



## COLORING OF GRAY-SCALE IMAGE USING FPGA

Ammar A. Hassan

University of Baghdad \ College of Engineering \ Department of Computer Engineering

### ABSTRACT

The image processing is one from the most powerful fields in the modern DSP techniques; also it has wide range of applications this day such as image compression, filtering and coloring. However, these processes required to a huge data processing so it has a problem under real time or movie.

The huge data processing under real time requires spatial processing tools such as super parallel processing computers or spatial hardware systems. This paper introduces a mechanism of coloring gray scale image algorithm through dedicated hardware devices.

The FPGA devices are used as a more suitable platform for image processing applications, special methods of parallelism and pipelining technique can be reconfigured and synthesized on FPGA categories. Xilinx series are selected as a platform of coloring algorithm by transferring the color property between pair image, source (colored) and target (gray) images.

The algorithm colorizes each gray scaled pixel by matching chromatic value of it with each pixel of colored image and synthesis it on the Xilinx FPGA devices using VHDL synthesizer tool. Many computational and process manners of this scheme are presented of 8-bit precision for each pixel of pair image.

Finally, testing and performance of this technique obtained on ISE 4.1i software implementation and comparing results with other simulator results.

### المستخلص

أن معالجة الصور هي واحدة من أغلب المجالات الرائعة في تقنيات معالجة الإشارة الرقمية الحديثة والتي لها مدى واسع من التطبيقات في الوقت الحاضر، مثلاً على ذلك ضغط الصور؛ الترشيح والتلوين. مع ذلك هذه العمليات تتطلب معالجة بيانات واسعة جداً لذلك تعتبر مشكلة ضمن تطبيقات الزمن الحقيقي والأفلام السينمائية

أن معالجة البيانات الواسعة ضمن تطبيقات الزمن الحقيقي تتطلب أدوات معالجة خاصة مثل حاسبات معالجة متوازية عالي أو منظومات كيان مادي خاصة. هذا البحث يقدم خوارزمية لتلوين الصور الرمادية من خلال أجهزة كيان مادي مخصصة.

أن أجهزة الكيان المادي المسماة حيز البوابات المرتب بهيئة صفوف قابلة للبرمجة (FPGAs) والتي تستخدم أرضية جداً مناسبة لتطبيقات معالجة الصور بطرق خاصة من التقنيات المتوازية ومجموعة التقنيات شبه الموازية التي يمكن

إعادة تشكيلها وتراكيبها على أصناف (FPGAs). أن سلسلة Xilinx التي تم اختيارها كأرضية لخوارزمية التلوين بواسطة نقل اللون المناسب بين الصورتين [ المصدر (ملونة) والهدف (الرمادية)]. الخوارزمية المقترحة تقوم بتلوين كل نقطة رمادية بمطابقة قيمة اللون الخاص بها مع كل نقطة بالصورة الملونة وتراكيبها على أجهزة ال (FPGAs) باستخدام أدوات تراكيب بلغة الكيان المادي المسماة بـ (VHDL). عدد من الحسابات وطرق المعالجة لهذه الهيكلية يتم طرحها بدقة "8" بت لكل نقطة لكلا الصورتين. أخيراً اختبار الأداء لهذه التقنية تم الحصول عليها من خلال برنامج البناء والتنفيذ المسمى بـ (ISE 4.1i) ومقارنة النتائج مع نتائج محاكاة أخرى.

**KEYWORDS:** coloring, Gray-scale, DSP, VHDL, Xilinx, FPGA.

## INTRODUCTION

Any form of digital image based on main elements called (pixels) and pixels characteristic lead to construct the characteristic of image. The acceptable appearance of image exceeds with respect to pixels characteristic are difference from image to other according to features. Image features selection and representation are largest problem in research of machine learning and vision science.

The color images have two main features: Luminance (brightness) and chromatic (color) channels which are represented in the RGB color space system (Red, Green and Blue). By assignment certain value of it leads to have pixel characteristics.

In gray scale image each pixel carry up three channels (RGB) equally values which are produced one feature (luminance) of image. To produce this feature there are many colors adding together in coloring image. Since, color images and gray scale image have important property which is called "*luminance*".

The mechanism of color transfer technique (colorization) has been studies in the movie industry since 1975's. Various analogue techniques have been used to accomplish this challenging [1].

The fundamental process of transferring color from one image to another is introduced by Welsh et. al. [2]. they attempted to provide a method to minimize the amount of human labor required for this task by matching luminance and texture information among images (color image (Source) and gray scale image (Target)). While, Karthilceyani [3] used Welsh idea to enhance his procedure by allowing the user to match areas of the two images using rectangular swatches.

Other works such as YaoLi [1] introduce the fast colorization using edge and gradient constrains to reduce the color confusion near the boundary and applying it on movie. Also, Bara'a A. Attea [4] applied the technique of evolutionary algorithms (EAs) adding to previous ideas, probabilistic search algorithms based on the model of natural evolution, for colorization problem.

