Performance Efficient FPGA Implementation of 2D Image Filtering using Xilinx System Generator (XSG)

A. Radha, R. Priyakanth
Dept. of ECE, D.V.R & Dr. HS MIC College of Technology, Kanchikacherla, AP, India

Abstract
This application note provides a Xilinx FPGA solution to two-dimensional filtering with a parameterized VHDL reference design. Two-dimensional linear filtering (2D FIR) has many applications in imaging and video processing. The range of applications vary from very precise medical imaging systems to low precision industrial imaging and consumer video applications. This paper presents an architecture for filters pixel by pixel and regions filters for image processing using Xilinx System Generator (XSG). This architecture offers an alternative through a graphical user interface that combines MATLAB, Simulink and XSG and explores important aspects concerned to hardware implementation.

Keywords
Digital Image Processing, Matlab, Xilinx System Generator

I. Introduction
Image processing is a broad term describing most operations that you can apply to image data which may be in the form of a 2D, 3D or 4D waves. Image processing may sometimes provide the appropriate analysis tools even if the data have nothing to do with imaging [1]. A digital image is described in a 2D discrete space derived from an analog image in a 2D continuous space through a sampling process that is frequently referred to as digitization.

The handling of digital images has become in recent decades a subject of widespread interest in different areas such as medical and technological applications, among others. We may cite lots of examples where image processing helps to analyze or infer and make decisions. The main objective of image processing is to improve the quality of the images for human interpretation, or the perception of the machines independently. This paper focuses in the processing pixel to pixel of an image and in the modification of pixel neighborhoods and of course the transformation can be applied to the whole image or only a partial region [2-4].

The need to process the image in real time, leading to the implementation level hardware, which offers parallelism, and thus significantly reduces the processing time, which was the reason for deciding to use Xilinx System Generator, a tool with a graphical interface under the Matlab/Simulink, based blocks which makes it very easy to handle with respect to other software for hardware description.

In addition to offering all the tools for an easy graphical simulation level. The filtering of images is a technique which gives an enhancement of the image characteristics. The purpose of filtering techniques is to process an image in a way that is more appropriate than the original specific application [5]. Among the applications of filter image are: the elimination of noise, enhancing edges and contours, and so on. This paper is intended to complement and extend this earlier work in the direction of a Two-Dimensional (2D) finite group-based theory, again with the intent of applying this work to image processing, architecture filtering images System Generator, which is an extension of Simulink and consists of a bookstore called “blocks Xilinx”, which are mapped architectures, entities, signs, ports and attributes, which Script file to produce synthesis in FPGAs, HDL simulation and development tools. The tool retains the hierarchy of Simulink when it is converted into VHDL.

II. Xilinx System Generator
Xilinx System Generator (XSG) is an integrated design environment (IDE) for FPGAs, which uses Simulink, as a development environment and is presented in the form of block set. It has an integrated design flow, to move directly to the configuration file (*.bit) necessary for programming the FPGA.

One of the most important features of Xilinx System Generator is possessed abstraction arithmetic that is working with representation in fixed point with a precision arbitrary, including quantization and overflow. You can also perform simulations both as a fixed point double precision. XSG automatically generates VHDL code and a draft of the ISE model being developed. Make hierarchical VHDL synthesis, expansion and mapping hardware, in addition to generating a user constraint file (UCF), simulation and test bench and test vectors among other things [5-6].

Xilinx System Generator was created primarily to deal with complex Digital Signal Processing (DSP) applications, but it has other applications like the theme of this work. The blocks in Xilinx System Generator operate with Boolean values or arbitrary values in fixed point, for a better approach to hardware implementation. In contrast Simulink works with numbers of double-precision floating point. The connection between blocks Xilinx System Generator and Simulink blocks are the gateway blocks.

In fig. 1 shows the broad flow design Xilinx System Generator. As already mentioned, you can then move to the configuration file to program the FPGA, and the synthesis and implementation steps are optional for the user but not for System Generator.

III. Design of the Proposed Concept
For the design of filters should be meet hardware requirements, it is therefore necessary for image preprocessing prior to the same architecture. Unlike the level software processing, where the image is a two-dimensional arrangement n x m, and processed as such, at hardware this matrix must be an array of one dimension, namely a vector. Then save the image information in a ROM memory. The coordinate (i, j) suffers the following transformation.

