Reconstruction and analysis of exploding wire particle trajectories via automatic calibration of stereo images

M. Szott, Z. Wang, and D. N. Ruzic

Citation: Review of Scientific Instruments **89**, 10K118 (2018); doi: 10.1063/1.5039373
View online: https://doi.org/10.1063/1.5039373
View Table of Contents: http://aip.scitation.org/toc/rsi/89/10
Published by the American Institute of Physics

REVIEW OF SCIENTIFIC INSTRUMENTS **89**, 10K118 (2018)

Reconstruction and analysis of exploding wire particle trajectories via automatic calibration of stereo images

M. Szott,[1,a)] Z. Wang,[2] and D. N. Ruzic[1]
[1]*Department of Nuclear, Plasma, and Radiological Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA*
[2]*Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA*

(Presented 18 April 2018; received 7 May 2018; accepted 18 August 2018; published online 24 October 2018)

Quantitative understanding of the physics of dust or granular matter transport significantly impacts several aspects of burning plasma science and technology. This work takes machine vision techniques popular in robotics and self-driving cars and applies them to identification and analysis of microparticles generated from exploding wires. Using only the image frames and knowledge of the intrinsic properties of the cameras, a Python code was written to identify the particles, automatically calibrate the relative image positions, and extract trajectory data. After identifying approximately 50 particles based on the timing of secondary particle explosions, the eight point and random sample consensus algorithms were used to determine the geometric correlation between the cameras. Over 100 particle matches were found between the two camera views. These correlated trajectories were used in subsequent 3D track reconstruction and analysis of the physics behind the particle motion. The 3D reconstruction resulted in accurate positioning of the particles with respect to the experimental setup. The particle motion was consistent with the effects of a 1 g gravitational field modified by drag forces. The methods and analyses presented here can be used in many facets of high temperature plasma diagnostics. *Published by AIP Publishing.* https://doi.org/10.1063/1.5039373

I. INTRODUCTION

A better understanding of the physics of particulate matter transport is needed in many regions of high temperature plasma science, including but not limited to: improving fusion energy production through impurity control, suppressing or mitigating plasma stabilities through edge-localized-mode (ELM) pacing, fueling and controlling particle inventory and density profiles, mitigating disruptions using shattered cryogenic pellets, and gaining insight into tritium retention and transport that will assist the development of tritium removal technology. These impacts will only grow stronger as the field evolves toward longer pulse devices in which accumulation of deposited layers and associated injection of particulate matter becomes more pronounced.

Recent studies have started examining some of these topics. Lithium granule injector experiments in EAST and DIII-D successfully demonstrated the usefulness of granule injection for disruption mitigation and ELM triggering.[1] Another promising method for disruption mitigation in ITER involves injecting shattered cryogenic pellets.[2] High velocity imaging systems have been designed and tested on both laboratory systems and NSTX.[3,4] The physical basis for diagnosing laboratory plasmas utilizing dust transport has been described, along with a proposal for microparticle tracer velocimetry systems for use in various plasma flows.[5]

This work builds upon earlier studies[6] of microparticle tracking by showcasing a more robust analysis framework for use alongside imaging technologies. Implemented in Python, this method automatically identifies particles, calibrates stereo image series, and reconstructs full 3D particle trajectories for subsequent analysis. As an analog to moving particles in a fusion device, an exploding copper wire was used to create two image series with high particle density.[7] To capture these image series, two Vision Research Phantom v2511 cameras were used, running at 100 kfps with a vertical and horizontal pixel size of 28 μm. The raw image size is 384 × 384 pixels. The stereo images of the hot wire just prior to disintegration are shown in Fig. 1.

The pinhole camera model and two-camera epipolar geometry were used to mathematically describe the imaging setup (see Fig. 2). For a point x_1 on the image plane for camera 1, there is an unknown depth w of the point from the optical center of the camera. This unknown depth creates a ray of potential 3D positions of x_1. In camera 2, this ray would be seen as a line across the image, known as an epipolar line (L_2). The representation of the real particle on the image plane of camera 2, x_2, lies somewhere along this line. Likewise, the epipolar line (L_1) that x_2 would create in image plane 1 should contain point x_1. The relationship between the points and their epipolar lines is given by

$$[F][x_1] = [L_2] \; and \; [F][x_2] = [L_1], \tag{1}$$

where F is known as the fundamental matrix.

