Sky Region Detection Algorithm Using Image Lightness Extraction and Enhancement

Garv Sudhir Nair
School of Engineering & Technology
Sunway University
Bandar Sunway, Selangor
garv.n@imail.sunway.edu.my

***Abstract* — Sky region detection has various uses within the fields of meteorology, weather detection, and environmental analysis. It is through computer vision that the process of analysing the sky becomes automated. By researching various existing methods and approaches to segmenting sky-regions within images, this paper has developed an algorithm that is the result of a combination of two primary ideas. Algorithmic Sky-Region detection alongside an image color extraction and enhancement algorithm. The proposed algorithm provides an increase in performance with regards to night-time images and can be further improved by adjusting parameters to the user's corresponding use-case.**

Keywords — OpenCV, Computer Vision, Sky Region, Detection, Extraction, Algorithm, Enhancement

I. Introduction

Human beings make use of various naturally occurring phenomenon to better predict and analyse environmental changes that could occur in the future. One can collect such data by simply observing these known occurrences and gauging a relevant hypothesis. An instance of such would be observing a cloudy sky and inferring that it is going to rain. However, constant observation of the sky, in this scenario, is simply not possible for a single human being to accomplish. Furthermore, minute changes in our peripheral view oftentimes go undetected due to lack of focus.

As such, researchers now make use of technology and automation to make these observations for them. Using modern technology, one can make use of the sky to forecast weather, detect solar exposure, or even navigate drones. However, creating an automated system of such scale would require a machine to be able to differ between the sky and a potential scenery. In such cases, algorithms specifically designed to detect sky-regions within images and videos are used. This report will discuss, compare, and contrast various methods and systems employed in the modern era to detect, interpret, and segment sky regions within outdoor imagery.

II. Literature Review

There are a number of systems and applications that make use of sky region detection in the current day; however, each may vary in terms of application. As such, five peer-reviewed journals will be cited and broken down in order to better analyse and surmise the weaknesses and strengths of each methodology..

A. *Sky Pixel Detection in Outdoor Imagery using an Adaptive Algorithm and Machine Learning*

Modern computer vision consistently makes use of the concepts of deep learning or neural networks. This paper [1] makes use of CNN-RNN Architecture as its primary method of distinguishing sky features. This particular method differs from traditional computer vision algorithms as it also uses neural networks to artificially replicate a human being's ability to detect patterns. The system follows conventional machine learning techniques and opts towards selecting the most optimum sky-detection algorithm for an input image.

This particular selection step allows the system to be adaptive and select input parameters and techniques that would return the best results [1]. The adaptability comes not just from the selection and parameter distribution itself, but also from the distinction of each algorithm provided to the CNN. The three main techniques employed and discussed within the paper are as follows:

1) *Mean-Shift Segmentation Algorithm*

Mean shift is a popular approach for image segmentation. One can practically apply this method of image segmentation by decomposing images into homogenous contiguous sections of pixels with comparable colours or grey levels. Mean shift selects search windows of a specified radius in an initial point in an image iteratively, computes a mean shift vector, and moves the search window by that amount until convergence. Within this paper, four input parameter variations were employed, each of which was determined experimentally through a sensitivity test to work across the largest range of images possible. The specified set of parameters is applied to each image, and pixels of the most common colour are identified as the sky. The obvious issue stemming from this approach is the possibility of false positive occurring. With images containing multi-coloured clouds or odd lighting, opting to select pixels on the basis of colour may not be the best approach.

2) *Using Sobel Operators*

Alternatively, the CNN may choose to select the hybrid probability model or Sobel Operators approach. Sobel operators are often used for edge detection within the field of computer vision, by calculating the difference between pixel intensities within an edge region [1]. To estimate sky colour, this approach creates grey-scale gradient images using x and y directional Sobel operators. Using the covariance matrices of an initial computation of sky and ground areas, an optimised objective function aims to locate the optimal sky/ground border in the gradient image.

Using this newly formed boundary, a probability model is generated on the basis of the standard deviations of each colour, gradient values, and the vertical position of each specific pixel. The pixels are given a value between 0 and 1 based on the probability of them being sky pixels.

3) K-Means Clustering

K-Means clustering and HSL (hue, saturation, and lightness) colour filtering were used for the third technique. K-means clustering divides a picture into K clusters iteratively, stopping when a set of conditions is met (i.e., maximum iterations and/or desired threshold) [1]. K-Means clustering being dependent on its parameter settings allowed for there to be multiple variations of its implementation, with different parameters allowing for specific types of sky-regions to be detected.

