Real Time Feature Detection

-Somani Patnaik
-Jon Losh
-Dember Giraldez
11/17/09

Overview

Objective: Real-time image processing and feature detection

- Histogram Equalization (Somani)

- Edge detection (Jon)

- Corner detection (Dember)
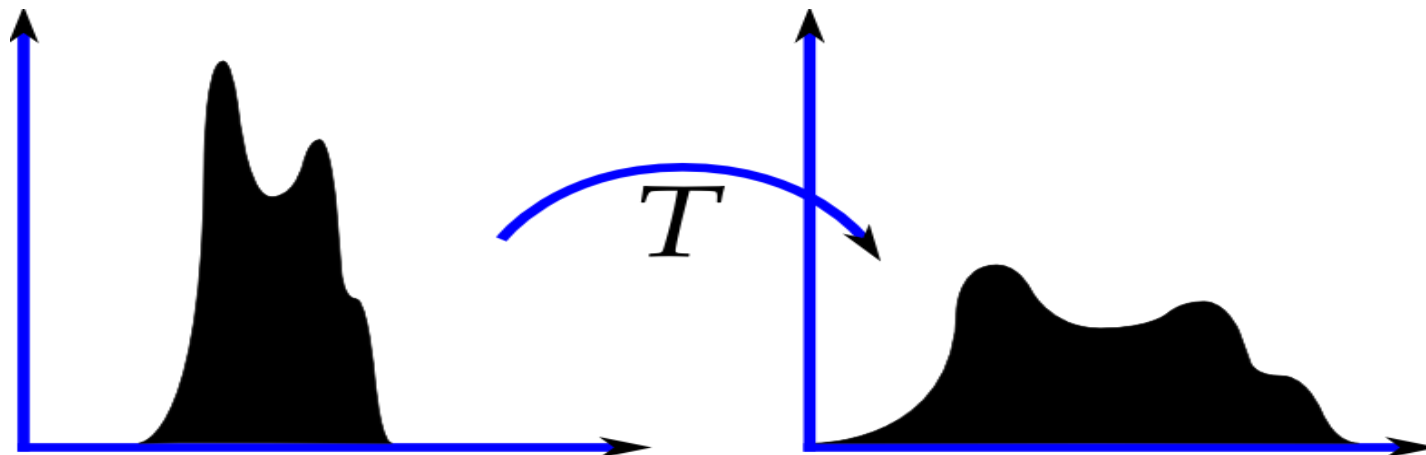
- Face detection with monochrome background

Why?

- Building blocks for computer vision

- Computationally intensive

- Take advantage of hardware platform

Histogram Equalizer

- Increases the contrast in the image
- Makes it possible to ignore different lighting conditions

Histogram Equalizer (Somani)

Histogram Builder
-The intensity histogram statistics for the image is calculated
- Counts the number of times a particular intensity appears in the image

Histogram Equalizer
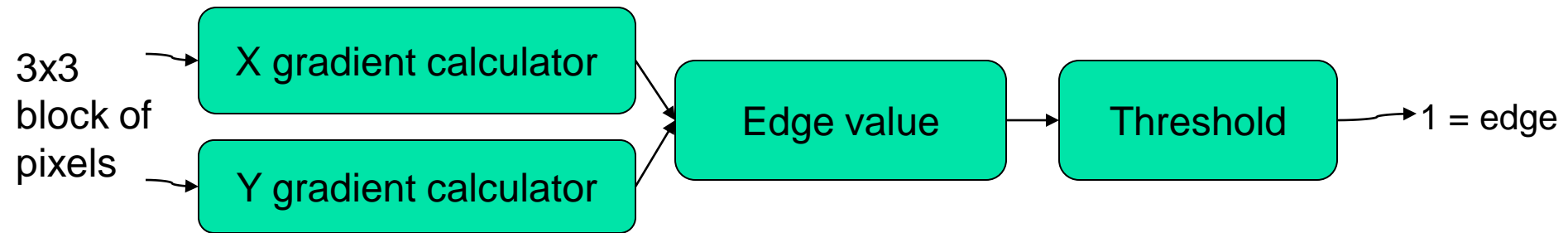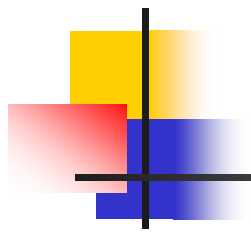- The intensity values are normalized by calculating the CDF

Image Builder
-It maps the CDF of the original intensity values to the entire range of intensity values
-For every pixel, it returns a new intensity value based on the look up table

Histogram Equalized image

Sobel Edge Detector (Jon)

3x3 block of pixels → [X gradient calculator] → [Edge value] → [Threshold] → 1 = edge

3x3 block of pixels → [Y gradient calculator]

