Errors in particle tracking velocimetry with high-speed cameras

Yan Feng,[a)] J. Goree, and Bin Liu

Department of Physics and Astronomy, The University of Iowa, Iowa City, Iowa 52242, USA

Velocity errors in particle tracking velocimetry (PTV) are studied. When using high-speed video cameras, the velocity error may increase at a high camera frame rate. This increase in velocity error is due to particle-position uncertainty, which is one of the two sources of velocity errors studied here. The other source of error is particle acceleration, which has the opposite trend of diminishing at higher frame rates. Both kinds of errors can propagate into quantities calculated from velocity, such as the kinetic temperature of particles or correlation functions. As demonstrated in a dusty plasma experiment, the kinetic temperature of particles has no unique value when measured using PTV, but depends on the sampling time interval or frame rate. It is also shown that an artifact appears in an autocorrelation function computed from particle positions and velocities, and it becomes more severe when a small sampling-time interval is used. Schemes to reduce these errors are demonstrated. © *2011 American Institute of Physics*. [doi:10.1063/1.3589267]

I. INTRODUCTION

Particle tracking velocimetry (PTV) is a method to measure particle velocities[1] with video camera recording. In experiments, the particles are small solid objects that can scatter enough light to be imaged separately. The velocity is calculated based on measured positions of particles. Among the algorithms used to calculate velocities,[2–4] the most common is simply to divide the difference in a particle's position in two consecutive video frames by the time interval between the frames.[1,5,6] Although PTV may allow tracking individual particles for many frames, for this algorithm it is only necessary to track for two frames.

Particle tracking velocimetry has been widely used for many years in topics in various fields, such as cell motion in biology,[7] flow in granular materials,[8] and kinetic temperature in dusty plasmas.[5,6,9–13] Small particles of solid matter can be added as tracers in a gas or liquid to study convection[14] and turbulence[3] in fluid mechanics. Many dynamical quantities can be calculated using velocities measurements with PTV, for example, velocity profiles in a shear flow,[8,10] mean-square velocity fluctuations,[6] velocity distribution functions,[9,12,15] and velocity autocorrelation functions.[16] Some of these uses for PTV, such as velocity profiles, can also be accomplished with particle image velocimetry (PIV) (Refs. 17 and 18). Compared to PIV, PTV provides a measurement of velocity at the location of a particle, without requiring an averaging over a grid.

Recently, many experimenters using PTV have taken advantage of the abundance of high-speed cameras now offered for commercial sale. They may be unaware, however, that velocities determined using the PTV method can have errors that become more severe as the camera frame rate is increased. For example, we show that when determining kinetic temperature by calculating the mean-square velocity fluctuation for random motion, the result will have an exaggerated value that

worsens at higher frame rates, due to one of the two kinds of errors studied here.

II. VELOCITY ERRORS IN PTV

We identify two kinds of velocity errors in PTV according to their source. One kind arises from acceleration of the particle during the time interval between measurements of its position. The other arises from errors in the position measurements themselves. While the former is made less severe by using a faster frame rate, the latter is actually made worse. Therefore, a faster frame rate is not always best for PTV.

A. Velocity error arising from acceleration

An error arises from acceleration, whether due to a change in a particle's speed or its direction. This error occurs for all kinds of acceleration, for example, a particle accelerating along a straight line, moving in a circle at a steady speed, or colliding with another particle.

It is impossible to eliminate this error when the motion is unknown during the sampling interval between frames. The only information available in PTV is the particle positions determined at the times that a video image was recorded, i.e., the video frames.

The size of this error also depends on the algorithm for calculating velocity from data for the particle positions. The simplest and most common algorithm is to assume that the particle moves with zero acceleration between two consecutive frames. In this two-frame tracking method, the particle is assumed to move in a straight line at a steady speed between the positions in two frames. Thus, the velocity is calculated simply as the difference in positions divided by the time interval between frames. Using two position measurements $\mathbf{x}_{\text{meas}}$, denoted by j and $j+1$ separated by a time Δt, the velocity at a time halfway between them is calculated as

$$\mathbf{V} = \frac{\mathbf{x}_{j+1,\text{meas}} - \mathbf{x}_{j,\text{meas}}}{\Delta t}. \tag{1}$$

---

a)Electronic mail: yan-feng@uiowa.edu.
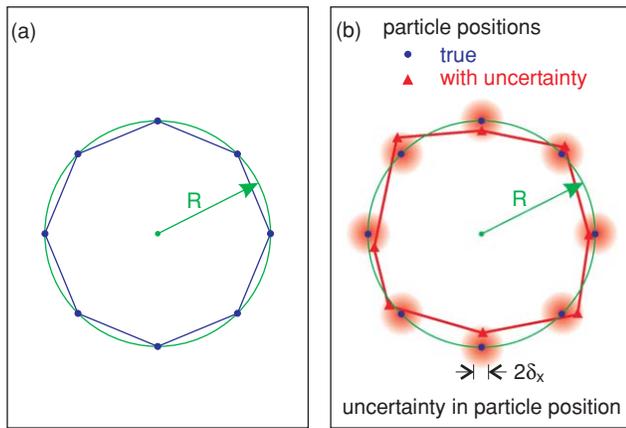
FIG. 1. (Color online) Simulations of particle trajectories for circular motion (a) with no particle-position uncertainty and (b) with particle-position uncertainty. The true particle positions at time intervals Δt are marked by circular dots. The triangles in (b) indicate particle positions including an error from a Gaussian distribution with an uncertainty δ_x.

