Flexible Optical B.V.

Adaptive Optics • Optical Microsystems • Wavefront Sensors

FrontSurfer wavefront analysis and control system

Version 1.4.5, May 2011

OKO Technologies

OKO Technologies is the trade name of Flexible Optical BV

Contents

1 Introduction to FrontSurfer 7

1.1 Basic features . . . 8

1.2 Features of the version for deformable mirrors . . . 9

2 Principles of Shack-Hartmann wavefront sensing 11

2.1 Wavefront sensing . . . 11

2.2 Wavefront reconstruction . . . 12

2.3 Error sources . . . 13

2.3.1 Mask geometry . . . 14

2.3.2 Image sensor geometry . . . 14

2.3.3 Error in the distance between the mask and image sensor . . . 14

2.3.4 Sampling errors . . . 15

2.3.5 Aberrations of the relay optics . . . 15

2.3.6 Errors introduced by the reconstruction algorithm . . . 16

2.3.7 Image sensor noise and nonlinearity . . . 17

2.4 Measurement schemes for optical shop testing . . . 18

2.4.1 Testing of transparent optics . . . 18

2.4.2 Testing of reflective optics . . . 19

2.4.3 Testing of microlenses . . . 20

3 Software manual 23

3.1 Installation (Windows 2000/XP/Vista/7) . . . 23

3.2 Video interface setup . . . 24

3.2.1 “Video for Windows” - compatible frame grabbers . . . 24

3.2.2 Plugin interfaces . . . 25

3.2.3 Development of a custom video plugin 26

3.3 Choosing the measurement mode 27

3.3.1 Absolute measurement mode 28

3.3.2 Reference mode 29

3.3.3 Manually defined aperture 30

3.4 Data processing in FrontSurfer 30

3.5 Calibration with a point source 32

3.6 Examples of wavefront reconstruction 37

3.6.1 Absolute measurement mode 37

3.6.2 Reference mode 42

3.6.3 Reconstruction with manually defined aperture 45

3.7 Alignment mode 48

3.8 Operation with adaptive optics correction loop 50

3.8.1 Mirror configuration 51

3.8.2 Mirror control 53

3.8.3 Feedback loop operation 54

3.8.4 Callback plugin interface 58

3.8.5 Loading and saving the calibration data 59

4 Software reference **61**

4.1 Program windows 61

4.2 “File” pull-down menu 63

4.2.1 Open main 63

4.2.2 Open reference 63

4.2.3 Save output 63

4.2.4 Exit 64

4.3 “Options” pull-down menu 65

4.3.1 Parameters 65

4.3.2 Working directory 67

4.3.3 Surface view 68

4.3.4 Camera 68

4.3.5 Extract terms 68

4.4 “Processing” pull-down menu 69

4.4.1 Do all . . . 69
4.4.2 Find spots . . . 69
4.4.3 Calculate wavefront . . . 69
4.4.4 Calculate far field . . . 69
4.4.5 Calculate interferogram . . . 69
4.4.6 Show report . . . 69
4.4.7 Loop . . . 69
4.5 “Preview” pull-down menu . . . 70
4.5.1 Start . . . 70
4.5.2 Stop . . . 70
4.5.3 Configuration . . . 70
4.6 “Mirror” pull-down menu . . . 71
4.6.1 Configuration . . . 71
4.6.2 Set values . . . 71
4.6.3 Feedback parameters . . . 71
4.6.4 Reference grid position . . . 71
4.6.5 Optimization callback . . . 72
4.6.6 Target function . . . 72
4.6.7 Calibrate mirror . . . 72
4.6.8 Singular values . . . 72
4.6.9 Start feedback . . . 72

Bibliography **73**

Index **73**

Chapter 1

Introduction to FrontSurfer

FrontSurfer wavefront analysis and control system introduces an integral solution for real time wavefront analysis and correction. Applications include optical shop testing, batch testing of microoptics, multi-parameter optical alignment (tip, tilt, defocus, astigmatism, coma) and adaptive optics.

Traditional optics uses solid components, fabricated with nanometer precision. Traditional optical shop testing is based on interferometric methods, involving expensive equipment and lack of real time operation.

FrontSurfer can be used as a wavefront sensor for traditional static wavefront sensing. The system includes all traditional tools and can reconstruct the component profile, interferogram and far field in graphic form. The precision of the reconstruction depends on the mode of operation and the conditions. The most inexpensive configuration features excellent noise-limited sensitivity of about $\lambda/50$ P-V. The system can be used for measurements in visible (using either monochromatic or white light) and near infrared up to 1550nm.

Unlike the interferometric systems, FrontSurfer is capable of real-time display of low-order wavefront aberrations (tip, tilt, defocus, coma and astigmatism). This unique feature makes it a tool of choice for optical alignment and collimation tasks.

FrontSurfer seamlessly integrates with OKO deformable mirror and driver electronic, producing unique system capable of complete diagnostics and real-time aberration correction.

FrontSurfer includes:

- FrontSurfer software that runs under Windows 95/98/2000/NT/XP/Vista/7 32-bit operation systems (Windows XP recommended). Linux is not supported for the latest versions of FrontSurfer, but the Linux version can be supplied by request with a complete system including computer.

- Measurement head, including a monochromatic CCD or CMOS camera with computer interface and a high-precision Hartmann mask or a lenslet array in a

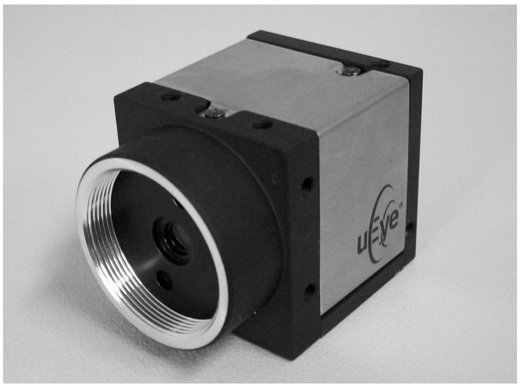


Figure 1.1: *Wavefront sensor measurement head*

C-mount. The measurement head consisting of uEye UI-2210M CCD camera and fused silica microlens array is shown in Figure 1.1.

FrontSurfer is available in two configurations: with and without support for deformable mirrors and adaptive optics feedback.

1.1 Basic features

All versions of FrontSurfer have the following features:

- Reconstruct optical wavefronts from Hartmann and Shack-Hartmann sensor data.
- Produce wavefront plot, synthetic interferogram, far field intensity and report on Zernike terms. All measurement data and results can be saved with one mouse click.
- Process spot patterns obtained with hexagonal, orthogonal or random Hartmann masks and microlens arrays with arbitrary number of apertures.
- Operate with and without reference pattern: in the last case the parameters of the Hartmann mask or microlens array are used as a reference.
- In the alignment mode, real time bar graph display of tip, tilt, defocus, coma and astigmatism; polar display of tilt, coma, astigmatism and position of the intensity maximum.
- Can be interfaced to any camera by developing a custom video plugin according to our specification.
- Allows extracting and canceling arbitrary Zernike terms.
- Allows manually defining a square, orthogonal, elliptical or circular aperture (area of interest).

1.2 Features of the version for deformable mirrors

The FrontSurfer version for deformable mirrors has the following features:

- Manual control of voltages supplied to the mirror.
- Closed-loop correction in real time. Average correction rate is 70 Hz for the wavefront sensor based on Basler A602f CMOS camera and up to 400 Hz with Andor iXon DU-860 EMCCD camera.
- Singular value decomposition (SVD) algorithm with user-defined number of modes for stable feedback.
- Can generate a given aberration in addition to the reference. The aberration is defined as a combination of Zernike polynomials and can be manually controlled during the closed-loop correction using arrow keys of the computer keyboard.
- Supports all types of Micromachined Membrane Deformable Mirrors (MMDM) and Piezoelectric Deformable Mirrors (PDM) produced by OKO Technologies.
- Can be configured for a custom-made mirror interfaced to OKO's PCI boards and USB driver modules.
- Can be interfaced to a custom mirror driver by developing a mirror plugin according to our specification.
- Allows monitoring the residual error of the closed-loop correction and perform custom optimization by developing a callback plugin according to our specification.

Chapter 2

Principles of Shack-Hartmann wavefront sensing

2.1 Wavefront sensing

Among many approaches to test the quality of optical components in the industry and to get the real time wavefront information in adaptive optics, Hartmann sensor has its own distinct place. Hartmann (and Hartmann-Shack) tests are simple and can be explained in terms of pure geometrical optics. The results of the test are easy to interpret and very robust.

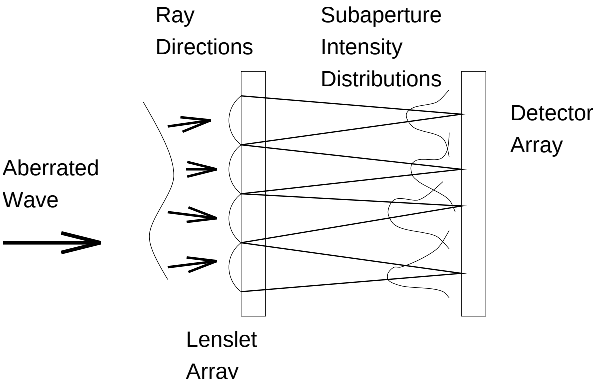


Figure 2.1: *Principle of Hartmann-Shack test.*

The principle of Hartmann (–Shack) test is described in numerous publications - see for example [1].

The wavefront to be reconstructed is sampled by a screen with many sub-apertures (in Hartmann test) or by a dense array of lenslets (Hartmann-Shack test) - see Fig. 2.1. Each sub-aperture (lenslet) produces a light spot on the screen or position sensitive detector. Position of the light spot is associated with the local slope of the incoming wavefront, averaged over the sub-aperture (lenslet) area. After all local slopes of the wavefront are registered, one can reconstruct the shape of the wavefront.

FrontSurfer uses a precision Hartmann mask or microlens array to sample the wavefront and a high-resolution CCD or CMOS camera to image the light spots.