- Detects sharp changes in grayscale

- x gradient operator
$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

- y gradient operator
$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$
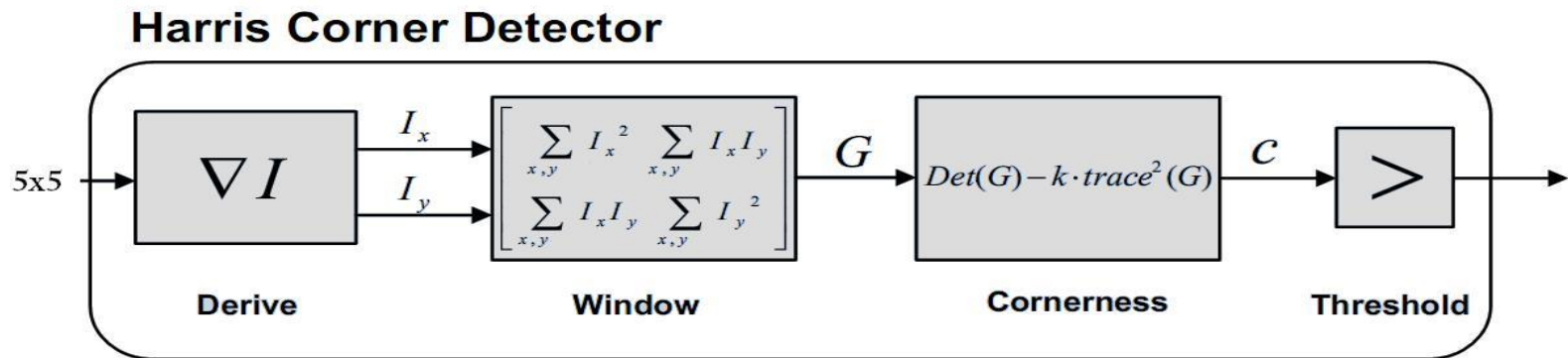
- Add results together to get edge value

- On FPGA, compare this edge value to a threshold value

Sobel example

# Harris Corner Detector (Dember)

Harris Corner Detector

$$5\text{x}5 \rightarrow \boxed{\nabla I} \begin{matrix} I_x \\ I_y \end{matrix} \rightarrow \begin{bmatrix} \sum_{x,y} I_x^2 & \sum_{x,y} I_x I_y \\ \sum_{x,y} I_x I_y & \sum_{x,y} I_y^2 \end{bmatrix} \xrightarrow{G} \boxed{Det(G) - k \cdot trace^2(G)} \xrightarrow{C} \boxed{>} \rightarrow$$

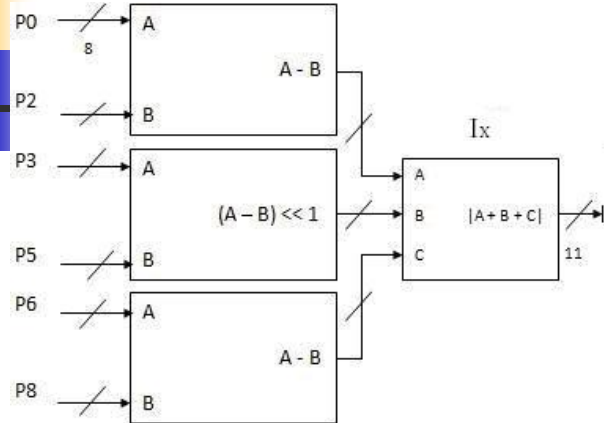| Derive | Window | Cornerness | Threshold |

-Module determines whether a pixel is a corner or not.

-Input consists of 5x5 window of pixels.

-Need to compute gradients in the x and y directions.

-FPGA can instantiate many of these modules and run them in paralell.

-Main advantage: parallelism.

-Main problem: memory bandwidth.

Some modules

Find dI/dx and dI/dy

(purely combinational)
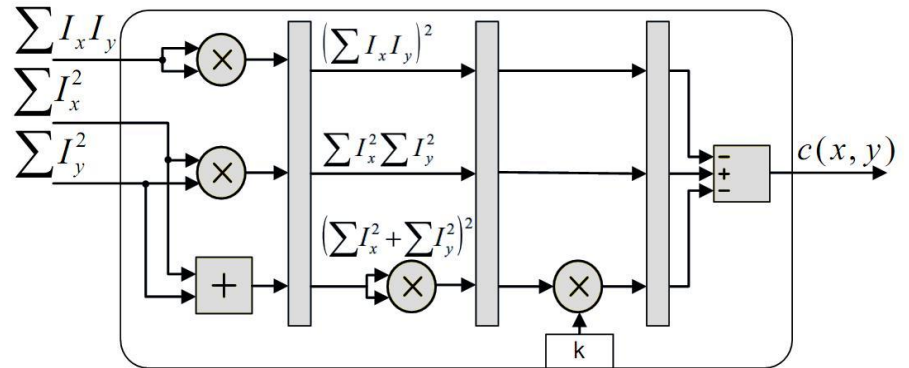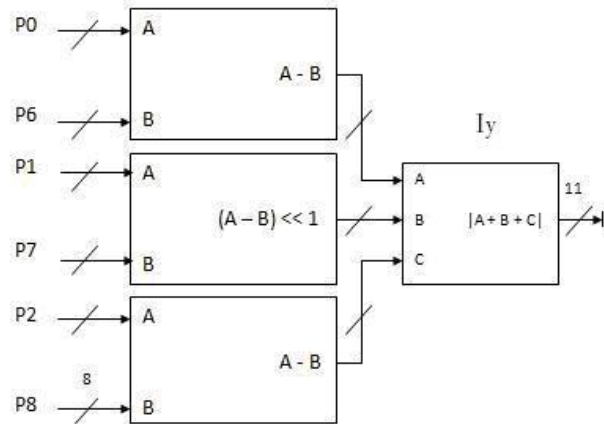
Pipelined corner score module

