ES203 PROJECT - EDGE DETECTION FILTERS USING FPGA

Prasad Athave-18110125, Priyam Tongia-18110129, Shril Mody-18110162, Shruti Katpara-18110084, Siddharth Soni-18110165

INTRODUCTION

Our project aims to apply different types of edge detection filters on an image by implementing it on FPGA using Verilog HDL. The second aim is to compare them based on the image environment. Image processing is one of the trending and essential tool in the digital world. Image Processing (filters) deals with the alter in the colour intensities of the pixels of an image. It suppresses the lower or higher frequencies of colours depending upon the filter which is in use. The output images are shown using VGA.
Images are defined as composition of RGB colors in a way to produce a broad array of colors. We are taking the size of the image as 160x115 px. So the COE file (kernel) has 96 columns and 18402 rows. In the images, the edges are defined as the set of curved line segments at which the image brightness changes sharply. Certain filters are used to detect such edges. Edge detection is a mathematical method to identify discontinuities in the image brightness.

FPGA used: Nexus 4 DDR
HDL used: Verilog HDL

IMPLEMENTATION AND PROCEDURE

1. COE and Kernel Generation

The generation of coefficient (COE) file was created using python code. COE file contains the value of each pixel of the image in binary form. Then the kernel of the image was generated by using this generated COE file.

2. Block Ram Generation

Block ram is used to store the image by using the generated COE file. The coefficient file can be called in Verilog by instantiating it as a module and then the image can be used in form of pixels inside the algorithms.

3. Algorithm

The principle is to find the convolution of each pixel of the image. Convolution is the mathematical function that is used to show the effect of one function on the other function. Here, the 'one' function is the image matrix and the 'other' is the kernel matrix of the filter. Each filter has its own kernel which is, when applied on a matrix (taken from the COE of the image), it changes the value of the middle element of the matrix. The frequency of pixel movement is so high that we do not notice that at a time, we are not having every pixel coloured simultaneously.

4. VGA Implementation

The output image or filtered image is shown on the screen by using Visual Graphics Array (VGA). VGA is an analog interface that connects the FPGA to the monitor screen. The VGA has quality of 640x480 px

DIFFERENT TYPES OF FILTERS

1. Laplacian edge - Laplacian is an operator which is applied on an image that involves taking partial double derivatives in X and Y direction and summing them up and then placing that value at the particular point in the matrix. Pixels are not continuous functions so we can not directly apply the double derivative, so we approximate dX =1 and dY = 1 hence we get the approximate kernel-(a 3X3 matrix).

0	-1	0
-1	4	-1
0	-1	0

2. Prewitt X - prewitt X filter is based on the Prewitt operator . The difference between two grayscale values of neighboring pixels along the X-direction is calculated using the kernel shown in right side.

-1	0	1
-1	0	1
-1	0	1

3. Prewitt Y - prewitt Y filter is based on the Prewitt operator . The difference between two grayscale values of neighboring pixels along the Y-direction is calculated using the kernel shown in right side.

-1	-1	-1
-1	0	1
-1	0	1

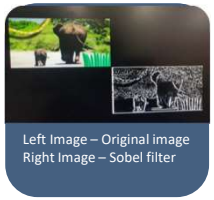
4. Sobel X - Sobel X is based on the Sobel operator which is basically difference along the X direction. The Sobel operator acts very similar to the Prewitt filter and can be formulated into the Convolutional matrix shown on right.

-1	0	1
-2	0	2
-1	0	1

5. Sobel Y - The working of the Sobel Y filter is exactly similar to the Sobel X. The difference is the only thing that it acts in the Y direction.

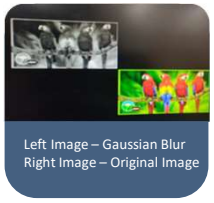
-1	-2	-1
0	0	0
1	2	1

VGA OUTPUT OF FILTERED IMAGES



Left Image – Original image
Right Image – Sobel filter

L.I.– Robinson Mask Northwest
R.I.– Robinson Mask Southeast

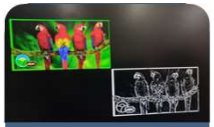


Left Image – Gaussian Blur
Right Image – Original Image

L.I.– Laplacian 8 (more precise)
R.I.– Laplacian 4

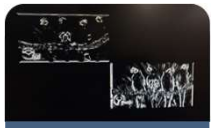
L.I.– Prewitt X
R.I.– Prewitt Y

L.I.– Original Image
R.I.– Coloured Laplacian edge

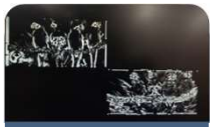


Left Image – Original image
Right Image – Sobel filter

Left Image – Laplacian 4
Right Image – Laplacian 8



Left Image – Prewitt X
Right Image – Prewitt Y



Left Image – Sobel y
Right Image – Sobel x

APPLICATION OF FILTERS

Edge detection filters are used to detect boundaries of objects in an image. These object boundaries are first step in many of computer vision algorithms like edge-based face recognition, edge-based obstacle detection, edge-based target recognition, image compression, etc.

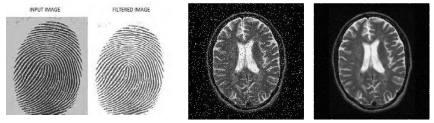
Various application of edge detection filters are :-

- Identification of fingerprints(Sobel edge)
- Satellite imaging (Canny edge)
- Self Driving cars (Laplacian edge)
- MRI scans (Sobel edge)
- License Plate Detection (Canny edge)

Other various types of filters can be used for:-

- Making the image smooth
- De-noising the image
- Sharpening the image
- Blurring the image

Famous android apps like FaceApp, Instagram, Snapchat etc. use such type of filters to improve their user's experience.



CONCLUSION

The process of designing image processing filters is complicated. Designing filters using Verilog gives more idea about the working procedure of FPGA. The storage of image in block ram provides understanding about the structure of the FPGA. The VGA display interface is a very interesting and useful tool for displaying images using FPGA. The project gave us a more in-depth understanding of the difference between the hardware description and the software programming languages. The project also helped us analyze many of the algorithms which help us detect edges. Overall Doing this project was a great learning and challenging experience.

ACKNOWLEDGEMENTS

Designing filters and implementing them on FPGA is not an easy task. It took a lot of efforts and hard work to finish this work. We want to thank our course associates Mr. Rajat and Mr. Varun, for their continuous help and support. We want to acknowledge the continuous guidance and assistance provided by Prof. Joyce Mekie. We feel very grateful to her for giving us this big learning opportunity of doing this project. We would also thank the IITGN electrical lab and electronics lab for providing us with all the required components necessary to complete our project.