Asian Journal of Mathematics and Computer Research

Volume 30, Issue 2, Page 1-8, 2023; Article no.AJOMCOR.11473
ISSN: 2395-4205 (P), ISSN: 2395-4213 (O)

Estimating the Obstructed Portion of the Night Sky Using Digital Image Processing

Felipe Alexandre da Silva [a], Vinicius Daniel dos Reis de Castro [a], Marcos William da Silva Oliveira [a], Lucas Antonio Caritá [a*], Irapuan Rodrigues [b] and Francisco Carlos Rocha Fernandes [c]

[a] *Grupo de Pesquisa em Matemática Científica e Computacional (GPMCC), Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP), São José dos Campos, SP, Brazil.*
[b] *Universidade do Vale do Paraíba (UNIVAP), São José dos Campos, SP, Brazil.*
[c] *Faculdade de Tecnologia de Bauru (FATEC), Bauru, SP, Brazil.*

Authors' contributions

This work was carried out in collaboration among all authors. All authors read and approved the final manuscript.

Article Information

DOI: 10.56557/AJOMCOR/2023/v30i28232

Original Research Article

Received: 25/02/2023
Accepted: 29/04/2023
Published: 05/05/2023

Abstract

Determining the portion obscured by clouds in the night sky recorded in an image from a meteor monitoring station can be useful to help in the discussion and analysis of studies in this area. Thereby, this work presents a proposal to quantitatively measure the proportion of the night sky covered by clouds from images of meteor records. The images used are from EXOSS database, particularly from the UVP1 and UVP2 stations. Two approaches were elaborated: the first one computes the proportion of pixels in clouds over the total number of pixels, the second approach calculates another percentage value by using image processing techniques and applying an equalization method. An application was created to perform the calculations in an automated way. The program displays an enhanced image and the two calculated percentages and can be used for specialists as a tool to guide their choice of coverage values. The results showed that the second approach proves to be better for "many" and "few" clouds; furthermore, the first approach proves to be better in low cloud density images.

**Corresponding author: Email: prof.carita@ifsp.edu.br;*

Keywords: Image processing; meteor monitoring; meteor shower; EXOSS.

1 Introduction

A meteor consists of a luminous phenomenon caused by the entry of a solid fragment, usually from comets or asteroids, at high speed into the Earth's atmosphere. Meteor showers occur when the Earth crosses the orbit of a comet and small dust particles enter the Earth's atmosphere [1]. The number of meteors observed in a meteor shower depends on the following: climatic conditions, the degree of light pollution in the observer neighborhood, the latitude of the observation location, etc. For this reason, observers from different locations, when looking at the same meteor shower, can see very different situations [2].

One way to allow comparisons between observations of meteor records made in different parts of the world is, for example, the use of the Zenithal Hourly Rate (ZHR) [3]. This parameter informs the rate of meteors that would be observed per hour with ideal observation conditions, with clear sky (objects visible up to magnitude 6.5) and with the observer positioned so that the radiant of the meteor shower is at its zenith. For the computation of the ZHR, among other variables, it is necessary to estimate the obstructed portion of the sky in the field of view. This is a problem, and a bare eye estimate is often performed. Also, a good estimate of the percentage of the observer's field of view which is obstructed could be very useful in discussing meteor shower records in studies such as [1,4-7].

Thereby, considering that similar searches (but not of the same problem) were done with satisfactory results [8,9], this work presents a proposal to quantitatively measure the proportion of the night sky covered by clouds from images of meteor records. All images used in this study are from the EXOSS Network database, particularly from the UVP1 and UVP2 meteor monitoring stations [2].

2 Methodology

The percentage of clouds in an image was calculated using two different methods using the programming language Python with OpenCV [10] library, wich was built to provide a common infrastructure for computer vision applications. The OpenCV has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS facilitating the other languages adaptations.

2.1 Database

EXOSS (EXplOring the Southern Sky for new meteor radiants) is a Brazilian project aimed primarily to register meteors and fireballs or bolides using low-cost cameras and meteor analysis softwares for video capturing and astrometry [11]. EXOSS started its activities in May 2015 with 9 associates, and with rapid expansion, it currently has 61 associates (20 institutions) in 13 Brazilian states, with 61 stations, operating 82 cameras daily in meteor records and data acquisition for studies of its dynamics and nature.

