

RESEARCH ARTICLE | MARCH 11 2021

## Toward 3D data visualization using virtual reality tools FREE

Special Collection: [Proceedings of the 23rd Topical Conference on High-Temperature Plasma Diagnostics](#)

J. L. Kline  ; P. L. Volegov 

 Check for updates

*Rev. Sci. Instrum.* 92, 033528 (2021)

<https://doi.org/10.1063/5.0040468>

 CHORUS



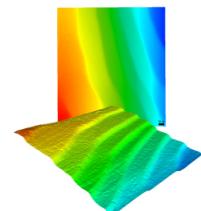
View  
Online



Export  
Citation

CrossMark

02 December 2023 23:27:49

 MAD CITY LABS INC. <a href="http://www.madcitylabs.com">www.madcitylabs.com</a>	Nanopositioning Systems 	Modular Motion Control 	AFM and NSOM Instruments 	Single Molecule Microscopes 
---	--	---	---	--

# Toward 3D data visualization using virtual reality tools

Cite as: Rev. Sci. Instrum. 92, 033528 (2021); doi: 10.1063/5.0040468

Submitted: 13 December 2020 • Accepted: 24 February 2021 •

Published Online: 11 March 2021



View Online



Export Citation



CrossMark

J. L. Kline<sup>a</sup> and P. L. Volegov

## AFFILIATIONS

Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

**Note:** Paper Published as part of the Special Topic on Proceedings of the 23rd Topical Conference on High-Temperature Plasma Diagnostics.

<sup>a</sup>Author to whom correspondence should be addressed: [jkline@lanl.gov](mailto:jkline@lanl.gov)

## ABSTRACT

Virtual Reality (VR) offers the opportunity to display data, instrumentation, and experimental setups in three dimensions and gives the user the ability to interact with the objects. This technology moves visualization beyond two-dimensional projections on a flat screen with a fixed field of view in which a keyboard or another similar controller is needed to change the view. Advances in both hardware and software for VR make it possible for the non-expert to develop visualization tools for scientific applications both for viewing and for sharing data or diagnostic hardware between users in three dimensions. This manuscript describes application development using two VR software tools, Unity gaming engine and A-frame, for visualizing data and high energy physics targets.

<https://doi.org/10.1063/5.0040468>

## I. INTRODUCTION

The dynamic topology of plasma physics experimental parameters is three-dimensional (3D). While experiments are often designed with a symmetry axis to effectively reduce the dimensionality of the system and thus the physics description, perturbations typically distort the symmetry. The simulation or experimental data and/or plots of the data or images are then typically projected onto a two-dimensional surface, losing some of the 3D information. Even 3D tomographic reconstructions from multiple projections are eventually rendered in 2D on a computer screen or in print. While such data or images can be plotted in a perspective geometry to provide the appearance of depth, the observer still has to mentally transform the data or image. The ability to directly visualize data in 3D enables an observer to examine objects as they appear in reality. Even in the 1980s, the power of 3D visualization of scientific data was realized by Gekelman and Stenzel<sup>1</sup> who published printed data that could be viewed in 3D with the proper glasses. Today's Virtual Reality (VR) capabilities make it possible to replace flat computer screens with tracked Head Mounted Displays (HMDs) and hand controls to enable both 3D views of data and instruments, as well as manipulate and share the view with others across the globe.

The current state of VR technology makes the tools for creating 3D visualizations accessible to non-experts, i.e., not requiring an

in-depth understanding of computer rendering or programming. The telecommunication and gaming industries have driven hardware advances that enable rendering 3D objects via platforms as simple as a smart phone with Google cardboard to sophisticated HMDs. The hardware also enables networking of displays for collaborative views and interactive data experiences.<sup>2</sup> Software tools, such as Unity<sup>3</sup> and A-frame,<sup>4</sup> make scene rendering very simple and available across a number of VR platforms. While the development of scientific visualization applications using VR is in progress, its use is not currently widespread with little to nothing in the literature for plasma or high energy density physics.

Using the Oculus Go VR platform, we developed entry-level software to display 3D objects for plasma science. The software enables the user to orbit an object while keeping the HMD pointed to the center of the coordinate system. At any position, the user can then freely rotate their head to naturally look in any direction to examine the data or the objects. This enables virtual interactions for viewing objects such as data, high energy density physics targets, and instruments in their natural 3D state.

