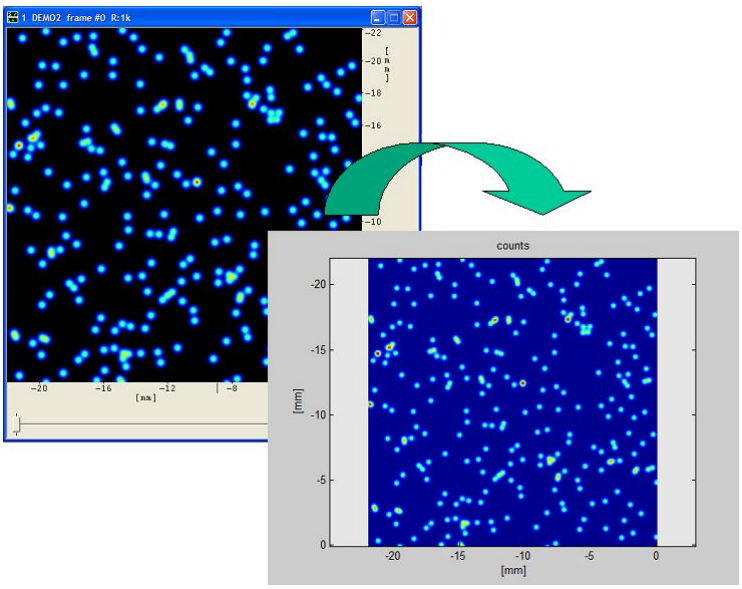
The *ReadIMX Loader* package for Matlab©

by LaVision



Manual for *ReadIMX Loader* package
October 31, 2005
LaVision GmbH, Anna-Vandenhoeck-Ring 19, D-37081 Göttingen
Produced by LaVision GmbH, Göttingen
Printed in Germany

Table of Contents

1 Introduction

1.1 About this manual

This manual helps to install and to get started with the *LaVision ReadIMX Loader* package. It describes how to use the loader and gives detailed information for using *DaVis* image files in Matlab.

1.2 Overview

The *ReadIMX Loader* package provides a dynamic link libary for loading *DaVis* image and vector files into the Matlab workspace. The package requires a Matlab version 5.2 or higher. The DLL returns an image structure, which contains the raw image data (intensities, vector components) and all assigned parameter (sizes, types attributes). An additional function allows to display the structures like in *DaVis* . Supported *DaVis* file formats are: IMX/IMG/VEC and IM7/VC7. Not supported are sparse vector files and files with *valid planes* (*DaVis* 7.2).

1.3 Download and installation

1. Download the latest *ReadIMX Loader* package (file `readimx4matlab_vX.X.zip`) from the download section of the *LaVision* homepage: www.lavision.de.

2. Extract the ZIP file in an installation folder on your PC (e.g. `C:\Programme\MATLAB704\toolbox\matlab\readimx`).

3. Start the Matlab programm and add the installation folder to the Matlab search path. For our example, use the command:

   ```
   >>addpath C:\Programme\MATLAB704\toolbox\matlab\readimx
   ```

4. Test the installation by the command:

   ```
   >>help readimx
   ```

   This command should print the readimx help text on the Matlab console.

5. **Note:** Already existing files readimx.dll, readimx.m and showimx.m need to be removed from the folders of the Matlab path.

1.4 The *readimx* function

After the correct installation, the *readimx* function is availbale in your Matlab program. A sample invocantion of the function is given by:

```
>>A=readimx('images/demo1.imx');
```

The single argument of *readimx* needs to be a string of an existing *DaVis* image file path. Supported *DaVis* file formats are IMX/IMG/VEC and IM7/VC7. Sparse buffer file are not supported. On success, the function returns an image structure A with the format:

```
A =
          DaVis: 55
         Source: 'images/demo1.imx'
           Data: [384x286 uint16]
       PackType: 19
          IType: 0
           Grid: 1
             Nx: 384
             Ny: 286
             Nz: 1
             Nf: 1
         ScaleX: [2x1 double]
         ScaleY: [2x1 double]
         ScaleI: [2x1 double]
          UnitX: 'cm'
          UnitY: 'cm'
          UnitI: ''
         LabelX: 'distance'
         LabelY: 'height'
         LabelI: 'counts'
       Comment1: 'Aceton Tracer LIF for a turbulent flow'
       Comment2: 'created with WinSC 4.1'
           Date: '25.01.95'
           Time: '14:43:31'
     Attributes: [1x879 char]
```

Table 1.1 gives a overview of the structure fields.