2.2 Wavefront reconstruction

Two approaches can be used in the reconstruction of the wavefront profile from the response of the Hartmann sensor. In the first approach, introduced in [2, 3], the phase is reconstructed at the nodes of a discrete grid which coincide with the centers of sub-apertures in the raster. The algorithm uses a finite-differential approach and does not allow for reconstruction of the wavefront as a continuous function. In the second approach, the required wavefront is represented by a series expansion over a system of linearly independent basis functions, and the coefficients of expansion are calculated in terms of this basis. The reconstructed aberration is then defined continuously throughout the whole aperture of the sensor raster and may even be extrapolated outside the aperture. These algorithms make it possible to calculate the coefficients in the series expansions for different dimensions and configurations of the sensor sub-aperture raster. Moreover, different sets of basis functions φ_i can be used.

Let the response ψ_i be corresponding to the input aberration φ_i. For a known set of the sensor responses ψ_i corresponding to the basis functions φ_i, the expansion of the arbitrary response ξ can be written in terms of basis responses ψ_i:

$$\xi = \sum \psi_i c_i$$

where c_i is the set of expansion coefficients. The set c_i should be calculated from the condition of the minimal length of the error:

$$\left|\xi - \sum \psi_i c_i\right|^2 = \min$$

The computational complexity of this minimization problem can be greatly reduced when responses ψ_i form an orthogonal basis. We have to note that even when an orthogonal basis is chosen for the expansion of the wavefront shape, it does not guarantee the orthogonality of the responses ψ_i. Even for aberrations described by orthogonal Zernike polynomials, the corresponding responses of Hartmann sensor may be non-orthogonal. The orthogonality may be established by forming an orthonormal basis α_i in the space of linearly independent responses ψ_i by using Gram–Schmidt orthogonalization:

$$\alpha_i = \sum a_{i,j} \psi_j$$

where $(\alpha_i, \alpha_j) = \delta_{ij}$. The matrix $a_{i,j}$ transforms the basis of arbitrary linearly independent responses ψ_i into orthogonal basis α_i in the space of Hartmann sensor responses. Now the response of a Hartmann sensor ξ can be expanded on the orthonormal basis α_i:

$$\xi = \sum b_i \alpha_i$$

where coefficients of expansion b_i in terms of the orthonormal basis α_i are found as scalar products

$$b_i = (\alpha_i, \xi)$$

It follows directly from previous expressions that the resulting expansion can be calculated as:

$$c_j = \sum b_i a_{i,j} \tag{2.1}$$

Finally, the reconstruction of the wavefront is divided in two stages:

1. Orthogonalization of the basis responses. The orthonormalization procedure is carried out once for given parameters such as the number and the positioning of sub-apertures in the Hartmann sensor, the nature of the basis functions and the number of expansion coefficients. Sensor responses ψ_i are recorded for each input aberration of the basis φ_i. This is followed by the calculation of the matrix $a_{i,j}$ and the orthogonal basis of the sensor responses α_i. But from practical point of view using of Gram-Schmidt ortogonalization is not a good choice because of the build-up of roundoff errors. On that reason FrontSurfer uses singular value decomposition algorithm [4] to construct an orthonormal basis.

2. The reconstruction process consists of measuring the response of the Hartmann sensor ξ, calculating coefficients b_i and finally, calculating coefficients c_i using (2.1).

FrontSurfer uses Zernike polynomials (see page 31) as a standard set to describe aberrations of optical systems, any other system of continuous linearly-independent functions can be used in special cases. For instance, a system of influence functions of a deformable mirror can be used for adaptive optics applications.

2.3 Error sources

Error sources in a Hartmann(-Shack) test are as follows.

- Mask / microlens array geometry errors (can be extracted; negligible if high-precision lithography used);

- Errors introduced by the deviations of the design parameters of the sensor: deviations in the distance between the mask and the CCD and the difference in the CCD pixel pitch in X and Y directions.

- Sampling errors introduced by a Hartmann mask or microlens array. Depends on the mask geometry and the wavefront nature. Very low for smooth wavefronts but can introduce a serious problem with “rippled wavefronts”.
- Aberrations of the relay optics.
- Errors introduced by the reconstruction algorithm.
- Image sensor noise, non-linearity, pitch and non-uniformity of intensity.

Below we analyze each error source mentioned above.

2.3.1 Mask geometry

FrontSurfer Hartmann masks are fabricated by patterning metal layer on plane-parallel glass plate, pretty much the same way as masks for photo-lithography are fabricated. This provides very high precision of the mask geometry: sub-apertures are centered with a precision better than 1μm. In fact, Hartmann sensor with FrontSurfer mask can be used without any preliminary calibration. *Rms* aberration introduced by the mask and the glass plate does not exceed 10nm.

Microlens arrays are fabricated by microstructuring of glass or fused silica or by replication from a silicon matrix using UV curable material. Precision of a microlens array is lower than one of a Hartmann mask. When an ideal hexagonal or square grid is used as a reference, it may result in wavefront reconstruction error up to $\lambda/6$. FrontSurfer allows extracting this error from the reconstruction results - see section 4.3.5.

2.3.2 Image sensor geometry

Modern CCD with square pixels may have pixel pitch difference between X and Y directions of up to 0.25%. If not taken into account, these errors can result in “fake” astigmatism in the reconstructed wavefront. FrontSurfer uses a special point-source based calibration procedure to eliminate these errors.

2.3.3 Error in the distance between the mask and image sensor

Error in the distance between the mask and the CCD reveals itself in the amplitude of reconstructed aberration. Since this distance is very difficult to measure, FrontSurfer uses special calibration procedure to precisely calculate the distance using the data obtained with a point source at two different distances from the Hartmann mask. Typically we calibrate with a point source (pinhole) placed on the optical axis at two different distances from the mask. This allows to improve the absolute precision of wavefront reconstruction from about 10% without calibration to better than 1% after calibration.

2.3.4 Sampling errors

Any regular structure used to sample the input beam will introduce sampling errors.

In the case of Hartmann mask the fill factor (sampled_area/total_area) is always smaller than 1, typically in the range 0.25 . . . 0.35. This means literally that we throw away up to 75% of information about the wavefront! According to famous Shannon theorem, the wavefront will be reconstructed correctly if there are at least 4 sampling points per period of the highest harmonic of the measured function, and the more the better. Example of wrong reconstruction produced with broken Shannon law is shown in Fig. 2.2 (all sub-apertures co-match with zero tilt regions resulting in reconstruction of flat wavefront, while the real wavefront is presented by a harmonic function). Calibrated Hartmann masks can be used for measurements of smooth wavefronts with low spatial frequencies. Lenses and mirrors are good objects to be controlled with amplitude Hartmann mask, because laboratory conditions allow for high quality illumination, moreover these components have mainly low-order aberrations.

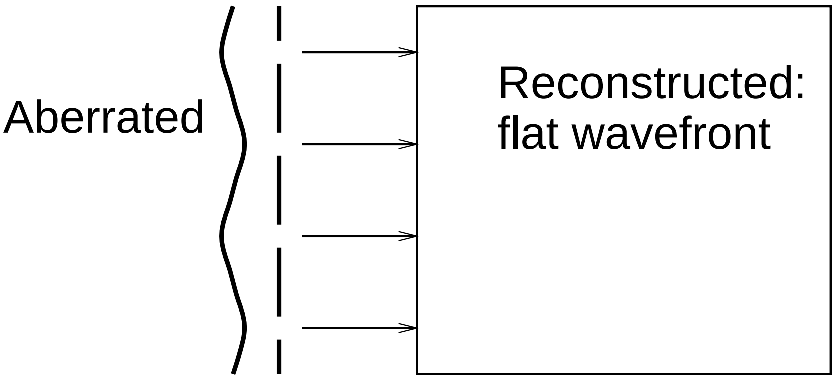


Figure 2.2: *Erroneous reconstruction as a result of under-sampling*

Hartmann-Shack approach solves the fill-factor problem by replacing the amplitude mask with lenslet array having fill factor close to 100%. Lenslets average the wavefront tilts over larger sub-apertures, producing more robust reconstruction. Besides, they are more efficient, collecting more light and producing sharper spots.

2.3.5 Aberrations of the relay optics

Hartmann (-Shack) sensor measures the wavefront shape in the plane of the sampling screen. The size of the screen depends on the size of the CCD. For instance, standard FrontSurfer hexagonal masks and microlens arrays have a clear aperture of 3.5. . . 4.5 mm.

In most cases the light beam should be scaled to the size of the Hartmann mask. This operation introduces additional aberrations, therefore special attention must be paid to the understanding of what exactly is measured in the plane of the wavefront sensor. Compared to interferometers, the Hartmann (-Shack) sensor is much more sensitive to the aberrations of the relay optics used to re-image the measurement plane to the sampling screen. In a typical interferometric setup, the interferogram is formed as **intensity modulation** in the plane of the object. Any imaging system can be used to relay this image, phase errors introduced during the image relay are not significant because the real phase information is already coded as intensity. A Hartmann(-Shack) sensor cannot distinguish between the aberration to be measured from the aberrations introduced by the relay optics. Special calibration is required to filter the parasitic relay aberrations from the "real ones" that should be measured.

Thus, the measurement setup should satisfy the following conditions:

- The optics should re-image the plane of the object to the plane of the Hartmann mask.
- The scheme should scale the beam in such a way that the image of the scaled object (wavefront) is densely filled with Hartmann sub-apertures.
- The optics should allow for calibration. In the general case, it consists of separate measurement of the complete setup aberration with ideal object or a source of ideal wavefront, replacing the one to be tested.

Three typical measurement setups satisfying these conditions are described in Section 2.4.

2.3.6 Errors introduced by the reconstruction algorithm

Any reconstruction algorithm relies on a set of suggestions about the nature and behavior of the wavefront to be reconstructed. Most commonly the wavefront is decomposed over a set of Zernike polynomials. This is a pretty good "suggestion" from the practical point of view, since Zernike polynomials nicely describe low-order aberrations of optical systems, moreover they are close to Karhunen-Loéve functions of phase distortions produced by atmospheric turbulence.

But, just for example, Zernike polynomials can be a bad choice to describe aberrations of a long rectangular window.

