Introduction

Dust composed of carbonates, clays, and metal oxides is the largest contributor to aerosols in Earth's atmosphere and significantly impacts climate, influencing everything from ocean temperatures to hurricanes ("The Dirt on Atmospheric Dust"). However, detecting dust plumes from satellite images at visible and infrared wavelengths is challenging because fog, dust, and clouds appear in similar colors. Scientists have found that by overlaying satellite images taken at different wavelengths into one composite image, the resulting coloration distinguishes between fog, dust, and clouds over a region ("Multispectral satellite applications: RGB products explained"). However, there is currently not an automated tool to interpret these composite images. We rely on meteorologists with specialized training to manually identify dust conditions based on the colors in these images. In this project I automate Dust RGB interpretation by providing an interactive application where users can retrieve imagery for specific times and regions and identify climate conditions by mousing over the image. This tool eliminates ambiguity, enabling non-experts to analyze Dust RGB images quickly.

Background

Satellites capture images at multiple wavelengths of light, which is useful because different particles in Earth's atmosphere reflect different wavelengths of light. To distinguish between dust, fog, and clouds in satellite images, scientists at the European Organization for the Exploitation of Meteorological Satellites developed a "Dust RGB" satellite image product, which takes the difference of satellite images at various visible wavelengths and merges that with infrared thermal images to create a composite image ("The Dirt on Atmospheric Dust"). The composite image is not a 'true color' image, but its colors highlight different dust and cloud conditions. Currently, meteorologists identify the meaning of specific colors in Dust RGB imagery, using example references such as that in Figure 1 ("Dust RGB Quick Guide"). They manually interpret individual images to determine dust, fog, and cloud conditions in a given area.

Methodology/Approach

The first step of this project was to devise an approach to map Dust RGB pixel colors to specific climate conditions. Using reference papers (Fuell et al.), I identified seven representative colors corresponding to different climate conditions (Figure 2). Initially, I categorized pixel colors by finding the Euclidean distance between their RGB values and the seven reference colors, assigning each pixel to the closest match. However, this approach was flawed because the RGB color space is not perceptually uniform, which means that Euclidean distances in the RGB space do not correspond to human perception of color differences. To address this, I switched to using the *L*a*b** color space, which is perceptually uniform and models human color perception. In this space, each color has three components. L* represents lightness, a* represents the red-green axis, and b* represents the blue-yellow axis ("Cielab Color Space"). I used the CIEDE2000 formula to calculate differences between colors in the L*a*b* space. The CIEDE formula adjusts

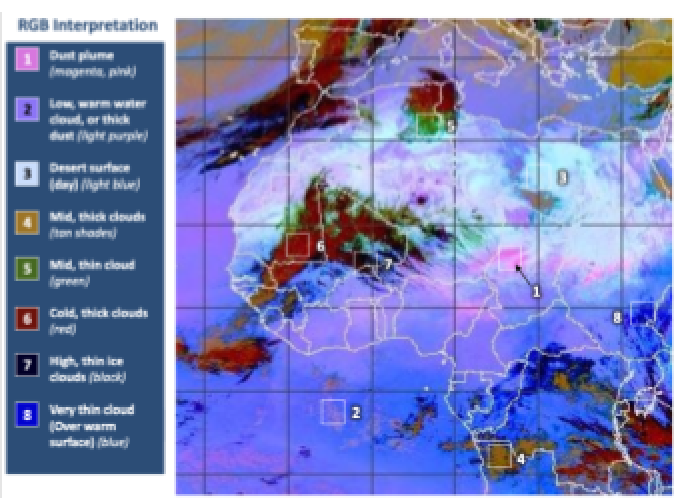

Figure 1. Example of a Dust RGB interpretation guide (GOES-R Dust RGB Quick Guide)

Feature	Red Pixel Value	Green Pixel Value	Blue Pixel Value
Dust aloft	235	50	175
High, thick cloud	150	5	5
High, thin cloud	5	1	5
Mid, thick cloud	140	75	15
Mid, thin cloud	15	90	20
Low, thick cloud (warm climate)	170	130	190
Low, thick cloud (cold climate)	170	130	50

Figure 2. Representative RGB values for 7 features within the *Dust RGB* product (Fuell et al).

for lightness, chroma, hue, and neutral tones, providing a precise measure of perceptual color differences (Katowitz).

Figure 3 shows a 3D plot of the L*a*b* color space created with Plotly in Python. Using *rgb2lab* from *skimage*, I converted the RGB values of the seven reference colors to L*a*b*. Then, I used the *CIEDE2000* function from *pyciede2000* to compute the color difference between each point in the space and the reference colors, assigning each point to the color of its closest match. This visualization demonstrates how any L*a*b* color can be classified into one of the seven dust condition categories.

I used *PyQt5* to build an interactive Dust RGB viewer (Figure 4) with input boxes for specifying timestamps and regions. The backend retrieves a Dust RGB image from the Meteomatics API and displays it. When a user clicks a pixel in the image, the program extracts its color, converts it to L*a*b*, and classifies it by finding the closest of the seven representative colors using the CIEDE2000 formula. Pixels with a CIEDE2000 difference over 20 from their nearest match are marked as unclassified, as such differences indicate significant visual disparity.