B. *Edge-Based Detection of Sky Regions in images for Solar Exposure Prediction.*

Neural Networks, while effective, are not yet conventional nor as easy to implement as regular computer vision algorithms. Thus, [2] brings forward a more friendly and intuitive approach towards sky-region detection, with the Canny edge detector and the morphological closing algorithm being used to discover the sections in the image that separate the sky from the rest of the image. Furthermore, the brightness and area of each zone are used alongside the FloodFill algorithm in order identify all pixels in a sky region. The approach is rather simple in nature,

1. Extract a colour plane.
2. Determine the image's boundary lines using the Canny edge detection algorithm.
3. Apply the morphological closing algorithm to close gaps in the boundaries.
4. Determine which of the enclosed areas is the sky.

Given the simplicity of the algorithm, the main takeaway comes from its lack of flexibility with respect to certain images. However, given the primary use case in the paper is related to solar exposure prediction, the chances of false positives are not as much of an issue due to the general focus on sun paths within the image and will not affect the solar exposure calculation by much, therefore the lack of flexibility can be considered negligible.

C. *Sky Region Detection in a Single Image for Autonomous Ground Robot Navigation*

This paper [3] makes use of a sky detection algorithm that bases itself off gradient information and energy function optimisation. First, the image's gradient information is acquired. Then, using the energy function optimization, the appropriate segmentation threshold in the gradient domain is computed, and the preliminary sky region is estimated. Finally, when no sky region appears in the image or when objects extrude from the ground, a post-processing procedure is used to refine the preliminary sky region detection result.

The general focus of the paper tends to go over its ability to segment the sky from both grayscale and colour images with very little computational cost [3]. This is certainly the most key factor to be considered for on the ground automated robots, as navigation requires quick and fast computation in order to better function on the go. The algorithm can be seen below,

Input: original image **I** for sky region detection.
Output: detected sky region.

Calculate the gradient image *grad* according to section 2.1.
Calculate the optimal sky border position function $b_{opt}(x)$ with algorithm 2.
Calculate *border_ave* and *ASADSBP* according to equations (12) and (13).
Test *border_ave* and *ASADSBP* according to equation (14).
if equation (14) is satisfied
  There is no sky region in the image.
  return
else
  Calculate $diff\_abs(x)$ according to equation (16).
  Test $diff\_abs(x)$ according to equation (17).
  if equation (17) is satisfied
    Recalculated the sky border position function $b_{new}(x)$ with algorithm 3.
    Calculate the sky region with $b_{new}(x)$ according to equation (8).
    return
  else
    Calculate the sky region with $b_{opt}(x)$ according to equation (8).
    return
  end if
end if

Figure 1. Summary of Algorithm [3]

While the algorithm itself is not simple, it is far more conducive and digestible than its neural network counterparts. Furthermore, given each step of the algorithm can be encapsulated within their own functions, it is possible to make modifications to the algorithm as need be.

D. *Sky Detection in Hazy Image*

The last paper, [4] makes use of a variety approach, which appears to combine both the ideas of classifiers as well as conventional computer vision techniques such as Canny edge detection. It follows the essential steps for sky detection, with image segmentation, feature extraction, and gradient features extraction, but where it differs is its use of region classification. [4] makes use of a two-stage sky detection algorithm, which applies two SVM (Support Vector Machines) to determine a region as sky or non-sky [4].

SVMs are supervised learning methods often used for classification, and in this context, are trained to detect and label sky and non-sky regions. The algorithm is simplified down as follows,

1. Segment image into small homogenous regions.
2. Compute features for each region within their respective samples.
3. Classify each sample using both SVMs and obtain their respective energy functions.
4. Detect the high-confidence non-sky regions using the formula detailed within the paper.

III. Proposed Algorithm

While these various methods have been proven effective within their own field of research, this paper will propose a solution that can make use of a combination of the various ideas proposed by the previously cited literature. More specifically, the algorithm proposed in this paper will be a combination of the functionality described in papers [2] and [3]. These papers were specifically chosen due to the simple nature of their algorithms and ease of implementation.