This paper interested in digital colorization and related work of digital image processing on Field Programmable Gate Array (FPGAs). On FPGAs, there are several researches of image processing which can be classified according to hardware topologies and image processing algorithm implemented. As in Dagg Ven. [5] that implemented and evaluated of image processing algorithm on reconfigurable architecture using C-based HDLs. This algorithm applied on Xilinx Virtex-E FPGA platform to speed up of processing approximately '15' times faster than the software implementation.

Finally, Muthukumar [6] used image processing algorithms (such as edge detection) on reconfigurable architecture using handel-C on Xilinx FPGA board.

## COLORIZATION ALGORITHMS

This section discusses main topologies of colorization algorithms such as color space conversion and pixel selection. Furthermore, it finds a suitable solution to accomplish this task on Virtex FPGA board with minimum cost and high performance.

### Color Space Conversions

To extract specific properties from an image (such as Luminance and colorimetric information) this leading to understand color space and color space conversions.

A digital color image is a digital image that includes color information for each pixel. The RGB color space system is commonly used in computer displays, but it does not have enough data to match between pair of image [7]. It is necessary to change RGB space to another space system which has more benefit of matching process and this called color space conversion.

This paper applied YIQ space model because it used in commercial color TV broadcasting by mapping function, it can convert RGB to YIQ color space as defined in [7,8].

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

In YIQ color space; the Y coordinate represents the luminance Y while I and Q coordinates represent the chrominance components I and Q respectively.

YIQ space is converted to RGB space by inverse matrix transformation [7].

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.620 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.108 & 1.705 \end{bmatrix} * \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad (2)$$

### Pixel Selection

The use of local memory to store pixels already loaded from the picture memory provides saving in the number of memory accesses to the picture memory [9]. Therefore; search pixels process must be fetched from two different data sources.

The algorithm of color transferring form source image (color image) to target image (gray scale image) is presented by Welsh et al [2]. This mechanism can be achieved by following steps: First; converting source and target images from RGB color space in to the YIQ space. This color space has been chosen because it promptly provides the luminance value (Y channel) which is a crucial datum for our procedure [10]. According to this, compression process based on comparison of each pixel of target image with all pixels of source image upon luminance value and selects the best value (closer) from source image. Then, color values (represent in I & Q channels) of the selected source pixel are transferred to the gray scale pixel with remains luminance (Y of target) value. This process will continuous for all target pixels until forming new image similar to gray scale image with color information. Finally, retain a new image into RGB space to discover the new color appearance of gray scale image.

This algorithm required a very high complexity and a very large of multiplication numbers or in result it will required to high speed or to parallel processing for real time. According this scheme, Virtex FPGAs series are typically suitable to use with intensive

computations; especially complex operations because of it provides fast hardware resources comparing with software resources.

## HARDWARE

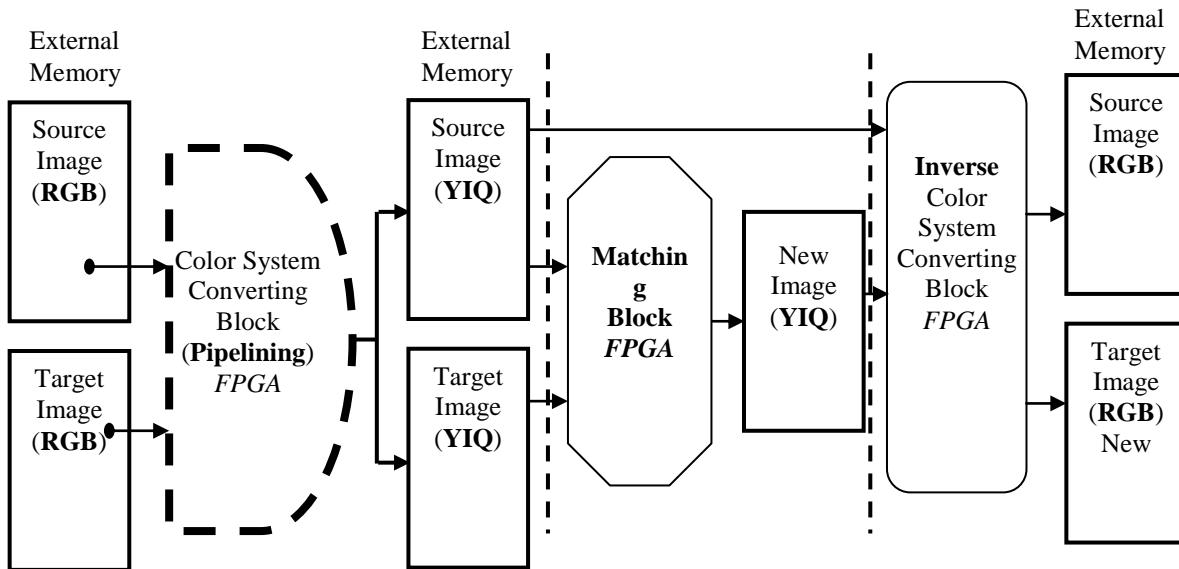
This section will discuss the principles of hardware resources, architecture, the implementation and the cost calculation.

