	35.315	cubic feet	ft^3
yd^3	cubic yards	0.765	meters cubed	m^3	m^3	meters cubed	1.308	cubic yards	yd^3
NOTE: Volumes greater than 1000 L shall be shown in m^3.									
		MASS					**MASS**		
oz	ounces	28.35	grams	g	g	grams	0.035	ounces	oz
lb	pounds	0.454	kilograms	kg	kg	kilograms	2.205	pounds	lb
T	short tons (2000 lb)	0.907	megagrams	Mg	Mg	megagrams	1.102	short tons (2000 lb)	T
		TEMPERATURE (exact)					**TEMPERATURE (exact)**		
°F	Fahrenheit	(F-32)/1.8	Celsius	°C	°C	Celsius	1.8C+32	Fahrenheit	°F

*SI is the symbol for the International System of Measurement

ACKNOWLEDGEMENTS

The authors would like to acknowledge the input and support of Messrs. Bert Hartman, Jonathan Rooper, Jeff Swanstrom, Richard Groff, and Steven Soltesz of the Oregon Department of Transportation. Mr. Quang Nguyen and Mr. Jason Killian of Oregon State University provided assistance during early development of the image processing algorithms.

DISCLAIMER

ENHANCEMENTS FOR DIGITAL IMAGING OF GUSSET PLATE CONNECTIONS: FISHEYE AND IMAGE STITCHING

TABLE OF CONTENTS

LIST OF PHOTOS/FIGURES

1.0 INTRODUCTION

Bridge inspection is a key component in bridge safety and management. Presently, using conventional methods to collect and archive field data for the evaluation of structures is often subjective and very time consuming. Efficient and economical visual inspection techniques are needed to improve quantitative field data collection and increase inspection production time. This is particularly true for gusset plate connections, which have become a recent focus of transportation agencies after the collapse of the I35W Bridge in Minneapolis, MN in 2007.

Visual inspection methods are beginning to deploy supporting technologies (*McCrea et al. 2002*). One such technology is digital image processing, which has been utilized in various areas in civil engineering. Several researchers implemented digital image technologies for assessment and inspection of steel, concrete and reinforced concrete structures. Although using digital image processing to detect a crack on a concrete surface is difficult due to voids, blemishes, shades and shape of the cracks, it attracts broad interest and has been studied by several researches such as Ito et al. (*2002*), Dare et al. (*2002*), Hutchinson and Chen (2006), Fujita et al. (*2006*), Yamaguchi and Hashimoto (*2006*), Sinha and Fieguth (*2006*), Yamaguchi et al. (*2008*), Yamaguchi and Hashimoto (*2009*).

Lee and Chang (*2005*) used digital image processing for the assessment of rust defects on steel bridges. Liu et al. (*2006*) utilized image processing methods to detect rivets for aircraft lap joints and recently, Higgins et al. (*2010*) has used digital image processing to enable rapid and accurate collection of field measurements of gusset plates.

In the paper by Higgins et al. (*2010*), close-range photogrammetry techniques were used to rectify field collected images taken with consumer grade cameras to produce metrified orthographic photos of gusset plate connections. Flat field lenses were used in order to minimize the effect of barrel distortion and/or pin-cushion. While these methods provide improvements in the amount and quality of geometric data obtained from gusset plates, improvements are reported here that allow the use of fisheye lenses and stitching together of multiple images. These enhancements are useful since images are most often taken by climbing on the structure or from a snooper. In these cases, it is sometimes difficult to capture the whole gusset plate in one picture with a short focal length lens at close distance. Furthermore, sometimes due to the obstacles such as another gusset plate, floor beam, lateral bracing, utility pipe etc., as shown in Fig.1.1a; it is impossible to capture the whole gusset plate in one picture with a flat field lens. For these cases, a fisheye lens can be utilized to capture a large part of the structure with short focal lengths as shown in Fig.1.1b. Alternatively, the entire gusset plate can be created from separate images of parts of the connection by stitching the individual images together to create a composite image.

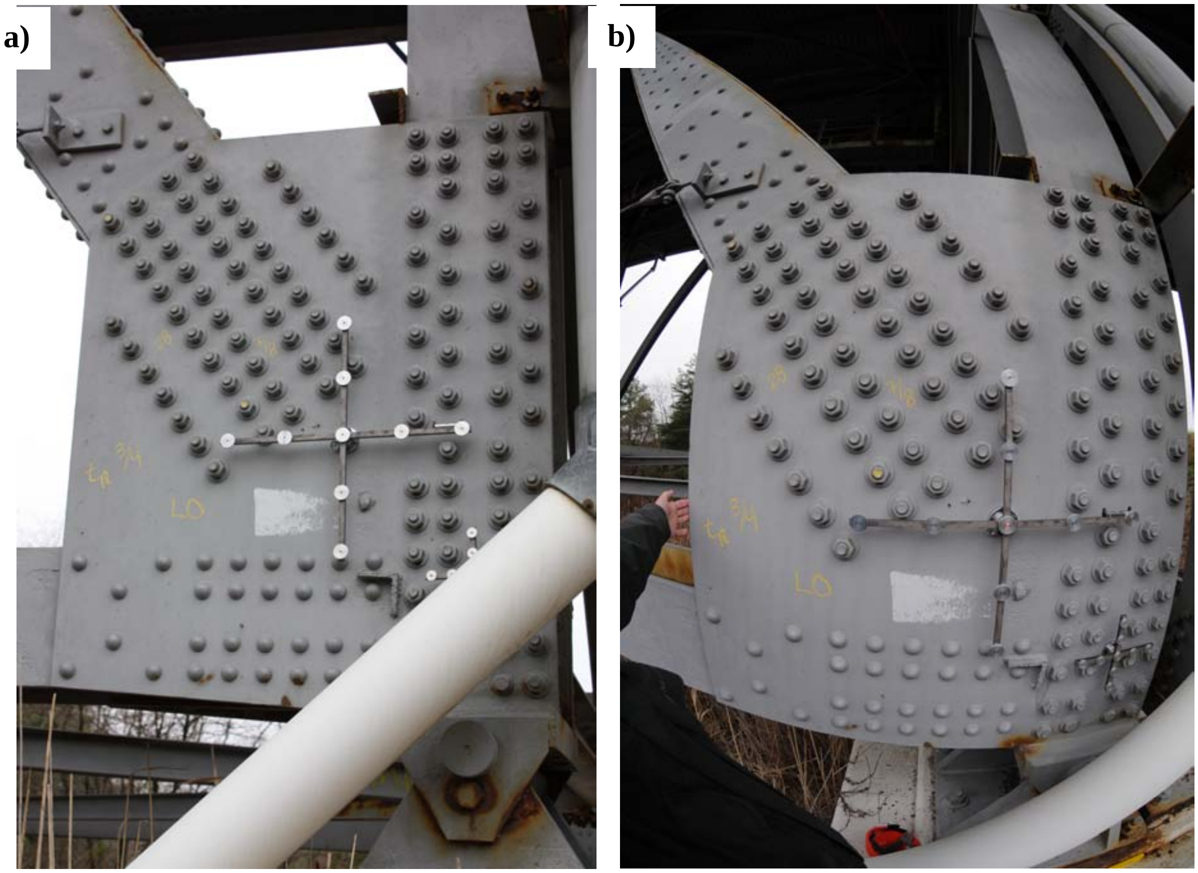


Figure 1.1: An image of a gusset captured by a) flat field lens b) fisheye lens. Note the downspout obstruction

2.0 OBJECTIVES

The objectives of this study were to:

1) Develop methods to convert fisheye images to perspective images (defish) so that the converted images can be used in rectification and metrification procedures as described in Higgins et al. (*2010*).
2) Develop methods to stitch together partial images of a gusset plate into a composite of the complete gusset plate so that the converted images can be used in rectification and metrification procedures as described in Higgins et al. (*2010*).

These methods are described in the subsequent sections, and appendices are provided that guide a user through the implementation process.

3.0 FISHEYE IMAGES

Fisheye lenses are lenses that can provide wide field of view with a very short focal length. These lenses can be used to capture 180^o or larger field of view with a single camera, from one stationary point and at a single moment (*Abraham and Forstner 2005*). Compared to the real panoramic cameras, they are cheap and can be combined with conventional cameras (*Schneider et al. 2009*).

In fisheye projection models, generally a sphere is projected on a plane and depending on their projection geometry, they can be classified in four different categories: Equidistant Projection, Equisolid-angle Projection, Othographic Projection and Stereographic Projection (*Ray 1994*).

An object in space can be described in the camera coordinate system as (x, y, z) and the same object can be described in the image coordinate system as (x', y') where x and y axes and x' and y' axes are parallel. Using this notation, the incidence angle, α , image radius, r', and the mapping function of an undistorted perspective image can be written as (*Schneider et al. 2009*):

$$\tan(\alpha) = \frac{\sqrt{x^2 + y^2}}{z} \tag{1}$$

$$r' = \sqrt{x'^2 + y'^2} \tag{2}$$

$$r' = c_1 \tan(\alpha) \tag{3}$$

$$\frac{x_1'}{y_1'} = \frac{x}{y} \tag{4}$$

where c_1 is the principal distance of the undistorted perspective image. Using Eqs.3-4 (x', y') can be written as:

$$\begin{aligned} x_1' &= \frac{r'}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}} \\ y_1' &= \frac{r'}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \end{aligned} \tag{5}$$

Utilizing Eq.1, 3; (x', y') can also be written as:

$$x_1' = \frac{c_1 \frac{\sqrt{x^2 + y^2}}{z}}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}}$$
$$y_1' = \frac{c_1 \frac{\sqrt{x^2 + y^2}}{z}}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \tag{6}$$

For the undistorted fisheye projection models, the mapping functions and the projection equations can be written (with correction parameters ignored) as (*Schneider et al. 2009*):

Equidistant Projection:

$$r' = c_2 \alpha \tag{7}$$

$$x_2' = c_2 \frac{\arctan(\frac{\sqrt{x^2 + y^2}}{z})}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}}$$
$$y_2' = c_2 \frac{\arctan(\frac{\sqrt{x^2 + y^2}}{z})}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \tag{8}$$

Equisolid-angle Projection:

$$r' = 2c_2 \sin(\frac{\alpha}{2}) \tag{9}$$

$$x_2' = 2c_2 \frac{\sin(\frac{1}{2}\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}}$$
$$y_2' = 2c_2 \frac{\sin(\frac{1}{2}\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \tag{10}$$

Orthographic Projection:

$$r' = c_2 \sin(\alpha) \tag{11}$$

$$x_2^{'} = c_2 \frac{\sin(\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}}$$

$$y_2^{'} = c_2 \frac{\sin(\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \tag{12}$$

Stereographic Projection:

$$r' = c_2 \sin(\alpha) \tag{13}$$

$$x_2^{'} = 2c_2 \frac{\tan(\frac{1}{2}\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{y}{x}\right)^2 + 1}}$$

$$y_2^{'} = 2c_2 \frac{\tan(\frac{1}{2}\arctan(\frac{\sqrt{x^2 + y^2}}{z}))}{\sqrt{\left(\frac{x}{y}\right)^2 + 1}} \tag{14}$$

3.1 DEFISHING EQUATIONS

In order to convert fisheye images to perspective images, fisheye images are assumed to be undistorted (correction parameters are ignored). For different fisheye projection models, fisheye projection equations were mapped to perspective image coordinates using real world image coordinates and Eqs. 5-14 as:

Equidistant Projection:

$$x_1^{'} = c_1 \frac{\tan(\frac{x_2^{'}}{c_2}\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1})}{\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1}}$$
$$y_1^{'} = c_1 \frac{\tan(\frac{y_2^{'}}{c_2}\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1})}{\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1}} \quad (15)$$

Equisolid-angle Projection:

$$x_1^{'} = c_1 \frac{\tan(2\sin^{-1}(\frac{x_2^{'}}{2c_2}\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1}))}{\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1}}$$
$$y_1^{'} = c_1 \frac{\tan(2\sin^{-1}(\frac{y_2^{'}}{2c_2}\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1}))}{\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1}} \quad (16)$$

Orthographic Projection:

$$x_1^{'} = c_1 \frac{\tan(\sin^{-1}(\frac{x_2^{'}}{c_2}\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1}))}{\sqrt{\left(\frac{y_2^{'}}{x_2^{'}}\right)^2+1}}$$
$$y_1^{'} = c_1 \frac{\tan(\sin^{-1}(\frac{y_2^{'}}{c_2}\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1}))}{\sqrt{\left(\frac{x_2^{'}}{y_2^{'}}\right)^2+1}} \quad (17)$$

Stereographic Projection:

$$x_1' = c_1 \frac{\tan(2\arctan(\frac{x_2'}{2c_2}\sqrt{\left(\frac{y_2'}{x_2'}\right)^2+1}))}{\sqrt{\left(\frac{y_2'}{x_2'}\right)^2+1}}$$

$$y_1' = c_1 \frac{\tan(2\arctan(\frac{y_2'}{2c_2}\sqrt{\left(\frac{x_2'}{y_2'}\right)^2+1}))}{\sqrt{\left(\frac{x_2'}{y_2'}\right)^2+1}} \qquad (18)$$

Using Eqs.15-18, a Matlab image processing program was written. The test images were collected by a Nikkon D300 camera with a Nikkor 10.5 mm lens which uses Equisolid-angle Projection. This configuration was used as the default values in the Matlab program. The properties of the camera and the fisheye lens are: Maximum resolution= 4288x2848, Sensor size=23.6 mm x 15.8 mm (sensor type is CMOS), focal length=10.56 mm. The principal distance in the x (c_{2X}) and y (c_{2Y}) directions are calculated as 1918.70 pixels (maximum resolution in the x direction (4288) is divided by the sensor size in the x direction (23.6 mm) and multiplied by the focal length (10.56 mm)) and 1903.44 pixel (maximum resolution in the y direction (2848) is divided by the sensor size in the x direction (15.8 mm) and multiplied by the focal length (10.56 mm)). The principal distance of the perspective image (c_1) was assumed as 600 pixels and established as the default in the program. These projection type and coefficient values can be changed in the program.

