DENSE PARTICLE IDENTIFICATION AND RECONSTRUCTION (DPIR)

APPLICATION NOTE V3V-FLEX-011 (A4)

Contents

Contents 1
Introduction 1
Dense Particle Identification and Reconstruction—Description of the Technique 2
Particle Identification 3
  Peaks 4
  Projections 4
  Paths 5
Comparison with Previous Techniques 6
Results 7
  Synthetic Data 7
  Real Data 8
Conclusion 8
References 9

Introduction

Dense Particle Identification and Reconstruction (DPIR) is a three-dimensional Particle Tracking Velocimetry (PTV) technique that utilizes a three-pronged approach to identifying and reconstructing the positions and velocities of densely-packed particles within a volumetric domain imaged onto camera sensors for the purpose of understanding and analyzing fluid motion (Boomsma and Troolin, 2018). 3D PTV techniques offer higher accuracy (Kahler et al., 2012) and higher spatial resolution compared to volumetric correlation techniques. In contrast to correlation techniques, 3D PTV reconstructing the position of tracer particles in 3D space so that the particles can be tracked over time and the velocity field extracted.

Particle reconstruction is made difficult by the fact that individual particle locations imaged on the camera sensor in 2D must be projected into 3D space very accurately. Unlike traditional planar or StereoPIV where the out-of-plane (z) location is determined by the light-sheet illumination, volumetric techniques illuminate a volume so that the depth position of a particle cannot be assumed. In fact, very small errors in determining the particle position on the sensor is magnified when projected into the measurement volume. Further, as particle density increases, the likelihood of particle overlap also increases and overlapping particles make locating particle centers more difficult.

For these reasons, accurately determining the positions of particles on the 2D images is important to achieve an accurate 3D particle location. Therefore, to improve 2D particle identification, DPIR determines the *number of particles* per support set (i.e., a blob) *prior* to the final particle fitting. DPIR relies on well-established methods first used in computer vision (e.g. Hartley and Zisserman, 2003), with additions to adapt to the unique challenges posed by volumetric velocimetry. DPIR utilizes Peaks, Projections, and Paths to determine the number of particles per support set. Each of these techniques are discussed in detail in the following sections.

Dense Particle Identification and Reconstruction—Description of the Technique

Three-dimensional PTV can be broken down into the following steps—each depending upon the prior step:

1. Calibration: relating points in 3D to 2D points on each camera's image.
2. Particle Identification: locating 2D particles on each camera's image.
3. Particle Matching: matching 2D particles among cameras and triangulation in 3D space.
4. Particle Tracking: tracking of 3D particles in time to calculate particle velocities.
5. Path Fitting: linking consecutive 3D particles in time and fitting with a polynomial; calculating useful quantities from position and velocity (e.g. acceleration, vorticity, pressure, etc.).

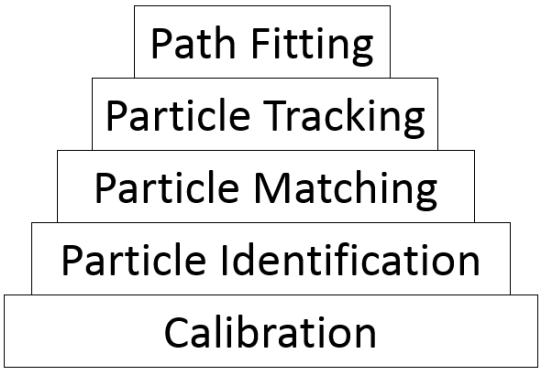

Figure 1. DPIR Processing Pyramid.

DPIR utilizes these steps iteratively to maximize the accuracy of the 2D particle positions, thereby increasing the accuracy of the particle matching and tracking steps. We see then that 2D particle finding is the heart of DPIR.

