FPGA Implementation of Morphologic Primitives

Agustín Cruz Contreras[1], Juan Luís Díaz de León Santiago[2]
Oleksiy Pogrebnyak[3] and Pablo Manrique Ramírez[4]

Instituto Politecnico Nacional, Centro de Investigacion en Computacion, Ave. Juan de Dios Batiz S/N, C.P. 07738, Mexico, D.F., Mexico
[1]aguscruz@prodigy.net.mx, [2]jdiaz@cic.ipn.mx, [3]olek@pollux.cic.ipn.mx, [4]pmanriq@cic.ipn.mx

Abstract. Specific hardware architecture for morphologic video image processing is proposed. The developed system processes high resolution black and white images in real time. The principle of the presented architecture is based on the local image data processing at the current moment. The image rows needed to cover the required pixel vicinity are formed by the use of shift registers. At the processing stage the operations of erosion and dilatation are performed using logic operators AND and OR. The hardware implemented in reconfigurable FPGA with a density of 100K logic gates and maximal frequency of 200 MHz.

Keywords: FPGA, morphologic image processing, edge detection

1 Introduction

Nowadays near real-time image processing systems are widely used in different applications, such as industrial monitoring, traffic control, vehicle inspection, robotic vision, security and vigilance, medical image processing. The data in such applications are often characterized by the presence of dynamic environment that requires the vision system must produce a fast response. Attempts to fit the requirements imposed for real-time image processing systems ususally are based on parallel type architectures: digital signal processors (DSP), PC-clusters, Single Instruction Multiple Data systems (SIMD), for example, 3DNow! from AMD y SSE (Streaming SIMD Extensions) from Intel, IMAP-VISION PCI board [1] that uses SIMD architecture and 16 processors.

The parallelism offer a lot of benefits, but the implementation of the processing algorithms is can not be done in one step. Under the requirement of a immediate response from the vision systems various alternatives were developed. Besides, the specific architectures plan algorithm implementation as a 100% hardware solution that provides dedicated resources for the algorithm execution. This alternative is usually implemented as Application Specific Integrated Circuits (ASIC), and FPGA can be used in case when the algorithm complexity permits it.

From the other hand, mathematical morphology theory arisen in 60's. Morphologic image processing techniques permit to extract easily image components useful for image region representation and description, such as edges, skeletons and convex forms. The mathematical morphology language is the theory of groups that represent

Control, Virtual Instrumentation and Digital Systems.
Research in Computing Science 24, 2006, pp. 1-10

the forms and shapes of the image objects. The fundamental morphologic operation is nonlinear and consists of the image compare to a simple geometric pattern named structure element. The mayor advantages of the morphologic image processing are: low computational complexity caused by logic operators used; the processed image preserves its geometric features; two basic primitives, namely dilation and erosion, in combination to themselves form all operators.

In this paper, we present a specific architecture based on the data flow that implements a morphologic image processing offering the following advantages: it is not necessary to acquire the all image to start processing; the introduced time delay is determinated by the buffer length (2 rows and three pixels for each operation); once the buffer is filled out the delay is zero; for its simplicity, the architecture can be implemented in FPGA.

2 Edge Detecting Using Basic Morphologic Operators

The operation fundamental of basic morphologic operations, dilation and erosion , one can found in the literature, for example, in [2]. The binary dilation can be calculated as a sum of Minkowsky $A \oplus B = \{x = a + b \in X | a \in A \wedge b \in B\}$, as a union of translation and from a geometric algorithm that can be formulated as follows. Let $A \subset X$ *and* $B \subset X$ are two sets, thus from Theorem 3.8 in [2] it follows

$$A \oplus B = \left\{ x \middle| \left(B^{-}\right)_x \cap A \neq \phi \right\}, \tag{1}$$

where B^{-} denotes reflected set B, $B^{-} = \{-b | b \in B\}$, $\left(B^{-}\right)_x$ denotes translated set B^{-}, $(B)_x = \{y = b + x | b \in B\}$, ϕ is an empty set. The geometric algorithm to dilate the set $A \subseteq X$ by a structural element $B \subseteq X$ can be described step bye step as follows [2]:

1. Choose a point $x \in X$
2. Calculate B^{-}, the reflection of the structural element B
3. Find $\left(B^{-}\right)_x$, the translation of B^{-} by x
4. Realize the intersection $\left(B^{-}\right)_x \cap A$
5. If the set $\left(B^{-}\right)_x \cap A \neq \phi$, thus $x \in A \oplus B$
6. Steps 1 to 5 are repeated for all the space X points.

