Computer Vision - Y. Fougerolle
Project Proposal for Stereo Camera System - 2025

yohan.fougerolle@u-bourgogne.fr

1 Project Overview

The goal of this project is to develop a portable and home-made software solution for 3D geometry reconstruction using two USB cameras connected to a standard computer (for the sake of simplicity, we will consider a standard windows laptop, but extra effort to handle Linux and Mac systems will be appreciated). The software will capture two images, process them, and generate a 3D point cloud. The project will be broken down into modular, independently evaluable parts. The expected outcomes of the projects are:

- A simple and ready to use demonstrator of a stereo vision system.
- Documentation and necessary installations needed to run the software
- Final report/synthesis + bullet list/byweekly reports
- **Deadline** - Submission of the final report and source files : Friday, April the 4th

1.1 Project Parts and Evaluation Criteria

The following sections only state as a simple proposal and a guideline for the evaluation. Feel free to add/merge any part you want if you deem it necessary. Simply keep in mind that each task should be addressed by at least one proposed solution, and that all the results should be criticized (when are the results satisfactory, or not, and why? What are the limitations of the proposed approach, etc.).

1.1.1 1. Camera Calibration (20%)

Description: Develop a module to calibrate the cameras and obtain intrinsic and extrinsic parameters. **Deliverable:** Code for calibration and a report detailing calibration accuracy. **Evaluation:**

- Correct implementation of intrinsic and extrinsic parameter estimation.
- Accuracy of the calibration results (e.g., reprojection error).

1.1.2 2. Stereo Rectification (15%)

Description: Implement stereo rectification to align the images. **Deliverable:** Rectified image pairs. **Evaluation:**

- Proper alignment of epipolar lines.
- Efficiency and clarity of the rectification process.

1.1.3 3. Feature Detection and Matching (15%)

Description: Detect and match features between the two images. **Deliverable:** Matched feature pairs displayed visually. **Evaluation:**

- Number of correct matches.
- Robustness against noise and distortions.

1.1.4 4. Stereo Geometry Estimation (20%)

Description: Compute the essential matrix and recover camera poses. **Deliverable:** Correctly estimated essential matrix and camera pose. **Evaluation:**

- Accuracy of essential matrix computation.
- Correct decomposition into rotation and translation.

1.1.5 5. Triangulation and 3D Reconstruction (20%)

Description: Implement triangulation to compute 3D points. **Deliverable:** Sparse 3D point cloud. **Evaluation:**

- Accuracy of the reconstructed points.
- Consistency with input images.

1.1.6 6. Point Cloud Post-Processing (10%)

Description: Refine and filter the generated point cloud. **Deliverable:** Enhanced 3D point cloud. **Evaluation:**

- Quality of noise removal and smoothing.
- Visual and quantitative improvement of the point cloud.

1.2 Evaluation Overview

The project will be graded based on the following criteria:

- Correctness and robustness of the implementation (60%).
- Documentation, including reports and code comments (20%).
- Presentation and demonstration of results (20%).

1.3 Final words

The project can be done by **groups of 3 students at most**, and the last member will be myself (Y. Fougerolle - to follow what is going on in each group). **Each "Labs" will be dedicated to your project**, implementation, follow-up, reporting, etc. and **each "class" to more fundamental aspects**: prepare your questions before attending them (use Teams to share them). On a more practical aspect, **pairs of cameras are available in the Research Plateau** (not sure it is the correct name): you can use them on site, but **you are not allowed to "borrow" them**. **All the Labs (2 hour cessions with smaller groups) will take place in the Robotics Plateau** (still not sure this is the correct name), basically the big room, at floor 0 in Technopole so that we will not waste time going back and forth to Condorcet.

Please, keep in mind that, at Master level, **you are expected to have a critical understanding of your solution**. In other words, you have to be able to prove that your production is not only ChatGpt-generated and that you understand what you write in your report as well as your source code. Try to go as far as you can, in the smoothest manner (rather many small steps than a huge effort).

You are free to choose your entire environment (OS, IDE, programming language) as long as you provide a clear and simple protocol/tutorial to install the necessary components for your software. However, I strongly recommend you to stick to common tools (Matlab or C++ or Python, OpenCV). As a matter of taste, I prefer C++ with OpenCV, then Matlab. I strongly recommend you to use Overleaf (overleaf.com) to edit your byweekly and final reports so that all the members of your group can participate. As the evaluator, I can also quickly see what is going on. For your source code, I also strongly recommend you to use GitHub (https://github.com/), for similar reasons. Regarding the project management and communication tools, feel free to use any tool you deem necessary. In any case, do not forget to invite me to join your group.

Most importantly, learn cool things and enjoy. Good luck!