Particle Identification

Particle Identification is initiated by determining support sets—or a contiguous group of pixels that contain pixels with intensity levels above the noise. Each support set can contain one or more particle images. Recall that in 3D techniques, a volume is illuminated; therefore, overlapping particles not only exist, but are exceedingly common for densely seeded flows such as those analyzed by DPIR. So a support set typically consists of a group of particles that are adjacent to each other on the 2D image, but are not necessarily close to each other in the real-world measurement volume. Figure 2 shows a portion of an image (left), and its corresponding support sets (right).

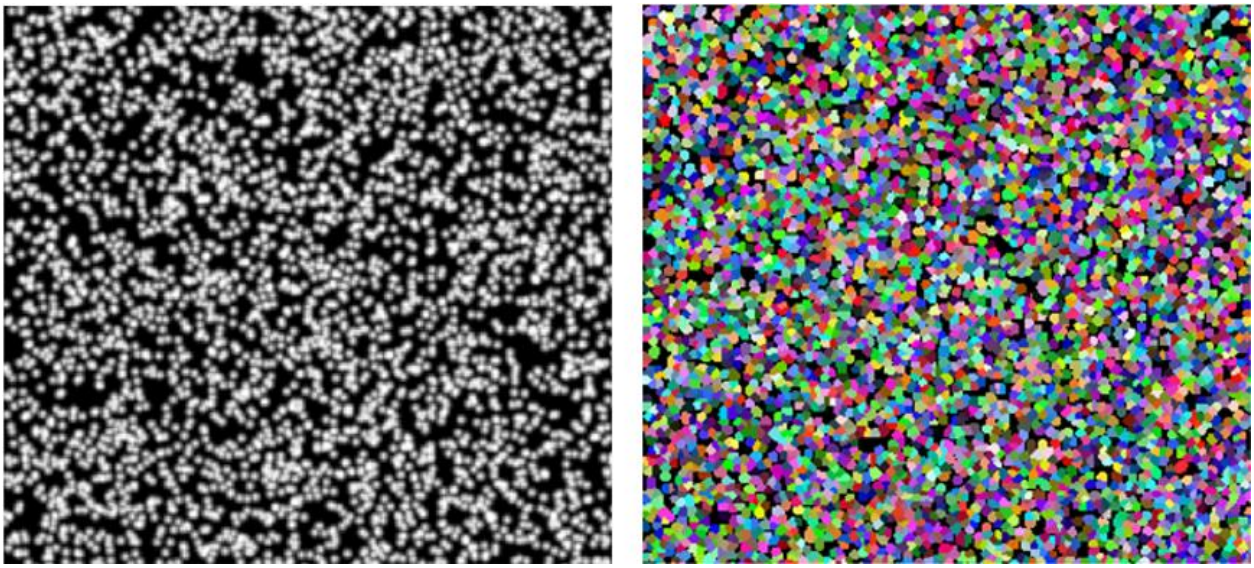

Figure 2. A portion of an image with ppp = 0.1, and its corresponding support sets (coloring is random so as to indicate separate support sets).

After a support set within the image is identified, the number of particles within the support set is determined (described in detail in the next section) and when completed, a nonlinear solver is used to fit the correct number of particles to the intensity profile of the support set. Obviously, the fit is much more accurate once the correct number of particles is known.

Fitting multiple particle images per support set was first documented by Lei et al. (2010). Multiple particle fitting has a distinct advantage over basic peak fitting, particularly in cases of high seeding densities, in that particle images can still be identified when the overlap ratio exceeds 50%. In fact, multiple particle fitting can fit an overlap ratio of 100%, as long as it is known that more than one particle resides within the support set.

Consider Figure 3, which illustrates a 2D representation of an intensity profile of a support set, shown on the far left. If a single particle is fit to the support set, the result would be an incorrect particle position as shown in the middle plot. Once the correct number of particles that make up a support set are known, a much more accurate fit can be performed, illustrated in the figure on the right.