The binary erosion can be found as a rest of Minkowsky [2]: $A \ominus B = \{x \in X | x + b \in A, \forall b \in B\}$. Besides, the erosion can be defined as a geometric algorithm, in the same way as it was defined for the dilatation [2]. Theorem 4.2 [2] states that

$$A \ominus B = \left\{ x \middle| (B)_x \subseteq A \right\}. \tag{2}$$

This result provide a geometric method to find the erosion of a set $A \subseteq X$ by a structural element $B \subseteq X$. The algorithm consists of the following steps [2]:

1. Choose a point $x \in X$.
2. Find $(B)_x$, the translation of B by x.
3. If $(B)_x$ is a subset of A, thus $x \in A \Theta B$.
4. Repeat steps 1-3 for all points of the space X. Finally, the set $A \Theta B$ will be found.

The opening operator [2] of the set A by the structural element B is denoted as $A \circ B$ and is defined as $A \circ B = (A \Theta B) \oplus B$. The effects of the opening on an image can be resumed in four aspects: it eliminates islands of the size smaller tan of the structural element; it eliminates picks o valley thinner than the structural element; it breaks isthmuses whose width is smaller than the structural element diameter; it smooth the image convex edges.

The closing operator [2] of the set A by the structural element B is denoted as $A \bullet B$ and is defined as $A \bullet B = (A \oplus B) \Theta B$. The effects of the closing are: it fills lakes or holes of the size lower than of the structural element; it fills cracks or gulfs thinner than the structural element; it fuses straits whose width is smaller of the diameter of the structural element; it planes the concave image edges, stuffing breaks.

Similarly to the operators of the erosion and dilatation, it is possible to design the closing operator from the opening operator and vice versa using the property of the duality between the erosion and dilation operators. Additionally, these operators share the property of *idempotence*, which means that once the opening or closing is performed, the repetition of such a process does not have any effect on the working set. In other words, the repetition of the opening or closing with the same structural element does not alter the result.

The Theorem 3.5 form [2] states that given the A, B subsets of X, the inner border is equal to

$$\Gamma(A) = A - (A \Theta B). \tag{3}$$

Fig 1 illustrates the mechanism of edge extraction: (a) shows the original image, (b) shows the structural element, (c) presents the erosion of A by B, (d) shows the found edge $\Gamma(A) = A - (A \Theta B)$.

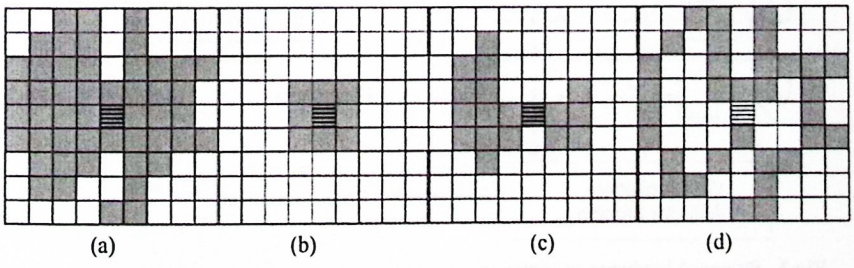


Fig 1. Edge extraction: (a) original image, (b) structural element, (c) erosion of A by B, (d) edges of A

Considering all values as one for non-zero pixels and zero for the background, the edge extraction operation can be implemented by means of logic operators, this is, for $-(A \Theta B)$ it can be used AND and an inverter to complement the operation, for $\Gamma(A) = A - (A \Theta B)$ it can be used XOR between A and the inverter output.

3 Proposed Hardware Architecture and Implementation

The before mentioned basic morphologic operations (erosion and dilatation) can be realized locally in the vicinity of the current pixel. For the vicinity of one pixel nine pixels (3 x 3) are required in three rows and three columns to define the desired structural element. In this manner the processing can start without the necessity to have the entire image. In the video signal the data are transmitted in a serial form, say the image arrives pixel by pixel. The image is formed from left to right and from top to bottom. This situation permits us to see that from three received image rows it is possible to start image processing. This way the image can be processed as a signal.

Fig. 2 shows the proponed architecture for the operations of erosion and dilation, with a structural element size of 3 x 3 pixels. This architecture consists of two shift registers having the longitude in pixels equal to one video row, and the third raw may have the longitude of three first pixels that it is sufficient to start processing.

With the first three pixels of three rows the processing window is formed to obtain the first processed pixel. To perform the dilatation OR is applied to the current pixel and its vicinity; in case of the erosion the operator AND is applied. The output of the gates thus corresponds to the processed signal. To process next pixel, a shift in the registers is performed, then the morphologic operators are applied. This way the processing is repeated until entire video frame image is passed through the registers.

