**Project Proposal**

For my project I will be creating a bubble sheet/multiple choice answer sheet scanner with OpenCV. The problem I will be solving is the excess effort and time needed to manually grade an answer sheet. I plan to automate this process to allow for easy, accurate grading by just taking a picture of the answer sheet through a webcam on a laptop. For this project I will be using OpenCv and Tkinter.

**Product Description**

At the main menu the user will get to choose between two choices: scanning test or grading tests. For scanning tests, the user just has to hold up the page to the webcam and press spacebar to capture a picture of test, and enter the filename he/she would like to save file as. For grading tests, the user must take a picture of the answer sheet using the webcam so the program knows what the answers for the test should be. After submitting the answer sheet, the user will then hold up the test to be graded by holding up the test to the webcam and pressing spacebar to take a picture. Using OpenCV, the Python program will then examine the test and store student’s responses in a dictionary. The teacher then has the option of pressing spacebar to grade another test or press “x” to indicate that all tests have already been graded. Once teacher presses “x,” the program will grade each of the submitted tests and display the test results for each test to the user via Tkinter. The summary report will display the choices for each problem highlighting the correct answer in green and highlighting any student’s incorrect answer in red. Next to each problem it will also be indicated whether the student answered it incorrectly or omitted the question. The score/percentage of the test is also calculated and displayed.

**Algorithm Description**

One of the main functions used is the four\_point\_transformation function. For scanning tests it is used to find the edges of the paper, remove any excess background around the answer sheet and straighten out the answer sheet so it is completely vertical. For grading tests, it is used to find the edges of the box around the multiple choice options, remove any excess background around the box and straighten out the box so it is completely vertical. How this algorithm works is first the image will be analyzed to find the four edges of the desired four-sided object. Using these four edges, the points of intersection will be calculated in order to determine the location of the four corners of the answer sheet. Then the width and height of the paper will be calculated using the four found points. Finally, with the four points and dimensions of the paper, a four point transformation of the sheet will be applied cropping the image so it only includes the answer sheet in a vertical position.

The circle detection algorithm will detect all of the multiple choice circles on the answer sheet, determine which of these circles are in the same row, and then compare the darkness of the circles to determine which circle has been filled in. In order to determine which bubble is filled in, first a threshold is applied to the image converting all pixels in image to either black or white. Then I created a white\_pixel\_density function which finds the density of white pixels of the given circle and used this function to compare the densities of circles in the same row. Once the user’s answers have been collected from the answer sheet, his/her answers will be compared to the real answers which the user had initially inputted. Finally a summary report of the answers will be displayed along with the score/percentage of the test.

**Conclusion**

Overall, this program will enable teachers to grade tests through image recognition while also giving them the option the save the tests as images in case they need to take another look at a student’s test in the future.