Interim Supercomputing 2024

Gene Huntley

January 2024

1 Problem

Climate change has increased global temperatures, drastically lengthening wildfire seasons, restructuring local ecosystems by changing the distribution of local wildlife, and endangering certain fragile species (McKenzie). Such wildfires not only affect local wildlife but also harm urban populations; in California, for example, due to the effects of global warming, there has been a major increase in the intensity and number of wildfires (Westerling). Countries like Greece are hit the hardest, with their forestry-dependent economies, primarily due to a failure of federal governments to utilize community resources and cohesion (Thomas; Lynn).

2 Solution

The most efficient solution for combating wildfires would be to utilize the communities of the impoverished regions, a solution many federal governments fail to grasp. An invaluable resource many said communities have are smartphones, which are so prevalent that 93% of citizens across 18 countries, from the United States to Israel, have one (Pew Research). With a few clicks, a citizen should be able to contribute to a database of images containing smoke plumes from different angles, making up a dynamic 3D model of a smoke plume that can be used to track the progress of a wildfire.

3 Progress

3.1 Calibration

To calibrate a camera, I had initially simulated a camera model, which consisted of an image plane and projected pixels. Each camera has rotation, distortion, and translation parameters, which were defined as the following, where α, β, and γ are defined as tilt, roll, and pan, respectively, and x_p is the position of the image plane:

$$\begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

To account for said distortions, each pixel on the image plane should be adjusted using the following formula, where x_n and y_n are defined as the adjusted pixels, k_n is defined as a radial distortion parameter and p_n is defined as a tangential distortion parameter:

$$\begin{aligned} r &= \sqrt{x^2 + y^2} \\ x_n &= x \frac{\sum_{n=1}^{q} k_n r^{2n}}{\sum_{n=1}^{w} k_n r^{2n}} + 2p_1 xy + p_2(r^2 + 2x^2) \\ y_n &= y \frac{\sum_{n=1}^{q} k_n r^{2n}}{\sum_{n=1}^{w} k_n r^{2n}} + p_1(r^2 + 2y^2) + 2p_2 xy \end{aligned}$$

The resulting camera model looks as follows: A method that can be used to calibrate the camera is by comparing the position of certain stars with identified stars in the image with a baseplate solver. The process in which stars are identified can be seen in Figure 2.

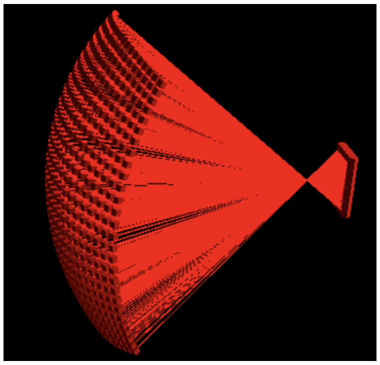
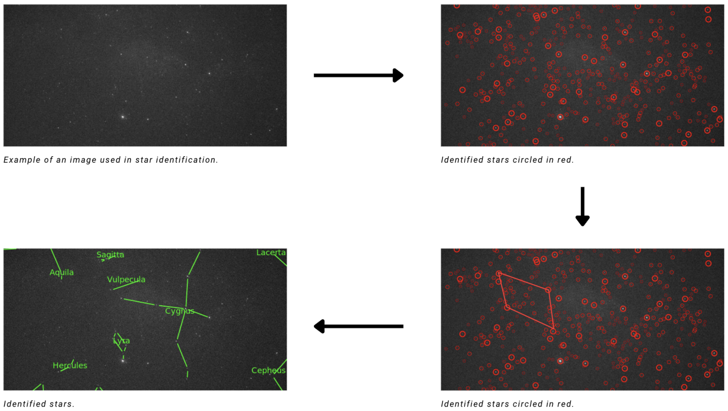


Figure 1: Simulated Camera



Example of an image used in star identification.

Identified stars circled in red.

Identified stars circled in red.

Identified stars.

Figure 2: Process of Identifying Stars/Baseplate Solving

After identifying stars, a variation of gradient descent and quadratic approximation, Levenberg-Marquardt, is used to optimize the camera parameters in order to reduce the error between projected points and the stars. Notice that for each step, the weight is adjusted by the following parameter, β, where λ is an arbitrary height, H is a Hessian matrix, J is a Jacobian matrix, and $E_n(x)$ is the error created by an uncalibrated camera.

$$\beta = (H + (\operatorname{diag}(H)\lambda))^{-1}(E_n(x)J^T)$$

Finally, similar features can also be used to calibrate a camera. The matching feature can be seen in two images, in which case two lines are projected from the camera model onto the real world. Otherwise, with only a single image, a match can be made between a real world object and the image, and a line is drawn from the center of the Earth to the selected location, and another is drawn from the camera's position through the image plane. The Levenberg-Marquardt algorithm can be used, but with $E(x)$ defined instead as:

$$\begin{bmatrix} t_o \\ s_o \end{bmatrix} = \begin{bmatrix} (x_1^2 + y_1^2 + z_1^2) & -(x_1x_2 + y_1y_2 + z_1z_2) \\ (x_1x_2 + y_1y_2 + z_1z_2) & -(x_2^2 + y_2^2 + z_2^2) \end{bmatrix}^{-1} \begin{bmatrix} -((a_1 - a_2)x_1 + (b_1 - b_2)y_1 + (c_1 - c_2)z_1) \\ -((a_1 - a_2)x_2 + (b_1 - b_2)y_2 + (c_1 - c_2)z_2) \end{bmatrix}$$

$$E(s,t) = \sqrt{(s_ox_2 + a_2 - t_ox_1 + a_1)^2 + (s_oy_2 + b_2 - t_oy_1 + b_1)^2 + (s_oz_2 + c_2 - t_oz_1 + c_1)^2}$$