Analysis of errors introduced by the reconstruction algorithms is quite complicated in general case and should take into account statistics of the wavefront to be measured and geometry of the raster [5]. However, there are a couple of rules of thumb:

- Reconstruction error will first drop, and then grow with the number of wavefront decomposition terms. For our "standard" arrays with 127 microlenses

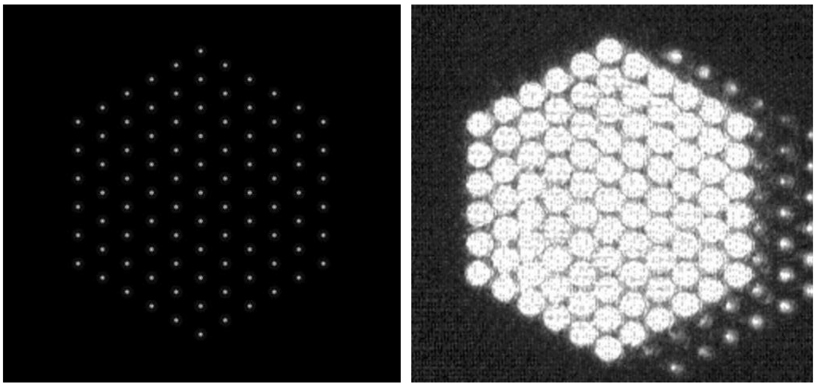


Figure 2.3: *Intensity patterns from the Shack-Hartmann sensor; a good quality one (left), one with noise and saturation (right).*

we advise using 44 Zernike terms. In general, the number of Zernike terms should be between 1/3 and 1/2 of the number of spots.

- Reconstruction error is maximal at the edges of the reconstruction area because it is influenced by the geometry of the mask and because the boundary conditions are not determined, or are determined by the reconstruction algorithm and not by the nature of the aberration. That is why we advise disregarding the edge areas for wavefront reconstructions.

2.3.7 Image sensor noise and nonlinearity

Each sub-aperture of the Hartmann mask (lenslet array) produces a light spot on the CCD or CMOS image sensor matrix; thus, the reconstruction software should deal with an intensity distribution consisting of a set of spots. For correct processing it is important to minimize the background noise and prevent saturation of spots. Although it may be possible to locate spots in presence of noise and saturation, these factors may affect the precision and repeatability of the results.

Things to check:

- The spots should be bright enough to be clearly visible and separated from each other and from the background noise.
- The image should occupy the linear range of the CCD camera. Light spots should look “gray-scale”, if there are saturated areas in the spot images, the result may be incorrect.
- **Set the gamma parameter of the camera to 1!**

Examples of a good and bad intensity patterns are shown in Figure 2.3.

It is also important to minimize the variation of intensity between the spots. This variation may be caused by the structure of the illuminating beam, interference

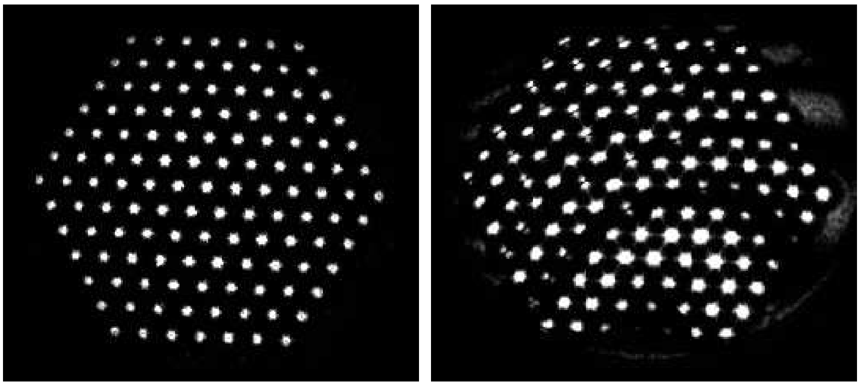


Figure 2.4: *Intensity patterns with inhomogeneous intensity; acceptable (left) and unacceptable quality (right).*

with parasitic beams reflected from the components of the optical system, bad optical conjugation between the sensor and the object under test, birefringence of the object and too strong aberrations. Example hartmanngrams with acceptable and unacceptable intensity variation are shown in Figure 2.4.

2.4 Measurement schemes for optical shop testing

Optical setup for testing of optical components with the Shack-Hartmann wavefront sensor should satisfy the following conditions:

- The relay optics should re-image the plane of the object to the plane of the Hartmann mask (lenslet array).
- The scheme should scale the beam in such a way that the image of the scaled object (wavefront) is densely filled with Hartmann sub-apertures.
- The optics should allow for calibration. In the general case, it consists of separate measurement of the complete setup aberration with ideal object or a source of ideal wavefront, replacing the one to be tested.

Three typical measurement schemes are considered below.

2.4.1 Testing of transparent optics

A typical optical setup for measurement of transparent optics with Shack-Hartmann wavefront sensor is shown on Figure 2.5. The object under test should be placed in a collimated laser beam. A telescopic system consisting of two lenses, L_1 and L_2, is used to re-image the aperture of the object to the microlens array (or Hartmann mask) of the wavefront sensor. The system should scale the area to be measured to

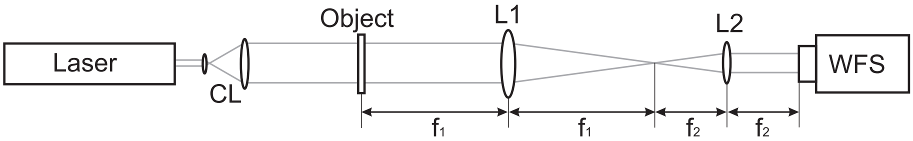


Figure 2.5: *Typical measurement scheme for testing of transparent optics with Shack-Hartmann wavefront sensor. Here CL is a collimator;* L_1 *and* L_2 *are lenses with focal distances* f_1 *and* f_2*, respectively; WFS is the wavefront sensor.*

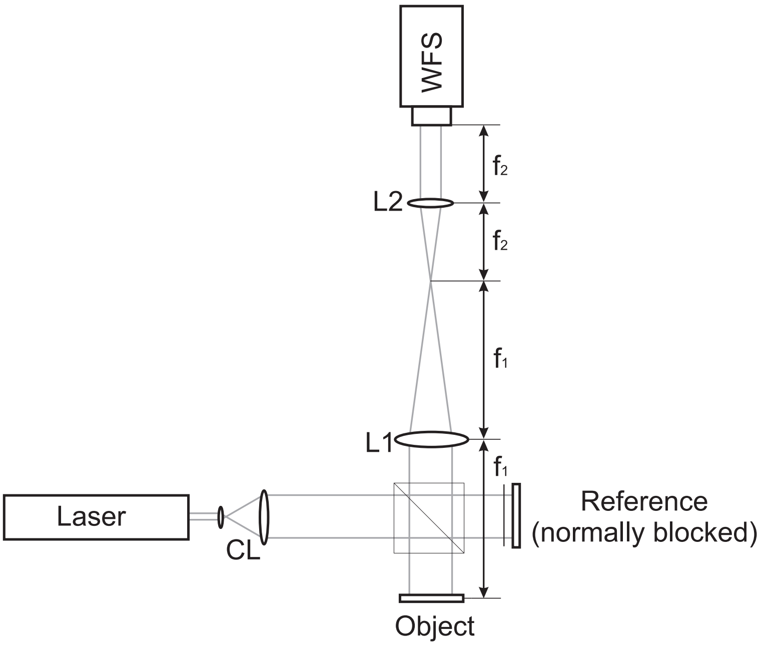


Figure 2.6: *Typical measurement scheme for testing of reflective optics with Shack-Hartmann wavefront sensor. See Figure 2.5 for notations.*

the aperture size of the wavefront sensor; the scaling factor is equal to f_1/f_2, where f_1 and f_2 are focal distances of the lenses L_1 and L_2, respectively.

Although "FrontSurfer" is able to perform "absolute" wavefront measurement without any reference pattern, it is advisable to switch to the reference mode. It allows to get rid of aberrations of the relay optics, which are always present due to imperfection of the components and misalignment in the system. Reference pattern can be measured with the object removed from the setup.

2.4.2 Testing of reflective optics

A typical optical setup for measurement of reflective optics is shown on Figure 2.6. To facilitate calibration, it is convenient to build it in a similar way to the Twyman-Green interferometer, where a beam splitter divides the incoming beam in two branches. The object is placed the in first branch behind the beam splitter, and the reference (normally a high-quality flat or spherical mirror) in the second one. The

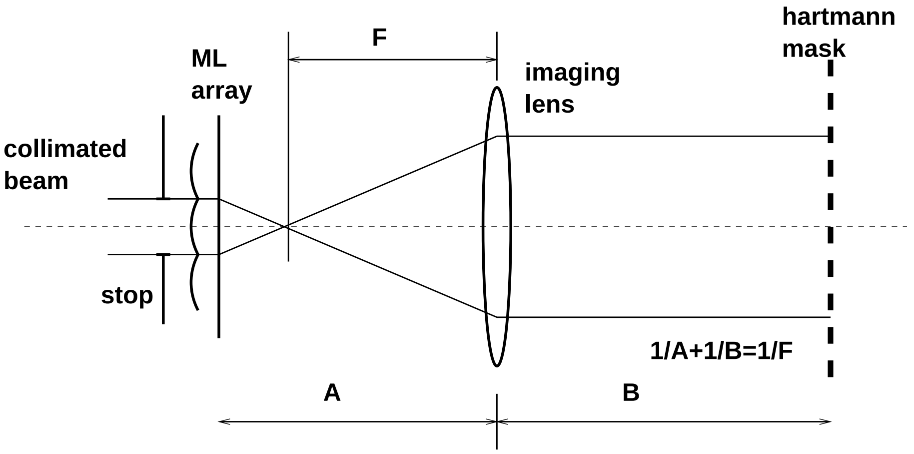


Figure 2.7: *Measurement scheme for testing of a microlens with Shack-Hartmann wavefront sensor.*

beams reflected from the object and reference are then directed to the wavefront sensor. A telescopic system similar to the one from the previous scheme (Figure 2.5) is used to provide conjugation between the object and WFS with proper scaling. One of the branches should always be blocked – the first branch for measurement of the reference pattern, and the second one for testing of the object.

