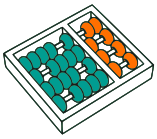

MC949/MO446 — Introduction to Computer Vision
INSTITUTO DE COMPUTAÇÃO — UNICAMP
2nd Semester, 2017

UNICAMP

Final Project Proposal

Prof. Adín Ramírez Rivera
adin@ic.unicamp.br

1 Description

The final project for the class might be the implementation and evaluation of an algorithm from a research paper (better if it is a recent one). The project's theme is very open-ended; ideally you can incorporate your own research agenda in the project. However, if you go this route, you need to use results that you will develop for the project and not just present previous results as part of the course work.

2 Project Proposal

The proposal should be a PDF with a title for the project, your names, and two to three paragraphs describing the project idea. **No more than one page!**

Your proposal should include one to three relevant papers that deal with your problem. If you are going to reproduce a current paper, you can check the relevant references there. If you are going to do something mixed, then cite the related papers that inspired you.

If you will be collecting your own data (images or video), describe any special experimental procedure (restrictions in illumination, motion, cameras, etc.). If you are not, you can get data from the author's website, or contact them to get the original data from them. In either case, explain where you will get your data from.

3 Poster Guidelines

- Posters should be A0 or A1 (landscape preferred)[1].
- Use high resolution or vector graphics. Remember, the poster will be much larger than the size of your computer screen.
- Use large enough fonts. You can print subsections of your poster out on standard computer paper to get an idea of how large your font is after printing.
- White space is your friend! Your goal is to convey information in a way that someone else can digest, not just throw results onto a slide. Less is usually better, and margins help the viewer organize and process your poster.
- A good rule of thumb for poster content is: 50% images or graphics; 30% text; and 20% whitespace. Full sentences should be rare; paragraphs should be extremely rare.
- Use only 2–3 fonts, and use them consistently (one for headings, one for text, one for captions). Keep formatting consistent between sections.
- Use only 2–3 main colors, and keep them consistent. There are many websites that will help you choose color schemes that look good (a quick Google search will get you there). *Red on blue never looks good*; there are many other rules of thumb for colors you can avoid by picking a standard color scheme.
- Practice explaining your poster to someone. Reformat and add images where you find yourself struggling to explain.
- For extra guidelines, check http://www.personal.psu.edu/drs18/postershow/.

[1] I'm trying to find a way of printing the posters. If we don't get funding or a printer, you will need to print it as a set of A4 papers in a regular printer and join them together with tape ☹.

4 Project Ideas

You are strongly encouraged to find a topic that is aligned or related to your research topics. In case you want to find something new or interesting, check the best conferences on Computer Vision:

- CVPR: http://openaccess.thecvf.com/CVPR2017.py
- ICCV: http://openaccess.thecvf.com/ICCV2015.py
- ECCV: http://www.eccv2016.org/proceedings/

You can also find inspiration from past conferences, check http://openaccess.thecvf.com/ for a list of open access versions of the conferences' papers.

5 Final Report

Your final report should be written like a paper (you must use the CVPR format http://cvpr2017.thecvf.com/files/cvpr2017AuthorKit.zip), **with a strict page limit of 8 pages including your references**.

You should include a setup for your problem, that is, motivation and related work, as well as your work (methods, experiments, and results). You should focus on what you found and did and did not work as well as the discussion why. You should discuss what you accomplished during the project, what you learned and your application, and evidence of the effort. For the experiments execute your methods several times and produce graphics and plots to summarize your results (check the papers you are taking inspiration from to produce a similar evaluation).

6 Evaluation

Your grade will be defined by the following aspects:

1. Project proposal 10%
2. Poster presentation 30%
3. Final report 60%
   (a) Introduction, motivation, and background 10%
   (b) Related work 10%
   (c) Explanation of your methodology 10%
   (d) Experiments, discussion, and results 30%

Each item corresponds to the questions and requirements defined in the previous sections. **Your English usage won't be graded**, but your ability to present your results, ideas, and how the are supported will be. Each other point will be evaluated according to the completeness and correctness of the requested items.

7 Submission

7.1 Proposal

The submission of the proposal is just a PDF that complies with the instructions of § 2.

7.2 Poster

The poster will be presented in the dates given in the calendar of the course. And you should submit a PDF with your poster in the Moodle.

7.3 Final Report

For the final submission, you need to create a folder named `pf-XX-YY` where `XX` and `YY` are the RAs of the members of your team. Note that the RAs **must** be sorted.

Your submission must have the following subfolders:

- `input`: a directory containing the input assets (images, videos or other data) supplied with the project. **Store the assets used to produce your experiments here.** For any other asset, setup your `Makefile` to automatically download them from a public server or repository. **Your assets shouldn't be too big.** If you need a lot of them, store them somewhere else and use a script to download them locally.
- `output`: a directory where your application should produce all the generated files (otherwise stated in the problem). This directory should be empty.
- `my-output`: use this directory to show your previous results (in case you want to show intermediary work).
- `src`: a directory containing all your source code. You only need to submit files that are not derived from other files or through compilation. In case some processing is needed, prefer to submit a script that does that instead of submitting the files.
- `Makefile`: a makefile that executes your code through the docker image. An image is already built and available for use (`adnrv/opencv` at the docker hub registry). The code will be executed through a standard call to `make`, so other dependencies must be provided by you under that constraints.
- `report.pdf`: a PDF file that follows the restriction given in § 5.

The principal folder must be zipped into `pf-XX-YY.zip` and submitted through the Moodle website. No other files will be accepted. **Note that upon unzipping your file, the original folder** `pf-XX-YY.zip` **must be created.** That is, do not zip the contents, but rather the folder.

8 Notes

- Note that there are several implementations that you can find in the internet. This project is for **you to implement** the algorithms. Thus, do not submit code from others. And if you re-use code from someone for a non-restricted part, disclose it in your report and code.
- All the submissions must be self contained and must be executable in a Linux environment. Specifically, your code must execute in the docker image `adnrv/opencv`, available at docker hub (`https://hub.docker.com/r/adnrv/opencv/`).
- It is your responsibility to make sure your code compiles and executes correctly. No effort will be made to run your code besides executing `make` inside a docker.
- You must program in Python 3.6 (or higher) or in C/C++ with `gcc` version 6.2.0, using OpenCV 3.2.0. All available within the image. If you need to install other packages you must do so within your `Makefile` as automatic prerequisites.
- Check the `Makefile` provided with the project zero. It will be used to automatically execute your code. You must pass the full path of your project folder: `make SOURCE_DIR=$(realpath ./pA-XX-YY)`.