

0.1 Main program

This is the part of the program that sets up the correlation problem

1. Name the correlation results save file
2. A new folder is created based on the name of the correlation results save file. This is so that the results are easy to find and are well separated.
3. Select images to correlate
4. Create mask so that the ROI to correlate within the reference image is identified
5. This mask is fed into a function which determine how to place the subsets of specified size and stepsize so that the maximum amount of them fit into the ROI
6. Use function to create the seed point - however I am not using this method now. I am using particle swarm to search for a first guess to the warp parameters although this does not seem to work well all the time. Will be looking into using FFT and convolution to determine appropriate initial guesses.
7. Determine the correlation order around the seed point so that the once the first subset is correlated its warp parameters can be used as an initial guess for the surrounding subsets and so on.
8. Create the necessary supporting functions based on the correlation algorithm used. For Newton-Raphson this includes
 - A function that deforms the subset according to the warp function by displacing the reference pixels, that are at integer pixel locations, to there deformed locations so that interpolation can be performed at these query points. (this function has extension pos)
 - A function to correct for distortion in the results - I have not been able to try this function yet since I don't know how to get the extrinsic and intrinsic parameters of the camera. (has extension dist)
 - A function to determine the Jacobian (extension jac) and a function to determine the Hessian (extension hes). The equations in these functions are based on symbolic derivatives and so they are not merely approximations to the Jacobian and Hessian but the true derivatives.

For Lucas-Kanade the supporting functions are

- A function to determine the Jacobian of the warp function - this is used by a for loop to determine the Hessian (extension dPFunc)
- A function that deforms the subset according to the warp function by displacing the reference pixels, that are at integer pixel locations, to there deformed locations so that interpolation can be performed at these query points. (this function has extension WarpFunc)
- A function to determine the new approximation to the warp function matrix from the old approximation and the update to the warp parameters - this is necessary for the inverse compositional version of the algorithm. (extension WarpMat)
- A function to determine the new approximation to the warp function parameters from the the new approximation to the warp function matrix - i.e. it extracts the parameters from the matrix so they can be used in the algorithm (extension Mat2Vec)

- A function to correct for distortion in the results - I have not been able to try this function yet since I don't know how to get the extrinsic and intrinsic parameters of the camera. (has extension dist)

These supporting function are created in this way so they automatically adjust to the warp function that the user desired to use.

9. Then a while for loop is initialised to step through the images to be correlated. If the Newton-Raphson method is being used the coefficients of interpolation will be determined for the deformed image prior to correlation. These coefficients are save in a .mat file which allows specific sets of coefficients to be called individually.
10. Once all the subsets have been correlated their correlation coefficients are analysed to determine whether there are outliers. If so the outliers are improved upon using a few methods. The outliers are determined using Matlabs isoutlier function using the median option since this was found to produce the best results.
 - First the method determines the nearest neighbouring subset, to the subset under consideration, that has a good correlation coefficient value and uses the warp function parameters of this subset as an initial guess for the subset under consideration.
 - If this does not work then a particle swarm method is attempted. Using Matlabs particle swarm function a range of warp function parameters are tried in order to determine the one that gives acceptable results. The idea behind this is that if a displacement discontinuity or artefact appears in the image this method should handle it well enough.
 - If the bad correlation coefficient still persists then the method of subset splitting is to be tried (have not fully implemented this yet). This algorithm uses a warp function that has 12 parameters as a result of splitting the subset into 2 subsets that each have their own 6 warp function parameters. First Matlabs particle swarm function is used to determine the location and orientation of the crack within the subset by using the warp parameters of neighbouring subsets as the initial guess (at this point the algorithm does not try to improve the warp parameters but only checks the correlation coefficient resulting from placing the crack at different orientations and locations). Thereafter once the orientation is found a subset splitting correlation function is used to improve the 12 warp function parameters to obtain the best correlation coefficient possible.

