Journal of
open research software

Particle Image Velocimetry for MATLAB: Accuracy and enhanced algorithms in PIVlab

SOFTWARE METAPAPER

]u[ubiquity press

WILLIAM THIELICKE

RENÉ SONNTAG

*Author affiliations can be found in the back matter of this article

ABSTRACT

PIVlab is a free toolbox and app for MATLAB®. It is used to perform Particle Image Velocimetry (PIV) with image data: A light sheet illuminates particles that are suspended in a fluid. A digital camera records a series of images of the illuminated particles. The input images are divided into sub-images (interrogation areas), and for each of these, a cross-correlation is performed. The resulting correlation matrix is used to estimate the most probable displacement within each interrogation area. PIV is extensively used for flow analyses where a thin laser sheet illuminates suspended particles in the fluid, but also for other moving textures, like cell migration or ultrasonic images. This paper presents several improvements that were implemented in PIVlab, enhancing the robustness of displacement estimates. The benefit of these improvements is evaluated using experimental images and synthetic images of particle and non-particle textures. Linear correlation and repeated correlation increase the robustness and decrease bias and root-mean-square (RMS) error of the displacement estimates. Particle images have a significantly lower bias and RMS error than non-particle images.

CORRESPONDING AUTHOR:
William Thielicke
OPTOLUTION Messtechnik GmbH, Lörrach, Germany; Biomimetics-Innovations-Center, City University of Applied Sciences Bremen, Germany
w.th@gmx.de

KEYWORDS:
particle image velocimetry; PIV; MATLAB; fluid dynamics; flow analysis; ensemble correlation; repeated correlation; texture image velocimetry

TO CITE THIS ARTICLE:
Thielicke W, Sonntag R 2021 Particle Image Velocimetry for MATLAB: Accuracy and enhanced algorithms in PIVlab. *Journal of Open Research Software,* 9: 12. DOI: *https://doi.org/10.5334/jors.334*

(1) OVERVIEW

INTRODUCTION

Originally, Particle Image Velocimetry (PIV) is applied as non-intrusive technique to measure velocities in fluids [e.g. 1–5]: A laser sheet illuminates reflective particles that were added to the fluid. A camera records the motion of the particles. The displacement of small groups of particles from one image to the next image is finally quantified by image cross-correlation. Displacement per time yields the space-resolved velocity in the images. PIV has become increasingly useful in research disciplines other than fluid dynamics in the last years: PIV is applied to cell motion [6], granular flows [7], ultrasonic images [8] and numerous other kinds of imagery where velocities or displacements need to be quantified (the reader can get a good impression of this diversity on the Microsoft Academic profile[1] or the Google Scholar profile[2] of PIVlab). As PIV has become an important method in the last decades, there are many commercial software packages available (e.g. Dantec DynamicStudio, ILA PIVview, LaVision Flowmaster, TSI INSIGHT). These are often not limited to 2D PIV, but can also process planar 3D (stereo) PIV and even more advanced data like volumetric PIV. There is also free PIV software (OpenPIV [9], Fluere [10], Fluidimage [11], mpiv [12], JPIV [13], UVMAT [14]) that is extensively used in scientific research.

PIVlab is aimed at planar, 2D PIV and has been initially published in 2010; 30 updates with new features or fixes have been released, based on user feedback and personal demands. A metapaper on PIVlab has been published in the Journal of Open Research Software in 2014 [15] which has been cited more than 1400 times to date. A more in-depth description of the algorithms and validation has also been published [16]. PIVlab was honoured as 'Pick of the Week' in MATLAB®'s official File Exchange Website. It currently is MATLAB®'s most popular non-official free toolbox and app. Recently, PIVlab's correlation algorithms have been improved and modified, and these changes and the resulting benefits are presented in this software metapaper.

1 IMPLEMENTATION AND ARCHITECTURE

PIVlab is implemented as a toolbox and app for MATLAB®. A flow chart of the main program is shown in ***Figure 1***. A PIV analysis starts with the image input and ends with exporting data. ***Figure 1*** also shows what files contain the individual functionalities. More details on the file content is given in the Github wiki of PIVlab. PIVlab is designed to be easy to use, therefore all settings and all processing are performed in a GUI (PIVlab_GUI.m, see ***Figure 2***). Selecting menu items changes the contents of the panel on the left side, allowing to see or change parameters easily. If desired, PIV processing can also be done without the GUI. An example that shows command-line processing is given with PIVlab_commandline.m.

Further details on the implementation are available elsewhere [15] and the remaining part of this section focusses on the implementation of new and enhanced features.

1.1 IMPLEMENTATION OF ENSEMBLE CORRELATION

Ensemble correlation has been introduced by Santiago et al. in 1998 [17]. It is especially helpful in micron-resolution particle image velocimetry (micro-PIV, µPIV), as it can deal with exceptionally low seeding densities,

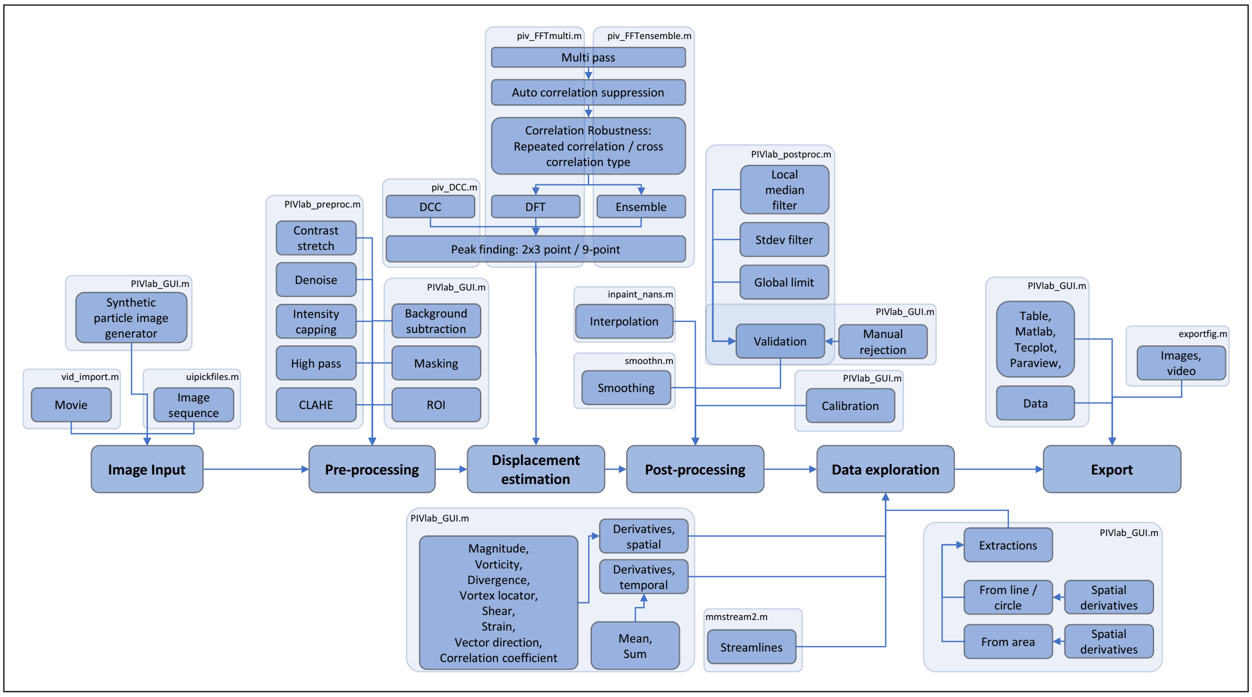
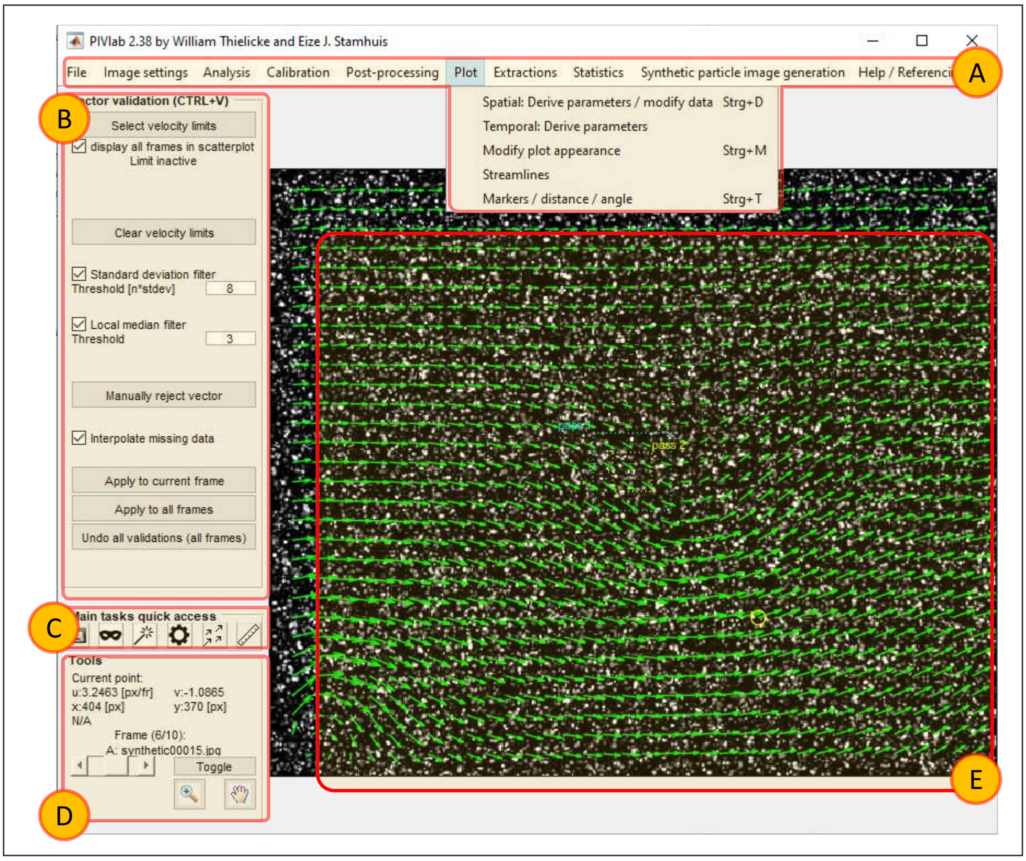


