Real Time Morphological Image Contrast Enhancement in Virtex FPGA

Jerzy Kasperek

Institute of Electronics, AGH Technical University Krakow
al. Mickiewicza 30, 30-059 Krakow, Poland
kasperek@uci.agh.edu.pl

Abstract. This paper describes the implementation of the real time local image contrast enhancement method. The system is based on Virtex FPGA chip and enhances the angiocardiographic data using the modified mathematical morphology multiscale TopHat transform. The morphological TopHat transform proved its effectiveness but the direct real time pipeline implementation of the multiscale version requires too many memory blocks. The author proposes a slight modification of the algorithm and presents satisfactory image contrast enhancement results and an efficient FPGA implementation. Proposed pipeline architecture uses the structural element decomposition and employs the Virtex *BlockRam* modules effectively. The processing kernel realises the contrast enhancement for the 512 x 512 image data with 8 bits/pixel representation in the real time in one XCV-800 Virtex chip.

1. Introduction

Mathematical morphology, hereafter referred to as MM, is a well-known and effective image processing environment. MM is successfully used in image filtering, segmentation, classification and measurements, pattern recognition or texture analysis, and synthesis. Yet MM algorithms generally require a significant computation power and usually in the real time image processing tasks the dedicated hardware is used. The new high capacity FPGA devices with on-chip RAM allows effective implementation even of complex morphological operations on reconfigurable hardware, which can effectively change its architecture to suit the processed task.

The paper describes the implementation of the real time local image contrast enhancement method based on MM TopHat transform. The proposed method was verified on human heart angiographic data DICOM records. During the clinical examination the patient's heart is irradiated by X-rays 25 (or 12.5) times a second with concurrent injection of a radiopaque substance to the examined vessels. Image data are digitally recorded, processed, and made available over the LAN for the medical inspection on CRT. They are also compressed and stored in DICOM standard for further inspection and analysis. Single frames have resolution of 512 x 512 with 8 bits/pixel representation, and generally show poor contrast, which inspired the work presented here. To enhance the image contrast the MM multiscale TopHat transform

was evaluated. Unfortunately, the direct implementation of the classic approach was not satisfied, assuming FPGA resources requirements. After slight modification, the author noticed satisfactory image contrast enhancement results and an efficient Virtex FPGA implementation.

Presented angiographic data are used with kind permission of the medical staff of the Collegium Medicum of the Jagiellonian University in Krakow.

2.   Fundamental Morphological Operations

Mathematical morphology is a theory devised for the shape analysis of objects and functions. MM operators treat the processed image as the set and are made of two parts: a reference shape – called the structuring element (SE) or function that is translated and compared to the original function all over the plane and a mechanism that details how to carry out the comparison. The set studied here represents either the objects of a binary image or the subgraph of the gray tone image.

The fundamental morphological operations are called the erosion and the dilation. Erosion of a set X by a structuring element B is denoted as $\varepsilon_B(X)$ and defined as the locus of points, x, such that B is included in X when its origin is placed at x.

$$\varepsilon_B(X) = \{x | B_x \subseteq X\} \tag{1}$$

Hence, the eroded value at a given pixel is the minimum value of the image in the window defined by the structuring element when its origin is at x.

$$[\varepsilon_B(f)](x) = \min_{b \in B} f(x+b) \tag{2}$$

Dilation is the dual operator of the erosion. The dilation of a set X by a structuring element B is denoted as $\delta_B(X)$ and defined as the locus of points, x, such that B hits X when its origin coincides with x.

$$\delta_B(X) = \{x | B_x \cap X \neq 0\} \tag{3}$$

Thus, the dilated value at a given pixel is the maximum value of the image in the window defined by the structuring element when its origin is at x.

$$[\delta_B(f)](x) = \max_{b \in B} f(x+b) \tag{4}$$

Hardware implementation of the dilation and the erosion for a small SE is usually the direct interpretation of the (2), and the (4). For larger SE its decomposition property (5) is frequently used.

$$\delta_{B_2}\delta_{B_1} = \delta_{\left(\delta_{B_2} B_1\right)} \quad \text{and} \quad \varepsilon_{B_2}\varepsilon_{B_1} = \varepsilon_{\left(\varepsilon_{B_2} B_1\right)} \tag{5}$$

