Journal of Engineering Design and Analysis
Volume 4, Issue 2 - 2021, Pg. No. 12-15
Peer Reviewed Journal

Research Article

Hardware Architecture for Color Image Mosaicing using DCT Approach

Jayalaxmi H[1], S Ramachandran[2], Shridhar H[3]

[1]Department of Electronics and Communication Engineering, Acharya Institute of Technology, Soladevanahalli, Karnataka, India.
[2]Department of Electronics and Communication Engineering, SJB Institute of Technology, Bengaluru, Karnataka, India.
[3]Department of Electronics and Communication Engineering, Government Engineering College, Haveri, Karnataka, India.

I N F O

Corresponding Author:
Jayalaxmi H, Department of Electronics and Communication Engineering, Acharya Institute of Technology, Soladevanahalli, Karnataka, India.
E-mail Id:
jayalaxmih1973@gmail.com
Orcid Id:
https://orcid.org/0000-0003-1079-7503
How to cite this article:
Jayalaxmi H, Ramachandran S, Shridhar H. Hardware Architecture for Color Image Mosaicing using DCT Approach. *J Engr Desg Anal* 2021; 4(2): 12-15.

Date of Submission: 2021-11-25
Date of Acceptance: 2021-12-18

A B S T R A C T

It is challenging to implement and develop software-based corner detection on hardware architecture. The proposed Color Image Mosaicing with DCT Approach (CIM-DCT) is presented together with a basic hardware architecture for Verilog implementation. New optimized hardware architecture has been introduced to provide high-performance image processing in hardware while reducing computing complexity using Verilog-HDL and Xilinx 14.7 software, the recommended hardware architecture CIM-DCT was constructed. to target the Artix-7 FPGA device XC7A100T-3CSG324. For simulation of the RTL Verilog codes, ModelSim was used. For a clock rate of 50 MHz, the whole processing of a large image 1600x1200 pixels takes 23040030 ns.

Keywords: CIM-DCT, HDL, RTL, Architecture, Hardware

Introduction

Image mosaicing has become more popular as a result of constraints in the processing capabilities of traditional digital cameras, which were initially designed to combine aerial and satellite images.[1] Image mosaicing is currently used in a variety of additional applications, such as automated video monitoring to obtain a broader view of the scene , allowing for effective detection of suspicious actions.[2] It has a variety of uses, including increasing the view of underwater photographs captured with an acoustic camera,[3] document images,[4] medical imaging for creating huge panoramic images for diagnostics and computer graphics for generating a photograph with a panoramic view.[5] The existing method necessitates global image alignment, which necessitates the computation of the transform to align two images.[6] Misalignment of the pixels in the two photos after global alignment may generate blur in the composite image. Using color or grayscale information from the images to identify a place in the mosaic always necessitates precise alignment.[7] DCT has a low computational cost and takes relatively little time to execute, making it ideal for the compression of images and videos. This provides an autonomous mosaicing method for events that occur in real time that meet today's needs.[8,9] A few attempts have been made to implement image mosaicing on FPGAs in an effective manner that gives real-time response and consumes less power.[10]

FPGAs have the advantage of being able to be quickly redesigned and reprogrammed and also changed without requiring any hardware modifications. The entire mosaic algorithm may be integrated into a single chip, yielding results that are equivalent to those produced from Matlab simulations.

Proposed Color Image Mosaicing Using DCT Approach

This paper presents a DCT-based color image mosaicing method (CIM-DCT). For image mosaicing, it takes into account two image samples, each of which has two RGB components. Images 1 and image 2 are retrieved. Figure 1, shows the proposed Color Image Mosaicing core building block utilizing DCT. This system consists of DCT, IDCT and an Image Registration Module, to put it simply DCT is calculated first, Correlation, IDCT and Image Mosaicing are the next modules to be added. In the next sections, the internal components of the proposed Color Image Mosaicing System with DCT Processor are detailed. Two color image inputs are given into DCT, which will be mosaiced, as shown in Figure 1.

