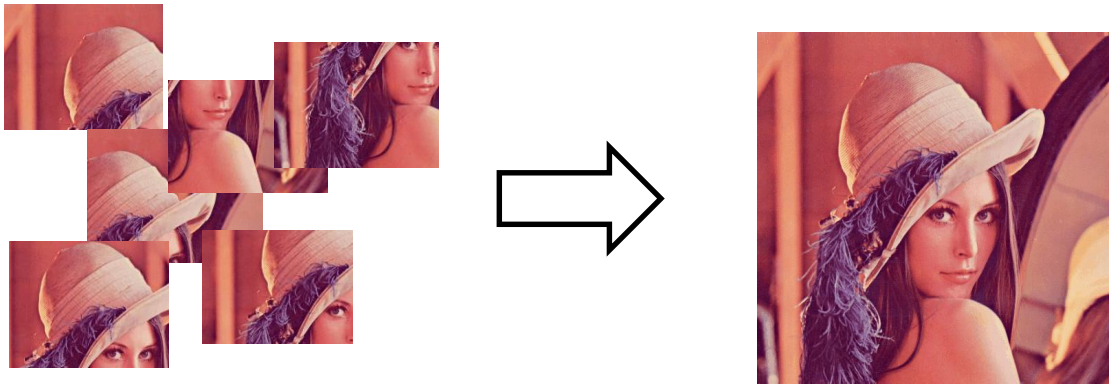
ELEC 474 Machine Vision Take Home Exam

Autostitch: Matching and Merging an Unordered Collection of Images

Due: Friday November 29th 2019, 11:59 pm



Introduction

The task is to create a panoramic image from a series of overlapping images of a scene. This task is divided into four steps. The first step is to extract features from the images, and automatically establish feature correspondences between image pairs. The second step is to use these correspondences to estimate transformations between each image, and establish the most likely transformations between the image pairs. The third step is to apply these transformations to compose a single composite image from all images. The final step is to document your solution and submit it to OnQ by the deadline.

Step 1: Match Features

The starting point is an unordered set of *N* images. The objective of this step is to extract features from each image, and establish which (if any) of the *N-1* other contain a sufficient number of good matches as to indicate a partially overlapping scene. You can use any feature that you prefer (e.g. ORB, SIFT, SURF) or some combination thereof.

You will need to develop a match metric that can be used to discriminate between good and bad image matches. For example, this match metric could take the number of matching features between images into account, as well as the quality of their match. Keep in mind that each image will only overlap with a small number of other images in the set, so ideally this match metric will score high for the images that do have overlap, and low otherwise.

Step 2: Transformation Estimation

For those image pairs that have a high enough match metric score in Step 1, calculate the transformation between them.

Step 3: Image Composition

Apply the transformations from Step 2, and create and store a single composite image from the *N* images in the set. Apply both geometric and radiometric transformations, so that the resulting composite image appears relatively seamless.

Step 4: Documentation

Write a brief (no more than 5 page) report describing your solution and results. Include the following information:

- Your name and your student number. If you're working in a pair, include both partners' names and student numbers. (The order that they appear does not matter.)

- An inventory of the code that you developed (i.e. a list of all source modules). Indicate which of the submitted code modules are original, and which (if any) are not.

- A declaration of originality, indicating the extent that you are the author of the submitted solution. For example, if you are working alone, you might declare that you developed the solution and wrote 100% of the submitted source code. Alternately, if you're working in a pair, then you might declare that the two partners jointly developed the code, with partner A focussing on step 1 and partner B focussing on step 2, (e.g. a 75%/25% split).

- A description of the feature extraction method and matching metric used in Step 1. Describe any specialized data structures and algorithms that you developed and applied.

- Details of the transformation estimation method developed in Step 2.

- A description of the transformation application and image composition method developed in Step 3. Include details of any geometric or radiometric transformations applied.

- Any further relevant details of the developed software (optimization methods, data types of interest, etc.).

- A description of the tests that you executed and the results that you obtained. Indicate the time performance of the method (e.g. how long did it take to execute for each tested image set)?

- A discussion of the correctness and effectiveness of your solution. Include here a declaration of the success of your system, i.e. the extent that you considered your solution to solve the stated task. Also include any limitations that you observed.

- A proposal on how you would improve your solution, if you were given more time.

Deliverables

Your deliverable should include the following material:

D1 – Complete development directory, including all source code. This should be complete, so that it could be easily rebuilt and executed on a mirror system (e.g. the OpenCV system used in the lab environment).

D2 – A listing of the matching metric values between each image, for each data set.

D3 – Resulting composite images (as indicated in the directory structure above).

D4 - Documentation, as indicated in Step 4 above, in pdf format. Include your student names and numbers on the first page.

D5 – Self Assessment Spreadsheet.

D6 (Optional) – Any additional material that you think is relevant. An example could be any extra image stitching samples that you tested.

Submit the material through OnQ, in a single zip file (described below). If you are working in a pair, then only make one submission – just make sure that both of your names and student numbers are indicated on the report title page.

Please submit the deliverables as a single zip file named ***autostich-<yourLastname(s)>.zip*** with the following folder structure:

```
autostich-<yourLastname(s)>.zip
|-- D1/Code/
|        |--- <yourCode>.cpp
|        |--- <yourCode>.h
|        |--- <etc..>
|
|-- D2/MatchingMetrics.pdf
|-- D3/Results/
|         |---<Resulting Composite Image 1>.jpg
|         |---<Resulting Composite Image 2>.jpg
|         |--- <etc..>
|
|-- D4/autostich_Report-<yourLastname(s)>.pdf
|
|-- D5/autostich_SelfAssessment-<yourLastname(s)>.xlsx
|
|-- D6/Extra/
```

The **Extra/** directory can be used to contain any additional material that you feel is relevant to your assignment. For example, if you acquire new image sets for testing, you can include them here.

Marking Scheme

Step 1:	3 marks
Step 2:	3 marks
Step 3:	4 marks
Documentation:	5 marks
TOTAL:	15 marks

Late submission policy

Late submissions are accepted, at a deduction of 5 marks per week. The following table illustrated the late-submission deduction policy:

Submission Period	Mark Deduction
up to 11:59 pm, 29th November	0
12:00 am 30th November to 11:59 pm 6th December	5
12:00 am 7th December to 11:59 pm 13th December	10
After 12:00 am 14th December	15

Exam Rules

- Do not cheat.

- You can work individually, or in groups of 2. Groups larger than 2 are not allowed.

- This is an exam, and is to be executed independently. You therefore cannot discuss anything about the exam, except with your partner. In particular, do not discuss solution methods or share code with other people or groups.

- All code that you submit should be original. Do not mine the internet for solutions. The coding task is relatively straightforward, and it will be easier (and of greater benefit) to develop the solution from scratch (as is required) than to start from another solution (which is not allowed). You are allowed to use source code that you developed as part of a previous lab solution. The incorporation of any other source code is not allowed.

- Questions about the exam will be entertained during lectures in the final week of class, in the presence of the entire class. If you have any questions, then please ask them during these lectures, when everyone is present. For fairness, neither the instructors nor the TAs will answer questions about the exam, in person, by email, or outside of lectures.

- Submit your complete solution (described above under *Deliverables*) through OnQ, by the deadline. The system will not accept submissions after the late submission deadline.

GOOD LUCK!