Hence, the eroding or dilating with a large SE (figure 1 − B) could be replaced by sequences of operations with smaller SE (B_1 and B_2).
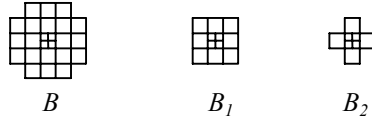
B   B₁   B₂

Fig.1 Structuring element decomposition

Erosions and dilations could be combined to create more complex morphological transforms. To present the used TopHat transform we introduce the opening and the closing. The opening γ of an image f by a structuring element B is denoted by $\gamma_B(f)$ and it is defined as the erosion of f by B followed by the dilation with the transposed SE $\hat{B}$:

$$\gamma_B(f) = \delta_{\hat{B}}[\varepsilon_B(f)] \qquad (6)$$

The closing ϕ of an image f by a structuring element B is denoted by $\phi_B(f)$ and it is defined as the dilation of f by B followed by the erosion with the transposed SE $\hat{B}$:

$$\phi_B(f) = \varepsilon_{\hat{B}}[\delta_B(f)] \qquad (7)$$

Usually the symmetric SE ($\hat{B} = B$) is used and the opening and the closing are just the combination of the erosion and the dilation.

Openings are the anti-extensive transformations are and closings are extensive transformations. Therefore, they are always satisfy the following ordering relationship (I is the identity transform).

$$\gamma \leq I \leq \phi \qquad (8)$$

3.   The TopHat Transform

The TopHat transform is a very useful tool for extracting features less the structuring element chosen from the processed image. There are two versions of the TopHat transform: the White TopHat (or TopHat by opening) denoted *WTH* and the Black TopHat (or TopHat by closing) denoted *BTH*. The *WTH* transform extracts the bright details from the background and is defined as the difference between the original image f and its opening γ.

$$WTH(f) = f - \gamma(f) \qquad (9)$$

The dual transform of the *WTH* with respect to set complementation is Black TopHat transform. In practice it is defined as the difference between closing $\phi(f)$ of the image f and the original image.

$$BTH(f) = \phi(f) - f \qquad (10)$$

The *BTH* transform extracts dark features from the image background. These remarks suggest that applying a combination of the *WTH* and the *BTH* we can achieve the contrast enhancement in the processed image. Figure 2a presents the original signal *f* and its erosion (in gray) by *B*. The figure 2c presents the construction of the *WTH* and the figures 2d, 2e, and 2f – the construction of the *BTH*. Finally, the figure 2g presents the corrected signal (the bold line) and its origin. It may be noticed that every peak and trough covered by *SE* has been enhanced. What is important, this correction has not changed the relations between particular signal samples and at the same it works independently from sample values.

The fact that the signal correction depends on the SE size was used by Mukhopadhyay and Chanda [9]. They propose to compute the new pixel value as the sum of original signal and *WTH* & *BTH* pyramids resulting from processing a successively enlarged *SE* sample. The new value $\tilde{g}$ (r,c) for the pixel with the (r,c) coordinates is calculated from:

$$\tilde{g}(r,c)= g(r,c)+0.5\sum_{i=n}^{m} WTH_{iB}(r,c)-0.5\sum_{i=n}^{m} BTH_{iB}(r,c) \qquad (11)$$

The selected 3x3 SE is enlarged in every pyramid level. Constant (e.q. 0.5) values were chosen to avoid saturation effect and to enhance both bright and dark features uniformly. In [9] result on the MR human brain image enhancement was presented for n=1 and m=6.

a) Erosion $\varepsilon_B(f)$ b) Opening $\gamma_B(f)$ c) $WTH_B(f)$ g) *TopHat* signal correction