Figure 1 Flowchart of the architecture of PIVlab.



☑ display all frames in scatterplot
Limit inactive

☑ Standard deviation filter
Threshold [n*stdev] 8

☑ Local median filter
Threshold 3

☑ Interpolate missing data

Current point:
u:3.2463 [px/fr] v:-1.0865
x:404 [px] y:370 [px]
N/A
Frame (6/10):
A: synthetic00015.jpg

Figure 2 Main graphical user interface. **A:** Menu with several sub menu entries. **B:** Main panels are shown in this area, the content changes depending on the menu item that was selected. **C:** Quick access toolbar, allowing to access the most important menu items. **D:** Tools panel, showing information about the currently selected point. It also allows to skip through the input images/video, and toggle within an image pair. Zoom and pan buttons are also shown. **E:** Image area. Images and results are shown here.

where standard PIV algorithms would fail [18]. In ensemble correlation, a series of sparsely seeded images of steady flow is analysed via cross-correlation. The resulting correlation matrices are averaged before a peak searching algorithm is used, resulting in much better signal-to-noise ratio and high vector resolution for low particles densities become possible. However, the flow needs to be steady, and must not change substantially with time. Otherwise the correlation peak would be significantly broadened, hindering the detection of the correct displacement. Ensemble correlation in PIVlab features all the advanced correlation techniques of the regular correlation (multiple passes, window deformation, suppression of auto-correlation, background subtraction etc.). The benefit of ensemble correlation vs. regular correlation with synthetic particle images and low seeding (< 1 particle per interrogation area) is presented in the section 'Quality control'.

1.2 CORRELATION ALGORITHM ENHANCEMENTS

1.2.1 Circular and linear cross-correlation

PIVlab can perform direct cross-correlation of the images in the spatial domain, or cross-correlate the signals in the frequency domain using discrete Fourier transform (DFT). DFT is betwenn 30 and 50% faster (depending on the implementation [15]). PIVlab typically uses a circular cross-correlation to perform the DFT. Circular cross-correlation assumes the signal (image data) to be periodic, which is not accurately representing reality. Hence the assumption of periodicity might introduce frequencies in the DFT spectrum that are not existent [5]. To suppress this negative effect, the signal can be zero-padded, yielding an approximation of a linear, non-periodic cross-correlation. As typical image data includes some noisy, non-zero background signal, zero-padding introduces an edge-discontinuity, which again deteriorates the spectrum of the DFT analysis and thereby the cross-correlation signal [5]. The effect of these 'sharp borders' can be attenuated by subtracting the average intensity from the input images, finally yielding a higher 'quality' (in terms of bias, RMS error and valid data yield) displacement estimate than circular cross-correlation. As two-dimensional zero-padding increases the size of the dataset significantly, these improvements via linear cross-correlation come at the cost of increased computing time. The effects are quantified in the section 'Quality control'.

1.2.2 Repeated correlation

Another approach to enhance the robustness of the cross-correlation is 'repeated correlation'. This concept was proposed as a non-post-interrogation method to reduce spurious vectors from PIV data [19]. The method can enhance the data yield for images with a bad signal-to-noise ratio. Cross-correlation is not performed once for every resulting displacement estimate, but five times in total: The interrogation windows are shifted left-up, right-up, right-bottom and left-bottom by 25% of the interrogation window length. These correlation matrices are multiplied, resulting in a new matrix with less noise and a more distinctive peak: Every correlation value that is not present in each of the five correlation matrices will be eliminated from the resulting correlation matrix [19].

This combined peak is then processed as usual with a sub-pixel peak finder to derive the displacement. Again, this enhancement comes at the cost of increased computational time, more details are given in the section 'Quality control'.

As PIVlab is often used by people that are just starting with PIV, or by people that cannot afford the time to look into every aspect of PIV processing, a single setting, called 'Correlation robustness' was introduced. The selection of this parameter is implemented as a drop-down menu and enables the user to find a suitable compromise between processing time and displacement estimation robustness (see ***Table 1***).

- 'Standard' robustness runs an analysis as described earlier [15], with linear window deformation and circular correlation
- 'High' uses spline instead of linear window deformation and additionally replaces the circular cross-correlation with linear cross-correlation
- 'Extreme' additionally uses repeated correlation

1.3 BACKGROUND ELIMINATION

Uneven illumination or stationary background objects in the input images can sometimes result in spurious displacement estimates if the correlation of the background is stronger than the correlation of the desired signal. These undesirable results can be effectively suppressed by calculating the average of a set of input images and subtracting the resulting image from every input image. This operation now can be executed directly in PIVlab.

1.4 AUTO-CORRELATION SUPPRESSION

Another supportive method of suppressing the influence of a stationary background signal in the correlation is the 'suppression of auto-correlation'. If the background signal dominates the correlation, but only a single image pair is available, the calculation of an average background image is impossible. Usually, the cross-correlation would then lock on the background signal, reporting zero displacement. This can be resolved by disallowing near-zero displacement (auto-correlation). It is achieved by masking the central peak in the correlation matrix. The peak finder will then detect the second highest peak in the correlation matrix, which likely is the desired signal of interest. The application of 'auto-correlation suppression' is limited to cases where the displacement of interest is never zero.

2 QUALITY CONTROL

In this section, we present the results of tests with synthetic and experimental datasets. This kind of functional testing is performed with every software addition in PIVlab to ensure reliable results. Every functionality of the software is tested with different PIV images, and errors are fixed before a release. Detailed analyses of earlier PIVlab features are presented elsewhere [15, 16]. All the data of the test cases that are presented on the following pages are available in PIVlab's Github repository.

2.1 ENSEMBLE CORRELATION

Ensemble correlation has been tested with 75 synthetic particle images that contain on average 0.64 ± 0.08 particles (diameter 3 ± 1 pixels) per interrogation area (64·64 pixels). These images have been analysed in PIVlab using regular cross-correlation and ensemble correlation. As expected, the regular PIV algorithm fails to detect the correct displacement. PIV can best detect a displacement for patterns formed by groups of particles (typically > 5 particles [5]). Averaging the calculated displacements for all 75 image pairs does not enhance the displacement estimate (see ***Figure 3***). Using ensemble correlation on the same dataset significantly enhances the robustness of the analysis (see ***Figure 3***), lowering bias and RMS displacement error (average of regular cross-correlation: –1.58 ± 1.42 pixels; ensemble correlation: –0.33 ± 0.57 pixels).

