Deliverable 1.2

Report 1 on production and diffusion of fluid mechanics images and data

Johan Carlier\textsuperscript{1}
Bern Wienke\textsuperscript{2}

Due date of deliverable: 30 November 2005
Actual submission date: 30 November 2005
Start date of project: 1 December 2004
Duration: 3 years

Project cofunded by the European Commission within the Sixth Framework Programme (2002-2006).

\textsuperscript{1}Cemagref Team, johan.carlier@cemagref.fr
\textsuperscript{2}LaVision, bwieneke@lavision.de
# Contents

1 Introduction .......................................................... 4

2 Experimental data ......................................................... 4

2.1 2D-2C PIV Images .................................................. 5

2.1.1 Wind tunnel ....................................................... 5

2.1.2 Measurements .................................................... 6

2.2 Time-Resolved 2D-2C PIV Images .................................. 7

3 2D-Synthetic Data .......................................................... 8

3.1 Description of Synthetic images Generator ....................... 8

3.2 Viscous flow ........................................................... 10

3.2.1 Poiseuille Flow .................................................. 11

3.2.2 Lamb-Oseen Vortex ............................................. 11

3.3 Potential flows ........................................................ 12

3.3.1 Uniform flow ..................................................... 12

3.3.2 Sink flow .......................................................... 14

3.3.3 Vortex flow ....................................................... 14

3.3.4 Potential flow around a cylinder with circulation .......... 15

3.4 DNS of 2D turbulent flow ........................................... 15

3.4.1 Description of the code ......................................... 15

3.4.2 Flow Behaviour .................................................. 17

3.5 2D-Image from PIV Challenge ....................................... 19

4 3D-synthetic data .......................................................... 21

4.1 Description of Volume Particle Generator ....................... 22

4.2 Description of Synthetic Data ....................................... 27
1 Introduction

In the Fluid project, WorkPackage 1 is devoted to the creation of an image sequence database with controlled 2D/3D experimental and numerical flow fields. The objective of this workpackage is to produce complete sets of data comprising:

- meteorological satellite atmospheric image sequences;
- first sets of image sequences of controlled fluid mechanics experiments (air flows in wind tunnels), produced in a classical way using laser-sheets and cameras, with a smoke or particle concentration chosen a priori in order to give reasonably detectable contrasts;
- second sets of experimental image sequences produced in an adapted manner (seeding, lighting, frame size, physical data range) after feedback from tests conducted in WP 2, WP 3 and WP 4;
- synthetic image sequences coming from the results of Direct Numerical Simulations (DNS) of shear flows;
- quantitative data characterising the flows corresponding to each set of experimental image sequences, among which global and local physical characteristics like velocity fields, statistics, spatial correlation as well as topological dimensions of specific regions of the flow (formation length, layer thickness, virtual origins, wave length, etc).

This report is the deliverable D1.2 on the production and diffusion of fluid mechanics images and data. All the dataset distributed during the first year to the different partners are described. Section 2 concerns two experimental datasets, the first one is not time resolved but the second is. Section 3 presents 2D-synthetic data of analytic flows, DNS of 2D sustained turbulent flow and a sample of 2D flow from PIV Challenge. The last section 4 is devoted to 3D-synthetic data.

2 Experimental data

The first section 2.1 describes a 2D-2C PIV experiment carried out in one of the wind tunnels of the Rennes Regional Centre of Cemagref. The second concerns a Time-Resolved 2D-2C PIV experiment provided by LaVision.

3The first deliverable D1.1 on selection of meteorological image sequence has been already provided by LMD partner
2.1 2D-2C PIV Images

2.1.1 Wind tunnel

The sketch (in front view) of the wind tunnel used in this experiment is presented in Fig. 1. The aeraulic circuit is mainly made of a centrifugal fan, a diffuser, a plenum chamber with honey comb and grids, a convergent with a contraction coefficient of 4 and a testing zone with transparent walls. The testing zone is squared in cross section with 280 mm of with dimensions and 100 cm in length. The upper wall is slightly tilted to suppress the longitudinal pressure gradient. The flow velocity can be chosen continuously between 1 and 8 m. s\(^{-1}\) with a good stability. The uniformity of the velocity profile at the entrance of test section was checked by hot-wire anemometry. The free stream turbulence intensity was less than 0.5%. The temperature was kept within ±0.2\(^\circ\) C by using an air-water heat-exchanger.

The circular cylinder has a length \(L = 280\) mm and a diameter \(D = 12\) mm. It is equipped with 2 thin rectangular end plates. The distance between the end-plates is 240 mm providing an aspect ratio \(L/D = 20\). The clearance between the walls and the end-plates is about 20 mm which is much larger than the thickness of the boundary layer on the wall. The blockage ratio is 4.3%. The circular cylinder is mounted horizontally at 3.5\(D\) from the entrance of the testing zone.