In this manuscript, we show examples using our applications for scientific use of VR, both data and laser driven physics targets. We present a short discussion on how 3D objects are rendered for VR followed by details of our VR development using A-frame and Unity. We show examples of 3D views of data and targets using

Oculus Go. Finally, we briefly discuss potential future directions in Sec. V.

## II. RENDERING 3D OBJECTS

Most VR rendering uses ray-tracing designed for graphics processing units that utilize fast matrix operations. This enables the rapid display of large high-resolution graphics (Fig. 1). While ray-tracing visualization can be done via an analytical description of objects and their intersection with light rays, typically, objects are approximated using many small planar polygon regions that simplifies ray-tracing calculations for complex objects. These objects are composed of polygons described by nodes, vertices, and faces, along with the faces' normal vectors. For rendering objects, the other two key elements are a light source and a virtual camera. The rays emanate from the light source. It is the interaction between these rays and the object's facelets that produces reflected rays. Typically, the polygon's facelets are triangles, making it easy to determine the path of the ray with matrix math for which graphic processing units are designed. The next important aspect to rendering is the transformation of the properties of the incoming light ray by the facelet properties contained in the texture information. This includes physics such as specular vs diffuse reflections, colors, and transparency. A virtual camera represented by a screen placed in the space, then, captures the reflected rays from the facelets. It is also possible for the facelet texture to be emissive such that an external light source is not needed. Once the scene is set up, it is simple to create two versions with cameras at slightly different angles to create a 3D rendering, i.e., using one for each eye.

Software for creating objects for 3D rendering is ubiquitous today from most computer aided design software to specialized software such as 3D builder, Blender,<sup>5–7</sup> and VisRad.<sup>8</sup> It is also simple to create 3D objects using standard mathematic software such as python and MatLab. Using mathematical software tools is critical for viewing scientific data, making them useful for generating 3D objects. Scientific computing software often contains built-in tools for generating objects and saving them in standard file formats for 3D visualization software. Two of the most common formats are stereolithography files (.stl)<sup>9</sup> and wavefront files (.obj). A valuable tool for working with 3D objects and preparing them for use with application software is Blender,<sup>5</sup> which is also used for astrophysics rendering.<sup>6,7</sup>

The remainder of this manuscript will concentrate on rendering two examples: hotspot data from layered inertial confinement fusion implosions and a hohlraum target for double shell pre-heat measurements. To determine the health of an inertial confinement

fusion (ICF) implosion requires measurements of the nuclear burn including the neutron production rate and the production location. For experiments at the National Ignition Facility,<sup>10</sup> the neutron imaging system<sup>11–13</sup> is the only diagnostic with three nearly orthogonal lines of sight and capable of reconstructing 3D data without the assumption of a symmetry axis. Using the image data from these instruments, a 3D tomographic reconstruction of the neutron production rate shape is produced.<sup>14–16</sup> Figure 2 shows an example starting with the 2D projections [Fig. 2(a)] and the reconstructed image [Fig. 2(b)]. The reconstructed image is still a project along a given line of site, forcing the reader to interpret depth based on the perspective view. In VR, the user can observe the depth of the objects stereoscopically, i.e., normal eyesight. This becomes more relevant for viewing the isocontours containing various levels of the neutron production rate, as shown in Fig. 2(c). In this particular experiment, the neutron production rate near the maximum breaks into two lobes and is visibly 3D.

The second example is a “keyhole” target (Fig. 3) for double shell experiments<sup>17–19</sup> in which measurements of pre-heat are made with the VISAR<sup>20</sup> diagnostic. The target shown is used to measure x-ray pre-heat from the hohlraum that reaches the inner shell of the double shell target. The inner shell is sealed around the gold cone and placed into an outer shell. This assembly is placed into a hohlraum target with a large shield extending from the target on the exterior of the hohlraum. This shield is to block stray light from the VISAR field of view. The inner shell and the gold cone are sealed and near the time of the experiment are filled with liquid D<sub>2</sub>. After the lasers fire, a VISAR laser penetrates a window on the gold cone through D<sub>2</sub> and reflects from the inner surface of the shell. As x-ray pre-heat drives shocks into D<sub>2</sub>, the VISAR tracks these shocks to determine the speed, which is related to the amount of pre-heat reaching the inner shell. The copper rods extending from the target are the cryogenic cold fingers used to control the target temperature and thus maintain D<sub>2</sub> in a liquid state. The key-hole target design was developed for hotspot inertial confinement fusion.<sup>21,22</sup>

