HW3: Accelerating Gaussian Smoothing with Stretch Processor

Due: April 4th, 2007.

In this assignment, you are asked to compare the performance of software Gaussian smoothing algorithm with the hardware-accelerated version implemented in the Stretch processor. Gaussian smoothing is used to blur images by removing detail and noise. Typically, the smoothing is achieved by convolving a kernel with the actual image. A kernel is an approximation of a Gaussian with a specific σ value. The basic descriptions of Gaussian smoothing and convolution operator are available at:

http://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm

Steps to complete this work:

- 1. Download the provided gray-scale image of size 256 x 256 pixels (*gaussian.bmp*). Each pixel is represented by an 8-bit value that must be read by your program.
- 2. Implement Gaussian smoothing function in the software using three 5x5 Gaussian approximation kernels (the values of *x* range from -2 to 2 and the values of σ are 1, 2 and 3). Refer to the above website for more information about 1-D and 2-D Gaussian functions and how to create the approximation matrices.
- 3. Convolve these kernels (one at a time) with the image data and generate smoothed output images. At the image boundaries, assume the pixels outside of the image to contain 0.
- 4. Create extension instruction(s) using the Stretch IDE to accelerate the performance of Gaussian smoothing. Compare the performance of these two implementations. Make any necessary changes to the data structure to best optimize the hardware solution. Notice that the Stretch processor does not support floating point; thus you need to use fixed-point arithmetic to perform convolution.

Deliverables:

- 1. Submit all the source files for both implementations. You can do this by simply creating an archive of the project directory.
- 2. Include a report that discusses
 - Major difficulties
 - Amount of time spent on the assignment
 - The performance outcome (is it as expected?)
 - Level of difficulty (0 for easy and 10 for impossible)