A Realtime Camera Fusion 3D Model with a Novel Feature-Matching and Star Identification-Based Calibration for Tracking Smoke Plumes

New Mexico

Supercomputing Challenge

Final Report

April 9th, 2024

V. Sue Cleveland High School

Team Members:

Gene Huntley

Teacher(s):

Ashli Knoell, Leah Felty

Project Mentor

Stephen Guerin

Contents

Executive Summary

Smoke plumes produced by wildfires pose a threat to public safety. However, a lack of accurate real-time information on smoke plumes provided by current methods like satellite imagery leaves communities unable to make informed decisions critical to ensuring the safety of themselves and their families. Despite the current lack of smoke plume information from satellites, there is a lot of photography of smoke plumes. This project aims to provide an accurate and quick method to locate smoke plumes through smartphone cameras to utilize this untapped source of information. The research aims to find a method to calibrate a camera using manual and automatic selections of similar or known locations, such as celestial objects. After calibration, the proposed method renders a real-time 3D smoke plume with planar-to-planar correspondences. This algorithm proves to be significantly faster than current satellite imagery methods (MODIS and HMS) while providing an improved level of granularity than current surveillance methods.

Introduction

Problem

An issue facing urban and rural populations is a lack of reliable smoke plume surveillance, leaving communities and local authorities with no information on dangerous smoke that can harm them. Specifically, climate change has increased global temperatures, drastically increasing the length and duration of wildfire seasons (McKENZIE et al., 2004). As a result of an increase in the frequency of wildfires, there has also been a significant increase in the number of smoke plumes emitted by said wildfires. This increase can pose a risk to public safety due to the emission of toxic particles such as PM2.5 by smoke plumes, causing chronic illnesses such as asthma and acute heart attacks (Haikerwal et al., 2015; Dennekamp and Abramson, 2011). Furthermore, a lack of information on smoke plumes can harm federal authorities conducting prescribed burns. Without information on the location and spread of smoke plumes, prescribed burns - necessary for understanding wildfires and clearing vegetation buildups - can threaten public health (Wu et al., 2023). Smoke plumes not only have the potential to cause death from poor air quality, but it also affects a large number of people; in the United States alone, 1.6 million people live in areas with dangerous air quality caused primarily by smoke plumes (Kelishadi and Poursafa, 2010). Despite the dangers smoke plumes can pose to public health, there is a severe lack of reliable real-time smoke plume information. Thus, the research aims to create a network of phone cameras to render a smoke plume and provide real-time surveillance information on the aforementioned smoke plumes.

The most widely used method to gather information on smoke plumes is through remote sensing, a process that collects visual information on orbital bodies like Earth (Earth Science Data Systems, 2019). One of the most extensive smoke plume algorithms that analyzes remote sensor information for smoke plumes is the National Oceanic and Atmospheric Administration's (NOAA) Hazard Mapping System (HMS) (Data and Service, 2019). In addition, many smoke plume monitoring systems also rely on NASA's Moderate Resolution Imaging Spectroradiometers (MODIS) and Visible Infrared Imaging Radiometer Suite (VIIRS) mounted on various satellites (NASA, 2000; Earth Science Data

Systems, 2019; Xie et al., 2007). However, a barrier that prevents organizations from utilizing remote sensing to monitor the spread of smoke plumes is the inability of remote sensors to capture images with sufficient resolution and speed. According to NASA, the resolution of MODIS is 1km, while the resolution of NOAA's HMS is 5km (Earth Science Data Systems, 2019; Data and Service, 2019). In addition, the time granularity of the remote sensors - the time between each capture - is a day, causing tracking small to medium smoke plumes live with remote sensing to be exceedingly challenging (Earth Science Data Systems, 2019).

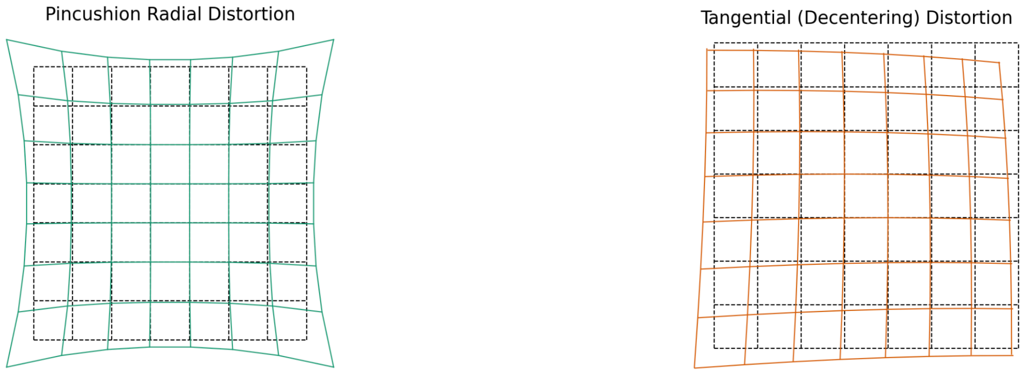
Currently, 93% of the global population owns cell phones, most of which are capable of video and image recording (Center, 2022). Accompanied by the massive increases in the processing capabilities of cell phones, smartphones can collect and analyze necessary data for analysis (Muhammad et al., 2018). Therefore, instead of relying on outdated forms of surveillance that require satellites, this research aims to construct a network of phones and other cameras to monitor the spread of smoke plumes (Almalkawi et al., 2010).

The proposed network calibrates each of its cameras, utilizing information ranging from celestial positions to feature identification to obtain the position, rotation, and distortion parameters of the camera. Finally, the algorithm uses visual media from various calibrated cameras to visualize a 3D point cloud or mesh of a smoke cloud, giving information on smoke plume geometry and location. Ideally, the user should be able to take a calibration and rendering image to produce a smoke plume. By relying on community-sourced information, the proposed method overcomes the lack of spatial and temporal granularity provided by remote sensing devices due to its speed and accuracy when creating 3D smoke plumes.

Background

Camera Model

A camera has both radial and tangential distortions, which are distortions that result in a misalignment of an image, as seen in Figure 1. Since lens distortion can cause triangulation errors over large distances, each pixel in the image was adjusted using a 4-parameter Brown-Conrady distortion model, a model that fixes the image distortion given a few distortion parameters (Brown, 1966).