d) Dilation $\delta_B(f)$ e) Closing $\phi_B(f)$ f) $BTH_B(f)$

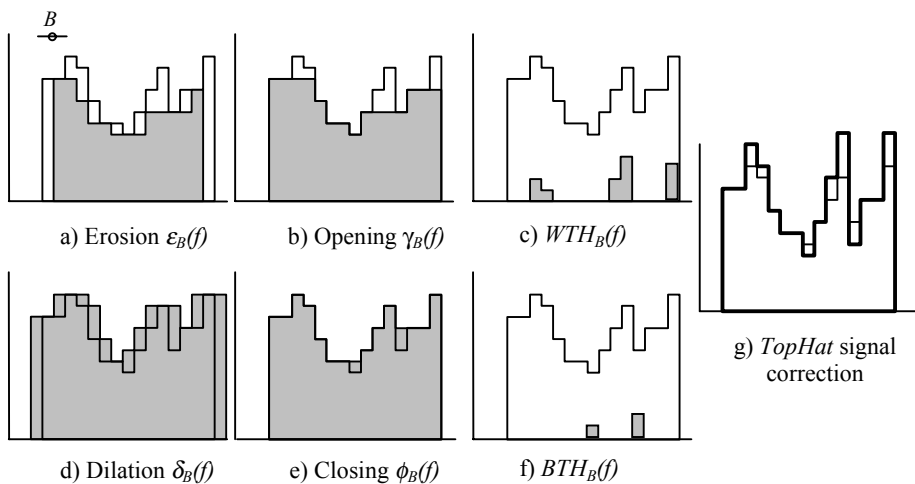Fig.2 *TopHat* transform contrast correction

4.   The Multiscale TopHat Transform Hardware Implementation in Virtex

Figure 3 presents direct implementation of the algorithm (11). The SE decomposition property (5) allows pipelining the multiscale TopHat computing. For clarity's sake, delay blocks for WTH_{iB} and BTH_{iB} products as well as for the original image are omitted. The erosion and the dilation are made with 3x3 SE. To achieve full pixel neighbourhood, the standard method utilising two delay lines is applied. Virtex *BlockRAM* module with two counters as the address pointers fulfils this FIFO task. Each XCV *BlockRAM* module is capable to store a single image line (512 pixels x 8bits). The main disadvantage of this implementation is the number of *BlockRAM* required. The number R_n of *BlockRAM* required only for the erosion /dilation modules of the n-stage multiscale TopHat realisation is given by (12).
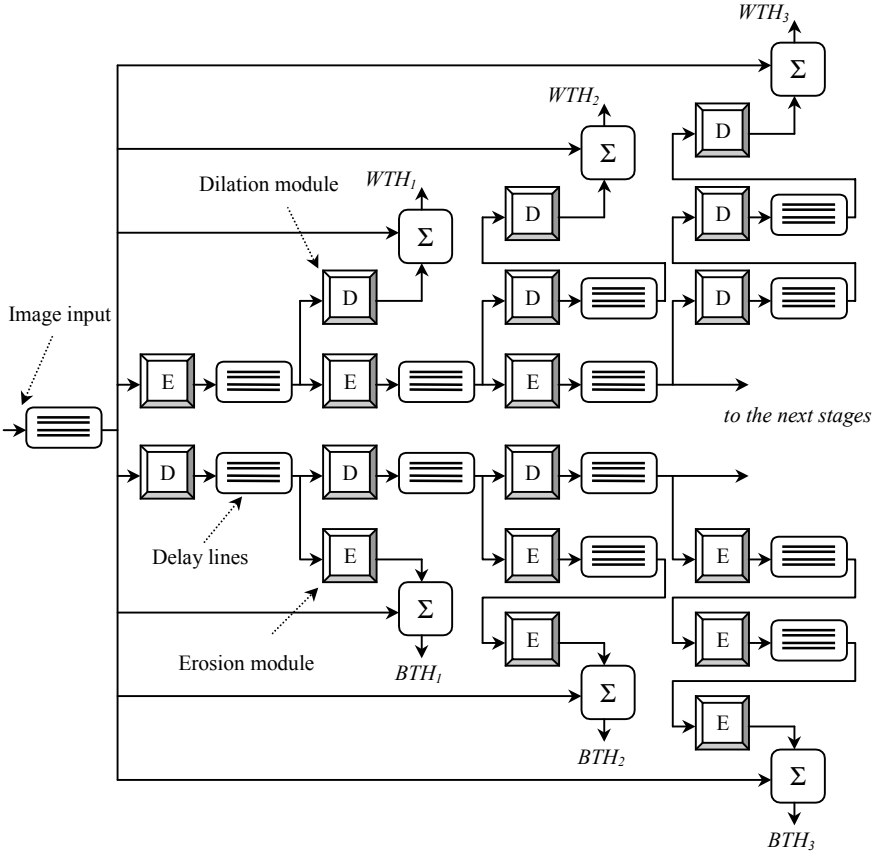