The fundamental matrix includes information about the intrinsic (inherent camera properties, such as pixel size, focal length, etc.) and extrinsic (translation and rotation) properties of the cameras. In most practical imaging configurations, the intrinsic properties of the cameras will be known. The extrinsic translation and rotation matrices between the cameras must

Note: Paper published as part of the Proceedings of the 22nd Topical Conference on High-Temperature Plasma Diagnostics, San Diego, California, April 2018.

a)Author to whom correspondence should be addressed: szott1@illinois.edu.

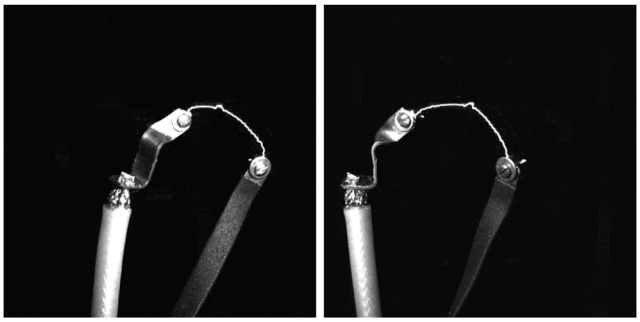

FIG. 1. Stereo images of the supporting leads and the wire immediately prior to disintegration. This support was removed via image subtraction during particle identification and analysis.

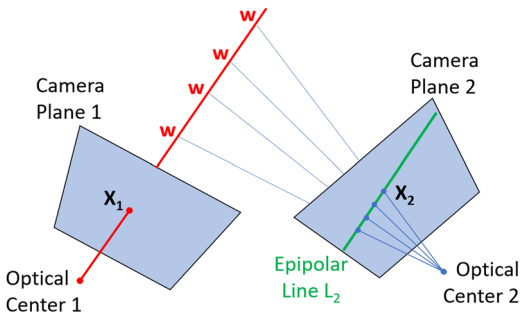

FIG. 2. Diagram of epipolar geometry for a two-camera stereo imaging setup, displaying how the depth ambiguity of particle w in image x_1 creates an epiline in camera plane 2 along which the second image, x_2, should lie.[8]

be found before 3-dimensional reconstruction of the particle positions can progress. This information is contained in the essential matrix, which can be found from the fundamental matrix. The 2-view reconstruction pipeline that was used to reconstruct these data can be described as follows:

(1) Identify particles and stitch trajectories in both views
(2) Find initial particle matches
(3) Determine and refine the fundamental matrix
(4) Estimate essential matrix
(5) Decompose essential matrix
(6) Estimate 3D particle positions

II. PARTICLE IDENTIFICATION AND TRAJECTORY STITCHING

To begin, the particles in each frame must be identified and logged. The open-source image analysis software ImageJ was used to subtract the background (the leads holding the wire before explosion) and artificially increase the number of pixels in the images.[9] The native 384 × 384 resolution of the fast frame camera images is too low for consistent particle identification, as particles sizes were on the order of 3 pixels wide or less. A bicubic interpolation routine was used to increase the resolution of each image in the series to 2000 × 2000. Although slower than linear interpolation, a bicubic interpolation uses more neighboring pixels (16 vs 4 for linear) and produces a smoother image with fewer interpolation artifacts. The interpolation appreciably increased the pixel size of each particle, allowing the Python module Trackpy to effectively identify them. Trackpy was developed as a particle identification suite for Python and includes useful qualifiers to help filter particle identification, including pixel size, brightness with respect to background, and particle separation.[10] In the case of these image series, standard particle brightness fell between 4000 and 6000 (a.u.), so a minimum brightness of 2000 was used along with an initial particle size estimate of 15 pixels. These qualifiers can be easily adjusted based on the image and application, and in this case, trial and error on smaller subsets of images determined that the above combination eliminated the vast majority of false particle identification.