Pincushion Radial Distortion
Tangential (Decentering) Distortion

(a) Radial Distortion

(b) Tangential Distortion

Figure 1: Distortion Examples, Source: Steward, Jeremy. “Symmetric Distortions.” Camera Modeling: Exploring Distortion and Distortion Models, Part I, TangramVision, 6 Aug. 2021, www.tangramvision.com/blog/camera-modeling-exploring-distortion-and-distortion-models-part-i.

Examples of image plane offset as a result of distortion.

Every pixel on an image plane has a position relative to the rest of the image defined as (u, v), which a projection matrix converts into 3D coordinates given known camera rotation and position parameters (OpenCV, 2021). Given only the (u, v) pixel coordinates are known, the rest of the paper will focus on solving for the unknown variables presented in the camera model needed to render a smoke plume. A camera with arbitrarily set rotation, distortion, and translation parameters and a fixed depth for each vertex can be seen in Figure 2.

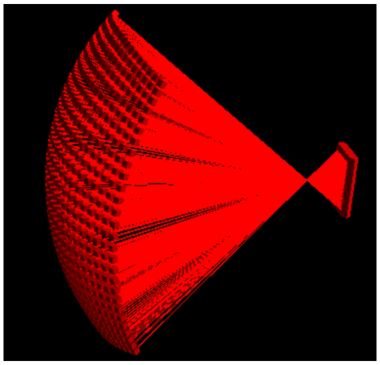


Figure 2: Simulated Camera

A visualization of a camera with each pixel projected at a fixed depth.

Levenberg-Marquardt and ECEF Coordinates

Levenberg-Marquardt algorithm, a popular variation of gradient descent commonly used to approximate the local minimum of a function, will be a heavily relied-upon algorithm (Gavin, 2022; Moré, 1978). Each iteration of the Levenberg-Marquardt increases each parameter by β_n, where w_n is a parameter such as position or tilt, and λ is a constant that is changed during each step of the algorithm:

$$J = \begin{bmatrix} \frac{\delta w_1}{\delta E_1(x)} & \cdots & \frac{\delta w_1}{\delta E_n(x)} \\ \dots & \dots & \dots \\ \frac{\delta w_m}{\delta E_1(x)} & \cdots & \frac{\delta w_m}{\delta E_n(x)} \end{bmatrix} \tag{1}$$

$$\beta = (J^T J + (\text{diag}(J^T J)\lambda))^{-1}(E_n(x)J^T) \tag{2}$$

To calibrate a camera, ideally, the research would need an error function, $E(x)$, as described in the above equation, such that the perfect calibration of a camera results in the minimum of said error function.

The Earth-Centered-Earth-Fixed coordinate system, a coordinate system that uses the center of the Earth as the origin is used heavily (Zhu, 1994). Although a latitude-longitude-elevation coordinate system produces negligible error at short distances, not accounting for the Earth's curvature at long distances can result in errors. To convert local camera coordinates into ECEF coordinates, the below change-in-basis is used, where $< X, Y, Z >$ is the ECEF coordinate of the focal point, and $< x, y, z >$ represents the local coordinate of a pixel (Smith, 2022).

$$\vec{a} =< X, Y, Z > \tag{3}$$

$$\vec{b} = \vec{a}\cdot < 0, 0, -1 > \tag{4}$$

$$\vec{c} = \vec{a} \cdot \vec{b} \tag{5}$$

$$f(x, y, z) = \vec{a}x + \vec{b}y + \vec{c}z \tag{6}$$

Hypothesis/Research Goal

This research aims to use cellphones and other devices capable of multimedia recording to create a 3D representation of a smoke plume, utilizing a multiple methods for convenient calibration and deployment of cameras.

Methodology

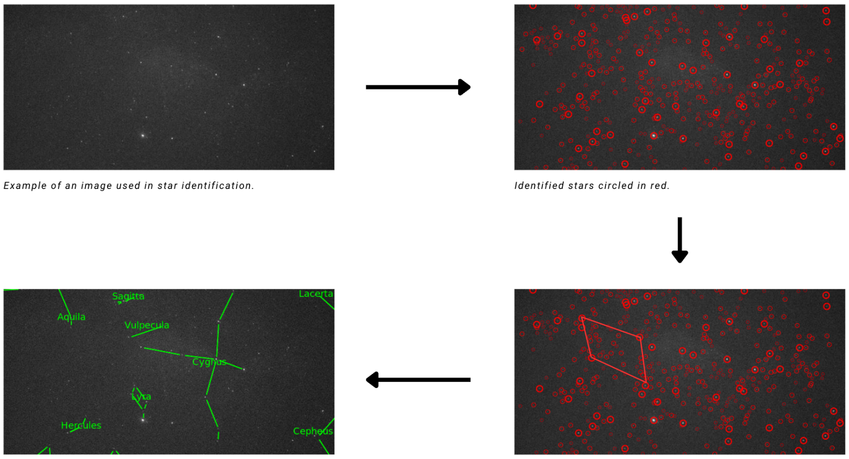
Calibration

Before any operations can extract information on a smoke plume's location, a camera's distortion, translation, and rotation parameters need to be known.

Star Identification

One method to obtain $E(x)$ would be to use an error in the projected locations of stars in the night sky and their respective celestial positions. The proposed calibration algorithm employs a baseplate solver - an algorithm that identifies stars in an image - to identify the locations of the stars in the night sky, as shown in Figure 3.

After the algorithm identifies the (u, v) positions and names of certain stars in an image, it references the star's current right ascension and declination (the star's relative position to the Earth) from Harvard's Astrophysics Data System for information. Afterward, the algorithm uses a projection matrix with arbitrarily set camera parameters to convert the stars' (u, v) coordinates into ECEF coordinates. Levenberg-Marquardt then attempts to



Example of an image used in star identification.

Identified stars circled in red.

Identified stars circled in red.

Identified stars.

