Applying Dynamic Hair Dyeing Simulation On FPGA to Improve User Experience and Life Quality

Han-Ching Ou[1] and Po-Jung Chang[2]

Department of Electrical Engineering, National Taiwan University

[1] b99901119@ntu.edu.tw

[2] b99901113@ntu.edu.tw

Abstract—**The purpose of this paper is to design a system avoiding user from getting unexpected hair dyeing result on FPGA DE2-115 which can perform real-time dyeing simulation. In the system, user can have 55 default hair types, which have already been set in the devices, commonly used in hair salons. If there is a new hair color type that is not in those 55 defaults, the hair salon can adjust the value of red, blue, and green until the exact color result is obtained and then added to the system. The color information in RGB values will be showed on the LCD screen and the hair salon can easily change them only by a few ticks on the multi-touch board.**

The system design architecture is consisted of three major parts. First, the camera collects the information and transports the data into the FPGA board. Second, the FPGA board processes and then an algorithm is applied to identify the correct region to be dyed. Also, in order to keep the hair texture, another algorithm is applied to imply for color scaling in the detected region. Finally, the multi-touch board displays the processing results and ready to receive user commands from the FPGA board.

The system provides a more efficient and precise way to simulate the dyeing results of user in real-time compared to the common method in dyeing hair by just using imagination based on category.

***Keywords*—Hair dyeing; FPGA; Real-time; Hair detection; Color scaling**

I. Introduction

Hair dyeing, which is an ancient art, has been a popular cosmetic method since human discovered a way to color their hair by making use of various chemical compounds. It can be used for covering gray hair due to aging or sun bleaching, change ones hair to a more fashionable and desirable color or adjust the color to coordinate with ones outfit.

However, different from trying on shoes and clothes, once hair dyeing has been done, you cannot "take off" the hair color or "try on" another one. If you don't like the color on your hair or regret it, all you can do is wait a long period of time and come back for another hair dyeing.

Besides, as research shows, frequently hair dying is a huge damage to ones hair, which will result in dry, rough and fragile hair. In extreme cases, the hair can be so damaged that it breaks off entirely.

Therefore a real time hair simulating system is proposed to solve this problem. You can "try on" as many new hair colors as you want in a "try on room" from any angle or position, just like a mirror that can show you in different hair color type that you may want to dye in next 30 minute.

The system architecture roughly includes three major parts: input ,processing and output. Input part includes camera and touch screen, processing is on the FPGA and output part is touch screen and VGA.

To realize the system, two algorithms are demanded. The first one is to detect "the area needed to be dye"(i.e. hair detection). The second is to dye the hair sophisticatedly meanwhile preserves the hair texture so it will look more natural.

Fig. 1 shows all the electric hardware used to realize this system, and fig. 2 shows compilation result of our implementation. The compiler we used is Quartus II 13.0. About 4% of logic elements on the FPGA are used.

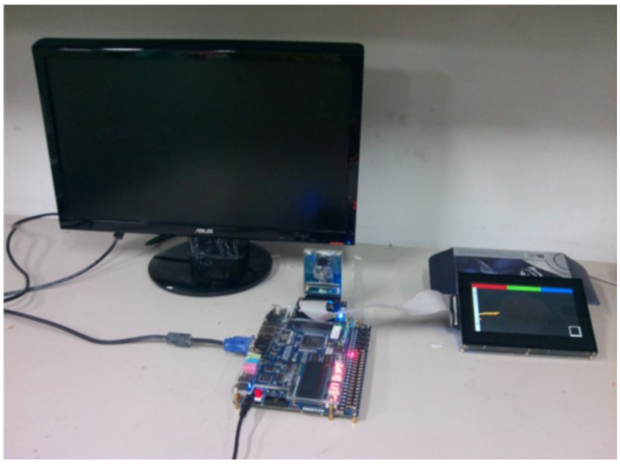

Fig. 1 Electric hardware

Flow Summary	
Flow Status	Successful - Sun Jul 21 14:56:46 2013
Quartus II 64-Bit Version	13.0.0 Build 156 04/24/2013 SJ Web Edition
Revision Name	DE2_115
Top-level Entity Name	DE2_115
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	4,789 / 114,480 (4 %)
Total combinational functions	4,053 / 114,480 (4 %)
Dedicated logic registers	2,539 / 114,480 (2 %)
Total registers	2539
Total pins	519 / 529 (98 %)
Total virtual pins	0
Total memory bits	789,964 / 3,981,312 (20 %)
Embedded Multiplier 9-bit elements	0 / 532 (0 %)
Total PLLs	2 / 4 (50 %)