### Hardware Resources

The selection is focused on the selected mechanism of the hardware devices as more suitable in digital image processing field and properly applied on color transferring task. Virtex FPGA is the more suitable for this task because it is based on basic element called logic cell (LC) which can be programmed and reprogrammed for any logic function (as a function generator for any operation) has four inputs and one output.

### Architecture

Block diagram of color transfer architecture (coloring of gray scale image) is constructed by most efficient method and suitable configuration on FPGA platform as shown in figure (1). Main stages of architecture are:-



**Figure 1: coloring gray scale image Architecture**

- **External Memory:** The function of this block is to store source and target images points (pixels) during processing (in "RGB" or "YIQ"). The same unit can be used in a stage because the process is sequenced; each stage is distinct from other stage.

- **Color System Converting Block:** The function of this block is to convert each point of images (source and target) from RGB into YIQ system. This block constructs on Virtex FPGA platform in pipelining technique (configuration is per each column).

- **Matching Block:** The function of this block is to match each point of target image with source points according to luminance (Y dimension) of target image with luminance (Y dimension) of source image and selects chromatic information (I & Q dimensions) from source point to target point. This block constructed on the Virtex FPGA platform.

- **Inverse Color System Converting Block:** The function of this block is to back convert each point of images (source and new target) from YIQ into RGB system. This block not needed to

construct on Virtex FPGA platform because it can be used the same block in the previous stage with new coefficient values as shown in figure (1).

### FPGA Implementation

FPGAs have traditionally been configured by hardware engineers using Hardware Design Language (HDL) [6]. Adding to this, several attempts have been introduced to use other resources of languages and simulation software to implement algorithms on FPGAs.

This section describes the methodology of color transfer algorithm on Xilinx FPGA device, in other words; number of hardware resources and arithmetic operations of algorithm implementation are discussed.

As an example, a pair of image (gray and color images) with size of  $(256 \times 256)$  pixels (8-bit / pixel) is demonstrated in details and compared with other sizes.

The same component of color space conversion will be used in the 1<sup>st</sup> and 3<sup>rd</sup> stages, then in 1<sup>st</sup> stage each point of image (source and target) converting from RGB to YIQ system as shown in figure (2)

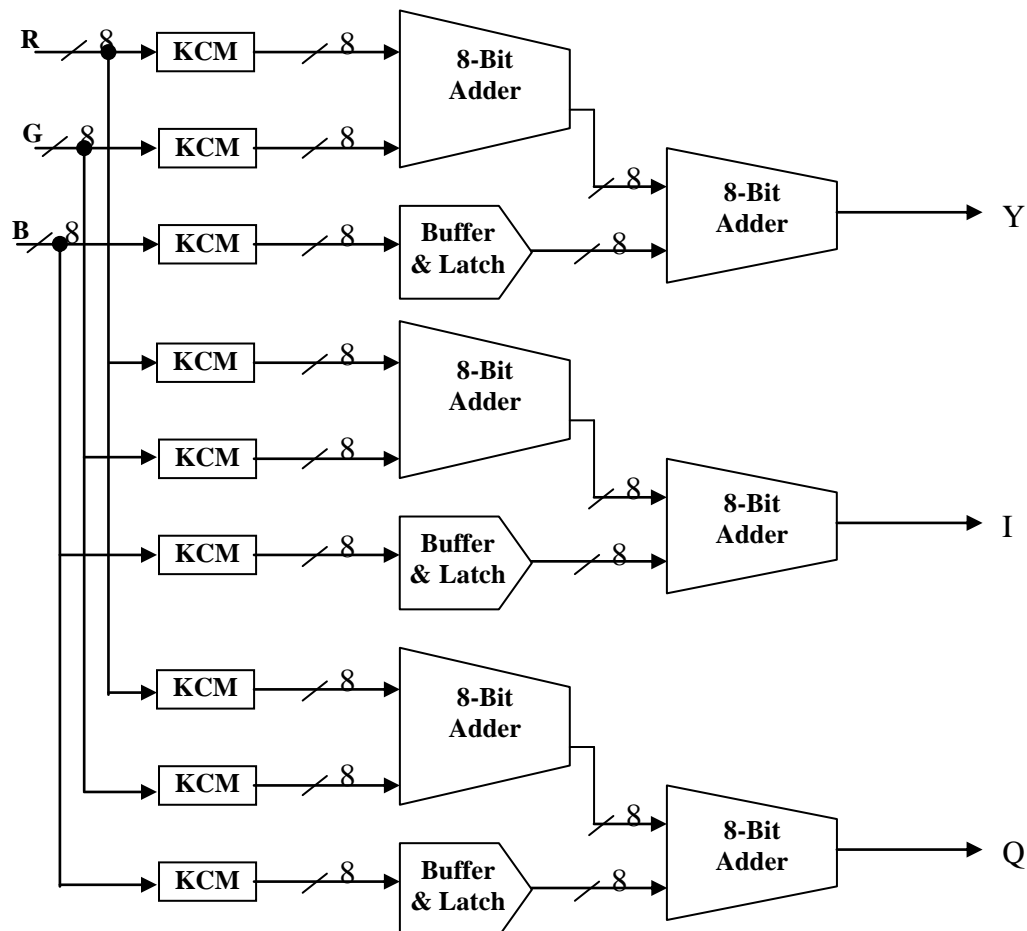


Figure (2): Block Diagram of each point in RGB-To-YIQ Converting on FPGA

### Components Cost

The cost of the system and its parts measured on Virtex FPGA platform arithmetic operations are based on basic logic storage called logic cell (LC) formed as a Look-Up-Table (LUT) of four inputs one output.