## III. A-FRAME, WEB-BASED VR

A-frame is a web framework for creating VR environments using hypertext markup language (HTML)-like tags, making it easy to get started. A Java engine interprets the tags converting the information for rendering in 3D. The framework enables the user to display 3D objects with a simple markup language file. Since A-frame is a web-based platform with a Java engine, the file must be accessed through a server to be displayed. This can be a local server on a laptop or desktop computer, a python application server, or a centralized server. A link to the A-frame Java engine<sup>23</sup> is required in the HTML file using the “script” tag. This can be done either by downloading the file and hosting it locally or adding a link in the header. For the development work here, we utilize the free server from “glitch.com,” which provides an easy means of editing files with a quick link to serving the web page. The setup used for our orbital controls and viewing targets with Oculus Go can be found at <https://glitch.com/edit/#!/oculusgo>.

For this project, we wrote an A-frame component in Java script to enable the user to change the orbital position around the target with the controller. The file named oculus-go-orbit.js contains the

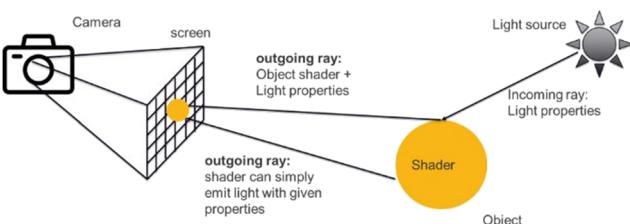
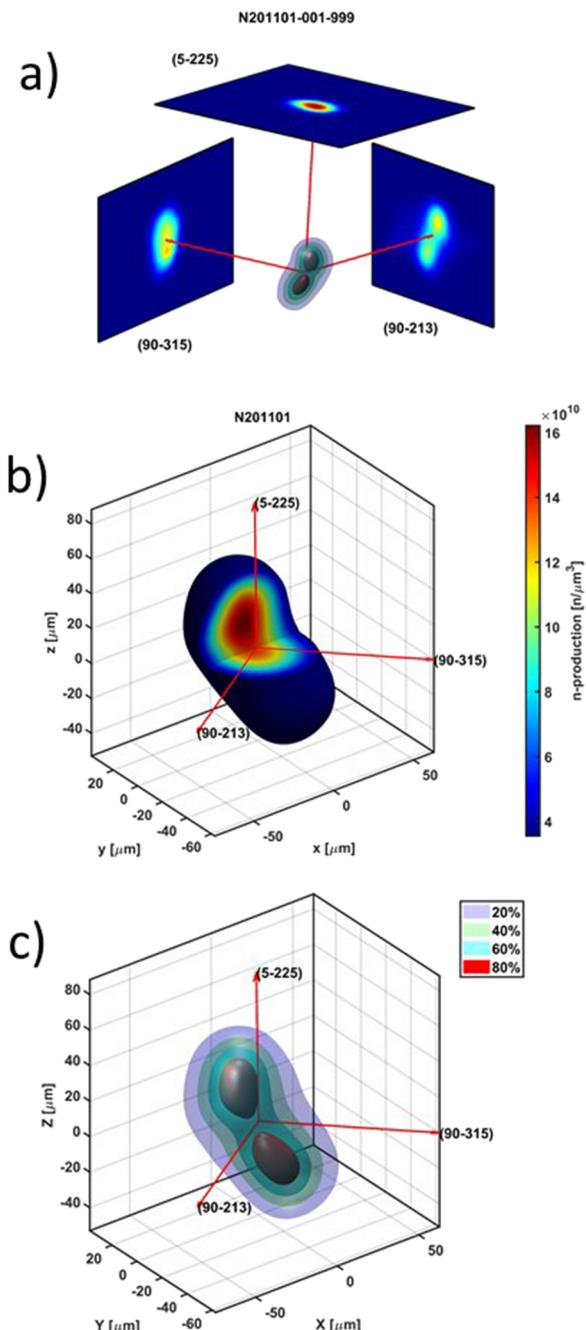
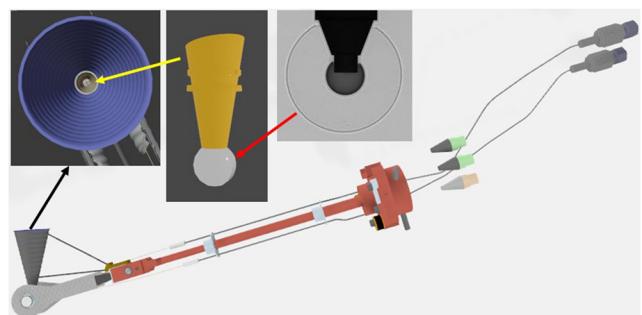