CORRELATION ROBUSTNESS	WINDOW DEFORMATION	CROSS CORRELATION	REPEATED CORRELATION	PROCESSING TIME
'Standard'	linear	circular	off	**–**
'High'	spline	linear	off	**++**
'Extreme'	spline	linear	on	**++++**

Table 1 Correlation robustness settings in PIVlab.

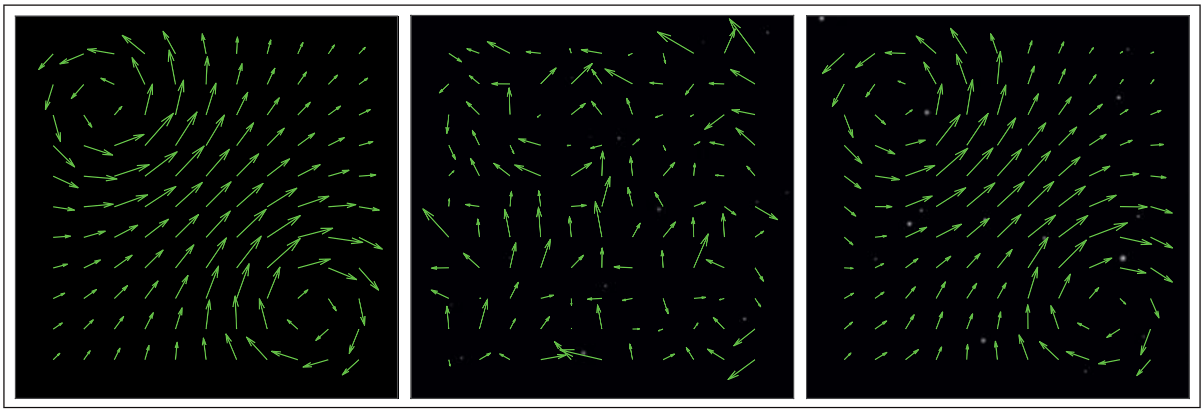


Figure 3 The effect of ensemble correlation for synthetic particle images with only 0.64 particles per interrogation area. **Left:** True displacement of the synthetic particle images. **Middle:** Average displacement of an analysis of 75 images using regular correlation. **Right:** Ensemble correlation of 75 images.

2.2 CORRELATION ROBUSTNESS SETTINGS

The accuracy of the three different correlation robustness settings was checked with synthetic particle images. The results were also compared with a 4k Euros commercial PIV software ('CS1'). 500 image pairs (with a particle displacement ranging from 0 to 4 pixels) for 6 different particle image quality sets with increasing gaussian noise (gaussian white noise variance 0–0.025) and increasing particle pair loss (0–25%) were tested [for details on these parameters see 16]. The synthetic images contain on average 10 particles (diameter 3 ± 1 pixels) per interrogation area (24·24 pixels). The following settings were used in PIVlab:

- Image pre-processing: Contrast limited adaptive histogram equalization (CLAHE) [see 15] enabled with 20 pixels window size
- PIV settings: FFT window deformation, 3 passes, interrogation areas 64·64, 32·32 and 24·24 pixels with 50% overlap
- Correlation robustness: 'standard', 'high', 'extreme'

Similar settings were used in the commercial software:

- Image pre-processing: Dynamic histogram equalization (0–96%)
- PIV settings: FFT correlation, uniform weighting. Multi-grid interrogation areas 64·64, 32·32, 24·24 pixels with 50% overlap, image B-spline interpolation
- Repeated correlation (3x)

After analysing all images, the bias and RMS error for each true displacement were calculated from the true displacement and the estimated displacement [see 16, for details]. Finally, we calculated the mean absolute bias and the mean RMS error for each analysis, which serves as a measure for the accuracy of that analysis. Furthermore, we counted the amount of displacement estimates that have a deviation of more than 1 pixel with regard to the true displacement. We take the inverse of this number as a measure for the robustness of an analysis. The results show that PIVlab can achieve a very low bias error with the 'extreme' correlation robustness which uses repeated correlation and linear cross-correlation. Even at 'standard' robustness, PIVlab outperforms the commercial software (see ***Figure 4***).

The RMS error of the commercial software is slightly lower for images with very low noise and low particle pair loss (see ***Figure 5***). However, for images with higher noise and particle pair loss, the 'extreme' correlation robustness setting of PIVlab outperforms all other analyses. The 'extreme' robustness setting has also the lowest number of wrong vectors, followed by the commercial software (see ***Figure 6***).

Although PIVlab yields results that are as good as or even better than the commercial software, it clearly has a much slower processing time. The 'extreme' robustness setting is by a factor of 30 slower than the commercial software due to the zero-padding and the repeated correlation. Even the 'standard' robustness is 3 times slower than the commercial software (see ***Figure 7***).

We believe that this limitation is acceptable, when taking the enhanced robustness of PIVlab into account. Furthermore, computers have so much computing power nowadays, that the analysis of a full HD image (1920·1080 pixels) with 66·119 displacement estimates and 2 passes on a standard laptop (Intel Core i5 7200U @ 2.5 GHz) takes only 3.2 seconds with 'standard' correlation robustness and 22 seconds with 'extreme' correlation robustness (including image pre-processing).

2.3 BACKGROUND ELIMINATION AND AUTO-CORRELATION SUPPRESSION

The functionality of background elimination and auto-correlation suppression was validated with 100 experimental particle image pairs in which a synthetic background was added. The background signal is stronger than the moving particles, hence a displacement of zero is calculated in the presence of the background signal. By subtracting the average intensity from each image before analysis, the background

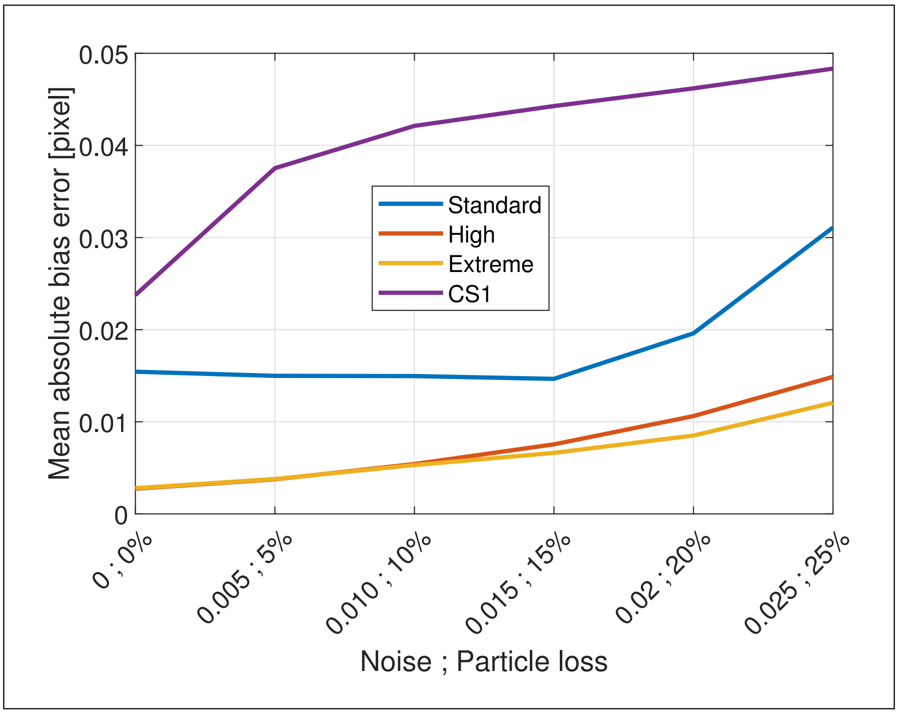


Figure 4 Mean absolute bias error for synthetic image pairs with increasing noise and particle pair loss. PIVlab's 'extreme' robustness setting performs best, and PIVlab performs generally better than the commercial software.