Figure 3: Baseplate Solver

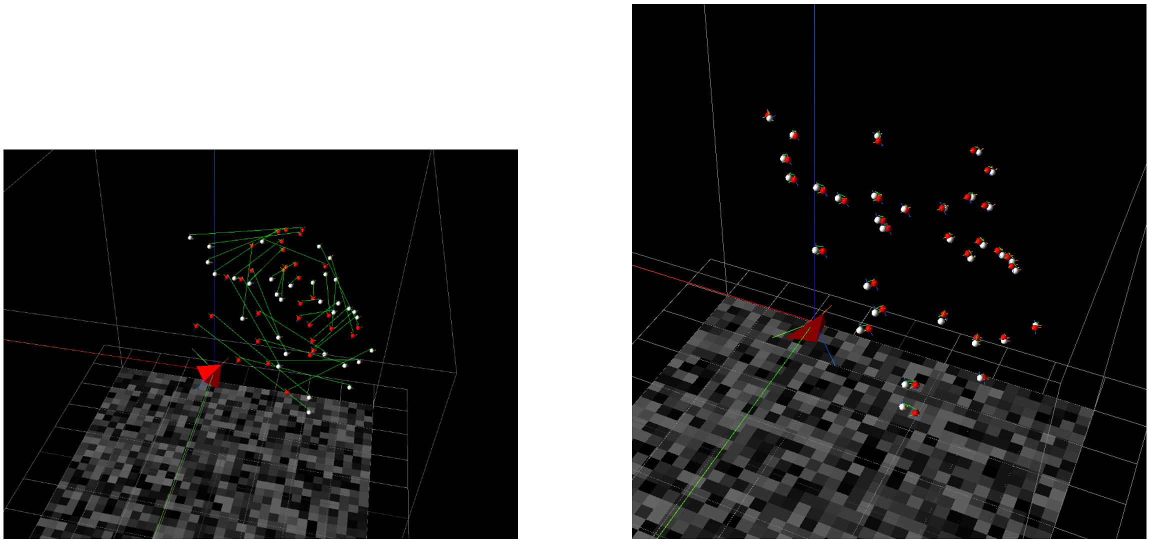
A brief description of how a baseplate solver identifies stars.

find the local minimum of $E(x)$, where $E(x)$ is defined as the squared altitudinal and azimuthal difference between the projected and actual positions of the stars. Figure 4 shows the Levenberg-Marquardt algorithm in use to reduce the error between the altitude/azimuth of the projected stars and their actual positions.

Point Correspondence Calibration

For situations where the stars in the night sky aren't visible, another powerful method of calibration is through point matching. In essence, a user selects two corresponding points, one in an image plane and another in a separate image plane or the real world.

The former method, the image-to-image point correspondence, is described as follows: within two arbitrary images, matching pixels are selected automatically or through user input. Those pixels are projected with the projection matrix into the real world at an infi-



(a) Projected Stars before Optimization

(b) Solved Camera Parameters

Figure 4: Solved Stars

An example of a camera with calibrated and uncalibrated parameters.

nite radius. After the two projected pixels are created for each correspondence, the error function is defined as follows, where (x, y, z) describes the position of the pixel on the image plane as a function, and (a, b, c) describes the position of the focal point:

$$r_n^2 = x_n^2 + y_n^2 + z_n^2 \tag{7}$$

$$q = x_1 x_2 + y_1 y_2 + z_1 z_2 \tag{8}$$

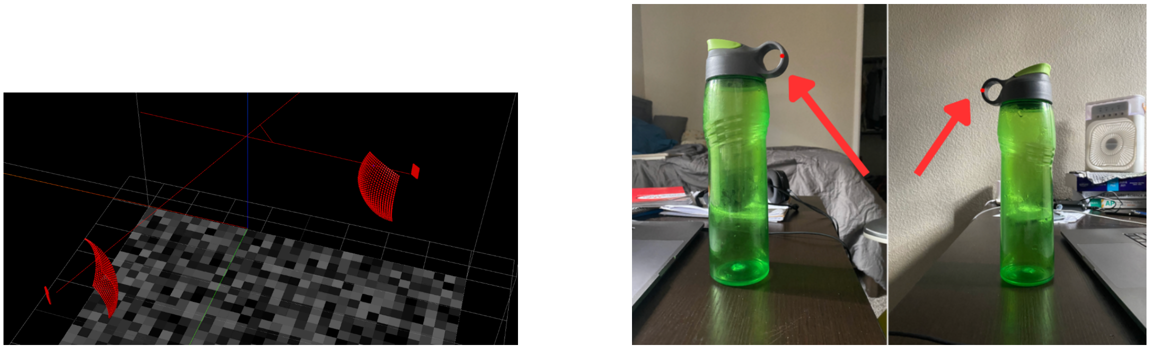
$$\Delta n = n_2 - n_1 \tag{9}$$

$$\begin{bmatrix} t_o \\ s_o \end{bmatrix} = \begin{bmatrix} r_1^2 & -q \\ q & -r_2^2 \end{bmatrix}^{-1} \begin{bmatrix} \Delta a x_1 + \Delta b y_1 + \Delta c z_1 \\ \Delta a x_2 + \Delta b y_2 + \Delta c z_2 \end{bmatrix} \tag{10}$$

$$f(x, a) = (s_o x_2 + a_2 - t_o x_1 + a_1)^2 \tag{11}$$

$$E(s, t) = \sqrt{f(x, a) + f(y, b) + f(z, c)} \tag{12}$$

The model of the camera function with the calculated errors represented as a red line between the pixel projections is shown in Figure 5.



(a) Projected Points and Error

(b) Selected Points

Figure 5: Multiple-Image Calibration

An example of how multiple images can be used to calibrate a camera.

Note that if the camera is perfectly calibrated, the error function will be at its absolute minimum. After all, for a perfectly positioned camera in the network, two lines drawn by the projected pixels should intersect at the pixel's position. Thus, Levenberg-Marquardt is run on the error function to calibrate a camera.

A similar method to using two points on separate image planes is to utilize a correspondence between a real-world feature and an image correspondence. The method is similar to the previously described image-to-image correspondence. However, instead of projecting two lines from two cameras, one line is projected from the camera to the image plane, while another is projected from the center of the Earth to the selected point. An example is shown in Figure 6.

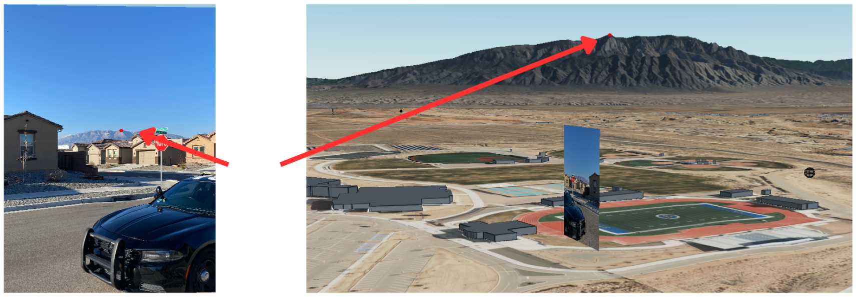


Figure 6: Simulated Camera

An example of how a single match between an image and a real-world correspondence can calibrate a camera.