FIG. 1. Diagram showing how to render objects in virtual reality.



**FIG. 2.** (a) Three measured projection of the neutron production rate of an ICF deuterium–tritium layered experiment. (b) Image of the reconstructed neutron production based on the projection. (c) Isosurfaces of the neutron production rate at 80%, 60%, 40%, and 20% of the maximum projection rate.

code. Adding the component to a camera object in the HTML file enables the functionality: <a-camera oculus-go-orbit></a-camera>. While this path to data visualization is quite simple, challenges remain. The largest issue is that not all web browsers support VR



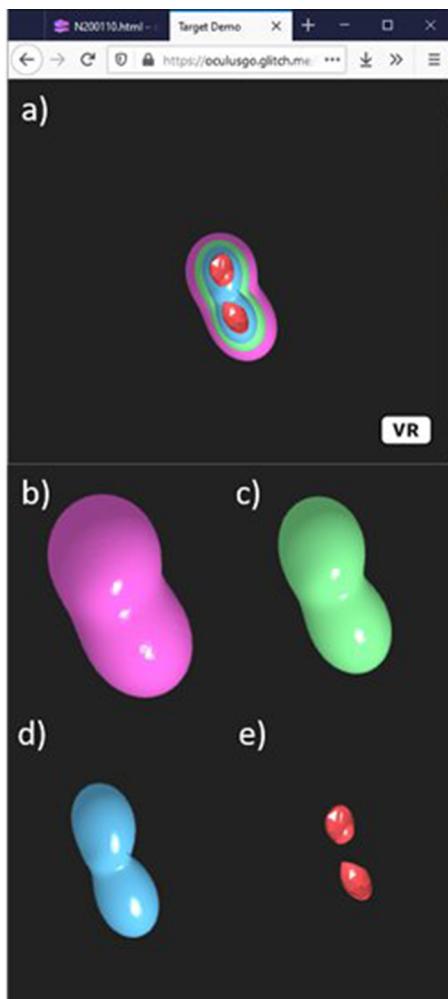
**FIG. 3.** Double shell keyhole target for x-ray pre-heat experiments at the National Ignition Facility with insets showing internal capsule components.

and they all do not have the same capabilities. This can lead to a significant investment of time to get the initial setup working. A-frame is supported by Mozilla, so we recommend using the Firefox web browser. There is a lot of additional capability using APIs from third party sources, which adds complexity. Figure 4 shows an image from a web browser of reconstructed neutron production rate contours. The image in the browser is initially rendered in 2D, but if the user has an Oculus Go HMD, the scene can be rendered in 3D by selecting the VR button in the lower right-hand corner. Using the Oculus go controller, the user can rotate and zoom in and out on the composite image containing all contours by pushing the trigger button cycles through the single contour reconstructions with 3D views. In this case, the peak production breaks into two lobes near peak production. The narrowing of the waist between the two lobes becomes more relaxed with a decreasing contour level. However, the overall peanut shape is very prominent when rendered in 3D.

#### IV. UNITY

Unity provides a 3D engine and an integrated design framework for developing VR applications. While principally used for game development, it is also being used to develop apps for training simulators, first-responder applications, and other business-focused applications. The Unity engine offers tools to enable importing graphic objects and to manipulate the objects. Programming object control in Unity is done using C# and/or Java script. Unity has a built-in functionality to simplify programming interactions with objects, and the integrated design environment is well developed, providing tools for testing and debugging applications. There is also an extensive set of resources and tutorials for learning to use Unity with a large user community for help.