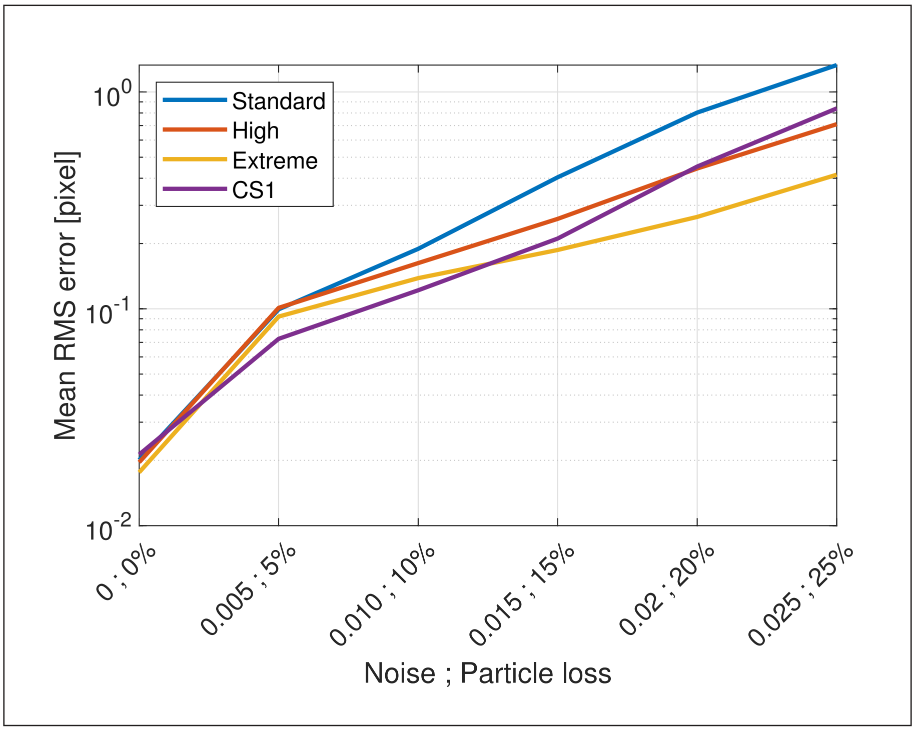


Figure 5 Mean RMS error for synthetic image pairs with increasing noise and particle pair loss. For image pairs with low noise and low particle pair loss, the commercial software performs slightly better. At higher noise and particle loss levels, PIVlab's 'extreme' correlation robustness performs best.

signal is effectively removed (see ***Figure 8***). If such a background removal cannot be performed (e.g. when only a single image pair is available), suppression of auto-correlation can do a similar job, although the difference to the data without artificial background is slightly higher (see ***Figure 8***).

2.4 EXPERIMENTAL DATA: COMPARISON OF FLOW VELOCITY MEASUREMENTS WITH A PROPELLER ANEMOMETER

Using synthetic particle images for analysing robustness and accuracy of PIV software is comfortable and useful to determine the influence of isolated image parameters

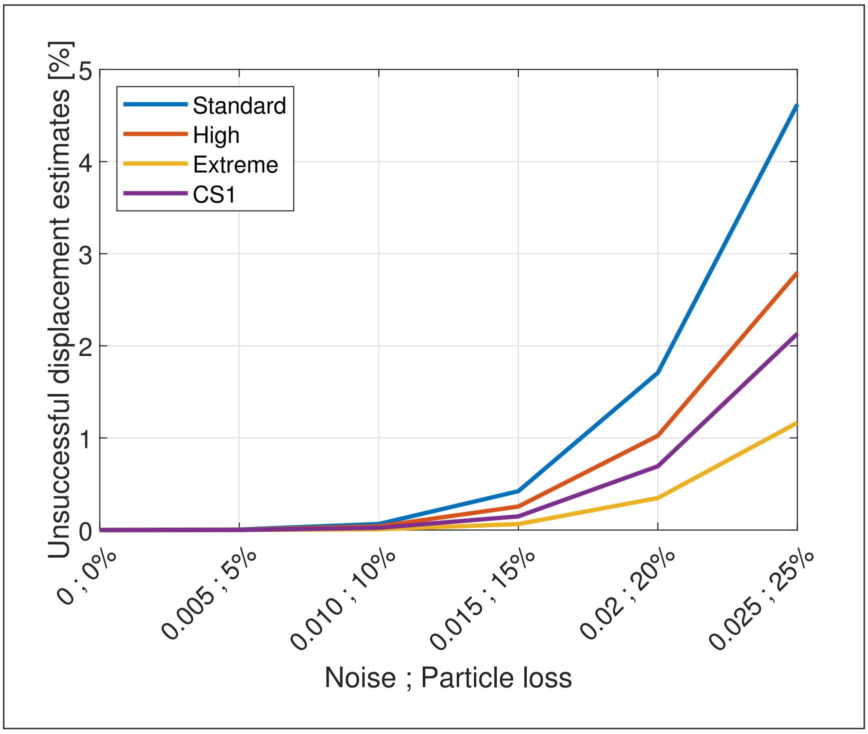


Figure 6 The amount of unsuccessful displacement estimates (estimates that deviate more than 1 pixel from the true displacement) serve as an indicator for the robustness of an analysis. PIVlab's 'extreme' correlation robustness performs best, followed by the commercial software.

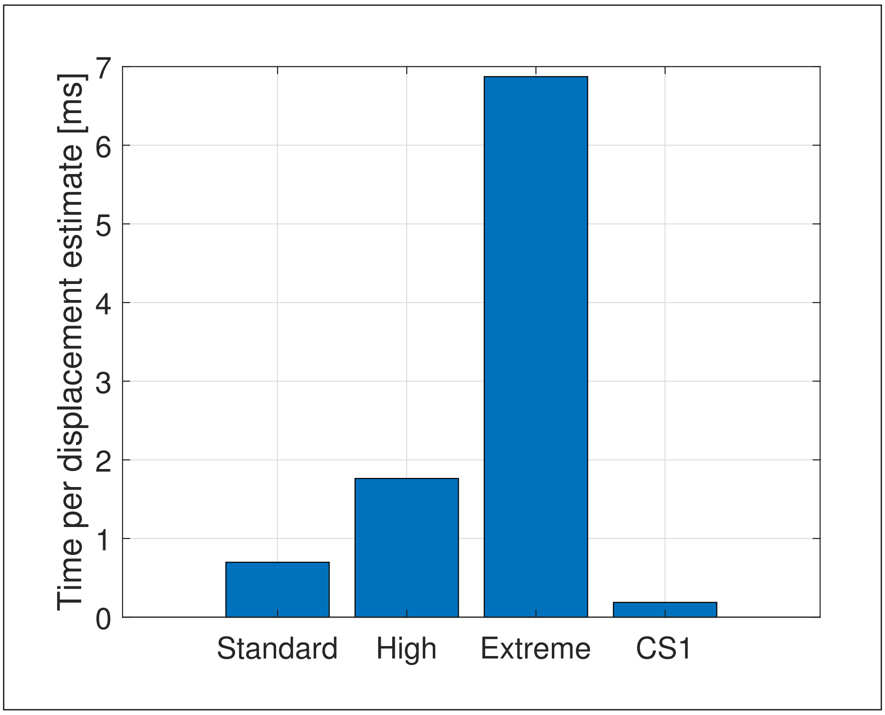


Figure 7 Computing time per displacement estimate. PIVlab is between 3 and 30 times slower than the commercial software.

like noise, particle size, particle pair loss etc. Such analyses have been performed extensively for PIVlab [16]. Analyses of PIV accuracy are more complicated with experimental data, as the true velocity is often unknown. Here, we present a comparison with propeller anemometer measurements (Schiltknecht MC20, MiniWater20 Micro) in a water tunnel (located at the Bremen University of Applied Sciences, water temperature 23°C, seeded with 57 µm polyamide particles). The PIV system consists of a Phantom VEO 410L camera (1280·800 pixels) and a New Wave Pegasus-PIV 200 dual cavity laser running at 400 Hz. The images were analysed in PIVlab (2 passes, interrogation areas 64·64 and 32·32 pixels with 50% overlap, 'extreme' correlation robustness).