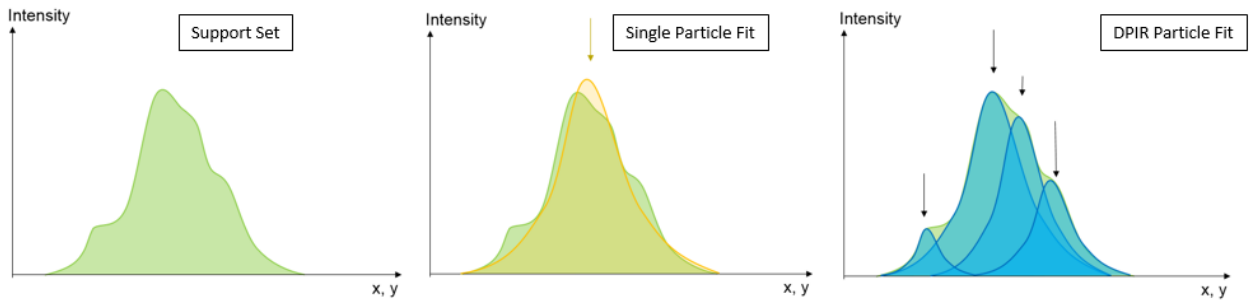

Figure 3. 2D representation of the particle fitting routine. On the left, a support set intensity profile is shown. In the middle, a single particle fit is shown, and on the right, a multiple particle fit is shown.

Peaks

Peaks are simply the local maximum of intensity on an image. They are the most easily identified, but least trustworthy source of a particle image. The primary reason for distrust is due to camera resolution—in terms of pixel size and number, but also bit depth. The camera sensor effectively discretizes what is in reality a continuous variation of intensity. This is why sub-pixel fitting (e.g., a Gaussian profile fit) is required. When the overlap ratio exceeds 50% for two particle images, the peak found on the image is no longer sufficient to indicate two particle images—rather it resembles just one particle image.

In any case, DPIR utilizes peaks as a source for particle images through each recursion—but particularly for the first recursion, as it is used to initiate the process.

Projections

Once the peaks within each of the images are determined and corresponding particle images are fit, those 2D particles are matched among cameras and then reconstructed in 3D space. Reconstruction (or triangulation) uses a mapping function created during the calibration process. If peak projections from multiple views intersect within a given tolerance, a 3D particle has been found. Considering this, there may be instances where a 3D particle is matched and triangulated using fewer than the total number of available sensors. For example, consider the case shown in Figure 4. A 3D particle ($P3D_0$) has been matched in sensors S0 ($P2D_0$), S1 ($P2D_1$), and S2 ($P2D_2$), but not in S3. In this case, the *inverse mapping function* can be used to project the location of the 3D particle ($P3D_0$) back into sensor S3. This would then yield the location of an *Expected Particle Image* (EPI). This EPI_3 is located within a support set in image sensor S3, and can therefore be added to the number of particles within that support set. On the next iteration, the number of particles within that support set will be increased to include any EPI's that were found in the *projections* step.

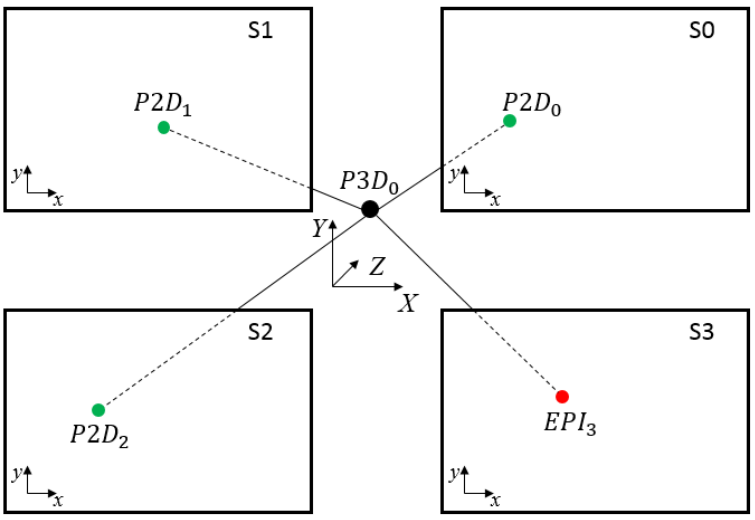

Figure 4. A particle matched from three sensors (but not in the fourth) is projected onto the fourth image as an expected particle image (EPI).

