**CPEG 585 – COMPUTER VISION**

**HOMEWORK 3**

**Name:** Ivan Sangines Escrig

**ID#:** 968606

**Instructor:** Dr. Mahmood

**Date:** February 28, 2019

**TABLE OF CONTENT**

[**INTRODUCTION**](#_Toc446970371) 3

[**SCREENSHOTS:**](#_Toc446970373) 4

[**SOURCE CODE:**](#_Toc446970374) 6

[**CONCLUSION:**](#_Toc446970375) 10

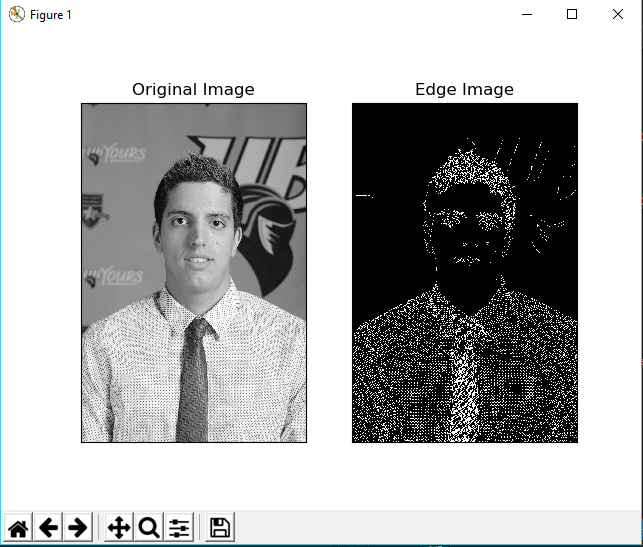
**INTRODUCTION**

The purpose of this assignment is to get familiar and understand better all the Canny edge detection and Harris corner detection. In order to practice this techniques, we will be using Open CV and emgu libraries

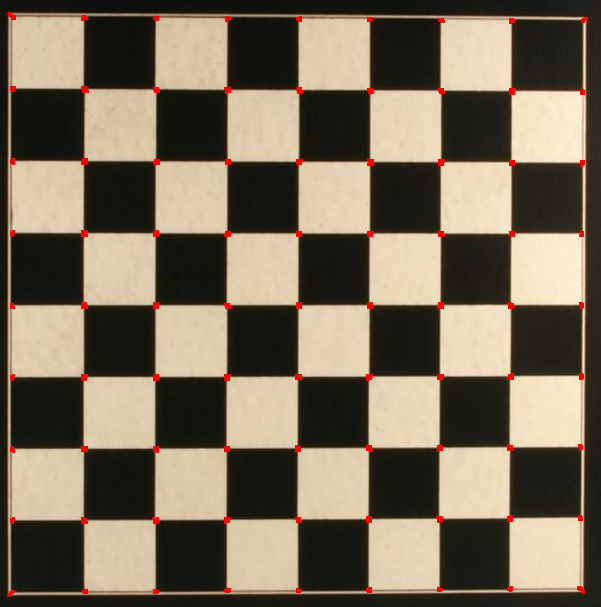
**SCREEN SHOTS:**

**Problem 1:**

In the following screenshot we can see the result of an image, after applying the Cannis function from the open CV library. As we can see on the results, the edges in the image were detected and shown in white color.

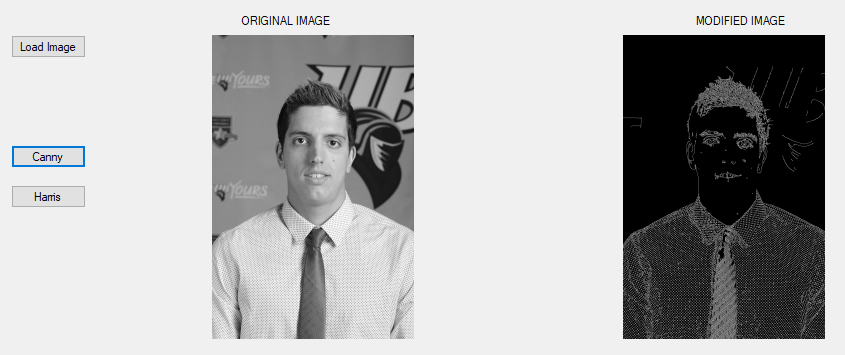
****

On the following screenshot, we can see the results of the Harris corner detection from the open CV library. As shown in the image, the detected corners are marked in red.

