Mini Project 2 – Camera Calibration

Teams: Teams are current undergraduate or graduate teams. Each team must have a team name. One team member will submit the report and Python code for the entire team.

Writing: All writing uses 1" borders, 11-point Times New Roman font, and single spacing. Writing is required to be typed – not handwritten. Submit as a single PDF. Also submit all Python codes and running instructions in the appendices (see below).

Project Objectives: (1) Local Students: Collect camera calibration data. Distance students: use images collected by local students. The instructor will make all datasets available to the class through a link to cloud storage.
(2) Use OpenCV to calibrate the camera. (3) Evaluate the calibrations. (4) Discuss a recent camera calibration method from the literature.

Instructions:

1. Please review material on the OpenCV websites. *Spend some time here, you will be responsible for basic understanding of what is covered in this project on exams*.

   OpenCV Camera Calibration
   https://docs.opencv.org/4.3.0/dc/dbb/tutorial_py_calibration.html

   OpenCV Camera Calibration and 3D Reconstruction
   https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

   These are also supplied as PDFs on Canvas.

2. Prepare Anaconda for OpenCV.
   a. Open Anaconda.
   b. Go to base environment. If not in (base), use `conda deactivate`.
   c. Create a new environment:

   Note: If you are using a different version of Python, put that number in.
   To check, in anaconda, type

```
python –-version

conda create --name camera_cal numpy python=3.7
conda activate camera_cal
conda install numpy
conda install -c conda-forge opencv
conda install -c conda-forge glob2
```

Note: If you get DLL errors with OpenCV, try this:

```
pip install opencv-contrib-python
```

Download CameraCalibrate.py from Canvas. You will need to fill in a few missing pieces. The chessboard size is (9,6).

3. Read this part carefully!

   There are two datasets.

   Each team member is to download this dataset, unzip it, and run it through the OpenCV camera calibration:

   data.zip

   At least one member of each team must download the class large dataset and run OpenCV camera calibration on it. This is a large dataset so it may take some time.

   data2.zip

   Everyone on the team must calibrate using this dataset.

   From this step, each team will have two calibration results.

   The datasets are large! We will not be using these datasets again in SPAV class, so you can delete them if desired after running the calibrations!

4. There are some small portions of the Python file missing. Use the OpenCV website to determine what to do.

For performance assessment, assess qualitatively (e.g., discuss if distortions are visible…) and quantitatively (what was the error), etc. Also prepare a table summarizing the number of images and calibration results for each dataset.

Report: Write a report using the following format:

- Cover page with names of students and NetId's, the team's name, the assignment "Mini-Project 2", the due date, and the MSU honor code (it is listed in the Syllabus).
- Next page - Table listing each student, their specific contributions, and time spent on project (hours) and total hours for each student. Also, add a column and each student certifies they ran camera calibrations on the small datasets. You can simply place a check in this column.
- Start the rest on a new page.

  Note: Include headers in your report, e.g., "Abstract". Feel free to also put questions in bold or italics and then provide answers.

Part 1 – Camera Calibration with Python Open CV

- Abstract – In one paragraph, summarize the report and calibration results.

- Introduction – Briefly discuss camera calibration. Answer the following questions: (hint: you can put these questions into the report)

  a. What is CV?
  b. What is OpenCV?
  c. What is glob2?
  d. What does the OpenCV camera calibration do?
  e. Why is camera calibration important and essential in automotive autonomy?
  f. Briefly discuss the intrinsic and extrinsic matrices. Insert the matrices (define them). Define each term in them.
     Which matrix (extrinsic or intrinsic) is associated with parameters inside the camera?

- Methodology – Discuss the general procedure for camera calibration using OpenCV. Include your Python code in an appendix. Answer the following questions:
  a. Explain in your own words what each call to cv2.XXX functions do.
  b. Explain the contents of the outputs of the cv2.calibrateCamera() function.
  c. Discuss basics of each dataset. List the number of images etc.

- Results. List results. For example, make a table of results. Show a few example images. For each figure and table, provide a figure or table caption (e.g., Figure 1: Sample images from XXX dataset.) and reference the caption in your text.

  Address these items:

  a. (Only one member must run the large dataset), Run the camera calibration routines on each image dataset. List results for each.
  b. Think of a scenario where bad camera calibration might cause the autonomy code to malfunction or even cause an accident. Describe that briefly.
  c. What does the calibration mean error and root mean square error in each dataset?
     Note: When placing in report, use three significant digits after the decimal point, e.g., 10.123.

   d. Copy and save the two output images for the class datasets into your report. Discuss the differences.

- Conclusions. Draw conclusions. Answer the following questions:
   a. This is the same camera. Discuss the differences in calibration errors and uncropped image results from the two datasets.
   b. What conclusions can be drawn in general from this work? Hint: Think of why camera calibrations are needed in autonomy.
   c. How did the datasets compare in calibration results?

- References – list in IEEE format.
  Hint: For websites, list the website and show the date accessed.
  For examples of IEEE format citations, refer to
  https://ieee-dataport.org/sites/default/files/analysis/27/IEEE%20Citation%20Guidelines.pdf

- Appendix A – Python codes. You can show all code or highlight changes.
- Appendix B – List of commands or actions to run the code. Copy the program output from the runs here. Give the commands/actions to run your code and list the directory that your code resides in.
- Start the rest on a new page.

Part 2 – Paper Review (Graduate teams only)

Each graduate team will find four recent (within the last four years) papers on camera calibration. The papers must have been published in the last four years. Choose one of the papers and have the group provide a one-page summary of one paper. Explain why this paper was chosen. List all the papers in the references. Thus, there is only one summary of one paper for each group.

Discuss (1) what are the main contributions of the paper, (2) give a description in your own words about how it works, (3) briefly discuss any prior work the authors used to develop their system (e.g., they took algorithm X and added Y to enable Z), (4) how did they verify the proposed solution, (5) would this solution be a good candidate for automotive autonomy? Why or why not?

Grading Rubric:

Item	Percent for undergraduates	Percent for graduates
Followed directions & report contains required materials	10	10
Part 1: Answered questions. Results & discussion contained both qualitative and quantitative discussion, tables, figures, etc. Good technical content. All figures and tables referenced.	60	40
Part 2 summary. Followed directions and gave all required material.	-	20
Citation format and adequate citations	10	10
Good grammar and proper technical writing style	10	10
Python code clear and adequately commented	10	10