The camera frame rate is $1/\Delta t$ if no frames are skipped. We use upper-case symbols such as V and X to indicate quantities that are *computed* from the positions that are measured in a single frame. If the particle actually undergoes acceleration, for example, if it has a curved trajectory, the simple algorithm of Eq. (1) will obviously lead to errors in the velocity V. Alternatively, one could use an algorithm using more than two position measurements, which can sometimes better account for acceleration. A spline fit[3] could reduce the error arising from acceleration, but it is more computationally expensive than two-frame tracking.

To illustrate how errors in V arise from acceleration and how these errors diminish with a higher frame rate, we consider the motion during a Coulomb collision of a pair of identical electrically charged particles. The largest acceleration in this case will occur when the particles are closest and the direction of motion is changing most rapidly. This highly curved portion of a trajectory can be approximated as a circular arc. This motivates us to choose uniform circular motion as a simple instructive example of velocity errors arising from acceleration, see Fig. 1. Suppose the particle position is measured accurately at each frame, as indicated by the dots in Fig. 1(a). The simple two-frame tracking method, which assumes zero acceleration between measurements, will describe the motion as a polygon instead of the ideal circle. At a fast frame rate, the polygon has more sides and more closely approximates the ideal circle.

B. Velocity error arising from position uncertainty

Another error in velocity arises from uncertainties in the particle positions from which the velocity is computed. Particle positions are uncertain for at least two reasons: random noise in the camera sensor and the finite size of pixels in the sensor. The latter leads to the phenomenon of pixel locking, where particles are wrongly identified as being located at favored positions such as the corner or middle of pixels. While it is possible to design an experiment to reduce these particle-position uncertainties,[1,13] they can never be eliminated.

Suppose that all measured particle positions contain an uncertainty $\delta\mathbf{x}$. We can use propagation of errors to estimate the uncertainty of the calculated velocity $\mathbf{V}$, arising from the uncertainties in the particle positions, for any given algorithm. Choosing the algorithm used in the simple two-frame tracking method, Eq. (1), the uncertainty in the calculated velocity $\mathbf{V}$ is

$$(\delta\mathbf{V})^2 = \frac{(\delta\mathbf{x}_{j+1})^2 + (\delta\mathbf{x}_j)^2}{(\Delta t)^2} = 2(\delta\mathbf{x}/\Delta t)^2. \tag{2}$$

Importantly, this source of error becomes larger, not smaller, as the time interval between measurements Δt is decreased. This is seen in Eq. (2), where the denominator diminishes with an increasing frame rate but the numerator does not vary with frame rate at all. Thus, this error in velocity is $\propto 1/\Delta t$. Now that high frame-rate scientific cameras have become more easily available, this error in PTV probably occurs more commonly.

To illustrate the combined effect of both types of errors, those due to acceleration and due to uncertainties in particle position, consider the sketch in Fig. 1(b) for a particle undergoing uniform circular motion. The uncertainty in particle positions is indicated in Fig. 1(b) by shading around the true positions. Uncertainties in particle position cause the measurements to fall on the vertices of a polygon that is deformed, as compared to Fig. 1(a) without particle-position uncertainties. Here we are interested primarily in errors in the velocity V. In Fig. 1(b), the length of the edges of the distorted polygon is an indication of velocity. Comparing this thick irregular polygon to the original ideal circle, it is obvious that the velocity vector will have errors in its direction. The magnitude of the velocity will also have errors, as we discuss next.

C. Illustration of PTV velocity errors for a particle in uniform circular motion

To demonstrate the combined effect of both sources of error in velocity, we present a simple simulation. A single particle is assumed to perform uniform circular motion with a radius R and period T. A time series of a particle's x and y coordinates is recorded at intervals Δt. The simulation duration is 1000 circular periods. To simulate a measurement error, we add a random error chosen from a Gaussian distribution to the true position. The probability of an error x_{err} in this Gaussian distribution is $\propto \exp[-x_{\text{err}}^2/(2\delta_x^2)]$, and the same for the y direction. We then calculate a time series $\mathbf{V}(t)$, for the particle velocity using the simple two-frame tracking method, Eq. (1). We use $\mathbf{V}(t)$ to calculate a time series for the kinetic energy, $\text{KE}(t) = m|\mathbf{V}(t)|^2/2$, where the mass m will cancel in our final results. We average over the entire time series yielding $\langle\text{KE}\rangle$, which we compare to the true kinetic energy KE_{true} in Fig. 2. We indicate the discrepancy between $\langle\text{KE}\rangle$ and KE_{true} as a single data point in Fig. 2. We vary the size of the position error δ_x and the sampling interval Δt, yielding all the data points shown in Fig. 2.