The reference frame is direct and the origin is located at the center of the cylinder. The longitudinal \(x\)-axis is the testing zone axis of the wind tunnel, \(y\)-axis is normal to the wake central plane (symmetry plane) and \(z\)-axis is the cylinder axis.
2.1.2 Measurements

2D2C PIV experiments (2 in-plane velocity components in a plane field) were carried out with NewWave Solo 3 Nd-YAG laser (Energy by pulse of 50 mJ) and a SensiCam PCO cameras (CCD size of 1280 \(\times\) 1024 px\(^2\), pixel size of 6.7 \(\times\) 6.7 \(\mu\)m\(^2\) and dynamics of 12 bits). The laser sheet was produced by means of a Rodenstock telescope and a cylindrical lens. The diameter of the particle seeding (diluted polyglycol in water) is less than 10 \(\mu\)m. The measurement area was located just behind the circular cylinder, in the plane \(z = 0\). The camera was located perpendicularly to the laser sheet at a distance of 36 cm. For the mounted lens, the focal length was 50 mm with an aperture of 5.6. The resulting field of view is about 3.6\(D\) \(\times\) 2.9\(D\). The free-stream velocity was adjusted at 4.8 m.s\(^{-1}\) so that the Reynolds number \(Re_D\) be 3,900 (Reynolds number based on the free stream velocity \(Ue\), the cinematic viscosity \(\nu\) and the diameter of the circular cylinder \(D\)). 5,000 image pairs were obtained with a time interval of 25 \(\mu\)s. Other experiments were carried out in stereoscopic configuration or in the plane \(y/D = 0\) and 1 but have not been distributed for the moment.

![Figure 2: Velocity field of the wake](image)

The PIV records were analyzed by cross-correlation using a Fast-Fourier-Transform algorithm in a multi-grid process with 3 iterations and sub-pixel shift (64 \(\times\) 64, 32 \(\times\) 32 and 16 \(\times\) 16 px\(^2\)) and 50\% overlap. The cross-correlation peaks were fitted with a Gaussian function on 3 pixels. Spurious velocities were identified using a median filter.
and were replaced by a local mean value. To give an idea of the present experiment accuracy, the particle displacement range was about \([0;10]\) px (with an error of 0.3\% on the displacement for this kind of typical PIV method) and the spurious velocities were less than 0.1\%.

100 image pairs were delivered in TIF-format with 16 bits in depth\(^4\). ‘wake/S007\_1\_\*’ is the first set of images, ‘wake/S007\_2\_\*’ is the second set of images and ‘wake/S007\_\*’ contains the corresponding PIV velocity fields. The exact image size is 47.37 mm in length (1280 px) and 37.89 mm in height (1024 px). The location of the lower left corner of the image is \(x = 6.74\) mm and \(y = -19.27\) mm. Figure 2 presents an example of velocity field.

### 2.2 Time-Resolved 2D-2C PIV Images

The images show a slightly turbulent air flow with small water droplets with a diameter of about 5 \(\mu\)m. The sequence of 100 images has been recorded with a Photron APX camera with \(1024 \times 1024\) px\(^2\) running at 1000 hz. The laser used was a NewWave Pegasus with a double-rod with a power of \(2 \times 10\) mJ running at 1000 hz. The field of view is about 7 cm of with dimensions. This sequence is delivered in TIF-format with a dynamics of 8 bits.

The sequence has been selected to show typical problems in PIV recordings. The seeding density is quite inhomogeneous even with regions without any seeding particles (see top-right of figure 3). Here, some sort of automatic masking is required to compute vectors only where there is enough signal. In addition the camera has some artifacts and noise. In general the image quality is quite typical for PIV. It is good enough to compute reliable vector fields.

Since one does not know the ‘true’ solution, the results of any algorithm can only be compared at least visually to check how it is able to cope with varying degrees of seeding density and gradients. Advanced algorithms might be able to use the information of many images instead of only two in order to compute a more reliable and more accurate displacement field for all times.

File ‘Movie\_of\_Vorticity.avi’ shows the results by LaVision PIV-algorithm with relatively large interrogation windows of \(64 \times 64\) px\(^2\) with 75\% overlap, which makes the vector field smooth and stable, but with somewhat less spatial resolution. The correlation has been performed between two successive images. The quite low fluctuations in the shown

\(^4\)With classical viewer, the images may appear relatively black and seem to be poorly enlighted. In fact, the original images are 12 bits in depth and are converted for more convenience in TIF-format with 16 bits in depth, giving artificially low values as compared to the global 16 bits range.
vorticity field from vector file to vector file indicate the accuracy of the used PIV correlation algorithm. These time fluctuations are a good indicator for real flows when one does not know the true solution.

Figure 3: At left, image sample of the sequence. At right, corresponding velocity and vorticity field.

3 2D-Synthetic Data

The first section (3.1) describes the home-made software used to generate sets of synthetic images. The three sets are the viscous flows (3.2), the potential flows (3.3) and 2D turbulent flow (3.4) provided by the Cemagref team. The dataset of the last section (3.5) was proposed by LaVision and was derived from a previous set used for the PIV Challenge 2005.