Paths

In addition to peaks and projections, particle paths can also be used to determine that a particle resides within a given support set. DPIR uses 2D particle trajectories built from multiple particle images (either forward or backward in time) in order to recover 'missing' particles. The key to the success of the technique is that the particle location, shape, and intensity are not assumed ahead of time (e.g. through the use of an optical transfer function which requires a calibration), but rather from the *real particle images* themselves. Consider Figure 5. In the *paths* step, a 2D particle trajectory (shown in green) is determined based on the real particle images in previous (or future) steps. The particle trajectory fit (e.g. polynomial, spline, or etc…) uses the intensity, size, shape, and position to predict what the particle will look like (and where it exists) at the time-step of interest (red dot).

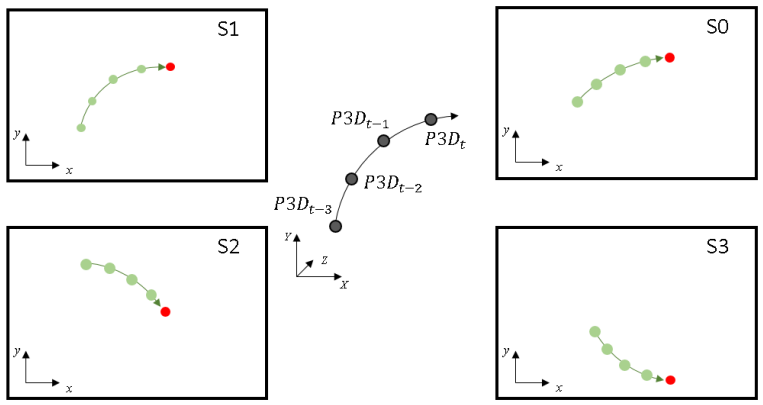

Figure 5. 2D particle trajectories can be used to determine the size, shape, intensity, and position of particles within support sets.

It is important to note that there are several distinct advantages to this approach. The first is that a particle that exists within a support set may not be distinguishable from neighboring or occluding particles, but, by using the trajectory the location of a particle can be determined *even if it is completely overlapped by another particle*. Additionally, since the size, shape, intensity and position of *each particle* is known both before and after the time-step under consideration, its characteristics are well-known, increasing the accuracy of the 2D particle intensity fit. Even in cases of non-uniform illumination (e.g. near the edge of the illuminated measurement volume where the intensity is typically less), imagine a particle approaching the edge of the illumination volume, steadily becoming less and less intense (and possibly smaller, among other things). These sorts of gradients in particle characteristics can be taken into account and used in the particle prediction.

Dense Particle Identification and Reconstruction is an iterative technique, meaning that once the peaks, projections, and paths have been used to more accurately determine the number of particles per support set, the multiple particle fitting is performed again, giving a more accurate initialization as the steps are iterated. The process repeats until all possible particles have been identified and tracked (Figure 6).

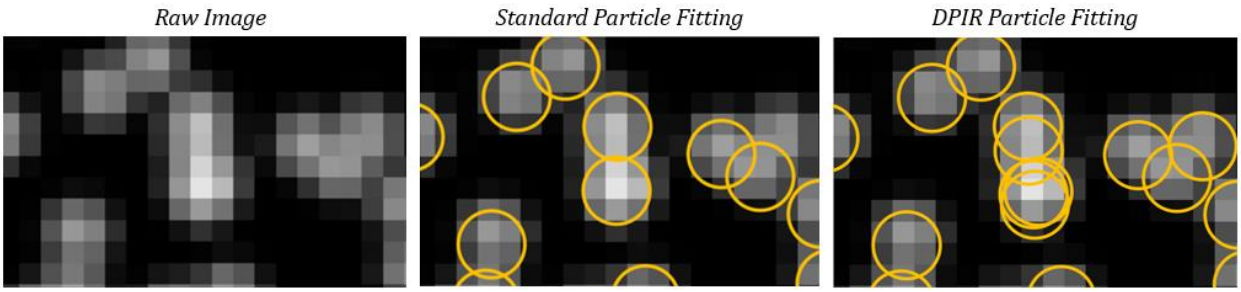

Figure 6. A portion of a particle image (left), the image after standard particle fitting (center), and the image after Dense Particle Identification and Reconstruction particle fitting (right).

