PIV/BOS synthetic image generation in variable density environments for error analysis and experiment design

Lalit K Rajendran, Sally P M Bane and Pavlos P Vlachos

1 School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907, United States of America
2 School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, United States of America

E-mail: pvlachos@purdue.edu

Received 12 December 2018, revised 6 April 2019
Accepted for publication 25 April 2019
Published 28 June 2019

Abstract

We present an image generation methodology based on ray tracing that can be used to render realistic images of particle image velocimetry (PIV) and background oriented schlieren (BOS) experiments in the presence of density/refractive index gradients. This methodology enables the simulation of aero-thermodynamics experiments for experiment design, error, and uncertainty analysis. Images are generated by emanating light rays from the particles or dot pattern, and propagating them through the density gradient field and the optical elements, up to the camera sensor. The rendered images are realistic, and can replicate the features of a given experimental setup, like optical aberrations and perspective effects, which can be deliberately introduced for error analysis. We demonstrate this methodology by simulating a BOS experiment with a known density field obtained from direct numerical simulations of homogeneous buoyancy driven turbulence, and comparing the light ray displacements from ray tracing to results from BOS theory. The light ray displacements show good agreement with the reference data. This methodology provides a framework for further development of simulation tools for use in experiment design and development of image analysis tools for PIV and BOS applications. An implementation of the proposed methodology in a Python-CUDA program is made available as an open source software for researchers.

Keywords: particle image velocimetry, background oriented schlieren, image generation, experiment design, gradient index optics

(Some figures may appear in colour only in the online journal)
2. Image generation methodology

The image generation process, shown schematically in figure 2, comprises four steps: (1) generating the light rays, (2) tracing the light rays through density gradients, (3) propagating the light rays through optical elements, and (4) intersecting the rays with the camera sensor to update the pixel intensities. Each of these steps is described in more detail in the following sections.

2.1 Generating the light field

The light rays generated from the particles/dot pattern are considered as vectors connecting source points in the flow field to points of intersection on the camera lens. The source point can be a particle for a PIV experiment or a dot pattern for a BOS experiment. The origin of the light ray vector corresponds to the position of the source point, and its direction corresponds to a unit vector connecting the origin to the point of intersection on the camera lens. Ideally, an infinite number of such light rays can be generated from each source point towards points located on the camera lens. Increasing the number of light rays increases the dynamic range of the generated images but also increases the computational cost.

The radiance of the light ray may have an angular dependence based on the type of scattering associated with the source point. In the case of a PIV particle field where the particle diameters are typically of the same order of the wavelength of the laser, the radiance of the light ray can be estimated using Mie scattering [11]. The scattering cross-section and efficiency depends on the size of the particle, the wavelength of the laser beam, the relative refractive index of the particle with respect to the medium, and the angle between the light ray vector and the direction of propagation of the laser beam. The Mie scattering computations are performed using the method outlined in Bohren and Huffman [12].

2.2. Tracing rays through density gradients

A light ray will experience changes in its direction as it passes through a medium containing density gradients due to the dependence of the refractive index on the local density as expressed by the Gladstone–Dale relation:

\[ n = K \rho + 1, \]

where \( n \) is the refractive index of the medium, \( \rho \) is the density, and \( K \) is the Gladstone–Dale constant, which has a value of 0.226 cm\(^3\) g\(^{-1}\) for air. Therefore, regions of density gradients also contain refractive index gradients. For a medium containing a continuous change of refractive index, Fermat’s principle from geometric optics enables a fast and accurate computation of the trajectory of a light ray through the medium, and the equation for the ray curve is given by

\[ \frac{d}{d\xi} \left( n \frac{d\xi}{d\zeta} \right) = \nabla n. \]

(2)

Here \( x(\xi) \) represents the ray curve and \( (\xi, \eta) \) are the ray-fitted co-ordinates as shown in figure 2. Equation (2) is transformed and discretized using a 4th order Runge–Kutta algorithm following the method of Sharma et al [14] and the position and direction of the light ray passing through the variable density medium can be updated based on the local refractive index gradient as follows:

\[ R_{t+1} = R_t + \left[ T_t + \frac{1}{6} \left( A + 2B \right) \right] \Delta \xi, \]

\[ T_{t+1} = T_t + \frac{1}{6} \left( A + 4B + C \right), \]

where \( R, T \) are 1D arrays representing the position and direction, respectively, and are given by

\[ R = \begin{pmatrix} x \\ y \\ z \end{pmatrix}; \ T = n \begin{pmatrix} dx/d\xi \\ dy/d\xi \\ dz/d\xi \end{pmatrix}. \]