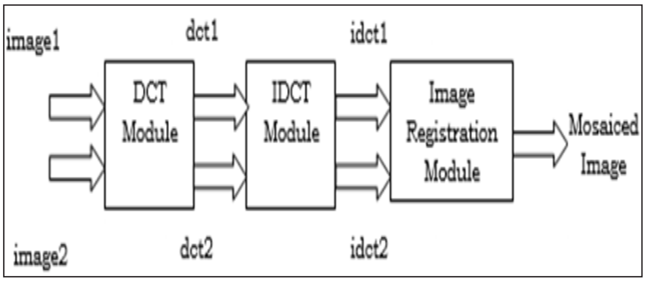

Figure 1.Basic Block Diagram of the Proposed DCT-CIM System

The CIM processor is fed two-color components, "image 1" and "image 2." R, G and B are the three color components that occur in these input images. Individual color components with an 8-bit size are sent to the DCT module, which generates DCT coefficients. The DCT module computes the coefficients for an 8x8 pixel block as two 1D transforms, "dct1" and "dct2." The IDCT module receives the output coefficients from the DCT modules and reconstructs the data into "idct1" and "idct2". The IDCT is used to generate 8 bit RGB mosaiced data output module's output is sent through an Image Registration module.

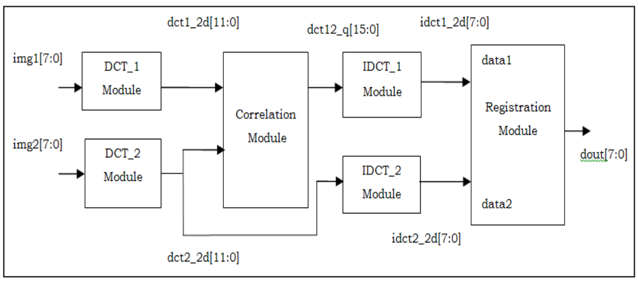

Figure 2.Proposed DCT Based CIM System

Proposed DCT Based CIM Approach

Figure 2, shows a more thorough picture of the DCT Based CIM System. It mosaics two pictures, "img 1 [7:0]" and "img 2 [7:0]", in which RGB color components are passed to the DCT 1 and DCT 2 modules, respectively, to form DCT coefficients. At the positive edge of the clock "clk" signal, the input pixels are valid. An asynchronous active low signal "rst" is used to reset the system when it is turned on. When a pixel is valid, the "start" signal is asserted and the two RGB color images "img 1 [7:0]" and "img 2 [7:0]" are fed into the DCT modules labelled "dct1 2d[11:0]" and "dct2 2d [11:0]" respectively, as illustrated in Figure 2. Other inputs include clock "clk", reset "rst" and "start", which are specified in the CIM module's architecture in the next section. Each DCT module is computed as two 1-D transformations to generate the coefficients for an 8x8 pixel block.

A block's picture information is packed into as few DCT Coefficients as feasible, allowing for picture compression if required. The DCT module computes DCT coefficients, which are then given to the correlation module to correlate the pixel block, resulting in the quotient value "dct12 q [15:0]". The image data is processed using 1D-DCT to store one-dimensional DCT constant coefficient values for multipliers. To accumulate the 64 coefficients in the first 64 regions of the RAM, the transpose of memory is employed to store DCT coefficients. After "rst" goes dormant, the RAM is enabled at the 14th clock and DCT coefficients show at the RAM1's output.

Simulation Results of CIM using DCT

The simulation results of mosaicing a color image using the DCT method are presented in this section. Image 1 inputs "img1 r [7:0]", "img 1 g [7:0]", "img 1 b [7:0]", image 2 inputs "img 2 r [7:0]", "img 2 g [7:0]", "img 2 b [7:0]" and corresponding data outputs "dout r [7:0]", "dout g [7:0]", "dout b", "done", "done g" and "done b". These inputs and outputs were retrieved using the Xilinx RTL View tool and are shown in the top module in Figure 3.

