

A Homologous Sequence of Coronal Mass Ejections

Simulation data from Yuhong Fan of the National Center for Atmospheric Research

Advanced Visualization Lab (AVL), NCSA, University of Illinois

Donna Cox, Bob Patterson, Stuart Levy, AJ Christensen, Kalina Borkiewicz, Jeff Carpenter

The AVL worked with astrophysics data scientists to create a narrative journey through the dynamic processes of the sun in a digital fulldome production titled “Solar Superstorms”, narrated by Benedict Cumberbatch, as part of the CADENS NSF ACI-1445176 grant. This visualization represents a part of the story of the dynamic processes of our sun, and how they affect life on Earth.

Yuhong Fan’s magnetohydrodynamic simulation of the development of a homologous sequence of two coronal mass ejections (CMEs) demonstrates their so-called cannibalistic behavior. These CMEs originate from the repeated formations and partial eruptions of kink unstable flux ropes as a result of continued emergence of a twisted flux rope across the lower boundary into a pre-existing coronal potential arcade field. The simulation shows that a CME erupting into the open magnetic field created by a preceding CME has a higher speed. The second CME is cannibalistic, catching up and merging with the first into a single fast CME before exiting the domain. Yuhong Fan simulated her non-uniform spherical grid data on the Yellowstone supercomputer at the National Center for Atmospheric Research.

This simulation shows the explosive power of ionized plasma that can be projected from the sun toward Earth. AVL shows Fan’s magnetic field lines in blue/cyan, traced between points on the Sun’s surface, and a current field ($|\text{curl } \mathbf{B}|$), computed from the simulation’s magnetic field, in yellow/orange. As the CMEs expand, note that the current is concentrated into a thin, rippling sheet - reminiscent of auroras on Earth, where current sheets are also involved.

The AVL team used a commercial visual effects tool called Houdini as its primary data processing environment to create derivative data assets which could demonstrate the particularly interesting features of the simulation. To allow the commercial tool to read and process scientific data, AVL programmer Kalina Borkiewicz wrote a C++ plugin using the Houdini Development Kit, that was able to read numerous file types and grid topologies, and produce Houdini-native assets that could be updated in the interactive graphic user interface. Attributes that could be updated include different varieties of volume domain edge falloff, asset volume resolution, and asset volume type. Using this, the AVL could adjust the asset volume to low resolution while experimenting with techniques. The volume type allowed the team to read the data into a rectangular grid, or into a sparse volume grid format called OpenVDB for render and processing optimization. Using the Houdini plugin, the AVL was able to convert the source non-uniform spherical grid into a rectangular grid containing scalar current values.

To create interactive previews of scientific datasets, the AVL uses an original open-source software tool called Partiview, which was primarily authored by AVL programmer Stuart Levy. Partiview is able to render particles, lines, triangle meshes, and star-like data and interactively update their visual attributes based on data attributes. To flexibly navigate through these Partiview environments, explore 3D data topology, and develop complex camera choreography in an immersive stereoscopic 4K environment, the team links Partiview with another original software tool called Virtual Director, authored by past AVL programmers Matt

Hall and Marcus Thieboux. Virtual Director is a remotely collaborative camera control and design tool capable of performing or keyframing path splines and editing them during interactive data playback.

To create an interactive time-evolving representation of this high resolution volume for camera choreography, the team seeded a uniform grid of points in the volume and colored them by the values of the nearest voxels. Additionally, Yuhong Fan's field lines were rendered as geometric objects.

Because the data cadence was not as high as the frame rate of the video, the team developed data interpolation techniques to create smooth visual playback. The Houdini plugin was augmented to do temporal b-spline interpolation. This method created noticeable stepping artifacts at low resolution in a uniform grid, but by pushing the resolution much higher by implementing a sparse VDB volume representation and convolving the voxels, the artifacts were removed. The magnetic field lines were challenging to interpolate because their shapes changed dramatically between datasteps. To interpolate these, the team created a method of "rubber band interpolation" that would non-uniformly stretch the field lines so they could associate both similar shapes and parametric coordinates. After this interpolation, some lines were still so problematic that they were culled using a volatility score that was calculated in an external script. Deriving velocity from the interpolation technique still produced distracting results when motion blurred, so the team decided to calculate line "echoes" to show motion of a line over time. The lines and their echoes were then splatted into sparse VDB volumes to create an integrated volumetric appearance, colored by the magnetic magnitude on the original data lines, and faded based on an echo's age.

A sun surface asset was created using Matthias Rempel's solar surface convection supercomputer simulation. Matthias's data was interpolated in image compositing software, tiled to cover a larger percentage of the sun's surface, and displaced based on intensity. The data was positioned so that the coronal mass ejection simulation appeared directly above a sunspot. Matthias's data also included magnetic vectors, which were brought into Houdini as RGB values and then translated into small volumetric pointers on the surface of the sun. A halo was also put around the sun, mapped with imagery observed through NASA's Solar Dynamics Observatory telescope.

These many different datasets were rendered in several image layers from a unified camera path, and composited together using a commercial software tool called Nuke.