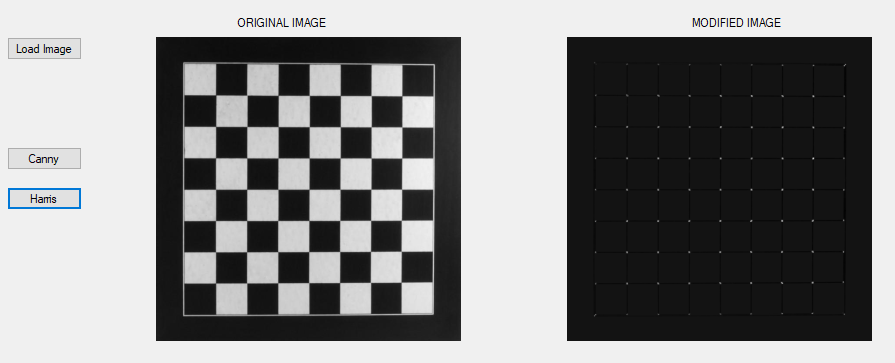


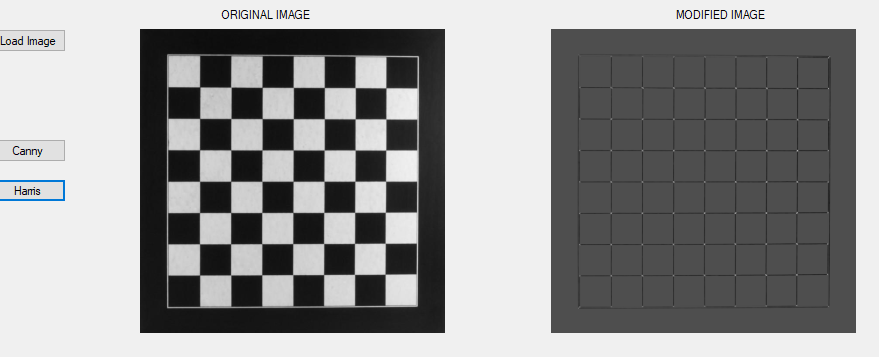
**Problem 2:**

Here we can see the result of applying the Canny edge detection to an image. As we can appreciate in the Modified image, the edges were detected and showed in white.

****

On this last screenshot, we can see how the corners of a chess image are detected and shown in white dots, all the image is converted to a dark color and the corners are marked as white dots. As we can see in the second output, the final color of the image depends on the thresholds we choose.

****

****

**SOURCE CODE:**

**Problem 1:**

import numpy as np

import cv2

from matplotlib import pyplot as plt

# Load a color image in grayscale

img = cv2.imread('C:/Users/ivans\_000/Desktop/MASTER/Spring2019/Computer Vision/Assignment3\_ComputerVision\_SANGINES/ivan.jpg',1) # 0 for gray, 1 for color

#cv2.imshow('image',img)

#---------------canny edge detection-------------

img = cv2.imread('C:/Users/ivans\_000/Desktop/MASTER/Spring2019/Computer Vision/Assignment3\_ComputerVision\_SANGINES/ivan.jpg',0)

edges = cv2.Canny(img,50,200)

plt.subplot(121),plt.imshow(img,cmap = 'gray')

plt.title('Original Image'), plt.xticks([]), plt.yticks([])

plt.subplot(122),plt.imshow(edges,cmap = 'gray')

plt.title('Edge Image'), plt.xticks([]), plt.yticks([])

plt.show()

#-------------------------------------------------

#--------------------Harris Corner Detection------

filename = 'C:/Users/ivans\_000/Desktop/MASTER/Spring2019/Computer Vision/Assignment3\_ComputerVision\_SANGINES/chess.jpg'

img = cv2.imread(filename)

gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

# find Harris corners

gray = np.float32(gray)

dst = cv2.cornerHarris(gray,2,3,0.04)

#result is dilated for marking the corners, not important

dst = cv2.dilate(dst,None)

# Threshold for an optimal value, it may vary depending on the image.

img[dst>0.01\*dst.max()]=[0,0,255]

cv2.imshow('dst',img)

#-------------------------------------------------

cv2.waitKey(0)

cv2.destroyAllWindows()

**Problem 2:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using Emgu;

using Emgu.CV;

using Emgu.CV.Structure;

namespace CannyHarris

{

public partial class Form1 : Form

{

Bitmap bm;

Image<Gray, Byte> img;

Mat resImage = new Mat();

Image<Gray, float> harrisImage = null;

public Form1()

{

InitializeComponent();

}

private void load\_btn\_Click(object sender, EventArgs e)

{

OpenFileDialog dialog = new OpenFileDialog();

dialog.Filter = "jpeg files (\*.jpg)|\*.jpg|(\*.gif)|gif||";

if (DialogResult.OK == dialog.ShowDialog())

{

img = new Image<Gray, Byte>(dialog.FileName);

original\_img.BackgroundImage = img.ToBitmap();

}

}

private void canny\_btn\_Click(object sender, EventArgs e)

{

CvInvoke.Canny(img, resImage, 50, 200);

modified\_img.BackgroundImage = resImage.Bitmap;

}

private void harris\_btn\_Click(object sender, EventArgs e)

{

harrisImage = new Image<Gray, float>(img.Size);

CvInvoke.CornerHarris(img, harrisImage, 2, 3, 0.01);

modified\_img.BackgroundImage = harrisImage.ToBitmap();

}

}

}

**Conclusion:**

After this assignment I have to say I got a better understanding of the Open CV and emgu libraries. This assignment helped me to understand better the Canny and Harris techniques, seeing the results of each technique really helped me understand better what each of the techniques are used for.

On the other hand, this assignment also helped me to learn new useful functions for image processing such as convert to gray or creation of images.