(4)
The variable $n$ is the refractive index and the subscript $i$ represents the grid point corresponding to the given location of the ray. The constants $A$, $B$ and $C$ are functions of the refractive index gradients and are given by

$$A = D \left( R_i \right) \Delta \xi$$
$$B = D \left( R_i + \left( \frac{1}{2} T_n + \frac{1}{8} A \right) \Delta \xi \right) \Delta \xi$$
$$C = D \left( R_i + \left( T_n + \frac{1}{2} B \right) \Delta \xi \right) \Delta \xi$$

and the function $D$ is given by

$$D = n \left( \frac{\partial n}{\partial x} \right) = \frac{1}{2} \left( \frac{\partial n^2}{\partial x} \right.$$

An open-source implementation of solving Fermat’s equation on a GPU with a piecewise linear approximation (1st order) was provided by SchlierenRay, an artificial schlieren
image rendering software developed by Brownlee et al [15]. Their methodology has been extended to include higher order discretizations and integrated with a full light field-based ray tracing approach for the present application.

2.3. Propagating light rays through optical elements

When light rays pass through optical elements, they can undergo one or more of the following processes: (1) reflection (mirrors), (2) refraction (lenses, windows), and (3) selective transmission (apertures). All of these processes are modeled in the ray tracing methodology, as shown in figure 2. In all cases, the intersection of a ray with the optical element is first computed based on the element’s geometry. For example, in the case of a spherical mirror/lens the intersection point is calculated based on the element center, diameter, and radius of curvature. After computing the intersection, the effect of the element is modeled as follows:

1. Reflection due to mirrors is modeled using the law of reflection based on the direction of the light ray with respect to the local surface normal.
2. Refraction due to lenses/windows is modeled using Snell’s law [16], given by
   \[ n_1 \sin(\theta_i) = n_2 \sin(\theta_f), \]
   where \( \theta_i \) is the angle of incidence, \( \theta_f \) is the angle of refraction, and \( n_1 \) and \( n_2 \) are the refractive indices of the two media on either side of the refractive surface. For elements with multiple refractive surfaces like a lens, the refraction is performed sequentially on each surface, considering the possibility of total internal reflection if the ray passes from a medium of higher refractive index to a medium of lower refractive index. It should be noted that this approach is quite general and does not require assumptions regarding the paraxial nature of the light rays (as used in matrix methods) or the thickness of the lens, and it is straightforward to include transmittance and dispersive effects of the lens as required. Further, an array of lenses, as in a plenoptic camera, for example, can also be modeled using this approach.
3. Selective transmission due to apertures is enforced by only allowing light rays that intersect the plane of the aperture and lie within its opening area (or pitch) and blocking the rest.

2.4. Intersecting a ray with the camera sensor and incrementing pixel intensities

The final step in the ray tracing process is the intersection of a light ray with the camera sensor, which is solved as a line-plane intersection problem. The diffraction spot is described by an Airy function and is approximated by a Gaussian in this application [17], and the integrated intensity across a pixel is calculated using an error function, as in the case of synthetic PIV image generation [18]. The point of peak intensity is the point of intersection of the light ray with the camera sensor, and the diffraction diameter is a function of the optical system as given by

\[ d_r = 2.44 \pi f_{\text{in}} (M + 1) \lambda. \]

Here \( d_r \) is the diffraction diameter, \( f_{\text{in}} \) is the f-number of the camera, \( M \) is the magnification, and \( \lambda \) is the wavelength of light [17]. For white light illumination in the case of BOS/calibration targets, an effective wavelength corresponding to the green color is used.

This procedure is repeated for all light rays that intersect the camera sensor to obtain an image of the particle field/dot pattern. The dynamic range of the final intensity distribution increases with the number of light rays used to render a particle or dot, but this also increases the computational cost and run time. It was observed from trials that about 10000 rays are sufficient to provide a 16-bit dynamic range.

