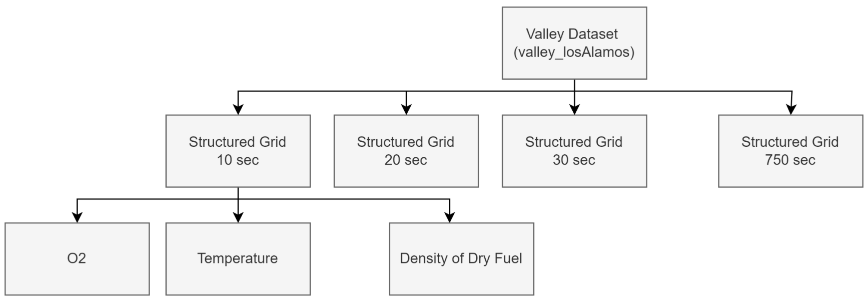
Visualize Wildfire Spread Progress in Valley

Po-Jen Hsu (hsu248@purdue.edu) · Shuang Wu (wu1716@purdue.edu)

Dataset

A dataset provided by IEEE 2022 SciVis Contest is selected for the project. In the provided datasets, three different datasets are obtained by running simulations of wildfire and atmosphere models by scientists. The researchers simulated various parameters, including radius of curvature, wind direction, and different topologies, mountains, and valleys. Two mountain datasets are simulated on idealized topographies, while the valley dataset models the shape of a northern New Mexico canyon. The valley simulation is based on a real wildfire incident. On June 26, 2011, a fire started in Santa Fe National Forest. The wildfire quickly spread and burned over 150,000 acres of forestland in northern New Mexico due to strong wind and low humidity, and was recorded as the largest wildfire in New Mexico state at the time, named “Las Conchas Fire”. In this project, we aim at the valley dataset (valley_losAlamos) because it is the only simulation result from a real disaster, Las Conchas Fire. To download the dataset automatically, download and run the download.py script (https://github.com/seanwu1105/valley-wildfire-viz/blob/master/download.py) to save the data in the project directory for further preprocessing and visualization.

The fire simulation ran for about 750 seconds and was sampled every ten seconds, so the valley dataset provided has about 75 VTK Structured Grid files. Each VTK Structured Grid file includes nine scalar fields. The hierarchy of the valley dataset is shown in the following figure. However, the first fifty seconds are used to ignite the fire and balance the wind. In this project, the visualization will focus on the fire after the ignition so the VTK Structured Grid files before the fifth output will be ignored.



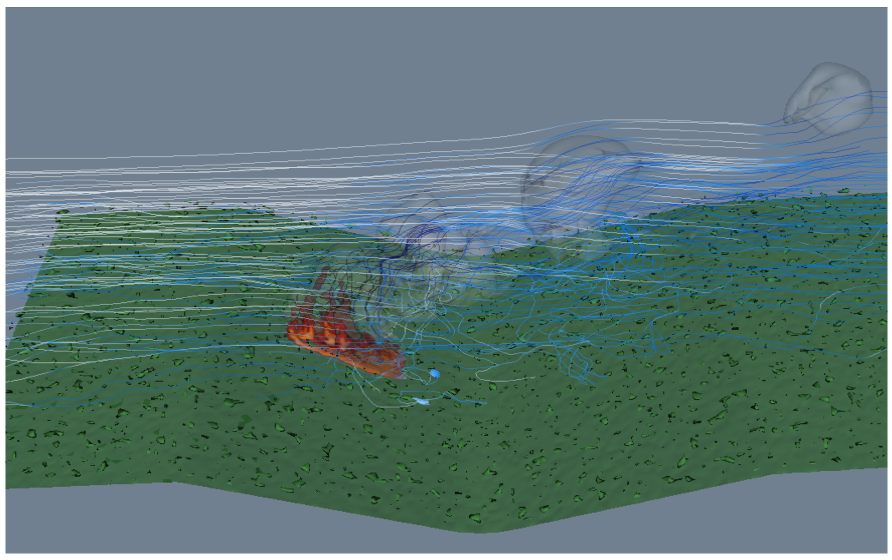
Besides, nine scalar fields are included in each VTK Structured Grid file, as shown in the following table, which allows researchers to do a holistic analysis of various factors, such as wind, humidity, and vegetation. In this project, only a subset of scalar fields are utilized to visualize interesting objects wanted to observe in the project, namely potential temperature, oxygen concentration, three component vectors of wind, and the density of dry fuel, highlighted in gray. While the potential temperature is used to visualize fire, the bulk density of dry fuel is adopted to visualize vegetation. Lastly, the oxygen concentration and the three component vectors of wind are used to visualize wind flow.