Rendering

With all the necessary information about a camera, such as its position, rotation, and distortion, an algorithm can extrapolate 3D information from an image's data.

LoFTR and Manual Pixel Matching Tracking with the Lukas-Kanade Method

Before any image matching can be done, Meta's Segment-Anything-Model segments all the objects in an image (Kirillov et al., 2023). After segmentation, the user should select the approximate location of the smoke plume to crop the mask from the rest of the image, as shown in Figure 7. To plot out pixel matches and render a smoke plume, the researcher then applies LoFTR, an algorithm that extracts and matches features in an image, to find pixel matches in smoke plumes (Sun et al., 2021). An example of LoFTR in action is seen in Figure 8. Given the homogeneity of clouds and smoke plumes, there are bound to be many false positives. Therefore, a pixel match is only accepted if the algorithm is more than 50% confident of its accuracy.

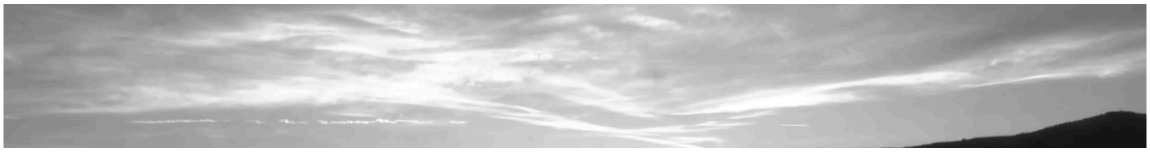


Figure 7: Cloud Cropped with SAM

A cropped cloud manually selected by a user after image segmentation by SAM.

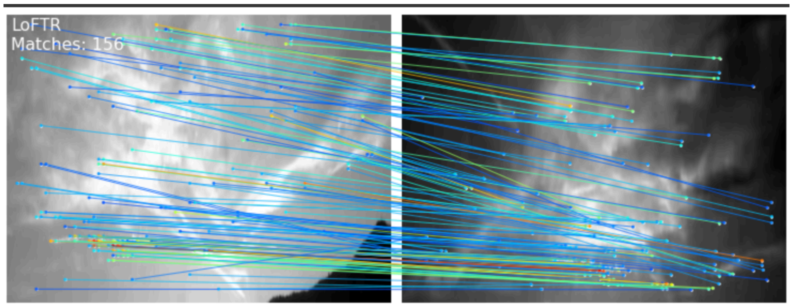


Figure 8: Pixel Matches with LoFTR

Each line drawn between two pixels is a correspondence between the two images.

When LoFTR fails for translucent figures with undefined features, manual annotation of pixel correspondences can be used, as shown in Figure 9.

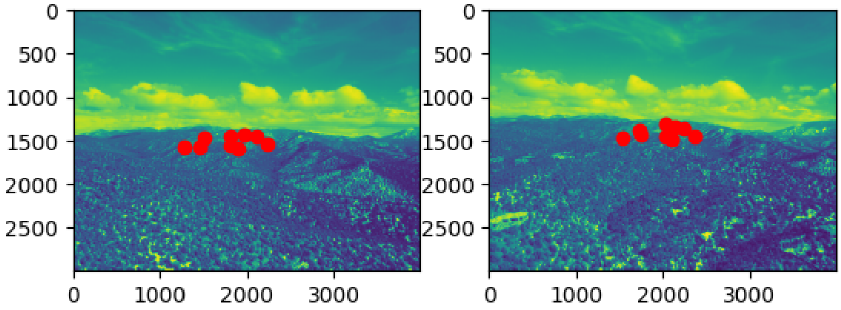


Figure 9: Manually Annotated Pixel Matches

Pixel matches were manually selected instead of utilizing LoFTR and SAM.

Most of the time, the visual media provided by smartphone cameras isn't static; instead, most of the information provided spans thousands of frames in a video format. Running both SAM and LoFTR or requiring manual annotations to track the dispersion of a smoke plume for each frame would be impractical. Therefore, after pixel correspondences have been identified in a singular frame, the Lucas–Kanade method - a method for tracking the movement of pixels in a video - is run on an initial set of point correspondences to track their location in each subsequent frame (Lucas and Kanade, 1981).

Sparse Depth and Depth Completion with DeLaunay Triangulation

Having a pixel match implies that it is visible in images A and B, providing information for triangulation. Two lines extending outwards into infinity, defined as $\vec{y_1}(t) = \vec{p_{c1}} + (\overrightarrow{p_{\text{pix1}} - p_{c1}})t$ and $\vec{y_2}(s) = \vec{p_{c2}} + (\overrightarrow{p_{\text{pix2}} - p_{c2}})s$, where p_{pix} is the position of the pixel calculated by the projection matrix at an arbitrary radius, and p_c is the position of the camera, is drawn between the two cameras and the two corresponding pixel matches. Since the calibration algorithm isn't infinitely precise, there are bound to be some errors, implying the two lines will not intersect. Thus, a vector $\vec{P}$ is drawn between $\vec{y_1}$ and $\vec{y_2}$ such that

$$\vec{P} \cdot \vec{y_1} = 0 \tag{13}$$

$$\vec{P} \cdot \vec{y_2} = 0 \tag{14}$$

To derive the location of P, s_o and p_o are derived from equations (7) - (10), scaling $\vec{y_1}, \vec{y_2}$ accordingly. The location of the pixel can be anywhere on line P; however, the assumption that the two cameras are equally inaccurate leads the pixel's location to be at the middle of the vector, at a point defined at p, defined as $p = \frac{P}{2}$. Using the described methods, the matching pixels from both images in Figure 8 formed Figure 10. A triangulation algorithm (an algorithm that creates as many triangles as possible given a set of points) - DeLaunay Triangulation - generates a 2D mesh of an object with its corresponding pixel matches to fill in the depth of the rest of the pixels (Lawson, 1972). Figure 11 shows an example of DeLaunay triangulation used on the pixel correspondences in Figure 8.

Afterward, each point in the DeLaunay mesh splits the triangle it is contained by into three separate triangles, as shown in Figure 12. An arbitrary point P has its depth calculated by the ratio using the following formula, where points A, B, C are the points of a DeLaunay-generated triangle containing point S, d_n denotes the depth of point n, and $\triangle XYZ$ denotes

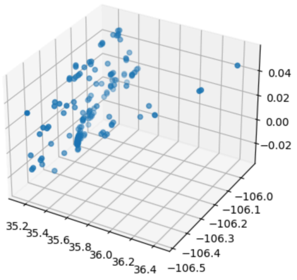


Figure 10: Sparse Depth of Cloud

Cloud triangulated with purely pixel matches.

