

Assignment OpenMP

Parallel and distributed computing

Image processing

Digital images are stored as 2D array of pixels. Each pixel in a color image is represented by 3 bytes (called the color depth). One byte for each of the RGB colors (red, blue, and green). An alternative to the 24-bit color depth is the indexed color mode. In the indexed color mode, fewer number of bits are used to index through a color map. In grayscale (also called black and white) images, 8-bits are stored per pixels to represent many shades of gray.

In digital image processing, a kernel or a convolution or a mask is a small matrix used to apply effects on the image like blurring, sharpening, outlining, etc. Also, they are used in machine learning for 'feature extraction'. You can find some of these kernels at [https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing)).

A 3x3 kernel is a matrix of 3x3 element that can be applied to a given image. The effect depends on the value of the elements. For example:

An edge detection Kernel is =
$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

A sharpen Kernel is =
$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

A blur Kernel is =
$$\frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

In this assignment you are to apply some of these kernels on a gray scale image to cause some effect. Follows the following steps to do that:

- 1) Start in the upper left corner of the image, place the 3 x 3 mask over the top leftmost 3 x 3 block of pixels.
- 2) Multiply each pixel value by the corresponding entry of the kernel and then take the sum.
- 3) The sum become the new pixel value in the new image that will have the desired effect.
- 4) Then move the mask to the right by 1 pixel.
- 5) When you finish a row, move the mask down by 1 row and back over to the left.
- 6) Go to step 2

Matlab

To see the image in matlab you need to do the following at matlab prompt



- 1) read the data from the file oimage.txt

```
>> A = dlmread('oimage.txt');
```
- 2) remove first row since it contains the size

```
>> A(1, :) = [];
```
- 3) See the image

```
>> figure; imshow(A, [0, 255]);
```

Q1: Serial program. Write a serial program that accept as input from the line command three parameters, the first parameter is the name of the file that contains the original image, the second parameter is the name of the file that contains the mask (given above), and the third parameter is just the name of the file to be used to store the new image.

For simplicity I will give you the image file as a two dimensional array of chars (1-byte integer). The first line of the file will contain two numbers, (number of rows and number of columns).

Your output file should have similar format as the input image. Time the run time of the execution of the program.

Q2: Parallelize the serial program using openmp. Time the execution of the parallel program when using 1, 2, ..., p cores, where p is the maximum number of cores that your machine has.

Q3: Write a 1 page summary of what you did and include the timing data, and the speedup

What to turn in?

A compressed tar ball that contains Five files:

- 1) The serial code file
- 2) The openmp parallel code file
- 3) The original image file that I provided you
- 4) The mask file
- 5) A text file that contain your writeup

Due date?

This assignment worth 2 assignments grade. You must complete and submit the assignment by Dec 4/2016