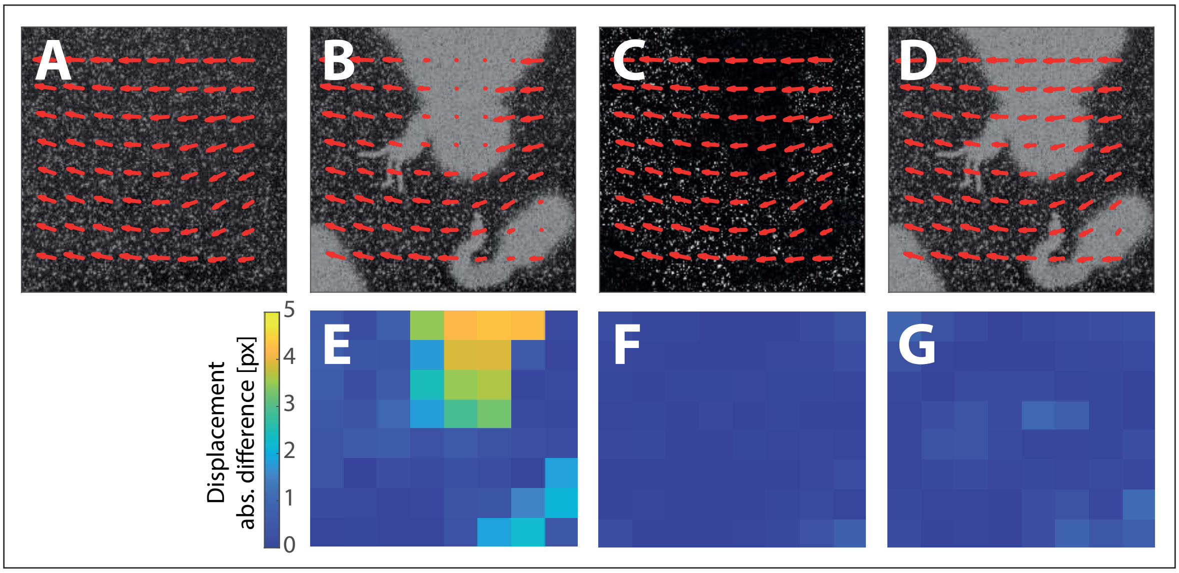


Figure 8 Background elimination and auto-correlation suppression in PIVlab. **A:** PIV data without artificial background (true displacements) **B:** A background signal is superimposed on the particle image. The signal of the background is stronger than the signal of the moving particles, therefore, a displacement of zero is detected in these areas. **C:** The background signal has been removed from the images before analysis by subtracting the average intensity of a larger amount of image pairs. The image has become darker in the background area, but a weak particle signal became visible again. Displacement estimates are possible also in the area where a strong background signal was present. **D:** Auto-correlation suppression ignores displacement estimates that are close to zero and takes the second highest peak in the correlation matrix for estimating the displacement. It can effectively remove a correlation of the background signal in this case. **E:** Absolute difference between the true displacements in A and the displacements in B. Displacements in the areas of strong background signal are incorrect. **F:** Absolute displacement difference between A and C. Displacement estimates are correct, removing the background signal does not affect the displacement estimate. **G:** Absolute difference between A and D. Displacement estimates have only a small difference to the true displacements.

As can be seen in ***Figure 9***, the anemometer has a large influence on the flow. A sphere in a moving fluid decelerates the flow in front of it (at 0°), and accelerates the flow at the sides (at 90° and 270°). We therefore chose an area at 45°, where we expect the influence from the anemometer to be minimal (lateral distance: about 40 mm). The water tunnel was running at several speeds between 0.15 and 0.6 m/s. For each speed, the anemometer reading was averaged over time. Eighteen image pairs were analysed with PIVlab for each speed. Every image pair yielded 4·7 vectors inside the red rectangle, resulting in 504 velocity measurements for each speed. The mean value of these measurements was compared to the mean anemometer reading. There is an exceptionally good agreement between the methods (see ***Figure 10***). The linear regression has a slope of 1.0082 and an offset of 0.0002 m/s. The correlation coefficient is 0.9998, indicating a strong linear relationship between the two methods. The bias error is 0.984% and the RMS error is 2.22%. Our analysis cannot assess which of the two methods more accurately measures the true velocity. Previous tests with PIVlab [16] imply displacement estimates with an uncertainty below 0.01 pixels with synthetic images, which is also achieved by other PIV software [20]. The analysis of experimental particle images via PIV inherently has a higher uncertainty of about 0.1 pixels due to particle intensity variations between consecutive images [21].

2.5 PERFORMANCE WITH NON-PARTICLE IMAGE TEXTURES

PIVlab is mostly used for analysing the displacement and velocity of particles suspended in fluids, but also for completely different data (see section 'Reuse potential' for some examples). This arises the question whether PIV is suitable for non-particle images. We therefore generated three simplified artificial textures ('checkerboard', 'difference clouds'[3] and 'gaussian noise'[4]) and an average of the previous three textures ('combined', see ***Figure 11***).

Integer displacements between 5 and 20 pixels were analysed (as we have no means of generating sub-pixel displacements with these textures). PIVlab was used with 'high' correlation robustness and a multi-pass window deformation analysis with interrogation areas of 128·128 and 64·64 pixels with 50% overlap and two passes.

Analysing the bias and RMS error shows that particle images work best (bias = 0.014% and RMS error = 0.01%). The other synthetic textures perform worse ('difference clouds' texture: bias = 5.6% and RMS error = 3.6%), the gaussian noise texture is least robust as 70% of all correlation estimates are unsuccessful (see ***Figures 12, 13*** and ***14***). The displacement of a noise texture is challenging to detect by cross-correlation, as the correlation of random noise with a displaced random noise is not much stronger than the correlation with some other random noise. The 'combined' texture works better than the other synthetic textures, as it has a texture that is less monotonic which is less ambiguous in cross-correlation.

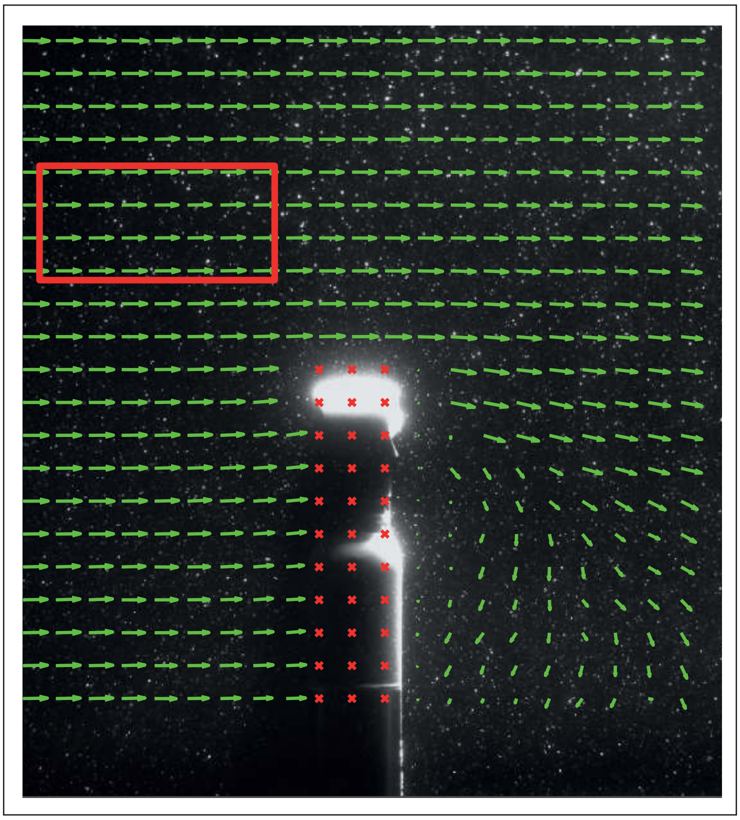


Figure 9 Propeller anemometer in a water tunnel. The flow around the anemometer was mapped using PIV. The mean flow velocity inside the red rectangle was used for the comparison.