Fig. 2 Compilation result

II. FUNCTION DESCRIPTION

A. Touch Screen Interface

On the touch screen, there are two dyeing modes(Fig. 3 & Fig. 4), default and adjusting. Users can switch them by sliding up and down the left edge of the touch screen.

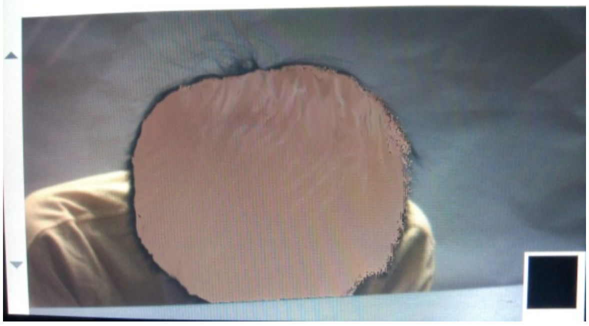

Fig. 3 Default Mode

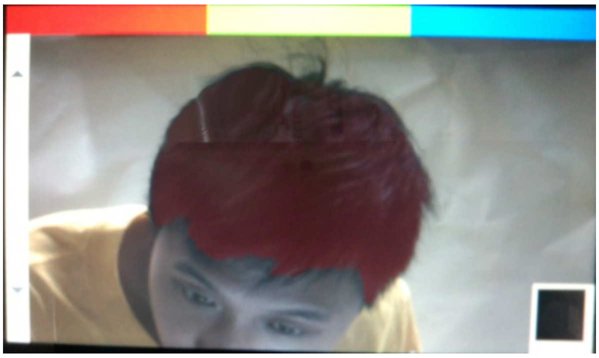

Fig. 4 Adjusting Mode

Meanwhile, the user can capture their images at any time they want by pressing the image at the right down corner.

B. Default Mode

This function is designed for customers of hair salon to compare and choose from hair color category. User can switch to default mode by sliding down the left edge.

Under default mode, user can simply change the current hair type to any types in the 55 default data base. Simply just slide the screen left or right to do so. The current hair type will be shown on the "SEG7 display" on the FPGA.

C. Adjusting Mode

As for adjusting mode, which is designed for the hair salon to add a new hair color type product ,user can switch to it by sliding up the left edge. Under this mode, user can choose red, green and blue form the top edge. Then increase or decrease its value by sliding left or right on the screen to compose the new product's hair type. The current RGB value will be shown on the LCD screen of FPGA.

D. Capturing Photo

In any mode, user can capture a picture of current image as a photo. This function is designed for the customer to show the hair salon which color they wants to dye. Or, if they couldn't decide by their own, take the picture home and ask for other's advice(The hair salon could charge them for the picture).

III. ENVIRONMENT & COMPONENTS

The system is based on the Altera Cyclone FPGA built on the DE2-115 board. A 500-megapixel D5M camera with 2560 x2160 full-resolution is used to capture the user's image, and a LTM LCD is for used to display the RGB value of hair color and 7 SEG displays used to show the current hair type. Besides, there is a multi touch LCD module for most of the user input. In addition, in order to make the system work more accurately and completely, a try on room is provided (Fig. 5 & Fig. 6) with adjustable light source so user can see the results under different lighting conditions.