2.4.3 Testing of microlenses

An example of a relay system designed to measure the aberrations of a microlens for fiber coupling applications is shown in Figure 2.7. In this case the microlens has a small diameter (1 mm) compared to the Hartmann mask aperture (4 mm). A telescope formed by the microlens itself and the imaging lens is used to couple the microlens aperture to the sensor mask. The Hartmann mask is co-incident with the microlens image.

Calibration of the setup consists of two steps:

1. Obtaining of an ideal reference wavefront.
2. Calibration of the imaging microlens.

In the first step we ensure that the aberration of the illuminating beam is negligible over the aperture of the microlens. In the simplest case this can be achieved by illumination of the microlens with slightly divergent wavefront produced by a remote pinhole source. If there are no optical components between the pinhole and tested microlens, we can be sure the illumination wavefront is close to ideal.

In the second step, we place another pinhole in the focus of the microlens. By doing that we eliminate all aberrations of the setup, except for the aberrations introduced by the imaging lens, to be able to measure the aberration of the relay optics only.

In the last step, we remove the pinhole and measure the total aberration. To obtain the aberration of the microlens, the aberration of the imaging setup measured at step 2 should be extracted from the measurement result.

In this example we have used slightly divergent wavefront instead of collimated. This does not introduce any significant problem if the radius of curvature of the wavefront is much larger than the focal length of the microlens under test. If the lens has focal length of 4 mm and the point source is placed at 1 m from the lens, spherical aberration introduced by “wrong” illumination is negligible. Nevertheless, it is always advisable to check expected aberrations using ray-tracing code.

Chapter 3

Software manual

3.1 Installation (Windows 2000/XP/Vista/7)

1. Start “setup.exe” from “fsurfer” directory of the installation CD to install FrontSurfer to your computer. Follow further installation instructions.

2. Start “Install.exe” from “keylok” directory of the installation CD to install drivers for the protection dongle. Select the option “USB dongle”. Please note that the installation should be completed BEFORE the dongle could be connected.

3. (Only in the version for deformable mirrors) To be able to operate with deformable mirrors, you need to install “DLPortIO” library. Go to the directory “DLPortIO” and locate the setup program “port95nt.exe”. Under Windows Vista and 7, it should be started in compatibility mode with Windows XP and under administrator access rights. To enable them, right-click on “port95nt.exe” and locate “Compatibility” property sheet. Enable the options “Run this program in compatibility mode for Windows XP (Service Pack 3)” and “Run this program as an administrator” and press OK to confirm. Install “port95nt.exe” and reboot after the installation.

4. Attach the FrontSurfer dongle to a free USB port. The system will recognize the device. Choose for automatic installation of the driver.

5. Under Windows Vista and 7, FrontSurfer should be started in compatibility mode with Windows XP and under administrator access rights. To enable them, right-click on “FrontSurfer” shortcut and locate “Compatibility” property sheet. Enable the options “Run this program in compatibility mode for Windows XP (Service Pack 3)” and “Run this program as an administrator” and press OK to confirm.

6. Start “FrontSurfer” from the Start menu.

If FrontSurfer is supplied with a wavefront sensor or as a part of an adaptive optical system including both the wavefront sensor and deformable mirror, read the technical notes or technical passport of your system for instructions regarding assembling and configuration of the system.

3.2 Video interface setup

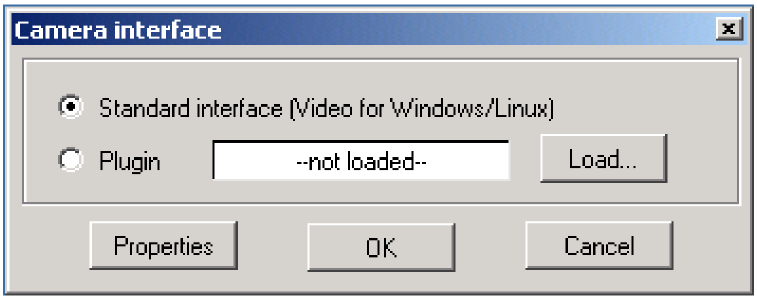
FrontSurfer has embedded support of standard Windows interface for CCD camera - “Video for Windows”. Using this interface, the software will work with any TV tuner devices, whose support is realized through this standard interface, such as low-cost BT848/868 based PCI boards and WinTV USB frame grabber. However, the alignment mode and the close-loop operation mode do not work with this interface. It is recommended to use a higher quality PCI frame grabber or a camera with Firewire or USB interface for full functionality.

Nevertheless, many scientific frame grabbers supplied with drivers which are not compatible with these common interfaces. FrontSurfer supports external video plugin modules. It makes possible to customize FrontSurfer for using with almost any camera and any frame grabber.

3.2.1 “Video for Windows” - compatible frame grabbers

1) Install the frame grabber and drivers for it, as described in documentation of your card.

2) Go to the menu command “Options ⇒ Camera”. The window “Camera interface” - see Fig.3.1 - will pop up.



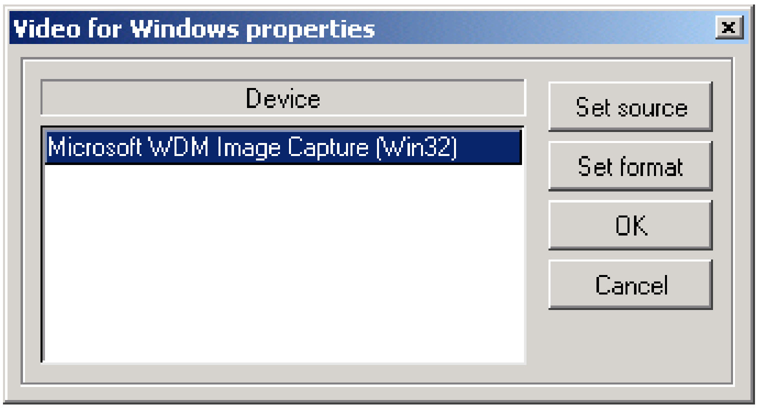
● Standard interface (Video for Windows/Linux)

○ Plugin --not loaded--

Figure 3.1: *The camera interface window.*

3) Choose “Standard interface” option and press “Properties” button. The dialog box “Video for Windows properties” will appear (Fig.3.2).

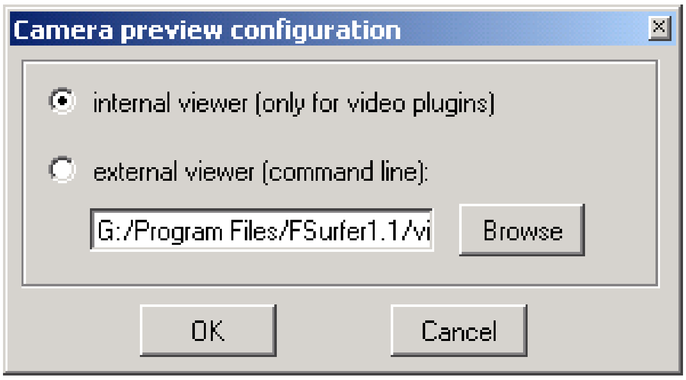
Select your device in the list of available devices and adjust the capture properties using “Set source” and “Set format” buttons. The exact view of “Set source” and “Set format” dialog boxes is device-dependent, as well as the set of available options.



Microsoft WDM Image Capture (Win32)

Figure 3.2: *Video for Windows properties window.*

4) Set the camera preview options. With this purpose go to the menu command "Preview ⇒ Configuration".



● internal viewer (only for video plugins)

○ external viewer (command line):

G:/Program Files/FSurfer1.1/vi

Figure 3.3: *Preview configuration window.*

Select the option "external viewer" and type "VidCon32.exe". The viewer program "VidCon32" is supplied with the FrontSurfer and can be found in the installation directory. Press "OK" to finish.

3.2.2 Plugin interfaces

Plugins for the following video interfaces are included in the distribution:

- Generic plugin for DirectShow-compatible video devices; supports most of webcams and TV tuners (file "fsplugin_ds.dll");
- Picasso PCI-2SQM framegrabber from Arvoo (file "p2sqm.dll");
- DFG/LC1 framegrabber from Imaging Source (file "DFG_LC1.dll");
- A601f and A602f CMOS digital cameras from Basler (file "BCAM1_8.dll" for BCAM driver version 1.8 and "BCAM1_9.dll" for BCAM driver version 1.9); may be suitable for some other models of Basler;

- A601f-HDR from Basler (file “BCAM_HDR_1_8.dll”);
- Basler Pylon interface for GigE cameras (file “PylonGigEPlugin.dll”);
- Basler Pylon interface for 1394 (Firewire) cameras (file “Pylon1394Plugin.dll”);
- Firefly CMOS digital camera from Point Grey Research (file “firefly.dll”);
- UI-2210M and UI-2410M CCD digital cameras from IDS Imaging; may be suitable for some other models of IDS (file “ueye.dll”);
- RM4200GE GigE CCD camera from JAI (file “RM4200GE.dll”).

Plugins for the fast cameras iXon DU-860 from Andor, XEVA-1.7-320 from XenICs and SVS340 from SVS Vistek are available by request.

In the general case, the configuration procedure is as follows.

1) Install the frame grabber and drivers for it, as described in documentation of your card.

2) Go to the menu command “Options ⇒ Camera”. The window “Camera interface” - see Fig.3.1 - will pop up.

3) Choose “Plugin” option and press “Load. . . ” button. Locate the corresponding DLL and press “OK”.

4) To change capture options, press “Properties” button. Configuration is specific for each type of a frame grabber or a camera. Normally this procedure is described in the “readme” file for each particular system.

5) Set the camera preview options. With this purpose go to the menu command “Preview ⇒ Configuration”. Select the option “internal viewer”. Optionally, you may use any other external viewer program you may find suitable. Press “OK” to finish.