Comparison with Previous Techniques

Dense Particle Identification and Reconstruction was compared with the results from the work of Weineke (2010).

Figure 7 shows the fraction of ghost particles as the seeding density is increased (solid lines). At a seeding density of 0.1 ppp, DPIR has a fraction of ghost particles near 11%, and Weineke (2010) has a fraction of ghost particles near 60%.

In addition, Figure 7 also shows the reconstruction error as a function of seeding density (dashed lines). At a seeding density of 0.1, DPIR has a reconstruction error near 0.32 pixels, while Weineke (2010) exhibits a reconstruction error of 0.6 pixels.

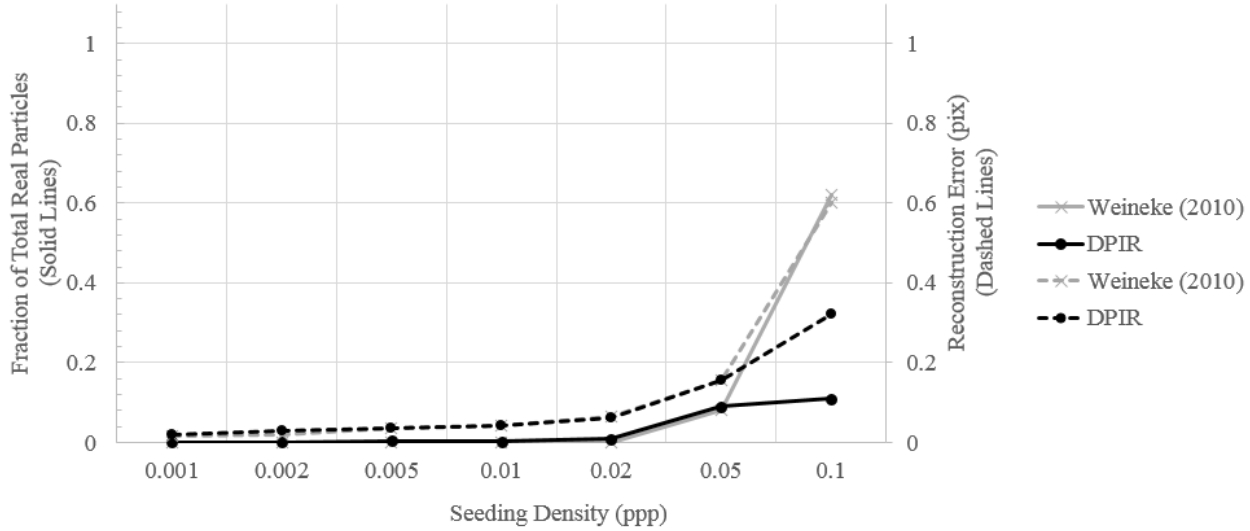

Figure 7. Fraction of ghost particles vs seeding density, and reconstruction error (in pixels) vs seeding density. Data source is synthetic particle images with 3 pixel diameter.

Results

Synthetic Data

DPIR was utilized in analyzing data from the John's Hopkins Turbulent Database for a realistic, synthetic turbulent flow. Details concerning their simulation can be found in Li et al. (2008), but pertinent to the current study, the bulk velocity was about 0.75 m/s. The dt=3.3 ms, or a repetition rate of about 150 Hz, which equated to a particle image displacement of about 15 pix. The turbulent velocity field imposed on the synthetic 3D particles was obviously three-dimensional in nature and ranged from 0–0.9 m/s in the primary flow direction. In the wall-normal and transverse directions, velocity magnitudes had a min/max of approximately 20% of the primary velocity component.

