

Computer Vision Assignment: 1

AI 728

31 Jan 2025

1 Line Detection using Hough Transform

1.1 Objective

The goal of this assignment is to implement a classical computer vision pipeline for detecting and counting straight lines using the Hough Transform. Each student will work with a *custom, deterministic dataset* generated from their roll number, ensuring reproducibility and fairness in evaluation.

Students are required to analyze three document scenarios of increasing difficulty and report their observations, including a comparison with a foundation model.

1.2 Provided Code and Setup

Students are provided with a file `playground.py`, shown below, which serves as the evaluation entry point. You can use this file as playground.

```
from data_generator import *
from hough_lines import *

#define my_roll_number and implement count_hough_lines in hough_lines.py

def test():
    roll = my_roll_number()
    name = "Biplab"

    img1 = blank_ruled_paper(roll)
    img2 = printed_paper(roll, name, roll)
    img3 = handwritten_paper(roll, name, roll)

    # Code to visualize the generated image.
    cv2.imshow("image", img1)
    cv2.waitKey(0)
    cv2.imshow("image", img2)
    cv2.waitKey(0)
    cv2.imshow("image", img3)
    cv2.waitKey(0)

    print("Blank ruled lines:", count_hough_lines(img1))
    print("Printed lines:", count_hough_lines(img2))
    print("Handwritten lines:", count_hough_lines(img3))

if __name__ == "__main__":
    test()
```

1.3 Your Tasks

You must complete the following tasks:

1. Define a function `my_roll_number()` in `hough_lines.py` that returns your official roll number as a string.
2. Implement the function `count_hough_lines(img)` in `hough_lines.py`.
3. Use the Hough Transform to count the number of **horizontal lines** present in the input image.

1.4 Test Scenarios

Your implementation will be evaluated on three types of images generated using your roll number as a random seed:

1. **Blank Ruled Paper:** Contains evenly spaced horizontal ruling lines.
2. **Printed Paper:** Contains at least 20 horizontal lines along with printed text showing the student name and roll number.
3. **Handwritten Paper:** Contains at least 20 noisy, handwritten-style horizontal lines with jitter and varying thickness, along with the student name and roll number.

Each image is generated deterministically from your roll number. Using a different roll number will produce a different dataset.

1.5 Important Constraints

- You must **not hardcode** the expected number of lines.
- Your solution must generalize to unseen test data generated with a hidden seed.
- Only horizontal lines should be counted; vertical or diagonal lines must be ignored.
- You may use OpenCV or NumPy, but deep learning models are not allowed for line detection.

1.6 Evaluation and Verification

Your submitted code will be automatically evaluated on a **new, hidden dataset** generated using a different seed. The evaluation will measure:

- Correctness of line counting
- Robustness to noise and fragmentation
- Consistency across all three scenarios

1.7 Foundation Model Comparison

In your report, you must compare your Hough-based line counts with the output of a foundation model (e.g., a vision-language model) asked to count the number of lines in the same images. Discuss differences in robustness, failure cases, and interpretability.

1.8 Submission Instructions and Folder Structure

Each student must submit their assignment as a single folder named using their roll number. The roll number is treated as a **string** throughout the codebase.

The required folder structure is shown below:

```
PH2022502/ <-- THis should be your roll number
|-- playground.py
|-- hough_lines.py
|-- data_generator.py
|-- report.pdf
```

Important Notes:

- The folder name must exactly match your roll number (e.g., PH2022502).
- The file names must remain **unchanged**. Submissions with renamed files will not be evaluated.
- You are allowed to modify:
 - `hough_lines.py` (mandatory)
 - `playground.py`
 - `data_generator.py`

Roll Number Specification You must define your roll number as a **string** by implementing the following function in `hough_lines.py`:

```
def my_roll_number():
    return "PH2022502"
```

The roll number string is used internally as a seed to generate a deterministic test dataset.

Report Submission You must include a file named `report.pdf` in the same folder. The report should describe:

- Your line detection approach
- Hough Transform parameter choices
- Results across all three scenarios
- Comparison with a foundation model
- Observed failure cases and limitations

Autograding Policy During evaluation:

- Your submission will be tested on **new, unseen data** generated using a hidden roll number string.
- Hardcoded assumptions about line count, spacing, or image size will result in zero marks for performance.
- The evaluation script will import and execute your code assuming the exact folder and file structure shown above.

1.9 Grading Rubric

Component	Marks
Coding and completion of all functional modules	4
Report (methodology and analysis)	3
Performance on hidden test data	5
Total	12