

# Digital Signal Processing Laboratory Work 521280S

Eero Petteri Huttunen  
student id: 2190361

## The CPU part:

Before the kernel is launched, buffers and parameters are initialized. For this implementation 6 memory buffers are created for the kernel. 2 buffers for the left and right image data as unsigned char array, 2 buffers are for the result image and a copy of this image for occlusion filter and the last 2 are for local memory.

For parameters there are global and local work group sizes, maximum disparity value and window size for SSD block matching. The global work group size is the height and width of the picture rounded up to match local work size. The local work size is hardcoded to { 32, 16 }. The reasoning is more explained in optimization part. The maximum disparity value is set to 90 and window size is 15x15. Both can be changed at the beginning of main().

## The GPU part:

Everything is implemented on one kernel since the right-left image values are needed for cross-check so it's easier to calculate both left-right and right-left values at the same time. The occlusion filler could have been implemented on another kernel but it doesn't add any speed since it has to wait for the result image to be ready anyway.

The work items whose global ids are outside the image dimensions minus window size set the value of pixel to 0. The valid work items perform SSD block matching algorithm to choose best disparity value. The cross-check has a threshold of 125 (number chosen with trial and error) and sets the pixels with more difference than this to 0. The occlusion filler takes a 11x11 window average around the pixel excluding the center value since it is always zero. Only the left-right image is saved but right-left image values are calculated for cross-check.

The testing was done with NVIDIA GeForce GT540M with max workgroup size of 1024 and local memory of 49151 bytes so some devices can create error if these numbers are too low.

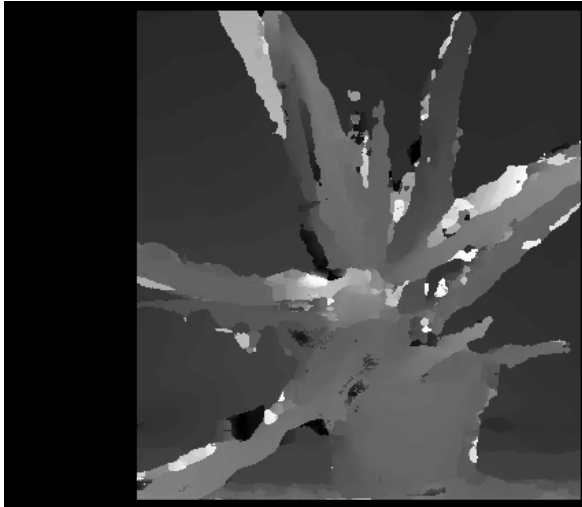
## Optimization:

The local work group size is set to be multiple of 32 because the NVIDIA gpu that the implementation was tested has 32 memory banks and this maximizes their usage. Because the banks read 32bit in one cycle and the pixel values are 8bit unsigned chars bank conflicts happen because one bank has 4 pixel values and one bank must read 4 times to get all values. The values are instead saved as uchar4 opencl vector and only the first element (.x) is used. This eats memory and the .y, .z, and .w parts are left empty.

One work group needs  $(\text{local\_width} + \text{window\_size} * 2 + \text{maximum disparity}) * (\text{local height} + \text{window\_size} * 2)$  amount of local memory. The workgroups then have partly same pixels put since the local memory cannot be shared with work groups this is necessary. The size of local memory now becomes a problem atleast with the NVIDIA GPU that the code was tested since the memory size is 49151 bytes and with 2 local buffers and { 32, 16 } local work group size the needed memory is already 32640 bytes. To ensure this implementation works in other devices the local work group size is left lower for both memory and max workgroup size. The fastest results were obtained with 32x32 local work group size that maximizes the work items on work group.

## Results:

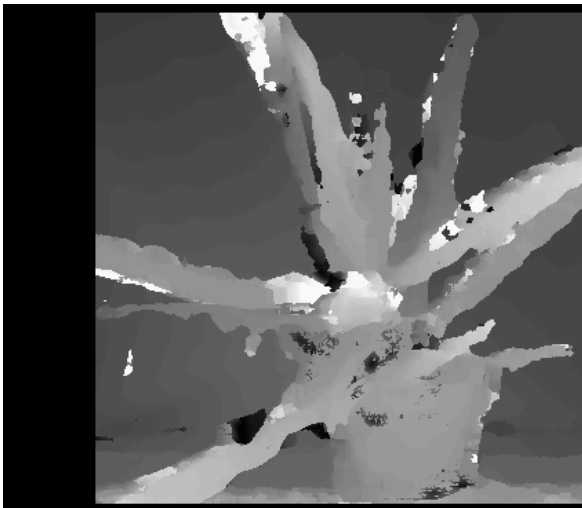
4 different result images are shown in figures 1 through 4 with different disparity values and window sizes.



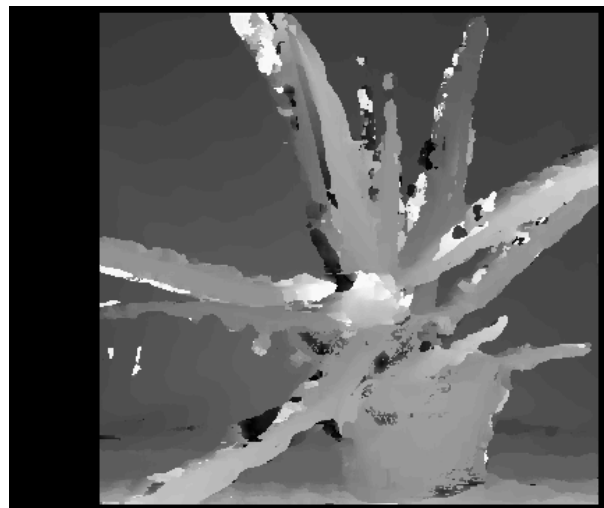
**figure 1.** left-right image with 15x15 window and 90 max disparity



**figure 2.** left-right image with 11x11 window and 90 max disparity



**figure 3.** left-right image with 15x15 window and 60 max disparity



**figure 4.** left-right image with 11x11 window and 60 max disparity

## Implementation execution times:

These are the times the GPU spent calculating.

	Disparity = 60	Disparity = 90
Window size = 11x11	0,2775 s	0,4932 s
Window size = 15x15	0,5780 s	0,8518 s

## Time taken:

Questionary : 6-8 h  
C-implementation : 12 h  
OpenCL-implementation: 25 h

Optimization :	20 h
Total :	62-65 h