### Color System Converting Block

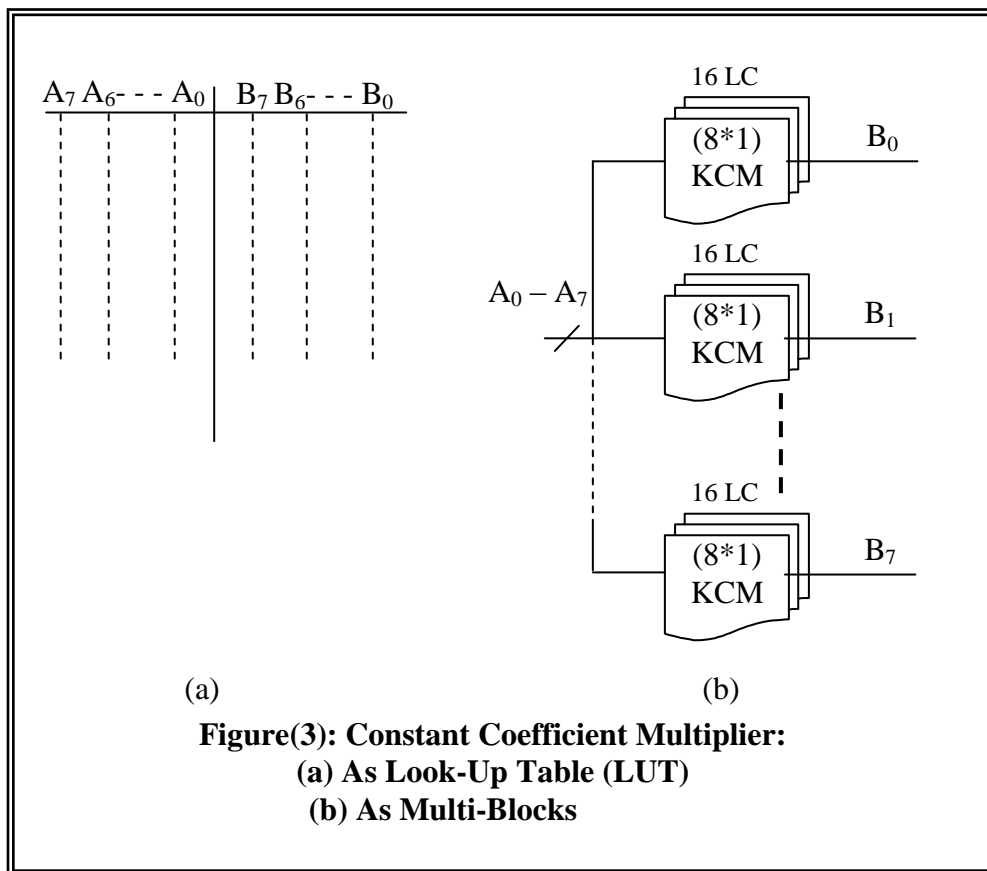
To implemented color system conversion block diagram on Virtex series. For (8 x 1) Constant Coefficient Multiplier (KCM) is required to "16" LCs, while (8 x 8) KCM needed to "128" LCs as shown in figure (3), while adder block has been implemented as adder technique in [11]. This scheme is suitable in both of size and propagation delay. For 8-bit adder, it required to "16" LCs at mostly. The total number of LCs for converting RGB-to-YIQ space system on Virtex FPGA is:

$$P_{oint_{YIQ}} = \{ [N * \alpha] + [M * \beta] + \delta \} * (3)_{YIQ} \tag{3}$$

Where,

N = Number of KCMs;  $\alpha$  = cost of each KCM in Virtex FPGA

M = Number of Adder;  $\beta$  = cost of each Adder in Virtex FPGA, and  $\delta$  = cost of 8-bit Buffer in Virtex FPGA.



**Figure(3): Constant Coefficient Multiplier:**  
**(a) As Look-Up Table (LUT)**  
**(b) As Multi-Blocks**

For (8 x 8) KCM, 8-bit Adder and 8-bit buffer the cost of each point for one channel in (YIQ) is about "425" LCs for maximum, for three channels is "1275" LCs. Using pipeline technique cost of one column is:

$$C_C = N_P * C_{P_{oint_{YIQ}}} \tag{4}$$

Where, C<sub>C</sub> is cost of one column; N<sub>P</sub> is Number of points per each column and C<sub>P<sub>oint<sub>YIQ</sub></sub> is Cost for one point (for 3-channels) which it's "1275" LCs. According to this representation, C<sub>p</sub> is equally to "326400" LCs; at mostly for each column contains "256" points.</sub>

Other columns are not required to compute cost but only propagation delay because they have been passed sequentially for both source and target images on the same component on platform.