The nature of the algorithm can be broken down to two main factors that define the algorithms of [2] and [3] respectively. With paper [3], the algorithm already returns a 95% accuracy and a processing time of 150ms, placing it as the fastest and most accurate method that has been cited. With paper [2], as is seen in Section II B, one can see a very clear steps that involve extracting the colour plane of the image and applying a morphological close algorithm on the resulting image. The reason the canny edge detection and segmentation steps are skipped entirely is because algorithm [3] already makes up for it with its optimum border function. Furthermore, given the general visual breakdown of the datasets provided for this project. It can be determined that extracting colour channels such as blue would not serve the purpose as paper [2] intends it. This can be further proven through test trials, which were conducted across the blue channel. These tests involved conversion of the test photo from an RGB colour space to a LAB colour space. This splits the sample image into three channels, lightness, channel a, and channel b. Channel b consists of those values that stretch across blue to yellow. Upon applying adaptive histogram equalization on the blue channel, the returned image does not reflect an abundance in blue as is required, and neither does it return an image that shows a stark contrast between the ground and sky segments.

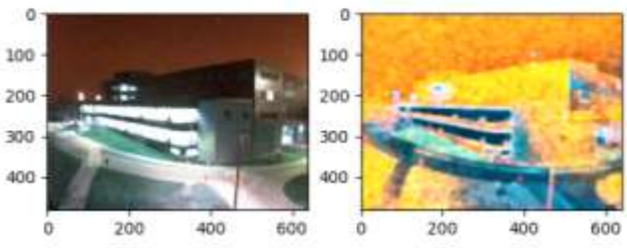

Figure 2. Comparison between image and its histogram equalization equivalent

Figure 2 provides the necessary information to conclude that extracting the colour channels as a step is too inconsistent when spread across images with different coloured sky. As such, the proposed algorithm will instead extract the lightness of the image as its initial step. The change of focus from contrast from colours to contrast from lightness allows for images within darker settings to be highlighted and return much more consistent results. This extraction of lightness is further bolstered by image enhancement steps, in which the saturation is decreased, the brightness & contrast increased, and the gamma corrected accordingly. The values used for each of these functions were selected using trial and error on the basis of returning the most optimal results. Apart from this primary change the rest of the algorithm stays relatively the same, with the section of algorithm [3] undergoing no changes as it has already been optimised to yield good results. As such when these are eventually combined, the newly proposed algorithm can be summarized as follows.

Sky Region Detection Algorithm
Input: Original Image (i) Output: Segmented sky region image

Image undergoes adaptive histogram equalisation.
 	
Image is then put through the following functions
- Saturation Decreased (Value = 0.7)
- Contrast Increased (Value = 1.2)
- Brightness Increased (Value = 1.2)
- Gamma Corrected (Value = 0.7
 	
Image is put through a morphological closing algorithm (dilation & erosion)
 	
Image is passed to gradient image calculation function.
 	
Calculate the optimal sky border position.
 	
Calculate border_ave and the absolute sum of absolute differences (abs_sum_abs_diff) according to the equation provided in [3].
 	
if equation (14) is satisfied
    There is no sky-region
else
    Calculate diff_abs(x) according to equation (16).
    if equation (17)
        Recalculated the sky border position function
        Calculate the sky region
        return
    else
        Calculate the sky region with bopt (x) according to equation (8)
        return
    end if
end if

It can be noted that all equations mentioned within the proposed algorithm can be found in [3].

With both algorithms using a very clear step-by-step approach to sky-region detection, it can be seen through our proposed algorithm that combining their functions is well within the feasibility of this project.

IV. Results & Discussion

A. Experimental Setup

The datasets required by this project were specified to be taken from the skyfinder registry of images (https://cs.valdosta.edu/~rpmihail/skyfinder/images/index.html). It is to be noted that this project primarily used and tested upon the 1093, 4795, 8438, and 10870 datasets respectively.

The general setup of the project is simple, with the folder structure being as follows,

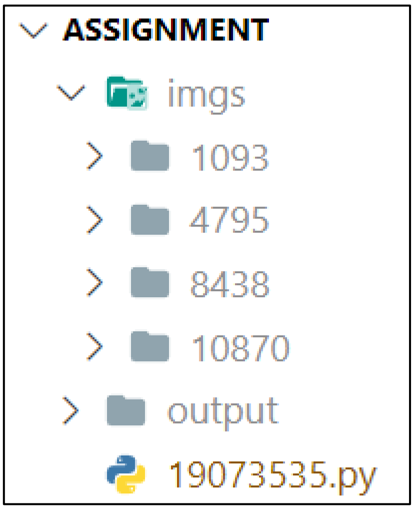

Figure 3. Folder Structure of the Project

The main algorithm built in `19073535.py` and our image datasets being placed within the `imgs` folder. Running the algorithm is fairly simple and only requires the user to input `python 19073535.py` into their command line.