Fig. 5 Try on room(front)

Fig. 6 Try on room(inside)

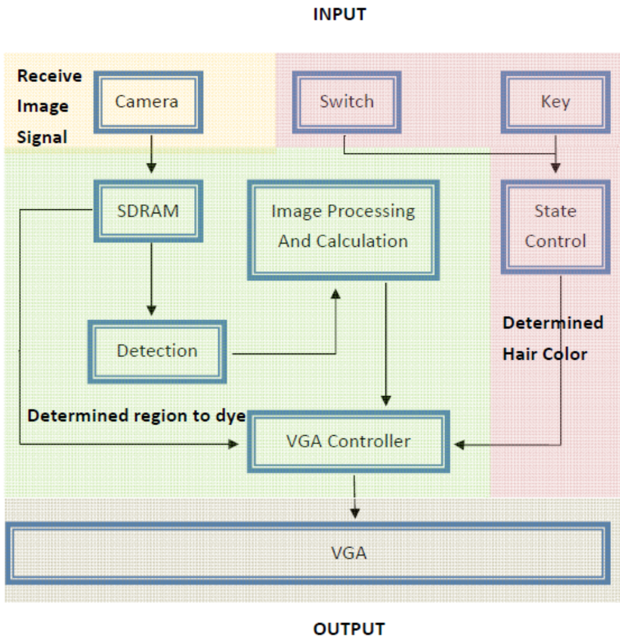

Fig. 8 Block diagram of Functions

IV. System Architecture

In the system, all functions are real time, instantly response. In this case, pre-processing time is really not much. Fortunately some efficient algorithms are established in this paper and will be describe in the next part.

Basically, most of the functions are process in parallel. Fig. 7 & 8 provides a simple view of our system. But for the sake of easily understanding. each function will be describe sequentially bellow.

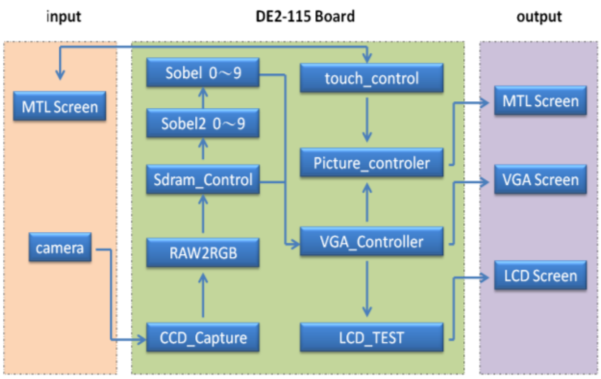

Fig. 7 Block Diagram of Modules

A. System Input

The system input is mainly the live video of the user in the "try on room"(Fig 6). Once the FPGA has receive the image's binary data, a module called "raw2rgb" is used to transfer it as RGB value of every pixel. Then those pixel will be send to "sdram control module" and be stored in the sdram on the FPGA.

B. Detecting Hair Region Stage

After a pixel have been stored in sdram, it will then be sent to 18 processors related to detecting hair region. There are two kinds of processors called sobel1 and sobel2. Each has nine of them which numbered from 1~18. The 18 processors work independently so they can process in parallel to make the detection faster.

C. User input

As mentioned in the previous part, a multi touch screen is used to make the system more instinct. The multi touch screen will continuously receiving the coordinates of user's finger and transport it to the "touch control" module, which will transfer them into gesture and position. Therefore another module called "picture controller" can decide which modes (Default or Adjusting) to switch the system, which color should be displayed and whether the system should take picture base on these information.

D. Display Stage

Finally, our biggest module called "VGA controller" receives "region to dye" from the "18th processor" and color type from "picture controller". It is ready to "dye the hair"!

The VGA controller will dye each pixel of hair into proper color based on our "Color Scaling Algorithm" so that the texture of hair can be preserved instead of making users look awkward with some strange color region on their heads. Also, the VGA controller handles the display of user interface on the touch screen, RGB value on the LCD screen and hair types number on the 7 Segment display(Fig. 9). So the user can manipulate them easily without any difficulties.