The program first creates an array with the same size of the input image (fisheye image) with zeros (black) in the three channels (RGB (red green and blue). In the test images, the array size is 4288x2848x3. For every pixel location in the defished image, using Eqs.15-18 and the principal distances defined by the user; the corresponding pixel location in the fisheye image is calculated. The RGB values in the fisheye image location are copied to the defished image location. If the fisheye location is out of the fisheye image index that pixel is left blank (black). Thus, for every defished image location pixel values, the fisheye image location pixel values are copied and the defished image is created. Defished image of Fig. 3.1b is shown in Fig. 3.1b-c as an example. Multiple images can be selected and when all the images are defished, a message dialogue box pops up that reports the number of images that were defished. In order to demonstrate the effect of the coefficient c_1, different c_1 values were used and the results are shown in Figs. 3.2-3.9. A walkthrough of the program is provided in Appendix A - User Guide for Fisheye Images. Once the images are defished, they can be rectified using the methods described in Higgins et al. [2010]. However, as a note of caution, because fisheye images are generally taken at close range, the target standoff distance is important for metrification. Thus the camera distance to the gusset plate should be measured and used in the standoff correction feature of the rectification process.

a)  b) 

c) 

Figure 3.1: a) Original fisheye image b) defished image c) defished and zoomed image of the gusset.

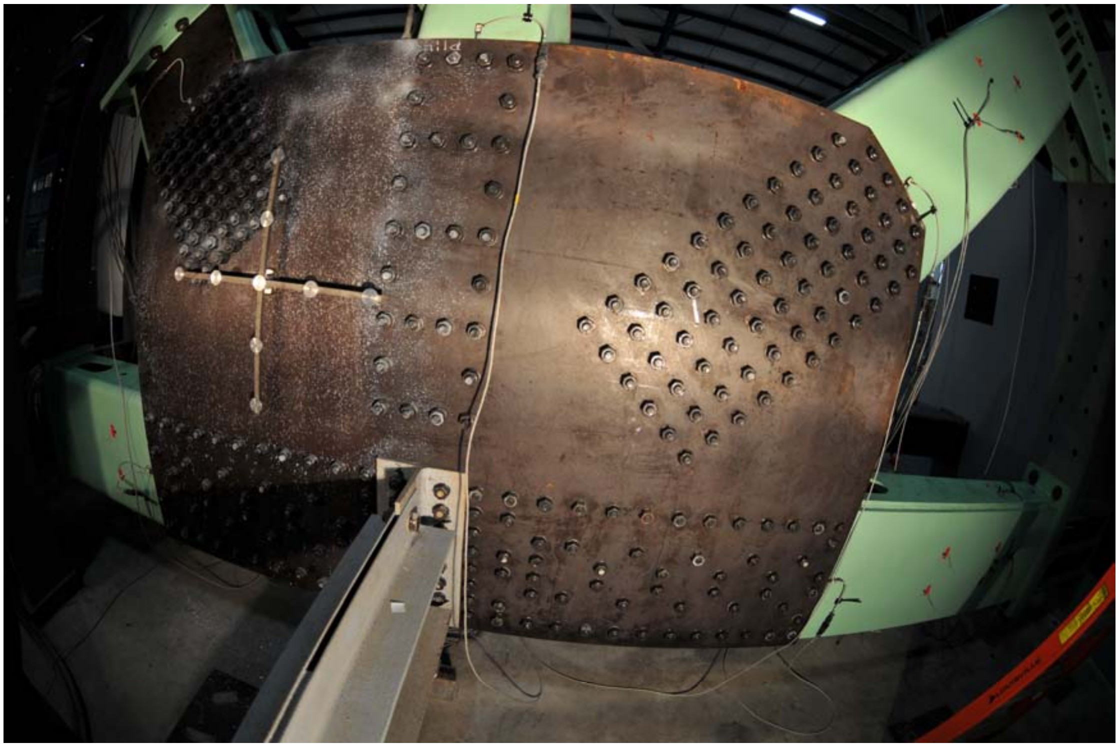


Figure 3.2: Original fisheye image of a gusset plate connection.