As it can be evaluated, the processing time is equal to the time of the logic gate trough pass plus the delay corresponding to the duration of dos video rows and three pixels of the third one. Once the registers are filled out the processing time is zero.

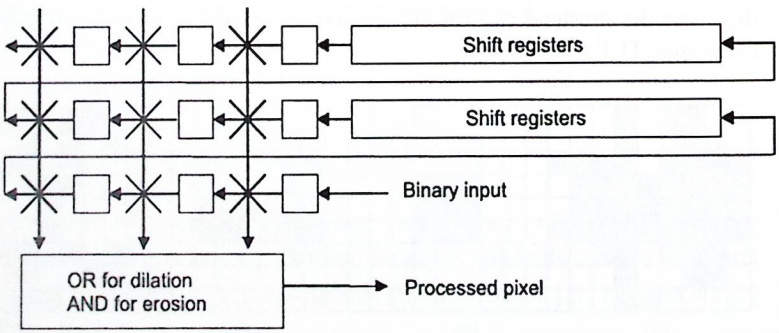


Fig 2. Proposed hardware architecture

At the first sight, the presented idea is simple and functional, but it is not very obvious the way to implement the proposed architecture in commercial electronic devices, given that the price of the design using a specific circuit may be very high for this purpose. The required device for the proposed architecture is the shift register that is common used and widely represented at market. The required shift register quantity depends on the length of the image row. For high quality video the image size often is 640x480 thus to implement the proposed architecture one requires 2 x 640 +3=1283 flip-flops.

FPGA are flexible electronic devices with high density of gates and fast propagation. Revising the possibilities of their inner architecture one can see that using only the flip-flops imply to waste the mayor percentage of the device resource. This situation causes that the implementation results expensive. Besides, we found the XILINX FPGAs predispose in the architecture of their devices the possibility to implement large shift registers. The XILINX devices such as SPARTAN II, SPARTAN III and VIRTEX offer the possibility to implement the shift registers without supersaturate the inner flip-flops of the device. For this purpose the look-up tables (LUT) can be used [3], [4]. In this manner all resources of the FPGA can be employed.

XILINX's design software include various libraries for circuits, among them are the primitives for the shift registers of 16 bits implemented by LUT. Using the XILINX primitive for 16 bit shift register, the frame for the shift register of 640 bits can be designed by means of 40 cascades of 16 bit shift registers. With this frame one row of the image pixels can be stored. Once the 640 bit register line is designed, it can be invoked the number of times it is necessary. Having the line of the shift register of 640 bits and discrete flip-flops, the proposed architecture for the implementation of the erosion and dilation primitives is conformed.

The primitives for the erosion and dilation have as a formative element the shift register of 16 bits implemented by LUT. To form the frame, various registers of this type are connected in cascade (see Fig. 3).

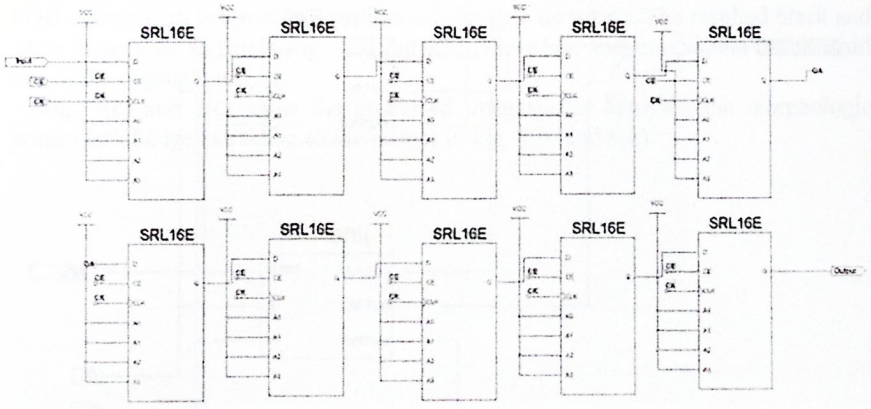


Fig. 3. Block of ten shift registers of 16 bits

Fig. 4 shows the block of 4 bits that is used at the end of the line to implement the operations OR and AND.