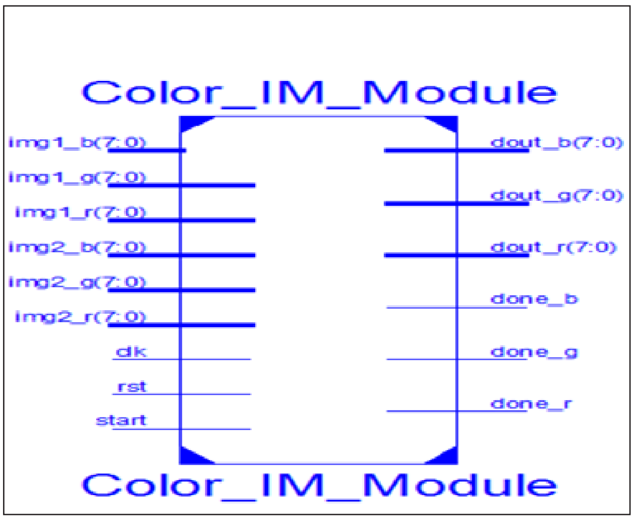

Figure 3.Color Image Mosaicing using DCT Top Module

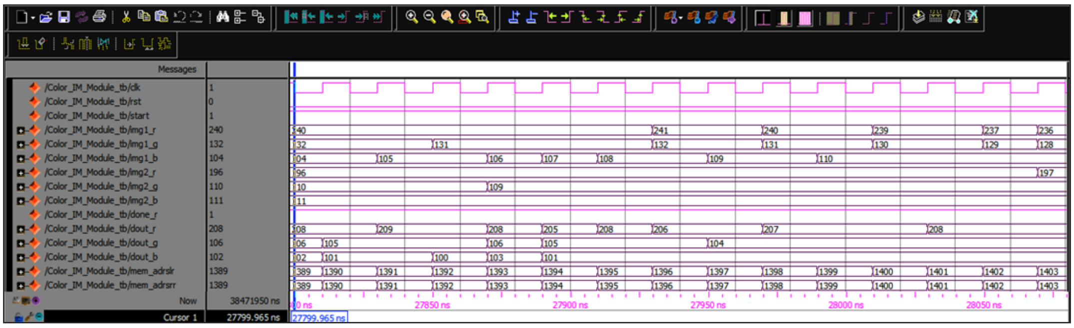

Figure 4.Color Image Mosaicing Using DCT Simulation Results, Commencement of Output Stage @ 27800 ns

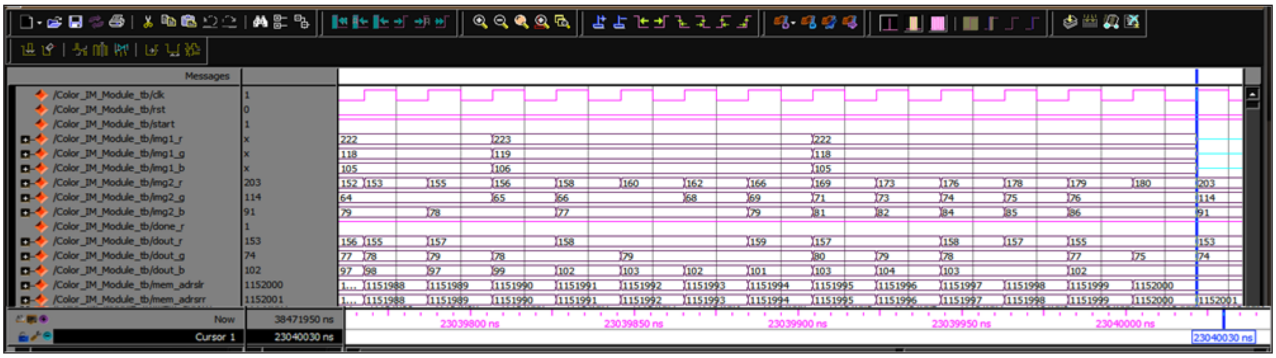

Figure 5.Overall Color Image Mosaicing Using DCT: Simulation Results Near Completion Point

