ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Hardware Implementation of Modified Image Guided Filter by Using Micro Blaze Processor

Myaka Sanjeev
M.Tech VLSI,
CMR Institute of Technology.

Prasad Janga, M.Tech, Ph.D
Associate Professor,
CMR Institute of Technology.

B.Bhavana
Assistant Professor,
CMR Institute of Technology.

Abstract:

Filtering is widely used in the image and video processing application for various technology. Field Programmable Gate Array (FPGA) technology has become a viable target for the implementation of real-time algorithms is suitable for the video image processing applications In this paper a explicit image filter called guided filter is proposed to remove the noise in images smoothing and sharpening of images .We proposed the systolic array architecture for modified guided filter. The guided filter is derived from the local linear model computing the filtered output by considering the content of guided image can be the input image or different image. This paper proposes that the implementation of guided filter with FPGA microblaze soft-core processor.

Key Words:

FPGA, Gate Array, guided filter, Microblaze.

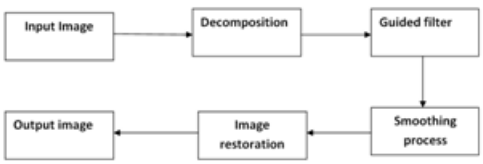
Introduction:

In image processing it is usually necessary to perform high degree of noise reduction in an image before performing higher-level processing steps. Digital images may be contaminated by different sources of noise. The imperfections in instruments used in image processing, difficulties with the data acquisition process, and interference lead to this unwanted noise generation. So we implement filters to reduce this distraction caused during image processing [2]. Most applications in computer vision and computer graphics involve the concept of image filtering to reduce noise and extract useful image structures. It is also essential for the researches in the areas of Science and Technology such as geographical information systems and astronomy. So inorder to reduce this noise based on smoothing and enhancement we implement guided filtering. The guided filter has an O(N) time (in the number of pixels N) exact algorithm for both gray-scale and color images.

And this process could be implemented in real time process based on hardware implementation by using controller as FPGA. Filtering is an image processing technique widely adopted in computer vision, computer graphics, computational photography, etc. More specifically, filtering can be applied in many applications such as noise reduction, texture editing, detail smoothing/enhancement, colorization, relighting tone mapping, haze/rain removal and joint up sampling[11], [13] The most popular technique is the edge-preserving bilateral filter [1],[8] and application of bilateral filter to image noise reduction and Durand [6] used bilateral filter on high dynamic range (HDR) images. Based on bilateral filter, joint bilateral filter is developed in flash/no-flash denoising [15] used joint bilateral filter for upsampling problems. For a bilateral filter, real-time implementation usually adopts histogram-based approximation due to its computation efficiency and memory concern [16]. Guided filter has the non-approximation characteristic and offers an ideal option for real-time filter applications on HD videos.

Proposed Guided filter:

In this proposed method, the noisy image is passed through guided image filter and some amount of noise gets reduced and the image become blur. Next the edge detection is performed. For that, the guided filter undergoes discrete wavelet transform. Form these coefficients, the edges are detected. Finally, the spatial domain and wavelet domain methods are combined together to form the final denoised output.



Block Diagram of Guided Filter

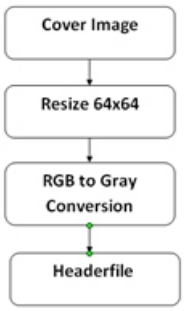
Creation of Headerfile:

By using Matlab software to create the headerfile .

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The creation of header file consisting of following steps

•Select Input image or cover image
•Resize the image into 64 x 64
•Convert Colour image into grayscale image
•Create header file in the form of .h format



Headerfile Creation

Guided Filter:

Guided image filter [Kaiming He et.al.] is an explicit image filter, derived from a local linear model; it generates the filtering output by considering the content of a guidance image, which can be the input image itself or another different image. Guided image filter has a fast and non-approximate linear-time algorithm, whose computational complexity is independent of the filtering kernel size. The guided filter output is locally a linear transform of the guidance image. This filter has the edge-preserving smoothing property like the bilateral filter, but does not suffer from the gradient reversal artifacts. Moreover, the guided filter has an O(N) time (in the number of pixels N) exact algorithm for both gray-scale and color images. The guided filter performs very well in terms of both quality and efficiency in a great variety of applications, such as noise reduction, detail smoothing/enhancement, HDR detail smoothing/ enhancement, HDRcompression, image matting/feathering and haze removal.