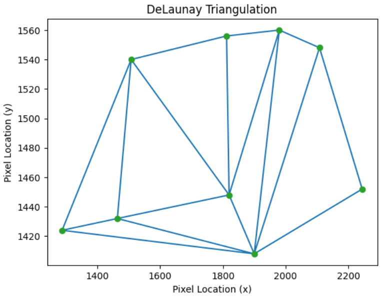


Figure 11: DeLaunay Triangulation

An example of DeLaunay being used on point correspondences.

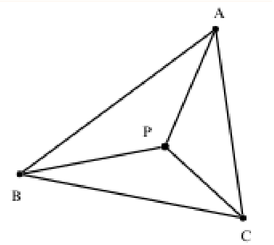


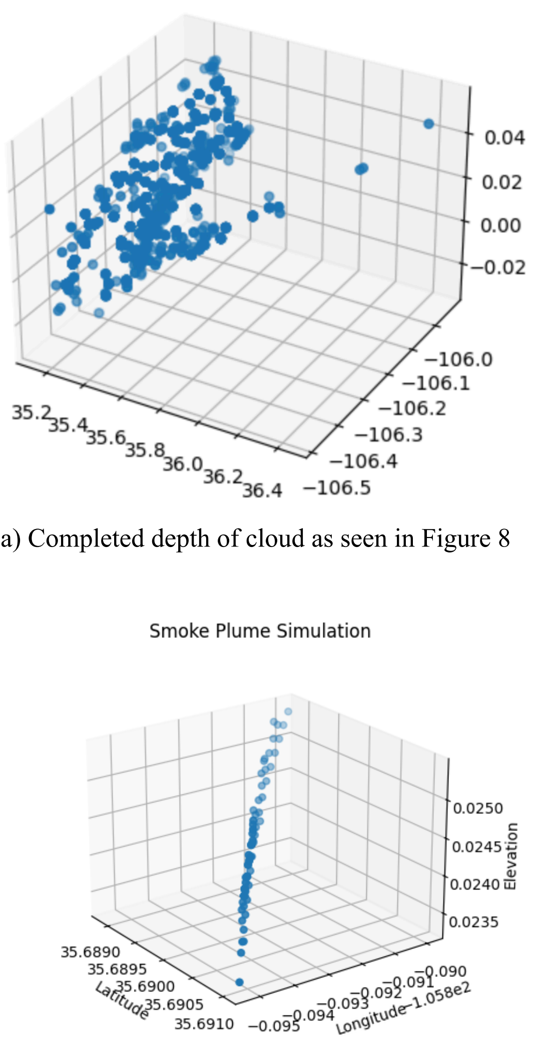
Figure 12: Triangle Split

An example of how a single match between an image and a real-world correspondence can calibrate a camera.

the area of a triangle containing points X, Y, Z:

$$d_P = \frac{\triangle BPC d_A + \triangle APC d_B + \triangle APB d_C}{\triangle ABC} \tag{15}$$

The results of a completed depth point cloud can be seen in Figure 13.



(a) Completed depth of cloud as seen in Figure 8

(b) Completed depth of plume as seen in Figure 9

Figure 13: Completed Depth Rendering (Granularity Reduced by 400 times for Visualization)