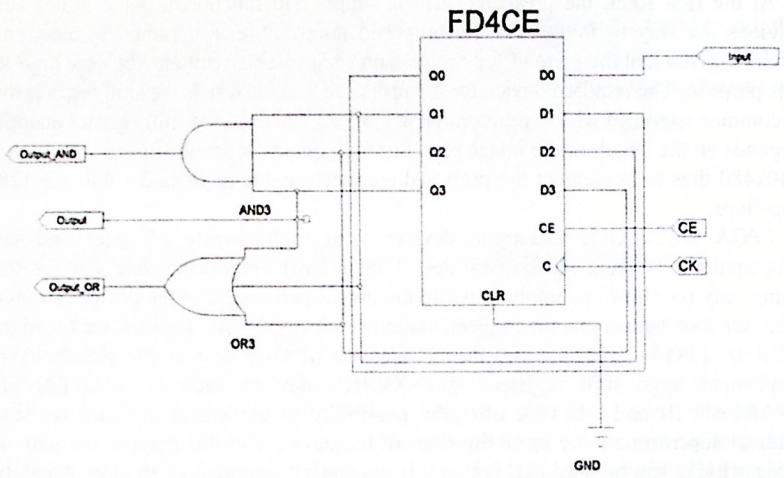


Fig. 4. Block at frame end

The basic primitive of erosion and dilation is shown in Fig. 5. This primitive faithfully implements the proposed architecture. It is formed by two frames of 640 bits and the logic gates OR (erosion) and AND (dilatation).

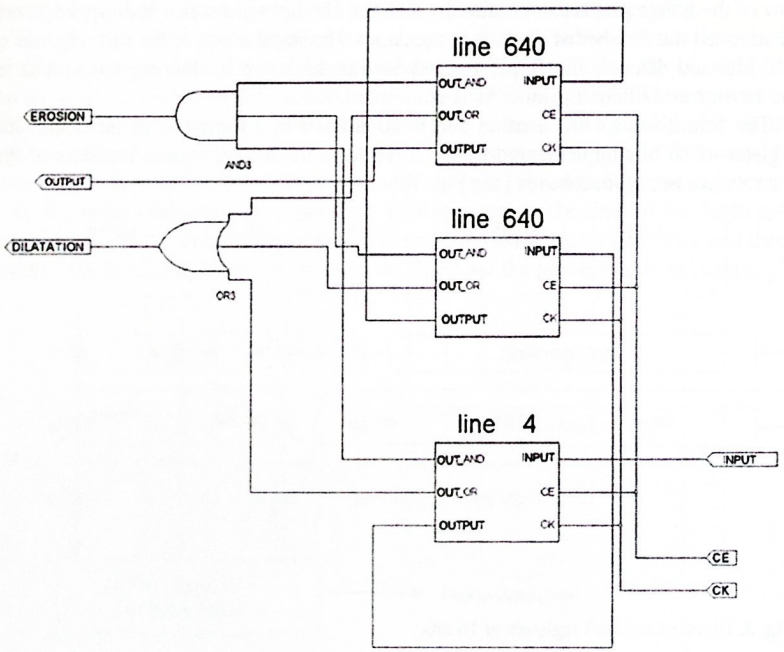


Fig. 5. Primitive for erosion and dilation

The primitive for edge extraction is shown in Fig. 6. It integrates the erosion-dilation primitive, one delay, one inverter and one XOR to implement the equation (3).

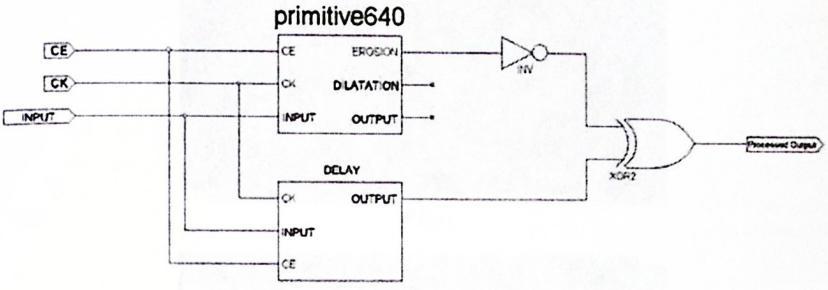
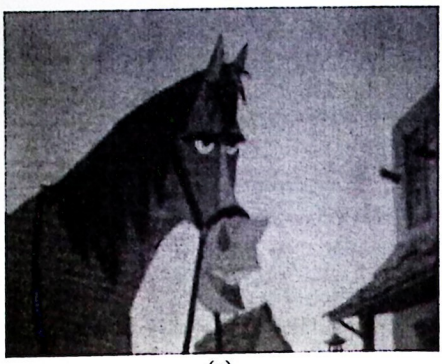


Fig. 6. Edge primitive.