The frame-by-frame positions of the identified particles were then stitched into individual trajectories using Trackpy tools. This was performed by extrapolating each particle's position and instantaneous velocity to the next frame, searching that frame for the particle and correlating the result frame-by-frame. The Trackpy module was built around the Crocker and Grier algorithm[11] for finding and tracking particles undergoing Brownian diffusion, which have no frame-to-frame velocity correlation, so the simplest application of the trajectory stitching functionality searches a range of surrounding pixels for the particle. Trackpy takes the Crocker-Grier algorithm farther and helps to maximize accurate trajectory stitching by including frame-by-frame instantaneous velocities of each particle and using them to correctly determine the continued trajectory if several potential matches are included in the search area.

Trackpy allows for the inclusion of a "memory" variable, which can account for particles that briefly become occluded or fall below the threshold brightness value. The position of the particle is remembered and extrapolated through subsequent frames, and if the particle reappears within the number of frames allotted, it will be re-matched to its original trajectory. This functionality was especially useful in a system involving numerous particles with low separation distances. Temporary occlusions were common, and this process proved capable of sorting out individual trajectories from high densities of overlapping particles. Some errors inevitably occur, so the data were filtered to remove any trajectories lasting fewer than 50 frames. This resulted in several hundred unique trajectories from each image series, offering a much larger data set than particle matching by the hand can provide.

III. AUTOMATIC CALIBRATION

After identifying and stitching the particle trajectories, calibration of the stereo images begins with identifying a set of particles that can be matched between the two views. Typically, this entails algorithmically identifying points of interest in a scene, such as edges and corners.[8] In this exploding wire scenario, the lack of extensive background and high particle density makes this type of correlation difficult. Instead, a set of matched particles was identified using particle brightness. During flight, the molten copper particles are extremely reactive. This leads to pyrophoric explosions of several particles, resulting in rapid, short-lived brightening events as they burst into several new particles. After the particle splits, the tracking algorithm tends to follow the brightest of the resulting

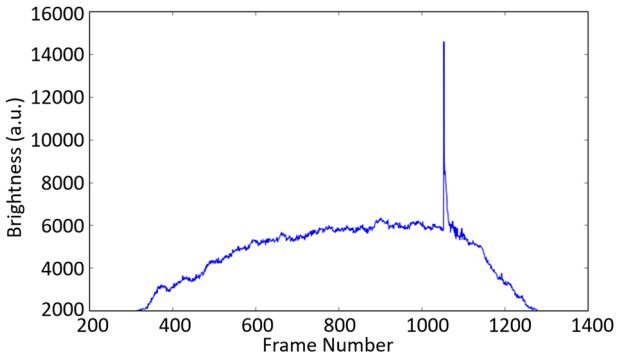

FIG. 3. Representative brightness plot of a particle vs. frame number, which is used to find a set of initial particle correlations for camera calibration.

fragments along its new trajectory. A representative particle brightness track is shown in Fig. 3.

These 2x-3x brightness increase events lasted 1-2 frames on average, providing an excellent filter for initial particle matches with highly accurate position data. The frequency of secondary explosions meant that several frames included more than 1 brightening event; these were discarded for ease of computation. Using this method, 57 particle pairs were found that brightened simultaneously in the same frame, while being the sole event in the said frame. This set of correlated initial particle matches was used as input for determination of the fundamental matrix describing the epipolar geometry.

There is ample theoretical basis for the correlation of stereo images. The eight point algorithm[12] requires at least 8 matched points between the two-image series to determine a fundamental matrix. The P ≥ 8 point pairs are given by

$$\{\vec{x}_1, \vec{x}_2\}_i = \left\{ \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \right\}_i. \tag{2}$$

The mean pixel position μ and variance σ^2 of the set of particles is computed.

$$\vec{\mu}_{1,2} = \sum_{i=1}^{P} (\vec{x}_{1,2})_i \big/ P, \tag{3}$$

$$\vec{\sigma}^2_{1,2} = \sum_{i=1}^{P} \left(\left(\vec{x}_{1,2}\right)_i - \vec{\mu}_{1,2} \right)^2 \Big/ P. \tag{4}$$

