Assignment 1

**Prepared by**Brendan Burke

**Computer Vision  
CAP 4410.01**

**2/14/18**

**Table of Contents**

[**List of Figures…………………………………………………………………………………………………..2**](#_q5spl5kbolwc)

[**Assignment 1…………………………………………………………………………………………………..4**](#_ndp69edkhypw)

[Assignment Details………………………………………………………………………………………………………………………………...4](#_1fob9te)

[Implementation……………………………………………………………………………………………………………………………………..4](#_2et92p0)

[Resources……………………………………………………………………………………………………………………………………………...4](#_6hcft6gjvm5s)

[Output Images……………………………………………………………………………………………………………………………………….5](#_yjz45vfwoyo6)

**References………………………………………………………………………………………….................7**

# **List of Figures**

**Figure 1** Initial………………....……………………………………………………………................... 2

**Figure 2** Half brightness…………............................................................................................................ 3

**Figure 3** Full brightness……….…………………………………………………………….................... 2

# 

# Assignment 1

## Assignment Details

For this assignment, we implemented a program that simulates the Levels/ Curve Tool from Photoshop. This tool is used to adjust the brightness/ contrast of an input video. During testing, I used the video “barriers.avi” as our test input video. This is a simple black & white video of a car driving on a bridge.

## Implementation

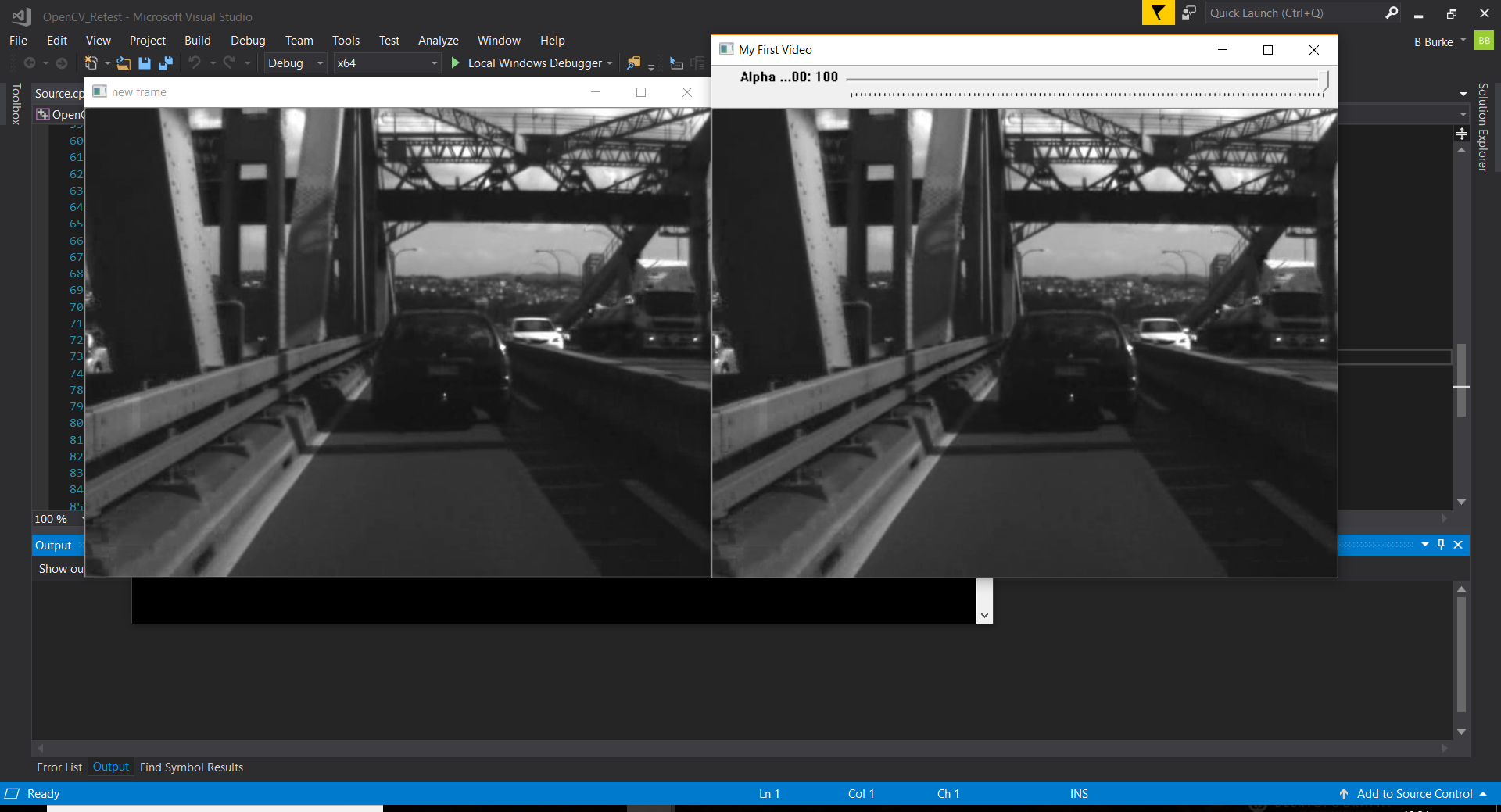
I used a slider, or trackbar, to allow the user to adjust the brightness and contrast of an input video. The program first performs the necessary initialization steps, like getting the input video, initializing the trackbar, and initializing the variables. Then the program creates two windows: one to display the unedited version of the input video (including the two trackbars), and another window to display the edited input video. Adjusting the trackbars values in the unedited window changes the brightness and contrast of the edited window. When the user is finished editing the brightness and contrast, simply by pressing “Enter” commands the program to accept the changes, applies the changes to the original video, closes the edited video window, and removes the trackbars. please note that in some cases the user will have to press “Enter” multiple times for the program to recognize the command. Also, the users may press “Esc” to close the program. Please see section Output Images for figures of the program.