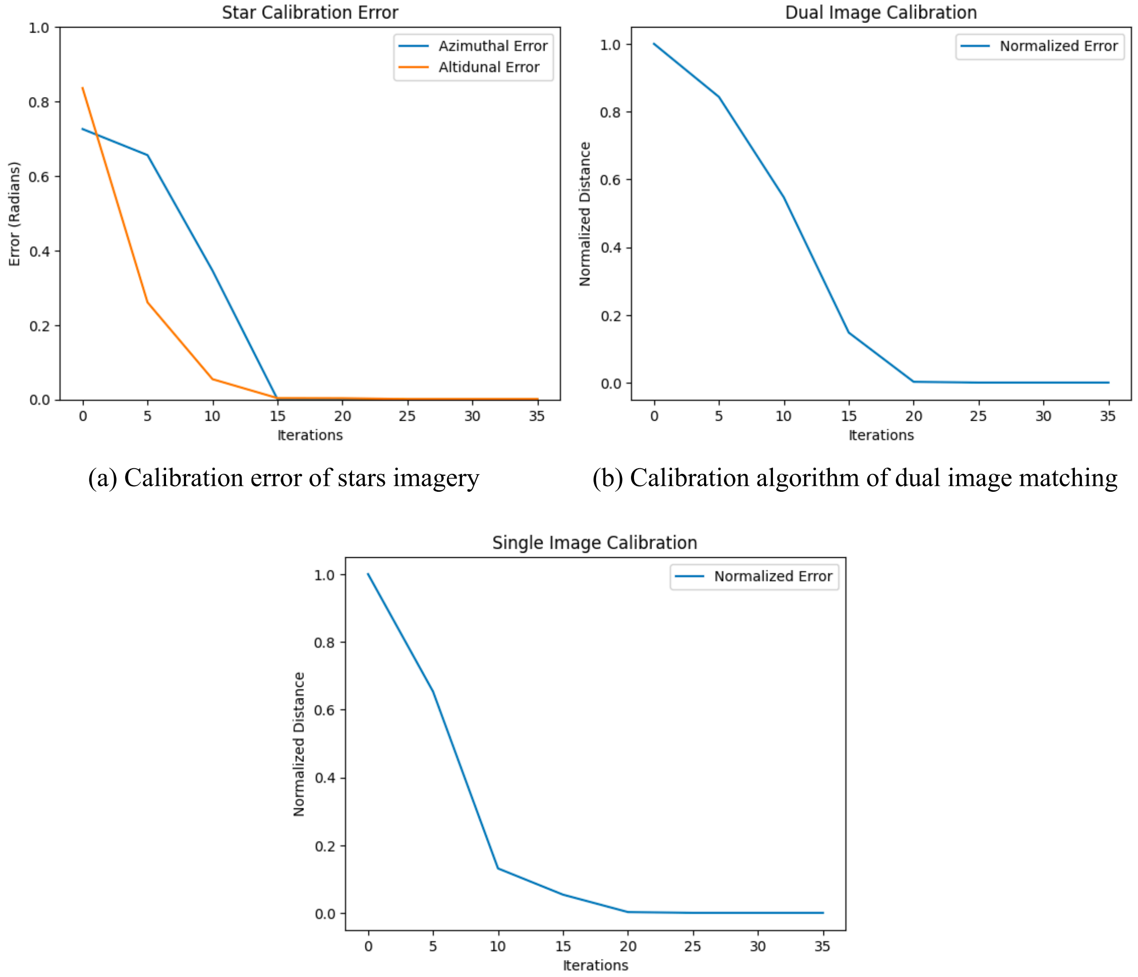
Results and Examples

Results

The calibration algorithm was tested with several images of stars and geographic features taken by the researcher. The collected pictures consisted of 20 images for each calibration method, all taken within the general Southwestern vicinity. For the star calibration algorithm, in under 15 iterations of the Levenberg-Marquardt algorithm, the azimuthal error between the projected position of the star and its actual position dropped from 0.725rad to $3.7 * 10^{-3}$rad, while the altitudinal error dropped from 0.835rad to $4.9 * 10^{-4}$rad. Meanwhile, in 20 iterations, the dual image calibration lowered the normalized error of 1 to 0.00346. Finally, in under 25 iterations, the single image calibration method lowered the normalized error of 1 to 0.05428.

The results demonstrate the speed at which the algorithm can solve position, rotation, and distortion parameters. However, a weakness of the above calibration method is its need for manually-annotated point correspondences. On rare occasions, users might not have access to distinct geographic features like a mountain range or a building.

The rendering algorithm was tested with over 50 smoke plume renderings. The smoke plume imagery was taken from 10 cameras from the ALERTCalifornia camera system and through SimTable's VAPIX cameras, capturing a total of 5 different wildfire incidents (wildfires from Aztec Springs, Blue Creek, El Dorado, Del Norte, and Tulare). The photos were taken by the user using VAPIX cameras mounted on buildings in Los Alamos and Santa Fe, and a few were provided by drone footage captured by SimTable, the researcher's sponsor. The first result compares the granularity of NASA's thermal imagery using smoke plumes to the proposed algorithm's accuracy. Random clusters of triangulated points were extracted from each 3D rendering, and the average granularity - or the average distance between each particulate and its closest neighbor - was calculated.



(a) Calibration error of stars imagery

(b) Calibration algorithm of dual image matching

(c) Calibration algorithm of single image matching

Figure 14: Graph of Results

Not only was the resolution of the algorithm tested but the runtime of the algorithm was also tested. Testing the algorithm on 10 randomly selected smoke plumes revealed that it is capable of updating smoke plumes significantly faster than current technologies. Current remote sensing satellites capture wildfire progression with a 1 day time resolution, while

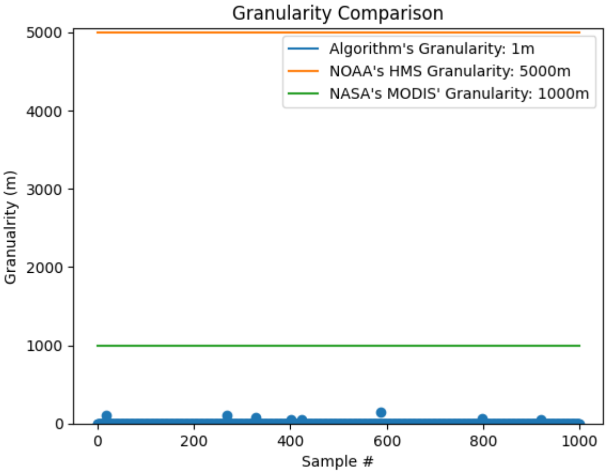


Figure 15: Smoke Plume Granularity

The average distance between each particulate from 1000 random samples.

the proposed algorithm can update a smoke plume's mesh in 5.27 seconds (Earth Science Data Systems, 2019; Data and Service, 2019). The results are shown in Figure 16. Finally, the accuracy of the model was calculated. Rough polygons of the smoke plume were cre-

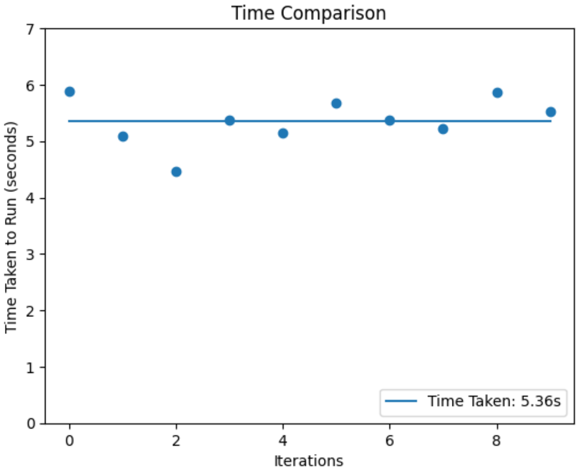


Figure 16: Algorithm Runtime

How long it takes, on average, to run the algorithm.

ated using aerial and remote-sensing photography and split up into individual pixels. For each smoke plume particulate, the average distance between said plume and the nearest pixel was calculated. The results can be seen in Figure 18.