The particle positions are transformed so that they are centered around the origin and have a normalized distance to the center. This decreases the error in the fundamental matrix.

$$\left(\vec{x}_{1,2}\right)'_i = \left(\vec{x}_{1,2}\right)_i \Big/ \vec{\sigma}_{1,2} - \vec{\mu}_{1,2} \Big/ \vec{\sigma}_{1,2} = \vec{T}_{1,2} \left(\vec{x}_{1,2}\right)_i. \tag{5}$$

The constraint matrix A is found using the following:

$$A = [A_1; A_2; \ldots; A_P], \tag{6}$$

$$A_i = [x'_{2i}x'_{1i}, x'_{2i}y'_{1i}, x'_{2i}, y'_{2i}x'_{1i}, y'_{2i}y'_{1i}, y'_{2i}, x'_{1i}, y'_{1i}, 1]. \tag{7}$$

The fundamental matrix is a 3 × 3 reordering of the eigenvector corresponding to the smallest eigenvalue resulting from the singular value decomposition (svd) of the constraint matrix.

$$[U, L, V] = svd(A), \tag{8}$$

$$F = [v_{19}, v_{29}, v_{39}; v_{49}, v_{59}, v_{69}; v_{79}, v_{89}, v_{99}]. \tag{9}$$

The fundamental matrix is adjusted to account for the transformed coordinates.

$$F_{final} = T_2^T F T_1. \tag{10}$$

The fundamental matrix is then refined using the Random Sample Consensus (RANSAC) algorithm.[13] The RANSAC method allows for any number of input points and finds a fundamental matrix using random subsets of 8 points from the input set. Then the distance of each point from the epiline created by its counterpart in the corresponding image is calculated. If the distance falls above a threshold, the point is considered an outlier. On each iteration, RANSAC tallies the number of inliers from the full input set and maintains the fundamental matrix that has the largest number of inliers. The full set of inliers is then used in the eight point algorithm to further refine the fundamental matrix. In this way, it was found that 50 of the initial 57 potential matches were inlier points, with 7 outliers. The maximum distance from any inlier point to its epiline was 2.2 pixels, in a 2000 × 2000 pixel image. These points, and their matching epilines, are shown in Fig. 4.

To check the accuracy of the fundamental matrix based on the brightening event particle correlation, the fundamental matrix was tested on both background points from the leads and particle matches identified in the initial attempt at identifying particle trajectories by hand.[7] The successful validation showing all particles lying along their matching epilines is shown in Fig. 5.

It is now possible to deconvolve the fundamental matrix to recover the intrinsic and extrinsic properties of the imaging setup. The intrinsic matrices include inherent properties of the cameras used to create the images being analyzed. This includes focal lengths, pixel sizes, and optical centers (other corrections and parameters are taken as negligible).

The extrinsic properties, in the case of 2 cameras taking stereo images, include the physical translation and rotation between the 2 cameras. The technical specifications, and therefore the intrinsic properties, of the cameras are known. Using the known intrinsic matrices $\Lambda_{1,2}$ and the fundamental matrix, the essential matrix E can be found.[8]

$$E = \Lambda_2^T F \Lambda_1, where\ \Lambda = \begin{bmatrix} f/p_u & \gamma & u_0 \\ 0 & f/p_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{11}$$

Here, f is the focal length, $p_{u,v}$ are the horizontal and vertical pixel sizes, respectively, u_o, v_0 are the pixel coordinates of the optical center, and γ is the skew. Based on

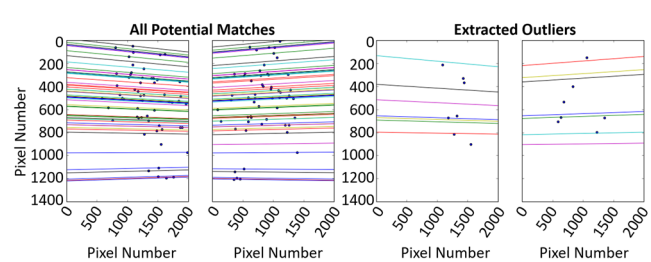

FIG. 4. (Left) All potential point matches used for correlation, plotted with their corresponding epilines. (Right) Outliers extracted after the RANSAC procedure. The epilines are color coded between stereo images, and all points are plotted at their position during their brightening event.

