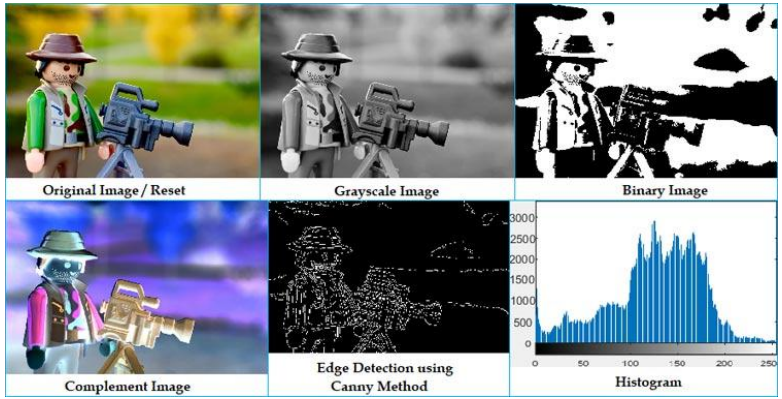
Image processing with Raspberry Pi 3

Group I: Jose Soto, Tomik Baghramian
Cedrik Jay Toledo, Bruce Chua

Image Processing

- Image processing is the process of manipulating images through the use of digital computers
- Variety of uses throughout many fields
  - Entertainment
  - Medicine
  - Remote sensing
  - Geological mapping
- Images can be manipulated through enhancement, restoration, analysis, and compression



Project Goals

- The software and hardware to perform image processing can be expensive
- It can be tedious and time consuming to alter images without the use of image processing software
- Use a cost effective method to perform image processing
- Testing the image process performance capability on a cost effective board such as the Raspberry Pi 3
- Use free open source image processing algorithms from OpenCV

Project Approach

Approach:

- **Testing capabilities that the Raspberry Pi 3 model B has on image processing**
- **Determine the quality of images through the alterations done with OpenCV**
  - **Resize**
  - **Grey scale**
  - **Threshold**
  - **Blur**
  - **Rotate**
  - **color**

OpenCV

Project Reasoning

Reasoning:

- **Why Raspberry Pi 3 and OpenCV?**
  - **Availability**
  - **Open-source library**
  - **Supports python**
- **We want to understand and analyze the performance of image processing on a Raspberry Pi 3**
- **Testing if this method of image processing is cost effective based on its performance**

OpenCV

- Open source software that runs python and c language that can do image processing.
- cv2.imread
  - Reads and loads a file based on directory
  - Must be exact or openCV will not be able to tell what file the user is using
- cv2.resize
  - Resizes an image dependent on what parameters are used
- cv2.imshow
  - Opens a window where user can see image processed
- cv2.imwrite
  - Saves file wherever the user wants and must be named what to be saved as
- cv2.cvtColor
  - Converts file to whatever color the user wants based on OpenCV library of choices

OpenCV

- cv2.blur
  - Smoothes an image by using the kernel of the device
- cv2.threshold
  - pixel values compared to threshold values changes the output of an image
  - Image must be set as a greyscale
- cv2.getRotationMatrix2D
  - Used to get transformation matrix that is needed to rotate the image
- cv2.warpAffine
  - Uses transformation matrix from getRotationMatrix2D to rotate the image

- Github contains python files of implementation with comments under FInal Project/Code

Conclusion

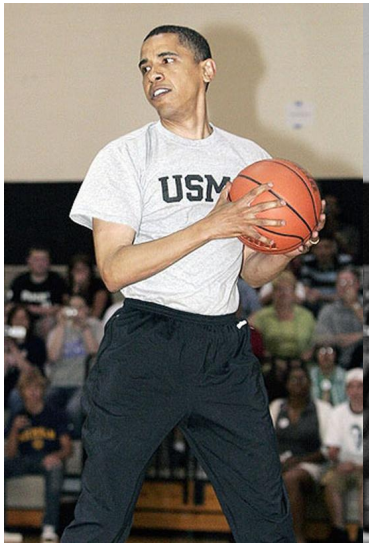
Image Processing

- the alteration of images through the use of digital computers.
- Used in different industries, such as remote sensing, medicine, entertainment.
- Our main goal was to find a cost effective method for image processing and to become familiar with image processing algorithms.

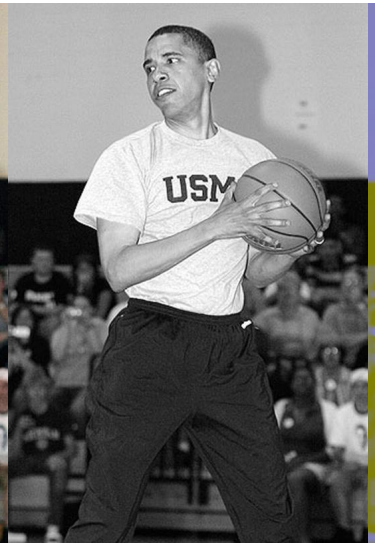
What we achieved and learned

- Processed an image with built-in functions of OpenCV and created the following alterations: Resize, Grey scale, Threshold, Blur, Rotate, Color.
- Raspberry Pi is viable for image processing; However it has limitations, such as low RAM and not being able to run Visual Studio
- OpenCV is also capable running with C++ programming language, however due to some limitations we stuck with python and built-in functions.

Processed Images



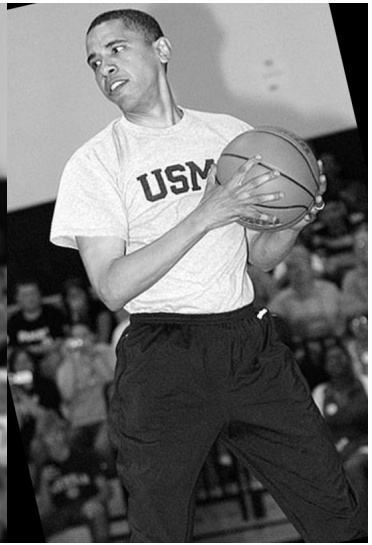
Resized



Grayscale

Colored

Blurred



Rotated

Execution Times

Alteration	Execution Time
Resized	0.465433 s
Grayscale	0.445596 s
Colored	1.261017 s
Blurred	0.489982 s
Rotated	0.461329 s