Fig.3 The multiscale TopHat transform hardware implementation

$$R_n = 2 + \sum_{i=1}^{n} 4 * i \tag{12}$$

These constraints on available *BlockRAM* modules encouraged the author to study how to simplify the algorithm and decrease the number of required delay lines. Simulation with MATLAB with specialised *SDC Morphology Toolbox* [13] proved that some kind of the TopHat transform degradation gives interesting results on image enhancement and much less hardware resources are required to implement it.

5.   The TopHat Transform Modification

Analysis of the block diagram from figure 3 suggests that the most *BlockRAM* consuming branch is the supplementation of the main processing stream for the larger SE. At this point, it is worth checking what the consequences of the degrading the TopHat transform are. Figure 4 shows the TopHat second stage case. In figure 4a an original signal and its double erosion is shown, yet instead of the double dilation known from the classic approach, we dilate only once – as in figure 4b. Next, we compute WTH_M (figure 4c). In the same way we construct BTH_M (figures 4d, 4e, and 4f). The final result is given in figure 4g. Despite small artefacts, we can notice much greater contrast enhancement than in the classic approach. After the MATLAB simulation, the desired number of stages for the proposed method was set to five.
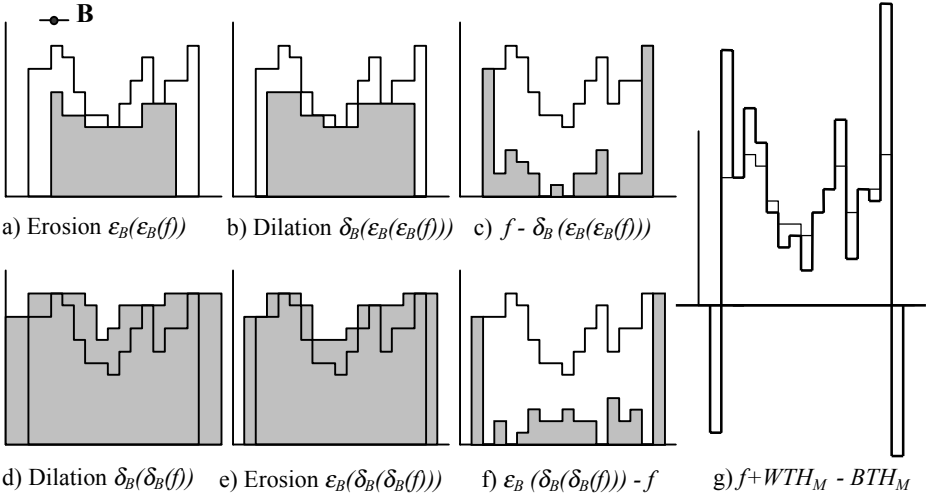

a) Erosion $\varepsilon_B(\varepsilon_B(f))$ b) Dilation $\delta_B(\varepsilon_B(\varepsilon_B(f)))$ c) $f - \delta_B(\varepsilon_B(\varepsilon_B(f)))$

d) Dilation $\delta_B(\delta_B(f))$ e) Erosion $\varepsilon_B(\delta_B(\delta_B(f)))$ f) $\varepsilon_B(\delta_B(\delta_B(f))) - f$ g) $f + WTH_M - BTH_M$

Fig.4 The modified *TopHat* transform results

6. Architecture for Modified Multiscale TopHat Hardware Implementation in Virtex FPGA

Implementation of the proposed algorithm was tested on XESS XSV-800 prototype board [17]. Besides the Virtex 2.5V XCV-800 device, two independent blocks of SRAM were used as the video buffers, as well as SVGA and RS232 periphery devices. In the future work, the Media Access Controller project is planned to allow work with real cardioangiographic data over the LAN. So far, the simple 115 kbps UART was used to test the processing kernel. A video controller has been developed to visualise the enhanced image in the RS-343A standard. SVGA 800x600 mode was used with 60Hz vertical refresh rate and 40 MHz pixel clock.

