Automated Crack Mapping

NEES @ UNR

Principal Investigator:

Sherif Alfass- *University of Nevada, Reno*

Project Mentor:

Rodney Porter– *University of Nevada, Reno*

Project Coordinator:

Kelly A Lyttle– *University of Nevada, Reno*

Gautam Shankar

Network for Earthquake Engineering Simulation

NEES Site: University of Nevada, Reno

Sponsors:

Introduction Statement

There is a growing need to study the details of crack damage on structures used in earthquake engineering simulation. Where crack damage originates, how it forms, and at what point the cracks cause structural damage and how they affect different types of columns has all been under research. Now the tools to analyze the crack damage are being created.

Importance of Automated Crack Mapping

The necessity to store earthquake simulated structures in the real world can be eliminated. Crack mapping processes tiles of images of the specimen and combines them so that they can be viewed as one final image. In addition, this image can then be mapped onto a visual model of the specimen. Further steps can be taken to study crack formations with computer tools for accuracy and ease of collecting data. Measuring crack size, identifying newly formed cracks from ongoing tests and performing quantitative analysis regarding crack data can all be accomplished with the help from computers.

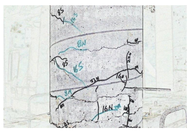
Methods of Image Processing

- ImageJ- An open source Java image processing program.

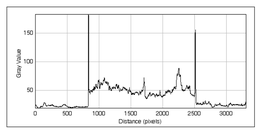
- Macros – scripts to automate tasks and create custom tools.
- Plugins – Extension of the existing set of program tools, may be created using ImageJ's built in text editor or a Java compiler.

The following are processing steps for mapping a cylindrical column:

- Find edges of column
- Use MosaicJ plugin to systematically combine one vertical set of pictures taken from top to bottom.
- Crop column in half vertically to set up for bi-linear mapping plugin.
- Apply Bi-linear mapping to fix column curvature.
- Use MosaicJ again to combine split sections into final processed image.



Edge Detection



Histogram of light intensity

ImageJ Tools

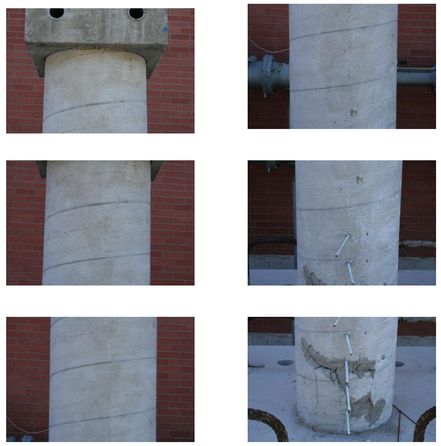
- Tools come in the form of plugins, macros, and built in features.
  List of Plugins used:
- Turboreg, MosaicJ, Geometric Mappings
  Built in Features:
- Crop, Find Edges, Scale, Polygon Selections and etc.

Overall Scope of the Project

The current scope of the project is working with earthquake simulation tested structures such as the columns that hold up bridges. Using before and after maps of this object may be required. A series of maps across different testing periods may also be needed to enhance the study in the area. Optimizing a program to automatically map cracks efficiently and accurately is the goal. In the future, the scope can grow to include bridges and other similar structures.

Current Results

Stage 1: Mosaic of one vertical picture set



One Vertical Set of pictures

Stage 2: Bi-linear Map

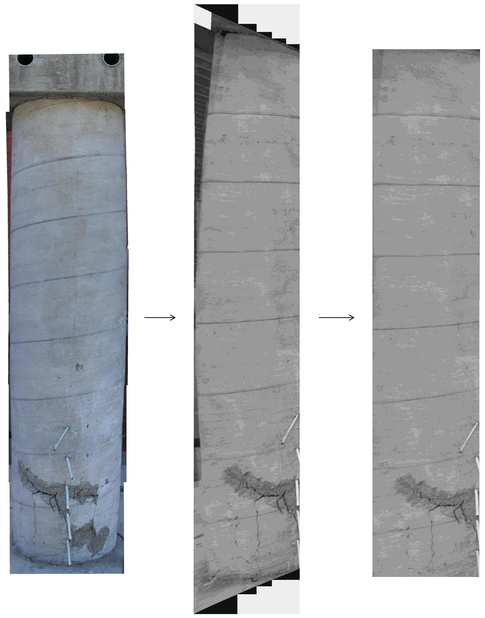
- Mosaic of picture from step 1 is cut vertically in half.
- Then the left half is cut again vertically into five sections.
- For each section, the bi-linear map is applied.

The right end of the left side of the cropped column is the center. As you move gradually towards the left end of the column, the bi-linear map needs to be applied more heavily. The width and height of each of these vertical sections are also changed using a formula that is programmed in Matlab. The vertical sections must fill up the space of quarter circumference of the column.

Stage 3: Perspective Fixes

- Fix perspective change caused by tilting the camera up and down while taking pictures of the top and bottom of the column.

  Another bi-linear mapping is done to correct this distortion. The areas towards the edge of the column appear to have lost pixels due to being stretched and flattened, so we can disregard that section. But, we need overlapping images from the rest of the vertical set of pictures around the column to make up for it.



Curved → Almost Flat → Flat

Stage 4: Repeat previous steps to other areas

- Apply the MosaicJ plugin again to combine all processed vertical sets of pictures.

  After this step, there will be one final processed picture that includes all the surface area of the entire column! So the final image is now mapped onto a 2-D plane, making it easier to analyze and study crack damage.

Mapping of Final Image

- MATLAB can be used to warp an image to a virtual cylinder.
- GUI tools can be developed to measure and analyze cracks.
- Series of maps of column across different testing periods of earthquake simulation can be gathered.
- Measuring cracks, data manipulation, and graphing tasks can all be performed once mapping is completed.

Picture Requirements

- Initial picture is taken of the middle of the column from known distance.
- Column should take up approximately ¾'s of the picture's width (see set of vertical pictures for examples).
- Overlapping pictures are taken consistently to get the top and bottom of the column.
- Steps above are repeated to all 8 or 16 sides of the column.
- The distance between each image from top - bottom should be the same.

Further Developments

- Automate processing stages in the background so there is no need to open and close images.
- MosaicJ can be automated along with the geometric mappings plugin.
- Picture requirements can be minimized and further developments need to be done to achieve that goal.

Conclusion

The flexibility of the final completed program that automates the mapping of cracks is the challenge. With time, user error will become less important as computers provide assistance in correcting images. If enough data is not available, then not much can be done, but if it is available and hard to retrieve, then there is a way to process and get that data out. Currently, a semi-automated solution can be implemented to achieve crack mapping but the goal is the same, full automation. Automation allows for more data to be gathered in shorter time and harnesses the power of computers.

Acknowledgements

I would like to thank my mentor Rodney Porter for guiding me along the project. He kept me on track and made sure time was to be managed wisely. Thanks to my Project investigator Sherif Alfass for clarifying the scope of the project. Thanks to Kelly A Lyttle for introducing me to the NEES program that made it all possible. This project is supported by the George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES) Program of the National Science Foundation under Award Number CMMI-0927178.

Reference:

ImageJ. (2010). **ImageJ Features**. Retrieved from http://rsbweb.nih.gov/ij/developer/macro/functions.html