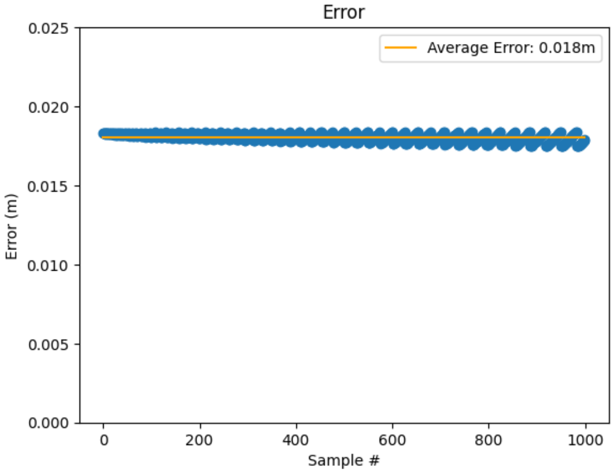


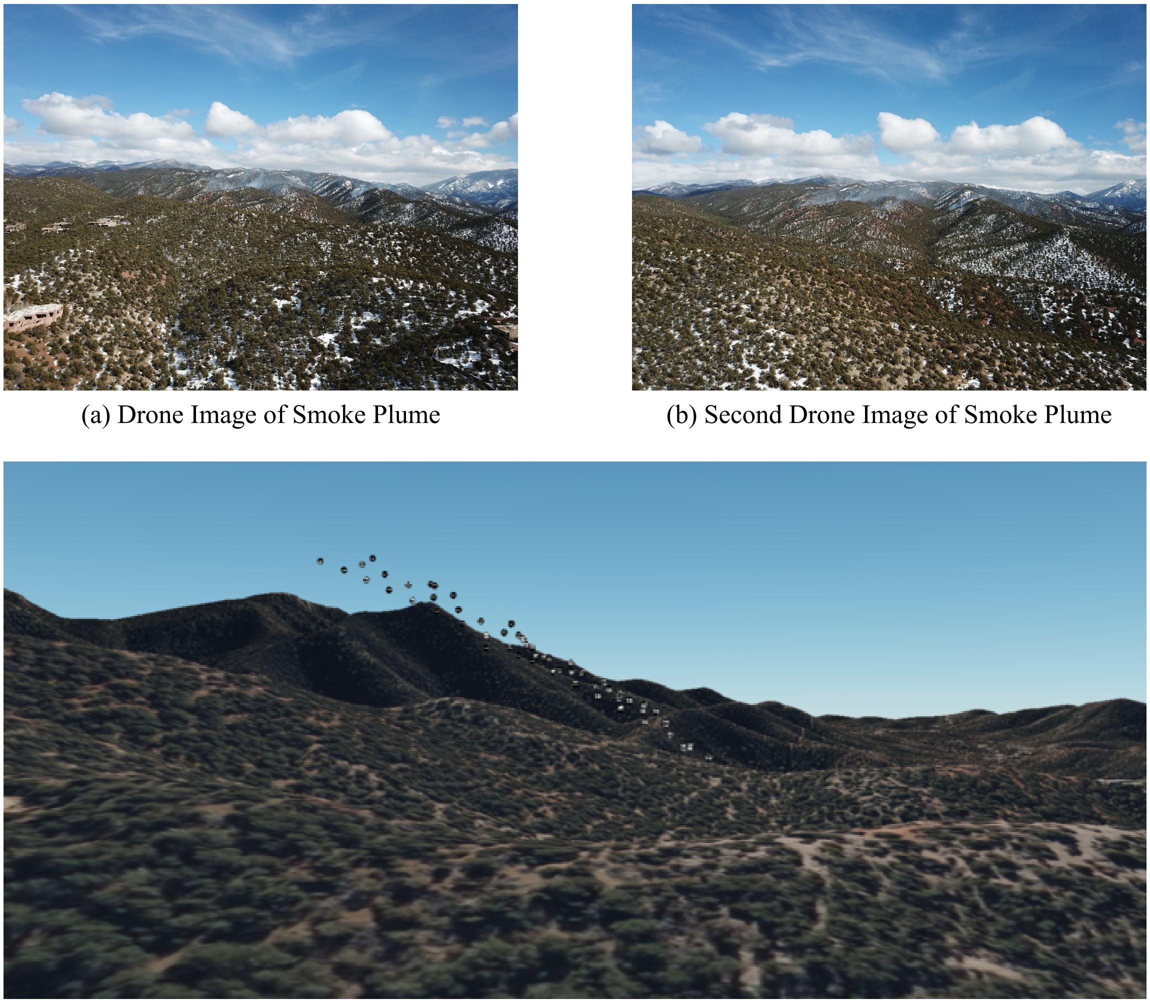
Figure 17: Algorithm Validation

Distance of particulate from its possible values.

The method mentioned above of testing for accuracy is extremely faulty, as low-resolution images of smoke plumes cause many smoke plume particulates to be compared to the same coarse pixels.

Rendering Example

The final product should be an application on a user's device that allows them to contribute to a database of smoke plumes. Ideally, the user should take a picture of an image used for calibration and another for rendering. If another user has also taken the following steps, they should be able to render a 3D model of a smoke plume. An example of this application in a prescribed burn can be seen in Figure 18.



(a) Drone Image of Smoke Plume

(b) Second Drone Image of Smoke Plume

(c) Plume Render with 400x Reduced Granularity

Figure 18: Aztec Spring Prescribed Burn Rendering

Discussion/Conclusion

Discussion

As mentioned in the results, the average distance between each smoke plume particulate is, on average, 1m, implying that it can render a smoke plume at a 1x1m resolution. On the other hand, NASA's MODIS has a resolution of 1000x1000m, while NOAA's HMS has a granularity of 5000x5000m (Earth Science Data Systems, 2019; Data and Service, 2019). Most smoke plume and fire monitoring algorithms rely heavily on MODIS or NOAA's HMS, implying that the training and validation of said models rely on low-resolution surveillance data, failing to note granular geographic variations such as buildings (van Donkelaar et al., 2021; Hammer et al., 2020; Wooster et al., 2013). Thus, with a more granular database of smoke plume dispersion over a certain time interval, an improvement in the granularity of fire and smoke plume surveillance would allow for more precise prediction of fire hotspots, smoke dispersion patterns, and air qualities.

For a rapidly spreading wildfire, time resolution is crucial for evacuation and containment. In addition, rapdily changing atmospheric conditions - such as pressure, wind, or even fire perimeter spread - can drastically affect the dispersion of a smoke plume (Kelishadi and Poursafa, 2010). Therefore, a daily, low-resolution update on a smoke plume can prove insufficient for certain situations. The proposed algorithm not only overcomes the lack of granularity provided by remote sensing tools, but also updates information on the smoke plume dispersion significantly faster. In addition, since most warning systems validate their data on an hour-to-hour or even day-to-day dataset taken MODIS or HMS, a smoke plume dispersion dataset that updates within seconds can provide crucial information to communities regarding evacuation or quarantine.

Finally, the results of the algorithm's accuracy show that it can identify the location of a smoke plume pixel with an average error of 0.02m, proving that the algorithm is significantly better than current smoke plume surveillance technologies and extremely accurate. As a result, if implemented on a large scale, the model can confidently report on the location and distribution of a smoke plume, informing local communities of air pollutants that may cause them harm accurately.

Recommendations

Currently, only a theoretical prototype has been developed. In the future, all the rendering and calibration should be accessible and easy to use. To do so, the researcher needs to take two steps. First, the researcher should streamline the theoretical prototype into a single program and deploy it on a web server. Finally, the researcher should create an interface where users can take images for rendering or calibration. Note that the only costs associated with the project would be to host a webserver to run computations. Otherwise, the proposed method is completely free to implement and maintain for agencies worldwide.