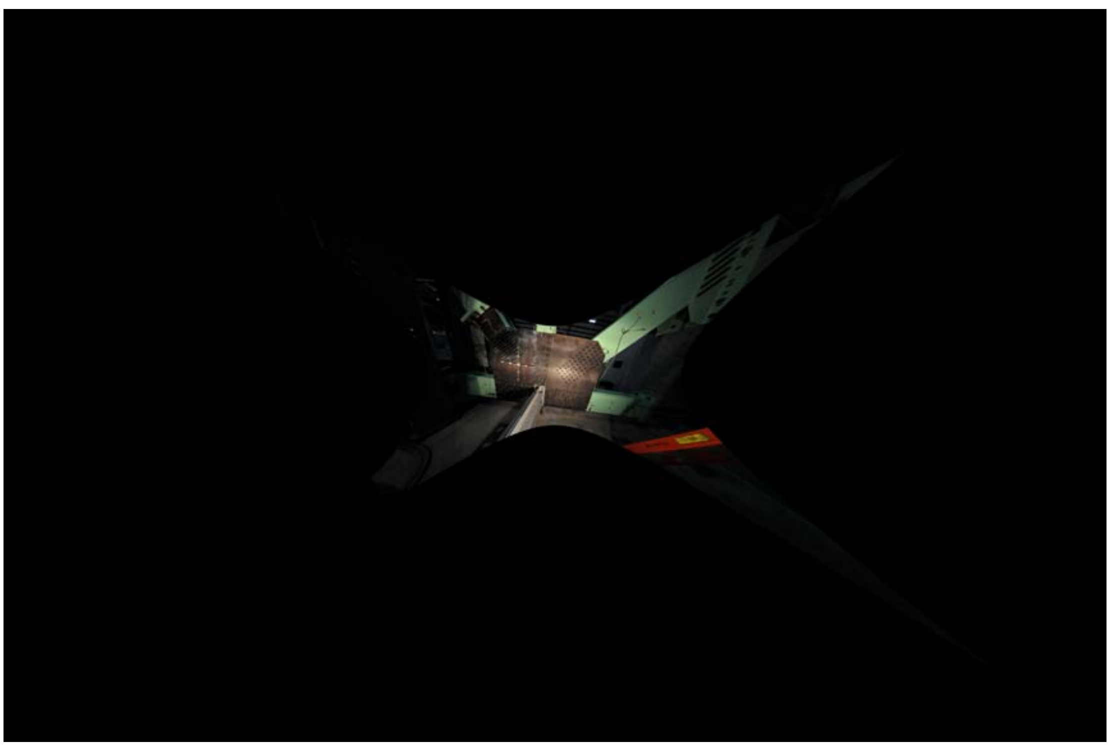


Figure 3.3: Defished image of Fig. 3.2 with c_1 =200 pixel

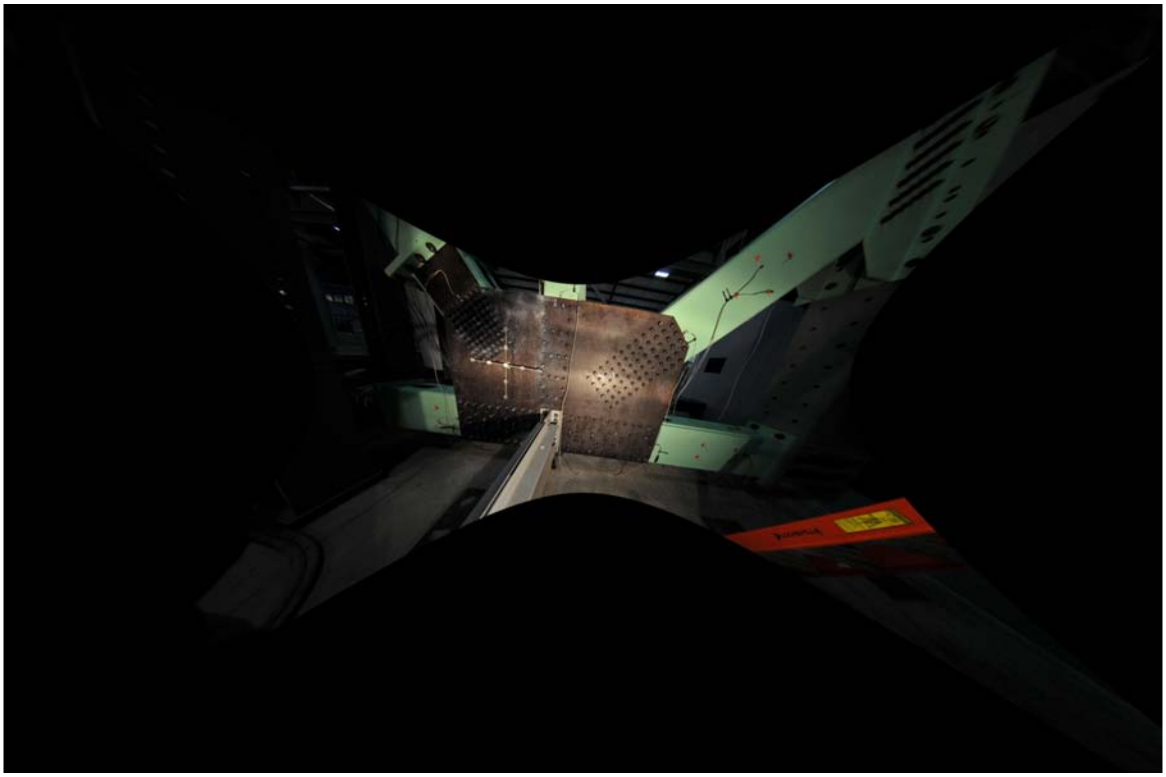


Figure 3.4: Defished image of Fig. 3.2 with c_1 =400 pixel

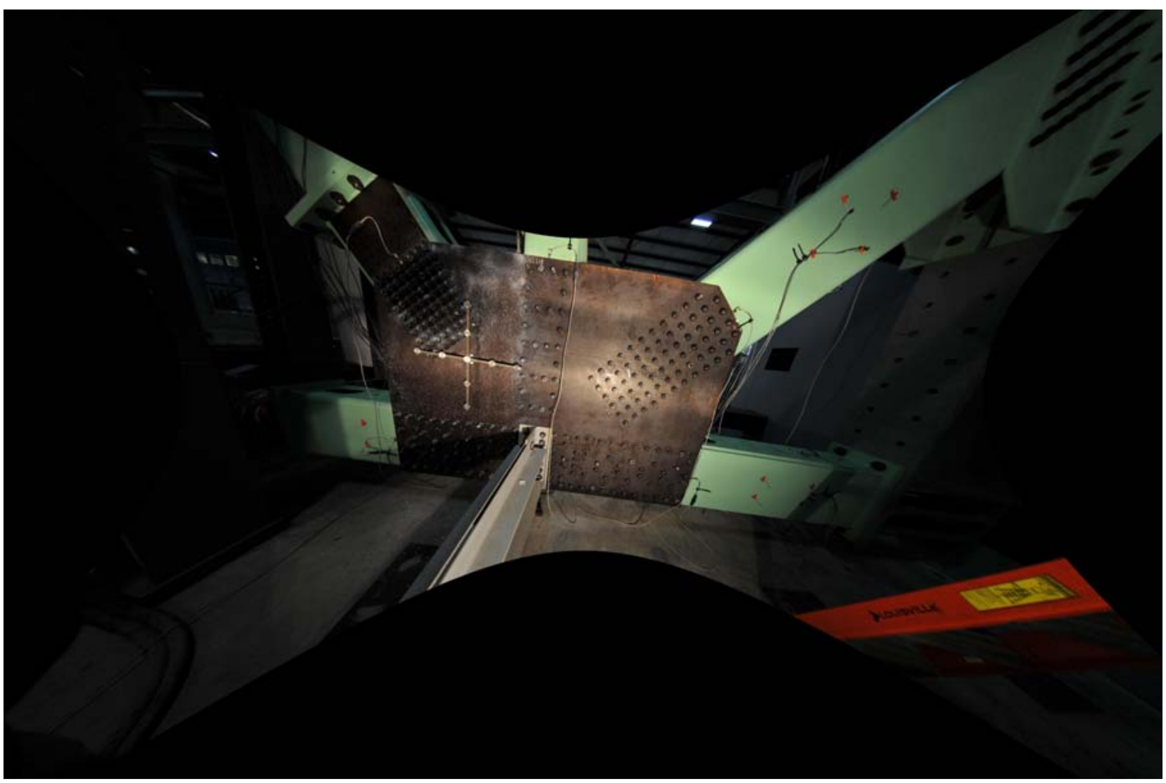


Figure 3.5: Defished image of Fig. 3.2 with c_1 =600 pixel

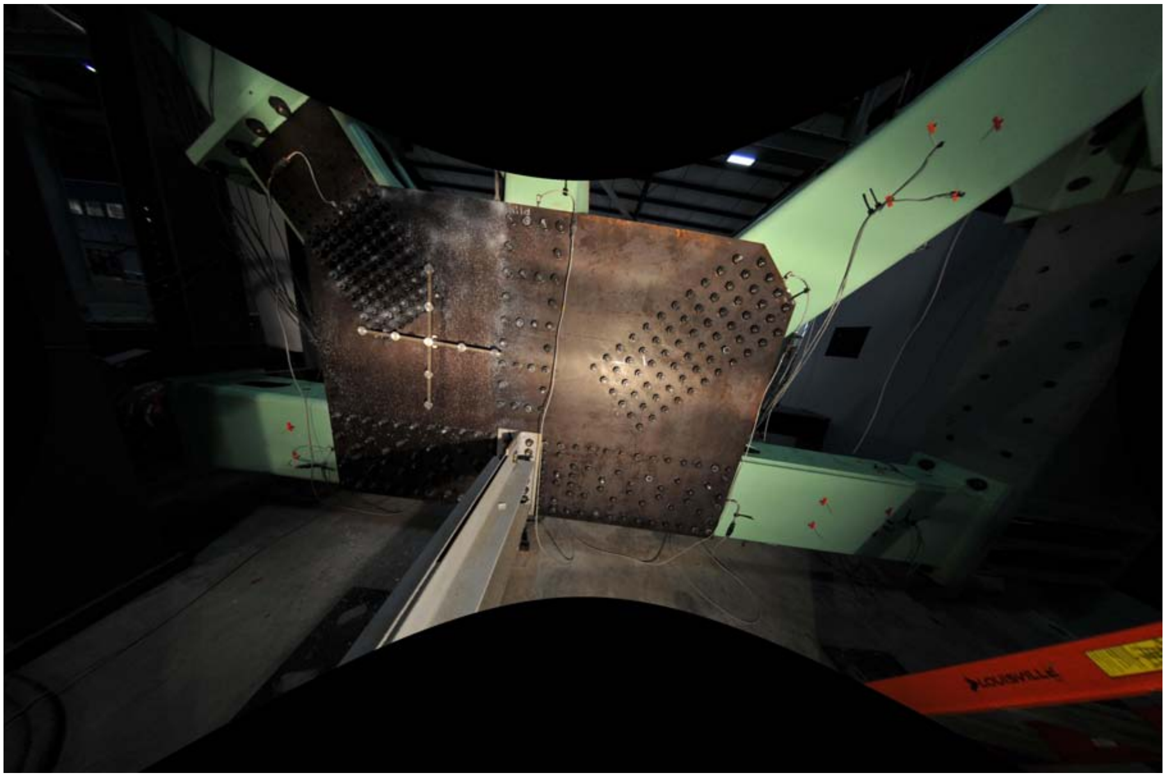


Figure 3.6: Defished image of Fig. 3.2 with c_1 =800 pixel

Figure 3.7: Defished image of Fig. 3.2 with c_1 =1000 pixel

Figure 3.8: Defished image of Fig. 3.2 with c_1 =1200 pixel

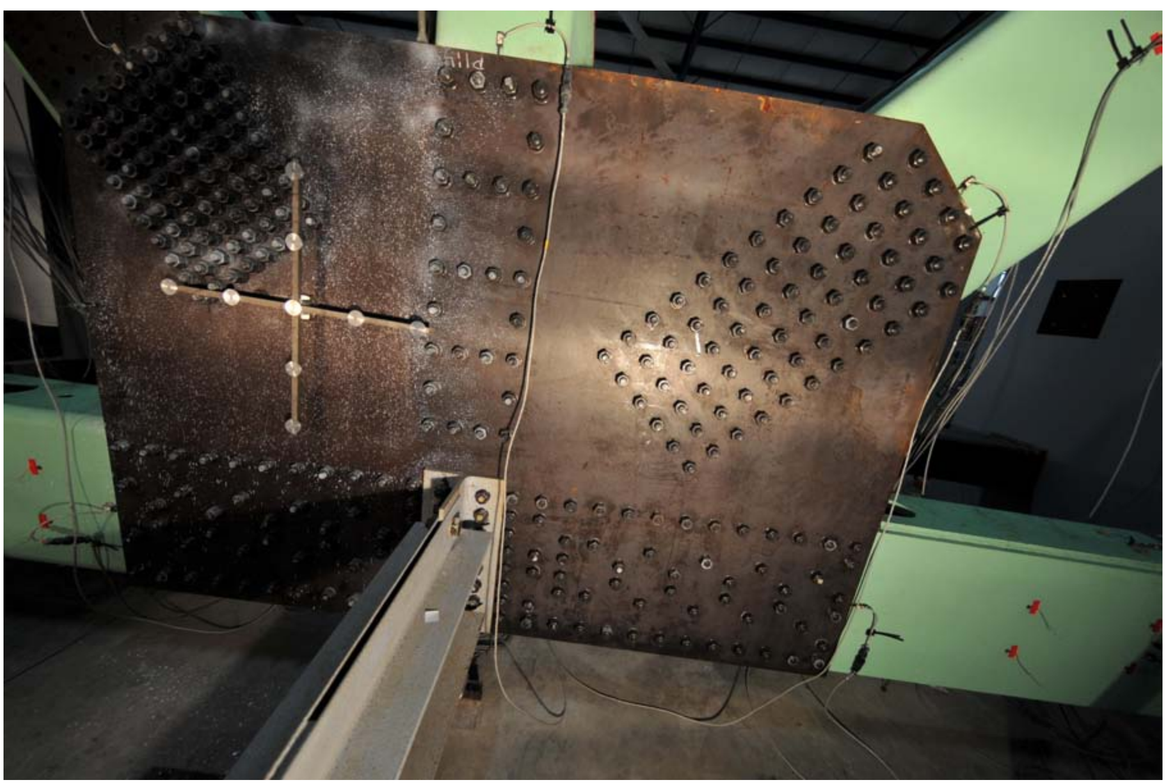


Figure 3.9: Defished image of Fig. 3.3 with c_1 =1500 pixel

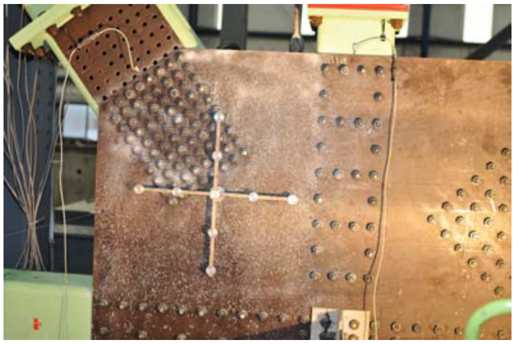
4.0 STITCHING MULTIPLE IMAGES

Sometimes it is not possible to capture an entire object in a single image. In this case, multiple images of the object are taken and these can be assembled to produce a single composite image. To do this, and retain the metrification quality of the images, a stitching program was created using Matlab. The program requires the add-on Image Processing Toolbox for Matlab.

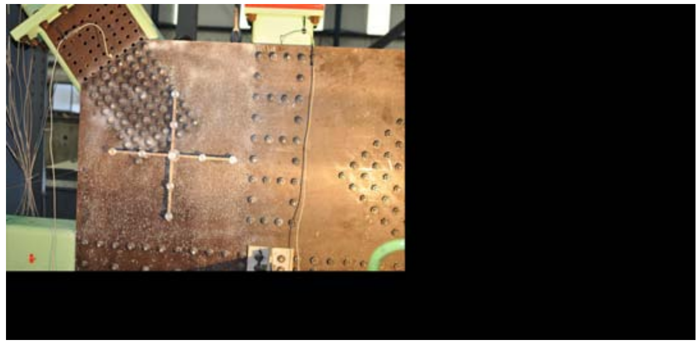
In order to stitch two images together, one of the images (Image 1) is selected as the base image and the other one (Image 2) is selected as the image that needs to be transformed to integrate with the base image (Image 1). In order to transform Image 2, points of correspondence between the two images are used to create a transformation matrix that is applied to Image 2. The Matlab command *cpselect* is used to choose the corresponding image points in Image 1 and Image 2. Using the Matlab command *cp2tform* and the corresponding image points in the two images, a transformation matrix is created. Two different programs were developed that use alternative transformations: 1) Nonreflective similarity transformation (translation, rotation, and scaling are applied. Straight lines remain straight, and parallel lines are still parallel); and 2) Projective transformation (Straight lines remain straight, however parallel lines converge toward vanishing points) (Matlab 2010b, Mathworks). For Nonreflective similarity transformation, at least two corresponding points in the images are needed and rectified images are used to stitch to each other. For Projective transformation, at least four points are needed and the images may or may not be rectified before stitching.

After creating the transformation matrix, the stitched image size is calculated. Two same size black (RGB=0,0,0) images are created and the base image and the transformed images are pasted onto these images as shown in Figs. 4.1 and 4.2. The transferred image is subtracted from the base image and the subtracted image is obtained as shown in Fig. 4.3. The subtracted image is added to the transformed image and the composite stitched image is obtained as shown in Fig. 4.4. The procedure for adding and subtracting image pieces is summarized in Fig. 4.5.

If obstructions occur in the composite image, these can be removed in the final image as shown in Fig. 4.6. Using the Matlab function *roipoly*, a closed polygon is defined within which the pixels can be replaced by either the base image or/and transformed image as shown in Fig 4.7. Presently stitching can only be performed using Matlab as the functions cannot be deployed in a standalone executable. A walkthrough of the Matlab program is provided in the Appendix B-User Guide for Stitching Images.

