

## Homework 05: Corner Detection

Kaveh Fathian, Email: [fathian@ariarobotics.com](mailto:fathian@ariarobotics.com)

Handout: 2025-09-29

**Due: 2025-10-06, 11:59pm, on Canvas**

### General Instructions:

- You should solve the homework and submit your report **individually**. Identical submissions will receive a grade of zero.
- Getting help from others or checking your answers with other students (not the TAs) is okay and encouraged.
- Ask any questions on **Ed Discussion** (instead of emailing).
- **Before** the homework due date, TAs are strictly prohibited from **pre-grading** your homework. Do not expect the TAs to help you verify if your answers are correct or give you the problem solution.
- **After** the homework due date, if you do not know how to solve a problem, reach out to the TAs. They will walk you through the solution and help you understand it. Note that homework solutions will **not** be posted because some problems will be used in next year's class.
- **Exams** may contain questions related to homework, so make sure you learn how to solve the homework problems correctly.
- The deliverables are outlined for each problem, and you should carefully **follow the instructions**. Failing to follow instructions will result in **points being subtracted**.
- You will submit a **single PDF** file to Canvas as your homework report. The PDF must contain your **answers** and any requested **outputs** (e.g., printouts, snapshots of code, or GUIs). If requested, follow the instructions specified by the problem to provide your **code** (e.g., in a compressed .zip or .tar file) in addition to the PDF file.
- **Grading:** Each homework in this class will contribute **5pts** to your final grade (there will be 12 homework assignments, each 5pts, leading to 60pts for all assignments). A detailed grading **rubric** will be posted on **Canvas** after the homework due date. Any bonus points will be added to your overall course bonus points, which will be added to your final grade.
- **Late submission:** Late or missed submission will not be accepted and will receive a grade a zero. Any excused absence must be documented and disclosed to the instructor (extensions will be granted on a case-by-case basis). Three or more missed homework lead to an INC grade.

## Homework 05: Corner Detection

Kaveh Fathian, Email: [fathian@ariarobotics.com](mailto:fathian@ariarobotics.com)

Handout: 2025-09-29

**Due: 2025-10-06, 11:59pm, on Canvas**

**EXERCISE 1** (5pts) – The objective of this homework is to understand the Harris corner detection algorithm. Do **NOT** rely on generative AI to code!

### Steps:

1. Download the template Harris corner detection code, called “hw05\_template”, from the course repo: <https://github.com/ariarobotics/cv/tree/main/code>
2. Load your **headshot image** (used in previous assignments). Convert it to the appropriate type, color, or resolution as needed (the image must be a minimum of 100 pixels in each dimension).
3. Complete the missing code in the template function “harris\_corner\_detector” to generate Harris corners. Pay attention to the comments in the code and **what they are asking you to do**:

```
# 1. Compute image derivatives
Ix = # compute horizontal derivative (x-direction)
Iy = # compute vertical derivative (y-direction)

# 2. Compute products of derivatives
Ixx = # Compute using hadamard product
Iyy = # Compute using hadamard product
Ixy = # Compute using hadamard product

# 3. Gaussian filter to smooth the squared derivatives
Sxx = # Gaussian filter of Ixx
Syy = # Gaussian filter of Iyy
Sxy = # Gaussian filter of Ixy

# 4. Compute cornerness
detM = # Must be a function of only Sxx, Syy, or Sxy
traceM = # Must be a function of only Sxx, Syy, or Sxy
C = # C Must be a function of only detM, traceM, and alpha
```

4. Report all deliverables below. Make sure your report is clear, organized, and well-explained.

### Deliverables:

- Snapshot of your entire code. In particular, your implementation for all values in Step 3 should be clearly visible and correct.
- Snapshot of your headshot input image.
- Snapshot of the cornerness image output.
- Snapshot of the detected Harris corners overlaid on the input image.