**Matching Block**

The matching process based on subtraction and comparing operations (all this process based on Y-channel value of each point for source and target images) as shown in figure (4). This can be accomplished by subtracted Y-channel value of target point ( $Y_t$ ) from Y-channel of all source points in one column ( $Y_s$ ).

In other words, comparing subtracted values of each subtractor to select smallest subtraction value by using comparator in a tree of comparators. This mechanism provides the closer source point to target point (with respect to luminance matching). According to this, other properties such as chromatic values (I & Q channels) of the selected point of sources image resultant from matching process are transferred to the target point of gray-scale image to form final image (i.e. replaced I & Q of target point by I & Q of the selected (winner) point resultant from tree comparators of source image). In pipeline technique, searching process continues for all points in source image with each point of target image.

Finally, using same component of color system converting block to retain new image which resultant from gray image to the color vision image by reversed color space transferring from YIQ-to-RGB dimensions and forming final image, this can be achieved without needed to present cost of computation on Virtex platform; it's one of most benefits of FPGA properties.

To implement matching architecture on Virtex series, it is required to find a mechanism to represent it as digital arithmetic operations technique to accomplish in simplification form, since a suitable architecture of 8-bit word length shown in figure (4).

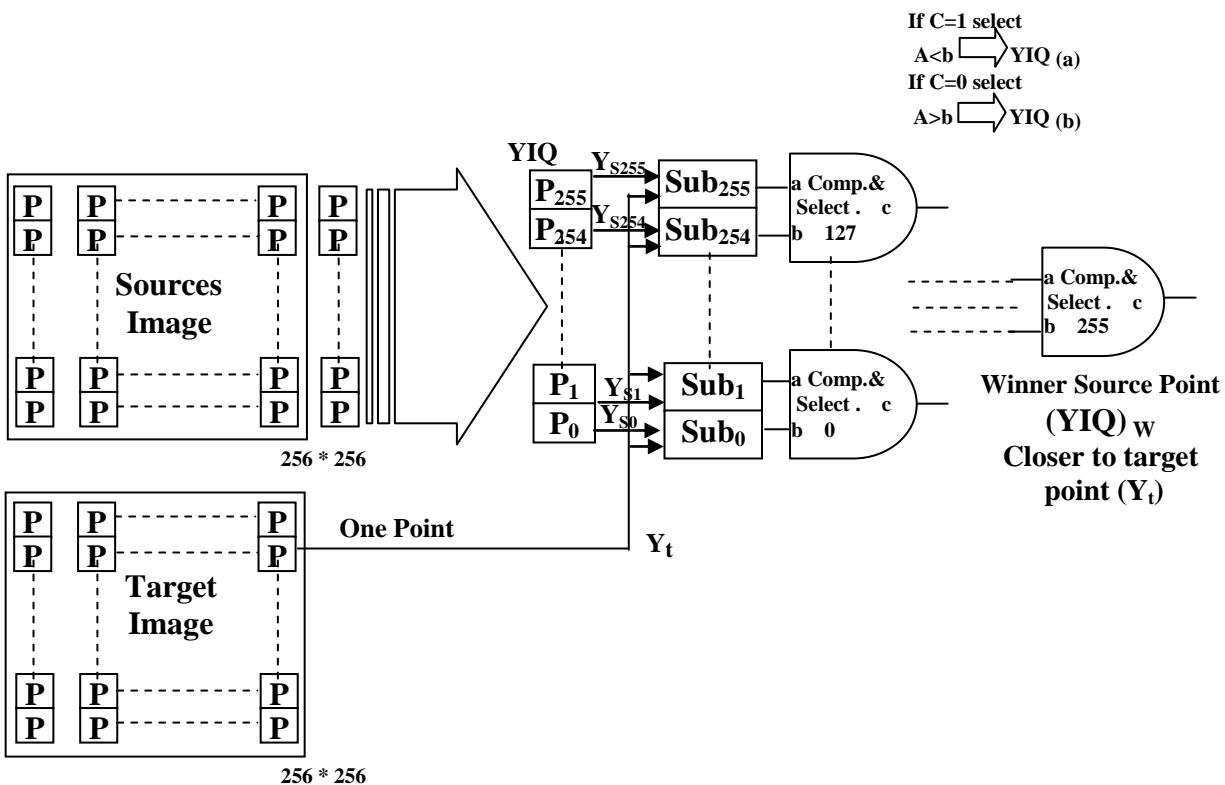


Figure (4): Matching Block Diagram

The main function of matching block is to find the closer point of source image to target point and selected the chromatic information from it. This scheme applied by obtaining minimum error value of point in target image with each points of source image. From figure (4) each point value (Y-channel) of target subtracts from each points of source image which arranged in one column using subtractor component (sub.) and finds minimum subtraction

value (winner point) from these subtractors by passing in a tree of comparator & selector component (comp & select).

Cost of matching block based on cost of (subtractors) and (comparators & selectors) components. Cost of 8-bit subtractor is similar to cost of 8-bit adder in previous section as demonstrated in [11], while; 8-bit of comparator & selector component based on technical idea shown in figure (5).