3.1 Description of Synthetic images Generator

An initial field of particles was generated with an uniform random distribution of the coordinates \((x, y, z)\) in a 3D domain. The size of the domain is given in meter and pixel for the width and the height and only in meter for the thickness. The figure 4 presents the 3D domain containing the light sheet and the particles. The origin is located at the center of the domain.

The particle concentration is obtained through the choice of a fixed number of particles which are located in the domain by randomly assigning a \((x, y, z)\) position to each particle.
Each particle is given a diameter. The random distribution of this diameter is normal with a given mean value and standard deviation. Each particle image is given a shape using a 2D Gaussian repartition of the intensity over the neighbouring pixels of particle central position \((x, y, z)\):

\[
I_k(x) = I(x) \exp^{-\frac{1}{2} \left( \frac{(x-x_k)^2 + (y-y_k)^2}{D_k^2} \right)}
\]  

The peak intensity of each particle image depends on the position in the light sheet. In \(x\) and \(y\) direction, the light sheet intensity is constant. In \(z\) direction, it has a Gaussian profile centered on the median plane with maximum value of grey-levels \(q\):

\[
I(x) = q \exp^{-\frac{1}{2} \left( \frac{4z^2}{w^2} \right)}
\]

The thickness of the domain corresponds to 4 standard deviations for this Gaussian profile. Moreover, a constant background can be added.

In all the initial images, the particle image concentration was set homogeneous by fixing the number of particles in each window (the image was divided in windows). The concentration value was chosen very large in the case of gaussian profile in thickness because particles located far from the center of the laser sheet result ‘not-visible’ due to their very low intensity.

The digitalization was done by adding the contribution of the integration of each particle on the pixel:

\[
I(x_i, y_i) = \sum_{k}^{N} I_k(x, y) \otimes \Pi(x, y) \otimes \delta(x - x_i, y - y_i)
\]

where \(\otimes\) is the convolution operator, \(\Pi\) the rectangular function and \(\delta\) the Dirac function.
Each field of particles was computed by applying an ‘analytic’ velocity field to the previous one, a time interval being adequately chosen between the successive images.

3.2 Viscous flow

In some simple viscous flows, it is possible to find the exact solution of the Navier-Stokes equations. Poiseuille and Lamb-Oseen Vortex flows are two well known examples with velocity gradients.

For each of those 2 viscous flows, 41 successive images were given in TIF-format with 8 bits in depth. The same parameters (particle size, concentration, intensity, etc) were used for all the initial images:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width of the image (in pixel):</td>
<td>1024</td>
</tr>
<tr>
<td>Height of the image (in pixel):</td>
<td>1024</td>
</tr>
<tr>
<td>Width of the image (in meter):</td>
<td>1.</td>
</tr>
<tr>
<td>Height of the image (in meter):</td>
<td>1.</td>
</tr>
<tr>
<td>Time interval between shoots (in second):</td>
<td>1.</td>
</tr>
<tr>
<td>Thickness of the laser sheet (in meter):</td>
<td>.001</td>
</tr>
<tr>
<td>Width of the window (in pixel):</td>
<td>32</td>
</tr>
<tr>
<td>Height of the window (in pixel):</td>
<td>32</td>
</tr>
<tr>
<td>Number of particles per window (integer):</td>
<td>256</td>
</tr>
<tr>
<td>Diameter and deviation of the particle images (in pixel):</td>
<td>1.5 .25</td>
</tr>
<tr>
<td>Minimum grey-level in the image (integer):</td>
<td>16</td>
</tr>
<tr>
<td>Light scattering at the center (integer):</td>
<td>64</td>
</tr>
</tbody>
</table>

The corresponding sequences were placed into the following folders:

Poiseuille: ‘Poiseuille/S002.*’

Lamb-Oseen: ‘Lamb-Oseen/S003.*’

One sequence describes the trajectory of the particles along 41 timesteps, the trajectory being time independant here. Different computations are possible like analysing a sequence in sequential mode (0-1;1-2;2-3...), analysing a sequence with a stride (0-2;1-3;2-4...) in order to increase the displacements, or reversing a sequence in order to change the velocity sign (i.e. change the rotation of a vortex for exemple). Figure 5 presents the velocity fields for the viscous flows.
3.2.1 Poiseuille Flow

The Poiseuille flow is a viscous flow between two parallel plates with a constant streamwise pressure gradient. The plates are located at the bottom and the top borders of the image. The equation for the permanent velocity field is:

$$u_x = U \left(1 - \frac{y^2}{h^2}\right)$$  \hspace{1cm} (4)

$$u_y = 0$$  \hspace{1cm} (5)

$$u_z = 0$$  \hspace{1cm} (6)

with $U = 1/64$ m.s$^{-1}$, $h = 0.5$ m and $y \in [-0.5, +0.5]$ m

3.2.2 Lamb-Oseen Vortex

The Lamb-Oseen vortex is a two-dimensional viscous flow with circular streamlines and a decreasing vorticity along the radial distance $r$ and the time $t$. The exact solution of the Navier-Stokes equations is:

$$u_\theta (r, \theta, t) = \frac{\Gamma_0}{2\pi \nu} \left(1 - e^{-r^2/4\nu t}\right)$$  \hspace{1cm} (7)

where $\nu$ is the cinematic viscosity and $\Gamma_0$ is the initial circulation. In the present work, $\Gamma_0 = 0.05$ m$^2$.s$^{-1}$ and $t$ is fixed with $4\nu t = (1/6)^2$ m$^2$ ($\sqrt{4\nu t}$ can be seen as the radius of the Lamb-Oseen Vortex).
3.3 Potential flows

