Parallel Programming Projects

Project Ideas for three Team Members

1. Parallel Low Pass Filter
2. Parallel High Pass Filter

Project Ideas for Four/Five Team Members

1. Parallel Low Pass Filter + Task
2. Parallel High Pass Filter + Task

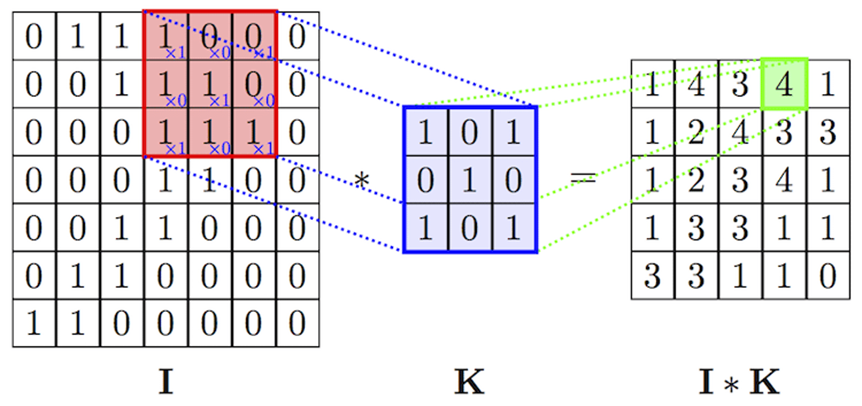
**General Notes:**

* Your code has to handle the problem in **parallel,** as this will be the primary thing to be evaluated.
* Test your work on a large variety of inputs with different number of nodes
* **The Number of MPI Nodes “Processes” should be dynamic and the input size is also dynamic**

Project #1: Parallel Low Pass Filter (Blurring)

Description:

It is used to make images appear smoother. Low pass filtering smooths out noise. It allows low frequency components of the image to pass through and blocks high frequencies. Low pass image filters work by convolution which is a process of making each pixel of an image by a fixed size kernel.



Imagine that the given image is I, the kernel is K. If we applied the filter on the red region at I, the result will be computed by aligning the kernel onto the image part then doing basic multiplication between the aligned elements as: (1\*1 + 0\*0 + 0\*1) + (0\*1 + 1\*1 + 0\*0) + (1\*1 + 1\*0 + 1\*1) = 4. Every kernel based filter has its kernel, the low pass filter has this as a kernel:



For the border elements of the input image (e.g. pixel (0,0) ), there are two ways to handle this either by zero padding or border replication. You can handle the case by any method of these. To see more on that check [this](https://www.youtube.com/watch?v=C_zFhWdM4ic&feature=youtu.be&t=71) video.

Input and Output:



\*Task:: make it work on a dynamic kernel filter size instead of fixed 3x3.

* render your results
* Test your code on 1, 25, 50, 100 pictures
* Record your Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 pic | 25 | 50 | 100 |
| Serial code |  |  |  |  |
| MPI Sol1 |  |  |  |  |
| MPI Sol (task) |  |  |  |  |

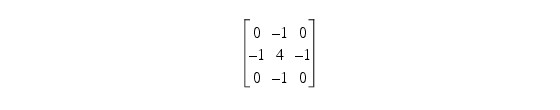
**Deliverables**

* The source code
* A report on the findings and enhancements made to performance.
* The project discussion will be individual

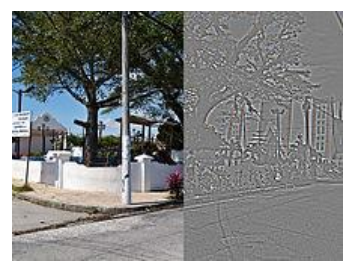
Project #2: Parallel High Pass Filter

Description:

It is used to make the image appear sharper. Note, if there is no change in intensity, nothing happens. But if one pixel is brighter than its neighbors, it gets boosted. High pass filters amplifies noise. It allows high frequency components of the image to pass through and block low frequencies. The image will The idea **is the same** as the Low Pass Filter above but with using another kernel which is:



Input and Output:



\*Task: make it work on a dynamic kernel filter size instead of fixed 3x3.

* render your results
* Test your code on 1, 25, 50, 100 pictures
* Record your Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 pic | 25 | 50 | 100 |
| Serial code |  |  |  |  |
| MPI Sol1 |  |  |  |  |
| MPI Sol (task) |  |  |  |  |

**Deliverables**

* The source code
* A report on the findings and enhancements made to performance.
* The project discussion will be individual