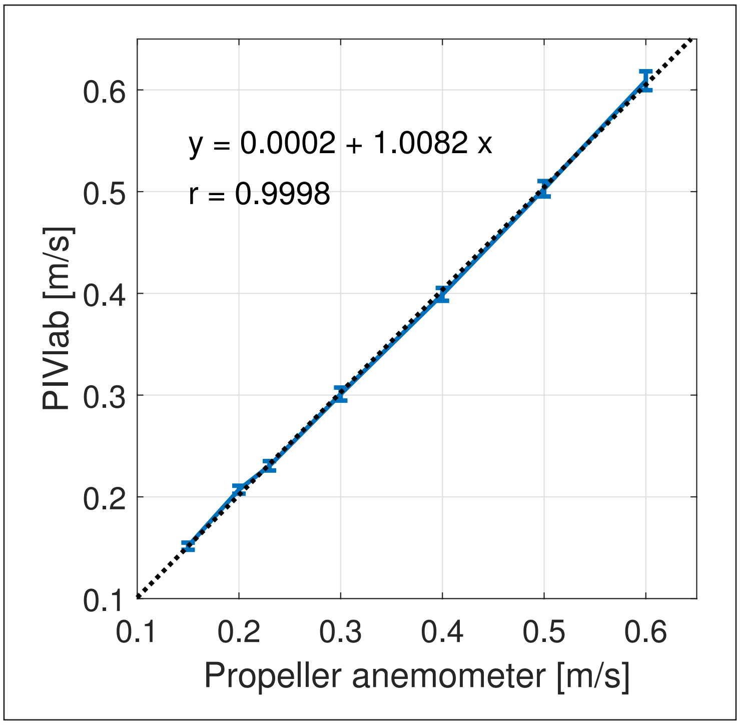


Figure 10 Comparison of flow speed measurements with PIVlab and a propeller anemometer. The relation is linear and has a slope of almost unity with almost zero offset.

An explanation for the performance with these synthetic textures can be found in the peak finding algorithm that is used in PIVlab. As in most (commercial and free) PIV tools, PIVlab fits a Gaussian function to the integer peak location in the correlation matrix (see ***Figure 16***). Only the adjacent four pixels around the integer peak location contribute to this fit. Using a Gaussian function is very appropriate for particle images, where each particle closely matches a Gaussian intensity distribution. Cross-correlating two Gaussian distributions

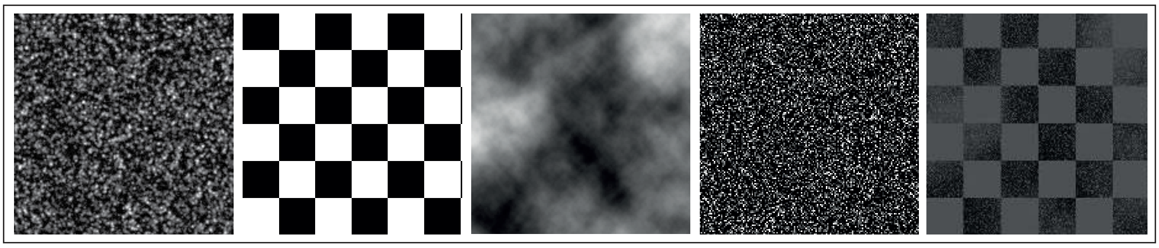


Figure 11 Textures that were tested in PIVlab. **From left to right:** Experimental particle image, checkerboard, difference clouds, gaussian noise, combination of checkerboard + difference clouds + gaussian noise.

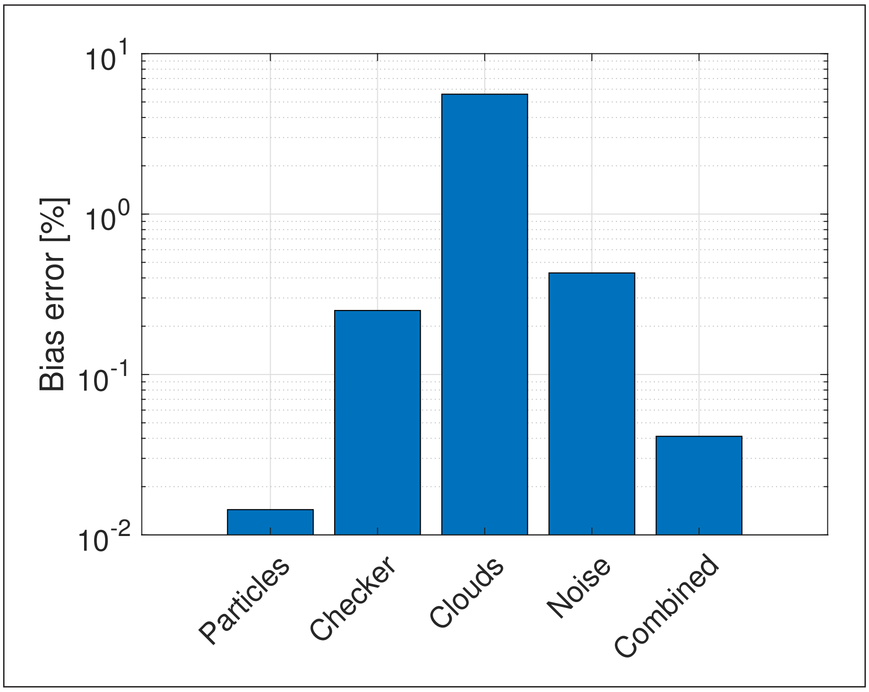


Figure 12 Bias error of the analysis of different texture types. Particle images perform best with a bias of 0.014%. The 'difference clouds' texture has a bias of 5.6%.

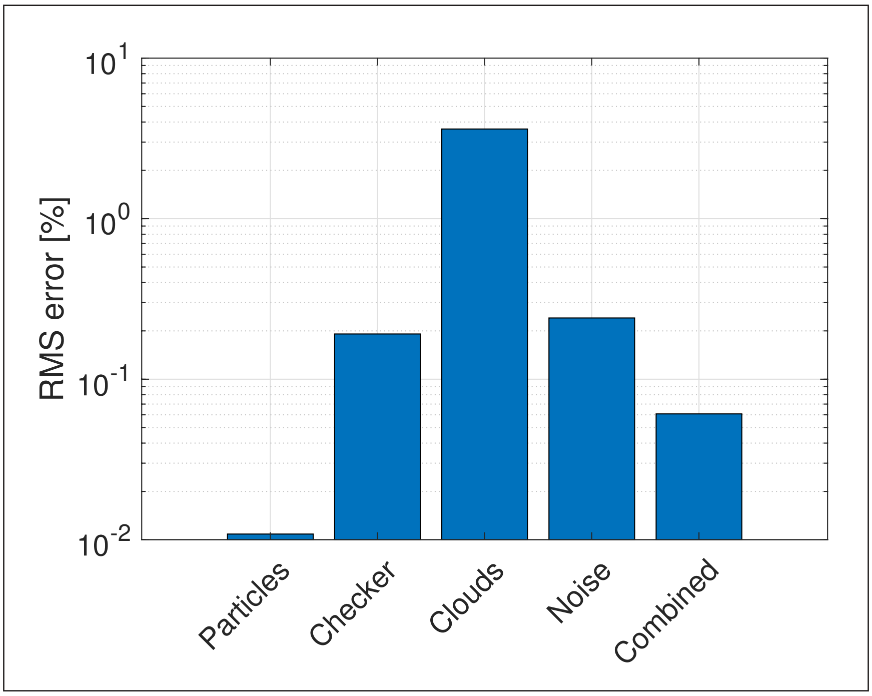


Figure 13 RMS error of the analysis of different texture types. Particle images perform best with an RMS error of 0.01%. The 'difference cloud' texture has an RMS error of 3.6%.

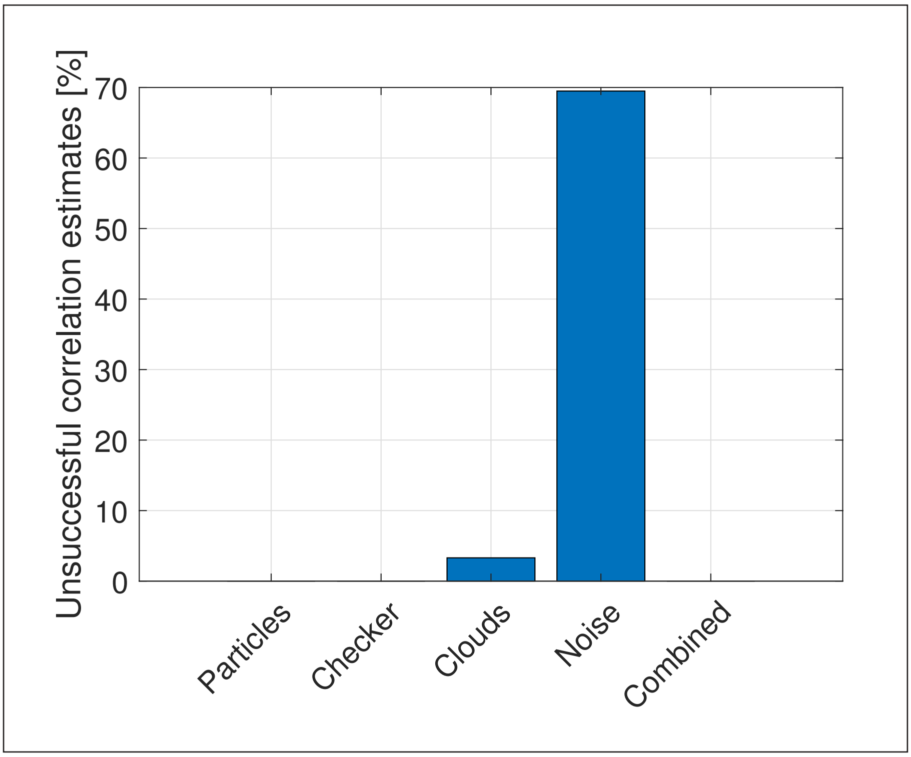


Figure 14 Unsuccessful correlation estimates in the analysis of different texture types. Gaussian noise images show a low rate of successful displacement estimates.