Figure 2 shows the total error in $\langle\text{KE}\rangle$ due to both sources combined. For $\delta_x = 0$, Fig. 2 also shows the error due to acceleration only, as indicated by circles. This error
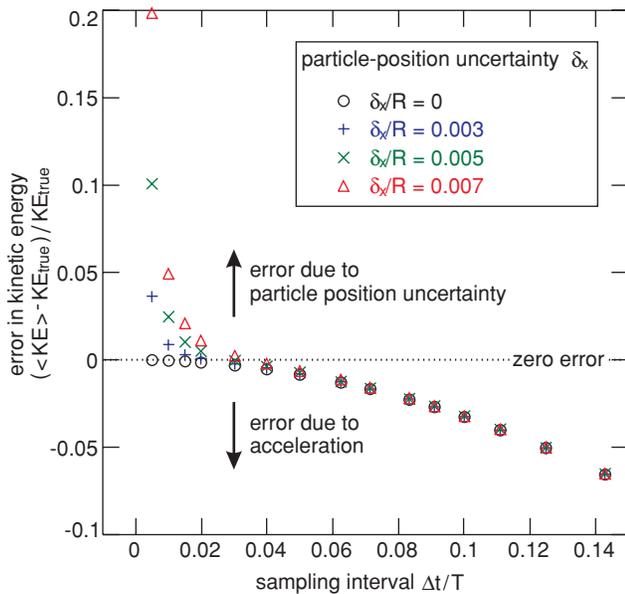
FIG. 2. (Color online) Velocity errors for a simulated particle executing uniform circular motion with period T, as in Fig. 1(b). The velocity is calculated using Eq. (1), and then the kinetic energy is calculated and averaged over 1000 cycles of the circular motion. The true kinetic energy is $m(2\pi R/T)^2/2$. The severity of the two sources of error in the kinetic energy depends on the sampling interval Δt. The dominant error arises at small Δt from particle-position uncertainty and at large Δt from acceleration. There is a nearly flat spot at $\Delta t/T \approx 0.3$, where the total error from two sources is smallest.

due to acceleration is always negative, and it is most severe for large Δt, i.e., for a slow frame rate. The total error in $\langle KE \rangle$, however, can be either positive or negative, as shown by the other data points in Fig. 2. The total error is positive at small Δt mainly due to the contribution of particle-position uncertainty, and it becomes negative at large Δt mainly due to acceleration. In between, the total error in $\langle KE \rangle$ has its smallest magnitude, which for this simulation occurs at a sampling interval of about 3% of the circular period, i.e., $\Delta t/T \approx 0.03$. This observation suggests that it may be possible to choose a best frame rate to minimize errors, as we discuss next.

In an actual experiment, one cannot independently measure the two sources of error in velocity or in quantities, such as $\langle KE \rangle$, that are calculated from the velocity. In most physical systems, the true motion of one particle will differ from that of another particle, and their accelerations will vary with time, unlike the idealized circular motion simulation in Fig. 2. Nevertheless, examining Fig. 2 suggests a possible scheme for choosing a Δt that has some promise to reduce the total error.

D. Scheme for reducing total error

The scheme we suggest for reducing the total error for a quantity (such as $\langle KE \rangle$) computed from velocities is as follows. The experimenter can record motion at a high frame rate, and then analyze the data not only at the frame rate, but also at larger Δt by skipping one frame to double Δt, two frames to triple Δt, and so on. Plotting the average of KE vs. sampling interval Δt will yield a graph similar to Fig. 2. If the graph exhibits a nearly *flat spot* at a particular value of Δt, that value is a candidate to consider as the best value

for computing KE. In Sec. III B, we will examine actual experimental data as a demonstration.