Scalar Fields	Meanings
`O2`	oxygen concentration
`convht_1`	convective heat transfer (W/m^3)
`frhosiesrad_1`	fire-induced radiative heat transfer to the fuels (W/m^3)
`rhof_1`	bulk density of dry fuel (kg/m^3)
`rhowwatervapor`	bulk density of the moisture released to the atmosphere as a result of fire (kg/m^3)
`theta`	potential temperature (K)
`u, v, w`	Three component vectors of wind

Problem Statement

To better understand the spread progress of the Las Conchas Fire wildfire, this project plans to achieve two main goals after adopting several visualization techniques and analysis. 1) Discover the location of the wildfire at each timestamp (static analysis), 2) Observe the movement of Las Conchas Fire, namely its spread progress between timestamps (dynamic analysis). The following two paragraphs depict the detailed objective of each objective, respectively.

Firstly, a static analysis is conducted for discovering the location of the wildfire at each timestamp. The first step of analyzing an object is to pinpoint its location in static visualizations. Once the location of the object is known, the following visualization and observation will focus on that specific place and time. Regarding the study of a wildfire, two main components of fire need to be found. Flame is the first object which this project plans to discover and visualize, while the smoke is the second one. The following figure shows the expected visualization result at a specific timeframe. The visualizations of smoke and flame are utilized to find the location of the wildfire. Moreover, the visualization of vegetation validates the location of the fire and gains a more realistic visualization. Lastly, the wind visualizations help analyze the causes of fire movement.



Second, a dynamic analysis is adopted to observe the wildfire moving according to various factors among timestamps. The ultimate target of the project is to obtain some sort of animation that clearly shows the movement of the Las Conchas Fire. After the site of the wildfire is located after achieving the first goal, the movement of the fire is visualized and examined. To achieve this goal, the same visualization techniques adopted in the first target are applied to all seventy-five timestamps. Finally, the project will combine all seventy-five static visualization results and create an animation to show the movement of wildfire. Besides, the variation of vegetation density, oxygen concentration, and wind strength are observed to help us better analyze the cause of wildfire spread progress. As a result, all visualized objects can help us have a better understanding of the spread of the Las Conchas Fire happening in New Mexico Canyon.

Methodology

Preprocessing

Originally, for each timestamp, the dataset has more than 1.1 GB, and more than 81 GB in total. Thus, rendering a dataset in one single timestamp would require more than 1 minute in our machine. To alleviate the resource demands for visualization, we found that there are several scalar arrays that were not needed. We only picked the following arrays and removed others to greatly reduce memory usage.

- Wind vector components: u, v, w
- Potential temperature: theta
- Oxygen concentration: O2
- Vegetation density: rhof_1

After removing the unnecessary scalar arrays, we further minimize the input data size by extracting only the region we were interested in. We used ParaView to experiment and locate the flame and wind flow around it. Also, we downsampled the extracted dataset a little bit without losing too much precision which affected the visualization results.

We combined three wind vector component arrays, u, v, and w, into a vector array, named wind. Thus, we do not need to do such calculations during the runtime, which could bring a better performance.

Last, in order to visualize the flame and smoke with volume rendering, we converted and exported the preprocessed theta array into an image data. Thus, there is no runtime data type conversion, which greatly reduces resource consumption.