2.5. Parallelization using CUDA

The ray tracing methodology just described is computation-intensive due to the large number of light rays (~ 1 billion) required to render an image with sufficient dynamic range. Since the trajectories of the light rays are independent of each other, the ray tracing calculations can be parallelized using GPUs. This methodology was implemented using a CUDA framework with a Python front-end. The images in the present work were generated using an NVIDIA Tesla C2050 GPU, which has 14 streaming multi processors each containing 32 cores for an overall total of 448 cores. Each multi-processor can launch a maximum of 1536 threads amounting to a total of about 21 000 threads at a time.

The details of the parallelization in terms of grids, blocks and threads are as follows. Each thread on the GPU corresponds to a single light ray, and all the computations starting from the ray generation to the intersection with the camera sensor are done independently. All light rays originating from the same particle/dots are organized in blocks, to take advantage of the shared memory in CUDA which has very fast read and write speeds [19]. Thus the information common to all light rays originating from the same particle are stored in shared memory, which frees up the local memory and enables launching a larger number of threads. The number of threads that can be stored in a block and the number of blocks that can be launched are subject to hardware limitations.

In summary, the approach presented in this paper is an integrated implementation of state of the art methods for the various components of the image generation methodology in one package for simulating general aero-thermodynamics experiments. The improvements presented in this approach and their possible applications are as follows:

1. Accurate ray tracing through density gradients with higher order Runge–Kutta schemes, which becomes important in simulating flows involving sharp changes in density, such as experiments with shock waves.
2. Non-linear ray tracing through the optical elements without any paraxial or thin lens approximations. This allows us to introduce optical aberrations in a controlled manner, as
3. Error analysis

The accuracy of the image generation methodology was simulated using three cases: (a) Luneburg lens to test the ray tracing through the density gradients, (b) a full BOS experiment with a known density field and user defined camera optics and comparing the final light ray deflections recorded on the camera sensor to predicted displacements from BOS theory.

3.1. Luneburg lens

The Luneburg lens [20] is a gradient index lens with the refractive index distribution within the lens given by

\[ n(r) = \sqrt{2 - \left(\frac{r}{R}\right)^2}, \]

where \( n \) is the refractive index, \( r \) is the radial co-ordinate, and \( R \) is the radius of the lens. The lens has the property that an incoming parallel beam of light rays is focused on to the optical axis at the back surface of the lens, and is a standard benchmark test in the gradient-index optics literature [14, 21, 22].

This lens is used as the refractive index medium and the light rays are traced through it using the method outlined above. Some sample light ray trajectories are shown in figure 3(a) and it is seen that rays entering at various heights are focused very close to the exit plane. The positions of the light rays on the exit plane of the lens are recorded and any deviation from zero is considered to be an error. This test enables us to isolate the part of the method used to trace rays through density gradients from the overall image generation methodology and validate it separately.

Monte-Carlo simulations were performed for 1000 light rays entering the lens at random \( X, Y \) locations, and the average exit height error was calculated upon leaving the lens. The number of grid points used to represent the refractive index field of the Luneburg lens were varied from 25 to 250, and the average exit height errors are plotted as a function of grid points for a 1st-order Euler method and the 4th-order Runge–Kutta (RK4) method in figure 3(b). It is seen that the error levels are low for both methods and that the RK4 method gives a lower error than the Euler method for any grid point. Further, it is seen that the RK4 method also has a higher rate of decrease of error with increasing grid points due to the higher order of the method. This test serves as a validation for the image generation methodology for tracing rays through the density gradients.

3.2. BOS simulation

For the BOS test, two density fields were considered: (1) a constant density gradient field designed to create a uniform displacement of the dot pattern and (2) a more realistic density field taken from a direct numerical simulation (DNS) of homogeneous buoyancy-driven turbulence. The layout of the BOS experimental setup modeled in the image generation software for simulations involving both density fields is shown in figure 4, and the parameters describing the placement of the elements are summarized in table 1. The refractive index experienced by a light ray is a function of its wavelength, and decreases with increasing wavelength in the visible range. Therefore the angular deflection experienced by a light ray also changes with wavelength, and this issue
has been analyzed for synthetic schlieren measurements by Kolaas et al [23]. A monochromatic light source with a wavelength of 532 nm was considered in this study and the ambient refractive index is set to be 1.00028.