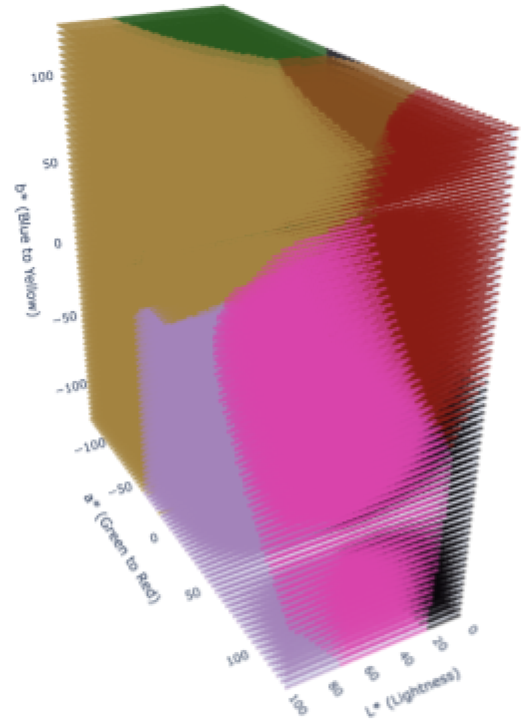

Figure 3. 3D visualization of the Lab* color space, with each point assigned the color of its closest match among the 7 representative colors.

Results

The application performs as expected, with its pixel classifications aligning with my manual interpretations based on the Figure 1 reference. Although I originally considered using multiprocessing to precompute classifications for all pixels, it proved unnecessary since real-time categorization on user clicks was instantaneous. Since there is no existing composite dataset of the dust, cloud, and fog conditions, my color interpretations cannot readily be validated against any observed climate data. My application is a first step towards automating the interpretation of Dust RGB imagery, and evaluating it would require a meteorologist to validate the tool's color interpretation against their own interpretation.

Discussion

This tool represents a first step in automating Dust RGB interpretation, allowing non-experts to access and understand satellite imagery while enhancing meteorologists' efficiency. If I had more time and resources, I would make two improvements to my application. First, I would integrate other climate datasets containing quantitative observational climate data into my program and would use that as supplemental information to aid in the program's interpretation of each pixel color in the image. Additionally, I would get feedback from meteorologists on how to improve the climate interpretation algorithm. These next steps could lead to a novel application that could be commercialized and used by researchers across the field of meteorology.

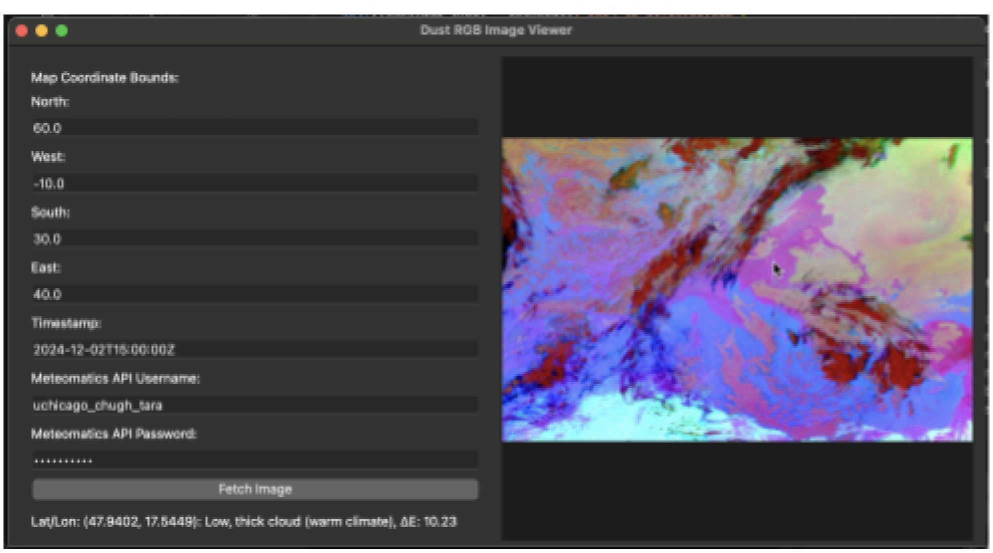

Figure 4. Screenshot of my application. The left panel has input boxes for the user to specify the image they want to retrieve. The right panel shows the retrieved Dust RGB map. The mouse is clicking a point on the map, and that point's coordinates, interpreted climate condition, and CIEDE ΔE value is shown on the bottom left.

References

“Cielab Color Space.” *Wikipedia*, Wikimedia Foundation, 22 Nov. 2024, en.wikipedia.org/wiki/CIELAB_color_space.

“The Dirt on Atmospheric Dust.” *National Environmental Satellite, Data, and Information Service*, 22 July 2016, www.nesdis.noaa.gov/news/the-dirt-atmospheric-dust.

“Dust RGB Quick Guide”, *NOAA & NASA SPoRT*, www.weather.gov/media/crp/QuickGuide_DustRGB_NASA_SPoRT.pdf. Accessed 3 Dec. 2024.

Fuell, Kevin, et al. “Next Generation Satellite RGB Dust Imagery Leads to Operational Changes at NWS Albuquerque.” *Journal of Operational Meteorology*, 9 Feb. 2016, nwafiles.nwas.org/jom/articles/2016/2016-JOM6/2016-JOM6.pdf.

Katowitz, Josh. “Demystifying the CIE ΔE 2000 Formula.” *Techkon USA*, 6 Mar. 2024, techkonusa.com/demystifying-the-cie-%CE%B4e-2000-formula/.

“Multispectral Satellite Applications: RGB Products Explained.” *Geostationary Operational Environmental Satellites—R Series*, NOAA & NASA, 2010, web.archive.org/web/20171129162511/www.goes-r.gov/users/comet/npoess/multispectral_topics/rgb/contrib.htm.