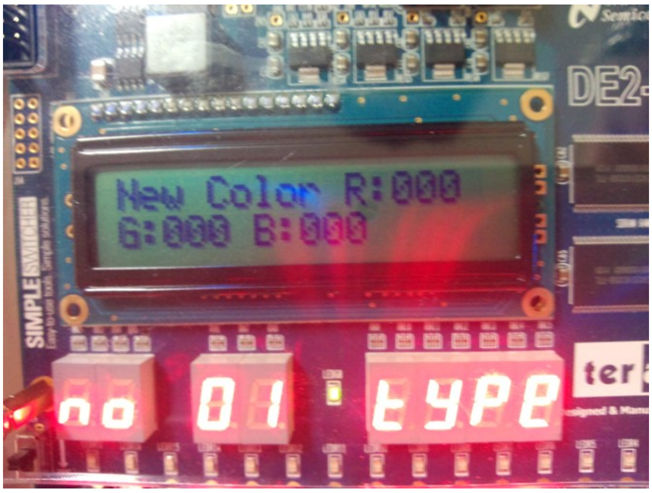

Fig. 9 LCD Screen & 7 segment display

All the functions above can be done in no time to realize real-time hair dyeing simulation.

V. Hair Detection Algorithm

Hair detection is one of the main challenges in our implementation. Most of the methods mentioned on papers required a long processing time. Yet in order to realize the real time simulation, such delay can't be afforded. Therefore, a simple but fast algorithm is established in this paper to detect the hair. That is "multiple erosion and dilation".

A. Detect Possible Area

After image is received from the camera, the possible area is detected based on the lightness and saturation of user's hair. After this stage, the possible area should include hair, eyebrows and eyes for those who has the same color of eyes and hair.

B. Multiple Erosion

In the verilog, the whole image can't be process at once because the memory and real-time limitation. All one can do is to process the image pixel by pixel base on little information around it.

So "Multiple Erosion" is used to filter out the eyebrows and eyes under this constrain.

For a pixel, "erosion" means to "And" the eight pixel around it. In fig. 10, P5 will be zero if any of pixels around it is zero. This algorithm is commonly used for phase out noise of binary image signal.

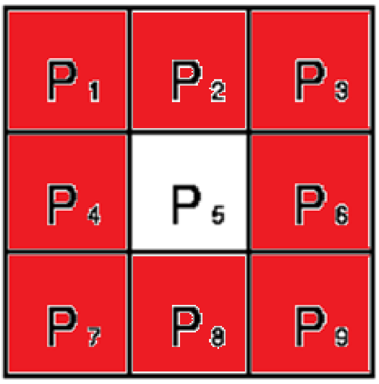

Fig. 10 Erosion & Dilation

In our algorithm, we simply do this again and again until the eye and eyebrow part disappear completely, but the hair region still remains a small area.

This is completely different from what erosion was meant for. The idea of this algorithm was come up by thinking of eye area as a "bigger noise". It is simple, but effective.

Furthermore , this algorithm takes advantage of the thin shape of eyebrow. P5 only "And" the P1~3 and P7~8 in fig. 10, so the letter "multiple dilation " can grow back faster with better effect.

C. Multiple Dilation

After multiple erosion has left a small area on hair, " multiple dilation" is used to grow back the hair region.

Like erosion, the dilation of a pixel reference the 8 pixel around it. The difference is that erosion uses "Or" for the 8 pixels so if any one of them is one, the target pixel will also become one.

The "multiple dilation" dilates the remain area until it exceed the origin possible hair area. Then "And" is taken to the exceeding region with the original possible hair area. So the details of hair edge can be preserved.

Fig. 11 and Fig. 12 shows the matlab simulation of a scenery picture before and after "multiple erosion and dilation".

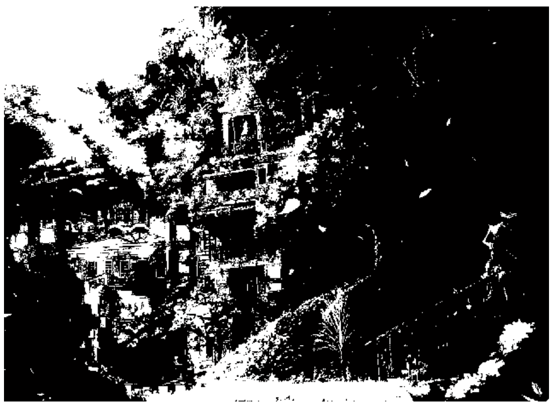

Fig. 11 Before Multiple Erosion & Dilation

Fig. 12 After Multiple Erosion & Dilation

As shown above, regions that are too small to be the target (noise, eyes and eyebrows in our implmentation) are filtered out. Larger regions (User's hair in our implementation) still preserve their detail.