3.2.3 Development of a custom video plugin

Plugin template source code (file “video_plugin.zip” in FrontSurfer installation directory) is supplied now with FrontSurfer. The sample can be compiled with Microsoft Visual C++. Corresponding dynamically loaded library should export 7 functions declared as follows

```
void properties();

int init_capture();

void* read_frame(int* Nx, int* Ny, int* nType);

void close_capture();

int start_preview();
```

`void stop_preview();`

`int stop_plugin();`

Function “properties” is used to display a dialog box for setting of driver-specific parameters.

Function “init_capture” and “close_capture” are used for initialization and release of frame grabber device and/or capture buffers. “init_capture” should return 0 if initialization was successful, otherwise it should return -1.

Function “read_frame” returns row-wise arranged array of values read from the CCD camera; its dimension is (*Nx)x(*Ny). Addresses Nx and Ny are used to store values of width and height of the picture in pixels. Parameter ”nType” is a pointer to the type description of buffer values; the buffer type can be described by one of the following values:

```
#define TYPE_BYTE 0 // unsigned 8-bit integer

#define TYPE_WORD 1 // unsigned 16-bit integer

#define TYPE_DWORD 2 // unsigned 32-bit integer

#define TYPE_DOUBLE 3 // floating point number with double precision
```

If any error occurs during the capture, “read_frame” should return NULL. Generation and displaying of error messages is up to developer.

Functions “start_preview” and “stop_preview” are used to display and close video preview window. “start_preview” returns 0 if the window was successfully opened, and -1 in the case of error. These functions are optional; FrontSurfer can be configured to use any external program for preview.

Optional function “stop_plugin” is used to close the video interface in a safe way. It can be useful for coolable cameras, where sudden interruption in the work of camera during the cooling can damage the image sensor. In case the camera cannot be closed safely at the moment, this function returns -1, preventing the user from closing FrontSurfer or from unloading the plugin. Otherwise, it returns 0, which means that the plugin can be stopped safely.

3.3 Choosing the measurement mode

FrontSurfer can operate with and without a reference pattern. In the latter case, the geometric parameters of the Hartmann mask or microlens array are used as a reference, which allows for absolute wavefront sensing. In the absolute measurement mode we deal with only one intensity pattern; we refer to it as the *main pattern* as it contains information about the wavefront to be reconstructed. In the reference mode, we also use the second intensity pattern, which contains information about geometry of the Hartmann mask (lenslet array) and aberrations of the relay optics. We refer to it as the *reference pattern*.

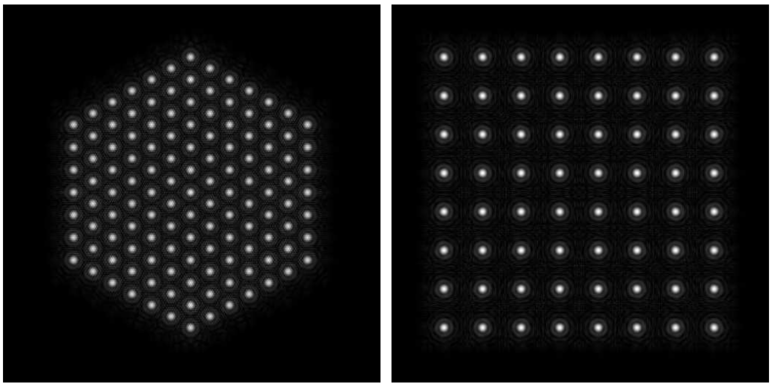


Figure 3.4: *Complete hexagonal (left) and orthogonal (right) spot patterns.*

The choice between the *absolute measurement* and *reference* modes depends on the application. If the wavefront sensor is used for alignment of an optical system (i.e., collimation or astigmatism removal) then the absolute measurement mode would be more useful. If it is used in an optical testing scheme similar to those described in Section 2.4 then the reference mode is preferable, as it allows to extract aberrations of the relay optics from the measured data.

3.3.1 Absolute measurement mode

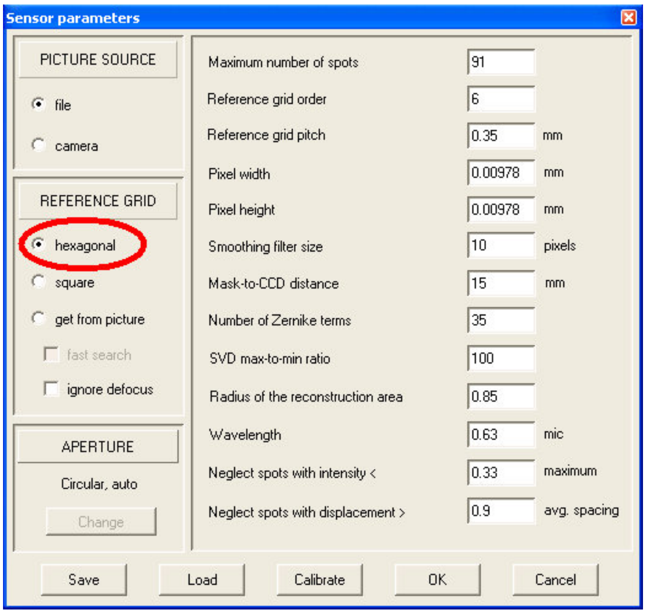
The absolute measurement mode can be used with a hexagonal or orthogonal array. A complete hexagonal or orthogonal pattern (i.e., a complete hexagon or square, as shown in Figure 3.4) is highly recommended, although the program can tolerate up to 5 % missing spots.

The precision achievable depends on the precision of the mask or array used. Any defect of the mask/array will be interpreted as a wavefront deviation, therefore we do not recommend using third-party masks or arrays in this mode.

To switch to the absolute measurement mode, go to the menu "Options ⇒ Parameters". In the section "Reference grid" choose "hexagonal" or "square" depending on the array type. In Figure 3.5, the absolute measurement mode is chosen for a hexagonal pattern. Dimensions of the array are defined by the parameters "Reference grid pitch" (distance between centers of sub-apertures in the array) and "Reference grid order" (number of spots in one side of the polygonal structure; the program calculates the total number of sports accordingly).

After defining the geometry of the array, it is important to calibrate the sensor to get the correct pixel dimensions and the distance between the mask and image sensor. However, usually FrontSurfer sensor is supplied with a calibration file, and it is enough to load this file. The calibration procedure is described below, in Section 3.5.

Choosing the option "Ignore defocus" allows to build an ideal reference grid even in presence of a large defocus, which greatly affects spacing between spots in a hart-



PICTURE SOURCE
● file
○ camera

REFERENCE GRID
● hexagonal
○ square
○ get from picture
☐ fast search
☐ ignore defocus

APERTURE
Circular, auto

Maximum number of spots	91	
Reference grid order	6	
Reference grid pitch	0.35	mm
Pixel width	0.00978	mm
Pixel height	0.00978	mm
Smoothing filter size	10	pixels
Mask-to-CCD distance	15	mm
Number of Zernike terms	35	
SVD max-to-min ratio	100	
Radius of the reconstruction area	0.85	
Wavelength	0.63	mic
Neglect spots with intensity <	0.33	maximum
Neglect spots with displacement >	0.9	avg. spacing

Figure 3.5: *FrontSurfer options for the absolute measurement mode with hexagonal pattern.*

manngram. In this case, the actual grid pitch will be determined from optimization. However, defocus component of the wavefront will be ignored.

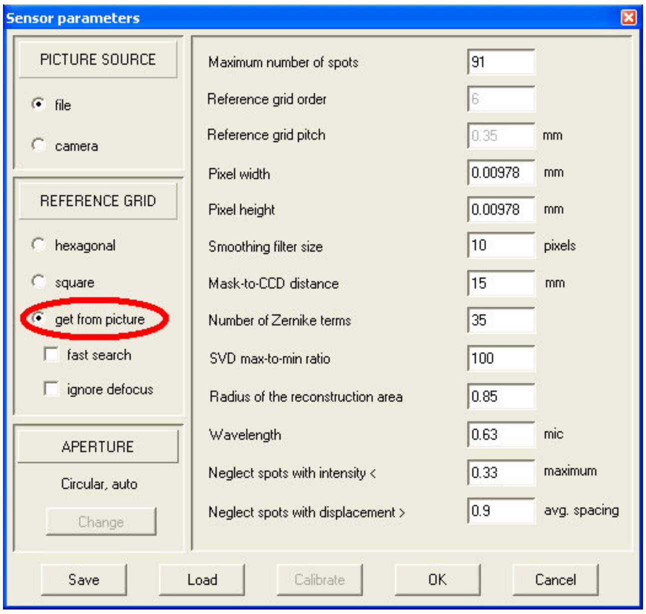
An example of using the absolute measurement mode is given in Section 3.6.1.

3.3.2 Reference mode

In the reference mode, the spot pattern can be neither complete nor regular; one can even use an array with random positioning of sub-apertures. To switch to the reference mode, go to the menu “Options ⇒ Parameters”. In the section “Reference grid”, choose the option “get from picture”, as shown in Figure 3.6. The number of spots to be found is variable; it depends on two parameters - the intensity threshold (with respect to the intensity of the brightest spot), which is specified in the field “Neglect spots with intensity <”, and the upper limit specified as “Maximum number of spots”. An example of using the reference mode is given in Section 3.6.2.

A very important option is “fast search”. When it is not selected, the program searches for spots in the main and reference patterns independently and then tries to find the correspondence between both patterns. Relatively large displacements between patterns can be tolerated in this mode.

When “fast search” is selected, the program first searches for spots in the reference pattern and builds the bounding boxes around them. Then, spots of the main pattern are searched within the bounding boxes calculated from the reference pattern. It makes processing of the main pattern much faster compared to processing of the reference one. Similar technique is used in the closed-loop operation mode (with



● file
○ camera

○ hexagonal
○ square
● get from picture
☐ fast search
☐ ignore defocus

Circular, auto

Maximum number of spots	91	
Reference grid order	6	
Reference grid pitch	0.35	mm
Pixel width	0.00978	mm
Pixel height	0.00978	mm
Smoothing filter size	10	pixels
Mask-to-CCD distance	15	mm
Number of Zernike terms	35	
SVD max-to-min ratio	100	
Radius of the reconstruction area	0.85	
Wavelength	0.63	mic
Neglect spots with intensity <	0.33	maximum
Neglect spots with displacement >	0.9	avg. spacing

Figure 3.6: *FrontSurfer options for the reference measurement mode.*

deformable mirrors), where fast processing is important. However, it is applicable only to spot displacements less than half pitch of the microlens array; otherwise the measurement results will be wrong.

