CENG471 Introduction to Image Processing
Fall 2020-2021
Lecturer: Dr. Duygu Sarıkaya
Teaching Assistant: Emre Doruk
Gazi University, Department of Computer Engineering
Assignment 1 due on 30th of November, Monday 23:59

Background Information:

Sergei Mikhailovich Prokudin-Gorskii (1863-1944) [1] a Russian chemist and photographer, is considered as a Pioneer of color photography. Convinced, as early as 1907, that color photography was the wave of the future, with Tzar's special permission, he traveled across the Russian Empire and took color photographs of people, buildings, landscapes, railroads, bridges... thousands of color pictures! His idea was simple: record three exposures of every scene onto a glass plate using a red, a green, and a blue filter. He even envisioned special projectors to be installed in "multimedia" classrooms all across Russia where the children would be able to learn about their country. These negative plates is now in possession by the Library of Congress of U.S.A., and have been recently digitized. They are available on-line at

[2] (You will use the low-resolution images only, see images folder to find the images you will use for the assignment).

References

[1] http://en.wikipedia.org/wiki/Prokudin-Gorskii

[2] http://www.loc.gov/exhibits/empire/gorskii.html

Overview

The goal of this lab is to take the digitized Prokudin-Gorskii glass plate images and, using image processing techniques, automatically produce a color image with as few visual artifacts as possible. There are a few steps in this process:

1. Split the image into three color channel images

2. Align the images to each other

3. Composite the channels into a single RGB image.

1. Divide the input image into three equal parts corresponding to RGB channels.
- The color channel order from top to bottom is BGR, not the typical RGB.
- Assume that the negatives are evenly divided into 3 plates (i.e., each plate is in exactly 1/3 of the negative).
2. Align the second and the third parts (G and R channels) to the first one (B channel). The easiest way to align the parts is to exhaustively search over a window of possible displacements (say [-15,15] pixels), score each one using some image matching metric, and take the displacement with the best score. There is a number of possible metrics that one could use to score how well the images match. The simplest one is just the L2 norm also known as the Sum of Squared Differences (SSD) distance which is simply sum(sum((image1-image2).^2)) where the sum is taken over the pixel values. Another is normalized cross-correlation (NCC), which is simply a dot product between two normalized vectors: (image1./||image1|| and image2./||image2||).
    - For each image, you will need to print the (x,y) displacement vector that was used to align the parts (for both G and R channels that were aligned to the B channel)
    - For a better matching score, you may ignore the image borders and use only the internal pixels, meaning you can crop the images first.

Submission:

You will submit a jupyter notebook (ipynb file) with executable Python script and a short report. Please do not forget to add comments at the top of the related section(comment line). You will write Python scripts, and you will use the libraries we covered in class (opencv, numpy, matplotlib, scikit-image). You should import all the libraries you will use at the top of your notebook. Please refer to course slides, tutorials and practicals to set up a running Python environment, Jupyter notebook and to import these libraries. You can check the documentation of each library (available online) to get more information about the functions you will use.

Your report should contain;

- a brief overview of the problem in your own words,

- the details of your approach,
- the results of your algorithm on at least 8 negative images with your comments and show the results of all of the main steps (unaligned result, result after alignment, what you did to improve results)
- If your algorithm failed to give a satisfactory result on a particular image, provide a brief explanation of the reason(s).

Please, give all references that you used.

Important Note:

This is an individual assignment, meaning that you will be working on it alone (please check the Class Rules and Expectations below, also available in the syllabus)

Grading

The assignment will be graded out of 100: You will receive full points only when your script 1)executes, 2)gives the correct answer, and when 3)the explainations are provided.

Course Rules and Expectations

All work on programming assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, however, everything that is turned in for each assignment must be your own work. In particular, it is not acceptable to: submit another person's assignment as your own work (in part or in its entirety), get someone else to do all or a part of the work for you, submit a previous work that was done for another course in its entirety (self- plagiarism), submit material found on the web as is etc. These acts are in violation of academic integrity (plagiarism), and these incidents will not be tolerated. Homeworks, programming assignments, exams and projects are subject to Turnitin (https://www.turnitin.com/) checks.