Simulation framework is presented in figure 5. VHDL possibilities to integrate text files with the actual data were intensively used. The cardioangiographic DICOM CDROM data were read by the *Osiris* medical imaging software package [10]. Then single image frames were exported into the TIFF format. Further, the TIFF image was converted into the PGM text file format which clear for the behavioural SRAM video buffer VHDL model. After processing, the output video memory buffer, the VHDL behavioural SRAM model, stored the processed data in the output file. Finally, the comparison with the MATLAB data was made.
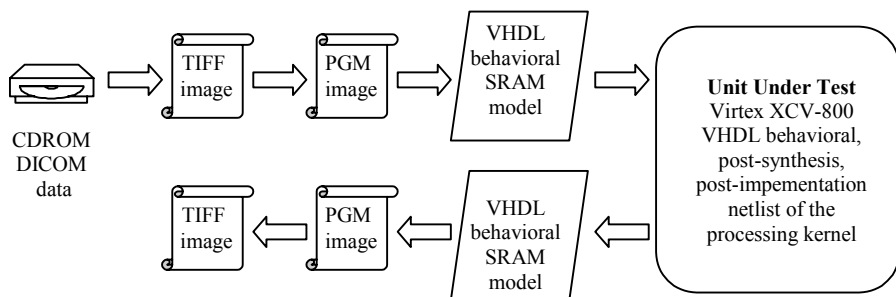

Fig.5 The simulation framework

The proposed architecture of the processing kernel is presented in figure 6.

To use logic resources and *BlockRAM* modules efficiently, the time multiplexing scheme in the computing path was designed. The processed image is divided into two half-planes: left and right. First, the left one is processed and the first delay lines unit is uploaded with 256 pixels from the left half-plane. Then the first erode/dilate stage performs its operation and the MUX0 unit uploads the main processing path (the lower) in such a way that the eroded and dilated pixels are mixed on time domain. Please, note that the pixel flow is doubled from this point. Then the second erode/dilate stage performs their job. The main processing path is now filled with double eroded/dilated pixels. Please, note also that when the dilation/erosion unit processes pixels for the main path to the next stage, the other one computes the Black/White TopHats to the upper path.
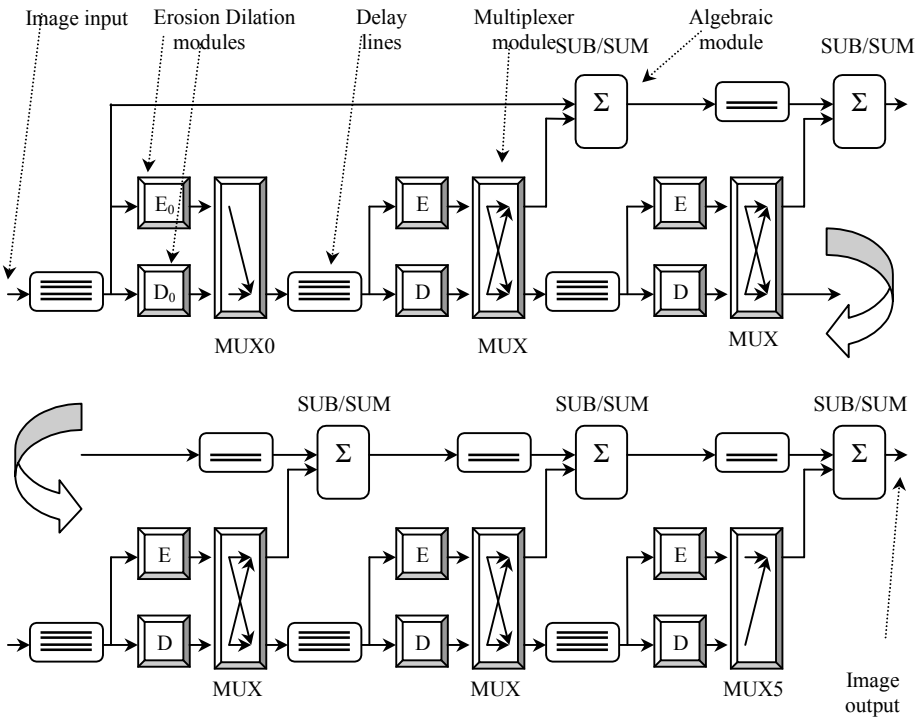
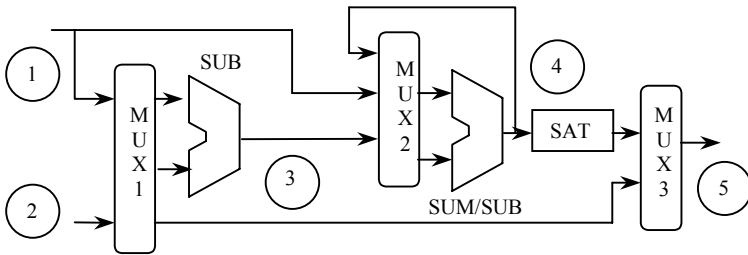Fig.6 The modified multiscale TopHat transform kernel pipeline implementation