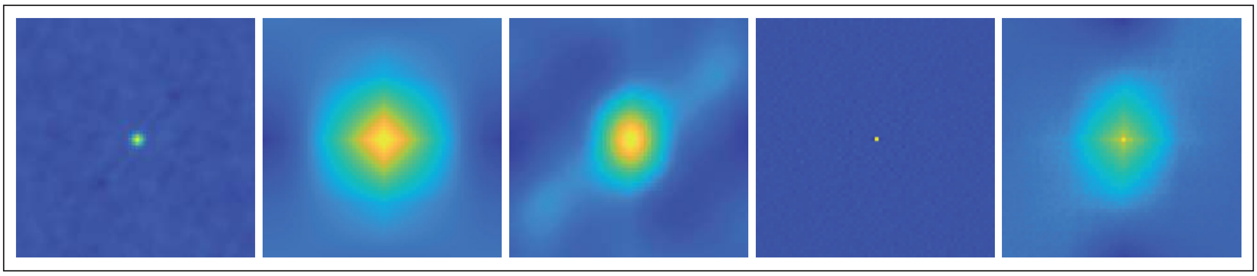


Figure 15 Correlation matrices for a displacement of 5 pixels. **From left to right:** Experimental particle image, checkerboard, difference clouds, gaussian noise, combination of checkerboard + difference clouds + gaussian noise.

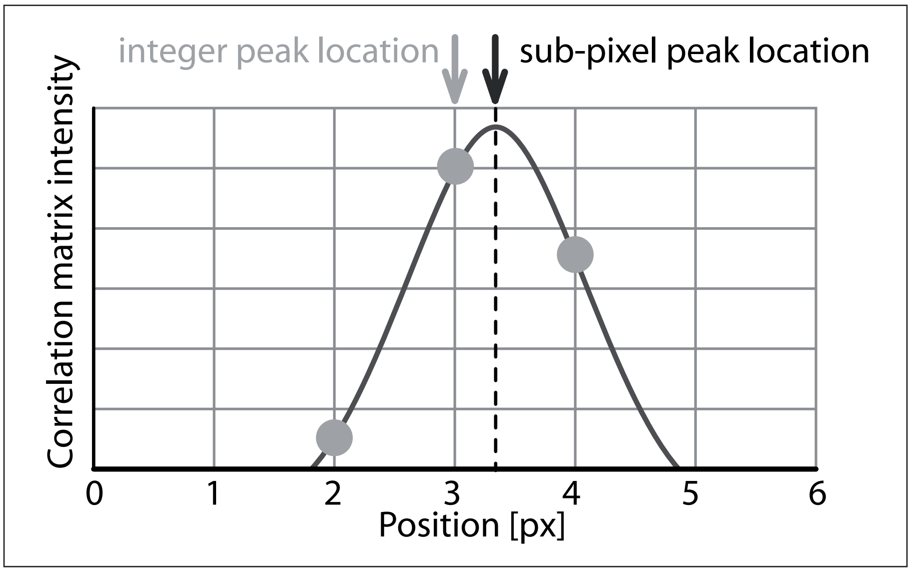


Figure 16 Principle of the Gaussian fit: Subpixel precision is achieved by fitting a one-dimensional Gaussian function (solid line) to the integer intensity distribution of the correlation matrix (dots) for both axes independently (only one axis is shown here). [Taken from 16].

again yields a correlation matrix with a Gaussian intensity distribution [22].

The artificial textures we tested do not have a Gaussian intensity distribution in the proximity of the integer peak location in the correlation matrix (see ***Figure 15***). Therefore, the Gaussian function cannot be fitted correctly, and the peak is not located reliably, resulting in relatively high bias and RMS errors. A non-suitable sub-pixel estimator will fail even for an integer displacement as in this experiment. Therefore, users have to take care when using PIVlab for non-particle images: The accuracy of the displacement estimates will be worse. To counter this issue, PIVlab should be enhanced with an additional sub-pixel peak finder that is optimized for non-particle images. Changing the sub-pixel peak finder algorithm can also be done automatically by detecting the broadness of the peak and switching from a Gaussian fit to a more suitable algorithm if a certain threshold is exceeded. The implementation of another displacement estimation technique (e.g. optical flow) would be a feasible option. These enhancements would make PIVlab more accurate for the analysis of non-particle images.

(2) AVAILABILITY

OPERATING SYSTEM

Based on MATLAB® 7.10.0 (R2010a): Windows, UNIX/Linux, Macintosh

PROGRAMMING LANGUAGE

MATLAB® 7.10.0 (R2010a), upward compatible

ADDITIONAL SYSTEM REQUIREMENTS

MATLAB® 7.10.0 (R2010a): 1 GB disk space, 1 GB RAM

DEPENDENCIES

MATLAB®'s Image Processing Toolbox is required

LIST OF CONTRIBUTORS

William Thielicke, OPTOLUTION GmbH: Programming, maintenance, and support, writing this paper.

René Sonntag, Bremen University of Applied sciences: Testing, feedback on PIVlab, writing this paper, experiments

Eize J. Stamhuis, University of Groningen: Initial design of the tool, tips, and discussions on the functionality

SOFTWARE LOCATION

Archive

Name: Github.com

Persistent identifier: https://github.com/Shrediquette/PIVlab

Licence: MIT License

Publisher: William Thielicke

Version published: 2.39

Date published: 02/2021

LANGUAGE

English

(3) REUSE POTENTIAL

The authors try to make PIVlab accessible to researchers with and without programming skills. The programming environment MATLAB® was chosen, as this software is available in many universities, with many students and researchers having access to it. The required 'Image Processing Toolbox' is highly likely available in most MATLAB® installations. MATLAB® has a hurdle free, flat learning curve, and many researchers are already used to the workflow. The download link of PIVlab is easy to locate on MATLAB®'s File Exchange server and the toolbox and app installs easily. No programming skills at all are required to use PIVlab.

After installation, users can quickly validate that PIVlab is working correctly by running the provided script 'Accuracy.m'. This script generates random synthetic images, pre-processes, and analyses these images, and performs a comparison between true displacements and measured displacements. The resulting accuracy is reported to the user.

Furthermore, users can easily perform an analysis themselves using one of the provided example datasets and following one of the quick start videos.[5] The default settings of PIVlab are carefully chosen to provide reasonable results for almost any dataset. Should the defaults fail, PIVlab can also suggest suitable settings for the user-specific dataset via image analysis by clicking a button.

PIVlab checks for required toolboxes and dependencies during start-up and reports the result. When a new version of PIVlab is released, a message during start-up of the GUI is displayed to notify the user of the update (internet connection required). This ensures that users keep their copy up to date and they do not miss enhancements, fixes, or new features.

Support is available via a Google Groups forum[6] with currently about 50 posts per month. Here, users can suggest new features or ask for assistance with analyses. PIVlab's wiki page on Github gives information on installation, tutorials, support and how users can contribute.[7] As mentioned in the introduction, PIVlab has been widely used for image analysis in quite different fields. It has also been reused to develop new software [see 15]. Particle Image Velocimetry with PIVlab has shown to have potential to help researchers not only in flow analyses [e.g. 23], but also in cell motion [e.g. 24], sand deformation during extraterrestrial-life exploration missions [e.g. 25], avalanche research [e.g. 26], inkjet printing related studies [e.g. 27], blood spray analyses [e.g. 28] and faeces disintegration research [e.g. 29]. On average a new study is published every day that uses PIVlab in an exciting new way. Due to the diversity, the term ‘Particle image Velocimetry’ is increasingly misleading, and ‘Texture Image Velocimetry’ might be more appropriate in the future.