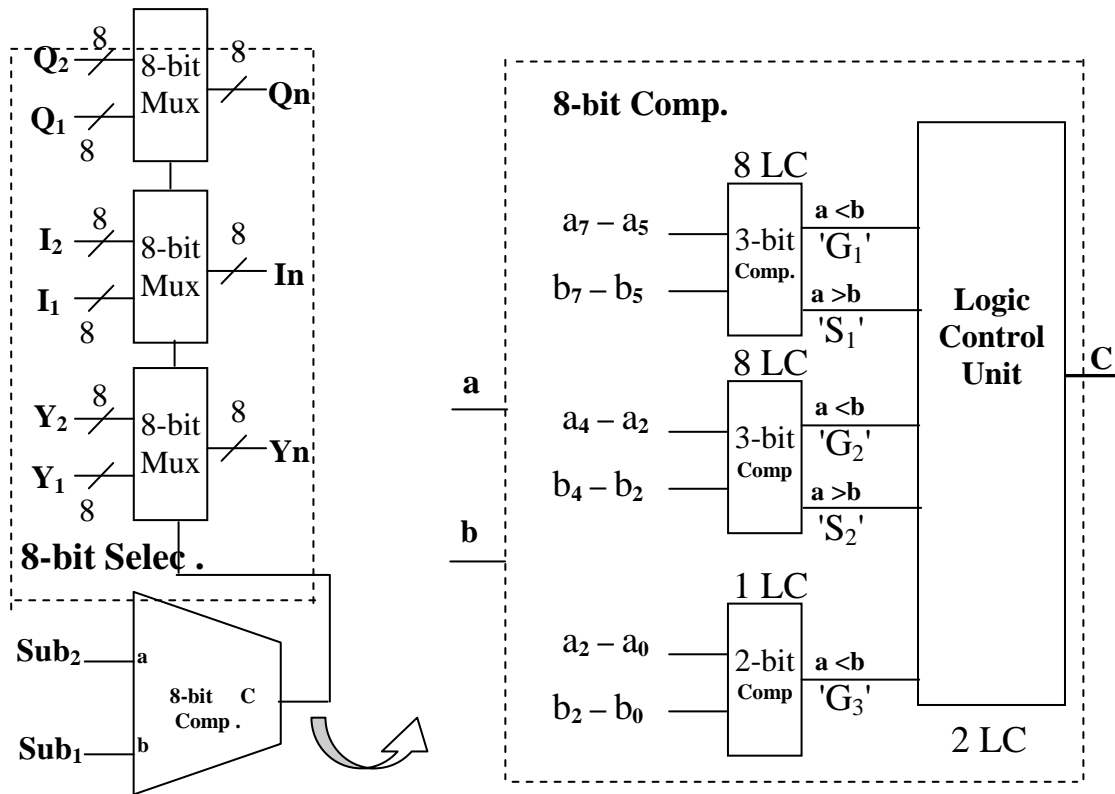


Figure (5): Comp. & Select. Block Diagram

Where,  $C = G_1 + \bar{G}_1 \bar{S}_1 G_2 + \bar{G}_1 \bar{S}_1 \bar{G}_2 \bar{S}_2 G_3 \approx 19 \text{ LCs}$ . While 8-bit selector represented as a multiplexer (Mux.) which occupied "8" LCs for each Mux in virtex platform, then "24" LCs for three selector (8-bit Mux) of 3-channels. Since, cost of 8-bit comp. & select. component is equal to approximately "43" LCs and this lead to cost of matching block is:

$$\text{Cost of Matching Block} = [\text{No. of subtractors} * \text{cost of subtractor}] + [\text{No. of (Comp. \& Select.) components} * \text{cost of component}] \quad (5)$$

Total cost of (8-bit) matching block is approximately equal to "15061" LCs. Finally, system cost produced by combining cost of color system converting block with matching block which approximated equally to "341461" LCs.

**Propagation Delay (PD)**

Propagation delay is the time consuming to achieve a specific task. To compute propagation delay of image colorization technique, it is divided into three stages. First stage is to convert both color and gray scale images from "RGB" to "YIQ" space system using color converting block. Each point in this stage based in idea on KCM and buffer as main



components which are required to one clock pulse, while 8-bit adder needed to maximum four clock pulses (if dividing to 2-bit adder) [11]. By using Xilinx FPGAs, maximum clock pulse 330 MHz [data sheet], then maximum propagation delay  $(PD)_{Max}$ . required to convert each point from "RGB" to "YIQ" space system is about vibrating between (5 - 10) ns without needed to storage state and applied pipeline technique. Then, propagation delay of an image is depended on the number of columns in the image. As a result, time delay one image to convert it from "RGB" to "YIQ" space systems is:

$$(PD)_{Image} = (PD)_{Max} * \text{Number of Columns in image} \quad (6)$$

Since, our example there are "256" columns for each of source and target images. Then,  $(PD)_{Image} \approx "2.56" \mu s$  of each image and for pair image  $\approx "5" \mu s$ .

The same delay required in the third stage when used inverse converting from "YIQ" to "RGB" space because of it used same components in stage one with other KCMs value according to inverse matrix transformation.