3.3.3 Manually defined aperture

It is possible to define area of interest (aperture) within the area of the intensity pattern. It is very useful for sensors with a large mask or lenslet array, whose subapertures fill the whole area of the image sensor, and for measurement of objects with rectangular or elliptical apertures. An example of using this mode will be given in Section 3.6.3.

3.4 Data processing in FrontSurfer

At the first stage of processing, FrontSurfer applies a smoothing filter to the intensity pattern in order to minimize the influence of the intensity noise. Then it tries to locate centroids of the brightest spots in the smoothed intensity distribution. Not more than a given number of spots with the intensity higher than a specified threshold value is being located. The window size of the smoothing filter, the maximum number of spots to be located and the intensity threshold (with respect to the maximum intensity) can be adjusted by the user via menu “Options ⇒ Parameters”.

Centroids of spots of the main pattern form the so-called *main grid*, and ones of the reference pattern form the *reference grid*. In the absolute measurement mode, FrontSurfer generates the reference grid as an ideal hexagonal or orthogonal grid with a specified pitch and dimension. FrontSurfer optimizes the position and rotation angle of the ideal grid to provide the closest correspondence between the nodes of the two grids.

At the next stage FrontSurfer calculates local tilts of the wavefront. This is implemented by finding a correspondence between coordinates of centroids of two grids – the main and reference ones – and calculating displacement of each spot due to aberrations. The minimum permitted displacement can be specified by the user. If no correspondence is found for a certain spot, or its displacement exceeds the specified limit, the spot is discarded. For calculation of the tilts, FrontSurfer uses the pixel dimensions and the distance between the Hartmann mask (lenslet array) and the CCD (CMOS) matrix. These parameters can either be set by the user manually, or obtained by calibrating the sensor.

Wavefront reconstruction is performed based on the reference grid and the array of the corresponding local tilts. FrontSurfer uses *modal reconstruction*, which means that the required wavefront is represented by a series expansion over a system of linearly independent basis functions, and the coefficients of expansion are calculated in terms of this basis (Zernike polynomials). The reconstructed wavefront is then defined continuously throughout the whole aperture of the sensor.

The reconstruction is divided in several stages:

1. **Calculation of the basis responses**. As the basic functions, FrontSurfer uses sets of tilts, which correspond to aberrations represented by Zernike polynomials. The number of Zernike terms can be adjusted by the user.
2. **Orthogonalization of the basis responses**. Even for aberrations described by orthogonal Zernike polynomials, the corresponding responses of the wavefront sensor may be non-orthogonal. FrontSurfer uses singular value decomposition (SVD) algorithm to construct an orthogonal basis [4]. Discarding of those modes having relatively low singular values allows to make reconstruction more steady, especially if the wavefront is approximated by large number of Zernike terms. The range of “good” SVD values can be adjusted by the user. Normally, taking into account modes with singular values larger than 1/100 of the maximum one provides stable reconstruction.
3. **Decomposition** of the tilts over the orthogonalized basis using SVD algorithm; it results in a set of Zernike coefficients representing the wavefront.
4. **Calculation of the wavefront** as a superposition of Zernike polynomials with the coefficients found.

We use Zernike polynomials with normalization as follows:

$$\begin{aligned} Z(n,-m) &= R_n^m(\rho)\cos m\theta, \\ Z(n,m) &= R_n^m(\rho)\sin m\theta, \end{aligned}$$

where R_n^m are radial polynomials

$$R_n^{\pm m}(\rho) = \sum_{s=0}^{\frac{n-m}{2}} (-1)^s \frac{(n-s)!}{s!\left(\frac{n+m}{2}-s\right)!\left(\frac{n-m}{2}-s\right)!}\rho^{n-2s},$$

ρ and θ are polar coordinates defined as follows: $x = \rho\cos\theta$, $y = \rho\sin\theta$.

3.5 Calibration with a point source

After connecting a new Hartmann mask or microlens array to a wavefront sensor it is necessary to calibrate the sensor to ensure good precision for future wavefront measurements. For calibration we use a point-like light source (such as a pinhole or narrow optical fiber) placed at two different positions before the wavefront sensor on its optical axis. For both positions we capture hartmanngrams and process them with FrontSurfer. Calibration requires switching to the absolute measurement mode and defining the geometry of the mask (array). FrontSurfer uses these data to calculate the pixel dimensions - width and height - and the distance between the mask and image sensor. As soon as the calibration is completed, FrontSurfer has enough data for precise wavefront measurement.

We shall demonstrate the calibration using the sample files distributed with FrontSurfer; they can be found in the directory “sample”.

Start the program.

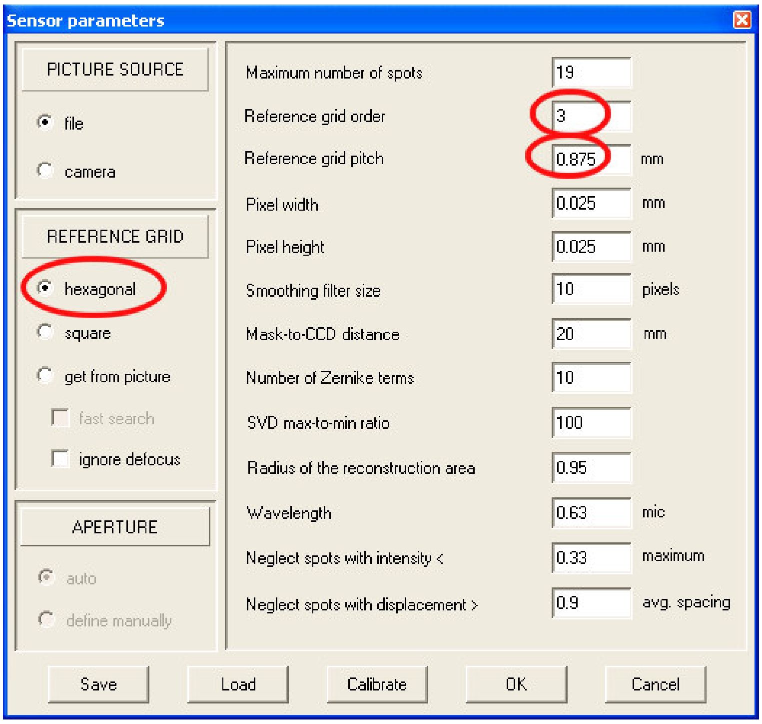
Welcome

Figure 3.7: *The main window.*

The main window - see Fig. 3.7 will pop up.

To calibrate the sensor go to menu “Options” and choose “Parameters”. Parameter window - see Fig 3.8 will appear.

Make sure the source is set to “file” and the reference grid is set to “hexagonal”. Set “Reference grid pitch” to 0.875 and “Reference grid order” to 3, and the number of spots will change to 19. All examples in this primer are done with grid order 6 (91 spots). The demo version does not work with more than 19 spots so you need to set the order to 3 to be able to proceed. The grid order for a hexagonal mask is simply the number of spots along the hexagon edge.



Sensor parameters

PICTURE SOURCE

● file

○ camera

REFERENCE GRID

● hexagonal

○ square

○ get from picture

☐ fast search

☐ ignore defocus

APERTURE

● auto

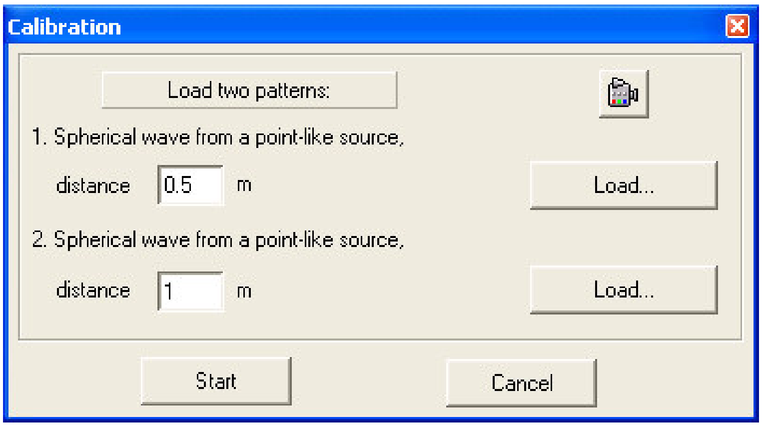
○ define manually

Maximum number of spots	19	
Reference grid order	3	
Reference grid pitch	0.875	mm
Pixel width	0.025	mm
Pixel height	0.025	mm
Smoothing filter size	10	pixels
Mask-to-CCD distance	20	mm
Number of Zernike terms	10	
SVD max-to-min ratio	100	
Radius of the reconstruction area	0.95	
Wavelength	0.63	mic
Neglect spots with intensity <	0.33	maximum
Neglect spots with displacement >	0.9	avg. spacing

Figure 3.8: *Setting the parameters for the calibration procedure.*

Make sure the wavelength is defined, if not – please enter the wavelength in the wavelength window. The processing is possible even with undefined wavelength, but then the interferometric pattern and the far field distributions are not calculated.

Click on “Calibrate” button, calibration menu will appear:



Calibration

Load two patterns:

1. Spherical wave from a point-like source,

distance 0.5 m

2. Spherical wave from a point-like source,

distance 1 m

Figure 3.9: *The calibration window.*

Set the right distances (0.5m and 1m) and load consequently two calibration files (cal50.bmp and cal100.bmp). After both patterns are loaded, push “Start” button in the calibration window. As a result the program will show the spot locations on both loaded patterns - see Fig 3.10.