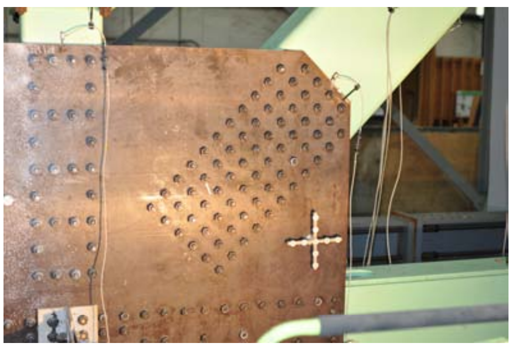


a)

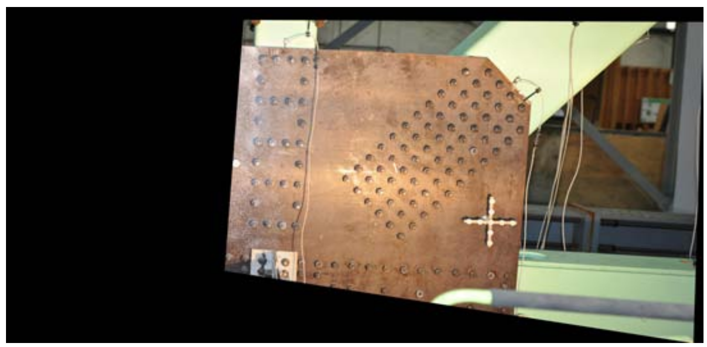


b)

Figure 4.1: Base image (Image 1) a) Original base image and b) Boundaries of composite image populated with Image 1



a)



b)

Figure 4.2: Transformed image (Image 2) a) Original base image and b) Boundaries of composite image populated with Image 2

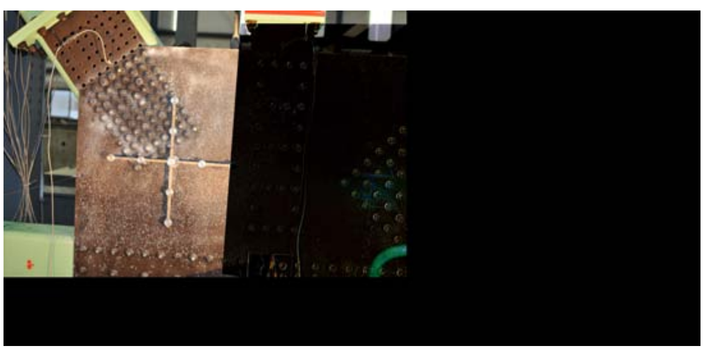


Figure 4.3: Subtracted image taken as Image 1 minus Image 2. This subtracted image is added to Image 2 next

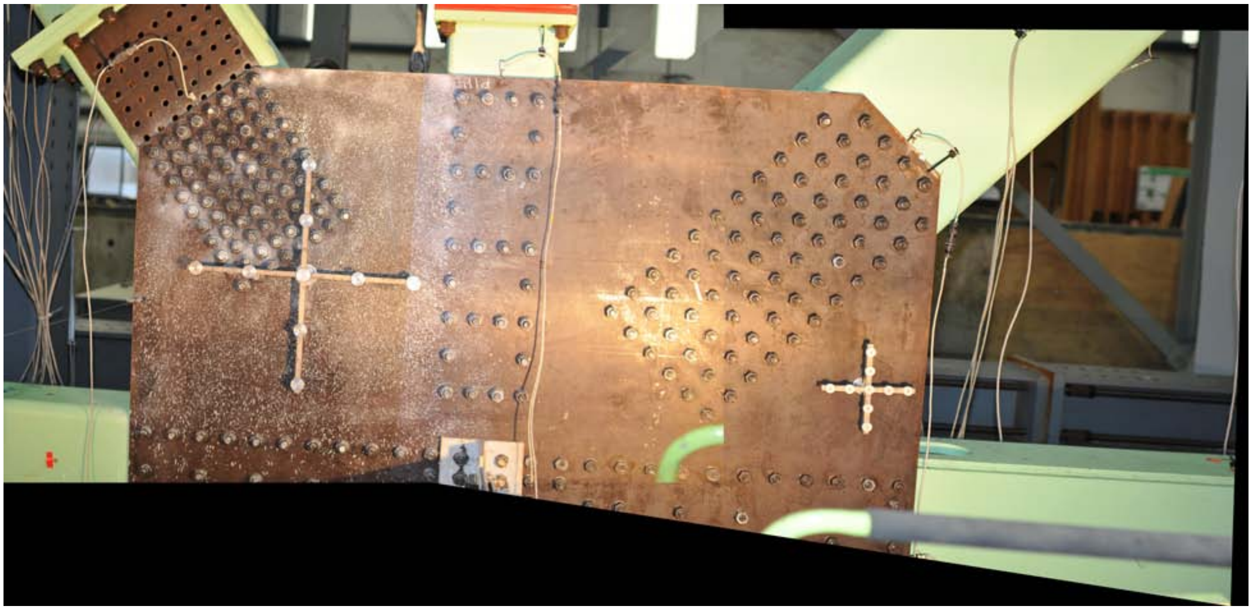


Figure 4.4: Subtracted image in Fig. 3.2 is added to Image 2 to produce stitched image

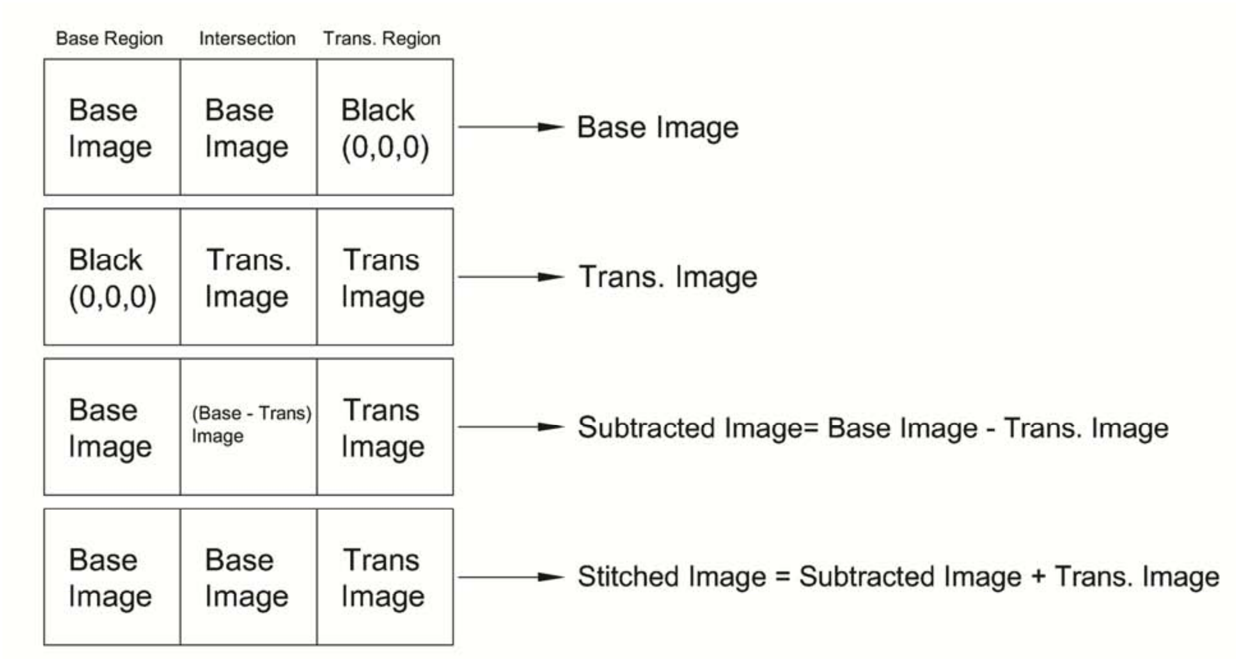


Figure 4.5: Illustration of adding and subtracting procedure for stitching two images together.

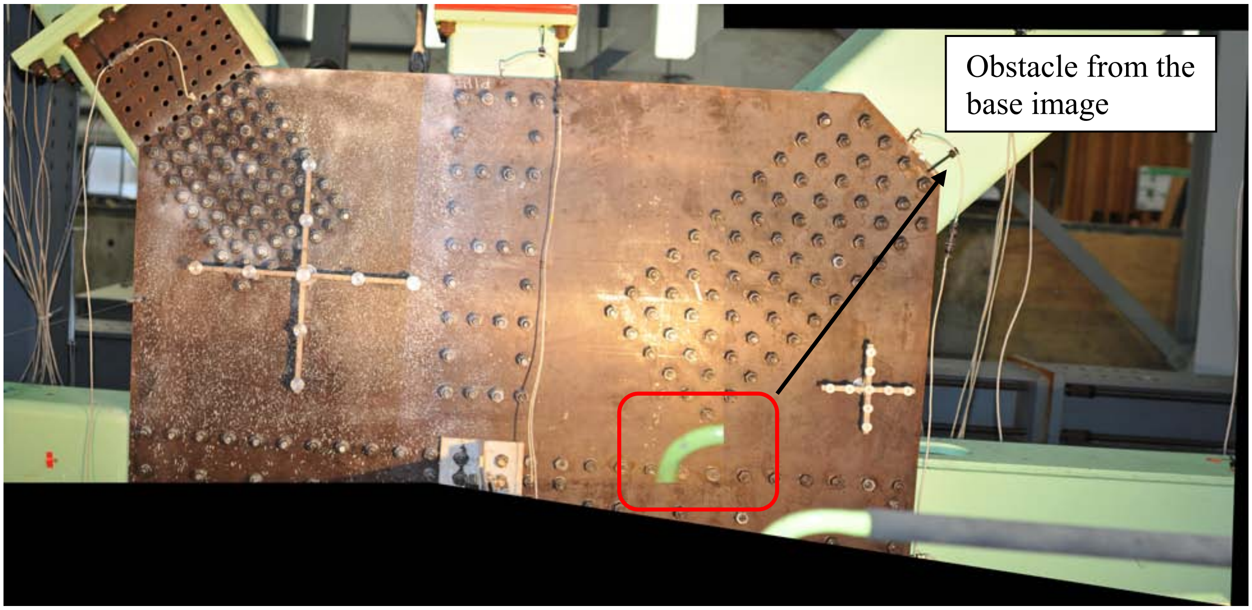


Figure 4.6: Stitched image with obstacle contained in the base image (Image 1). A polygon area including the obstruction is selected and the user can replace that area with part from Image 2

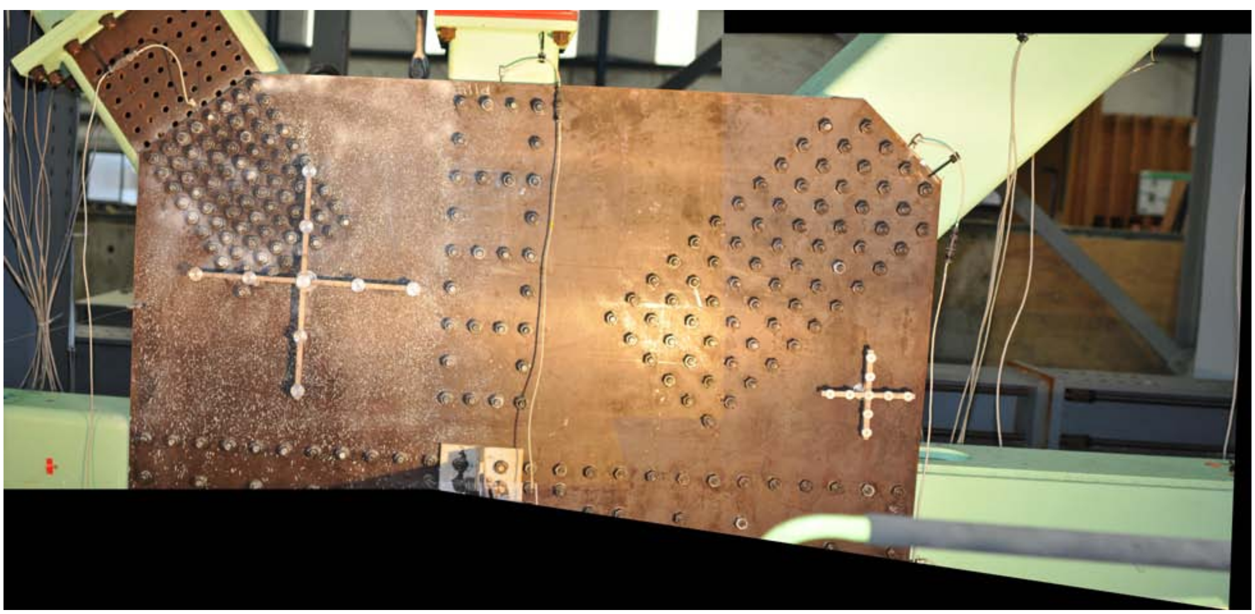


Figure 4.7: Final stitched image without obstacle

5.0 CONCLUSIONS

Two new methods were developed that will enhance geometric data collection of gusset plate connections for steel truss bridges. These include application of fisheye lenses and stitching multiple images together to produce a single composite image. Using these methods, the geometry of the connection plates and fasteners can be collected from the processed images. The developed methods were illustrated with step-by-step instructions for their implementation.

6.0 REFERENCES

Abraham, S., and W. Forstner. Fish-eye-stereo calibration and epipolar rectification. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 59, No. 5, 2005, pp. 278-288.

Dare, P. M., H.B. Hanley, C.S. Fraser, B. Riedel, and W. Niemeier. An operational application of automatic feature extraction: the measurement of cracks in concrete structures. *The Photogrammetric Record*, Vol. 17, No. 99, 2002, pp. 453-464.

Fujita, Y., Y. Mitane, and Y. Hamamoto. A method for crack detection on a concrete structure. The 18th International Conference on Pattern Recognition, 2006.

Higgins, C., Q.D. Nguyen, and O.T. Turan. Digital Imaging Tools for Gusset Plate Evaluation. The Fifth International Conference on Bridge Maintenance, Safety and Management. The International Association for Bridge Maintenance and Safety (IABMAS), 2010.