But, after using "line buffer" to handle nine pixel by nine pixel. A delay problem was discovered because a pixel needs information before it. The result will shift to left down corner one pixel by a time after each erosion or dilation, causing the result being unacceptable deviate from the origin image.

Many ways were tried to make this algorithm more faster, but in the end this problem was solved by a easy but enlighten method. Simply makes the origin image slower to consist with result so that the detecting region can easily keep up with it without using any speed-quality trade off on the algorithm.

VI. Color Scaling Algorithm

After the dye region has been determined, all the work left is to decide the color to be dyed.

The VGA controller will receive a RGB value from whether the database or user defined. If the values are simply added onto hair, the color will look like one pure region without any texture of hair, making user looks like a strange clown with a funny hat.

Therefore an algorithm is required to scale the color based on original hair and the new dyeing color will be what the user wants.

In this paper, the defect of human eyes' ability on receiving brightness is being used.

A. Taking Advantage of Human eyes' defect

As a well known knowledge, human eyes aren't as perfect as camera from many points of view. Fig. 13 shows an example of human eyes' ability to tell the difference of luminous various on spectrum of image. The sensitivity changes with the spatial frequency of the stripes.

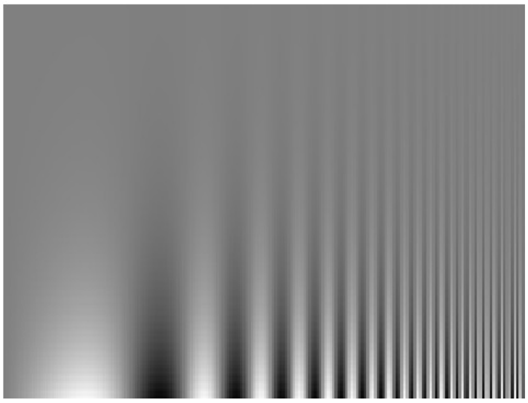

Fig. 13 Human eyes' defect

The implementation can take advantage of the sensitivity of "human eyes' perceivable luminance threshold".

Observe fig. 14, human eyes can tell the best different at medium luminance. In the origin dyeing method., the pixel has been add to over 1000 cd/m^2, where human eyes' ability to tell different becoming worse.. That's why instead of preserving hair texture, the origin result shows a pure color area without any difference.

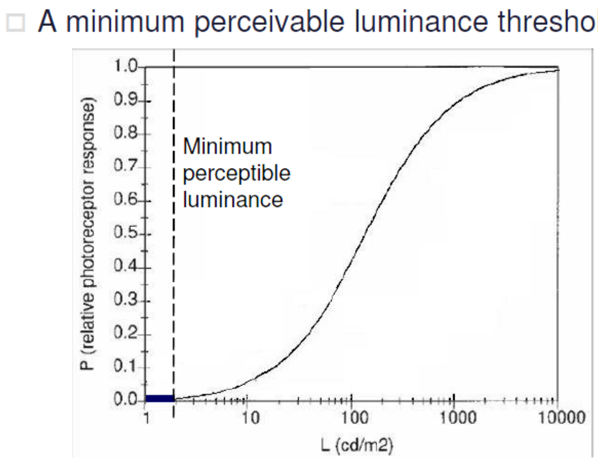

Fig. 14 Human eyes' perception curve

Since the problem of origin hair dyeing method has been found, the knowledge should now be used to improve the result. The target is to extend the detail to the region that human eye can tell difference the most. Fig. 15 shows how the implementation works.

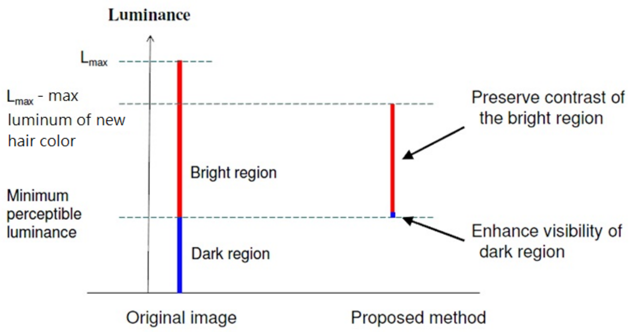

Fig. 15 Use Human eyes' detection to dye the hair

Combine with Fig. 14, the region which human eyes can't tell difference well (under 50 cd/m^2 and over 1000 cd/m^2) has been shifted to where human eyes work best (That is 50~1000cd/m^2, where has the sharpest slope). Also, the upper limit for the maximum value of the RGB value has been set so that it won't exceed 255 after we add the dyeing color on it. 255 is the maximum value of RGB value and will go back to zero if exceed the limit. (The result will apear some strange stain-like spots if we didn't handle the limitation and some pixels exceed the 255 value .).

