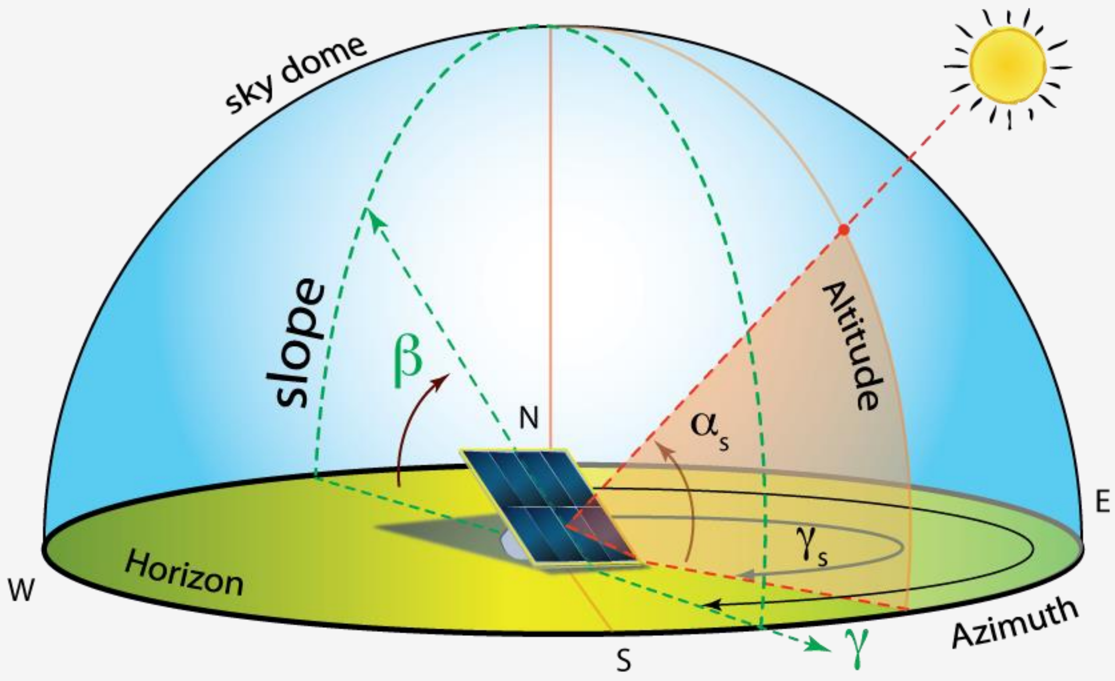
Identification of Sky Regions in a Photo

Description (Problem NM381)

Generating local sky horizon has important applications for analysis of solar energy potential in an urban setting. Develop a mobile application for automatically detecting sky pixels in a photograph. The application should generate a mask image consisting of sky pixels marked in white color in the image and other pixels marked in black color. Further, using information about camera optics, the application should give angle of elevation of the lowest sky pixel for all pixel columns in the mask image.



Use case: Determining feasibility of solar panels and optimization.

Hardware and Software

Sobel Operator
for Edge Detection

Hardware

- Camera Module

Programming Language

- Python -

Software Techniques

- Open Source Computer Vision (OpenCV)
- NumPy
- Sobel Operator for Edge Detection

Flowchart (Work Proposed for the Day)