Hutchinson, T. C., and Z. Chen. (2006), Improved image analysis for evaluation concrete damage. *Journal of Computing in Civil Engineering*, Vol. 20, No. 3, 2006, pp. 210-216.

Ito, A., Y. Aoki, and S. Hashimoto. Accurate extraction and measurement of fine cracks from concrete block surface image. IEEE 2002 28th Annual Conference of the Industrial Electronics Society. Vol. 3, 2002, pp. 2202-2207.

Lee, S., and L. Chang. Digital image processing methods for assessing bridge painting rust defects and their limitations. ASCE International Conference on Computing in Civil Engineering, 2005.

Liu, Z., D.S. Forsyth, A. Marincak, and P. Vesley. Automated rivet detection in the EOL image for aircraft lap joints inspection. *NDT&E International*, Vol. 39, 2006, pp. 441-448.

Matlab 7.11.0 (R2010b). The MathWorks, Inc., Natick, MA, 2010.

McCrea, A., D. Chamberlain, and R. Navon. Automated inspection and restoration of steel bridges-a critical review of methods and enabling technologies. *Automation in Construction*, Vol. 11, No. 4, 2002, pp. 351-373.

Ray, S. F. *Applied photographic optics: lenses and optical systems for photography, film, video, electronic and digital imaging*, Third Edition. Focal Press, 2002.

Schneider, D., E. Schwalbe, and H.G. Maas. Validation of geometric models for fisheye lenses. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 64, No. 3, 2009, pp. 259-266

Sinha, S. K., and P.W. Fieguth. Automated detection of cracks in buried concrete pipe images. *Automation in Construction*, Vol. 15, 2006, pp. 58-72.

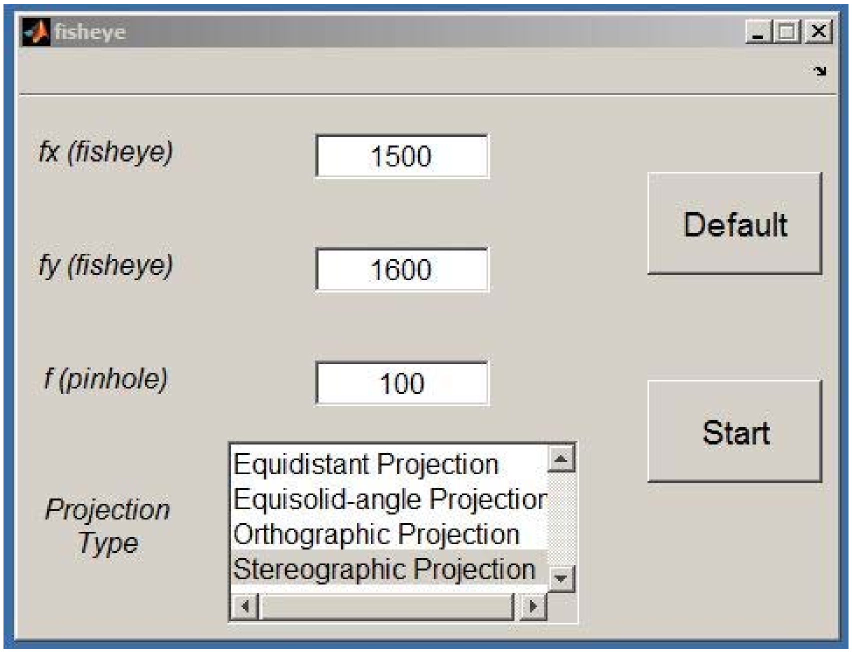
Yamaguchi T. and S. Hashimoto. Automated crack detection for concrete surface image using percolation model and edge information. Proceedings of the 32nd Annual Conference of the IEEE Industrial Electronics Society, 2006.

Yamaguchi, T., and S. Hashimoto. Fast crack detection method for large size concrete surface images using percolation based image processing. *Machine Vision and Applications,* Vol. 21, No. 5, 2009, pp. 797-809.

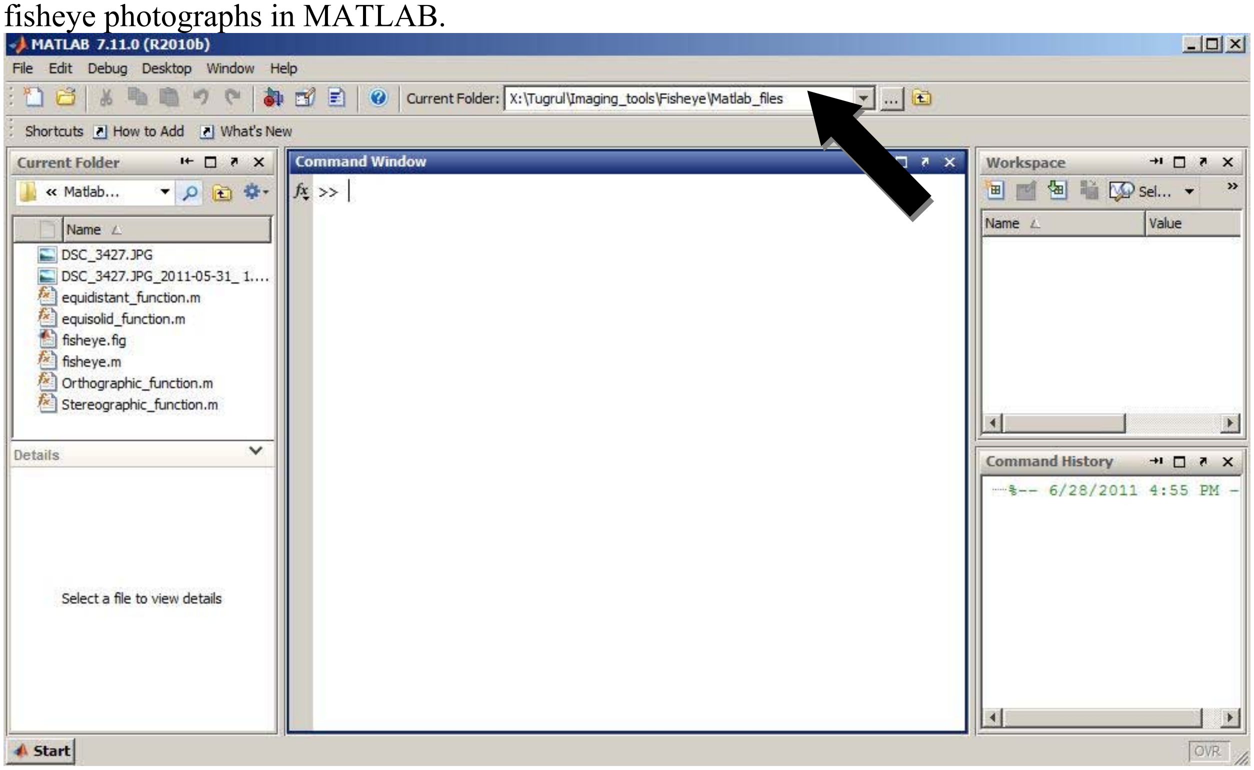
Yamaguchi, T., S. Nakamura, and S. Hashimoto. An efficient crack detection method using percolation-based image processing. Proceedings of the 3rd IEEE Conference on Industrial Electronics and Applications, 2008.

APPENDIX A:
USER'S GUIDE FOR FISHEYE IMAGES

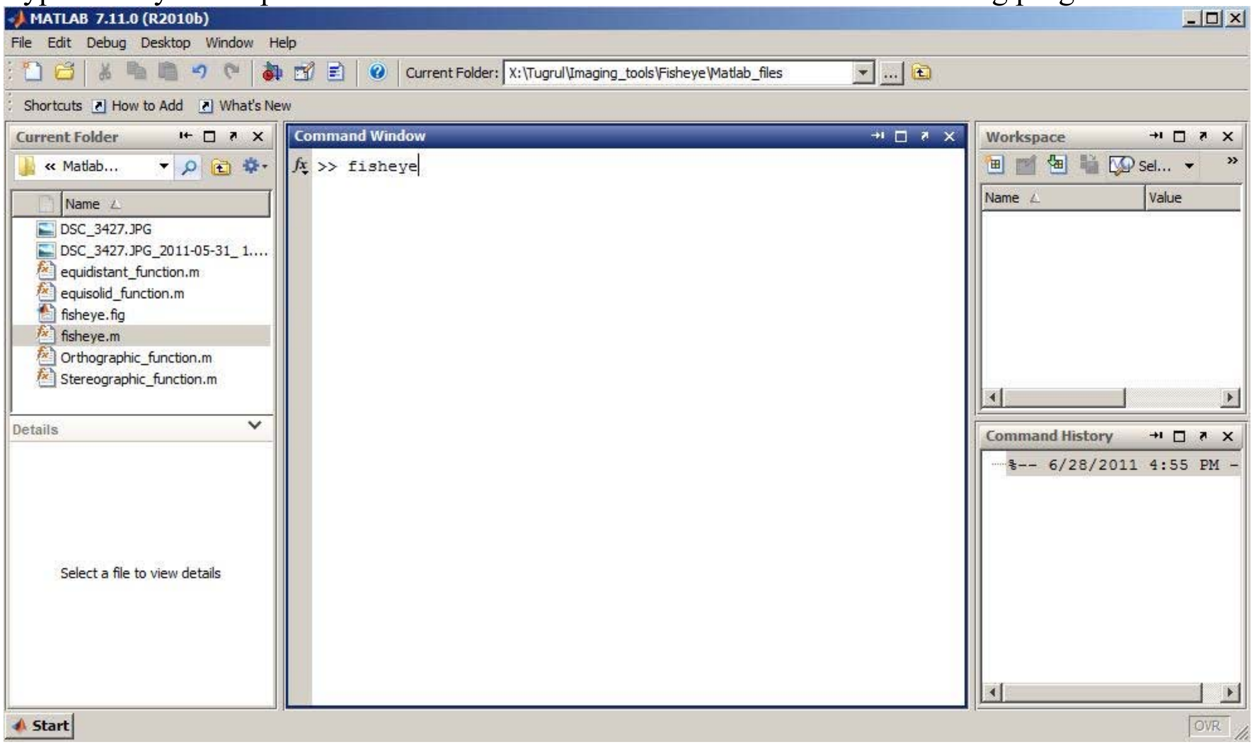
Run “fisheye.exe”



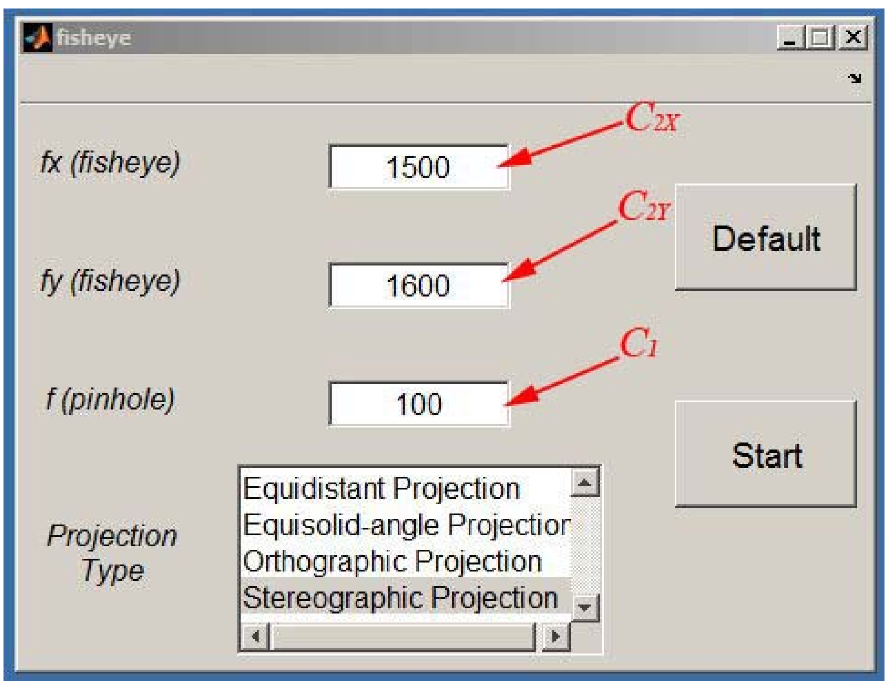
or open Matlab. Open the directory with fish eye.m, fisheye.fig, equidistant_function.m , equisolid_function.m, Orthographic_function.m and Stereographic_function.m; and digital fisheye photographs in MATLAB.