3.2 Rendering

To render the calibrated cameras, an algorithm must first find a handful of pixel matches between the two images. SAM, a model that utilizes masks as prompts to identify objects, first isolates the image of a smoke plume (Kirillov). Afterwards, LOFTR, an algorithm utilizing self and cross-attention layers for feature extraction, uses the isolated images of a smoke plume and finds as many pixel matches as possible (Sun). The results are shown below in Figure 3. The

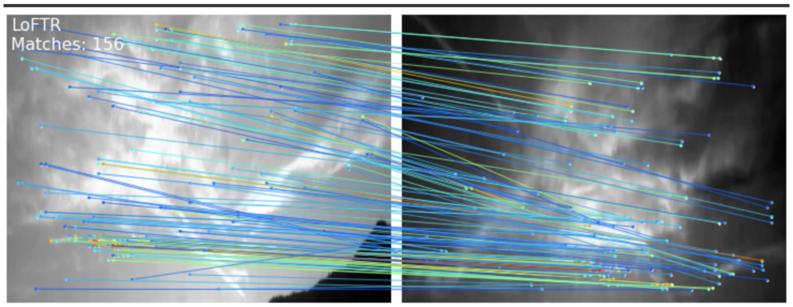
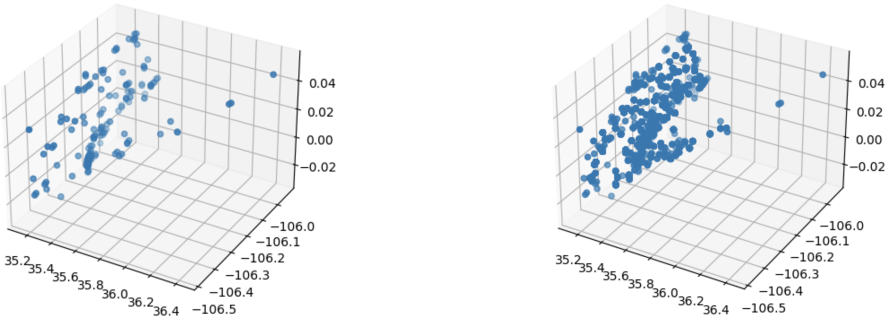


Figure 3: Pixel Matches with LOFTR

pixel matches are then projected out, and the intersection of said images define the location of the point in 3D space. At this point, LOFTR isn't sufficient for the creation of a point cloud, as shown below. Thus, Delaunay triangulation is used to populate the point cloud with other extrapolated position of clouds.



(a) Sparse Depth of a Cloud

(b) Depth Completion

Figure 4: Multiple-Image Calibration

4 Expected Results

As of submitting the interim report, the smoke plume creation prototype is largely complete. With two images of a cloud, an algorithm can generate a point cloud of a cloud within seconds. In addition, with a few manual annotations, anyone can calibrate their cameras to allow for such utility. However, the prototype has not been implemented into an application that is available for public use. This can prove to be a challenge, as the prototype was created with multiple different programming languages and has information stored in a plethora of databases. Despite these challenges, I am confident I will be able to produce such an application before the end of the SuperComputing Challenge.

5 Bibliography

"Camera Calibration and 3D Reconstruction¶." Camera Calibration and 3D Reconstruction - OpenCV 2.4.13.7 Documentation, docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html. Accessed 9 Jan. 2024.

"Camera Modeling: Exploring Distortion and Distortion Models, Part I." RSS, www.tangramvision.com/blog/camera-modeling-exploring-distortion-and-distortion-models-part-i. Accessed 9 Jan. 2024.

"Internet, Smartphone and Social Media Use." Pew Research Center's Global Attitudes Project, Pew Research Center, 6 Dec. 2022, www.pewresearch.org/global/2022/12/06/internet-smartphone-and-social-media-use-in-advanced-economies-2022/.

Kirillov, Alexander et al. "Segment Anything." ArXiv abs/2304.02643 (2023): n. Pag.

Lynn, Kathy, and Wendy Gerlitz. Mapping the Relationship Between Wildfire and Poverty, U.S. Department of Agriculture, 2006, scholarsbank.uoregon.edu/xmlui/bitstream/handle/1794/2417/firepovertydraft_11-21-05.pdf;sequence=1

McKENZIE, DONALD, et al. "Climatic change, Wildfire, and conservation." Conservation Biology, vol. 18, no. 4, 2004, pp. 890–902, https://doi.org/10.1111/j.1523-1739.2004.00492.x.

Sun, Jiaming, et al. "LOFTR: Detector-free local feature matching with Transformers." 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, https://doi.org/10.1109/cvpr46437.2021.00881.

Thomas, Alyssa S., et al. "A burning issue: Reviewing the socio-demographic and Environmental Justice Aspects of the wildfire literature." PLOS ONE, vol. 17, no. 7, 2022, https://doi.org/10.1371/journal.pone.0271019.

Westerling, A. L., and B. P. Bryant. "Climate change and wildfire in California." Climatic Change, vol. 87, no. S1, 2007, pp. 231–249, https://doi.org/10.1007/s10584-007-9363-z.