Field	Format	Description	Notes
DaVis	Scalar	The DaVis file format number	e.g. 43,55,70
Source	String	The source file path	e.g. `'images/demo1.imx'`
Data	array	The raw image data	see below
PackType	Scalar	The pack type number	for internal use only
IType	Scalar	Format numbers for the data field	e.g. 0=image, see below
Grid	Scalar	The grid spacing of vector formats	
Nx	Scalar	The display width	
Ny	Scalar	The display height	
Nz	Scalar	The display depth	
Nf	Scalar	The number of image frame	
ScaleX	array	Linear column scaling parameter [a,b]	$x = a \cdot x_i + b$ at pixel [i,j]
ScaleY	array	Linear row scaling parameter [a,b]	$y = a \cdot y_i + b$ at pixel [i,j]
ScaleI	array	Linear intensity scaling parameter [a,b]	$I = a \cdot I_{i,j} + b$ at pixel [i,j]
UnitX	string	Unit for columns	
UnitY	string	Unit for rows	
UnitI	string	Unit for intensities	
LabelX	string	Label for columns	
LabelY	string	Label for rows	
LabelI	string	Label for intensities	
Comment1	string	1. comment string	
Comment2	string	2. comment string	
Date	string	Acquisition date	format dd.mm.yy
Time	string	Acquisition time	format hh:mm:ss
Attributes	string	list of attributes	see below

Tab. 1.1: Image structure fields

1.5 The Data field

In this section you will find instrcutions of how to use the **Data** field of the image structure. A *DaVis* image file contains the image data as well as format parameter (i.e. size, frames, types) and image attributes (i.e. used cameras, comments, recording parameter). The format parameter will be used for the interpretion of the data and the attributes gives additional informations of the images. The main format parameter is the scalar **IType** field, its interpretion is described in this section. The **Data** field is a 2-dimensional array of image data divided into equal blocks. The blocks sizes are given by the fields **Nx** and **Ny**. The contents of the blocks depend on the value of the **IType** field.

- $IType = 0$
  In this case the **Data** field contains the data for a multi-frame or multi-plane grayvalue image. Each block contain one image frame/plane. The number of frames/planes is given in the field **Nf**/**Nz**. See figure 1.1.

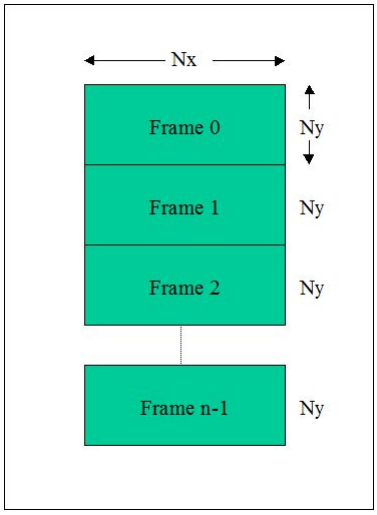


Fig. 1.1: $IType = 0$

- $IType = 1$ **or** $IType = 3$
  In this cases the **Data** field contains the components of 2D vectors from by a PIV calculation. The first block contains an array of "best choice" indices. Valid choice values i are in $0 \cdots 5$, whereas $0 \cdots 3$ are results from the vector calculation and $4, 5$ are from a vector postprocessing (interpolation, filling). The next 8 blocks contain consecutively the vector components v_x^i, v_y^i for $0 \cdots 3$. The blocks 8 and 9 contain also the

components from postprocessing. In case of $IType = 3$, the block 10 contains the *peak ratio* values form the vector calculation. The precision of the data may be *uint8, uint16, single* or *double* (Matlab convention). See figure 1.2.