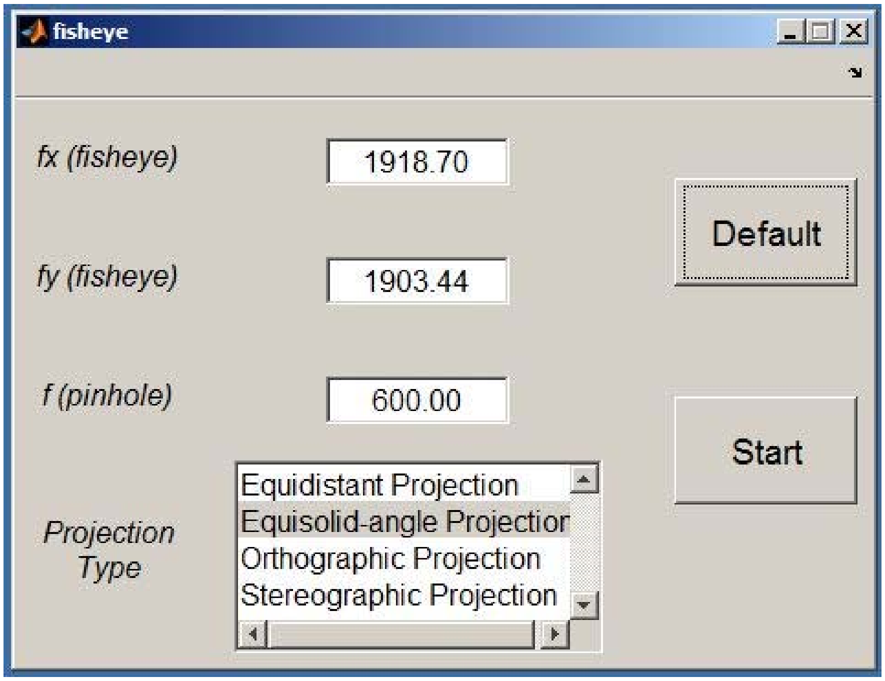
Type “fisheye” and press enter in the Command Window to start the defishing program.



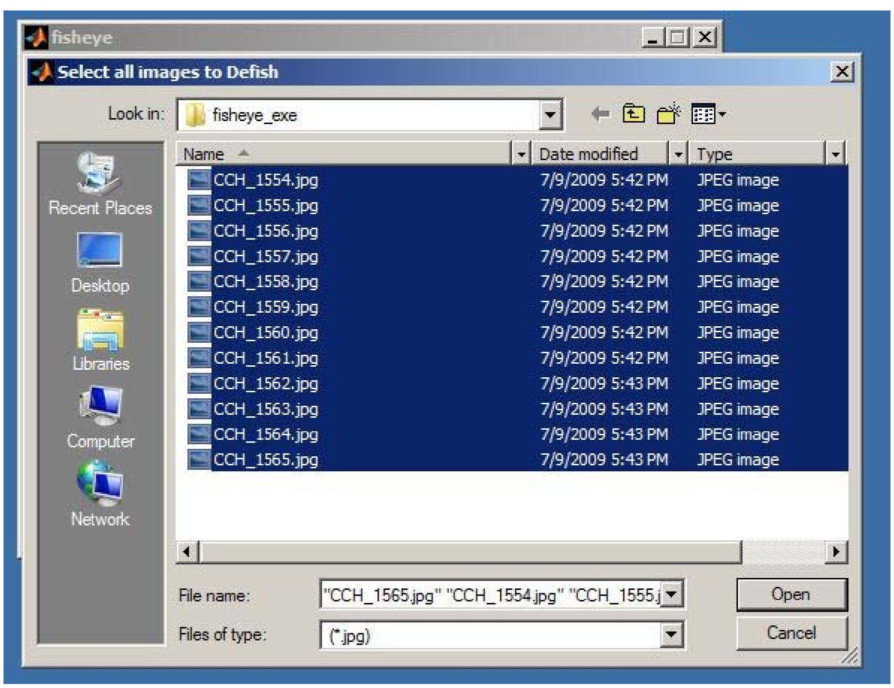
Depending on the lens and the camera, change fx(fisheye) (c_{2X}), fy(fisheye) (c_{2Y}), f(pinhole) (c_1) and/or Projection Type.



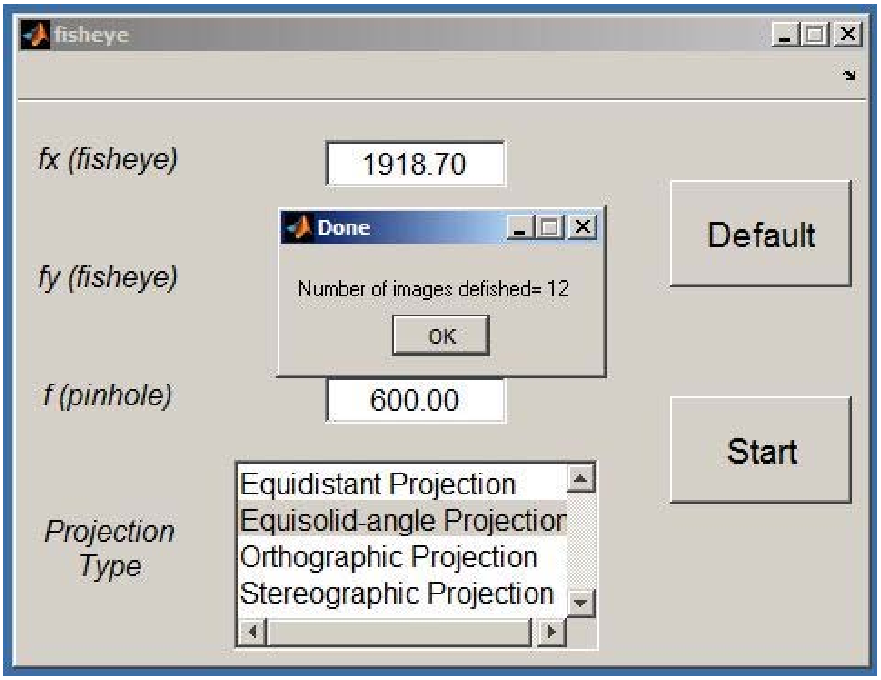
In order to return back to Nikkon D300 camera with a Nikkor 10.5 mm lens and c_1=600 pixel, click on “Default”.



Click on “Start” and select pictures that are going to be defished. Different types of pictures such as *.jpeg, *.tif and etc can be selected.

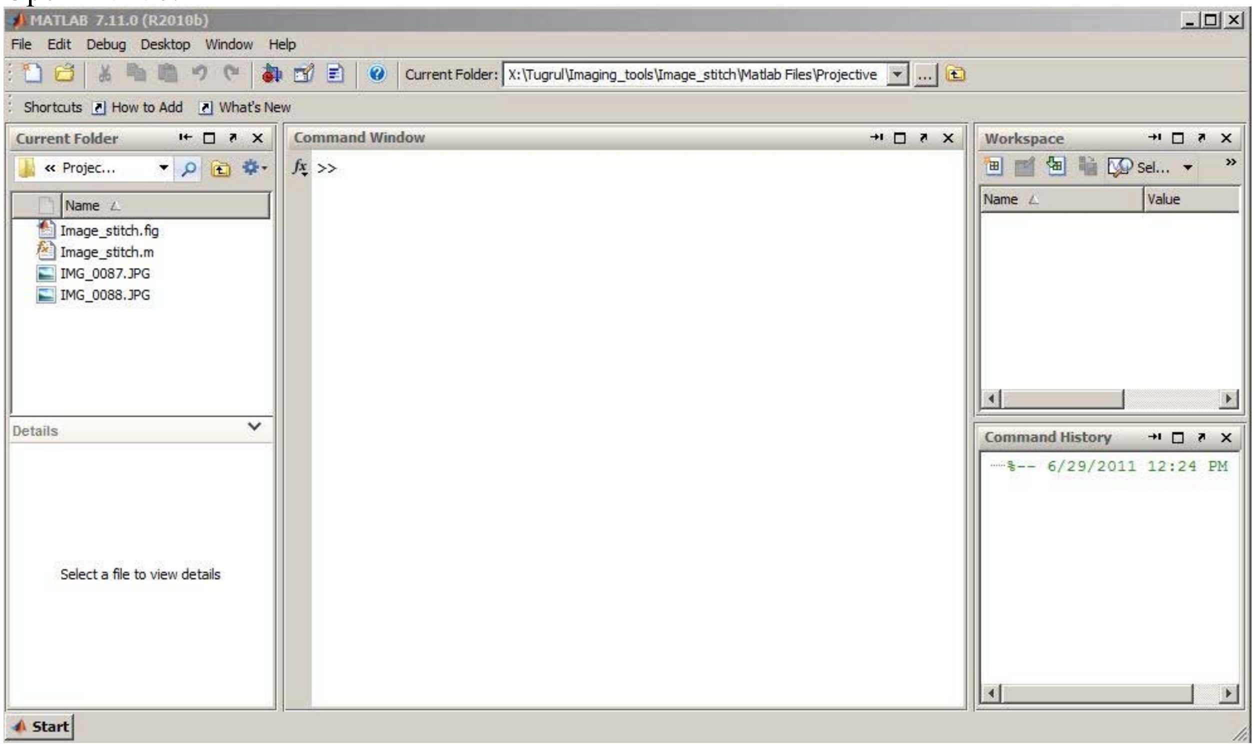


When all the images are defished, a message dialogue box pops up reporting the defished number of images.