In a 2D incompressible ($\nabla \cdot \mathbf{u} = 0$), irrotational fluid flow ($\nabla \times \mathbf{u} = 0$), it is possible to introduce an analytic function of the complex variable $z = x + iy$ for which the real and imaginary parts correspond to the velocity potential and the stream function:

$$f(z) = \phi(z) + i\psi(z) \quad (8)$$

The components of the velocity can be derived from the stream function derivatives:

- in cartesian coordinates:
  $$u_x = +\frac{\partial \psi}{\partial y} \quad (9)$$
  $$u_y = -\frac{\partial \psi}{\partial x} \quad (10)$$

- in polar coordinates:
  $$u_r = +\frac{1}{r} \frac{\partial \psi}{\partial \theta} \quad (11)$$
  $$u_\theta = -\frac{\partial \psi}{\partial r} \quad (12)$$

The potential flow provided are an uniform flow, a sink, a vortex and a potential flow around a cylinder with circulation. Their corresponding velocity field solutions are presented on figure 6.

The strategy and image parameters used for the image generation were the same as for the previous image sets with 41 successive images given in TIF-format with 8 bits in depth. The corresponding sequences were placed into the following folder:

**Uniform:** `uniform/S001_*`

**Sink:** `sink/S004_*`

**Vortex:** `vortex/S005_*`

**Cylinder:** `rotcyl/S006_*`

3.3.1 Uniform flow

An uniform flow can be seen as a potential flow. The analytic function of the complex variable for a such flow is:

$$f(z) = Uz \quad (13)$$
with \( U = 1/128 \text{ m.s}^{-1} \) in our case. However, it is more convenient to write the equation for the displacement field between successive images as:

\[
\begin{align*}
\Delta x &= 8 \text{ px} \\
\Delta y &= 8 \text{ px} \\
\Delta z &= 0 \text{ px}
\end{align*}
\]
3.3.2 Sink flow

The analytic function of the complex variable is:

\[ f (z) = \frac{D}{2\pi} \log z \quad (17) \]

The imaginary part is the stream function with:

\[ \psi = \frac{D}{2\pi} \theta \quad (18) \]

The stream function allows to get the velocity field in polar coordinates with:

\[ u_r = \frac{\partial r}{\partial t} = \frac{D}{2\pi r} \quad (19) \]
\[ u_\theta = \frac{1}{r} \frac{\partial \theta}{\partial t} = 0 \quad (20) \]

This system of Ordinary Differential Equations can be integrated easily to get the Lagrangian representation of the particle coordinate evolutions:

\[ r^2 = r_0^2 + \frac{D}{2\pi} (t - t_0) \quad (21) \]
\[ \theta = \theta_0 \quad (22) \]

with the flow rate \( D = -0.01 \text{ m}^2\cdot\text{s}^{-1} \) in the provided case.

3.3.3 Vortex flow

The analytic function of the complex variable is:

\[ f (z) = -i \frac{\Gamma}{2\pi} \log z \quad (23) \]

The imaginary part is the stream function with:

\[ \psi = -\frac{\Gamma}{2\pi} \log r \quad (24) \]

The stream function allows to get the velocity field in polar coordinates with:

\[ u_r = \frac{\partial r}{\partial t} = 0 \quad (25) \]
\[ u_\theta = \frac{1}{r} \frac{\partial \theta}{\partial t} = \frac{\Gamma}{2\pi r} \quad (26) \]

The evolution of particle coordinates is:

\[ r = r_0 \quad (27) \]
\[ \theta = \theta_0 + \frac{\Gamma}{2\pi r} \quad (28) \]

with the circulation \( \Gamma = -0.01 \text{ m}^2\cdot\text{s}^{-1} \) in the provided case.
3.3.4 Potential flow around a cylinder with circulation

This flow is the superposition of a uniform flow, a doublet (not defined here) and a vortex. The analytic function of the complex variable is:

\[ f(z) = Uz - i \frac{\Gamma}{2\pi} \log z + \frac{K}{2\pi z} \]  

(29)

The corresponding system of Ordinary Differential Equations is:

\[ \frac{\partial r}{\partial t} = +U \left( 1 - \frac{R^2}{r^2} \right) \cos \theta + 0. \]  

(30)

\[ \frac{\partial \theta}{\partial t} = -\frac{U}{r} \left( 1 + \frac{R^2}{r^2} \right) \sin \theta + \frac{\Gamma}{2\pi r^2} \]  

(31)

This system was integrated by a Runge-Kutta algorithm with \( U = 1/128 \text{ m.s}^{-1}, \) \( R = 1/6 \text{ m}, \) \( \beta = \pi/4, \) \( K = 2\pi R^2 U \) and \( \Gamma = 4\pi RU \sin \beta. \) \( U \) is the external velocity, \( R \) the radius of the cylinder and \( \beta \) the angle defining the positions of the two stagnation points in the cylinder.