Algorithm of guided filter:

1.Read the image say I (gray scale image), it acts as a guidance image.
2.Make p=I, where p acts as our filtering image (gray scale image).
3.Enter the values assumed for r and ε, where r is the local window radius and ε is the regularization parameter.
4.Compute the mean of I, p, I*p
5.The compute the covariance of (I,p) using the formula: cov_Ip=mean_Ip -mean_I.*mean_p;
6.Then compute the mean of (I*I) and use it to compute the variance using the formula:- var_I = mean_II - mean_I .* mean_I
7.Then compute the value of a, b. where a,b are the linear coefficents.
8.Then compute mean of both a and b.
9.Finally obtain the filtered output image q by using the mean of a and b in the formula q = mean_a .* I + mean_b;

Smoothing Process:

The most common benefit of image smoothing is to remove the noise from the image. Different edge preserving image smoothing methods are used for preserving the important features or structures or salient edges in the image, so as to lead the improvement in the visual quality of the image. This is a method for edge preserving smoothing, which is related to the previous methods like bilateral filter and fast bilateral filter for the display of high dynamic range images signal processing approach , edge preserving decompositions, multi-scale image decomposition based on local extreme, histogram based image smoothing, L0 gradient minimization . Our method is based on the L0 gradient minimization method. This method globally control total number of non-zero gradients between pixels to enhance the prominent edges.

Image Restoration:

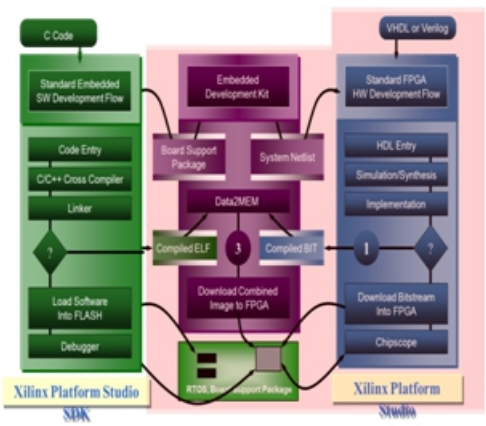
The goal of image restoration is to reconstruct the original (ideal) scene from a degraded observation. The recovery process is critical to many image processing applications. Ideally, image restoration aims to undo the image degradation process during image acquisition and processing. If degradation is severe, it may not be possible to completely recover the original scene, but partial recovery may be plausible. Typical forms of degradation during image acquisition involve blurring and noise. The blurring may be from, for example, sensor motion or out-of-focus cameras. In such a situation, the blurring function (called a point-spread function) must be known prior to image restoration. When this blurring function is unknown, the image restoration problem is called blind image restoration. Image restoration is the process of simultaneously estimating both the original image and point-spread function using partial information about the image processing and possibly even the original image.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The various approaches that have been proposed depend upon the particular degradation and image models.

Xilinx Platform Studio:

The Xilinx Platform Studio (XPS) is the development environment or GUI used for designing the hardware portion of your embedded processor system. B. Embedded Development Kit Xilinx Embedded Development Kit (EDK) is an integrated software tool suite for developing embedded systems with Xilinx MicroBlaze and PowerPC CPUs. EDK includes a variety of tools and applications to assist the designer to develop an embedded system right from the hardware creation to final implementation of the system on an FPGA. System design consists of the creation of the hardware and software components of the embedded processor system and the creation of a verification component is optional. A typical embedded system design project involves: hardware platform creation, hardware platform verification (simulation), software platform creation, software application creation, and software verification. Base System Builder is the wizard that is used to automatically generate a hardware platform according to the user specifications that is defmed by the MHS (Microprocessor Hardware Specification) file. The MHS file defines the system architecture, peripherals and embedded processors]. The Platform Generation tool creates the hardware platform using the MHS file as input. The software platform is defmed by MSS (Microprocessor Software Specification) file which defines driver and library customization parameters for peripherals, processor customization parameters, standard 110 devices, interrupt handler routines, and other software related routines. The MSS file is an input to the Library Generator tool for customization of drivers, libraries and interrupts handlers.