We developed a Unity application for viewing targets, instruments, or data starting with a menu system using a graphics ray caster as a pointer. Each category contains several example objects that can be selected by pushing the controller's trigger button. The touch pad input provides the means to orbit the object in r, theta, and phi to change the principal viewing position. Figure 5 shows three views from our double shell keyhole example. The first is outside the hohlraum, looking toward the cryogenic target holder with the user as the scale of the target. The second view is down the VISAR cone into the capsule. From this view, one can see the VISAR mirror and hole in the side of the cone for the polar VISAR

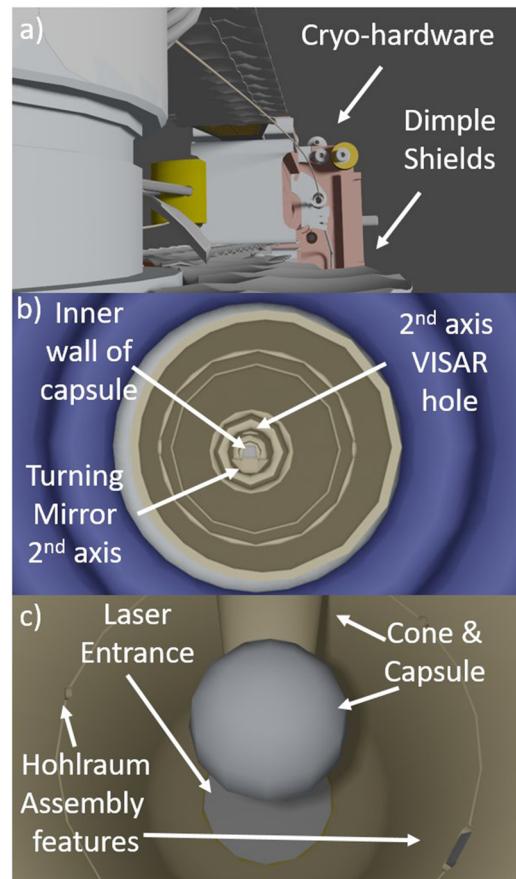


**FIG. 4.** (a) Composite 3D image shows contours at 80%, 60%, 40%, and 20% of peak neutron production rates, matching Fig. 2(c). The individual isocontours of the neutron production rate for (b) 80%, (c) 60%, (d) 40%, and (e) 20%, all of which can be viewed in 3D in VR.

measurement. The third view puts the viewer inside the hohlraum, looking at the capsule and features on the hohlraum walls. Simulations do not typically capture such assembly features nor their effects on the experiment. In this case, the ability to tour the target in 3D provides a scientist the ability to look for differences between the targets as designed and the simulations to investigate sources of 3D perturbations for the experiment. While the 2D figure in this paper is unremarkable, the 3D views bring the target to life, especially with the ability to move the viewing position and to look around simply by rotating your head. Given the amount of details to describe the entire app, it is not possible to elaborate it in this short article. The project itself can be found in github.

## V. FUTURE DEVELOPMENT

The current examples are simple, but we plan to add capabilities. While it is clear that 3D visualization tools enable natural views



**FIG. 5.** (a) View outside the keyhole target toward cryogenic hardware. (b) View down the VISAR light shield into the inner double shell capsule. (c) View inside the hohlraum at the capsule and the cone showing hohlraum assembly features.

of data and objects, much work is needed before we will know what impact such tools can have on scientific analysis. In the near future, we hope to enable viewing of collections of targets including the primary target and backlighters, as well as the laser beams to provide an immersive perspective for an experimental design. Implementing data analysis tools for converting measurements from multiple diagnostics into a common 3D data view enables cross diagnostic analysis. Analysis tools for enhancing the viewing of data such as slicing, real time contour selection, and spatial measurement capabilities enhance 3D data analysis. Another desire is multi-user capabilities for collaborative experiences.

## VI. CONCLUSIONS

Here, we showed a couple of paths forward to enable current technology to create apps to enable VR views of targets, data, and instruments. VR tools and hardware make it a viable means to provide novel views of data and experimental setups. We plan to continue developing tools to provide more quantitative analysis in the VR environment so that data can be evaluated in its natural 3D state. This technology also enables collaborative work in these environments to connect people for shared views.

## ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy through the Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the U.S. Department of Energy (Contract No. 89233218CNA000001). The authors wish to thank LANL and LLNL target engineering for the key-hole target drawings and acknowledge LLNL for executing the DT layered experiment from which the neutron imaging data are shown.

## DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## REFERENCES

- <sup>1</sup>W. Gekelman and R. L. Stenzel, *J. Geophys. Res.* **89**, 2715 <https://doi.org/10.1029/ja089ia05p02715> (1984).
- <sup>2</sup>A. S. Garcia, T. Fernando, D. J. Roberts, C. Bar, M. Cencetti, W. Engelke, and A. Gerndt, *Multimedia Tools Appl.* **78**, 33191 (2019).
- <sup>3</sup>J. K. Haas, *A History of the Unity Game Engine* (Worcester Polytechnic Institute, 2014).
- <sup>4</sup>Aframe development started with Mozilla and is currently maintained by supermedium. Information can be found at <https://Aframe.io/>.
- <sup>5</sup>R. Hess, *Blender Foundations: The Essential Guide to Learning Blender 2.6* (Focal Press, 2010).
- <sup>6</sup>B. R. Kent *Publ. Astron. Soc. Pac.* **125**, 731 (2013).
- <sup>7</sup>B. R. Kent, 3D Visualization of Astronomical Data with Blender, 2015.
- <sup>8</sup>J. J. MacFarlane *J. Quant. Spectrosc. Radiat. Transfer* **81**, 287 (2003).
- <sup>9</sup>Stereolithography Interface Specification, 3D Systems, Inc., 1988.
- <sup>10</sup>J. A. Paisner, E. M. Campbell, and W. J. Hogan, *Fusion Technol.* **26**, 755 (1994).
- <sup>11</sup>F. E. Merrill, D. Bower, R. Buckles, D. D. Clark, C. R. Danly, O. B. Drury, J. M. Dzenitis, V. E. Fatherley, D. N. Fittinghoff, R. Gallegos, G. P. Grim, N. Guler, E. N. Loomis, S. Lutz, R. M. Malone, D. D. Martinson, D. Mares, D. J. Morley, G. L. Morgan, J. A. Oertel, I. L. Tregillis, P. L. Volegov, P. B. Weiss, C. H. Wilde, and D. C. Wilson, *Rev. Sci. Instrum.* **83**, 10D317 (2012).
- <sup>12</sup>M. D. Wilke, S. H. Batha, P. A. Bradley, R. D. Day, D. D. Clark, V. E. Fatherley, J. P. Finch, R. A. Gallegos, F. P. Garcia, G. P. Grim, S. A. Jaramillo, A. J. Montoya, M. J. Moran, G. L. Morgan, J. A. Oertel, T. A. Ortiz, J. R. Payton, P. Pazuchanics, D. W. Schmidt, A. C. Valdez, C. H. Wilde, and D. C. Wilson, *Rev. Sci. Instrum.* **79**, 10E529 (2008).
- <sup>13</sup>G. P. Grim, R. J. Aragonez, D. P. Atkinson, D. P. Atkinson, S. H. Batha, M. A. Barrios, D. E. Bower, D. K. Bradley, R. A. Buckles, D. D. Clark, D. J. Clark, J. R. Cradick, C. Danly, O. B. Drury, V. E. Fatherley, J. P. Finch, F. P. Garcia, R. A. Gallegos, N. Guler, S. M. Glenn, A. H. Hsu, N. Izumi, S. A. Jaramillo, G. A. Kyrala, S. Le Pape, E. N. Loomis, D. Mares, D. D. Martinson, T. Ma, A. J. Mackinnon, F. E. Merrill, G. L. Morgan, C. Munson, T. J. Murphy, P. J. Polk, D. W. Schmidt, R. Tommasini, I. L. Tregillis, A. C. Valdez, P. L. Volegov, T. S. F. Wang, C. H. Wilde, M. D. Wilke, D. C. Wilson, J. M. Dzenitis, B. Felker, D. N. Fittinghoff, M. Frank, S. N. Liddick, M. J. Moran, G. P. Roberson, P. Weiss, R. M. M. Alberti, S. S. Lutz, R. M. Malone, and A. Traillé, in *IFSA 2011—Seventh International Conference on Inertial Fusion Sciences and Applications*, edited by P. Mora, K. A. Tanaka, and E. Moses (EDP Sciences, Cedex A, 2013).