Figure 8 highlights instantaneous contours of the primary component of normalized velocity with h being the half-channel height in the simulation.

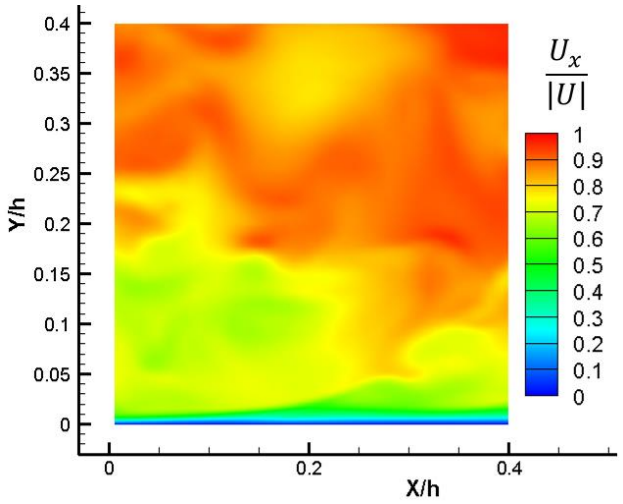

Figure 8. Instantaneous contours of the primary component of normalized velocity.

Table 1 shows the mean and standard deviation of errors (in pixels) for the predicted image space values, using a 2nd order polynomial fit to four prior consecutive image space locations for each particle image. Prediction errors are lowest for the uniform case, which is not surprising, but they are still very low for even the turbulent flow case, with all errors falling at or below 0.0022 pixels.

Table 1. Mean and standard deviation (units of pixels) values for both sub-cases for each seeding density case.

	Prediction Error (Mean, Std. Dev.), in pixels					
Sub-Case	**ppp = 0.02**		**ppp = 0.05**		**ppp = 0.1**	
	Mean	**Std. Dev.**	**Mean**	**Std. Dev.**	**Mean**	**Std. Dev.**
Uniform Flow	0.0023	0.00010	0.0022	0.00010	0.0022	0.00011
Turbulent Flow	-0.0025	0.18	-0.0030	0.17	-0.0003	0.18

Real Data

Real data was acquired in a turbulent water tank. 8MP cameras were used along with an Nd:YAG laser with 200 mJ/pulse and a Model 610036 synchronizer timing box. The data was time-resolved and analyzed using the DPIR algorithm. A sample result can be seen in Figure 9, where 5 time-steps are shown.