3.2.1. Uniform density gradient. For this case, a density field with a constant gradient was simulated, where the density gradient was designed to create a uniform displacement of the light rays emerging from a dot pattern. The magnitude of the density gradient field was calculated from the desired pixel displacement and the optical layout of the system using BOS theory. The theoretical displacement of a light ray for a BOS experiment is given by

$$\Delta \vec{X} = \frac{MZ_D}{n_0} \left( \nabla \rho \right)_{\text{avg}} L_z,$$

(10)

where $\Delta \vec{X}$ is the theoretical deflection of a light ray, $(\nabla \rho)_{\text{avg}}$ is the path-averaged value of the density gradient, $K$ is the Gladstone–Dale constant, $n_0$ is the ambient refractive index, and $L_z$ is the depth/thickness of the density gradient field [2]. Using the above equation, a value of $(\nabla \rho)_{\text{avg}}$ is calculated using the values of the experimental parameters from table 1. This test enables us to test the entire simulation chain without the spatial resolution limitations involved with BOS measurements.

Tests were conducted with the theoretical displacement field being varied from 0 to 3 pix. to provide a range representative of typical BOS experiments, and the average displacements of all light rays from the field of view is shown in figure 5. It can be seen that there is good agreement between the theoretical displacements and calculated displacements from the ray tracing simulations.

3.2.2. Buoyancy driven turbulence. The DNS data used for this test are from simulations performed by performed by Livescu et al [24–26] and downloaded from the Johns Hopkins University Turbulence Database (JHU-TDB) [24, 27, 28]. 2D $(x, y)$ slices of the flow field from two time instants were chosen, and for each time instant, a 3D density volume was constructed by stacking the same 2D slice along the $z$-direction, thereby ensuring that the gradient of density in the $z$ direction was zero. This was done to account for the depth integration limitation of BOS measurements and to enable a better comparison of the simulated light ray deflections to theory. The refractive index was calculated using equation (1) using the Gladstone–Dale constant for air, and for the case with the DNS data, the non-dimensional density field was scaled by a factor of 1.225 kg m$^{-3}$ to simulate air properties. The values of the experimental parameters were taken from table 1. The contours of the input density and density gradients, the theoretical displacements calculated from equation (9), and the light ray displacements from ray tracing simulations are shown in figure 5. The depth averaged density gradient $(\nabla \rho)_{\text{avg}}$ used to calculate the theoretical displacements is taken to be the 2D density gradient field shown in figure 6, as identical 2D slices were stacked to create a 3D density field during the simulations.

The light ray displacements from the ray tracing simulations will be randomly scattered on the camera sensor due to
the random positions of the dots on the target from which the light rays originate. The ray displacements corresponding to a single dot are averaged and interpolated onto a regular grid using a bilinear interpolation and displayed in figure 4. The figure shows that the contours of light ray displacements from the simulations closely correspond to the theoretical displacements except that they are smoothed out. The mismatch between the theoretical and simulated light ray deflections is due to two reasons: (1) the theoretical equation is based on small angle approximations, and (2) the spatial resolution limitation of the BOS experimental setup whereby the light ray deflection of a dot is the average light ray displacement of all rays comprising a ray cone. Both these effects are consistent with well-known characteristics of BOS experiments [2, 29, 30]. Further, it is to be noted that when these two effects are negligible, as in section 3.2.1 with the uniform density field, the methodology is able to accurately match the theoretical displacements.

Overall, these results show that the proposed methodology is capable of generating accurate synthetic images for user-defined density fields and optical layouts.

4. Application: trade-off between measurement sensitivity and spatial resolution for BOS experiments

To further illustrate the capability of the proposed image generation methodology to aid in experiment design, we show the application of the methodology for assessing the tradeoff between measurement sensitivity and spatial resolution for BOS experiments.

In BOS experiments, the relative placement of the density gradient field with respect to the dot pattern and the camera lens is a crucial parameter that determines the measurement quality. In particular, the parameter $Z_D$ which denotes the distance between the dot target and the mid plane of the density gradient field (shown in figure 4), controls both the measurement sensitivity, defined as the apparent displacement of a light ray produced per unit angular deflection, and the spatial resolution of the measurement. This issue has been theoretically analyzed in the past by Gojani et al [31] using simplified models for the optics and the density gradient field. Here, we apply the ray tracing methodology to directly evaluate this effect without simplifying assumptions.

The parameter $Z_D$ has a contradictory effect on the sensitivity and spatial resolution because, the measurement sensitivity ($= MZ_D$) increases with $Z_D$, but for increasing values of $Z_D$, a ray cone emerging from the dot pattern covers a larger area of the density gradient field. Since the apparent displacement of a dot recorded on the sensor is the average displacement of all light rays within the ray cone, a larger ray cone leads to more averaging and a loss of spatial resolution.

