

DTBDM(Decision-Tree-Based De-noising Method)

by

CHETHAN KUMAR H B

Abstract

The project is aimed at implementation of decision-tree-based impulse noise detector to detect the noisy pixels, and an edge-preserving filter to reconstruct the intensity values of noisy pixels ("ieeexplore.ieee.org/iel5/12/6471716/06122015.pdf") on DE1 Cyclone V soc . We use standard 512x512 test images to validate the design. Input image is sent to FPGA and the processed image is received on PC through USB blaster using Altera JTAG-AVALON bridge IP core. This project is implemented in Qsys tool (Verilog HDL).

Contents

1	Design	5
1.1	JTAG to Avalon Master Bridge	5
1.2	Block Memory	6
1.3	De-Noising Module	6
1.4	VGA Interface	6
2	Methodology	7
2.1	Simulation	7
2.2	Hardware	7

List of Figures

1.1	Block Diagram	5
1.2	DTBDM Algorithm	6
2.1	Hardware setup	8
2.2	Noisy Image	8
2.3	Output Image	8

Chapter 1

Design

This section describes project architecture and it's modules in detail. Design aspect of each individual module is explained as part of this section. Figure 1.1 depicts high level block diagram of the project.

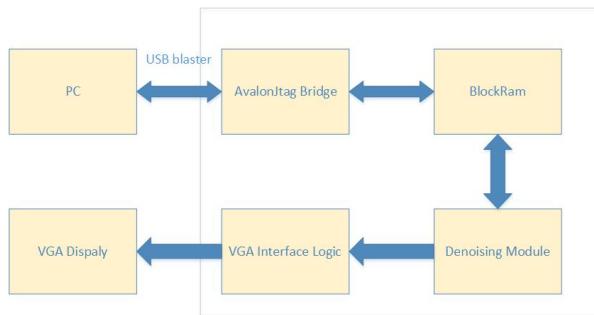


Figure 1.1: Block Diagram

1.1 JTAG to Avalon Master Bridge

The JTAG to Avalon Master Bridge core [3] provides a connection between host systems and Qsys systems via the respective physical interfaces (USB-Blaster in this project). Host system can initiate Avalon Memory-Mapped (Avalon-MM) transactions by sending encoded streams of bytes via the cores physical interfaces. Avalon Master Bridge and the JTAG to Avalon Master Bridge core accept encoded streams of bytes with transaction data on their respective physical interfaces and initiate Avalon-MM transactions on their Avalon-MM interfaces

1.2 Block Memory

We instantiate block ram of 512x512 bytes depth in our design to store an input image frame. This block ram is treated as memory mapped device by host and it is accessible through JTAG to Avalon Master Bridge core.

1.3 De-Noising Module

DTBDM [2] consists of two components: decision-tree-based impulse detector and edge-preserving image filter. The detector determines whether P_{ij} is a noisy pixel by using the decision tree and the correlation between pixel P_{ij} and its neighboring pixels. If the result is positive, edge preserving image filter based on direction-oriented filter generates the reconstructed value. Otherwise, the value will be kept unchanged.

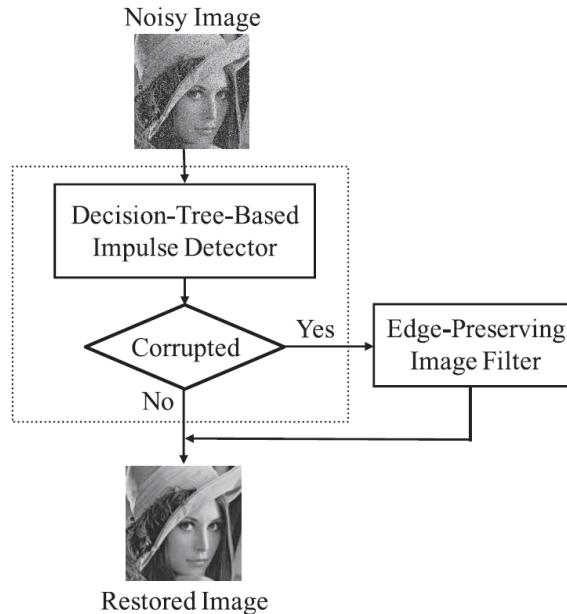


Figure 1.2: DTBDM Algorithm

1.4 VGA Interface

VGA Interface module reads processed (Filtered) image from the block RAM and pass it to the VGA display connected to the board. Same processed image is being transmitted to the display at the rate of 60fps to achieve video streaming.

Chapter 2

Methodology

2.1 Simulation

We use Altera ModelSim Starter Edition 10.0 to simulate our design and verify its functionality. We write **tcl** scripts to automate the simulation process. This tcl script is responsible for compiling verilog files, libraries and generating input vectors. RTL design verification is as follows.

- Convert a 512x512 noisy image into a text image using MATLAB or ImageJ application.
- Test bench reads test image file and send pixel data to the design.
- The pixel being processed is passed through De-noising module and the processed output pixel is written into an another text image file.
- Open resulted text image in ImageJ application

2.2 Hardware

We compile our design in Qurtus 15.1 targeting DE1 Cyclone V soc. Our hardware setup is depicted in figure 2.1. An input image is transmitted to the FPGA logic over USB-Blaster using JTAG-Avalon master Ip core and the processed image is displayed on VGA monitor.



Figure 2.1: Hardware setup



Figure 2.2: Noisy Image



Figure 2.3: Output Image

References

- [1] https://www.altera.com/content/dam/alterawww/global/en_US/pdfs/literature/ug/ug_system_c...
- [2] ieeexplore.ieee.org/iel5/12/6471716/06122015.pdf.
- [3] https://www.altera.com/content/dam/alterawww/global/en_US/pdfs/literature/hb/nios2/qts_qii...