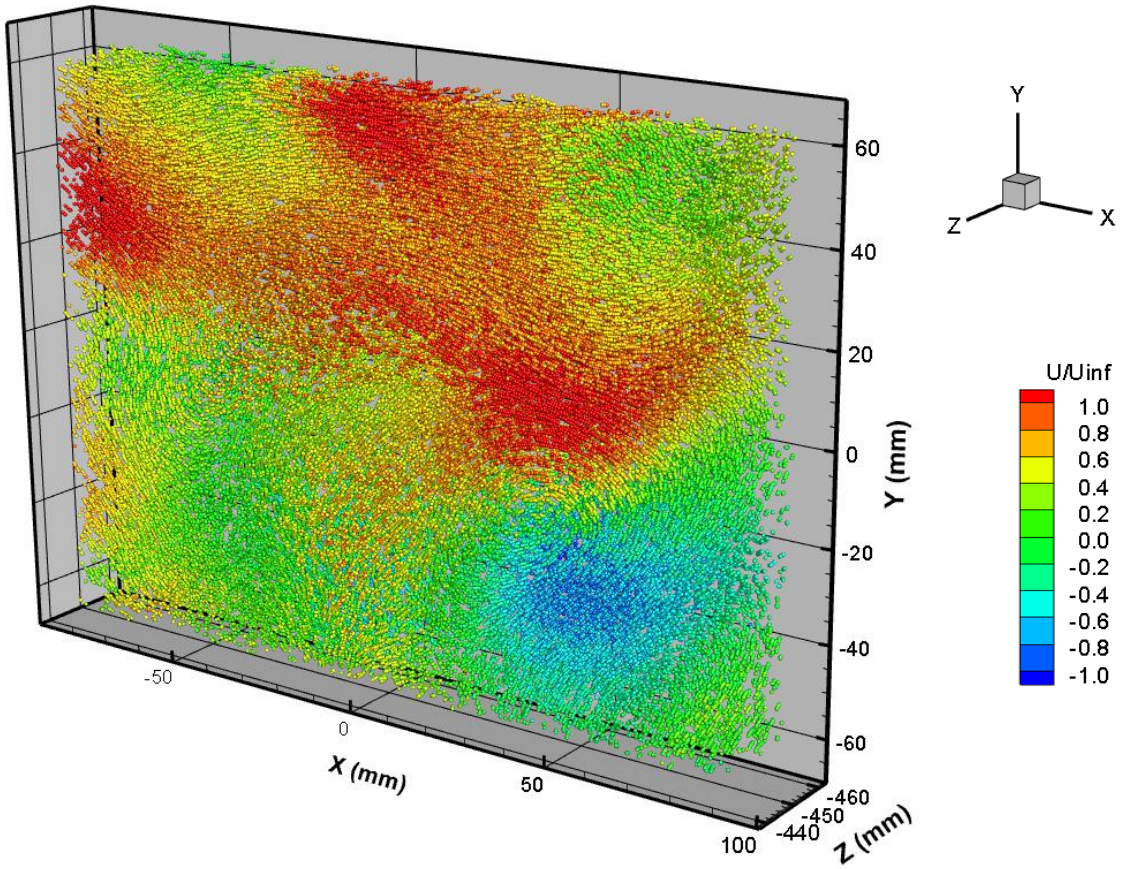

Figure 9. Water flow volumetric data analyzed using Dense Particle Identification and Reconstruction.

Conclusion

Dense Particle Identification and Reconstruction (DPIR) is a technique used to extract velocity information from dense volumetric particle fields. The technique locates particles imaged on camera sensors and projects them into 3D space, tracking them over time in order to compute velocity fields and other quantities. The results of testing with both synthetic and real data indicate a substantial increase in the accuracy and decrease in ghost particles over prior techniques.

References

Boomsma, Troolin (2018) "Time-Resolved Particle Image Identification and Reconstruction for Volumetric 4D-PTV," *19th International Symposium on the Application of Laser and Imaging Techniques to Fluid Mechanics*, Lisbon, Portugal July 16-19, 2018.

Cardwell, N., Vlachos, P., and Thole, K., 2011, "A Multi-Parametric Particle-Paring Algorithm for Particle Tracking in Single and Multiphase Flows," *Meas. Sci. Tech*., **22.**

Hartley and Zisserman (2003) *Multiple View Geometry in Computer Vision*.

Kahler, C., Scharnowski, S., and Cierpka, C., 2012, "On the Uncertainty of PIV and PTV Near Walls," *Experimental Fluids*, **52**.

Lei, Y., Tien, W., Duncan, J., Paul, M., Ponchaut, N., Mouton, C., Dabiri, D., Rosgen, T., Hove, J., 2010, "A Vision-Based Hybrid Particle Tracking Velocimetry (PTV) Technique Using a Modified Cascade Correlation Peak-Finding Method," *Exp. Fluids*.

Weineke, B., 2013, "Iterative Reconstruction of Volumetric Particle Distribution," *Meas. Sci. Technol.*, **24**.

UNDERSTANDING, ACCELERATED

TSI Incorporated – Visit our website **www.tsi.com** for more information.

USA	**Tel:** +1 800 680 1220	**India**	**Tel:** +91 80 67877200
UK	**Tel:** +44 149 4 459200	**China**	**Tel:** +86 10 8219 7688
France	**Tel:** +33 1 41 19 21 99	**Singapore**	**Tel:** +65 6595 6388
Germany	**Tel:** +49 241 523030		

V3V-FLEX-011 Rev. A (7/18/2019) A4 Printed in U.S.A.