NOTES

[1] *academic.microsoft.com/paper/2138221697/citedby/*.

[2] *scholar.google.com/scholar?cites=819244312015141543*.

[3] *helpx.adobe.com/photoshop-elements/using/render-filters.html*.

[4] *helpx.adobe.com/photoshop-elements/using/noise-filters.html*.

[5] e.g. *https://youtu.be/g2hcTRAzBvY*.

[6] *groups.google.com/forum/?hl=en-GB#!forum/pivlab*.

[7] *https://github.com/Shrediquette/PIVlab/wiki*.

ACKNOWLEDGEMENTS

See *www.mathworks.com/matlabcentral/fileexchange/27659*.

FUNDING STATEMENT

WT’s personal MATLAB® license was crowd-funded by PIVlab users.

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHOR AFFILIATIONS

William Thielicke *orcid.org/0000-0001-8866-9769*
OPTOLUTION Messtechnik GmbH, Lörrach, Germany; Biomimetics-Innovations-Center, City University of Applied Sciences Bremen, Germany

René Sonntag
Biomimetics-Innovations-Center, City University of Applied Sciences Bremen, Germany

REFERENCES

1. **Adrian RJ.** Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow: Speckle velocimetry vs particle image velocimetry. *Applied Optics*. 1984; 23: 1690–1691. DOI: *https://doi.org/10.1364/AO.23.001690*
2. **Willert CE, Gharib M.** Digital particle image velocimetry. *Experiments in Fluids*. 1991; 10: 181–193. DOI: *https://doi.org/10.1007/BF00190388*
3. **Keane R, Adrian R.** Theory of cross-correlation analysis of piv images. *Applied Scientific Research*. 1992; 49: 191–215. DOI: *https://doi.org/10.1007/978-94-011-2690-8*
4. **Stamhuis EJ, Videler JJ.** Quantitative flow analysis around aquatic animals using laser sheet particle image velocimetry. *Journal of Experimental Biology*. 1995; 198: 283–294. DOI: *https://doi.org/10.1242/jeb.198.2.283*
5. **Raffel M, Willert C, Wereley S, Kompenhans J.** *Particle Image Velocimetry*. Springer 2 edn. 2007. DOI: *https://doi.org/10.1007/978-3-540-72308-0*
6. **Booth-Gauthier E, Alcoser T, Yang G, Dahl K.** Force-induced changes in subnuclear movement and rheology. *Biophysical Journal*. 2012; 103: 2423–2431. DOI: *https://doi.org/10.1016/j.bpj.2012.10.039*
7. **Sarno L, Carravetta A, Tai Y-C, Martino R, Papa MN, Kuo CY.** Measuring the velocity fields of granular flows – employment of a multipass two-dimensional particle image velocimetry (2d-piv) approach. *Advanced Powder Technology*. 2018; 29: 3107–3123. DOI: *https://doi.org/10.1016/j.apt.2018.08.014*
8. **Voorneveld J, Kruizinga P, Vos HJ, Gijsen FJ, Jebbink EG, van der Steen AF, de Jong N, Bosch JG.** Native blood speckle vs ultrasound contrast agent for particle image velocimetry with ultrafast ultrasound – in vitro experiments in *2016. IEEE International Ultrasonics Symposium (IUS)*. 2016; 1–4. DOI: *https://doi.org/10.1109/ULTSYM.2016.7728614*
9. OpenPIV *http://www.openpiv.net/*.
10. Fluere *https://www.softpedia.com/get/science-cad/fluere.shtml*.
11. Fluidimage *https://pypi.org/project/fluidimage/*.
12. mpiv *https://de.mathworks.com/matlabcentral/fileexchange/2411-mpiv*.
13. JPIV *https://eguvep.github.io/jpiv/*.
14. UVMAT *http://servforge.legi.grenoble-inp.fr/projects/soft-uvmat*.
15. **Thielicke W, Stamhuis E.** Pivlab – towards user-friendly, affordable and accurate digital particle image velocimetry in matlab *Journal of Open Research Software*. 2014; 2. DOI: *https://doi.org/10.5334/jors.bl*
16. **Thielicke W.** *The Flapping Flight of Birds – Analysis and Application* Ph.D. thesis Rijksuniversiteit Groningen. 2014. DOI: *https://doi.org/10.13140/RG.2.2.18656.94728*
17. **Santiago JG, Wereley ST, Meinhart CD, Beebe DJ, Adrian RJ.** A particle image velocimetry system for microfluidics. *Experiments in Fluids*. 1998; 25: 316–319. DOI: *https://doi.org/10.1007/s003480050235*
18. **Delnoij E, Westerweel J, Deen NN, Kuipers JH, van Wpm Wim Swaaij.** Ensemble correlation piv applied to bubble plumes rising in a bubble column. *Chemical Engineering Science*. 1999; 54: 5159–5171. DOI: *https://doi.org/10.1016/S0009-2509(99)00233-X*
19. **Hart DP.** Piv error correction. *Experiments in Fluids*. 2000; 29: 13–22. DOI: *https://doi.org/10.1007/978-3-642-56963-0*
20. **Nobach H, Damaschke N, Tropea C.** High-precision sub-pixel interpolation in particle image velocimetry image processing. *Experiments in Fluids*. 2005; 39: 299–304. DOI: *https://doi.org/10.1007/s00348-005-0999-z*

21. **Nobach H, Bodenschatz E.** Limitations of accuracy in piv due to individual variations of particle image intensities. *Experiments in Fluids.* 2009; 47: 27–38. DOI: *https://doi.org/10.1007/s00348-009-0627-4*
22. **Lourenco L, Krothapalli A.** On the accuracy of velocity and vorticity measurements with piv. *Experiments in Fluids.* 1995; 18: 421–428. DOI: *https://doi.org/10.1007/BF00208464*
23. **Wu K-T, Hishamunda JB, Chen DTN, DeCamp SJ, Chang Y-W, Fernández-Nieves A, Fraden S, Dogic Z.** Transition from turbulent to coherent flows in confined three-dimensional active fluids. *Science.* 2017; 355. DOI: *https://doi.org/10.1126/science.aal1979*
24. **Saw TB, Doostmohammadi A, Nier V, Kocgozlu L, Thampi S, Toyama Y, Marcq P, Lim CT, Yeomans JM, Ladoux B.** Topological defects in epithelia govern cell death and extrusion. *Nature.* 2017; 544: 212–216. DOI: *https://doi.org/10.1038/nature21718*
25. **Mori D, Ishigami G.** Excavation model of soil sampling device based on particle image velocimetry. *Journal of Terramechanics.* 2015; 62: 19–29. DOI: *https://doi.org/10.1016/j.jterra.2015.02.001*
26. **Gaudel N, de Richter SK, Louvet N, Jenny M, Skali-Lami S.** Granular avalanches down inclined and vibrated planes. *Physical Review E.* 2016; 94: 32904. DOI: *https://doi.org/10.1103/PhysRevE.94.032904*
27. **Shimobayashi SF, Tsudome M, Kurimura T.** Suppression of the coffee-ring effect by sugar-assisted depinning of contact line. *Scientific Reports.* 2018; 8: 17769. DOI: *https://doi.org/10.1038/s41598-018-35998-w*
28. **Comiskey PM, Yarin A, Attinger D.** High-speed video analysis of forward and backward spattered blood droplets. *Forensic Science International.* 2017; 276: 134–141. DOI: *https://doi.org/10.1016/j.forsciint.2017.04.016*
29. **Penn R, Maurer M, Michalec F-G, Scheidegger A, Zhou J, Holzner M.** Quantifying physical disintegration of faeces in sewers: Stochastic model and flow reactor experiments. *Water Research.* 2019; 152: 159–170. DOI: *https://doi.org/10.1016/j.watres.2018.12.037*

TO CITE THIS ARTICLE:
Thielicke W, Sonntag R 2021 Particle Image Velocimetry for MATLAB: Accuracy and enhanced algorithms in PIVlab. *Journal of Open Research Software,* 9: 12. DOI: *https://doi.org/10.5334/jors.334*

Submitted: 13 June 2020 Accepted: 13 May 2021 Published: 31 May 2021

Journal of Open Research Software is a peer-reviewed open access journal published by Ubiquity Press.