FINAL YEAR PROJECT
FINAL PRESENTATION

HARDWARE-BASED FACE DETECTION

Supervisor: Dr. Benjamin Carrion Schafer
Student: Zhang Yunlei
Contents

• 1. Introduction
• 2. Design Methodology & Result
  – 2.1 Design Objective
  – 2.2 Real-time video display
  – 2.3 Real-time face detection based on face color and hair color
  – 2.4 Higher accuracy with the help of motion detection
• 3. Reflection & Outlook
• 4. Conclusion
1. Introduction
1. Introduction

Face is Identity
1. Introduction

Face Detection (Location) is basis
1. Introduction

Why Use Color Detection

- Three key points: speed, accuracy, stability
- Many methods: Principal Component Analysis (PCA), Neural Networks (NN), Support Vector Machines (SVM), Hough Transform (HT), Geometrical Template Matching (GTM), color analysis, etc
Why Use Color Detection

• Real-time face detection requires large data flow

• Repetitive data accessing is impossible
1. Introduction

Improve Accuracy

• However, simple color detection may not be reliable
1. Introduction

Improve Accuracy

• Only head has hair on the top
• Hair helps locate face

Use hair as assistance
1. Introduction

Improve Accuracy

- How about black background?

Motion Detection!!
1. Introduction

Why FPGA

- Parallel processing
- More powerful functions in further research
1. Introduction

• Hardware:
  – Altera Cyclone IV (EP4CE15F17C8)
  – TFT Screen (AT070TN83) (800*480, RGB565)
  – CMOS Camera (OV7670) (640*480, RGB565)
  – SDRAM(H57V2562GTR)

• Software:
  – Quartus II
  – Modelsim

• HDL:
  – Verilog
2. Design Methodology & Result
2.1 Design Objective

- Step 1: Real-time video display
- Step 2: Real-time face detection based on face color and hair color
- Step 3: Real-time face detection with the help of motion detection
2.2 Real-time Video Display
2.2 Real-time Video Display

Block Diagram of Video Display Module
2.2 Real-time Video Display

Data Path of Video Display Module
Data Path of Video Display Module

2.2 Real-time Video Display

Clock Domain Crossing

Use burst mode
Ping-Pong Operation

- Single SDRAM based Ping-Pong Operation
Ping-Pong Operation

- Single SDRAM based Ping-Pong Operation
2.3 Real-time Face Detection Based on Face Color and Hair Color
2.3 Real-time Face Detection Based on Face Color and Hair Color

Block Diagram of Face Detection Module
Data Format and Color Space

- Simple face color detection
- RGB565

Range for R,G,B: [0, 31]

Selected face color space:

\[ 5 < R < 17 \land 5 < G < 17 \land 5 < B < 17 \land R - B > 1 \land R - G > 1 \]
2.3 Real-time Face Detection Based on Face Color and Hair Color

Color Space Testing

Hair can help!
Color Space of Hair

- Hair color space is selected as following

\[ R < 9 \& G < 9 \& B < 9; \]
\[ |R - G| \leq 1 \& |R - B| \leq 1 \& |B - G| \leq 1; \]
2.3 Real-time Face Detection Based on Face Color and Hair Color

Face Boundary Obtain
2.3 Real-time Face Detection Based on Face Color and Hair Color

Face Boundary Obtain

- Flow chart of hair boundaries obtain module
2.3 Real-time Face Detection Based on Face Color and Hair Color

Eliminate Noise
2.3 Real-time Face Detection Based on Face Color and Hair Color

Eliminate Noise
2.3 Real-time Face Detection Based on Face Color and Hair Color

Eliminate Noise
2.3 Real-time Face Detection Based on Face Color and Hair Color

Find Center Point of Face

```verbatim
else
begin
    if (frame_done_fail)   // what to do when one frame ends: row++ clear column
        begin
            num <= 26'd0;
            whole_col <= 26'd0;
            whole_row <= 26'd0;
        end
    else if (dataen)   // when frame and row effective
        begin
            if(red <= 10'd17 && red > 10'd5 && green < 10'd17 && green > 10'd5 && blue < 10'd17 && blue > 10'd5 && red - b
                begin
                    whole_col <= whole_col + cont_col_num;
                    whole_row <= whole_row + cont_row_num;
                    num <= num + 1;
                end
        end
end

if ||(column_num - centre_column < 10 || centre_column - column_num < 10) && (row_num - centre_row < 10 || centre_row - row_num < 10) ||
    // if ||(column_num < 320 && column_num > 300) && (row_num < 240 && row_num > 200) ||
begin
    CMOS_oDATA[15:0] <= 16'b11111_000000_00000; // show the centre point of face and mark it with red color
end
```
2.3 Real-time Face Detection Based on Face Color and Hair Color