The biggest limitation of this project is a potential lack of surveillance. At times, there might not be anyone with a cellphone to capture a smoke plume, especially if a fire is burning in the wild. In addition, if weather conditions limit visibility, using visual resources to detect smoke plumes may prove challenging, if not impossible. Finally, a community-sourced surveillance method may allow people to exploit the system through false visual cues or cyberattacks, which may cause casualties.

Conclusion

In conclusion, the paper presents a significant step forward in leveraging technology to address the pressing challenges posed by wildfires. While current smoke plume surveillance methods, such as remote sensing capabilities, suffer from drawbacks like speed and resolution, the proposed program overcomes such obstacles, improving wildfire surveillance technology by leveraging the abundance of cell phones. Such advantages can offer greater validation and training to cutting-edge smoke plume prediction models, causing an explosion in the accuracy of many smoke plume models.

Acknowledgement of Major Assistance

The research was aided by Stephen Guerin and his company, SimTable. He gave the researcher devices - drones and VAPIX cameras - to gather data, and advice to the student regarding testing and debugging. The research was also aided by Leah Felty (teacher sponsor), Ashli Knoell (teacher sponsor), and Cole Monical (science teacher), all of whom assisted in reviewing the paper and materials.

Bibliography

Almalkawi, I. T., Guerrero Zapata, M., Al-Karaki, J. N., & Morillo-Pozo, J. (2010). Wireless multimedia sensor networks: Current trends and future directions. *Sensors*, *10*(7), 6662–6717. https://doi.org/10.3390/s100706662

Brown, D. (1966). Decentering distortion of lenses.

Center, P. R. (2022, December). Internet, smartphone and social media use. https://www.pewresearch.org/global/2022/12/06/internet-smartphone-and-social-media-use-in-advanced-economies-2022/

Data, N. E. S., & Service, I. (2019, May). Noaa’s office of satellite and product operations. https://www.ospo.noaa.gov/Products/land/hms.html#about

Dennekamp, M., & Abramson, M. J. (2011). The effects of bushfire smoke on respiratory health. *Respirology*, *16*(2), 198–209.

Earth Science Data Systems, N. (2019, August). What is remote sensing? http://www.earthdata.nasa.gov/learn/backgrounders/remote-sensing

Gavin, H. P. (2022, November). The levenberg-marquardt method for nonlinear least squares curve-fitting problems. https://people.duke.edu/~hpgavin/ExperimentalSystems/lm.pdf

Haikerwal, A., Akram, M., Del Monaco, A., Smith, K., Sim, M. R., Meyer, M., Tonkin, A. M., Abramson, M. J., & Dennekamp, M. (2015). Impact of fine particulate matter (pm 2.5) exposure during wildfires on cardiovascular health outcomes. *Journal of the American Heart Association*, *4*(7), e001653.

Hammer, M. S., van Donkelaar, A., Li, C., Lyapustin, A., Sayer, A. M., Hsu, N. C., Levy, R. C., Garay, M. J., Kalashnikova, O. V., Kahn, R. A., & et al. (2020). Global estimates and long-term trends of fine particulate matter concentrations (1998–2018). *Environmental Science; Technology*, *54*(13), 7879–7890. https://doi.org/10.1021/acs.est.0c01764

Kelishadi, R., & Poursafa, P. (2010). Air pollution and non-respiratory health hazards for children. *Archives of Medical Science*, *4*, 483–495. https://doi.org/10.5114/aoms.2010.14458

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., & Girshick, R. (2023). Segment anything.

Lawson, C. L. (1972). Transforming triangulations. *Discrete Mathematics*, *3*(4), 365–372. https://doi.org/https://doi.org/10.1016/0012-365X(72)90093-3

Lucas, B. D., & Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. *Imaging Understanding Workshop*.

McKENZIE, D., GEDALOF, Z., PETERSON, D. L., & MOTE, P. (2004). Climatic change, wildfire, and conservation. *Conservation Biology*, *18*(4), 890–902. https://doi.org/10.1111/j.1523-1739.2004.00492.x

Moré, J. J. (1978). The levenberg-marquardt algorithm: Implementation and theory. *Lecture Notes in Mathematics*, 105–116. https://doi.org/10.1007/bfb0067700

Muhammad, K., Hamza, R., Ahmad, J., Lloret, J., Wang, H., & Baik, S. W. (2018). Secure surveillance framework for iot systems using probabilistic image encryption. *IEEE Transactions on Industrial Informatics*, *14*(8), 3679–3689. https://doi.org/10.1109/tii.2018.2791944

NASA. (2000). https://modis.gsfc.nasa.gov/about/

OpenCV. (2021). Camera calibration. docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html

Smith, K. E. (2022). https://dept.math.lsa.umich.edu/~kesmith/CoordinateChange.pdf

Sun, J., Shen, Z., Wang, Y., Bao, H., & Zhou, X. (2021). Loftr: Detector-free local feature matching with transformers.

van Donkelaar, A., Hammer, M. S., Bindle, L., Brauer, M., Brook, J. R., Garay, M. J., Hsu, N. C., Kalashnikova, O. V., Kahn, R. A., Lee, C., & et al. (2021). Monthly global estimates of fine particulate matter and their uncertainty. *Environmental Science; Technology*, *55*(22), 15287–15300. https://doi.org/10.1021/acs.est.1c05309

Wooster, M. J., Roberts, G., Smith, A. M., Johnston, J., Freeborn, P., Amici, S., & Hudak, A. T. (2013). Thermal remote sensing of active vegetation fires and biomass burning events. *Thermal Infrared Remote Sensing*, 347–390. https://doi.org/10.1007/978-94-007-6639-6_18

Wu, X., Sverdrup, E., Mastrandrea, M. D., Wara, M. W., & Wager, S. (2023). Low-intensity fires mitigate the risk of high-intensity wildfires in california's forests. *Science Advances*, *9*(45), eadi4123. https://doi.org/10.1126/sciadv.adi4123

Xie, Y., Qu, J., Xiong, X., Hao, X., Che, N., & Sommers, W. (2007). Smoke plume detection in the eastern united states using modis. *International Journal of Remote Sensing*, *28*(10), 2367–2374.

Zhu, J. (1994). Conversion of earth-centered earth-fixed coordinates to geodetic coordinates. *IEEE Transactions on Aerospace and Electronic Systems*, *30*(3), 957–961. https://doi.org/10.1109/7.303772