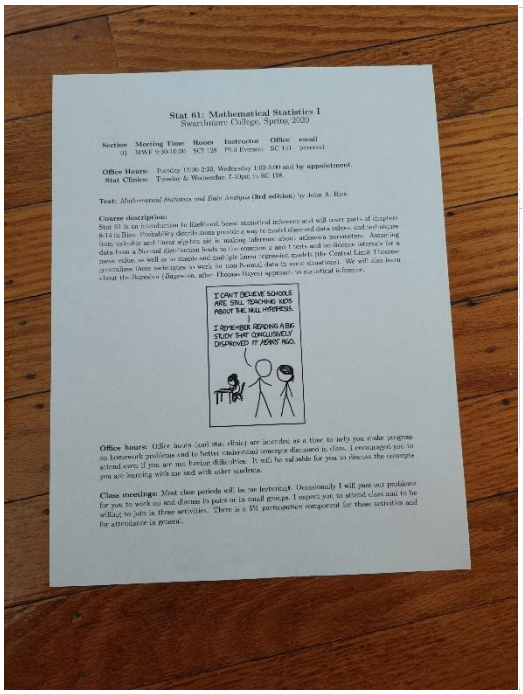
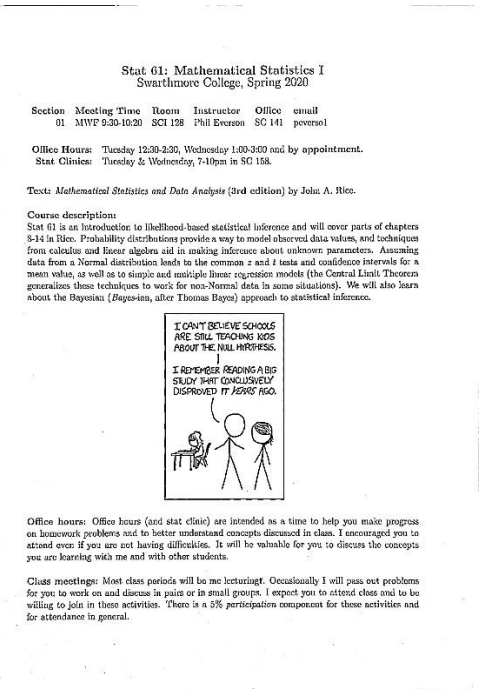
CS 72 Final Project Report

Xihao Luo

For the final project, I built a simple PDF scanner that turns a picture of a document into a PDF-like image (similar to this tutorial). For example,





Stat 61: Mathematical Statistics I
Swarthmore College, Spring 2020

Section	Meeting Time	Room	Instructor	Office	email
01	MWF 9:30-10:20	SCI 128	Phil Everson	SC 141	peverso1

Office Hours: Tuesday 12:30-2:30, Wednesday 1:00-3:00 and by appointment.
Stat Clinics: Tuesday & Wednesday, 7-10pm in SC 158.

Text: *Mathematical Statistics and Data Analysis* (**3rd** edition) by John A. Rice.

Course description:
Stat 61 is an introduction to likelihood-based statistical inference and will cover parts of chapters 8-14 in Rice. Probability distributions provide a way to model observed data values, and techniques from calculus and linear algebra aid in making inference about unknown parameters. Assuming data from a Normal distribution leads to the common *z* and *t* tests and confidence intervals for a mean value, as well as to simple and multiple linear regression models (the Central Limit Theorem generalizes these techniques to work for non-Normal data in some situations). We will also learn about the Bayesian (*Bayes*-ian, after Thomas Bayes) approach to statistical inference.

Office hours: Office hours (and stat clinic) are intended as a time to help you make progress on homework problems and to better understand concepts discussed in class. I encourage you to attend even if you are not having difficulties. It will be valuable for you to discuss the concepts you are learning with me and with other students.

Class meetings: Most class periods will be me lecturing[. Occasionally I will pass out problems for you to work on and discuss in pairs or in small groups. I expect you to attend class and to be willing to join in these activities. There is a 5% *participation* component for these activities and for attendance in general.

In terms of workflow, I first perform Canny Edge Detection on the Gaussian Blurred image to detect all the edges. Then, I used OpenCV's findContours() function to find all the contours in the image. Since we are working with pictures of documents, we can safely assume that:

1. Users will try to focus on the document, so the document contour will have the largest area.
2. Document contour will be a closed rectangle with 4 corner points.

Thus, to find the coordinates of the 4 corner points of the document, the code examines each contour starting at the one with the largest contour area. For each contour, we then calculate the perimeter and estimate a rectangle around the contour allowing 1% error from the actual perimeter of the contour.

After finding the contour points, we will order the points by their position relative to the document (top left, top right, bottom left, bottom right) and perform perspective transform on them. At the end, we will perform adaptive threshold with a block size of 13 and constant 8. The block size and constant are chosen from trial and error to improve image quality.

Revisiting the initial project goals, I think I have met all the basics. However, there are still many limitations to my approach and a lot more work is needed to make the program more robust. For example, although Canny edge detection returns edges nicely, sometimes it is difficult to find the correct contour for the document. (For example, when the edges for some reason do not form a closed shape.) Adaptive thresholding is also tuned through trial and error with experimenting different block sizes and constants. The program also does not account for different color scans. If I could work on this project for another two weeks, I would expect to make the program more robust by accounting for different color documents.