Result
2.3 Real-time Face Detection Based on Face Color and Hair Color

Result
2.4 Higher Accuracy with the Help of Motion Detection
2.4 Higher Accuracy with the Help of Motion Detection

Function

• Remove Unchanged Background
2.4 Higher Accuracy with the Help of Motion Detection

Block Diagram with Motion Detection Module
Construction of Motion Detection Module

- Similar to Range_Obt module
3. Reflection and Outlook
3. Reflection and Outlook

FPGA Report

Modify algorithm to reduce resource consumption

<table>
<thead>
<tr>
<th>Flow Status</th>
<th>Successful - Tue Apr 14 12:04:35 2015</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quartus II 64-Bit Version</td>
<td>12.0 Build 178 05/31/2012 SJ Full Version</td>
</tr>
<tr>
<td>Top-level Entity Name</td>
<td>sdram_ov7670_vga</td>
</tr>
<tr>
<td>Family</td>
<td>Cyclone IV E</td>
</tr>
<tr>
<td>Device</td>
<td>EP4CE15F17C8</td>
</tr>
<tr>
<td>Timing Models</td>
<td>Final</td>
</tr>
<tr>
<td>Total logic elements</td>
<td>2,992 / 15,408 (19%)</td>
</tr>
<tr>
<td>Total combinational functions</td>
<td>2,848 / 15,408 (18%)</td>
</tr>
<tr>
<td>Dedicated logic registers</td>
<td>721 / 15,408 (5%)</td>
</tr>
<tr>
<td>Total registers</td>
<td>721</td>
</tr>
<tr>
<td>Total pins</td>
<td>84 / 166 (51%)</td>
</tr>
<tr>
<td>Total virtual pins</td>
<td>0</td>
</tr>
<tr>
<td>Total memory bits</td>
<td>16,384 / 516,096 (3%)</td>
</tr>
<tr>
<td>Embedded Multiplier 9-bit elements</td>
<td>0 / 112 (0%)</td>
</tr>
<tr>
<td>Total PLLs</td>
<td>1 / 4 (25%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flow Status</th>
<th>Successful - Sat May 09 23:24:00 2015</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quartus II 64-Bit Version</td>
<td>12.0 Build 178 05/31/2012 SJ Full Version</td>
</tr>
<tr>
<td>Top-level Entity Name</td>
<td>sdram_ov7670_vga</td>
</tr>
<tr>
<td>Family</td>
<td>Cyclone IV E</td>
</tr>
<tr>
<td>Device</td>
<td>EP4CE15F17C8</td>
</tr>
<tr>
<td>Timing Models</td>
<td>Final</td>
</tr>
<tr>
<td>Total logic elements</td>
<td>3,519 / 15,408 (23%)</td>
</tr>
<tr>
<td>Total combinational functions</td>
<td>3,215 / 15,408 (21%)</td>
</tr>
<tr>
<td>Dedicated logic registers</td>
<td>1,181 / 15,408 (8%)</td>
</tr>
<tr>
<td>Total registers</td>
<td>1181</td>
</tr>
<tr>
<td>Total pins</td>
<td>93 / 166 (56%)</td>
</tr>
<tr>
<td>Total virtual pins</td>
<td>0</td>
</tr>
<tr>
<td>Total memory bits</td>
<td>53,248 / 516,096 (10%)</td>
</tr>
<tr>
<td>Embedded Multiplier 9-bit elements</td>
<td>0 / 112 (0%)</td>
</tr>
<tr>
<td>Total PLLs</td>
<td>1 / 4 (25%)</td>
</tr>
</tbody>
</table>
Determine Color Space

- Use logic analyzer to transmit data to computer
- Find the typical color space of hair and face for this camera
- Find different sets of color space in different light conditions
- Find different sets of color space for people of different hair and skin color
Support More Faces

• More faces can be supported by calculating the density of hair/face pixels.
4. Conclusion

• 1. In this project, a face detection system is built on FPGA, utilizing its parallel processing nature.
• 2. Color detection approach is used in this project.
• 3. Hair is used as boundaries of face, to filter the noise in the background.
• 4. Motion detection helps to filter the hair color noise in the background.
• 5. Several work are involved in future research: color space determination, supporting more faces and improve algorithms.
Thank you!