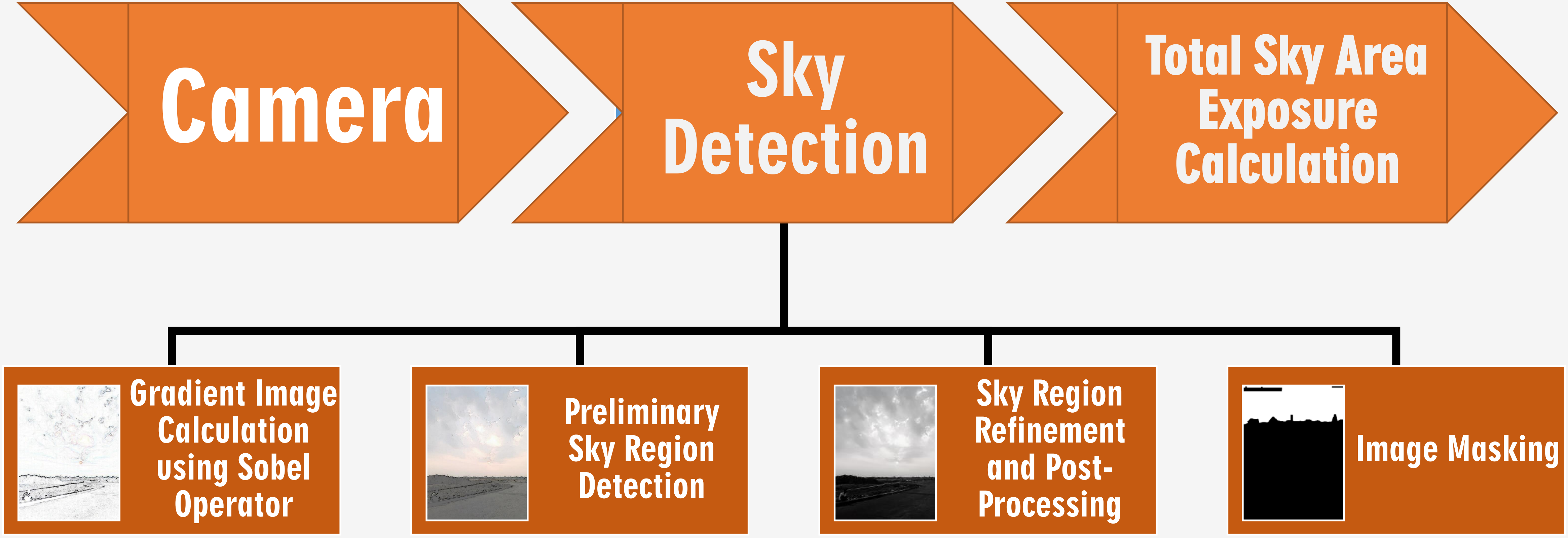


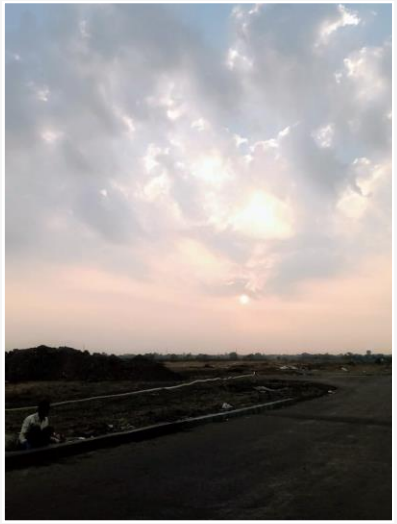
Image Pre-Processing

Steps Involved:

1. Converting RGB image into greyscale.

2. Converting greyscale image into gradient image with Sobel Operator (for image processing and edge detection).

3. Two images hence obtained with horizontal and vertical gradients are superimposed to obtain border of every object in the coloured image.



Original

Pre-Processed

Superimposed

* To get better results image filters have been applied. This helps to reduce noise in the image.

Preliminary Sky Region Detection

Mathematical Modelling

To distinguish between sky and ground we use Energy Function, the original colour image is divided into sky and ground regions. The pixels in both regions are denoted by RGB components. The energy function defined is as follows:

$$J = \frac{1}{|\boldsymbol{\Sigma}_s| + |\boldsymbol{\Sigma}_g| + (\lambda_1^s + \lambda_2^s + \lambda_3^s)^2 + (\lambda_1^g + \lambda_2^g + \lambda_3^g)^2} \tag{1}$$

where $\boldsymbol{\Sigma}_s$ and $\boldsymbol{\Sigma}_g$ are the covariance matrices of the pixels that are described by RGB values in sky and ground regions, respectively. They are both 3×3 matrices, which are defined as follows:

$$\boldsymbol{\Sigma}_s = \frac{1}{N_s} \sum_{(y,x)\in sky} \left(\mathbf{I}^s(y,x) - \boldsymbol{\mu}^s\right)\left(\mathbf{I}^s(y,x) - \boldsymbol{\mu}^s\right)^T \tag{2}$$

$$\boldsymbol{\Sigma}_g = \frac{1}{N_g} \sum_{(y,x)\in ground} \left(\mathbf{I}^g(y,x) - \boldsymbol{\mu}^g\right)\left(\mathbf{I}^g(y,x) - \boldsymbol{\mu}^g\right)^T \tag{3}$$

where N_s and N_g are the number of pixels in the sky and ground regions, respectively, while $\boldsymbol{\mu}^s$ and $\boldsymbol{\mu}^g$ are 3×1 column vectors that represent the average RGB values in the sky and ground regions, respectively.

