

CS3221 Project 1

NAME: GOH KA HIAN

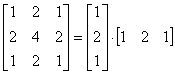
Student ID: A0164383B

|  |
| --- |
| 1. Tabulation of results |

|  |
| --- |
| 1. Separable Convolution |

Most of the filter are implemented using a 3x3 kernel direct convolution for this project. Direct convolution requires M\*N multiplications for a MxN matrix, and hence an attempt has been made to lessen the numbers of multiplications by using separable convolution which only requires M+N multiplications.

An M\*N matrix is separable if it can become a product of a 1 row and a 1 column matrices.

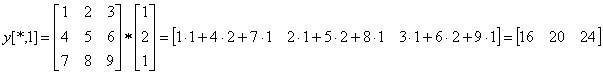


Shows a 3x3 sobel kernel separated in to a 3x1 and 1x3 matrix

Using direct convolution:

http://www.songho.ca/dsp/convolution/files/conv2d_eq23.gif, 9 multiplications in total using direct convolution.

Using Separable convolution:



First, by performing a vertical convolution, note that though figure <> shows 9multiplication, it is actually 3multiplications for 3input. A disadvantage of separable convolution is that we requires an auxiliary storage for the vertical convolution.

http://www.songho.ca/dsp/convolution/files/conv2d_eq25.gif

Secondly, performing a horizontal convolution, which requires 3 multiplications. In total, 3+3=6 multiplications for a single input

In this project, an attempt has been made

We can break the this filter function into two parts, one for performing a vertical 1D convolution first, before performing the 1D horizontal convolution.

1. **var** filt = gpu.createKernel(**function**(A) {
3. **if** (**this**.thread.y > 0 && **this**.thread.y < 600-2 && **this**.thread.x < 800-2 && **this**.thread.x >0 && **this**.thread.z <3) {
5. **return** A[**this**.thread.z][**this**.thread.y+1][**this**.thread.x]\*0.25+
6. A[**this**.thread.z][**this**.thread.y][**this**.thread.x]\*0.5+
7. A[**this**.thread.z][**this**.thread.y-1][**this**.thread.x]\*0.25;

Vertical