Push “OK” button in the success message window and go to “Options”, “parame-

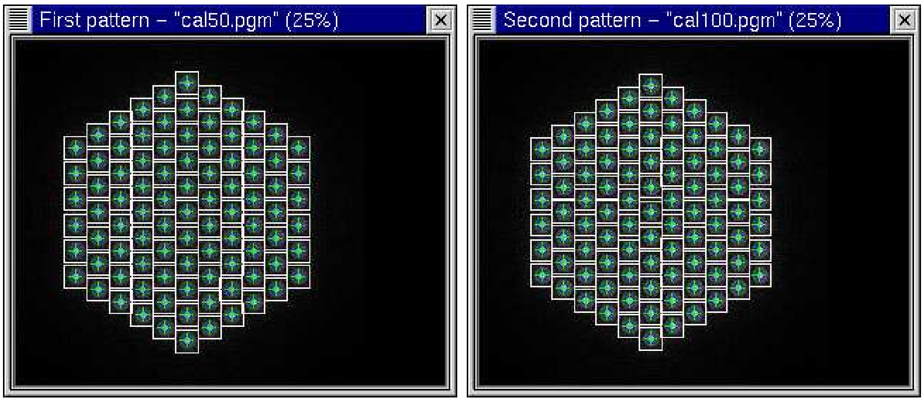


Figure 3.10: *Successful calibration results in framing of all spots in both calibration patterns with ideal grid superimposed.*

ters" again. As a result of calibration, the program has calculated the right pixel pitch of the CCD, the right reference grid pitch and right mask-to-CCD distance. Set "Sizing of the reconstruction area" to 0.95 and "Number of Zernike terms" to 10 and "Smoothing filter size" to 10. Make sure the button "Ignore defocus" under "reference grid" is not pressed. Go to "File", "Open main" and load the file cal50.bmp again.

There are two parameters important for successful calibration: the "Reference grid order" and the "Reference grid pitch". Other parameters (Number of spots, CCD pixel X and Y pitch and CCD-to-mask distance are calculated as a result of calibration.

Now you can just push the blue arrow button on the button bar and observe the reconstructed wavefront represented as surface plot, interferometric pattern and far field distribution - see Fig. 3.11:

The wavefront reconstructed has a strong spherical component. To be able to reconstruct the fine structure, ignoring the spherical component - go to "Options" Parameters" and select "Ignore defocus" option. It resets the whole working field and the processing should be repeated by pushing the blue arrow button. The result reveals weak initial aberration of the calibration beam with spherical component completely removed.

Now you may continue with wavefront reconstruction examples given in Section 3.6.

Also, please note the following.

- If FrontSurfer is shipped with a wavefront sensor, you can skip the calibration procedure by loading of calibration parameters from the calibration file supplied on the CD. If you have to recalibrate after changing the mask (array), please specify correct mask geometry.
- If you use a large mask (array) with sub-apertures filling the whole area of the image sensor, we recommend calibration according to the following procedure.

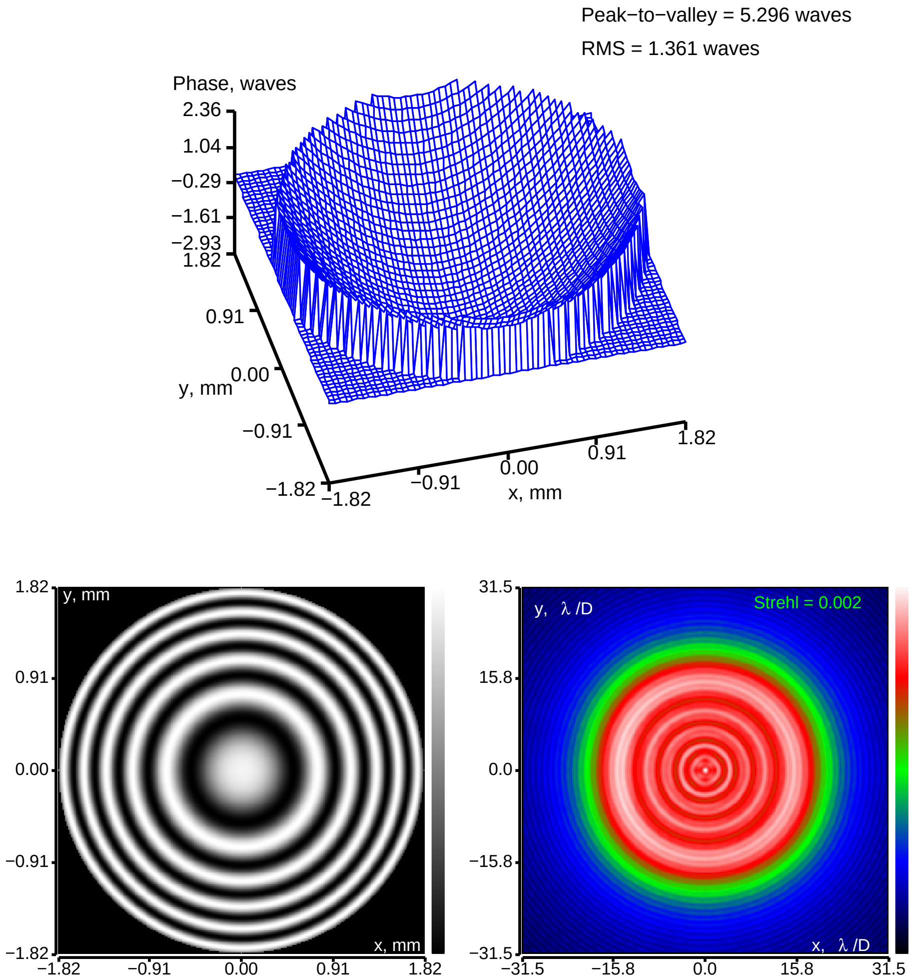


Figure 3.11: *Reconstructed spherical wavefront of the calibration source.*

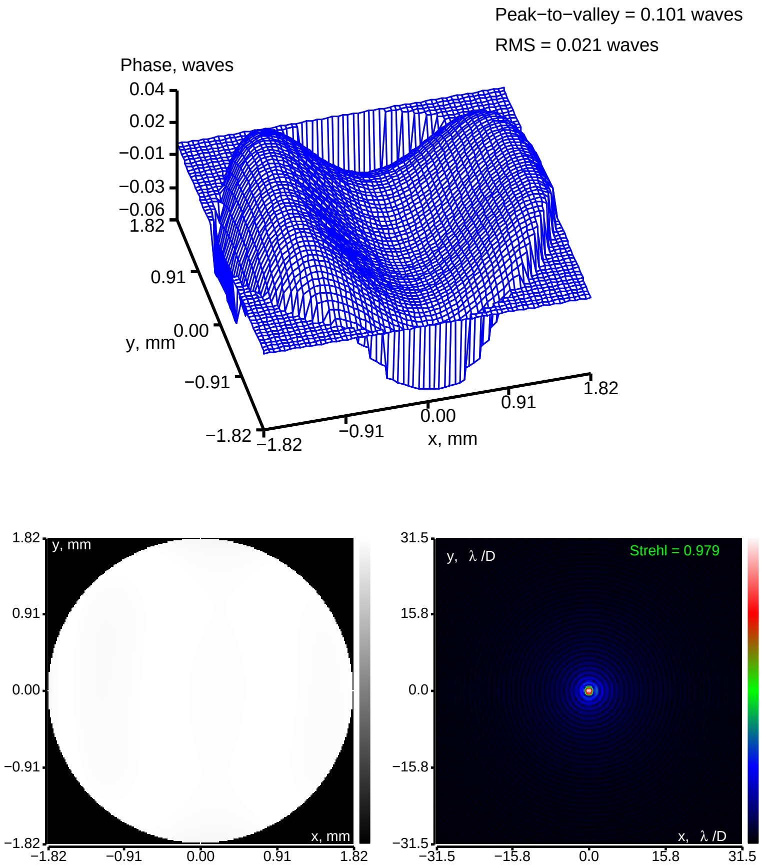


Figure 3.12: *Reconstructed wavefront of the calibration source with ignored spherical component.*

1. In the dialog box “Sensor parameters”, press the button “Aperture” and choose “define manually”; then choose “circular” if you have a hexagonal mask or “square” if you have a square one.
2. In the same dialog box, choose for hexagonal or square reference grid and define the grid order according to the number of spots you wish to use for the calibration.
3. Press “Calibrate” and capture two calibration patterns, as described higher in this section.
4. Define the aperture on one of the patterns, as described in section 3.6.3 The aperture should be as close as possible to a complete complete hexagonal and square structure (see Figure 3.4).
5. Press “Start” to complete calibration.

3.6 Examples of wavefront reconstruction

To demonstrate how FrontSurfer operates in different modes, we shall continue with the sample files distributed with FrontSurfer in the directory “sample”. Please use the calibration data obtained in Section 3.5 to process these samples.

3.6.1 Absolute measurement mode

After the calibration with the sample files, you are in the absolute measurement mode (Section 3.3.1). For an example of a real aberration, load the file “3_3_4_0.bmp”. This pattern corresponds to a mixture of spherical aberration and trifoil. Push the big blue button and observe the reconstruction progress - see Fig. 3.13.