Time delay interval of the second stage is based on maximum Propagation Delay  $(PD)_{Max}$ . of matching block components (Subtractor component and comparator & selector component). Since, PD of (8-bit) subtractor is also reach to about four clock pulses (because of it constructed by the same technique of (8-bit) adder in the previous stage), While (8-bit) comparator & selector component is required maximum three clock pulses in virtex FPGAs platform (because of it divided to three parts as shown in figure (5). Then,  $(PD)_{Max}$ . is about to (10 ns) as demonstrated in later . It become clear, time delay of matching block is:

$$(PD)_{Matching} = (PD)_{Max} * [1 + \gamma] * N_C * N_T \quad (7)$$

Where:  $(PD)_{Max}$ . is maximum Propagation Delay;

$[1 + \gamma]$  is number of subtractor block & Comp. & Select. Blocks in propagation tree;

$N_C$  is number of columns in source image;  $N_T$  is number of points in target image

Since, our example there are "256 \*256" points each of source and target images. Then,  $(PD)_{Matching}$  equal to "1497.6"  $\mu s$  or  $\approx "1.5" s$  as propagation delay of second stage.

Finally, propagation delay of the system  $(PD)_{System}$  is the total delay of three stages which approximately to "1502.6 "  $\mu s$ .

## RESULTS

The colorization algorithm of Gray image using color transferred architecture is presented and synthesized simulation using software implementation of ModelSim Xilinx edition and Integrated Software Environment ISE 4.1i Xilinx. All results are demonstrated in reports (such as Routing, Mapping, Number of LCs, propagation Delay (PD) and warning notes and compared with software results introduced from Pentium machine using Visual Basic (Visual Studio 7.0) program by simply way without any optimization. table (1) includes some of importance results that dealing with special hardware resources (time consuming (PD) and number of LCs), where it considered in order to present two major parameters in the implementation of gray-scaled image colorization on FPGAs. Since; VHDL tool is used in software implementation to achieve high performance (minimum area and high speed processing; less Propagation Delay) result rather than schematic tool.

The hardware results of coloring algorithm are compared with it results on simulation tool according to Welsh et al [2] without optimization. In presented solution and by exhaustion from FPGA resources it can speed up performance and achieved agreeable good image outfacing of color transfer algorithm.

**Table 1:** results of Gray scale coloring Images

Colored image	Required Time according to [4] (sec.)	System Cost_ Logic Cells (LCs)	Propagation Delay (PD) <sub>System</sub> on Virtex FPGAs (sec.)
(A)	448	144029	0.284
(B)	235	154973	0.154
(C)	645	216065	0.251
(D)	718	212063	0.286
(E)	715	312893	0.291
(F)	500	189221	0.314
(G)	496	224069	0.186
(H)	312	143489	0.202
(I)	2726	295682	1.203
(J)	668	206468	0.342
(K)	790	208061	0.320

3rd column consider in table (1) demonstrate number of Logic cells of the system according to ISE 4.1i software results. ISE 4.1i offers integration with Synopsys Inc.'s Formality™ and Verplex Systems Inc.'s Conformal™-LEC equivalency checkers, leveraging the same technology that was adopted in past years to check highdensity ASICs. In the equivalence method of testing, design passes can be checked in “blocks of logic” against a previous knowngood version. This can occur at any point in the design cycle, particularly in post-synthesis and place-and-route passes. This strategy offers a rapid checking method that is proving invaluable to high-density design work For Virtex-II designers moving to 1-million gate designs and above [12].

**Source Image**

171 x 108

**Target Image**

171 x 108

**Colored Image**

171 x 108

(A)



135 x 99



108 x 117



108 x 117

(B)

5941



135 x 162



135 x 153



135 x 153

(C)



144 x 159



144 x 153



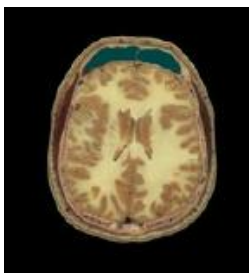
144 x 153

(D)

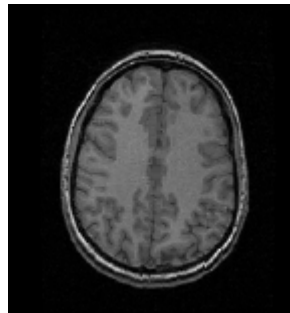
**Source Image**

**Target Image**

**Colored Image**



144 x 156



144 x 156



144 x 156

(E)



126 x 96



192 x 144



192 x 144

(F)  
5942



111 x 168



111 x 168

(G)



111 x 168



144 x 108

Source Image



144 x 108

(H)

Target Image

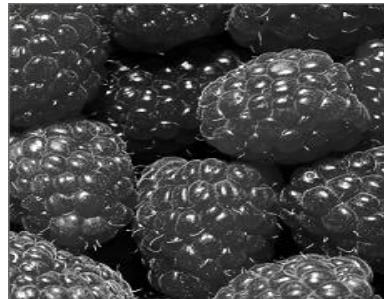


144 x 108

Colored Image



198 x 150



300 x 225

(I)