To illustrate this effect, we simulate a BOS experiment with the density field given by a sharp jump designed to represent a normal shock. The density and corresponding density gradient distribution along $x$ are shown in figure 7, and both fields are uniform along $y$ and $z (\frac{\partial \rho}{\partial y} = \frac{\partial \rho}{\partial z} = 0)$. For a fixed distance $Z_B$ between the camera lens and the dot pattern, three positions of the density gradient field are simulated ($Z_D/Z_B = 0.25, 0.5, 0.75$). For each case, an image of a dot pattern with and without the density gradient field is rendered, and the apparent displacements of the light rays are calculated.

Figure 8(a) shows the density gradient field experienced by all rays emerging from a single dot in the center of the image, and it is seen that as $Z_D/Z_B$ increases, the region of the density gradients covered by the ray cone increases. Since the apparent displacement of a dot on the sensor corresponds to the average
of this density gradient distribution, light rays from adjacent dots travel through overlapping regions of the density gradient field, leading to a loss in spatial resolution. This effect is shown in figure 8(b) showing the average displacements of light rays emerging from a row of dots along $x$, where it is seen that the displacement field recorded on the sensor has a higher peak displacement for higher values of $Z_D/Z_B$ (corresponding to a higher measurement sensitivity), but more smoothed, corresponding to a lower spatial resolution. Overall, these trends are in agreement with the analysis by Gojani et al. [31] It is to be noted however, that as $Z_D/Z_B$ increases, the magnification of the density gradient field (different from the magnification of the dot pattern) increases, and hence for the cases with $Z_D/Z_B = 0.5$ and 0.75, the displacement field appears asymmetric because of clipping due to the finite size of the camera sensor.

These results, in addition to the sample particle images shown in figure 1, illustrate the capability of the proposed image generation methodology to accurately generate realistic PIV/BOS images. The methodology enables the introduction of experimental artifacts such as optical aberrations and distortions due to density gradient fields into the image generation process in a deliberate and controlled manner.

5. Conclusion

An image generation methodology was proposed and implemented to render realistic PIV and BOS images in variable density environments with a user-defined optical setup. The methodology involves generation of light rays from a particle or dot pattern, propagation of the light rays through density gradients using Fermat’s equation and a 4th order Runge–Kutta scheme, reflection/refraction/transmission of the light rays by optical elements, and intersection of the rays with the camera sensor to update pixel intensities using a diffraction model. The computationally intensive ray tracing process was parallelized and implemented on GPUs using CUDA, resulting in a significant acceleration of the computations. The accuracy of the methodology was evaluated using three cases: (1) a Luneburg lens, (2) a BOS experiment a uniform density gradient field.
and (3) a BOS experiment with a density field obtained from DNS of buoyancy-driven turbulence. The light ray deflections from the ray tracing show good agreement with the theoretical estimates. This methodology provides a framework for further development of simulation tools for use in experiment design by incorporating additional features specific to a given experiment. The methodology can also be a valuable tool for error analysis to study the effect of various elements of an optical setup on the final error, and provide directions to improve image analysis tools for PIV and BOS applications. A Python-CUDA program implementing the proposed methodology is available for download at the following link: https://github.rcac.purdue.edu/lrajendr/photon.git.

Acknowledgment

Rod LaFoy is acknowledged for early efforts in the development of the ray tracing-based image generation methodology presented in this work. This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Fusion Energy Sciences under Award Number DE-SC0018156. This work was also supported by NSF 1706474.

ORCID iDs

Lalit K Rajendran https://orcid.org/0000-0002-5673-6063
Pavlos P Vlachos https://orcid.org/0000-0002-8040-9257

References

[1] Raffel M, Willert C E, Wereley S T and Kompenhans J 2013
Particle Image Velocimetry: a Practical Guide (Berlin: Springer)

[2] Dalziel S B, Hughes G O and Sutherland B R 2000 Whole-field density measurements by ‘synthetic schlieren’ Exp. Fluids 28 322–35

[3] Alahyari A and Longmire E K 1994 Particle image velocimetry in a variable density flow : application to a dynamically evolving microburst Exp. Fluids 17 434–40