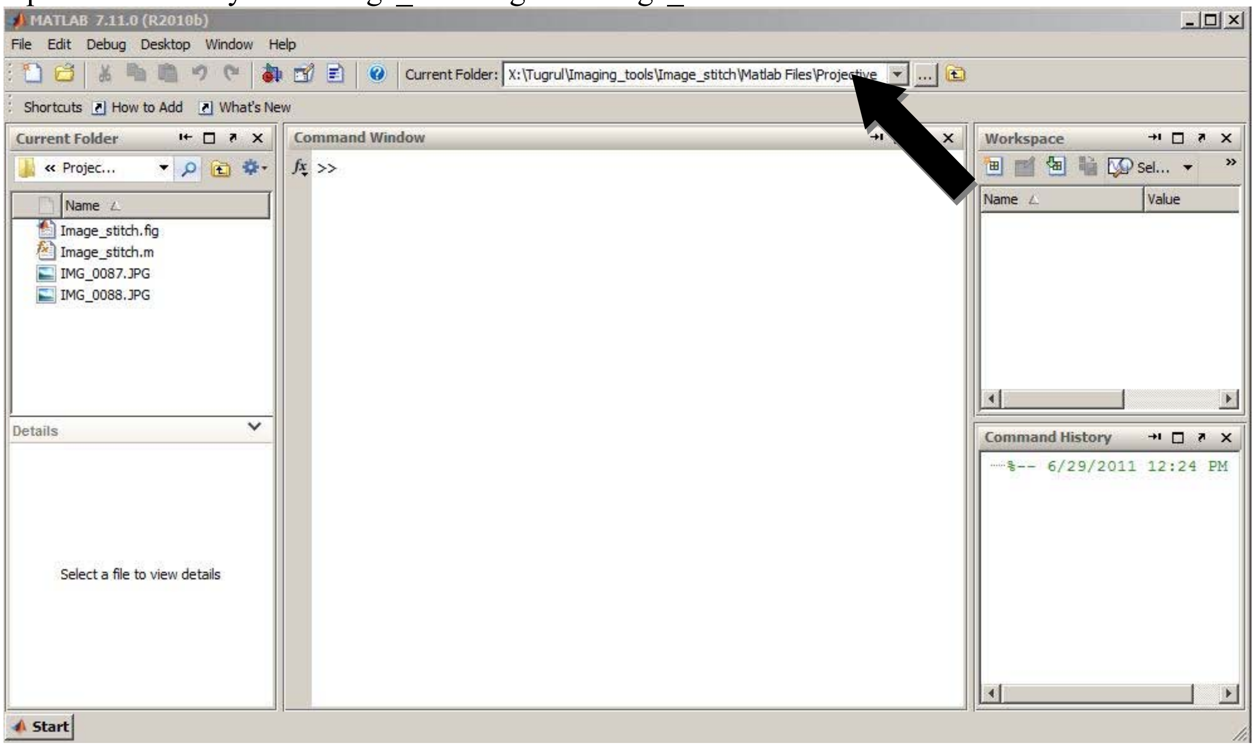


APPENDIX B
USER'S GUIDE FOR STITCHING IMAGES

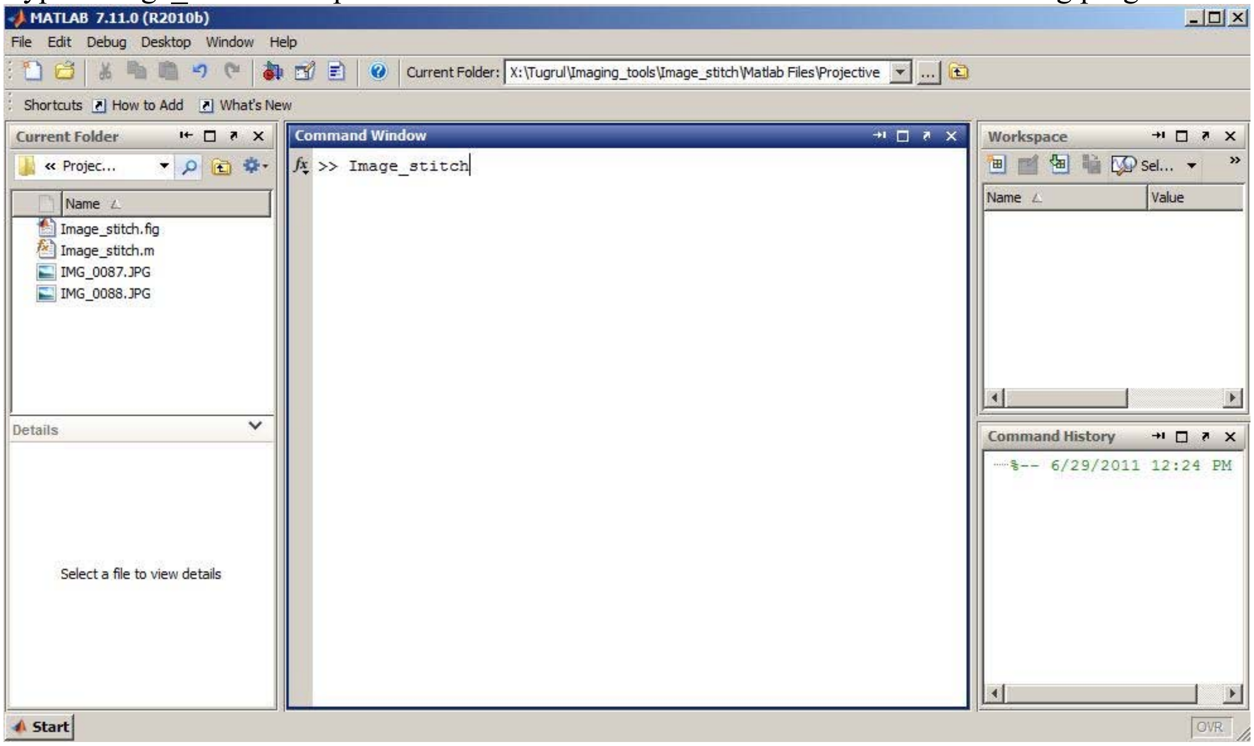
Open Matlab.



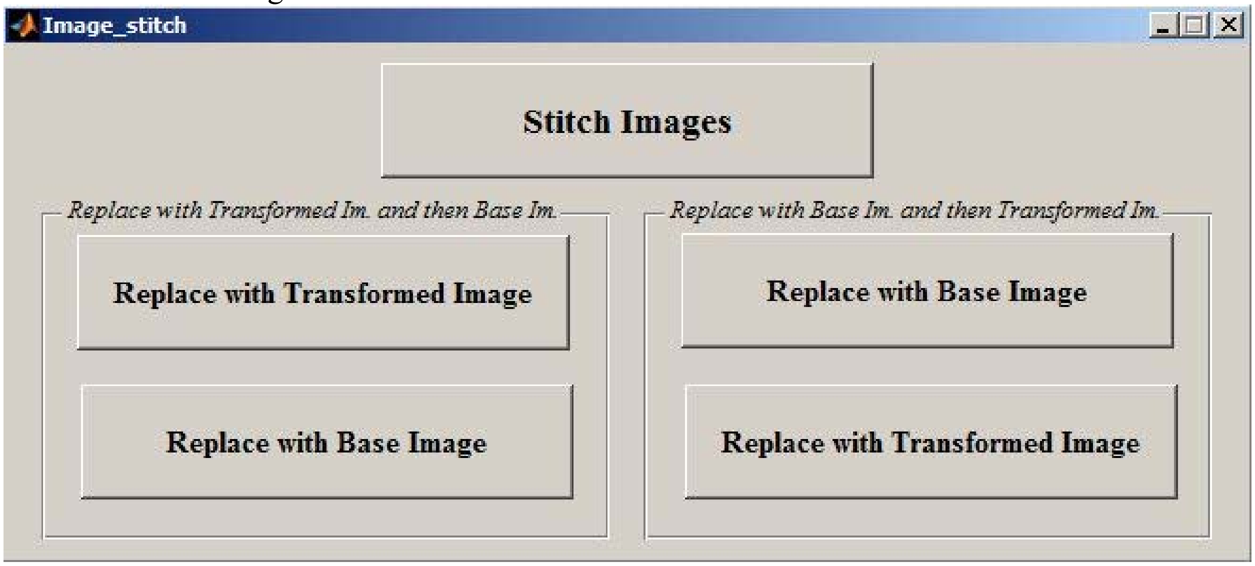
Open the directory with Image_stitch.fig and Image_stitch.m in MATLAB.



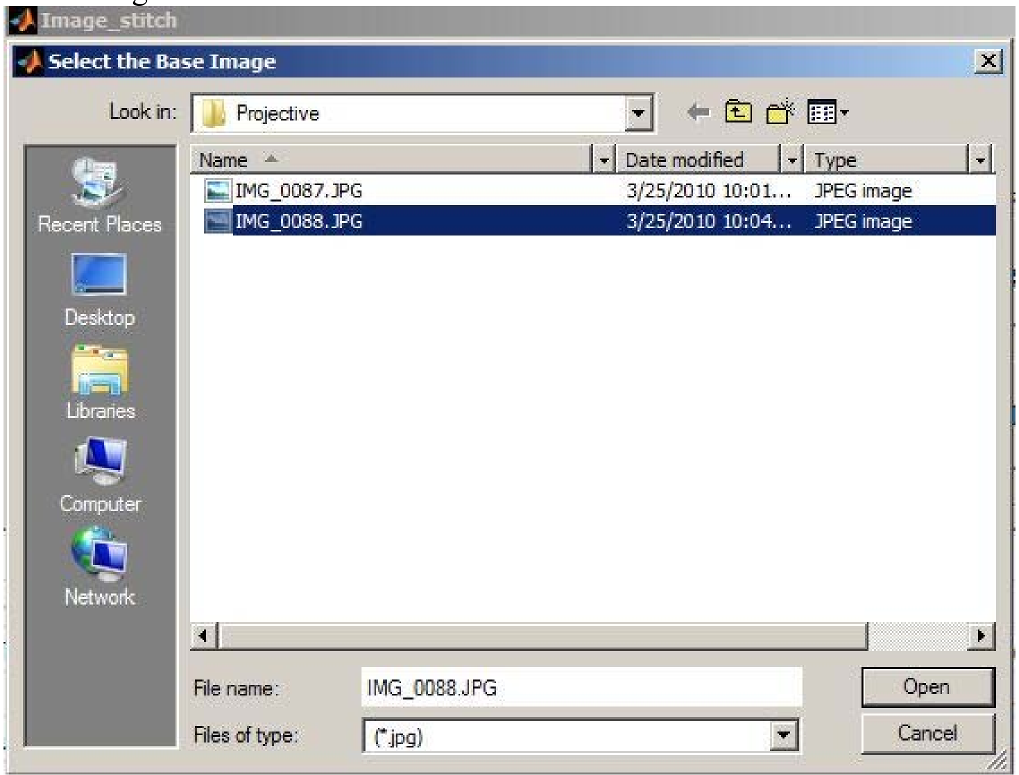
Type “Image_stitch” and press enter in the Command Window to start the stitching program.



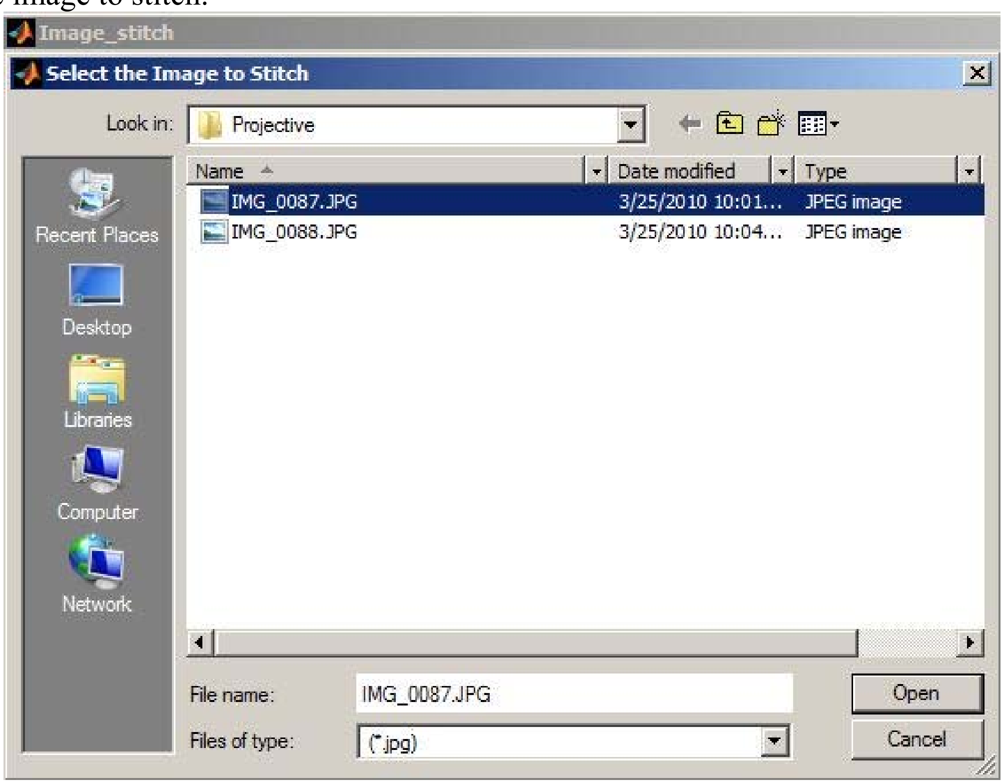
Click on “Stitch Images”.



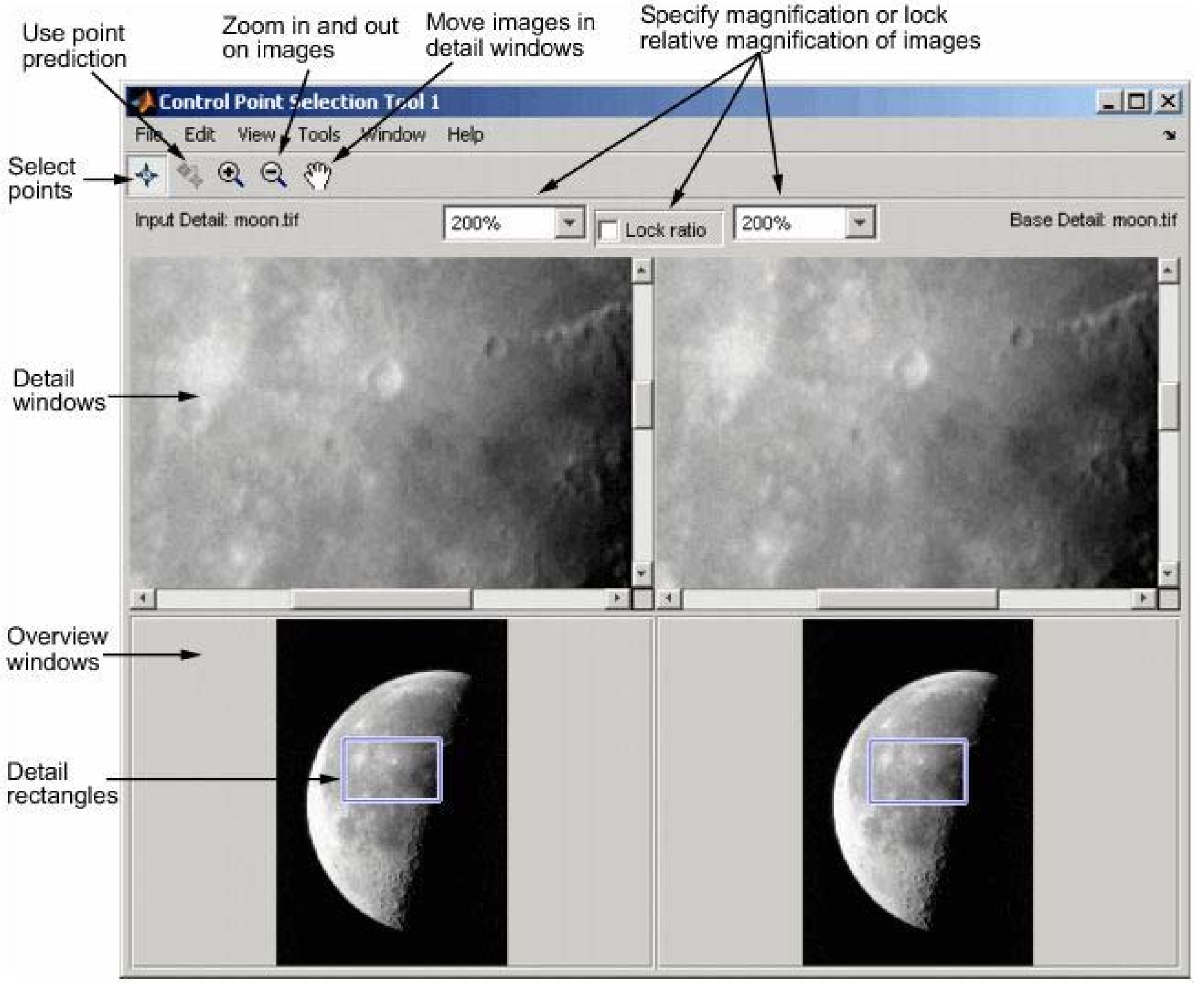
Select the base image.