Corner Detection Example

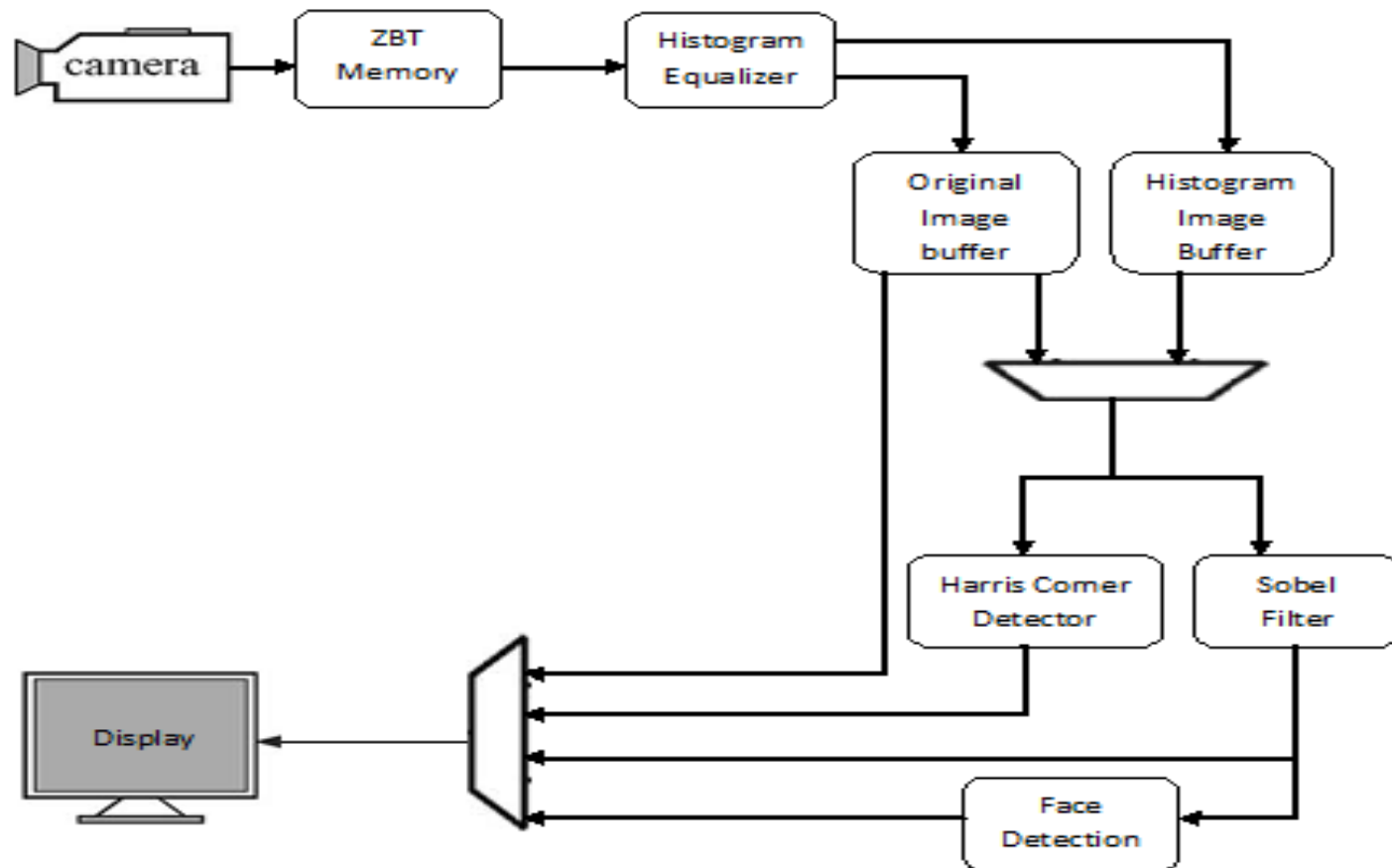Picture from: http://psychmamma.files.wordpress.com/2009/01/face.jpg

Threshold = 0.01

The Big Block Diagram

Challenges

- ## Memory details
  - Limited memory bandwidth

- ## Interfacing between individual modules
  - Adjusting different time constraints and instantiating multiple modules.

- ## Face detection
  - Huge training overhead
  - Tradeoff: software implementation vs. digital design

Time Line

- Specifications for each module. Interface camera with NTSC decoder. Software implementations for each module. → 11/15 (completed)

- Module that writes camera output to memory array that edge and corner modules can use. --> 11/18

- Individual modules (Harris Corner, Sobel, Histogram modules) --> 11/22 at 1pm

- Integration of modules --> 12/3

- Adding face detection capability with monochrome background --> 12/7