3.4 DNS of 2D turbulent flow

The main interest of a synthetic image sequence database of numerically generated 2D turbulent flow fields is the knowledge of the exact solution along the time for flow with a large scale range in the energy spectrum. For this purpose, DNS was performed to solve particle trajectories in 2D sustained turbulent flow.

3.4.1 Description of the code

The code used for this study solved the Navier-Stokes equations:

\[ \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} \]  

(32)

with the incompressible condition:

\[ \nabla \cdot \mathbf{u} = 0 \]  

(33)

and the Lagrangian equation for non heavy particle transported by the flow:

\[ \frac{d\mathbf{x}}{dt} = \mathbf{u} \]  

(34)

By applying the \( \nabla \) operator to the Navier-Stokes equations in 2D, the conservation equation for the vorticity \( \omega \) is obtained:

\[ \frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} = \nu \nabla^2 \omega \]  

(35)
The vorticity $\omega$ is computed at each time step in Fourier space. Then, each Fourier coefficient is governed by the following equation:

$$\frac{\partial \hat{\omega}_{\alpha\beta}}{\partial t} - T_{\omega_{\alpha\beta}} = -\frac{(\alpha^2 + \beta^2)}{Re} \hat{\omega}_{\alpha\beta}$$

(36)

with

$$T_{\omega_{\alpha\beta}} = -i \sum_{(\alpha_1,\beta_1)+(\alpha_2,\beta_2)=(\alpha,\beta)} (\alpha_2 \hat{u}_{\alpha_1 \beta_1} + \beta_2 \hat{v}_{\alpha_1 \beta_1}) \omega_{\alpha_2 \beta_2}$$

(37)

Both the vorticity equation and the Lagrangian equations for particles $i = 1, N$ give the whole differential system:

$$\frac{\partial \left( \hat{\omega}_{\alpha\beta} \exp \left( \frac{(\alpha^2 + \beta^2)}{Re} t \right) \right)}{\partial t} = T_{\omega_{\alpha\beta}} \exp \left( \frac{(\alpha^2 + \beta^2)}{Re} t \right)$$

(38)

$$\frac{\partial x_i}{\partial t} = u$$

(39)

$$\frac{\partial y_i}{\partial t} = v$$

(40)

The vorticity equation is solved in Fourier space using dealiased Fourier expansions in two directions with periodic boundary conditions. The time integration is third-order/three steps with a Runge-Kutta scheme. $T_{\omega_{\alpha\beta}}$ is computed in physical space to reduce the time computation. The code is the called pseudo-spectral.

$x_i$ and $y_i$ are solved in the physical space. The computation is done in parallel with the vorticity equation using the same Runge-Kutta scheme in order to use the intermediate velocity components $u$ and $v$ determined in $T_{\omega_{\alpha\beta}}$. However, an interpolation (bicubic) of the velocity field has to be applied because there is no reason for the particles to be located on the nodes of the grid.

For the Fluid project, sustaining turbulence along the time is of great interest to allow time averaging of statistical quantities in the post-processing. Then, the vorticity equation is slightly modified:

$$\frac{\partial \hat{\omega}_{\alpha\beta}}{\partial t} - T_{\omega_{\alpha\beta}} = -\frac{(\alpha^2 + \beta^2 + Re_{\nu})}{Re} \hat{\omega}_{\alpha\beta} + f$$

(41)

where $f$ is a forcing term proportionnal to the vorticity and random on the Fourier coefficients, and $Re_{\nu}$ a friction term adding only for the first modes. That results an enstrophy injection dissipated by small scale with $Re$ and an energy injection dissipated by large scales with $Re_{\nu}$ (see figure 7).
3.4.2 Flow Behaviour

Different sets of parameters have been tested. The strategy was to select the maximum number of nodes for a basic computer, the maximum $Re$ for this grid size and a target Energy/Enstrophy injection which can be dissipated by the system. Finally, the selected parameters are summarized here:
Figure 8 presents the time evolution of the energy and enstrophy. The turbulence is well sustained after few time steps. Note that the proportion of random/proportional forcing is useful for adjusting the life time of individual vortical structures. The observation clearly shows that the larger the proportion of random forcing is, the shorter life times are.

Figure 9 is an instantaneous Vorticity field in which vortical structures interact with each other.

The files containing the particle locations along the time have been generated. Some synthetic particle images have been already created but some tests need to be done before proposing a new dataset in the fluid project (delivery date: February 2006).
3.5 2D-Image from PIV Challenge

This image pair is used in the current PIV-Challenge 2005 (case A4). It has been designed by LaVision & TU-Delft in a way to analyse the capability of PIV algorithms to cope with different degrees of local flow gradients without and with varying degrees of image noise. The purpose was to design a single image which shows immediately in a visual as well as quantitative way the strengths and weaknesses of a particular algorithm.