Embedded Development Kit Design Flow

The creation of the verification platform is optional and is based on the hardware platform. The MHS file is taken as an input by the Simgen tool to create simulation files for a specific simulator. Three types of simulation models can be generated by the Simgen tool: behavioral, structural and timing models. Some other useful tools available in EDK are Platform Studio which provides the GUI for creating the MHS and MSS files. Create / Import IP Wizard which allows the creation of the designer's own peripheral and import them into EDK projects. Platform Generator customizes and generates the processor system in the form of hardware netlists. Library Generator tool configures libraries, device drivers, file systems and interrupt handlers for embedded processor system. Bitstream Initializer tool initializes the instruction memory of processors on the FPGA shown in figure2. GNU Compiler tools are used for compiling and linking application executables for each processor in the system [6].

There are two options available for debugging the application created using EDK namely: Xilinx Microprocessor Debug (XMD) for debugging the application software using a Microprocessor Debug Module (MDM) in the embedded processor system, and Software Debugger that invokes the software debugger corresponding to the compiler being used for the processor. C. Software Development Kit Xilinx Platform Studio Software Development Kit (SDK) is an integrated development environment, complimentary to XPS, that is used for C/C++ embedded software application creation and verification. SDK is built on the Eclipse opensource framework. Soft Development Kit (SDK) is a suite of tools that enables you to design a software application for selected Soft IP Cores in the Xilinx Embedded Development Kit (EDK).The software application can be written in a "C or C++" then the complete embedded processor system for user application will be completed, else debug & download the bit file into FPGA. Then FPGA behaves like processor implemented on it in a Xilinx Field Programmable Gate Array (FPGA) device.

Conclusion:

In this paper we propose a novel explicit guided image filter. Originated from a local linear model, this filter can be used to replace the soft matting step and lead total real-time performance. Show the edge-aware and gradient-preserving properties of this filter. The guided filter is a faster and better technique than previous filter.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Edge-aware techniques have more applications in computer vision/graphics than what we have introduced in this section. In many applications, we assign each pixel an estimated value, which can be a cost, confidence, a vote or any other data.

References:

[1] B. Weiss, "Fast Median and Bilateral Filtering," ACM Trans. Graphics, vol. 25, no. 3, pp. 519-526, July 2006.

[2] C. Liu, W. Freeman, R. Szeliski, and S. B. Kang, "Noise estimation from a single image," in Proc. IEEE CVPR, vol. 1, 2006, pp. 901.

[3] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast Cost-Volume Filtering for Visual Correspondence and Beyond," in Proc. IEEE Conf. CVPR, 2011, pp. 3017–3024.

[4] C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images," in Proc. IEEE 6th ICCV, 1998, pp. 839– 846.

[5] Chieh-Chi Kao, Jui-Hsin Lai Shao-Yi Chien, "VLSI Architecture Design for Guided Filter for 30 Frames/s Full HD Video" IEEE Trans. on circuits and systems for video technology, vol. 24, no. 3, march 2014.

[6] F. Durand and J. Dorsey, "Fast Bilateral Filtering for the Display of High Dynamic- Range Images," ACM Trans. Graph., vol. 21, no. 3, pp. 257–266, Jul. 2002.

[7] Hosni, C. Rhemann, M. Bleyer, and M.Gelautz, "Temporally Consistent Disparity and Optical Flow via Efficient Spatiotemporal Filtering," in Advances in Image and Video Technology, LNCS. vol. 7087, Y.-S. Ho,Ed. Berlin, Germany: Springer, 2012, pp. 165–177.

[8] J. Kopf, M. Cohen, D. Lischinski, and M. Uyttendaele, "Joint Bilateral Upsampling," Proc. ACM Siggraph, 2007.

[9] J. Zhang, L. Li, Y. Zhang, G. Yang, X. Cao, and J. Sun, "Video Dehazing with Spatial and Temporal Coherence," Vis. Comput., vol. 27, pp. 749–757, Jun. 2011.

[10] K. He, C. Rhemann, C. Rother, X. Tang, and J. Sun, "A Global Sampling Method for Alpha Matting," in Proc. IEEE Conf. CVPR, 2011, pp.2049–2056.

[11] K. He, J. Sun, and X. Tang, "Guided Image Filtering," in Proc. 11th ECCV: Part I, 2010, pp. 1–14.

[12] K. He, J. Sun, and X. Tang, "Single Image Haze Removal Using Dark Channel Prior," Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2009. 908G.

[13] P. Bauszat, M. Eisemann, and M. Magnor, "Guided Image Filtering for Interactive High-Quality Global Illumination," Computer Graphics Forum, vol. 30, no. 4, pp. 1361-1368, June 2011.