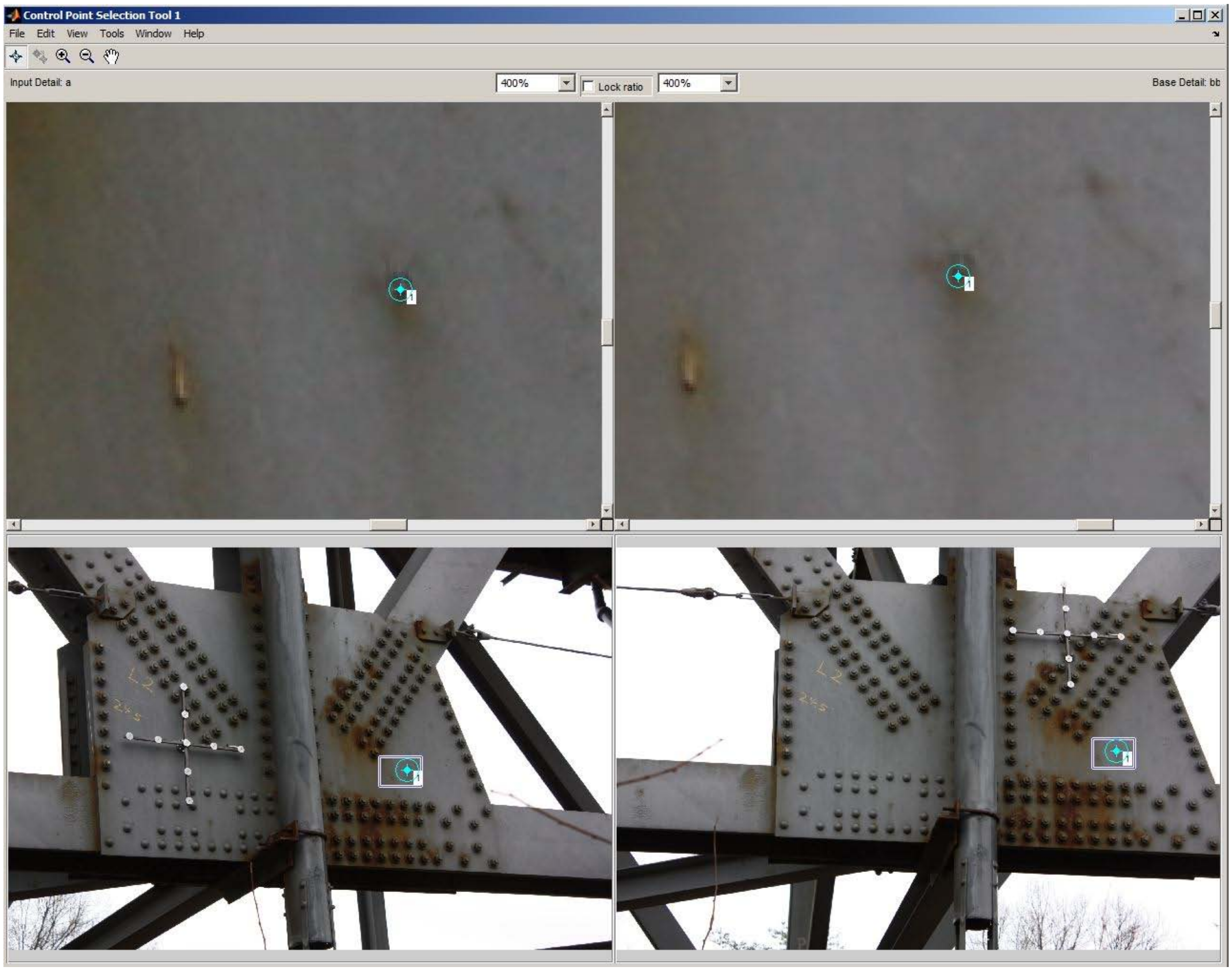
Select the image to stitch.



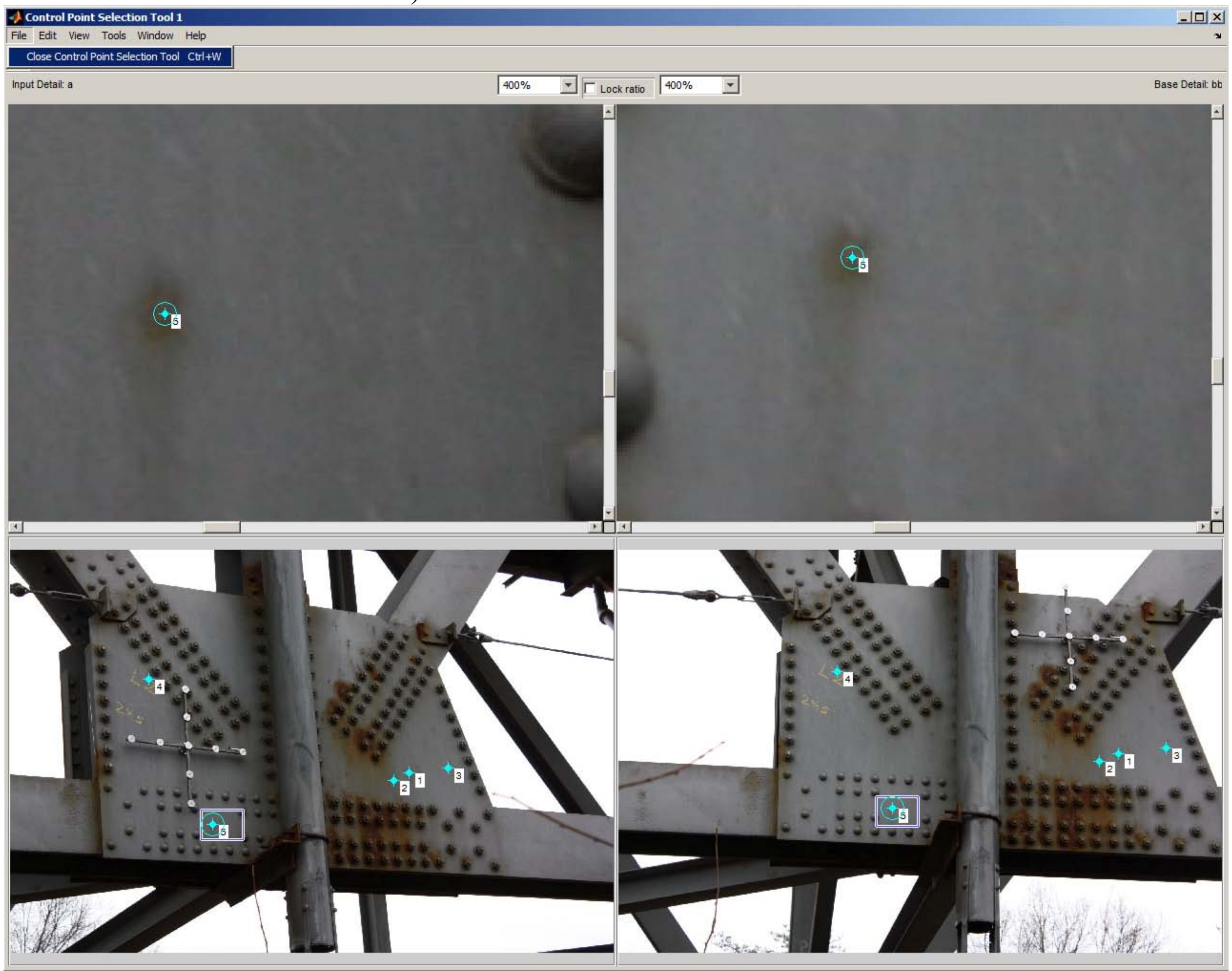
Control Point Selection Tool is activated. Default appearance of the tool is shown below (Matlab 2010b, Mathworks).



Using Control Points Selection Tool, select the similar points. At least two points are needed for Nonreflective Similarity Transformation; for Projective Transformation, at least four points are needed.

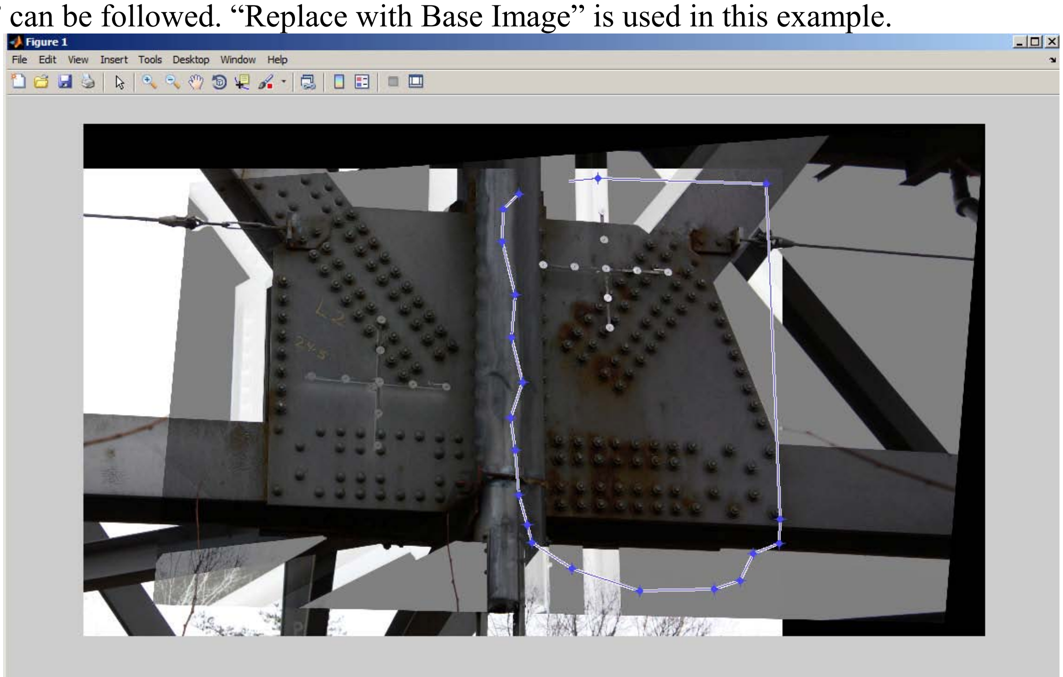


After selecting all the points, close the Control Points Selection Toolform (**File>Close Control Point Selection Tool** or **Ctrl+w**).



Stitched image is saved to the current directory.

To remove the obstacles, if needed; click on Replace with Base Image or Replace with Transformed Image and create a polygon using *roipoly*. Either the column "Replace with Base Image and then Transformed Image" or "Replace with Transformed Image and then Base Image" can be followed. "Replace with Base Image" is used in this example.



Close the polygon by selecting the first polygon point (zoom-in and zoom-out is not available in this function). After closing the polygon, double click on a region inside the polygon. Selected polygon is replaced by the Base Image. New image is saved to the current directory.

If needed, “Replace with Transformed Image” can be selected and the previous two steps can be repeated. New image is saved to the current directory.

Examples of stitched images:

Figure B1: Base Image (Image 1)

Figure B2: Image to be stitched (Image 2).

Figure B3: Stitched images

Figure B4: Final image.