Files supplied ‘P4_1’ and ‘P4_2’ are the corresponding first and second image in uncompressed 8/16 bits grey-level format. Suffix ‘8/16-bit’ denotes that the TIF-format are stored as 8/16 bits. The 16 bits data has intensities of up to 4096 (i.e. 12 bits). While TIF was defined originally only in 8 bits format, there are a lot of TIF-headers available which can read 16 bits format. It is expected that there will be little difference in the results between 8 and 16 bits. Figure 10 shows the first image.
Figure 10: First image

File `A4reference.txt` contains the true solution in an ASCII-format similar to Tecplot-format with one vector for every pixel. File `P4LAVIS.dat` contains the best PIV-LaVision evaluation in the same format. Figure 11 shows the vertical $v$-component of the best result of LaVision. The vectors are spaced by 4 px, so even very small wavelengths can be resolved. Note that 4 px is about the average spacing between the seeding particles. The data is simply stored as one vector each line in the format $x$, $y$, $u$, $v$ and $flag$:

```
3.00000 3.00000 0.00000 0.00000 0
5.00000 3.00000 0.00000 0.00000 0
7.00000 3.00000 0.01114 0.02144 3
9.00000 3.00000 0.01266 0.01948 3
...
```

The $flag$ (only for the best PIV evaluation) is zero when no vector is computed. The coordinate system starts at the bottom-left with the $x$-axis to the right and the $y$-axis up. The first pixel is at $(0, 0)$ px. All displacements are given in units of pixel.

The top left quadrant shows boundary layer flows. Close to the walls there is a high shear region with a thickness of from just a few pixel (difficult) up to $20 - 30$ pixel (easy).
The right side of the image is divided into 5 sections with different degrees of noise. The top one has no noise. The flow is purely a sinusoid of the vertical flow component with a slightly varying amplitude of about $\pm 3$ px and a wavelength which is large on the left and getting smaller down to less than 16 px to the right. Here the standard PIV algorithms with $e.g. \ 32 \times 32$ px$^2$ interrogation windows will fail, while optical flow might perform better. Of course, the limiting factor is also the seeding density.

The second and third section has increasingly higher image noise. The forth section has very low particle intensity. Especially with the 8 bits images there will be probably quite some degradation in the results. The lowest section has a fixed wavelength of 60 px, and a varying seeding density and some fixed amount of noise. The bottom left quadrant has different size 2D-vorticies to check how the algorithms can cope with very small 2D-gradients.

4 3D-synthetic data

3D-Synthetic Data is provided by LaVision.
4.1 Description of Volume Particle Generator

The purpose for generating particle volumes and from it projected camera images is many-fold. First it can serve as a test for Stereo-PIV or 2D/3D-PTV algorithms, but the main goal is to test new tomographic reconstruction algorithms to measure the flow in a complete volume.

Recorded camera images are scaled in pixels with the top-left pixel located at (0, 0). More accurately it is the middle of the pixel which has this position. The $y$-axis is pointing downwards (see figure 12). Camera coordinates are written as $(x_i, y_i)$ with $i = 1 - 4$ for camera 1 - 4.

Volumes are composed of voxels and the origin $(x_{\text{voxel}}, y_{\text{voxel}}, z_{\text{voxel}}) = (0, 0, 0)$ vx is at the front-top-left with $x$-axis to the right, $y$-axis down, and $z$-axis from front to back (right-handed coordinate system, see figure 13). Again more accurately it is the middle of the voxel which has this position. For example, below is given a volume of $nx \times ny \times nz = 513 \times 513 \times 33$ vox. The voxel indices run from 0 - 512 for $x$ and $y$, and from 0 to 32 for $z$.

More important is the scaled world coordinate system in millimeter: $(X_W, Y_W, Z_W)$. Per definition, 1 pixel or voxel is set to 1 mm to make things easier. The origin of the world coordinate system is in the middle of the volume at:

$$ (X_{W0}, Y_{W0}, Z_{W0}) = (0, 0, 0) \text{ mm} \quad (42) $$

$$ = (x_{\text{voxel}0}, y_{\text{voxel}0}, z_{\text{voxel}0}) \quad (43) $$

$$ = (256, 256, 16) \quad (44) $$

The $y$-axis is pointing upwards, the $z$-axis to the back, away from the cameras (left-handed coordinate system). The origin of the world coordinate system is always at an integer voxel position. For example, if $nx = 513$, the origin is at index 256, if $nx = 512$, then it will be 256. So the origin is at integer value $nx/2$. The conversion between voxel index and world coordinate in millimeter is therefore given by:

$$ (x_{\text{voxel}0}, y_{\text{voxel}0}, z_{\text{voxel}0}) = (\text{int}(nx/2), \text{int}(ny/2), \text{int}(nz/2)) \quad (45) $$

with $nx$, $ny$, $nz$ the size of volume and

$$ (X_W, Y_W, Z_W) = (x_{\text{voxel}} - x_{\text{voxel}0}, -y_{\text{voxel}} + y_{\text{voxel}0}, z_{\text{voxel}} - z_{\text{voxel}0}). \quad (46) $$

Finally, there is a transformation $M_i$ between world coordinate system and camera pixel location, also called calibration mapping function:

$$ (x_i, y_i) = M_i (X_W, Y_W, Z_W) \quad (47) $$
with $i = 1 - 4$ for camera $1 - 4$.

