

Image Processing using TBB

Part 1

Using the stencil pattern, implement a TBB (Threading Building Block) kernel to blur an image using a Gaussian kernel. Note: You can use the example load function given to load images (greyscale):

- The code should be written both in a sequential and parallel approach. Code should be made parallel using the TBB `parallel_for` structure.
- The speed differences between the sequential and parallel approaches should be investigated and documented, this should also include comparisons to investigate whether different step sizes in the `parallel_for` make a difference to performance.
- The resulting blurred image should be saved (*grey_blurred.png*).

Part 2

Update the current image loading code from splitting the RGB values in a sequential approach to a parallel method:

- The code needs to load two colour images and compute an image with the values of the absolute differences between the two input images
 - $Q(i, j) = P_1(i, j) - P_2(i, j) \vee$
- With an image of absolute differences computed, a binary threshold operation should be applied to the image that makes any pixels that are not black into white.
- The resulting thresholded image of absolute differences should be saved (*RGB_processed.png*).
- You will need to count the number of white pixels in the image and work out the percentage of total white pixels in the final image.
- Once all white pixels have been counted you will need to place a single red pixel at a random location in the image. Once placed you will need to search the image for the red pixel and output its location.

Part 3

You need to write a short section (500 words) in addition to your report demonstrating how the image blurring code in Part 1 could be implemented using OpenCL (A full OpenCL implementation is not required but code snippets should be included to aid your discussion).

Deliverables

- A report documenting your test plan, results, code snippets and discussion also containing the source code of your implementation as an appendix.