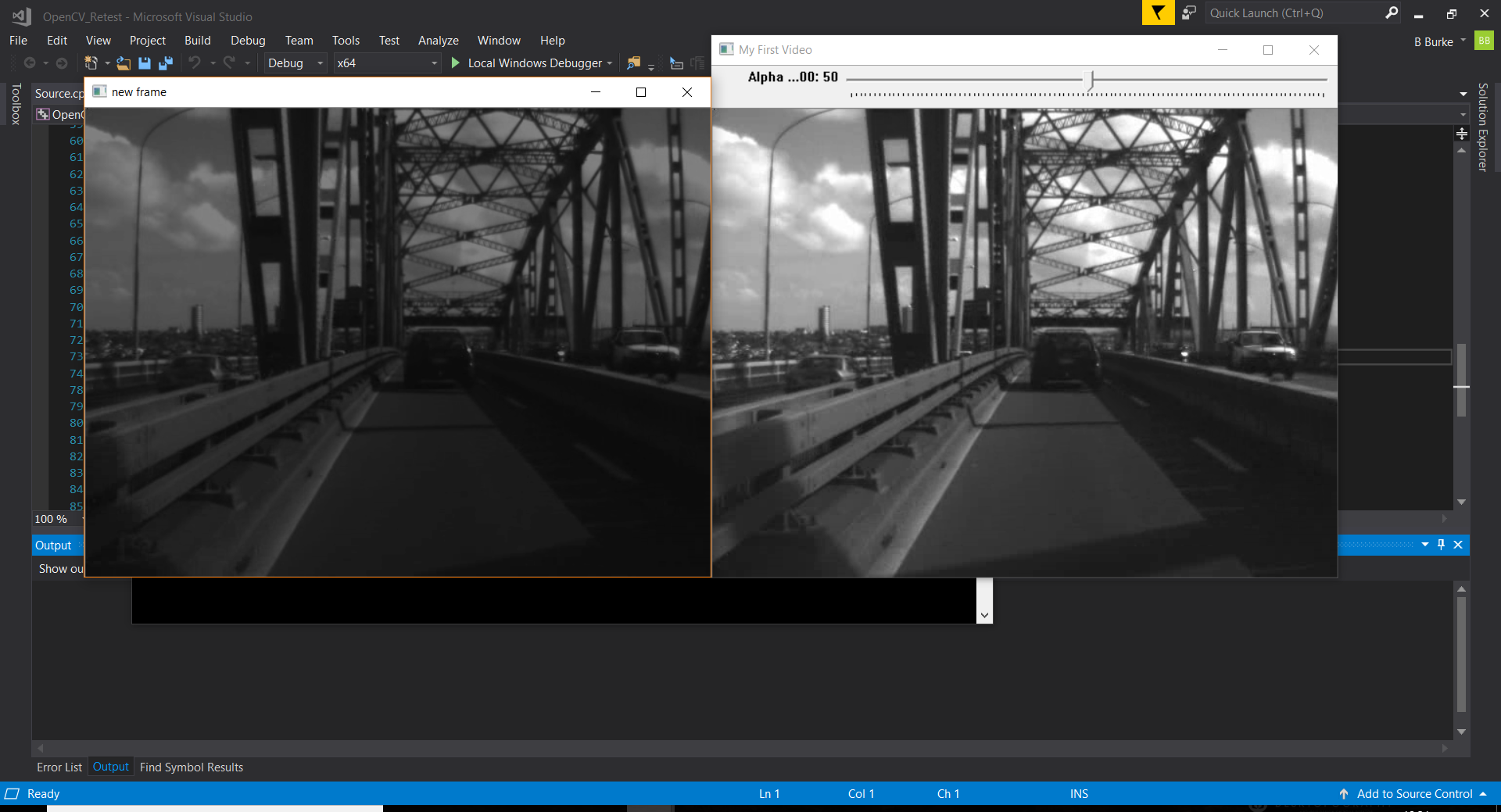
## Resources

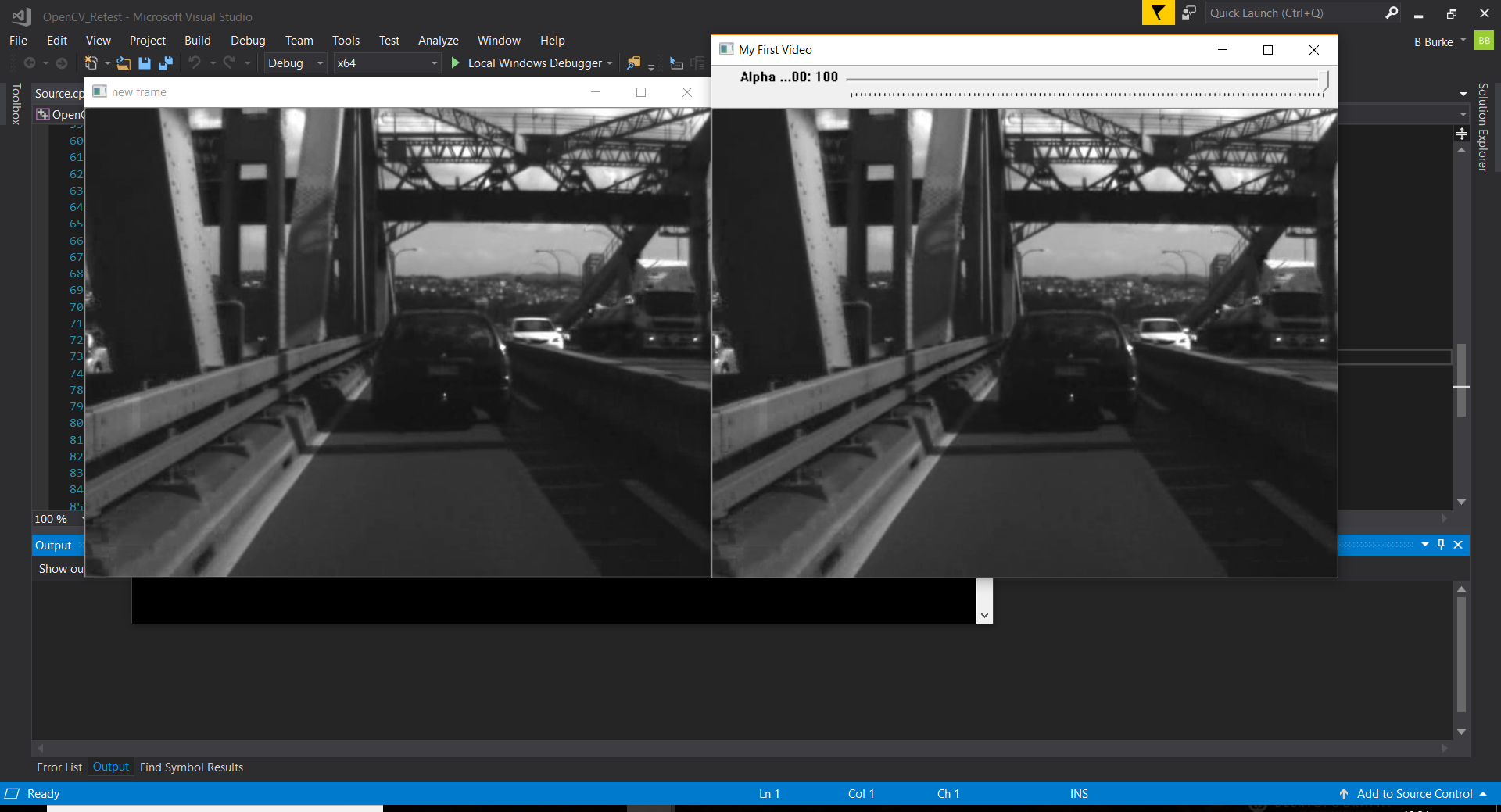
I used several resources for this project. The following list explains the reference and includes a brief explanation:

* StackOverflow - OpenCV: how to restart a video when it finishes?
  + This helped me understand how to play a video on loop. I wanted to implement this functionality so that the user can continue to edit their video, and not be constrained by the duration of their video (program ends when video ends; this helped me resolve this issue).
* OpenCV 3.1.0 documentation
  + VideoCapture - CAP\_PROP\_POS\_AVI\_RATIO
    - This also helped me play a video on loop. Specifically, when the last frame of the video is reached, this code is used to tell the video controller to look at the first frame of the video, and start playing from there.
  + User Interface - createTrackbar
    - This helped me understand how the trackbar was instantiated, and learn more about its parameters.
* Opencv-srf.com
  + Play Video from File or Camera
    - This tutorial guided me on how to play a video from a file.
  + Load & Display Image
* OpenCV 2.4 documentation
  + VideoCapture::set()
    - This helped me understand the parameters of this function. This function is used to loop the video.
  + Changing the contrast and brightness of an image!
    - This tutorial guided me on which functions & equations I would need to use to adjust the brightness or contrast of an image. I simply modified this to also apply to a video.
  + Adding a Trackbar to our applications!
    - This tutorial guided me on implementing a trackbar.
  + User Interface - createTrackbar
    - This helped me understand how the trackbar was instantiated, and learn more about its parameters.

## Output Images

  
Figure 1: Initial

  
Figure 2: Half brightness

  
Figure 3: Full brightness

**References**

[1] “Reading and Writing Images and Video¶.” Reading and Writing Images and Video - OpenCV 2.4.6.0 documentation, docs.opencv.org/2.4.6/modules/highgui/doc/reading\_and\_writing\_images\_and\_video.html?highlight=cvqueryframe#videocapture-set.

[2] “OpenCV: how to restart a video when it finishes?” Stack Overflow, stackoverflow.com/questions/10057234/opencv-how-to-restart-a-video-when-it-finishes.

[3] OpenCV: cv::VideoCapture Class Reference, docs.opencv.org/3.1.0/d8/dfe/classcv\_1\_1VideoCapture.html#a8c6d8c2d37505b5ca61ffd4bb54e9a7c.

[4] “User Interface¶.” User Interface - OpenCV 3.0.0-Dev documentation, docs.opencv.org/3.0-beta/modules/highgui/doc/user\_interface.html.

[5] Fernando, Shermal Ruwantha. “OpenCV Tutorial C .” Play Video from File or Camera, [www.opencv-srf.com/2017/12/play-video-from-file-or-camera.html](http://www.opencv-srf.com/2017/12/play-video-from-file-or-camera.html).

[6] “Changing the contrast and brightness of an image!¶.” Changing the contrast and brightness of an image! - OpenCV 2.4.13.5 documentation, docs.opencv.org/2.4/doc/tutorials/core/basic\_linear\_transform/basic\_linear\_transform.html#basic-linear-transform.

[7] “Adding a Trackbar to our applications!¶.” Adding a Trackbar to our applications! - OpenCV 2.4.13.5 documentation, docs.opencv.org/2.4/doc/tutorials/highgui/trackbar/trackbar.html.

[8] “User Interface¶.” User Interface - OpenCV 2.4.13.5 documentation, docs.opencv.org/2.4/modules/highgui/doc/user\_interface.html?highlight=createtrackbar#createtrackbar.

[9] Fernando, Shermal Ruwantha. “OpenCV Tutorial C .” Load & Display Image, www.opencv-srf.com/2017/11/load-and-display-image.html.