All images in the database used for this study were registered by the UVP1 and UVP2 EXOSS stations located in São José dos Campos, SP (Observatório de Física e Astronomia at Universidade do Vale do Paraíba).

A sample of 228 images were selected, equally distributed and roughly visually classified with "few", "many" or "no" clouds.

2.2 Approach 1

The first approach computes the proportion of pixels in clouds over the total number of pixels. In this method, an image from the database is used as input, then a blur filter (also known as normalized box filter) is applied for noise smoothing, using Open CV function *cv.blur(image,(5,5))*. After that, the image is converted to gray scale, using *cvtColor* function, changing its color space from RGB to gray scale (Fig. 1).

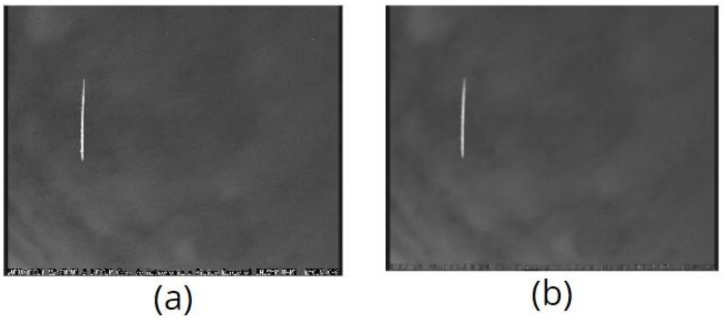

Fig. 1. An original image (a) and a post-conversion to gray scale image (b)

In Fig. 1 the pixels intensity values are modified, despite the visual inspection indicates no significant changes. The value of a color channel intensity can variate from 0 to 255. When the input image is converted to gray scale, changing color channels R, G, and B to a same value, turning all colors into gray intensities.

The output of approach 1 is the fraction of pixels in the gray image above a selected threshold (empirically determined as 100).

2.3 Approach 2

The second approach continuates the first one and calculates another percentage of clouds coverage by applying image enhancement techniques.

To select clouds in this method, the gray scale images pass through a contrast limited adaptive histogram equalization process (CLAHE) [12] to increase global contrast intensity, configurated with *clipLimit=4* and *tileGridSize=(8,8)*. A global equalization method distributes color channels intensity based on a global limit; however, an adaptive equalization divides an image into small blocks; each block histogram is equalized maintaining an intensity limit. After that, a linear interpolation is used to combine every block by removing its artificial limits generating an image with better contrast between clouds and background (Fig. 2).

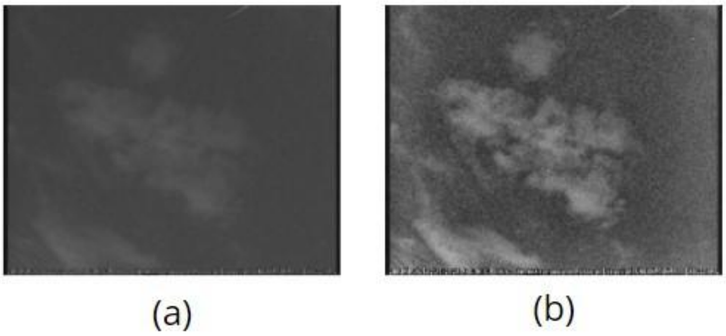

Fig. 2. A gray scale image before (a) and after applying CLAHE (b)

In the next step, a bilateral filter [10] is applied as a noise smoothing technique, same as a blur function, however a bilateral filter also preserves edges, improving the separation between clouds and the background. This application was made with *cv.bilateralFilter(image, 3, 75, 75)*.

After that, applying Otsu limiarization method [13] the clouds are separated from the background, generating a binary image, using *cv.threshold(image, 120, 255, cv.THRESH_BINARY + cv.THRESH_OTSU)*. Furthermore, applying a closing morphology filter, with kernel 7x7, in the binary image to close small holes in the foreground and small black points on the selected clouds improving the boundaries of the selected clouds contours, an example is shown in Fig. 3.