Fig.7 Algebraic module architecture

The algebraic module architecture is presented in figure 7. Presenting the signals in the circuit nodes explains its operation most efficiently. The point '1' is the original image multiplexed with calculated TopHat from the previous stage, whereas the point '2' corresponds to the input data from upper path from figure 6. The MUX1 module prepares the data for the WTH_{MOD} & BTH_{MOD} calculation according to (9) and (10). The point '3' is WTH_{MOD} & BTH_{MOD}. Please, note that due to (8) there is no need for saturation correction at this point. MUX2 together with SUM/SUB module performs addition $f + WTH_{MOD}$ and in the next cycle subtraction $(f + WTH_{MOD}) - BTH_{MOD}$ (point '4'). The SAT block corrects the optional pixel range correction. Finally, the

MUX3 mixes again the original image with just calculated TopHat for the next processing stage (point '5').

Processing image using half planes avoids extravagance in the application of Virtex *BlockRAM* modules and allows efficient logic resources utilisation. Also this way allows one data pass algorithm, which simplifying the memory management. The only additional cost is the requirement to process middle image columns twice due to the border effect.

Figure 8 presents a frame of the source data with its histogram and figure 9 presents the enhanced contrast image and its smoothed histogram by proposed modified TopHat transform. The vessels edge enhancement and overall visibility is noticeable and there is no "the block effect" like in the convolutional edge detection technique on the DICOM compressed image data.
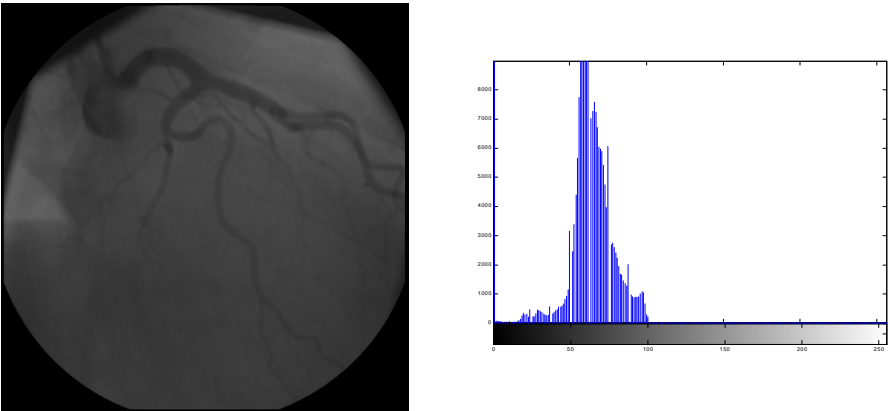

