

## Bonus Homework 02: Perspective Correction

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

Handout: 2025-11-19

**Due:** 2025-12-01, 11:59pm, on Canvas

### General Instructions:

- You should solve this homework and submit your report **individually**. Identical submissions will receive a grade of zero.
- For bonus assignments, getting help from other students is okay and encouraged, but the **TAs** are **not** required to help you.
- Ask any questions on **Ed Discussion** (instead of emailing).
- The deliverables are outlined for each problem, and you should carefully **follow the instructions**. Failing to follow instructions will result in **points being subtracted**.
- Follow the instructions specified by each problem to submit the homework. Upload your **code** or other deliverables as instructed.
- **Late submission:** Late or missed submission will not be accepted. Any excused absence must be documented and disclosed to the instructor (extensions will be granted on a case-by-case basis).

## Bonus Homework 02: Perspective Correction

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

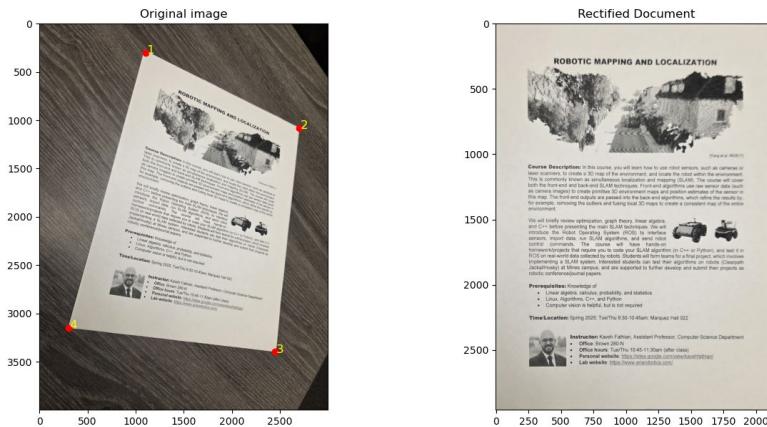
Handout: 2025-11-19

Due: 2025-12-01, 11:59pm, on Canvas

**Bonus Problem** (1pts) – In this problem you will continue building the second stage of a **document scanner** that can rectify a scanned document given its corners (which you detected previously). Use the template code “bhw02\_template.ipynb” provided in the course repo (<https://github.com/ariarobotics/cv/tree/main/code>), and complete the function “def rectify\_document(img\_bgr, corners)” in the code to rectify the image “input.jpg” in the data folder. You **must** use/compute the homography matrix for rectification. Display the resulting **rectified image**, and the **homography matrix**.

### Deliverables:

- A single PDF file that has the screenshot of your entire Python/Jupyter script. We will look at your implementation of the rectify\_document function.
- Display of the computed homography matrix
- Display of the rectified image



### Notes:

- You are encouraged to research online for solution ideas, algorithmic approaches, or for example codes that can help you solve the problem.
- You are free to decide how to implement your algorithm (there is not a single required method), however, you **must** compute and use the homography matrix for transformation. Any pre-built or high-level document rectifier that does not use the **homography** matrix is not allowed.
- You do **not** need to submit code, only a single PDF file as instructed above.
- You are expected to debug and solve any implementation issues that arise during development. This is part of the learning process for this problem.