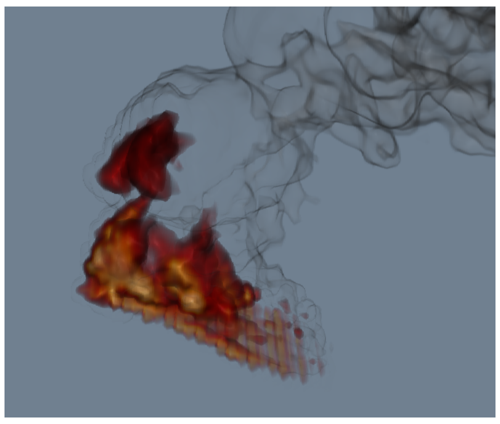
After the preprocessing, the data size for each timestamp shrinks to approximately 21 MB. You can get the preprocessed dataset by executing the extract.py script after downloading the original dataset:
https://github.com/seanwu1105/valley-wildfire-viz/blob/master/extract.py

Fire

The potential temperature is the scalar field employed to visualize fire in the project. The fire consists of two sub-components, fire, and smoke. By referring to the suggestions from the dataset document and visualizing on Paraview, the temperature of the flame is found between 400~800K, while smoke is found at 310k. In the project two methods are implemented to visualize fire, Volume Rendering and Isosurfacing. The detail of the two implementations is described in the following paragraphs.

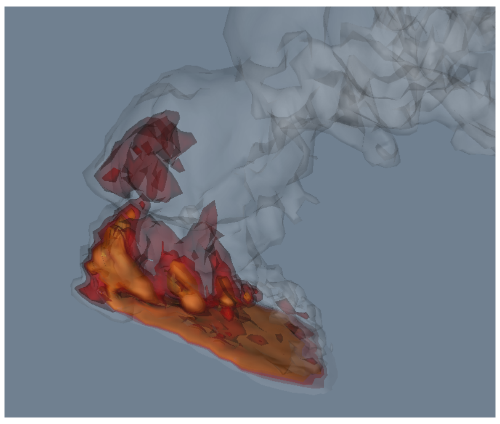
Volume Rendering

A color transfer function for volume rendering is set to visualize flame to red, dark red, or orange, to white as the potential temperature increases from 400K to 800K. Moreover, the temperature at 310K is set to gray for the smoke visualization. On the other hand, an opacity transfer function is designed to reveal the internal structures of flame. Since the potential temperature at the inner part of the flame is higher than the outer part of the flame. The designed alpha value linearly increases from 0.1 to 0.5 corresponding to the temperature from 400K to 800K which allows users to observe the hottest part of the flame and looks more like a real flame. The alpha value of smoke is set to 0.1 at 310K potential temperature. The result of volume rendering is shown in the following image.



Isosurfacing

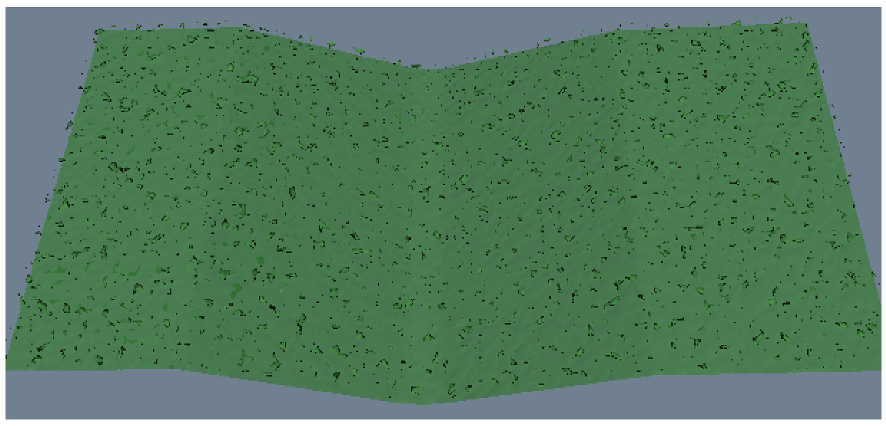
Because the performance of volume rendering is slow in our machine, we also implemented isosurfacing as an alternative visualization of fire. Similar to the temperature range and colors adopted in volume rendering, five isovalues are adopted to show flame, namely potential temperature at 400K, 500K, 600K, 700K, and 800K. The isovalue of smoke used is the same as volume rendering which is the potential temperature at 310K. While the selected potential temperature is alike to volume rendering, the assigned alpha value is adjusted to achieve a better result. The alpha value employed by isosurfacing linearly rises from 0.5 to 0.7 corresponding to the temperature from 400K to 800K. However, the alpha value of smoke is set to 0.3. The result of isosurfacing is shown in the following image.