Firstly, we define a sky border position function $b(x)$:

$$1 \le b(x) \le H (1 \le x \le W) \tag{7}$$

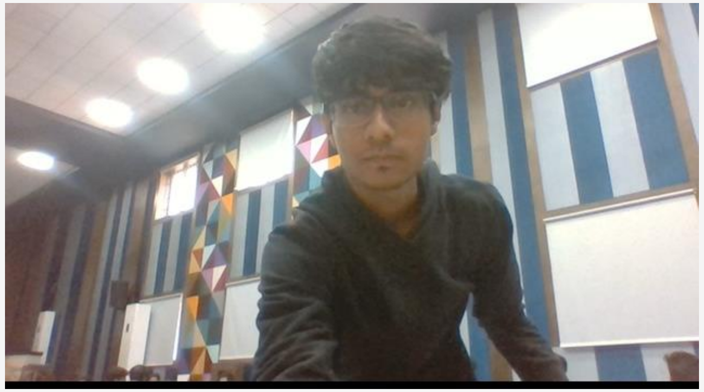
where W and H are the width and height of the image, respectively, and $b(x)$ determines the sky border position in the x^{th} column. That is to say, the sky and ground regions can be calculated with the following equations:

$$sky = \{(x,y) \mid 1 \le x \le W,\ 1 \le y \le b(x)\} \tag{8}$$

$$ground = \{(x,y) \mid 1 \le x \le W,\ b(x) < y \le H\} \tag{9}$$

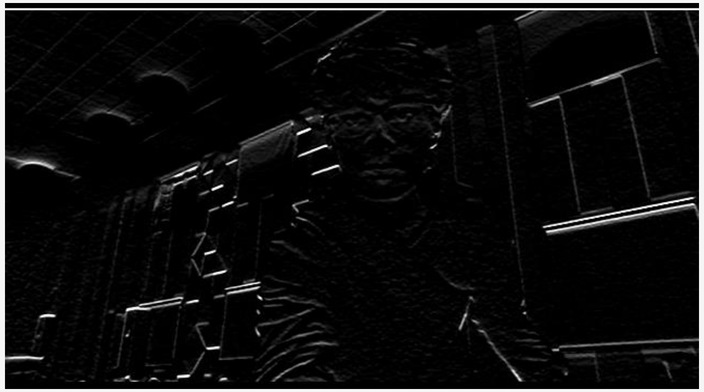
The sky and the ground colour signatures will hence help us to distinguish between sky and ground regions.

Live Experiment (Implementation for border masking)



Original

Greyscale



Horizontal Gradient

Vertical Gradient

Superimposed(Horizontal and Vertical)

Suggested Edits

Implementations/Corrections

1. Distinguish Between Sky and the Ground.

2. Differentiate between Sky and the Sea.

3. Measure intensity of light from the sky.

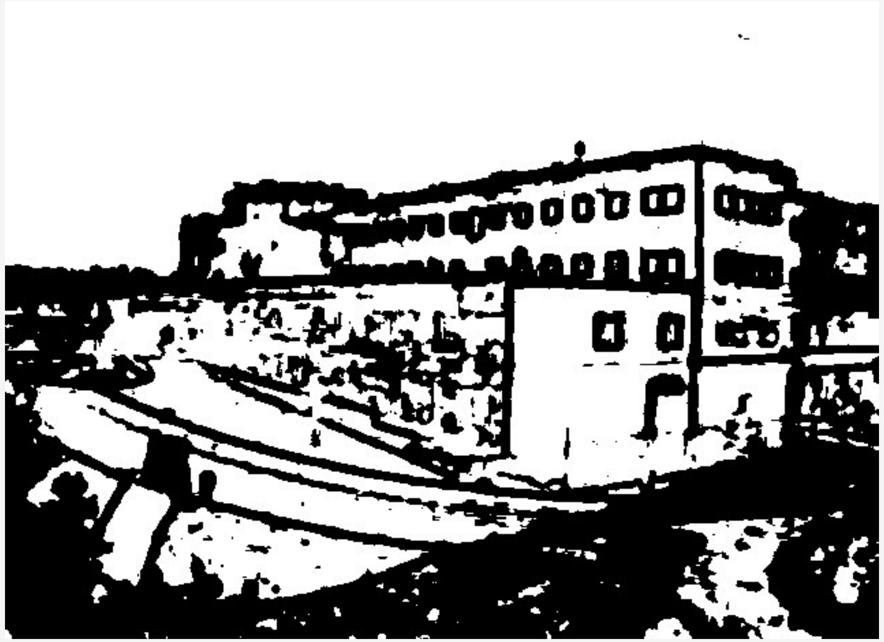
1. Mathematical Modelling of technique to differentiate sky and ground and hence implemented.

2. Refined algorithm to detect ripples in the moving water by making the image sharper and thus differentiate it from the smooth sky.

3. If the image is taken in an area where sufficient amount of light is unavailable, it will show low light error.

Limitations

1. Does not work on over exposed or blur images.
2. Does not work after daytime.
3. Inaccurate inside any medium such as water or glass.



Thank You

Team 1D345

- Prayag Bhatnagar 18BCE10196
- Hritvik Semwal 18BCG10048
- Shreeyash Jejurkar 18BCE10254
- Kritika Shah 18BCE10145
- Aditya Yadav 18BCE10015
- Mihir Semwal 18BCY10054