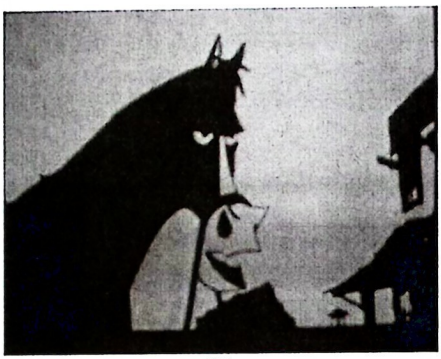
4 Example of edge detection application

In this section, we present the results related to image edge extraction as an example of the morphologic processing. Fig. 7 and 8 show the original and processed images that are the photos captured from PC monitors. The original color images (see Fig. 7(a) and Fig. 8(a)) of 640x480 pixels are the frames of video stream of 60fps. First, the color images are converted to their black/white representation. For this purpose the video data are got form DVI connector of a VGA monitor and are converted to the RGB format using a hardware decoder, for example, SIL161B, SIL163B from Silicon Image or TFP101, TFP401 from Texas Instruments can be used [5]-[7]. Next, the RGB components are summed and passed trough a threshold. The resulted black and white images are shown in Fig. 7(b) and 8(b). These b/w images maintain the size and rate of the original ones.

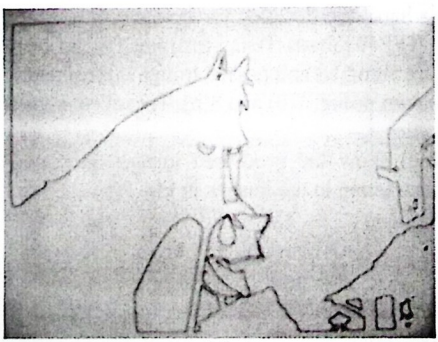
Fig. 7(c) and 8(c) show the processed images alter applying the morphologic primitives of edge extraction to the images in Fig. 7(b) and 8(b).



(a)



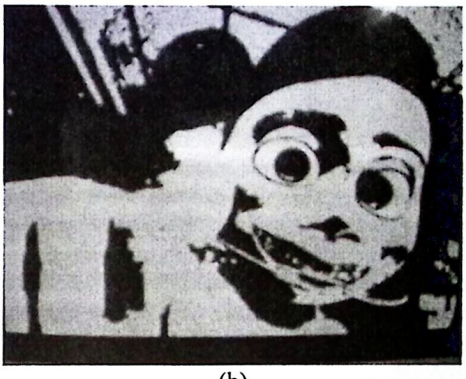
(b)



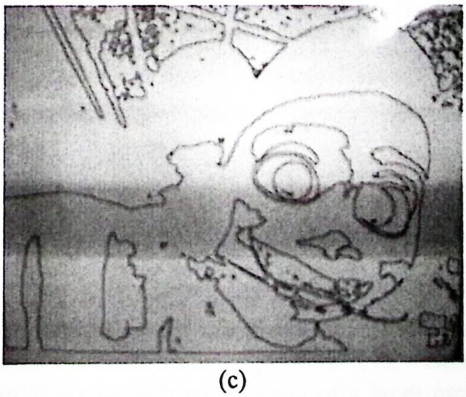
(c)

Fig. 7. (a) color image, (b) black and white image, (c) edge image.

(a)



(b)



(c)

Fig. 8. (a) color image, (b) black and white image, (c) edge image.

5 Conclusions

We have presented the hardware architecture that permits to process the morphologic primitives with a minimal time delay. The introduced delay directly related to the structural element size and for the size of 3 x 3 pixels the delay corresponds to the duration of three rows of image. The employed FPGA has the density of 100,000 gates and system frequency of 200Mhz. The use of FPGA look-up tables as shift registers benefits in the full usage of FPGA resources and permits to implement in average 40 morphologic primitives. The designed FPGA based system processes satisfactory the images with the resolution of 640 x 480 pixels and rate of 60 frames per second. Using the developed frames for the morphologic primitives and software scheme editor the designed system is flexible and can be used in various application of morphologic image processing.

References

1. Moler, van der, M.W., Kyo, S., "Documentation for the IMAP-VISION image processing card and the 1DC language", NEC incubation Center, 1997
2. Díaz de León Santiago J., Yáñez Márquez C. (2003), *Introducción a la morfología matemática de conjuntos*, IPN, UNAM, FCE, México, D.F.
3. Spartan-II 2.5V FPGA Family: Complete Data Sheet, Xilinx
4. Using Look-Up Tables as Shift Registers (SRL16) in Spartan-3 Generation FPGAs, Xilinx
5. TFP101, TFP101A Design Notes, Texas Instruments.
6. Sil161B PanelLink Receiver Datasheet, 2001, Silicon Image, Inc.
7. TFP401, TFP401A, TI PanelBus DIGITAL RECEIVER, Texas Instruments.