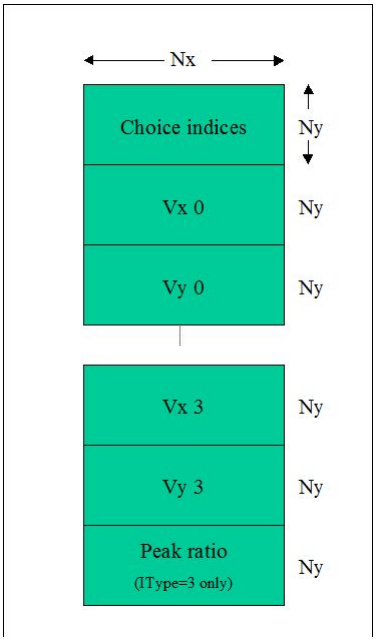
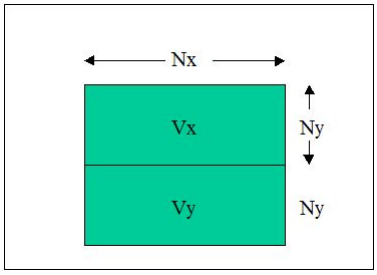


Fig. 1.2: $IType = 1$ or $IType = 3$

- $IType = 2$
  In this case the **Data** field contain the v_x, v_y components of simple 2D vectors from by a PIV calculation. The precision of the data is *single*. See figure 1.3.



w

Fig. 1.3: $IType = 2$

- **IType=4**
  In this case the **Data** field contain the v_x, v_y, v_z components of simple 3D vectors from by a Stereo PIV calculation. The precision of the data is *single*. See figure 1.4.

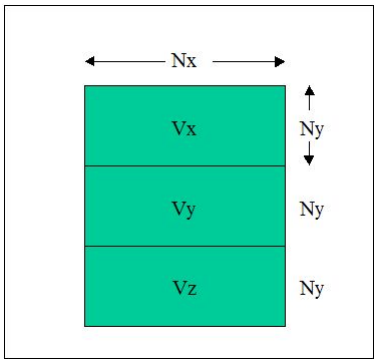


Fig. 1.4: $IType = 4$

- $IType = 5$
  This case equals the case $IType = 5$, but with 3D vector components v_x, v_y, v_z. This conatins 14 blocks, 1 choice index block, 4×3 vector component blocks and 1 *peak ratio* block. The precision of the data is *single*.

```
attrib_list =

_SCALE_Z=1.000000;0.000000;pixel;
_TIME=09:43:10.871
PivCalculationTime=1192.92
FrameDts=10.000000
FrameProcessing0=0
AttributeDisplayFrameInfo=SetupDialogBufferAttrFrameInfo(-1)
CCDExposureTime0=0 μs
CameraName0=1:  RandomParticleTestImage
```

Fig. 1.5: A sample for the Attribute field of an image structure

1.6 The Attributes field

The Attributes field contains a string list of all attribute values of an image or vector file. A sample is shown in figure 1.5. The list consists of a series of assignments separated by newlines. An assignment is a formatted string `a:<label>=<value>`. The label defines the attribute name and the value gives the attribute value. If the value is a vector all entries will be separated by a semicolon.

1.7 The *showimx* function

The ReadIMX loader is delivered with an M-file showimx.m. The function *showimx* displays an image structure similar to the *DaVis* program. The function will be used by

```
>> A=readimx('demo1.imx');
>> showimx(A);
```

For help type the command `>>help showimx`. Please, be invited to learn from the file showimx.m and to use it as a template for your own functions.

1.8 Examples

With this packages some sample image files are delivered (in the subfolder images of the readimx install folder). They can be display for example by:

```
>> img_path='C:\Programme\MATLAB704\toolbox\matlab\readimx\images\';
>> A1=readimx([img_path 'demo1.imx']);
>> showimx(A1);
```

or

```
>> A5=readimx([img_path 'demo5.vec']);
>> showimx(A5);
```

LaVision GmbH
Anna-Vandenhoeck-Ring 19
D-37081 Goettingen
E-Mail: info@lavision.de
www.lavision.de
Tel.: +49-(0)551-9004-0
Fax.: +49-(0)551-9004-100

LaVision Inc.
301 W. Michigan Ave., Suite 403
Ypsilanti, MI 48197, USA
E-Mail: sales@lavisioninc.com
www.lavisioninc.com
Phone: (734)485-0913
Fax: (240)465-4306