Digital Image Correlation User Manual

By
Mr. Aditya Vipradas
IASc Summer Research Fellow
Indian Institute of Science

under the guidance of
Prof.(Retired) B.K. Raghu Prasad
Advisor, Centre for Campus Management and Development
Indian Institute of Science
Bangalore-560 012

August 2013

Instructions to the User

Codes in Matlab used for the Digital Image Correlation and Tracking are originally written by Rob Thompson, Daniel Gianola and Christopher Eberl from Johns Hopkins University. Refer (*http://www.mathworks.in/matlabcentral/fileexchange/12413-digital-image-correlation-and-tracking*) for the original codes.

This work focuses on the modification of these codes and writing new ones in Matlab for completion of the desired project. New names have been assigned to some of the modified codes. This user manual gives instructions to be followed to use the new and modified functions. All the Matlab codes and files can be accessed from *(https://github.com/adityavipradas/DIC)* address.

'An experimental investigation on the FPZ properties in concrete using digital image correlation technique' authored by ZhiMin Wu, et al. has been used as the reference research paper. Familiarity with the study of fracture in notched beams is a prerequisite for the implementation of these codes.

The following steps are to be followed to perform DIC for the concerned specimen.

1. Copy and replace all the files from DIC folder to the folder containing all the images
2. Open Matlab
3. Open the folder containing the images in Matlab
4. Use the following commands to perform the required tasks. Select validx.dat and validy.dat files when asked. You can type the commands in the Matlab console or use them from the command history. In order for the commands to appear in the command history, they should at least be typed and used once from the console.

   new_filelist_generator: Run this command before any other command. This command creates the filelist in the folder. Specify the number of images when asked by the prompt.

   new_grid_generator: Run this command after the *new_filelist_generator* command. This command is used for generating the grid. Select the required image when asked. Following options are available in the menu for grid creation.

   (a) *Rectangular:* This option creates rectangular grid. The user is supposed to click two points in one line along the notch in y-direction. A prompt then asks the user to specify the number of pixels to the right and left of the selected points. The rectangular grid is then created accordingly. Use this command for executing *new_displacement, both_sides_disp, fpz, codfpz and crackprop* functions

   (b) *Two Markers:* This option creates only two grid points. The user is supposed to select a point along the notch in y-direction. A prompt then asks the user to specify the number of pixels on the left and right of the selected point. The two markers are then created accordingly one on each side of the selected marker. Use this command for executing *cmod, cmod_3_axis, ydisp, ydisp_3_axis and split* functions.

(c) *Set extreme coordinates:* This option can be used to find out the number of pixels corresponding to 1mm of the specimen. For fulfilling this purpose, choose one grid point each at the top right corner and bottom left corner of the specimen. The x and y coordinates of these grid points are stored in xcoord.dat and ycoord.dat files in the folder which can be used to find the relation.

new_automate_image: After the grid creation, use this command to process all the images and store the x and y coordinates of the grid points in pixels in validx.dat and validy.dat files in the folder.

cmod: This command is used to calculate the Crack Mouth Opening Displacement (CMOD) at the notch of the beam. The *Two Markers* grid option is to be used with each marker on either sides of the notch. The output is obtained as a plot of load in newton (obtained from the specified excel datasheet in column 7) against the DIC CMOD and experimental CMOD (column 4 in datasheet) in mm. Dialog boxes to obtain the scales of the axes of the plots are prompted during the code execution. Specify the excel datasheet name when asked.

cmod_3_axis: This command plots graphs with 2 y-axes, one of them being the load and the other one being the CMOD in mm. The x-axis contains the image numbers. Colors of the plots can be changed at will. This flexibility is provided in the form of dialog boxes. In order to obtain the plots, the experimental data is cubic spline interpolated. Plot showing relationship between the experimental data and the interpolated data is also displayed.