144 x 108



153 x 129

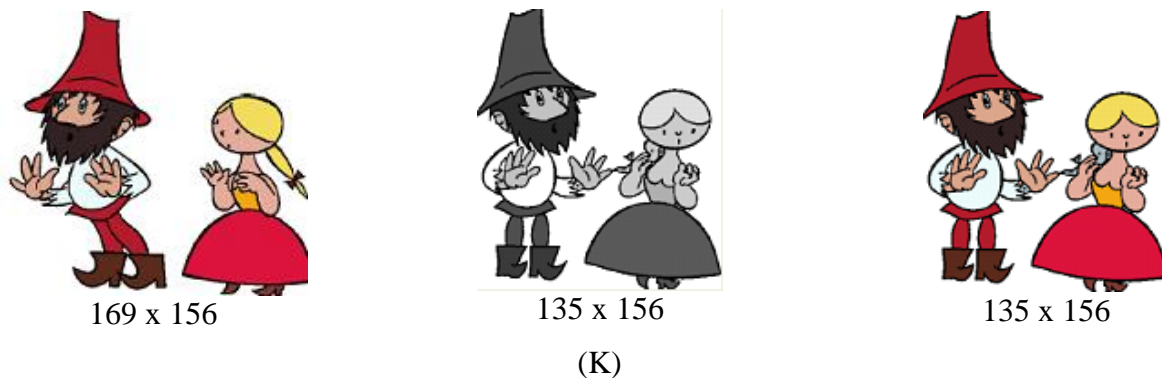


159 x 156

(J)



159 x 156



**Figure 6: Examples of Color images for some of Gray-Scale Images acquired by color transferred approach. First column is the source image, second column is the target image and third column is colored image.**

## CONCLUSIONS

A fast and low hardware components (number of logic cells) were presented in this paper to solve hardware resources problem of color transferring technique using FPGA devices.

Various different ideas have been combined together to produce low hardware cost and high performance scheme by using the flexibility of Virtex FPGA technologies. Pipelining technique applied as a main idea to obtain a suitable number of LCs with acceptable PDs. Table (1) shows the influence of different sizes of source & target images on hardware cost and time consuming in system processing. All images in figure (6) colorized within one second or less and the system cost can be implemented on one chip with all blocks of algorithm process. Where, maximum number of logic cells presents in image (I) of table (1) because the target image size (300 x 225) points is relatively large size for image to be colored although time required to do this is about one second and this is fast than it result in [4].

Finally, this chip can be used in more critical and complex applications such as colorization of images which have lost some data of information, also in wireless image applications and aerial photos according to relationship to chromaticity information (I&Q channels) with the luminance value (Y channel) between source & target images.

## REFERENCES

- Yao Li, Ma Lizhuang & Wu Di, " Fast Colorization Using Edge and Gradient Constrains ", Copyright UNION Agency – Science Press, Plzen, Czech Republic. 2006.
- Welsh T., Ashikhmin M. & Mueller K., " Transferring color to grayscale images " ACM Transactions on Graphics 21, 277-280, 2002.
- Karthikeyani.V., Duraiswamy.K. & Kamalakkannan.P., "Conversion of Gray-scale image to Color Image with and without Texture Synthesis ", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.4, April 2007.
- Bara'a A. Attea & Amina D. Aboud, " Evolutionary Colorization of Grayscale Images", Int. J. Advanced Media and Communication, Vol. 1, No. 3, 2007.
- Daggu Venkateshwar Rao, Shruti Patil, Naveen Anne Babu and V Muthukumar, "Implementation and Evaluation of Image Processing Algorithms on Reconfigurable

- Architecture using C-based Hardware Descriptive Languages", International Journal of Theoretical and Applied Computer Sciences Volume 1 Number 1 pp. 9–34, 2006.
- Muthukurnar K. & Daggi V., "Image Processing Algorithms on Reconfigurable Architecture Using Handel-C", Journal of Engineering and Applied Sciences Volume 1 Number 2 pp. 103-111, 2006.
  - Laylan M. Rashid, "A Genetic Algorithm for Texture Synthesis and transfer", A M.Sc. Thesis submitted to Department of Computer Science, University of Baghdad, Baghdad-Iraq, 2005.
  - GONZALE, R.C., & WINTZ, P., "Digital Image Processing", Addison-Wesley Publishing, Reading, MA, 1987.
  - Tiago M. B. S. Dias, "High-Performance VLSI Motion Estimation Processors: Data Reuse and Sub-Pixel Accuracy", A M.Sc. Dissertation submitted to Department of Electrical & Computer Engineering, Pires, September 2004.
  - G. Di Blasi, and R. D. Reforgiato, "Fast colorization of gray images", In proceedings of Eurographics Italian Chapter, D.M.I., University of Catania, 2003.
  - Dafer R. Z., "Design and Implementation of A general Digital Down Converter using FPGA", A PhD thesis submitted to Department of Electrical Engineering, University of Baghdad, Baghdad-Iraq, 2002
  - "Xilinx ISE Delivers the Speed You need", [www.xilinx.com/ise/xcell](http://www.xilinx.com/ise/xcell), winter 2001.