The user will be prompted to enter a folder name and can simply enter any of the datasets that exist within their `imgs` folder, as follows.

```
Enter Folder Name (Must be in imgs folder): 1093
```

The algorithm then runs while iterating through all images within the folder (jpg, jpeg, and png). The segmented image output is then saved into the folder `output`, in the order in which the images were iterated through.

B. Results

The proposed algorithm provides a varying level of results based on the image at hand. It is notable that regular skyline images continue to be detected with the same level of accuracy as was previously discussed in section III.

Figure 4. Segmented Sky Output from Proposed Algorithm applied on dataset 1093.

As such, one can compare our algorithm to the original algorithm proposed in [3]. With the application of lightness extraction and morphological closing, the use-case for darker images extends considerably.

Figure 5. Segmented Sky Output from Proposed Algorithm applied on night-sky image.

While the above output may seem inconsistent in describing the horizon across the edge of the building, it is an improvement when compared to the original algorithm that did not detect a horizon at all and therefore generated no output.

This stays consistent across the night-sky images, as the lack of detail within the imagery does prove to be hard to circumnavigate through, but our algorithm retrieves a relevant amount of segmented sky from the image regardless.

C. Errors

An issue does arise from our implementation however, which causes inconsistent results based on the lighting within the image. As our algorithm now makes use of dilation and erosion in order to denoise and close contrasted edges within the image, there comes certain instances in which dilated or highlighted areas within the image get detected as non-sky regions due to a spot of light reflecting across a surface.

This issue arises as the image enhancement procedure focuses on increasing the overall brightness and contrast of the image, therefore putting emphasis on certain light spots within an image. Reflective surfaces such as windows or white painted buildings blend in with the sky and as such are detected as such.

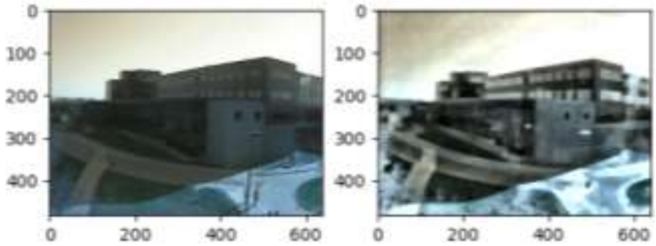

Figure 6. Image from 1093 dataset pre and post enhancement.

As can be seen in figure 6, the post-enhanced image has a notable light-spot emerging from the right. This light spot is therefore also reflected in the resultant output image as seen in figure 7 below.

Figure 7. Resultant Output Image from Figure 6

While it may seem like a breaking issue, this is not too concerning in the scope of the algorithm. This can be inferred as the pre-processing algorithm makes use of finely tuned parameters within their functions. These parameters were selected on the basis of trial and error and are far from the most optimum values that could be used for the algorithm. With just a little adjustment, the results can return back to the original accuracy, however this would require more extensive look into the pre-processing functionality.

V. Conclusion & Future Work

It can thus be inferred that the algorithm can be further improved by fine-tuning the parameters chosen within the pre-processing algorithm. This would result in far more effective results and would make the algorithm more consistent than it currently is. This fine-tuning could possibly be implemented using a neural network as is followed by papers [1] and [4], therefore cutting the manual labour that would usually be required for such a task.

With regards to each of the algorithms and systems, it is safe to say that there is no single ideal approach to detecting sky-regions within images. This paper, however, was able to reach a close to ideal approach by combining the ideas proposed by papers [2] and [3]. The application of pre-processing algorithm, involving lightness extraction and morphological closing, onto an already effective sky-region detection algorithm provides the user with consistent results across all datasets and can be seen to be effective with both daytime as well as night-time images.

References

[1] K. Nice et al., "Sky pixel detection in outdoor imagery using an adaptive algorithm and machine learning", *Urban Climate*, vol. 31, p. 100572, 2020. Available: 10.1016/j.uclim.2019.100572.

[2] N. Laungrungthip, A. McKinnon, C. Churcher and K. Unsworth, "Edge-based detection of sky regions in images for solar exposure prediction", *2008 23rd International Conference Image and Vision Computing New Zealand*, 2008. Available: 10.1109/ivcnz.2008.4762101.

[3] Y. Shen and Q. Wang, "Sky Region Detection in a Single Image for Autonomous Ground Robot Navigation", *International Journal of Advanced Robotic Systems*, vol. 10, no. 10, p. 362, 2013. Available: 10.5772/56884.

[4] Y. Song, H. Luo, J. Ma, B. Hui and Z. Chang, "Sky Detection in Hazy Image", *Sensors*, vol. 18, no. 4, p. 1060, 2018. Available: 10.3390/s18041060.