- <sup>14</sup>S. H. Batha, P. L. Volegov, V. E. Fatherley, V. Geppert-Kleinrath, and C. H. Wilde, *Rev. Sci. Instrum.* **89**, 10I147 (2018).
- <sup>15</sup>P. L. Volegov, S. H. Batha, V. Geppert-Kleinrath, C. R. Danly, F. E. Merrill, C. H. Wilde, D. C. Wilson, D. T. Casey, D. Fittinghoff, B. Appelbe, J. P. Chittenden, A. J. Crilly, and K. McGlinchey, *J. Appl. Phys.* **127**, 083301 (2020).
- <sup>16</sup>P. L. Volegov, C. R. Danly, D. Fittinghoff, V. Geppert-Kleinrath, G. Grim, F. E. Merrill, and C. H. Wilde, *J. Appl. Phys.* **122**, 175901 (2017).
- <sup>17</sup>P. A. Amendt, H. F. Robey, H.-S. Park, R. E. Tipton, R. E. Turner, J. L. Milovich, M. Bono, R. Hibbard, H. Louis, R. Wallace, and V. Yu. Glebov, *Phys. Rev. Lett.* **94**, 065004 (2005).
- <sup>18</sup>D. S. Montgomery, W. S. Daughton, B. J. Albright, A. N. Simakov, D. C. Wilson, E. S. Dodd, R. C. Kirkpatrick, R. G. Watt, M. A. Loomis, E. C. Merritt, T. Cardenas, P. Amendt, J. L. Milovich, H. F. Robey, R. E. Tipton, and M. D. Rosen, *Phys. Plasmas* **25**, 092706 (2018).
- <sup>19</sup>W. S. Varnum, N. D. Delamater, S. C. Evans, P. L. Gobby, J. E. Moore, J. M. Wallace, R. G. Watt, J. D. Colvin, R. Turner, V. Glebov, J. Soures, and C. Stoeckl, *Phys. Rev. Lett.* **84**, 5153 (2000).
- <sup>20</sup>R. M. Malone, G. A. Capelle, J. R. Celeste, P. M. Celliers, B. C. Frogget, R. L. Guyton, M. I. Kaufman, T. L. Lee, B. J. MacGowan, E. W. Ng, I. P. Reinbachs, R. B. Robinson, L. G. Seppala, T. W. Tunnell, and P. W. Watts, *Proc. SPIE* **6342**, 634220 (2006).
- <sup>21</sup>H. F. Robey, P. M. Celliers, J. L. Kline, A. J. Mackinnon, T. R. Boehly, O. L. Landen, J. H. Eggert, D. Hicks, S. Le Pape, D. R. Farley, M. W. Bowers, K. G. Krauter, D. H. Munro, O. S. Jones, J. L. Milovich, D. Clark, B. K. Spears, R. P. J. Town, S. W. Haan, S. Dixit, M. B. Schneider, E. L. Dewald, K. Widmann, J. D. Moody, T. D. Döppner, H. B. Radousky, A. Nikroo, J. J. Kroll, A. V. Hamza, J. B. Horner, S. D. Bhandarkar, E. Dzenitis, E. Alger, E. Giraldez, C. Castro, K. Moreno, C. Haynam, K. N. LaFortune, C. Widmayer, M. Shaw, K. Jancaitis, T. Parham, D. M. Holunga, C. F. Walters, B. Haid, T. Malsbury, D. Trummer, K. R. Coffee, B. Burr, L. V. Berzins, C. Choate, S. J. Brereton, S. Azevedo, H. Chandrasekaran, B. K. Young, M. J. Edwards, B. M. Van Wonterghem, B. J. MacGowan, J. Atherton, J. D. Lindl, D. D. Meyerhofer, and E. Moses, *Phys. Rev. Lett.* **108**, 215004 (2012).
- <sup>22</sup>J. D. Moody, H. F. Robey, P. M. Celliers, D. H. Munro, D. A. Barker, K. L. Baker, T. Döppner, N. L. Hash, L. B. Hopkins, K. LaFortune, O. L. Landen, S. LePape, B. J. MacGowan, J. E. Ralph, J. S. Ross, C. Widmayer, A. Nikroo, E. Giraldez, and T. Boehly, *Phys. Plasmas* **21**, 092702 (2014).
- <sup>23</sup>See <https://aframe.io/releases/1.1.0/aframe.min.js> for A-frame engine reference.