III. PTV VELOCITY ERRORS IN EXPERIMENTS

A. Experiment

Various experiments using PTV can have much in common, even when the physical systems being studied are completely different. Particle positions are measured in images that correspond to individual video frames, and to do this they must be spaced sufficiently that they can be distinguished. In a granular media experiment, the particles might fill about half the volume,[8] while in a dusty plasma experiment they are more widely spaced, filling typically one millionth of the total volume. There are several choices for image analysis methods to measure the particle position, which have various advantages for different physical systems; for our dusty plasma we use the moment method of Ref. 1. What might differ the most, from one physical system to another, is the nature of the true particle motion. In granular media, particles interact as hard spheres, with large accelerations during the brief collision events. Dust tracers in fluid mechanics experiments represent the opposite extreme: they do not collide at all, but are merely swept along with the flow of the fluid. Between these two extremes is dusty plasma with particles that collide softly with a screened Coulomb repulsion, so that they undergo a gradually changing acceleration. For recording videos of physical systems where the particles do collide, it is useful to characterize a typical time scale. A typical time scale is $\approx 10^{-2}$ s between hard-sphere collisions in granular flow.[8] The duration of a Coulomb collision in dusty plasma is $\approx 10^{-1}$ s.[19]

We carry out a demonstration experiment[20] using PTV. Polymer microspheres (which we will refer to as dust particles) of 8.1-μm diameter are immersed in a partially ionized argon gas under vacuum conditions. This four-component mixture (micron-size dust particles, rarefied gas atoms, electrons, and positive ions) is called a dusty plasma. The dust particles are electrically charged, and they are levitated against the downward force of gravity by a vertical electric field in the plasma, Figs. 3(a) and 3(b). Due to their mutual Coulomb repulsion, the dust particles tend to arrange themselves so that they are separated by a distance $b \approx 0.7$ mm, which is much larger than their diameter. Using a laser heating method,[15, 20] the kinetic energy of dust particles is augmented, so that their velocities fluctuate with a typical magnitude of ≈ 1 mm/s, as we will find in Sec. III B. The dust particles always remain within a single horizontal layer, which is well suited for imaging.

This horizontal layer of dust particles is imaged by a high-speed camera viewing from above. We use a 12-bit Phantom v5.2 camera, with 1152×720 pixel resolution, and a Nikon 105 mm focal-length macro lens. The dust particles are illuminated by a horizontal sheet of 576-nm laser light, and the camera lens is fitted with a filter to block unwanted light at other wavelengths. The camera 36.2×22.6 mm² field-of-view (FOV) includes $N \approx 2100$ dust particles. A video is recorded as a series of bit-map images. The image
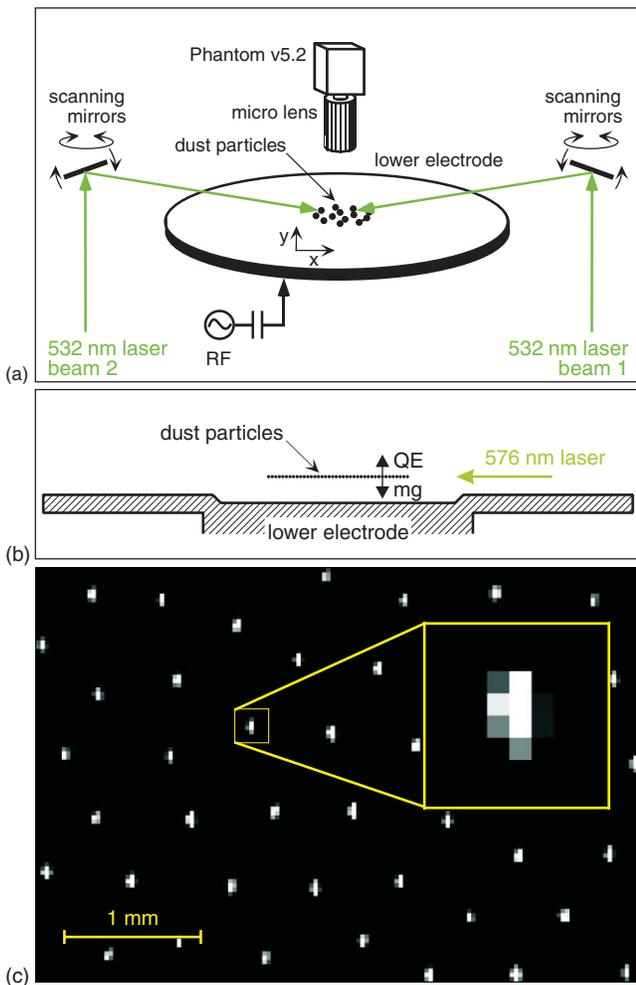
FIG. 3. (Color online) Configuration for the experiment. (a) Dust particles were levitated in a horizontal layer above a lower electrode. A vacuum chamber, not shown here, was filled by a weakly ionized plasma, consisting of electrons, positive argon ions, and a neutral argon gas. A pair of 532-nm laser beams accelerated the dust particles to raise the kinetic temperature. A top-view camera imaged the dust particles, which were illuminated by a horizontal sheet of 576-nm laser light. (b) Dust particles are electrically levitated in a horizontal layer. (c) Experimental bitmap image from one frame of a video. Only $\approx 1/60$ of the experimental image is shown. The inset is a magnified image of one dust particle.

of one dust particle fills about ten pixels, see Fig. 3(c). Due to the large interparticle spacing, only about 2% of the pixels in an image is brighter than the background. The camera recorded a 20-s movie at 250 frames/s, i.e., a time between frames of 4 ms.