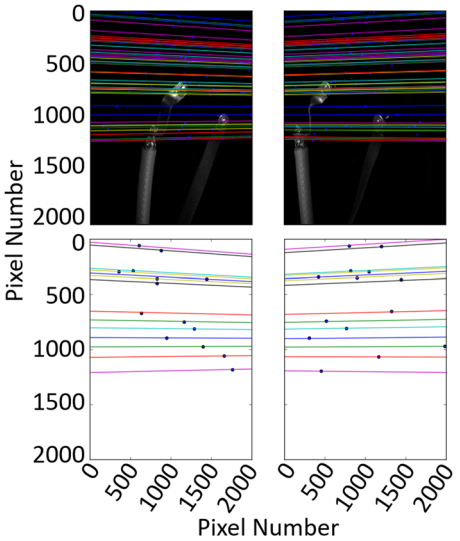

FIG. 5. Stereo views of points on the background wire frame (top) and initial analysis data gathered by hand (bottom)[7] match their corresponding epilines, validating the fundamental matrix.

the singular value decomposition of the essential matrix, the translation (τ) and rotation (Ω) matrices that describe the position of one camera in relation to the other can be derived.

$$W = [0,\ -1,\ 0;\ 1,\ 0,\ 0;\ 0,\ 0,\ -1], \tag{12}$$

$$[U,\ L,\ V] = svd(E), \tag{13}$$

$$\tau_\times = ULWU^T = \begin{bmatrix} 0 & -\tau_z & \tau_y \\ \tau_z & 0 & -\tau_x \\ -\tau_y & \tau_x & 0 \end{bmatrix} \rightarrow \tau = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix}, \tag{14}$$

$$\Omega = UW^{-1}V^T. \tag{15}$$

From this knowledge of the geometric relationship between the cameras, 3D reconstruction can progress.

IV. 3-DIMENSIONAL RECONSTRUCTION

To begin full 3D trajectory reconstruction, the fundamental matrix was used to correlate particles between frames using their corresponding epilines. Ideally, this should yield a single perfectly matched particle for the entire image series. However, due to the effects of finite pixel size, temporary occlusions, and high particle density/low separation distance, imperfect correlation was common. Therefore, all potential matches within a threshold distance were logged for each particle in each frame, and the resulting list of potential matches was filtered to yield only strongly matched trajectories with greater than 80% correlation over more than 500 frames. These matches are shown in Fig. 6.

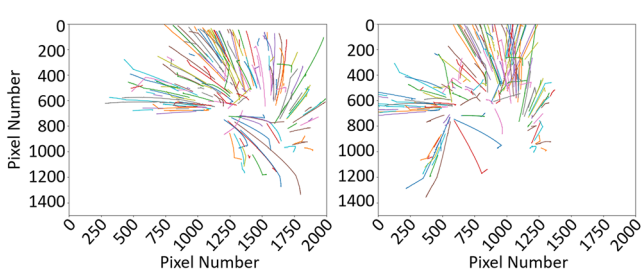

FIG. 6. Stereo images displaying the strongly matched trajectories. Trajectories are color coded between the images.

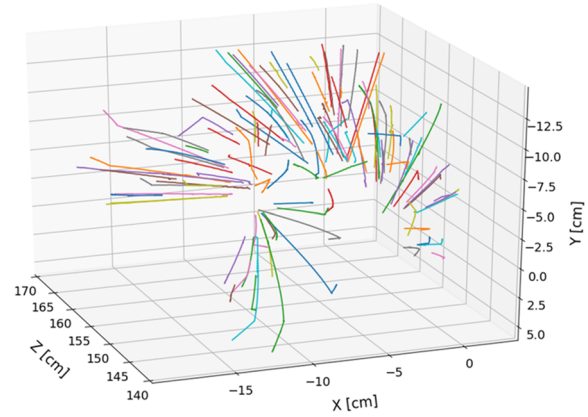

FIG. 7. Accurate 3D reconstruction of well-matched trajectories from 2 stereo images.