Comparison

When comparing two implemented visualization methods for fire, the volume rendering is more suitable for fire. The visualization result of volume rendering shows more details in each layer because the utilization of the opacity transfer function shows a wide range of isovalues. However, the isosurfacing only showed the chosen five isosurfaces and the outer isosurfaces (low temperature) occlude the inner isosurface (high temperature). If the alpha values are lowered, the color of the outer isosurface is too pale to observe. Therefore, lowering the alpha values cannot solve the occlusion. On the other hand, the performance of isosurfacing is much better than volume rendering and is more suitable for observing the movement between timestamps, so the isosurfacing visualization is preserved in the project.

Vegetation

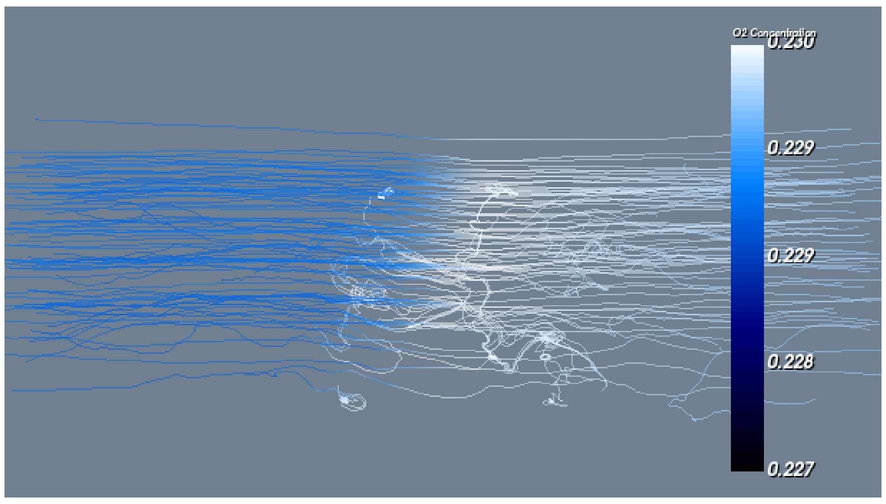


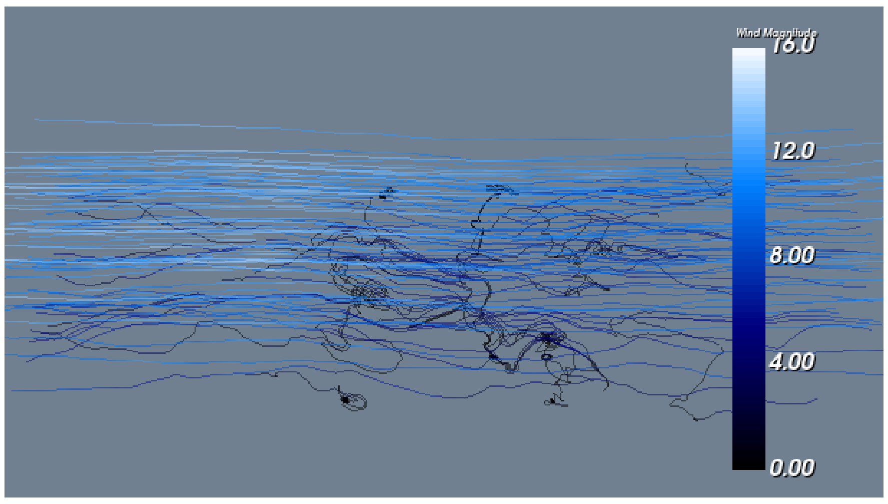
At first, we attempted to apply volume rendering to show the continuity of the fuel density in opacity. However, since the dataset is in a structured grid, converting the dataset into an image data for the volume rendering generated an array of noticeable discontinuous gaps. We finally decided to use three isosurfaces mapping to different densities, from 0.0 to 1.0, to avoid such artifacts.