We can compare our 4-ms time interval between frames to *two physical time scales* in the experiment, which are both of interest in determining the velocity error. One physical time scale is for acceleration for a Coulomb collision among dust particles. This is typically $\approx 10^{-1}$ s, based on time required for the dust-particle velocity to change significantly in the binary-collision experiment of Konopka *et al.*[19] The other physical time scale (for errors arising from particle-position uncertainty) is the time required for a dust particle to move one pixel. A thermal velocity of ≈ 1 mm/s is a typical velocity in our experiment, and at this speed a dust particle would require ≈ 30 ms to cross one pixel.

Images for each frame in the movie are analyzed using the method of Ref. 1 to measure the dust particle positions. We compute velocities using the simple two-frame tracking method, Eq. (1). We then calculate the kinetic temperature and a correlation function, as we describe next.

B. Kinetic temperature

We will demonstrate here, using experimental data, that PTV does not yield a unique value for the kinetic temperature. Instead, it yields a value that depends on the sampling interval Δt between images that are analyzed. All the data presented below come from the same experiment with the camera always operated at the same frame rate. To demonstrate the dependence on the sampling interval, we will repeat our analysis by skipping frames.

The quantity that we test here is the kinetic temperature, averaged over a time series. The time series for kinetic temperature is calculated from mean-square velocity fluctuations

$$T(t) = \frac{1}{Nk_B}\left[\sum_{i=1}^{N} \frac{m}{2}\left|\mathbf{V}_i(t) - \overline{\mathbf{V}}(t)\right|^2\right], \quad (3)$$

where N is the number of dust particles analyzed. (Alternatively, one could calculate the temperature as a fit parameter assuming a Maxwellian velocity distribution.[13]) We then average over the entire time series, yielding the calculated result for the kinetic temperature $\langle T \rangle$. In Eq. (3), $\overline{\mathbf{V}}(t)$ indicates a velocity averaged over all dust particles in one frame, i.e., the center-of-mass velocity. (The bar indicates an average over all dust particles in the FOV in one image, while the brackets $\langle \cdots \rangle$ indicate averages over time.)

The results for the calculated kinetic temperature, as the sampling time interval Δt is varied, are shown in Fig. 4. We see that $\langle T \rangle$ decreases gradually as the sampling time interval increases. The error in kinetic temperature is the difference of $\langle T \rangle$, which varies with Δt, and an unknown true value, which does not vary. Thus, it is clear that PTV does not yield a unique value for the kinetic temperature, but instead a value with an error that depends on the experimenter choice of Δt. Moreover, the error is not necessarily reduced by choosing a faster frame rate and using every frame without skipping frames; if one did this, the result for $\langle T \rangle$ would become steadily larger as the frame rate is increased.

Having observed experimentally that there must be an error in $\langle T \rangle$ that depends on the experimenter choice of Δt, we ask how the experimenter should choose Δt. Motivated by Sec. II D, we will use our flat-spot scheme with our graph of $\langle T \rangle$ vs. Δt, see Fig. 4.

Examining Fig. 4, we see a general trend of $\langle T \rangle$ diminishing with the sampling time interval Δt. For the smallest Δt shown, the curve has a steep slope, while for larger Δt there is what appears to be a nearly flat spot. Comparing to Fig. 2 for ideal circular motion, we interpret the steep slope at small Δt as an indication that the error is dominated by particle-position uncertainties, which is the source of error that becomes worse with higher frame rates. In the nearly flat spot of Fig. 4 for our experiment, we would choose $\Delta t \approx 0.03$ s,
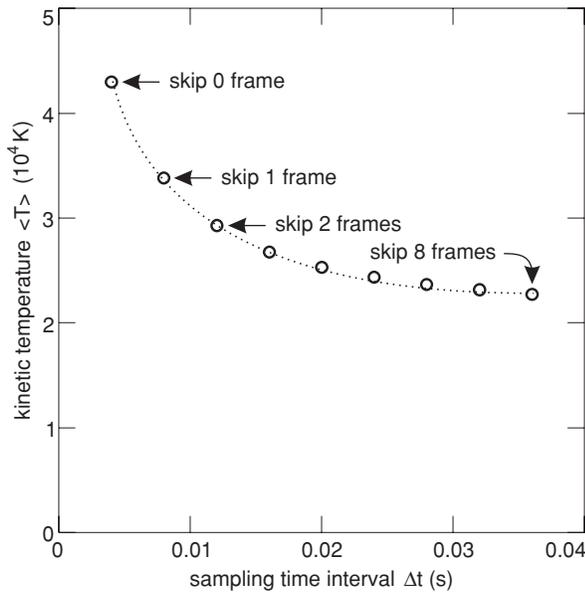
FIG. 4. Experimental data for the kinetic temperature, calculated using Eq. (3). To vary the sampling time interval Δt, we skipped frames in the data analysis. We note the appearance of a nearly flat spot, similar to the one seen in Fig. 2 for the circular-motion simulation. In Sec. III B we suggest that for calculating kinetic temperature, it is desirable either to choose Δt in the flat spot of this graph or to choose Δt comparable to the time required for a particle at the thermal velocity to move a distance of one pixel. Both of these schemes lead to the same choice in this experiment, $\Delta t \approx 0.03$ s.

corresponding to $\langle T \rangle \approx 2.5 \times 10^4$ K and thermal velocity is 1.3 mm/s.