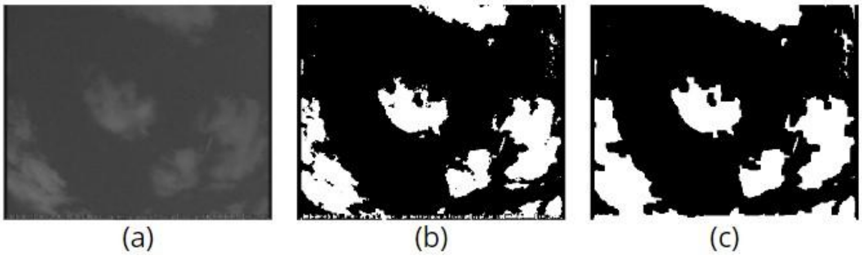

Fig. 3. Image shows a gray scale image (a) and the result of two steps Otsu limiarization (b) and closing morphological filter (c)

Finally, the second percentage value is calculted by the fraction of white pixels in the binary image over the total number of pixels

3 Results

The results are based on the analysis of 228 images in a database previously classified by experts in three categories: “few”, “many” or “no” clouds. An application developed shows the results of both methods (see the Appendix for access to the code in Python) using one image as the input and, then, calculates the percentage values for both methods, approaches 1 and 2, displaying the original and filtered image together with the two percentages.

There are two buttons and a display on the program, the first button, named “select image”, selects an image to apply both approaches and the second button, named “exit”, shutoff the program. Two display fields show the image after applying a bilateral filter and gray scale. Fig. 4 shows the program interface.

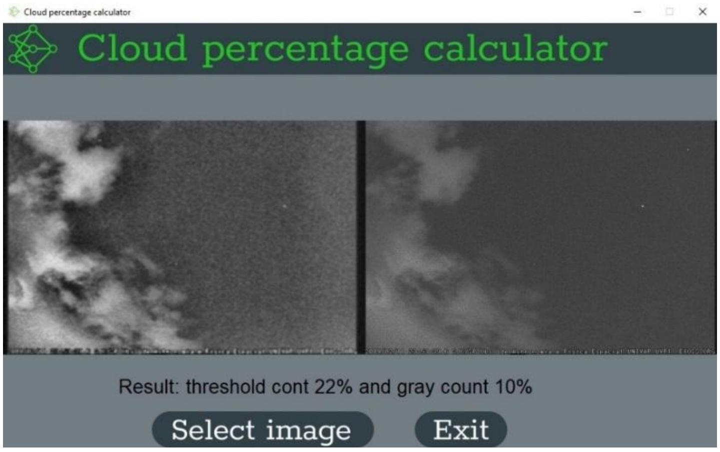

Fig. 4. Program interface when an image is inserted and processed

For each previous category an approach is visually better than the other. For “many” clouds category approaches 1 and 2 have a small difference, however approach 1 is visually better, results can be seen in Table 1. In the “few” clouds category, approach 2 is more suitable than Approach 1, because Approach 2 can capture the clouds shapes, which approach 1 is not able to perform.

Nevertheless, for the “no” clouds category approach 1 is visually better than 2 because in this category, the filter applied for approach 2 uses the image illumination, ignoring the lack of clouds giving high percentage values, however in approach 1, the count of gray pixels indicates that there are no clouds. Overall, approach 2 proves to be visually better for “many” and “few” clouds; furthermore, approach 1 proves to be visually better in “no” cloud density images, we assume it is to be expected, since the Otsu algorithm will always segment an image into two regions, even if there is no region of interest.

Table 1. The table shows results for some images of each classification many, few and no clouds. On the left, the names of the files in the database are displayed. On the right, the performance of the two approaches

Calculated percentages		
Many clouds		
Image name	**Approach 1(%)**	**Approach 2(%)**
M20211024_060518_UVP_1P	92	97
M20211024_061912_UVP_1P	95	96
M20190516_003804_UVP_1P	84	87
M20210910_065757_UVP_1P	96	88
Few clouds		
Image name	**Approach 1(%)**	**Approach 2(%)**
M20210918_224553_UVP_1P	31	33
M20190912_223930_UVP_1P	46	43
M20190717_212325_UVP_1P	10	38
M20210304_072120_UVP_1P	3	69
No clouds		
Image name	**Approach 1(%)**	**Approach 2(%)**
M20170727_033421_UVP_1P	0	78
M20170729_051623_UVP_1P	0	82
M20170730_031204_UVP_1P	0	81
M20170725_072344_UVP_1P	0	80

4. Final Remarks

This paper presents two techniques to analyze the coverage of clouds in night sky images and an application developed to apply these two approaches.