Wind

We applied 100 streamlines to trace the wind flow. By placing the source at the ignition point and propagating the stream tracer in forward and backward directions, we were able to visualize the airflow in the whole region we are interested in. We provided two modes for color mapping, including oxygen concentration and vector magnitude, to discover more interesting insights.

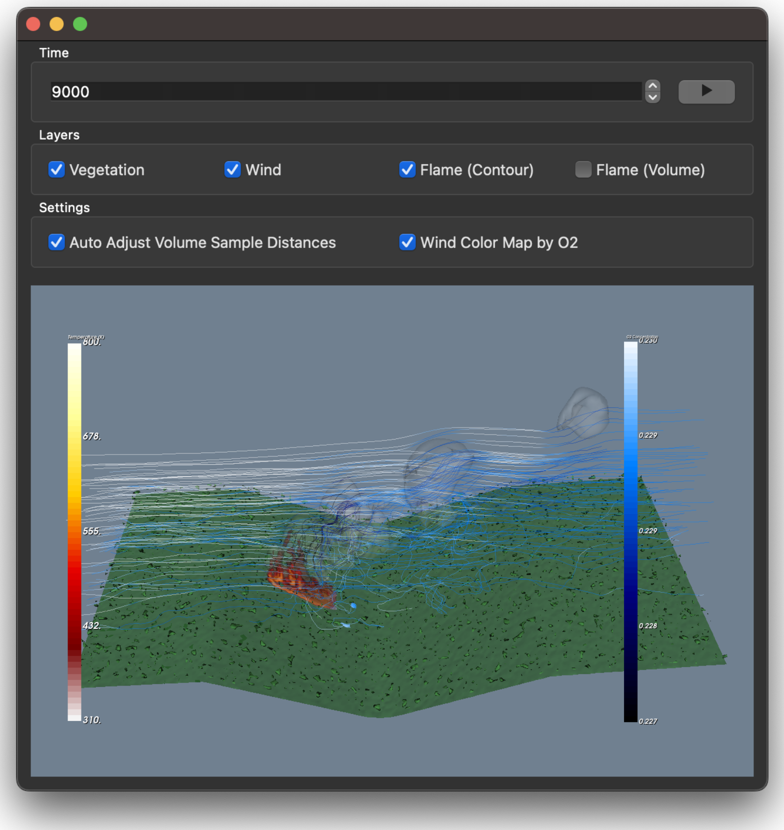
For the oxygen concentration color mapping, we found the most significant change ranging from 0.227 to 0.23. Thus, we set the color scale to focus on this range to visualize the effects caused by the flame. For the magnitude color mapping, we found the velocity of the wind mainly falls between 0.0 and 16 statistically, and thus we set it as the range of the color scalar bar to indicate the place having the strongest or weakest wind.





Results

Experiment Tool



We implemented an interactive GUI application to experiment and analyze the data. Since the dataset has been preprocessed to shrink the size, the loading and rendering time in the application is minimized, which achieves near real-time responsiveness.