Fig. 1: Design flow of System Generator
Coordinate latter is the position that it occupies in the ROM. We have sorted by columns, is the first to say this in memory the first column and then the second and so on until end with the values, you can choose to be kept in rows [7]. Once processed image, it applies a reverse transformation that reversed the process of a settlement to a one-dimensional two dimensional.

Fig. 2: Image Values are Stored in a ROM Memory

To access the information, you must address the ROM that contains the value or values of the pixels that are required for processing, depending on the filter to be used and thus obtain the new values of the image. Therefore requires two more blocks, the generator and address block for processing operations. As shown in Fig. 3, the processing stage consists of three blocks: the address generator, which feeds the block ROM, it is the entire image information; this in turn is used by the block Operational information processing.

Fig. 3: Block Diagram Representation of Image Processing System

IV. Data on Image Processing

We are presenting the main filter processing technique of the picture: the processing pixel by pixel and the pixel neighborhood [1]. Both are working with the intensity of the image.

A. Processing Pixel by Pixel

This technique is the simplest. It works with the intensity of each pixel [2], as shown in fig. 4.

Fig. 4: Processing Operation

Fig. 4 shows the f(x,y) is the intensity of un processing pixel, in the output we have other intensity, the coordinates are unchanged, only the intensity was modified.

B. Processing of Pixel Neighborhoods

Basically consist in transform the value of a pixel in the position (i, j) taking into account the values of neighbors pixels. For example, if we consider a pixel neighborhood and multiply with a different weight, by consider the values of the neighbors, the result of this amount is the value of the new pixel of the image output in the same position (i, j). All that remains is to define the values of the weights, which is usually defining a mask with constant values. The mask is actually a filter, so that depending on its nature, and will be the end result. For example, if we define the mask next.

\[
\begin{align*}
\frac{1}{1} & \frac{1}{1} & \frac{1}{1} \\
\frac{1}{1} & \frac{1}{1} & \frac{1}{1} \\
\frac{1}{1} & \frac{1}{1} & \frac{1}{1} \\
\end{align*}
\]

In (1) we have the strengths of the region’s image dimension equal to the mask, in this case is 3 × 3, in subsection (2) is the mask, i.e. the weights it will multiply each of the intensities of the pixels. So the intensity of f(i, j) is replaced by:

V. Results of Proposed Concept

Among the image operations we have the shine, contrast threshold applied to the image in fig. 5.

Fig. 5: Original Image
The images with low contrast can be due to various causes, such as poor lighting, lack of dynamic range in the sensor or even incorrect selection of the opening of the lens during the capture of the image. Fig. 6 shows a typical conversion used to enhance the contrast and in fig. 7 architecture.

Fig. 6: Contrast

The negatives digitized images are useful in many applications, such as medical imaging and representation in photographs of a monochrome screen with films with the idea of using the resulting negative slides as normal. Fig. 10, illustrates the use of this simple transformation and architecture in fig. 11.

Fig. 10: Negative

The operator threshold transforms an image of binary output from an image of gray scale, where the level of transition is given by the input parameter p1. Fig. 8 shows the implementation of this transformation and fig. 9, architecture.

Fig. 8: Threshold

---

Fig. 7: Architecture for Contrast

Fig. 9: Threshold Architecture

Fig. 11: Negative Architecture
Fig. 13 shows the negative image of the fig. 12 and the fig. 14 architecture.

**V. Conclusion**

The Xilinx System Generator tool is a new application in image processing, and offers a friendly environment design for processing. The filters are designed by blocks and it even supports Matlab codes through user customizable blocks. It also offers an ease of designing with GUI environment. This tool may support software simulation, but most importantly it generates necessary files for implementation in all Xilinx Systems, with the parallelism, robust, speed and automatic area minimization. These features are essentials in real time image processing applications.

**References**


Mr. R. Priyakanth received B.Tech degree in Electronics & Control Engineering from PVP Siddhartha Institute of Technology, Vijayawada, A.P, India and M.Tech degree in Communication and Radar Systems from KLUniversity, Vaddeswaram, Guntur Dist., A.P, India in 2002 and 2005 respectively. From 2005 till date he is working in the department of Electronics and Communication Engineering in Devineni Venkata Ramana & Dr. HimaSekhar MIC College of Technology, Kanchikacherla, A.P, India. Presently he is pursuing Ph.D from JNTUK, Kakinada, A.P, India. His research interests include Multimodal Signal Processing, Biomedical Image Processing.