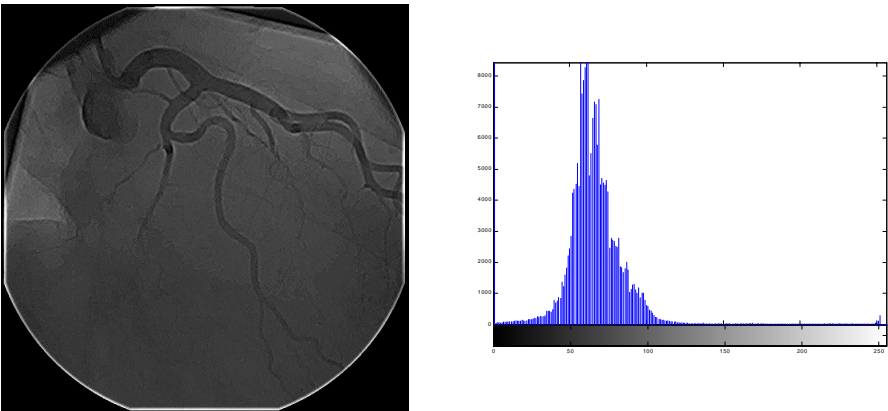Fig.8 The source frame of the cardioangiography data with its histogram

Fig.9 The processed frame and its histogram

7.   Conclusions and Future Work

The proposed architecture was successfully implemented on XESS XSV-800 prototype board. The processing kernel uses only 16 *BlockRAM* modules and 18% available slices from the XCV-800 chip which makes room for additional processing units. So far, the proposed algorithm was tested with the auxiliary serial 115kbits UART. In the future planned LAN interface will allow work with the full speed. Also, an integration a small microcontroller (Xilinx KCPSM is considered) will add the flexibility to the whole system.

The main data stream currently works well with 50 MHz clock. The processing time for one frame is about 11 ms, which is 25% the time allowed. This gives time for additional processing tasks which are also planned.

JBits tool suite is also considered as a very promising for adding the computational power to the reconfigurable hardware image processing platform.

Acknowledgements

References

[1] Chaudhiri A.S., Cheung P.Y.K., Luk W., *A Reconfirable Data-Localised Array for Morphological Algorithms.* Proceedings of the International Workshop on Field-Programmable Logic and Applications, (FPL), Springer-Verlag 1997.

[2] Coltuc.D, Pitas I., *Fast Runnig Max/Min Filters*. . Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing NSIP-95 Neos Marmaras, Greece

[3] Dougherty E.R., Astola J. T., *Nonlinear Filters for Image Processing*, SPIE Optical Engineering Press, USA, 1999.

[4] Gasteratos A., Andreadis I., *Non-linear image processing in hardware,* Pattern Recognition Vol. 33, 2000 pages 1013-1021.

[5] Giardina Ch. R., Dougherty E.R., *Morphological Methods in Image and Signal Processing*, Prentice-Hall, Englewood Cliffs NJ, USA 1988.

[6] Goren S., Balkir S., Dundar G., Anarim E., *Novel VLSI Architectures for Morphological Filtering*. . Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing NSIP-95 Neos Marmaras, Greece.

[7] Haralick R. M., Shapiro L. G., *Computer and Robot Vision*, Addison Wesley Publishing Company, USA, 1992.

[8] Jonker P. P., *Morphological Image Processing: Architecture and VLSI design*, Kluwer Technische Boeken B.V., Holland 1992.

[9] Mukhopadhyay S., Chanda B., *A multiscale morphological approach to local contrast enhancement,* Signal Processing, Vol.80, 2000, pages 685-696.

[10] *Osiris ver* 3.6, University Hospital of Geneva. www.expasy.ch/UIN

[11] Pitas I., Venetsanopoulos A.N., *Nonlinear Digital Filters Principles and Applications,* Kluwer Academic Publishers, USA 1990.

[12] Schavemaker J.G.M., Reinders M.J.T., Gerbrands J.J., Backer E., *Image sharpeninig by morphological filtering,* Pattern Recognition Vol. 33, 2000, pages 997-1012.

[13] SDC Information Systems, *SDC Morphology Toolbox for Matlab 5 User's Guide, 2000*. www.morph.com

[14] Serra J., *Image analysis and mathematical morphology* Academic Press 1988

[15] Soille P., *Morphological Image Analysis Principles and Applications*, Springer-Verlag, Berlin Heidelberg, 1998.

[16] Woolfries N., Lysaght P., Marshall S., McGregor G., Robinson D., *Fast Adaptive Image Processing in FPGAs Using Stack Filters.* Proceedings of the International Workshop on Field-Programmable Logic and Applications, (FPL), Springer-Verlag 1998.

[17] X Engineering Software System Corporation. www.xess.com

[18] Xilinx *Data Book 2000.* www.xilinx.com