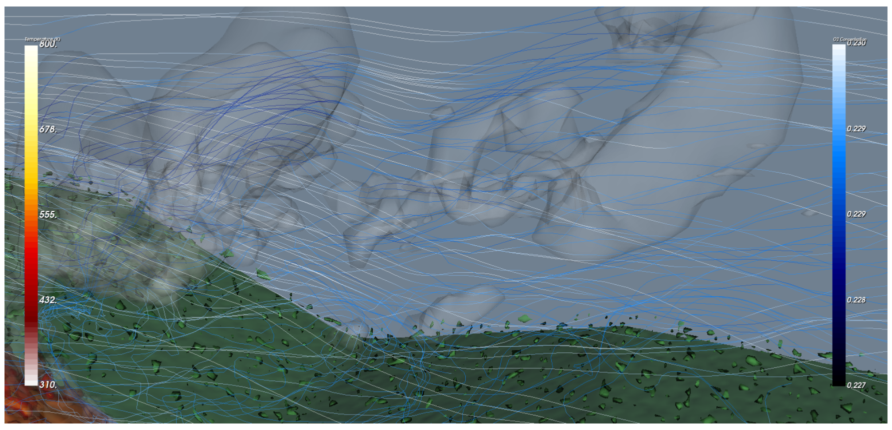
Users can set the desired timestamp with the top spinbox. Also, on clicking the play media button, the app will automatically start an animation playback to render all datasets sequentially till the end of the whole data series. Simultaneously, the application will export the rendered images at /images folder for each frame, which helps us to generate motion pictures for further processing and analysis.

We allow the user to toggle each visualization component, including vegetation geometry, wind streamlines, flame with isosurfaces, and flame with volume rendering, to display on the VTK visualization frame.

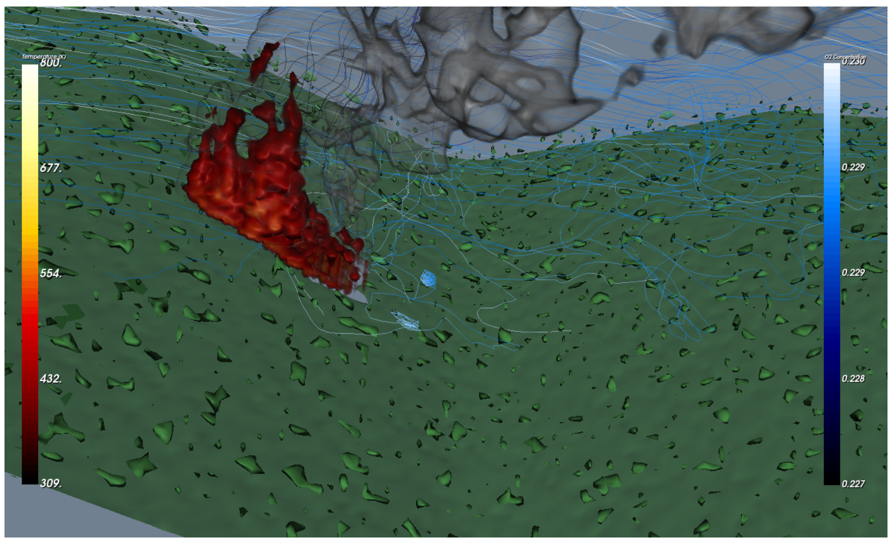
Two detailed settings are built for us to tweak the visualization results and gain more insights with them. “Auto Adjust Volume Sample Distances” is by default enabled, which makes the volume rendering much more performant but loses rendering precision. We usually turn it on when interacting with the VTK visualization frame and disable it when exporting the frame into high-quality images. Being able to adjust the settings in runtime makes the app more flexible and less time-consuming to reach the analytic goal. When “Wind Color by O2” is disabled, the wind streamlines map the color by the magnitude, which allows us to know how strong the wind is in a certain location. On the other hand, when it is enabled, the streamlines map the color by the oxygen concentration. Thus, we can visualize multiple properties of the wind in one visualization frame.