A limitation of this scheme for choosing Δt is that the dust particles in the experiment do not undergo the same acceleration all the time, unlike the simple circular-motion that is simulated in Sec. II C. Therefore, there is no strong reason to expect that the nearly flat spot in the graph of $\langle T \rangle$ vs. Δt will be the same as in Fig. 2 for the circular-motion simulation. Nevertheless, this seems to be the best guidance available to us, given what we know about the errors in **V** and therefore $\langle T \rangle$ that arise from particle-position uncertainty and acceleration.

In addition to the flat-spot scheme described above, we can suggest an alternate scheme which leads to the same choice of Δt in the case of our experiment. As described in Sec. III A, there are two physical time scales that can be compared to Δt. As *an upper limit* for Δt, to avoid excessive velocity errors arising from acceleration, it is desirable to choose Δt significantly smaller than the time scale for velocity to change significantly in the physical system. As *a lower limit* for Δt, to avoid large errors arising from particle-position uncertainty, Δt should be comparable to, or larger than, the time required for a particle at a typical velocity (in our experiment the thermal velocity) to move one pixel. Our reasoning behind the lower limit is that the particle-position uncertainty is generally a certain fraction of one pixel,[1,13] so that if the displacement of a particle in Δt is much smaller than one pixel a large velocity error arising from particle-position uncertainty should be expected. For this experiment, the upper limit due to acceleration is about 100 ms, while the lower limit due to particle-position uncertainty is about 30 ms.

Choosing Δt significantly smaller than the upper limit, and comparable to the lower limit, leads us to choose the same value as the flat-spot scheme, $\Delta t \approx 0.03$ s in our experiment.

We note that another approach to reduce the effect of particle-position uncertainty was reported by Knapek *et al.*[21] In a dusty plasma experiment similar to ours, they operated a camera at 500 frames/s. Before carrying out computations to track the particles and compute the kinetic temperature, they superimposed three consecutive bit-map images, averaging them pixel by pixel. This averaging of images has the effect of reducing particle-position uncertainty with the trade-off of a reduction of time resolution. In our scheme, instead of averaging information while it is still in the form of bit-map images, we focus our efforts on the particle-tracking algorithm and choosing the time interval.

C. Transverse current autocorrelation function

Aside from the kinetic energy and related quantities, such as the kinetic temperature, there are other quantities one might wish to compute from velocity data produced by PTV. For a correlation function computed from experimental velocity and position data, we will demonstrate here that artifacts can arise at high frame rates that are different from the errors that appear in calculations of the kinetic temperature. We will also demonstrate that reducing these artifacts can require a scheme different from the one described in Sec. II D for kinetic temperature.

The correlation function, we will choose, is the transverse current autocorrelation function $C_T(t)$, which is used in the study of shear motion and viscoelasticity.[20,22,23] This correlation function is computed from records of particle positions $x_i(t)$ and velocities $v_{y,i}(t)$ in orthogonal directions,

$$C_T(t) = \langle j_y^*(k, 0)\, j_y(k, t) \rangle / \langle j_y^*(k, 0)\, j_y(k, 0) \rangle, \qquad (4)$$

where

$$j_y(k, t) = \sum_{i=1}^{N} v_{y,i}(t) \exp[ikx_i(t)]. \qquad (5)$$

The quantity $j_y(k, t)$ is called a "transverse current," even though it has nothing to do with electric current. Both the current and the correlation function $C_T(t)$ depend on a parameter k that is adjustable. In the study of viscoelasticity,[20] values of k comparable to $3/b$ are typical, where b is a mean interparticle distance. (In our experiment, $b \approx 0.7$ mm.) Like many autocorrelation functions, $C_T(t)$ has an initial decay at small time t; this initial decay is of great interest in the study of viscoelasticity.[20]

Using particle velocity and position data from our experiment, we find an undesired artifact in the initial decay of the calculated $C_T(t)$, which can be seen in Fig. 5(a). This artifact is most prominent when the camera frame rate is high and when we use the simple two-frame tracking method to prepare the velocity data used in calculating $C_T(t)$. We attribute this artifact to particle-position uncertainty.

