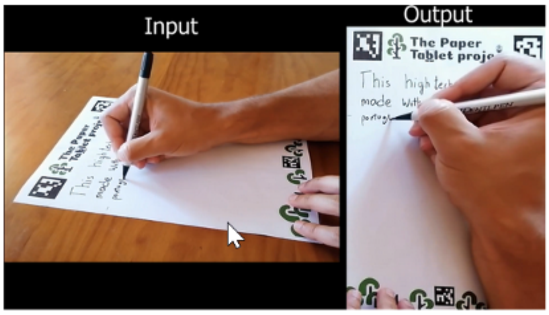
PIV PROJECT 2022

PIV's Smart Board



Synopsis

In this project you will implement a system that should accomplish two tasks:

- Given a template (one image) of a paper sheet, the system must detect the template and correct its shape in a stream of images taken from a camera. The camera can be static or can be moving around and the paper as well .
- Given a sequence of images of a board-like surface, the system must combine all images into a single panoramic image.

The code you must develop accepts the images as input and a set of keypoints (coordinates+descriptors) detected in the image. For the tablet application, the output should be the transformation to the template image and the rendered image in the new perspective. Similarly for the board the input is the same and the output comprises the transformations and the final mosaic

The two applications are in fact the same techniques with a subtle difference. The main blocks of processing that you should develop are the following (recommended only!):

1. Extract keypoints (SIFT, ORB, SURF, whatever suits you)
2. Match corresponding keypoints between template and images
3. Align correspondent points using RANSAC or other outlier-removal method.
4. Compute the transformation between each image and the template. You should select the adequate transformation and verify all conditions and possible errors (not enough points, too large error, etc)

5. Render the final transformed image

Except for steps 1 and 5, you must develop the algorithms “from scratch”, that is, you are not allowed to use libraries (Matlab/Octave Toolboxes, OpencCV, …) to implement these tasks.

If you are using Python, SciPy+Numpy should be enough (crash course on NumPy http://cs231n.github.io/python-numpy-tutorial/ . If you use Matlab/Octave the main engine contains all the operations you need.

Recommended Approach

As mentioned in classes, this pipeline is trivially built using opencv. In fact looks like there are already tools for this and the two instructions

```
stitchy = cv2.Stitcher.create()
(dummy,output)=stitchy.stitch(imgs)
```

would do the job ! Don’t use this but as a first stage of development use the toolboxes/libraries to develop parts of your project. In other words, implement the whole pipeline and then steadily replace each block with your own code.

Creativity vs Resilience

Of course, we do not require your code to be professional-grade, however engineering requires both practical “attitude” and critical thinking. Underlying all this there is the old saying “there is nothing more practical than a good theory”, so: we accept code with bugs here and there but, blaming on the “bug” the crash due to the inversion of a matrix with null determinant won’t be accepted :-).

On the other hand, time is scarce and you may want to do a lot of stuff that makes your project shine (e.g. detect hands, segment the background …) at the expense of better testing. Both options are legitimate here, it’s your call : some people can not handle imperfection and are always systematic, others surprise us with new things or came up with an idea we did not ask or anticipate but … implementation crashes more than usual. This is a school and with 7 weeks to do something you’ll learn only in the 4th week both approaches are ok.

Grading

If you do the project as stated above, nothing to fear you'll be successful and have a good grade. Top grades (18-19-20) require excellence in the mastery of the techniques, applying knowledge from other areas to solving a computer vision problem (that's what CV is all about) or going beyond the strict pipeline 1-5 and apply other techniques (suitable to the problem).

In conclusion: the project is worth 35% of your final grade and as you realize evaluating people's work is more complex than tick-in-the-list ! We teach engineers (learn, think and decide) so, your decisions, your thinking and the way you did it counts.

Finally, check here https://github.com/pjreddie/uwimg/tree/main/src/hw3 in the Univ. of Washington they do a similar thing … as homework (in the 3rd week) ! So … get your project running asap so that we can go one step further !

Project evaluation :

In the 7th week you will submit your code and one paragraph with notes you wish to let us know. In the 8th week you'll run it for us and if it does all it is required and you answer properly to clarifying questions … you get a good grade. Top projects may have an additional session where few groups will explain in more detail what they did.

Note that your code will run in a computer with basic matlab and python with scipy and numpy only !

Datasets

We already posted on google drive datasets but we recommend you build your own: create a template, get your cellphone and record a video around the template while somebody writes something (or just occludes part of the template). Anyway we will be posting new datasets as we go along. You are welcome to share your datasets with your colleagues.

Running Code

In this github repository https://github.com/andrejfsantos4/Paper-Tablet you have the full pipeline implemented with docker containers (which you should use to develop your code). In fact, if we have enough time, that is how we will test your code: insert it in a docker that receives images and outputs transforms !

Project Submission

We will post the form for submission with a standardized function call.