In the following, a very simple setup of the four cameras is used (see figure 14). For example, camera 1: $Y_W = 0$ at 45° to the left, camera 2: $Y_W = 0$ at 45° to the right, camera 3: $X_W = 0$ at 45° to the top, and camera 4 is at the middle $X_W = Y_W = 0$ looking perpendicular onto the volume.

When generating camera images, the projected volume is shifted by 100 px into the middle of the image, whose size is set to $(nx + 200, ny + 200)$. This avoids boundary effects due to missing particles. There is also no perspective distortion included, i.e. the
cameras are at $Z_W = -\infty$. Then, the transformations $M_i$ are very simple:

- **camera 1**, $e.g. \alpha_y = -45^\circ$

  $$(x_1, y_1) = M_1 (X_W, Y_W, Z_W)$$
  $$= (X_W \cos \alpha_y + Z_W \sin \alpha_y + x_{voxel0} + 100, -Y_W + y_{voxel0} + 100)$$

- **camera 2**, $e.g. \alpha_y = +45^\circ$

  $$(x_2, y_2) = M_2 (X_W, Y_W, Z_W)$$
  $$= (X_W \cos \alpha_y + Z_W \sin \alpha_y + x_{voxel0} + 100, -Y_W + y_{voxel0} + 100)$$

- **camera 3**, $e.g. \alpha_x = -45^\circ$

  $$(x_3, y_3) = M_3 (X_W, Y_W, Z_W)$$
  $$= (X_W + x_{voxel0} + 100, -Y_W \cos \alpha_y + Z_W \sin \alpha_y + y_{voxel0} + 100)$$

- **camera 4**, $e.g. \alpha_x = \alpha_y = 0$

  $$(x_4, y_4) = M_4 (X_W, Y_W, Z_W)$$
  $$= (X_W + x_{voxel0} + 100, -Y_W + y_{voxel0} + 100)$$
The origin \((X_{W0}, Y_{W0}, Z_{W0}) = (0, 0, 0)\) mm is always projected to the pixel position \((x_{voxel0} + 100, y_{voxel0} + 100)\) for all cameras. The figure 15 shows how particle images look like with \(nx = 513, ny = 513\) and \(nz = 33\).

The procedure of generating particles in a specified volume is done twice for time \(t_0\) and \(t_1\) with a displacement field in between. Particles are generated with a gaussian intensity profile in 3D with a specified size and peak height. Typically the particle size is set to 2 px with a maximum height of 100 grey-level which provide very good condition for particle detection schemes or PIV processing (no peak locking). So far a few displacement
functions \((u, v, w)\) as a function of \((X_W, Y_W, Z_W)\) have been integrated, modes 0 – 4:

- **mode 0**, constant velocity \(u_0, v_0, w_0\) everywhere

\[
\begin{align*}
    u &= u_0 \\
    v &= v_0 \\
    w &= w_0
\end{align*}
\]

- **mode 1**, compression wave in \(u\) with wavelength \(L\)

\[
\begin{align*}
    u &= u_0 \sin \left( \frac{2\pi X_W}{L} \right) \\
    v &= v_0 \\
    w &= w_0
\end{align*}
\]

- **mode 2**, compression wave in \(u\) and gradient in \(z\) (at front, full amplitude \(-u_0\); in the middle \((Z_W = 0)\), 0; at back, \(+u_0\))

\[
\begin{align*}
    u &= u_0 \sin \left( \frac{2\pi X_W}{L} \right) \left( \frac{Z_W}{D_z} \right) \\
    v &= v_0 \\
    w &= w_0
\end{align*}
\]

- **mode 3**, shear flow in \(v\) with wavelength \(L\)

\[
\begin{align*}
    u &= u_0 \\
    v &= v_0 \sin \left( \frac{2\pi X_W}{L} \right) \\
    w &= w_0
\end{align*}
\]

- **mode 4**, shear flow in \(v\) and gradient in \(z\) (at front, full amplitude \(-v_0\); in the middle \((Z_W = 0)\), 0; at back, \(+v_0\))

\[
\begin{align*}
    u &= u_0 \\
    v &= v_0 \sin \left( \frac{2\pi X_W}{L} \right) \left( \frac{Z_W}{D_z} \right) \\
    w &= w_0
\end{align*}
\]
with $D_z = \text{int}(nz/2) = \text{half thickness of volume (distance from origin to rim)}$

The number of particles generated is an extra parameter specified usually in units of ppv (particles per voxel) or ppp (particles per pixel) in the camera image. Typically, one uses up to 0.03 to 0.1 ppp to avoid too many overlapping particles in the image. With 0.1 ppp more than 50% of the particles are overlapping since each particle covers an average pixel area of $3 \times 3 \text{ px}^2$. With 0.03 ppp, a $1024 \times 1024 \text{ px}^2$ camera observes 30 000 particles. For comparison, state-of-the-art PTV algorithms typically only work for up to a few thousand particles reliable.