Although this artifact arises due to a high camera frame rate, it is unattractive to attempt to eliminate it by reducing the frame rate. Many data points during the initial decay are
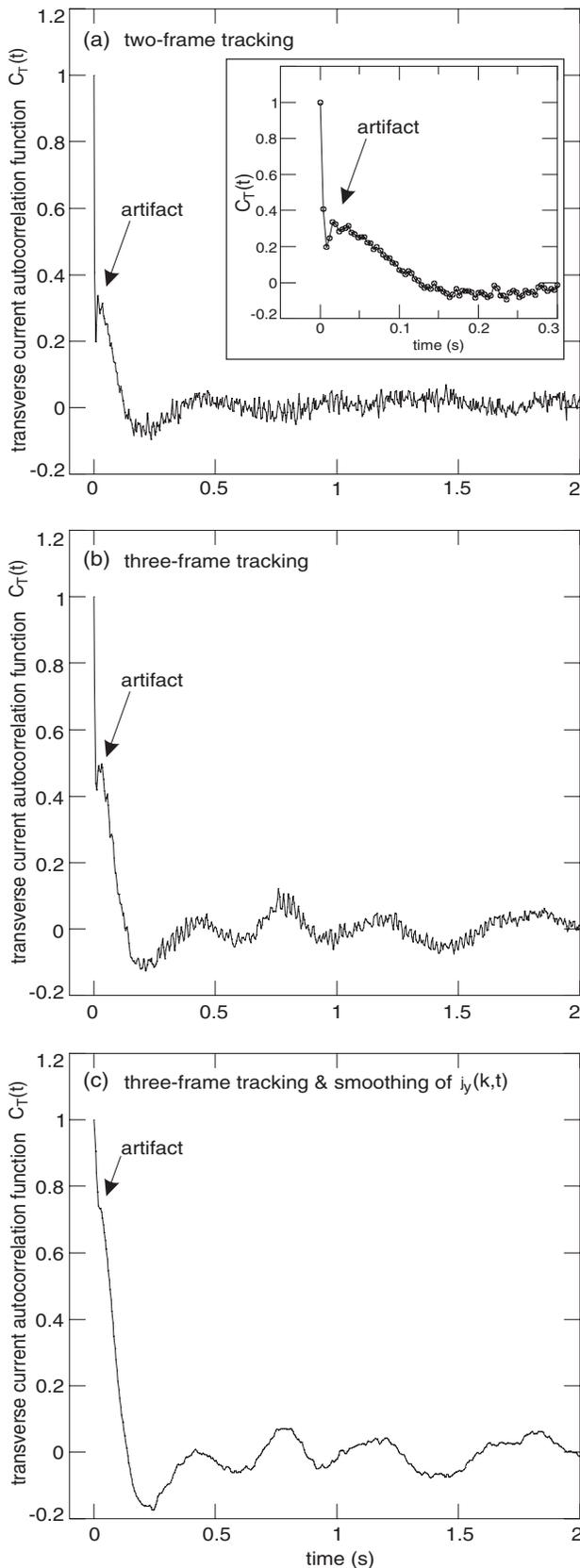
FIG. 5. Experimental data for $C_T(t)$, an autocorrelation function of the time series $j_y(k, t)$, as defined in Eq. (5). An unwanted artifact appears in the initial decay in (a) with two-frame tracking, i.e., using Eq. (1) to calculate velocities. This artifact is reduced in (b) by using three-frame tracking, i.e., using Eqs. (6) and (7). It is further reduced in (c) by smoothing the the time series $j_y(k, t)$ before calculating its autocorrelation. Data shown here were computed for $kb = 3.5$, where b is the mean interparticle distance.

required for the study of viscoelasticity, and this requires a high frame rate. Thus, we need a different approach to reduce the artifact, such as the two-step approach we describe next, for reducing the effects of particle-position uncertainty.

In the first step, we use a three-frame tracking method, which reduces the artifact as seen in Fig. 5(b). In this three-frame tracking, we do not entirely skip any frames, because that would reduce the temporal resolution of $C_T(t)$. Instead, we interlace pairs of frames and calculate the positions and velocities as

$$\mathbf{X}_j = (\mathbf{x}_{j-1} + \mathbf{x}_j + \mathbf{x}_{j+1})/3, \qquad (6)$$

$$\mathbf{V}_j = (\mathbf{x}_{j+1} - \mathbf{x}_{j-1})/2\Delta t. \qquad (7)$$

Here, x_{j-1}, x_j, x_{j+1} are the positions of a particle in any three consecutive frames. Equations (6) and (7) can be derived from the linear regression for three data points. Using three frames rather than two requires a sufficiently fast frame rate, which is satisfied in our experiment with a time interval between frames of 4ms, which is much shorter than the time scale for changes in the dust particle's velocity $\approx 10^{-1}$ s. (While this three-frame tracking method has the advantage of reducing the unwanted artifact in the initial decay of $C_T(t)$, it may also have an effect on the long-time oscillations of $C_T(t)$, which we have not studied.)