The current work helps astronomers who study meteor showers by giving calculated non-empirical values to cloud percentages and displaying an enhanced image.

The program can be used as a guide line to help in the specialist choice of values for coverage of clouds by giving to the specialist two percentage values, calculated by approaches 1 and 2, and a enhanced image, that can influence his decision on the coverage value.

Despite the proposal sometimes presents two clearly different values for the two approaches, we believe that this is a first step towards obtaining a clear distinction between cloudy and cloudless skies. Our code is available in this work and we invite our fellow researchers to help in its improvement.

In future works the idea is to create a machine learning model, using the percentage values generated in approaches 1 and 2, to choose the best coverage value, creating a more assertive method to calculate cloud coverage percentages.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Silva RCA, Pimentel GAJ, Fernandes FCR, Caritá LA, Rodrigues I. Statistics of South Delta-Aquariids (SDA) meteor showers registered at UNIVAP in 2017 and 2018, Revista Brasileira de Ensino de Física. 2019;41.

[2] da Silva FA, de Castro VD, da Silva Oliveira MW, Caritá LA. Estimating the Obstructed Portion of the Night Sky. Proceeding Series of the Brazilian Society of Computational and Applied Mathematics. 2022;9(1).

[3] Brown P. On the cause and nature of error in zenithal hourly rates, WGN, Journal of the International Meteor Organization. 1990;18:141–145.

[4] Fernandes FCR, Caritá LA, Rodrigues I, Pimentel GJA, Silva RCA, Matos PB, Castro VDR, Oliveira JC. Southern delta Aquariids (SDA) meteor shower registered by UNIVAP stations in the triennium 2017, 2018 and 2019, WGN, Journal of the International Meteor Organization. 2020;48(6): 173–180.

[5] Johannink C, Miskotte K. Results for the Aquariid-expedition to Namibia, July 2012, WGN, Journal of the International Meteor Organization. 2012;40:65–68.

[6] Borovička J, Spurný P, Shrbený L, Štork R, Kotková L, Fuchs J, J. Keclíková, H. Zichová, J. Mánek, P. Váchová, I. Macourková, J. Svoreň, H. Mucke, Data on 824 fireballs observed by the digital cameras of the European Fireball Network in 2017–2018, Astronomy & Astrophysics, 2022;667.

[7] Matlovič1 P, Kornoš1 L, Kováčová1 M, Tóth J, Licandro J. Characterization of the June epsilon Ophiuchids meteoroid stream and the comet 300P/Catalina , Astronomy & Astrophysics. 2020;636.

[8] Cazorla A, Husillos C, Antón M, Alados-Arboledas L. Multi-exposure adaptive threshold technique for cloud detection with sky imagers, Solar Energy. 2015;144: 267-277.

[9] Dev S, Savoy FM, Lee YH, Winkler S. Estimation of solar irradiance using ground-based whole sky imagers. In 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). 2016;7236-7239.

[10] Bradski G. The OpenCV Library, Dr. Dobb's Journal of Software Tools. 2000;25(11): 120–123.

[11] de Cicco M, Lazzaro D, Santiago EP, Exoss Team. Brazilian video monitoring meteors network EXOSS: Status and first results—a citizen science project, in Proceedings of the International Meteor Conference (M. Gyssens and J.-L. Rault, eds.). 2018;162–164.

[12] Reza AM. Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement, Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology. 2004;38(1):35–44,.

[13] Otsu N. A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics. 1979 ;9 (1):62–66.