[5] Jackson P, Musalem R, Rehmann C and Hill D 2002
Particle image velocimetry errors due to refractive index fluctuations Int. Symp. on Hydraulic Measurements and Experimental Methods pp 1–9

[6] Elsegla G E, Van Oudheusden B W and Scarano F 2005
Evaluation of aero-optical distortion effects in PIV Exp. Fluids 39 246–56

[7] Dalziel S B, Carr M, Sween J K and Davies P A 2007
Simultaneous synthetic schlieren and PIV measurements for internal solitary waves Meas. Sci. Technol., 18 533–47

[8] Acheson B, Ihlke I, Heidrich W, Tevs A, Bradley D, Magnor M and Seidel H-P 2008
Time-resolved 3d capture of non-stationary gas flows ACM Trans. Graph. 27 1

[9] Berger K, Ihlke I, Acheson B, Heidrich W and Magnor M 2009
Tomographic 4D reconstruction of gas flows in the presence of occluders Vision, modeling, and Visualization Work pp 1–8

[10] Hartmann U and Seume J R 2016
Combining ART and FBP for improved fidelity of tomographic BOS Meas. Sci. Technol., 27 097001

[11] van de Hulst H C 1981
Light Scattering by Small Particles (New York: Dover)

[12] Bohren C F and Huffman D R 1998
Absorption and Scattering of Light by Small Particles (Weinheim: Wiley)

[13] Born M and Wolf E 1990
Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light (New York: Pergamon)

[14] Sharma A, Kumar D V and Ghatak A K 1982
Tracing rays through graded-index media: a new method Appl. Opt. 21 984–7

[15] Brownlee C, Pegoraro V, Shankar S, McCormick P and Hansen C 2010
Physically-based interactive schlieren flow visualization IEEE Pacific Vision Symp. 2010, PacificVis 2010—Proc. pp 145–52

[16] Bedritti F L, Pedrotti L M and Pedrotti L S 2007
Introduction to Optics (San Francisco: Pearson)

[17] Adrian R J and Yao C S 1985
Pulsed laser technique application to liquid and gaseous flows and the scattering power of seed materials Appl. Opt. 24 44–52

[18] Raffel M, Willert C E, Wereley S T and Kompenhans J 2007
Particle Image Velocimetry (Berlin: Springer)

[19] NVIDIA 2015
CUDA C Programming Guide (NVIDIA Corporation)

[20] Marchand D 1978
Gradient Index Optics (New York: Academic)

[21] Hewak D and Lit J 1985
Numerical ray-tracing methods for gradient index media Can. J. Phys. 63 234–9

[22] Moore D T 1980
Gradent-index optics: a review Appl. Opt. 19 1035

[23] Kolaas J, Hervold B, Sven K and Jensen A 2018
Bichromatic synthetic schlieren applied to surface wave measurements Exp. Fluids 59 1–13

[24] Livescu D, Canada C, Kalin K, Burns R, Staff I and Pulido J 2014
Homogeneous buoyancy driven turbulence data set Los Alamos National Laboratory Technical Report 2014.LA-UR-14-20669 (http://turbulence.pha.jhu.edu/docs/README-HBDT.pdf)

[25] Livescu D and Ristorcelli J R 2008
Variable-density mixing in buoyancy-driven turbulence J. Fluid Mech. 605 145–80

[26] Livescu D and Ristorcelli J R 2007
Buoyancy-driven variable-density turbulence J. Fluid Mech. 591 43–71

[27] Li Y, Perlman E, Wan M, Yang Y, Meneveau C, Burns R, Chen S, Szalay A and Eyink G 2008
A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence J. Turbul. 9 N31

[28] Perlman E, Burns R, Li Y and Meneveau C 2007
Data exploration of turbulence simulations using a database cluster Proc. 2007 ACM/IEEE Conf. Supercomputing (SC ’07) (https://doi.org/10.1145/1362622.1362654)

[29] Elsegla G E, Van Oudheusden B W, Scarano F and Watt D W 2004
Assessment and application of quantitative schlieren methods: calibrated color schlieren and background oriented schlieren Exp. Fluids 36 309–25

[30] Hargather M J and Settles G S 2012
A comparison of three quantitative schlieren techniques Opt. Lasers Eng. 50 8–17

[31] Gojani A B, Kamishi B and Obayashi S 2013
Measurement sensitivity and resolution for background oriented schlieren during image recording J. Vis. 16 201–7