To further reduce the artifact, in the second step, we smooth $j_y(k, t)$ before calculating $C_T(t)$. This averaging requires a sufficiently high frame rate. We smooth the transverse current $j_y(k, t)$ with a moving average over five consecutive frames before calculating $C_T(t)$. The result for $C_T(t)$, Fig. 5(c), is a great reduction of the artifact without an unnecessary loss of time resolution in the initial decay.

To verify that these two steps reliably remove artifacts without changing the true shape of the initial decay, we carried out numerical simulations, where the true shape is known. These simulations used a Langevin molecular dynamic method[20] to track particles by integrating their equations of motion. To mimic the experiment particle-position uncertainty, we added random errors to the positions in the simulation as in Sec. II C. The simulation results, not shown here, confirm that the artifact is diminished by the two steps described above, and they also confirm that this improvement results in only a negligible change in the initial decay of the calculated $C_T(t)$, as compared to the true $C_T(t)$ without the random errors.

IV. SUMMARY

In summary, we have studied two sources of errors in PTV using high-speed cameras. What may surprise some users is that a higher frame rate is not necessarily better, and in fact it can greatly worsen velocity errors arising from particle-position uncertainty. We described our solutions to reduce these errors: two schemes that lead to the same choice for Δt when calculating kinetic temperature and a two-step approach for computing an autocorrelation function. We demonstrated these solutions using the data from a dusty plasma experiment.

ACKNOWLEDGMENTS

[1] Y. Feng, J. Goree, and B. Liu, Rev. Sci. Instrum. **78**, 053704 (2007).

[2] V. Hadziavdic, F. Melandsø, and A. Hanssen, Phys. Plasmas **13**, 053504 (2006).

[3] B. Lüthi, A. Tsinober, and W. Kinzelbach, J. Fluid Mech. **528**, 87 (2005).

[4] N. P. Oxtoby, J. F. Ralph, D. Samsonov, and C. Durniak, Proc. SPIE **7698**, 76980C (2010).

[5] S. Nunomura, D. Samsonov, S. Zhdanov, and G. Morfill, Phys. Rev. Lett. **95**, 025003 (2005).

[6] Y. Feng, J. Goree, and B. Liu, Phys. Rev. Lett. **100**, 205007 (2008).

[7] R. Lima, T. Ishikawa, Y. Imai, M. Takeda, S. Wada, and T. Yamaguchi, Ann. Biomed. Eng. **37**, 1546 (2009).

[8] N. Jain, J. M. Ottino, and R. M. Lueptow, Phys. Fluids **14**, 572 (2002).

[9] A. Melzer, A. Homann, and A. Piel, Phys. Rev. E **53**, 2757 (1996).

[10] V. Nosenko and J. Goree, Phys. Rev. Lett. **93**, 155004 (2004).

[11] Y. Ivanov and A. Melzer, Phys. Plasmas **12**, 072110 (2005).

[12] V. E. Fortov, O. S. Vaulina, O. F. Petrov, M. N. Vasiliev, A. V. Gavrikov, I. A. Shakova, N. A. Vorona, Y. V. Khrustalyov, A. A. Manohin, and A. V. Chernyshev, Phys. Rev. E **75**, 026403 (2007).

[13] C. Knapek, Ph.D. dissertation, Max-Planck-Institut für extraterrestrische Physik, Garching and Ludwig-Maximilians-Universität, München, Germany, 2010.

[14] V. Dore, M. Moroni, M. L. Menach, and A. Cenedese, Exp. Fluids **47**, 811 (2009).

[15] V. Nosenko, J. Goree, and A. Piel, Phys. Plasmas **13**, 032106 (2006).

[16] O. S. Vaulina, X. G. Adamovich, O. F. Petrov, and V. E. Fortov, Phys. Rev. E **77**, 066404 (2008).

[17] J. Westerweel, Meas. Sci. Technol. **8**, 1379 (1997).

[18] J. D. Williams and E. Thomas, Jr., Phys. Plasmas **13**, 063509 (2006).

[19] U. Konopka, G. E. Morfill, and L. Ratke, Phys. Rev. Lett. **84**, 891 (2000).

[20] Y. Feng, J. Goree, and B. Liu, Phys. Rev. Lett. **105**, 025002 (2010).

[21] C. A. Knapek, D. Samsonov, S. Zhdanov, U. Konopka, and G. E. Morfill, Phys. Rev. Lett. **98**, 015004 (2007).

[22] U. Balucani, J. P. Brodholt, P. Jedlovszky, and R. Vallauri, Phys. Rev. E **62**, 2971 (2000).

[23] Z. Hu and C. J. Margulis, J. Phys. Chem. B **111**, 4705 (2007).