0.2 Newton-Raphson

- Take in the reference and deformed images. Reduce the reference image to only the subset of the reference image that is under consideration.
- Determine the centre of the subset
- Create the matfile object from which the interpolation coefficients can be called.
- Link the functions that the algorithm uses to the code created specifically for this correlation run. In each correlation run the necessary supporting functions are created prior to beginning the correlation and these functions are named according to the name of the file in which the correlation information is to be saved. This is done to allow different warp functions to be used for different runs of the correlation program.

- Hereafter the iterative improvements are made to the warp parameters according to the Newton-Raphson method combined with the golden section search method. This method works by taking in an initial approximation to the warp parameters and then determining the Jacobian and Hessian for these warp parameters. Traditionally the inverse of the Hessian times the Jacobian is used to determine the step change to improve the warp function parameters however this method was found to be too sensitive to be practical (it is reasoned that since the correlation problem has 6 independent dimensions the Newton-Raphson method becomes unstable). As such it was improved by using the Newton-Raphson method to determine the search direction (the traditional step serves as the search direction) and then the golden section search method is used to determine the step size in this search direction that gives the best improvement to the correlation coefficient. Although this method is more robust it is much more time consuming (I want to change the algorithm to first use the Newton-Raphson step and only if the correlation coefficient doesn't improve use the golden section search method).
 - With in the Newton-raphson part of the algorithm the approximation to the warp function parameters are used to determine the query points for interpolation (by deforming the reference subset).
 - Thereafter the deformed image is interpolated at the query points using the coefficients of interpolation that were calculated earlier. These coefficients are then passed to the functions for the Jacobian and Hessian that were created earlier so that these matrices can be determined.
 - The Jacobian and Hessian are then used to determine the newton step/search direction.
 - Having the search direction the bracketing function determines the resulting correlation coefficient for a range of step sizes in the search direction. These correlation coefficients are then used to determine in which range of the step size is the minimum of the correlation coefficient. In other words it determines a lower and upper limit of the step size with the minimum of the correlation coefficient falling within these limits.
 - The golden section search method is then used to search between this limits for the minimum of the correlation coefficient. The step size that corresponds to this minimum is then passed back to the algorithm as the update to the warp function parameters.
- For the first 2 iterations the function uses an artificial search direction. First it searches in the x direction and then it searches in the y direction. The idea behind this is that the x and y displacements are usually the ones that change most between images and so by allowing the algorithm to do this a better starting point can be achieved for the Newton-Raphson algorithm to start at. Doing this allows the first true search direction to be more robust.
- After this first iteration, the Newton-Raphson method is used to determine the search direction and the golden section method searches along this direction for the minimum. If the warp parameter result in a correlation coefficient that is worse than the previous iteration the algorithm tries search in the 6 directions (corresponding to the 6 dimensions of the warp function) independently and uses the best result of this as the new approximation to the warp parameters. This is done to improve the robustness of the code.
- This is done until the norm of the change to the warp function parameters is below some threshold value or until a max amount of iterations have been undertaken.

0.3 Lucas-Kanade

- Take in the reference and deformed images. Reduce the reference image to only the subset of the reference image that is under consideration.
- Determine the centre of the subset
- Determine the gradient of the light intensity values of the reference subset
- Link the functions that the algorithm uses to the code created specifically for this correlation run. In each correlation run the necessary supporting functions are created prior to beginning the correlation and these functions are named according to the name of the file in which the correlation information is to be saved. This is done to allow different warp functions to be used for different runs of the correlation program.
- Thereafter the Hessian matrix is determined.
- Then an interpolation function is created for the deformed image so that the algorithm can quickly determine the light intensity value anywhere in the deformed image.
- The deformed subset is deformed according to the warp function and the current estimate of the warp function parameters. This gives the interpolation points.
- The new light intensity values for the deformed subset are determined by using the interpolation function and the interpolation points.
- Then the Jacobian is determined according to the chosen correlation coefficient equation. Note this is not the true Jacobian but the Jacobian according to the Lucas-Kanade method.
- The Hessian and Jacobian are then used to determine the update to the warp function parameters.
- The previously mentioned support functions are used to update the warp function parameters.
- The new correlation coefficient is then determined and if this value is low enough or if the norm of the update to the warp function parameters is small enough then the iterations will stop.