Appendix: Code in Phyton

```
import tkinter.filedialog from tkinter
import * import os import cv2 as cv
import numpy as np from PIL
import Image, ImageTk
def exibir(imagem):
    cv.namedWindow('njanela',
    cv.WINDOW_NORMAL) cv.imshow('njanela',
    imagem) cv.waitKey() cv.destroyAllWindows()
def resource_path(relative_path): try:
        base_path = sys._MEIPASS
    except Exception: base_path =
    os.path.abspath(".")
    return os.path.join(base_path, relative_path)

def valor_do_resultado(area_por1, area_por2):
    area_por2 = round(float(area_por2))
    area_por1 = round(float(area_por1))
    result_txt.configure(text=f'Result: threshold cont {area_por1}% and gray count
    {area_por2}%')

def contando_pixels(thr_imagem):
    contagem_bright = np.sum(thr_imagem == 255)
    cr = (contagem_bright / thr_imagem.size) * 100
    return cr

def contando_pixels_gray(thr_imagem):
    contagem_gray = np.sum(thr_imagem >= 100)
    cr_gray = (contagem_gray / thr_imagem.size) * 100
    return cr_gray

def botao_selecao():
    filename = tkinter.filedialog.askopenfilename(initialdir=os.getcwd(), filetype=(
        ('JPG File', '*.jpg'), ('JPG File', '*.jpeg'), ('PNG File', '*.png'), ('All File', 'how are you.txt')))
    imagem = cv.imread(filename, 1)
    kernel = np.ones((7, 7), np.uint8)
    imagem2 = Image.fromarray(imagem)
    img2 = ImageTk.PhotoImage(imagem2.resize((500, 327)))
    lbl2.configure(image=img2)
    imagem = cv.blur(imagem, (5, 5))
    gray = cv.cvtColor(imagem, cv.COLOR_BGR2GRAY)
    coverage_original = contando_pixels_gray(gray)
    clahe = cv.createCLAHE(clipLimit=4, tileGridSize=(8, 8))
    gray = clahe.apply(gray)
    gray = cv.bilateralFilter(gray, 3, 75, 75)
    imagem = Image.fromarray(gray)
    img = ImageTk.PhotoImage(imagem.resize((500,
    327))) lbl.configure(image=img)
    lbl.image = img
    lbl2.image = img2
    ret, thresh_img = cv.threshold(gray, 120, 255, cv.THRESH_BINARY +
    cv.THRESH_OTSU)
    thr_mod = cv.morphologyEx(thresh_img, cv.MORPH_CLOSE, kernel, iterations=2)
    coverage = contando_pixels(thr_mod) valor_do_resultado(coverage, coverage_original)
```

```
window = Tk()
window.title('Cloud percentage calculator')
window.geometry("1000x600")
window.configure(bg="#ffffff")
window.iconbitmap(resource_path('GRAFO
VERDE.ico'))
canvas = Canvas( window, bg="#ffffff", height=600, width=1000,bd=0, highlightthickness=0, relief="ridge")
canvas.place(x=0, y=0)
background_img =
PhotoImage(file=resource_path(f"background.png"))
background = canvas.create_image( 500.0, 300.0,
image=background_img)

lbl = Label(window, bd=0)
lbl.place(x=0, y=145)
lbl.pack(side='left')
lbl2 = Label(window, bd=0)
lbl2.place(x=0, y=290)
lbl2.pack(side='right')
img_back = Image.open(resource_path('nuvens back.jpg'))
img_back = ImageTk.PhotoImage(img_back.resize((500,
327))) lbl.configure(image=img_back)
lbl2.configure(image=img_back)
img0 = PhotoImage(file=resource_path(f"img0.png"))
b0 = Button(
    image=img0, borderwidth=0,
    highlightthickness=0,
    command=botao_selecao,
    relief="flat")
b0.place(
    x=208, y=542, width=310,
    height=51)
img1 = PhotoImage(file=resource_path(f"img1.png"))
b1 = Button(
    image=img1, borderwidth=0,
    highlightthickness=0,
    command=window.quit,
    relief="flat")
b1.place(
    x=575, y=542, width=129,
    height=51)
canvas.create_text( 499.5, 99.0,
    text="Enhanced image and percentage of coverage", fill="#000000",
    font=("RopaSans-Regular", 25))
result_txt = Label(window, bd=0, background='#747D83', text='Result: ...%', font=("RopaSans-Regular", 20))
result_txt.place(x=160, y=490)
window.resizable(False, False) window.mainloop()
```