Fig. 16 shows Matlab simulation of the process before dyeing the desire color on it.

(a) (b) (c)

Fig. 16 Matlab Simulation of Color Scaling Algorithm

As shown above, (a) is the origin picture, (b) is the result that directly scale the color linearly from zero to max luminance of limitation, and (c) is the result that using the "color scaling algorithm" mentioned in this paper.

It can be easily observed that (c) shows much more details for human eyes than (b). After putting the desired color on it, method (c) will preserves much more texture for human eyes then method (b). This is how "color scaling algorithm" works

VII. Conclusion

In this paper, a cosmetic system named "Virtual Stylist", which can detect user's hair based on the property of their hair color and phase out eyes, eyebrows and even some small noises by applying some simple but smart algorithms based on FPGA implementation, is presented. Also, the defect of human eyes in low luminance detection is being taken advantage to make the color dyeing algorithm preserves more hair texture than simply added colors on hair.

The color of hair to be dyed can be chosen from "default mode" or "adjusting mode". The previous one is for the user to "try on" products in hair salon and the post one is for the hair salon to add new product to its category. No matter how you move or change your position during the system working time, the system can always change your hair color accurately. It is just like a mirror that reflects yourself after you dye your hair.

In addition, these functions have been put on multi touch board to make them more instinct for users. Combined with new design ideas and hardware techniques, this system provides users a flexible real-time simulation of hair dyeing.

Acknowledgment

We are grateful that Altera® Inc. and Terasic® Inc. give us a chance to show our project. Their sponsor and support make us understand more about the hardware system. Also, we would like to thank Professor Shao-Yi Chien and National Taiwan University, for the excellent training and comfortable environment so that we can focus on our project and finally fulfill our goal successfully.

References

[1] I. Choi , H. Shim, and N. Chang, "Low-power color TFT LCD display for hand-held embedded systems," in Proc. ISLPED, Monterey, California, pp. 112-117, 2002.

[2] W. C. Cheng, Y. Hou, and M. Pedram, "Power minimization in a backlit TFT-LCD display by concurrent brightness and contrast scaling," in Proc. Conf. Design, Automation and Test in *Europe, pp. 10252-10259,* Feb. 2004.

[3] L. Cheng, S. Mohapatra , M. E. Zarki , N. Dutt , and N. Venkatasubramantan , "A backlight optimization scheme for video playback on mobile devices," in *Proc. CCNC, Las Vegas, Nevada, pp.* 833-837, Jan. 2006.

[4] Tony F. Chan, Member, IEEE, and Luminita A. Vese "Active Contours Without Edges" in *IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 10, NO. 2, pp.266-277., FEBRUARY 2001*.

[5] Hong Chen and Song-Chun Zhu, "A Generative Sketch Model for Human Hair Analysis and Synthesis" in *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 28, NO. 7, pp.1025-1039JULY 2006*

[6] Uri Lipowezky[1], Omri Mamo[2], and Avihai Cohen."USING INTEGRATED COLOR AND TEXTURE FEATURES FOR AUTOMATIC HAIR DETECTION" *Electrical and Electronics Engineers in Israel, 2008. IEEEI 2008. IEEE 25th Convention of*

[7] C. Rousset and P.Y. Coulon."FREQUENCY AND COLOR ANALYSIS FOR HAIR MASK SEGMENTATION"*Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*

[8] Linjie Luo, Hao Li, Sylvain Paris, Thibaut Weise, Mark Pauly, Szymon Rusinkiewicz"Multi-View Hair Capture using Orientation Fields"*Computer Vision and Pattern Recognition (CVPR), June 2012*