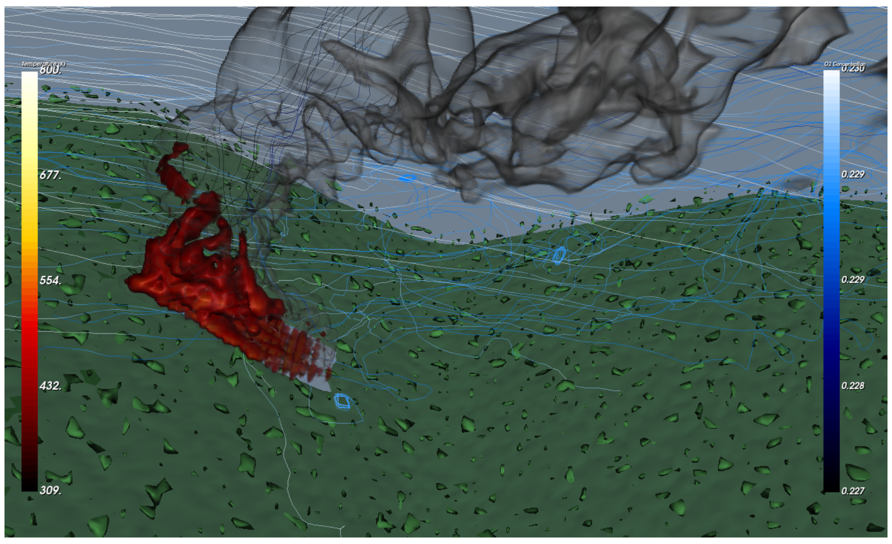
Discoveries

By comparing the wind streamlines mapped by oxygen concentration and the flame isosurfaces, we notice that the location of smoke plumes has a relatively lower oxygen concentration. We believe the main factor is combustion. This relation is consistent across all timestamps.

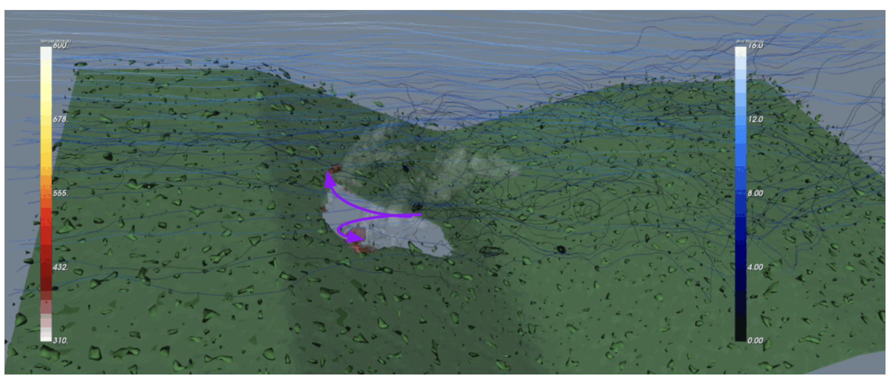


Furthermore, we found that the stagnant air in the valley usually has a lower oxygen concentration. Thus, the chemical reaction could be less intensive in the area. It could limit the direction of the wildfire spread.





Finally, the vegetation density decreases significantly after the flame passes through. Thus, we think the region where vegetation density is low could indicate the route of the flame progress. Also, it could prevent the flame from moving backward as the trail has lower fuel. The flame moves toward the hill since the incoming wind has a higher oxygen concentration and the vegetation density is high on the hill. However, according to the wind streamlines colored by magnitude, the wind velocity on the hill is also higher than the one in the valley. Thus, the flame could be forced to diverge and turn its progress direction before reaching the top.



References

- Project code repo: https://github.com/seanwu1105/valley-wildfire-viz
- IEEE 2022 SciVis Contest: Vorticity-driven Lateral Spread Ensemble Data Set. https://www.lanl.gov/projects/sciviscontest2022/