ydisp: This command is used to calculate the Load Point Displacement (LPD) and the fracture energy. The left grid point obtained from the *Two Markers* option is considered for evaluation. The output is obtained as a plot of load in newton (obtained from the excel datasheet in column 7) against the DIC LPD and experimental LPD (column 1) in mm. Dialog boxes to obtain the scales of the axes of the plots are prompted during the code execution. Specify the excel datasheet name and area of the beam above the notch when asked. The value of the fracture energy can be obtained from the console. The fracture energy is calculated using cubic spline interpolation of the DIC data. The plot showing the DIC and interpolated data is also drawn to get an idea about the relationship between them.

ydisp_3_axis: This command plots graphs with 2 y-axes, one of them being the load and the other one being the LPD in mm. The x-axis contains the image numbers. Colors of the plots can be changed at will. This flexibility is provided in the form of dialog boxes. To obtain the plots, the experimental data is cubic spline interpolated. Plot showing relationship between the experimental data and the interpolated data is also displayed.

new_displacement: This command is used to obtain the plots of deformations of the points in x-direction along y-direction of the beam in mm against the depth of the beam. These plots also display the linear regression of the displacements of all the points in the graph. Select the *Rectangular* grid above the notch and specify the horizontal grid size such that only two grid columns are obtained with one on each side of the notch such that the grid encompasses the crack. All the generated images are stored in *disp_against_depth* folder in the folder containing the images.

both_sides_disp: This command is used to obtain the plots of displacements of the grid points on each side of the notch. Select the *Rectangular* grid above the notch and specify the horizontal grid size such that only two grid columns are obtained with one on each side of the notch such that the grid encompasses the crack. The linear regression of the displacements of the grid points on each side of the notch is also performed and plotted accordingly. All the generated images are stored in *both_sides_disp* folder in the folder containing the images.

split: This command is used to obtain the plots of load vs x-displacement for split tensile specimen. The program gives flexibility for choosing the degree of the curve to be fitted to the DIC data. Plots of load in newton against the DIC and regressive x-displacements in mm along with the displacement in x-direction corresponding to the maximum load are obtained as output in the console. The *Two Marker* grid option should be used in order to implement this code.

fpz: This command is used to obtain plots of displacements of the raster points along x-direction against the location of the points above the notch. It only generates plots for the horizontal line just above the notch. It takes the midpoint of the notch width as its origin for identifying the location of the grid points along x-direction. Make sure to generate a large number of raster points along the horizontal line in order to obtain good and meaningful plots. The images generated by this command are stored in the folder named *fpz*. This command takes a lot of time to execute because of the processing of incredibly large number of raster points. These raster points can be generated using the *Rectangular* option made available after executing the *new_grid_generator* command. Each generated image is stored in jpg and fig formats. Flexibility is provided to change the least count of the x and y axes in the plots. This code generates results in accordance with the Fig.4 in the reference research paper.

codfpz: This command is used to generate the crack propagation path and Crack Opening Displacement (COD) plots separately. It is not of much use in this project.

crackprop: This command is used to generate the plots showing the development of the fracture process zone. The main plot also contains an inset box which shows the plot of the beam depth against the COD. This command provides the flexibility to alter the notch width and depth parameters. The COD plot is drawn along with the crack propagation plot so that both of them can be compared. The x-axis of the main plot has positive and negative limits equal in their magnitudes. The user can specify these limits along with the upper limit of the y-axis. The lower limit of the y-axis is always zero. The inset box can be positioned anywhere along the x-axis. The user is supposed to give the x-coordinate of the lower left corner of the inset box in order to decide its location in the main plot. The plots obtained as the output are stored in the *plotcod* folder in jpg and fig formats. The user should choose as many raster points as he can along the vertical and horizontal directions. This is because, the more the raster points, the better the results but greater the execution time. 70000 raster points take about 30 minutes to process. These raster points can be generated using the *Rectangular* option made available after executing the *new_grid_generator* command. The crack propagation path is plotted by considering the mean value for each of the grid points to the left and right sides of the midpoint of the notch width. The obtained plots are similar to Fig.5 in the reference research paper.