The sharp turns apparent in many of the trajectories is caused by the pyrophoric explosions of select particles. As they exploded into several new particles, the algorithms tended to track the largest piece along its new trajectory.

These secondary tracks were then filtered for analysis of particle motion. Utilizing the intrinsic and extrinsic properties of the cameras, the 3D reconstruction algorithm develops a set of constraints on the particle position in 3 dimensions.

There is inherent ambiguity in the sign of the translation and rotation matrices, resulting in 4 potential 3D reconstructions.[8] Once several particles are reconstructed, the extrinsic properties are found by determining the set that reconstructs the most (or all) points in front of both cameras. The 3D reconstruction, seen in Fig. 7, successfully recreated particle track with excellent agreement to experiment, placing particles between 140 and 160 cm from the cameras. Initial experimental wire placement was approximately 150 cm.

V. ANALYSIS OF PARTICLE MOTION

The reconstruction process yields reliable 3D position, velocity, and acceleration data for subsequent analysis of particle motion. However, in many trajectories, there is large variation caused by the "pixel effect." The particles move in a stair-step fashion because of the lack of subpixel resolution. In the original 384 × 384 resolution of the fast frame camera images, the particles were at most 3 pixels in diameter. Their smooth velocity is therefore segmented as they move across pixels. The bicubic method used to increase particle pixel size cannot eliminate this problem since it simply interpolates the original data. Since the experimental tests were performed in atmosphere, it was assumed that the motion of the particles was governed by gravity and drag. Simple equations of motion were derived from Stokes drag, utilizing a drag constant b and velocity dependent drag force.

$$F_D = -bv, \tag{16}$$

$$m\frac{d\vec{v}}{dt} = m\vec{g} - b\vec{v}. \tag{17}$$

The terminal velocity v_t is defined as

$$v_t = \frac{mg}{b} = \sqrt{\frac{8rg}{3C_D}}. \tag{18}$$

Here radius r and drag coefficient C_D are determined by the particle size and shape. Integrating Eq. (17) yields equations of motion for the 3D particle position.

$$x = \frac{v_{x0}v_t}{g}\left(1 - e^{-gt/v_t}\right) + x_0, \tag{19}$$

$$y = v_t t + \frac{v_t\left(v_{y0} - v_t\right)}{g}\left(1 - e^{-gt/v_t}\right) + y_0, \tag{20}$$

$$z = \frac{v_{z0}v_t}{g}\left(1 - e^{-gt/v_t}\right) + z_0. \tag{21}$$

The pixel effect mentioned above led to some uncertainty when deriving the initial particle position and velocity for use in trajectory fitting. In most cases, an initial velocity found by averaging velocity over the first stair-step eliminated most error, but better methods of smoothing the data and accounting for the pixel effect are still being investigated. Initially, the terminal velocity was determined using the median particle size of $r = 12.5\ \mu$m, based on post-experiment particle collection in ethanol.[7] The drag coefficient was set at 0.1, based on Reynold's number ($\sim 4 \times 10^5$) and smoothed aerodynamic shape of the recovered particles.[7] Using these values, the theoretical curves consistently undercut the real particle velocity, as can be seen in green in the representative particle trajectory in Fig. 8.

Several phenomena can help explain this discrepancy. First, the particles studied are most likely larger than the median. Based on qualitative examination, the particles chosen for analysis were predominantly larger and brighter than average and included many of the pyrophoric particles. Each of these particles split into an average of approximately 3.5 pieces before they would be collected, suggesting that the analyzed particles are much bigger than the median size.

To test this, a function was written to allow optimization of the drag fit to the particle motion, changing the particle radius to match the terminal velocity of each particle. This resulted in a much more accurate description of the particle motion, seen in red in Fig. 8. The optimized particle radii lie in the range of $r = 25 - 30\ \mu$m, which is consistent with the bias in the particle size discussed above.