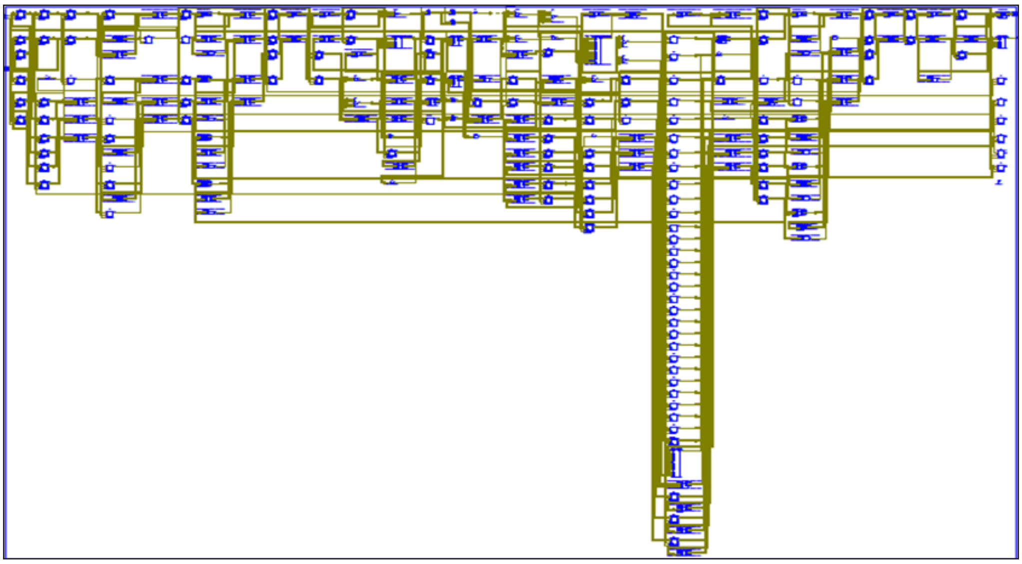

Figure 6.DCT Module RTL Architecture Reported by Xilinx Tool

The suggested DCT-based color mosaicing system architecture was written in Verilog and validated in Matlab. The Modelsim simulator simulation results are shown in Figure 4 and Figure 5, where input pictures 1 and 2 have 8 bit signals for R, G, and B and matching outputs. The results of the CIM-DCT simulation at the initial stage are shown in Figure 4. The picture inputs are saved in memory locations once the "start" signal is strong at 27800 ns.

For the mosaiced output of the Lena image at 23040030 ns, the simulation results of total CIM utilizing DCT are shown in Figure 5. Based on the outputs generated by the CIM-DCT operations, the CIM-DCT simulation output begins at 27800 ns.

Figure 6, is the RTL Schematic of a DCT Processor. Exhibit the numerical results obtained by the proposed system's DCT module in terms of time summary and resource utilization. These results were acquired after the suggested system was successfully synthesised using the Artix-7 FPGA.

Conclusion

It was created the Color Image Mosaicing System with DCT Approach (CIM-DCT). The CIM-DCT design's optimized hardware architecture ensures high-performance mosaicing of pictures in real time. For CIM-DCT design, the proposed architecture provides the best hardware complexity in real time. The system design was also subjected to a timing and chip utilization analysis. For HD (High Definition) Image/Video sequences, greater PSNR values (over 35 dB for high resolution of 1600x1200 pixels) are achievable as image sizes rise. CIM-RTL DCT's Design can mosaic 1600x1200 pixel color images at 30 frames per second. CIM-DCT outperforms CIM-SWM in terms of comparative outcomes.

References

1. Aoudia S, Samy. Satellite and Aerial Image Mosaicing-A Comparative Insight. 16th International Conference on Information Visualization (IV), IEEE, 2012.
2. Heikkila, Marko, Pietikäinen M. An Image Mosaicing Module for Wide-area Surveillance. Proceedings of the third ACM International Workshop on Video Surveillance and Sensor Networks, ACM, 2005.
3. Rzhanov Y, Linnett LM, Forbes R. Underwater Video Mosaicing for Seabed Mapping. International Conference on Image Processing, Vancouver. 2000; 1: 224-227.
4. Liang, Jian, Daniel DM et al. Camera-based Document Image Mosaicing. ICPR, 18th International Conference on Pattern Recognition. 2006; 2.
5. Abraham, Rintu, Philomina S. Review on mosaicing techniques in image processing. Third International Conference on Advanced Computing and Communication Technologies (ACCT), IEEE, 2013.
6. Zhao, Feng, Qingming H et al. Image Matching by Normalized Cross-Correlation. ICASSP Proceedings, IEEE International Conference on Acoustics, Speech and Signal Processing. 2006; 2.
7. Sawhney, Harpreet S, Kumar R. True Multi-image Alignment and its application to Mosaicing and lens Distortion Correction. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1999; 21(3): 235-243.
8. Bhosle, Udhav, Chaudhuri S et al. A Fast method for Image Mosaicing using Geometric hashing. *IETE Journal of Research* 2002; 48(3&4): 317-324.
9. Jain, Kumar D, Saxena G et al. Image Mosaicing using Corner Techniques. International Conference on Communication Systems and Network Technologies (CSNT), IEEE, 2012.
10. Zhou, Weiguo. Real-time Implementation of Panoramic Mosaic camera based on FPGA. IEEE International Conference on Real-time Computing and Robotics (RCAR), IEEE, 2016.