CS GY 6643 - Computer Vision, Fall 2024

Project 2

Due: 2024/11/14 11:59 PM

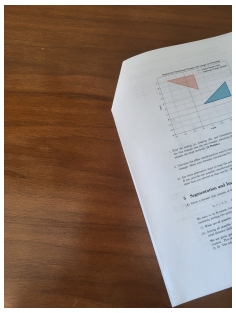
Note: For the project you are not allowed to use pre-built libraries to implement the core algorithm of the problem. You can use pre-built libraries to load/process the images
You have to submit a final jupyter notebook and a pdf report as a submission. Your pdf report should be informative enough on its own. We will use your jupyter notebook file to check your implementation of the code. Use markdown cells to write text explanation wherever needed. Upload the jupyter notebook to google drive and add the sharable link to brightspace.

---

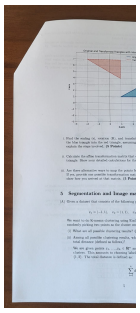
1 Image Transformation and Stitching

After the midterms, the teaching assistants (TAs) were occupied with scanning the submitted answer sheets. In their rush to attend their next class, they mistakenly captured only partial images of each sheet. Below are the three photos they took in haste.

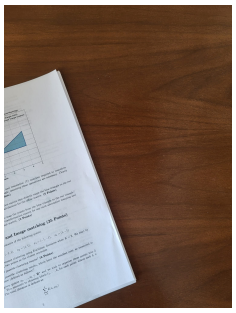
Help the TAs rectify this mistake by combining these images to produce a complete, rectangular image that displays all parts of the answer sheet, resembling an A4 sheet as closely as possible.



(a) Image 1



(b) Image 2



(c) Image 3

Figure 1: Scanned image sections

Suggested Approach

- **Feature Identification**: Determine the location of corresponding features between the images, which will facilitate accurate alignment and stitching. This can be done in one of the following ways:
  - **Manual Identification**: Select corresponding points between the images by hand. **(3)**
  - **Automated Detection**: Use a feature descriptor, such as SIFT or ORB, to automatically identify matching points across the images. **(5)**
- **Image Stitching**: Using the identified points, align and stitch the images to form a composite. **(6)**

- **Rectification**: Correct any perspective distortions and non-uniform scaling to ensure that the final image is rectangular, resembling an A4 sheet. **(6)**
- **Comparative Analysis**: Evaluate and discuss the quality of stitching obtained through manual feature selection versus using an automated feature descriptor. Comment on the accuracy, smoothness of blending, and ease of implementation for each method.**(5)**

2 Hough Transform

After the space shuttle has completed its mission in orbit, it begins its descent back to Earth. As it approaches the atmosphere, its navigation systems activate advanced visual processing techniques to prepare for a precise landing. Upon entering the atmosphere, the shuttle's sensors scan for a safe landing runway. Using Hough Transform algorithms, the shuttle identifies straight lines on the runway, which helps it align perfectly for touchdown.

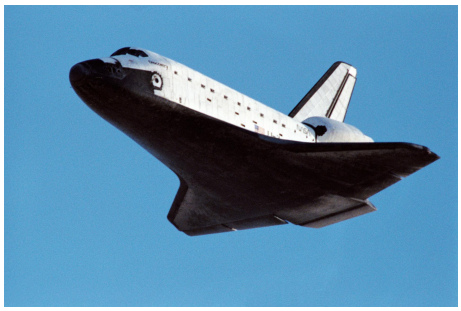


Figure 2

Your mission is to simulate the space shuttle's final approach. Given an image representing the shuttle's view of the landing area, use Hough Transform to detect and delineate the runway lines. This will aid in guiding the shuttle for a safe landing.

Figure 3

In a groundbreaking mission, SpaceX's Falcon 9 booster launches a payload into orbit. After the booster separates, it autonomously begins its descent, aiming to land precisely on a circular landing pad on land. During its descent, the booster's cameras and sensors analyze the view of the landing zone searching for the landing pad. To ensure a safe landing, the booster scans the surface, using a powerful algorithm to detect the circular landing zone amidst other infrastructure and clutter.

Your mission is to assist the Falcon 9 booster's landing process. Provided with an image 5 that mimics the booster's view as it nears the landing zone, use Hough Transform to locate the circular landing pad. Accurate detection of this circle is crucial for a safe touchdown.

Figure 4

Figure 5

Suggested Approach

- **Implementation**: Implement the Hough transform from scratch for both the scenarios
  - **Edges of runway**: Identify the bounding edges of the runway for precise touchdown and landing **(7)**
  - **Edges of landing pad**: Identify the both the bounding edges of the circular(the smaller circle and the larger circle) Landing pad for precise touchdown and landing **(7)**
- **Overlaying**: Using the implementation of hough transform function overlay the final result on the two original images **(3+3)**
- **Improvements**: Discuss if the detection is satisfactory if not what changes would you suggest to make the mission successful **(5)**

3 Segmentation

After the successful completion of various space missions, you are now traveling in your car equipped with Full Self-Driving (FSD) capabilities. However, despite its sophisticated algorithms for autonomous driving, the system is glitching when it comes to identifying stop signs—a critical aspect of safe navigation on the road. As a computer vision expert, you are tasked with implementing a quick and effective solution to resolve this issue using image segmentation techniques.

Objectives

Your goal is to perform image segmentation on a given image 6 that include a stop sign. You will employ two different methods of image segmentation and compare their effectiveness.

- **Method 1: Mean Shift Segmentation**
  - Implement the Mean Shift algorithm to segment the images. This technique groups pixels based on color and spatial proximity, which can help in isolating the stop signs from the background.

- **Method 2: Normalized Graph Cut Segmentation**
  - Utilize the Normalized Graph Cut method for segmentation. This method formulates the image as a graph and partitions it based on the minimum cut, allowing for better delineation of complex structures within the image, including stop signs.

Figure 6

Deliverables

- For each segmentation method, provide:
  - A brief description of the algorithm and its implementation. **(10)**
  - The original image and the segmented result for comparison. **(10)**
  - A discussion on the strengths and weaknesses of each method in the context of identifying stop signs. **(5)**

4 Creative Section

Points (25)
If you beat this section, you will get **(TBD)** bonus points.

The task of this creative section is to come up with the best segmentation algorithm which is not machine/deep learning based approach and follows classical approaches.

4.1 Task Description

- **Image Selection**: Choose a challenging image for segmentation. The complexity of the image will provide a better opportunity to showcase your algorithm's effectiveness.
- **Performance Evaluation**: - Calculate the Intersection over Union (IoU) values for all segmented objects within the image. This metric will provide a quantitative measure of segmentation accuracy, defined as:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

  - Report the IoU values for each identified segmented region.

- **Detailed Approach Description**: Provide a comprehensive explanation of the approach you implemented, including: - The rationale behind your choice of algorithm. - Step-by-step details of the segmentation process. - A justification of why you believe this method performs better than alternative approaches.
- **Comparison and Empirical Evidence**: Compare your results with other classical segmentation techniques (if applicable) and present empirical proof of your method's performance. This may include visual comparisons, numerical results, and any relevant metrics that support your claims.
- **Function Requirements**: Your segmentation algorithm should be encapsulated in a function that accepts an image as input and outputs all identified segmented regions.
- **Bonus Eligibility**: The top **TBD%** of students who achieve the highest average IoU for all identified regions will be eligible to receive the bonus points.