When generating particles in the two volumes at $t_0$ and $t_1$, the particle in the first volume is displaced by $(-u/2, -v/2, -w/2)$ and in the second volume by $(+u/2, +v/2, +w/2)$ in order to have symmetrical displacements. All particle locations and displacements are stored in an ASCII-File (`ParticlePosition.txt') with one particle per line in the format: $X_W, Y_W, Z_W, u, v, w, \text{ size and height}$ where the position of the particle in the first volume is $(X_W - u/2, Y_W - v/2, Z_W - w/2)$ and in the second volume $(X_W + u/2, Y_W + v/2, Z_W + w/2)$. Using the mapping functions $M_i$, one can easily compute the position of the particles in the camera images.

### 4.2 Description of Synthetic Data

This dataset has been produced by the previous Volume Particle Generator. It contains synthetically generated volumes with seeding particles. The characteristics are summarized in the following table:

<table>
<thead>
<tr>
<th>Volume size in voxel:</th>
<th>$512 \times 513 \times 33$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particle density in ppv:</td>
<td>0.001</td>
</tr>
<tr>
<td>Particle size in pixel:</td>
<td>2</td>
</tr>
<tr>
<td>Particle center intensity:</td>
<td>100</td>
</tr>
</tbody>
</table>

The File `Volume.rar’ contains $2 \times 33$ 8 bits TIF-file: ‘Volume_A_001.TIF’ to Volume_A_032.TIF’ for the first volume, and ‘Volume_B_001.TIF’ to Volume_B_032.TIF’ for the second.

The particles move between the first volume A and the second volume B as the mode 3 with $L = 64 \text{ px}$ and $v_0 = 2 \text{ px}$. This is a simple vertical shear flow with no $y$- and $z$-dependance. $(X_W, Y_W, Z_W)$ is the world coordinate system, as described in detail previously. The origin of the world coordinate system is in the middle of the volume.

File ‘ParticlePosition.txt’ contains the position of all particles in an ASCII-format with one particle per line. A header describes the file. The position and displacement are given in world coordinates in millimeter (Note that 1 mm = 1 px).
File ‘Vectors_true_grid=1-16.txt’ contains the velocity field in an ASCII-format. ‘Grid’ means that one velocity value is given for each pixel. Again the position of the vectors and the displacements are given in world coordinates. The displacements given are not averages over some area (e.g. 4 x 4 for ‘grid=4’), but they are the exact displacement at that position. The same is true for any computed velocity fields later. So obviously, ‘grid=4’ is undersampling the flow field and does not provide an absolute accurate estimate of the real flow.

File ‘Projected Images 1-4A/B.tif’ contains the images of volume A and B projected onto the 4 cameras. The location of the cameras and the mapping function from world coordinate to camera pixel is described in the section 4.1. Suffix 1 – 4 relate to the four cameras and A and B relate to frame A and B. The data is given as 16 bits TIF-format. Figures 17 are the images for camera 1 – 4.

4.3 Data Processing

There are different possibilities, how to use the provided data in order to compute a final volume flow field:

- Assume that by some experimental technique one is able to record a set of volume data directly, and then compute from the recorded volume the full 3D-velocity field. An example would be e.g. a tomographically sliced volume by X-ray tomography or a CT-scan. Then the two volumes recorded at two times can be processed by different algorithms e.g. by:
  - 3D-particle location in the volume & matching (kind of 3D-PTV);
  - volume 3D-correlation;
Standard 3D-PTV: compute 2D particle locations in the recorded camera images, compute the 3D-position of the particles by stereoscopic triangulation, then match corresponding particles in the two volumes in order to compute the full 3D-flow field at the location of the particles. Often one is using a series of images in time, which makes the matching (and 3D-positioning) more robust. Note that the seeding density is already quite high, which makes 3D-PTV quite erroneous due to overlapping particles.

At first, reconstruct the volume from the projected camera images by tomographic reconstruction. Then, compute from the volume the final 3D-flow field according to the methods of the first item. The reconstruction can be done in different ways.
LaVision implements a modified MART-algorithms which works quite well for low seeding densities. However, there is still a lot of possible improvement.

These possibilities are summarised in the figure 18.
The provided volume data is designed to be extremely simple and of perfect image quality, so that it should be easy to get started. Depending on the demand one can easily create more complicated flow fields, much bigger and thicker volumes, and higher seeding densities, added image noise, and so on.

This data serves as a reference where all information is known and a good flow field computation should be possible with different algorithms. One might even use standard Stereo-PIV which also gives acceptable results, but the oblique viewing of the cameras produces already quite some errors.

Last not least it can not be stressed enough, that this kind of tomographic reconstruction approach might be the new upcoming PIV method to record full volume 3D-flow fields digitally even with time resolution when high-speed cameras are used. This would be the final goal of almost all flow researchers!