Other factors that may help explain discrepancies between the modeled and observed particle motion include: the pixel effect biasing the initial velocity to a value lower than reality; a lower drag coefficient due to a higher Reynold's number or non-spherical particle shape; air expansion forces due to the explosion of the wire; and heat from the molten particles changing the local characteristics of the air. These avenues have yet to be fully explored.

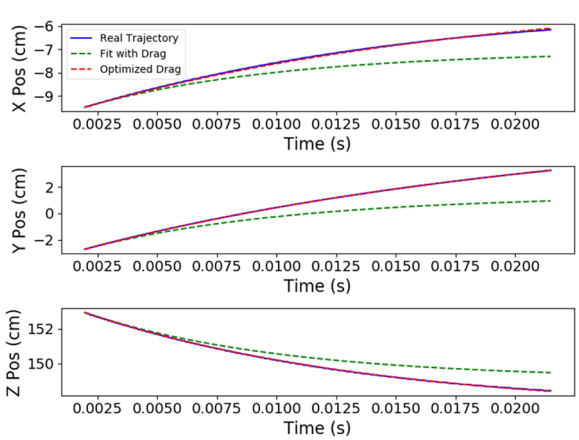

FIG. 8. Representative 3D particle trajectory, including simple drag (green) and optimized drag (red) curves based on the initial particle position and velocity.

VI. CONCLUSIONS

In this work, a Python framework was developed for accurate 3D reconstruction of particle motion from stereo 2D fast-frame camera image series. The reconstruction shown here does not require accurate foreknowledge of the placement of the cameras or any background calibration methods, merely the intrinsic properties of the cameras themselves. Requiring no manual input, the process self-calibrates the camera views and outputs 3D motion data for subsequent analysis. This pipeline for particle correlation can be easily adjusted for many applications, including tracking particles and impurities in a variety of high temperature plasma systems.

A sound basis for the physics governing particle motion was found, consistent with motion in a 1 g gravitational field governed by drag. Further refining of the results should correct for the pixel effect and justify the variation in terminal velocity seen in the optimized trajectory fits. Size bias may result from the fact that the large, long-track particles were more likely to explode, so the average size may be larger than indicated. Additional forces acting on the particles, such as localized heating of air, should be investigated.

ACKNOWLEDGMENTS

This work was supported by the Department of Energy Office of Science Graduate Student Research Program.

[1] D. K. Mansfield *et al.*, Nucl. Fusion **53**, 113023 (2013).
[2] L. R. Baylor, S. K. Combs, C. R. Foust, T. C. Jernigan, N. Commaux, S. J. Meitner, D. A. Rasmussen, P. B. Parks, and S. Maruyama, IAEA-CN–180, 2010.
[3] L. A. Dorf, A. L. Roquemore, G. A. Wurden, C. M. Ticos, and Z. Wang, Rev. Sci. Instrum. **77**, 10E517 (2006).
[4] Z. Wang, C. M. Ticoş, and G. A. Wurden, Phys. Plasmas **14**, 103701 (2007).
[5] C. M. Ticoş, Z. Wang, G. L. Delzanno, and G. Lapenta, Phys. Plasmas **13**, 103501 (2006).
[6] W. U. Boeglin, A. L. Roquemore, and R. Maqueda, Rev. Sci. Instrum. **79**, 10F334 (2008).
[7] Z. Wang, Q. Liu, W. Waganaar, J. Fontanese, D. James, and T. Munsat, Rev. Sci. Instrum. **87**, 11D601 (2016).
[8] S. J. D. Prince, *Computer Vision: Models, Learning and Inference* (Cambridge University Press, 2012).
[9] J. Schindelin *et al.*, Nat. Methods **9**, 676–682 (2012).
[10] D. Allan, T. Caswell, N. Keim, and C. van der Wel, Trackpy v0.3.2, 2016.
[11] J. C. Crocker and D. G. Grier, J. Colloid Interface Sci. **179**, 298–310 (1996).
[12] R. Hartley, IEEE Trans. Pattern Anal. Mach. Intell. **19**, 580–593 (1997).
[13] M. A. Fischler and R. C. Bolles, Commun. ACM **24**, 381–395 (1981).