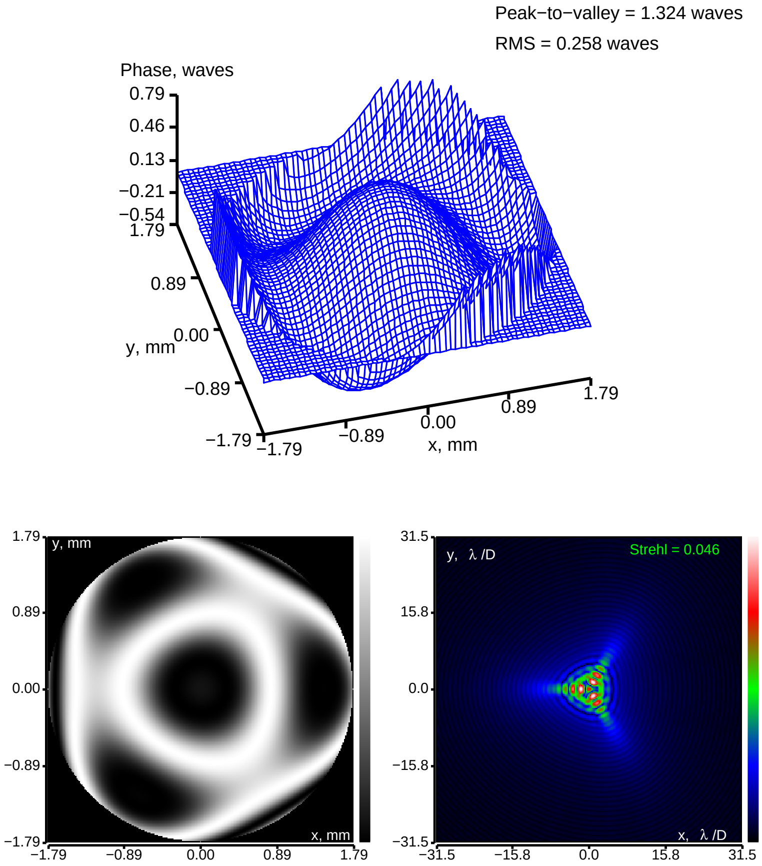


Figure 3.13: *Reconstructed wavefront with a mixture of spherical aberration and trifoil.*

Report file

To see the summary results of reconstruction and wavefront analysis, go to the menu command “Processing ⇒ Show report”. The final report contains parameters of the reconstructed wavefront, Zernike coefficients, primary Seidel aberration coefficients and the address of Flexible Optical BV:

```
Test report, generated on
Mon Jul 09 04:43:44 PM
---------------------------------------------------------------

Parameters:
---------------------------------------------------------------
Decomposition area diameter = 3.675000e+000 mm
Aperture: circular, auto
Diameter = 3.491250e+000 mm
Reconstructed from 19 wavefront slopes
Wavelength = 6.300000e-001 mic
Strehl factor: 7.379608e-003
Phase peak-to-valley = 2.028999e+000 waves
Phase RMS = 3.952469e-001 waves
Distance to focus = -9.176207e+000 m

Zernike coefficients:
---------------------------------------------------------------
C[1,1] = 0.000000e+000 waves  (tip)
C[1,-1] = 0.000000e+000 waves  (tilt)
C[2,0] = -1.460128e-001 waves  (focus)
C[2,2] = -3.969286e-002 waves  (astigmatism)
C[2,-2] = 6.916365e-002 waves  (astigmatism)
C[3,1] = -1.775043e-001 waves  (coma)
C[3,-1] = -1.740987e-001 waves  (coma)
C[3,3] = 1.248089e-002 waves  (trifoil)
***C[3,-3] = -8.522331e-001 waves  (trifoil)
C[4,0] = 8.020112e-001 waves  (spherical aberration)

Zernike polynomials are in the form given in
the manual of FrontSurfer.
Decomposition is made over the decomposition
area (outer red circle).

Seidel aberrations:
---------------------------------------------------------------
Tilt = 0.000000e+000 waves
    {C*[1,1]^2 + C*[1,-1]^2}^1/2
```

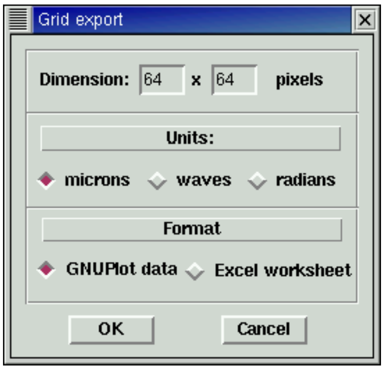
```
Focus = -6.887590e-001 waves
    2C*[2,0]
Astigmatism = 1.433782e-001 waves
    2{C*[2,2]^2 + C*[2,-2]^2}^1/2
Coma = 6.352972e-001 waves
    3{C*[3,1]^2 + C*[3,-1]^2}^1/2
Spherical aberration = 3.910605e+000 waves
    6C*[4,0]

Seidel coefficients are calculated by integration
over the aperture.

(c) Flexible Optical BV
---------------------------------------------------------------
Polakweg 10-11,
2288 GG Rijswijk ZH, The Netherlands
fax: +31-70-2629420
phone: +31-70-7101400
e-mail: oko@okotech.com
web site: http://www.okotech.com
```

Note the difference between the decomposition area and the reconstruction area (aperture). Wavefront decomposition by Zernike polynomials is performed over an area inside the circle surrounding all spots in the reference grid; this area is shown as an outer red circle at the main hartmannogram picture. To reduce the reconstruction errors in the peripheral area when increasing the number of Zernike polynomials, the wavefront can be reconstructed over an aperture of a smaller radius (0.8. . . 0.9 of the decomposition aperture radius). The radius of the reconstruction area (aperture) relative to the radius of the decomposition area is defined in “Options” ⇒ “Parameters” ⇒ “Sizing of the reconstruction area”.

All Zernike coefficients in the final report correspond to the polynomials defined over the *decomposition* area, whereas Seidel aberrations are obtained from the reconstructed phase surface by integration over the *aperture* (reconstruction area). In case when the wavelength is not defined, all coefficients are expressed in microns. Specification of the Zernike polynomials used can be found on page 31.

Saving the results and graphic formats

To save ALL results at once go to “File” “Save output”. You can save all results by marking “Everything” and pushing the button”Save”. You need to define the filename, the program will save all available information by creating 10 files in bmp and postscript format, including report file “..._report.txt” and wavefront grid “..._wf.txt”. The wavefront grid file is saved as grid file in Gnuplot (one data point per line, blank line as row separator) or in Microsoft Excel worksheet format (one row per line, data points are separated by spaces). The surface deviation is given in micrometers, waves or radians. To export the surface data, choose “Wavefront grid” and the dialog box will appear. The wavefront profile can be saved as a grid of any dimension in microns, waves or radians - for direct export to beam propagation software such as LightPipes - see Fig. 3.14. FrontSurfer uses the latest values of these options for further batch export operations.



Grid export

Dimension: 64 x 64 pixels

Units:

● microns ○ waves ○ radians

Format

● GNUPlot data ○ Excel worksheet

Figure 3.14: *Wavefront export menu.*

To finish the program go to “File” “Exit”. The program will save all current operation parameters and calibration data to files “∼/.fsurfer1” and “∼./fsurfer2” (Linux version) or in “fsurfer.ini” file in Windows directory (Windows version).

Saving of the results is disabled in the demo version.

3.6.2 Reference mode

Under the reference measurement mode (see Section 3.3.2), FrontSurfer can be operated with masks (microlens arrays) with arbitrary geometry of sub-apertures. All light spots should have approximately equal size and equal integral intensity.

Go to “Options” “Parameters” and check “hexagonal” under “reference grid”. Load and reconstruct the file “irr_main.bmp”. The reconstruction result contains an irregular aberration, which is the result of using mis-formed Hartmann mask - see Fig. 3.15.

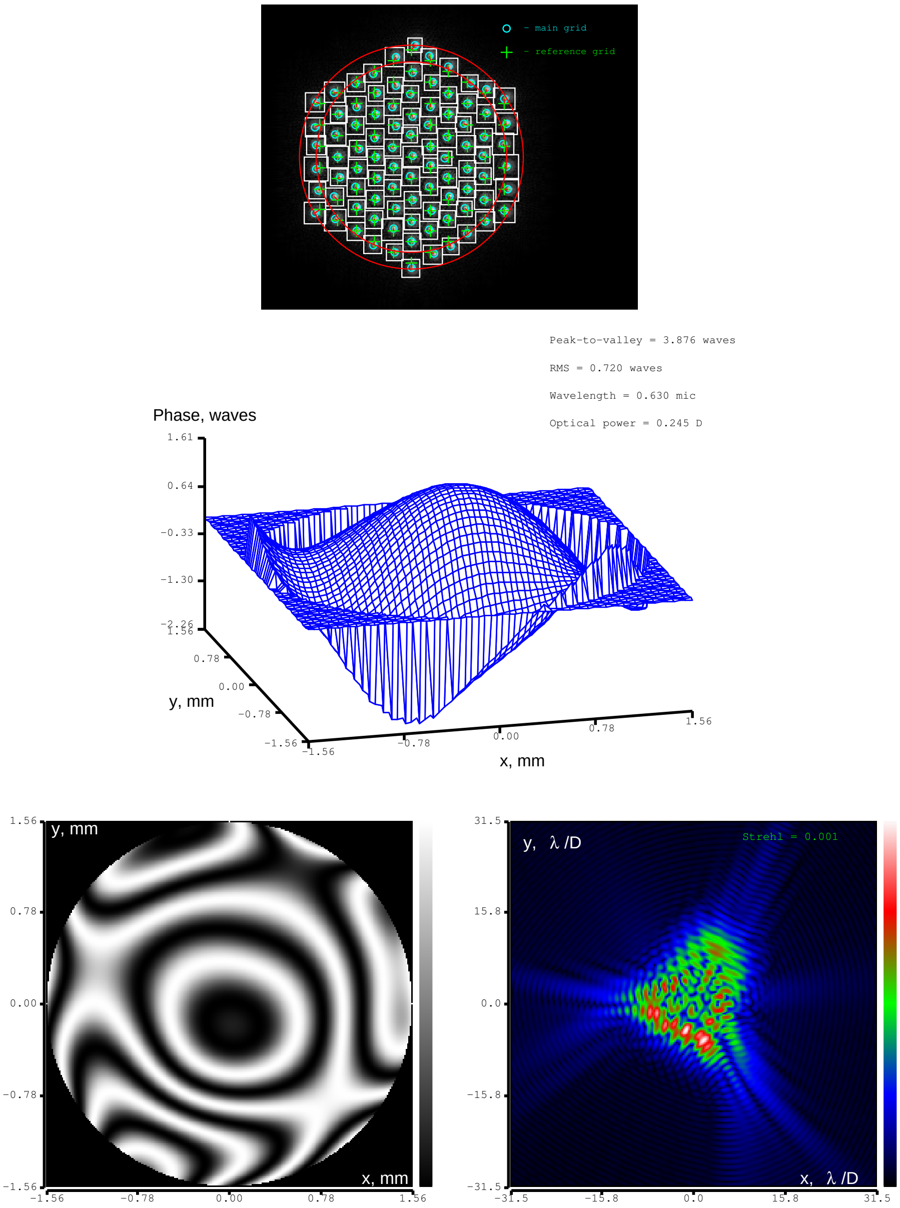
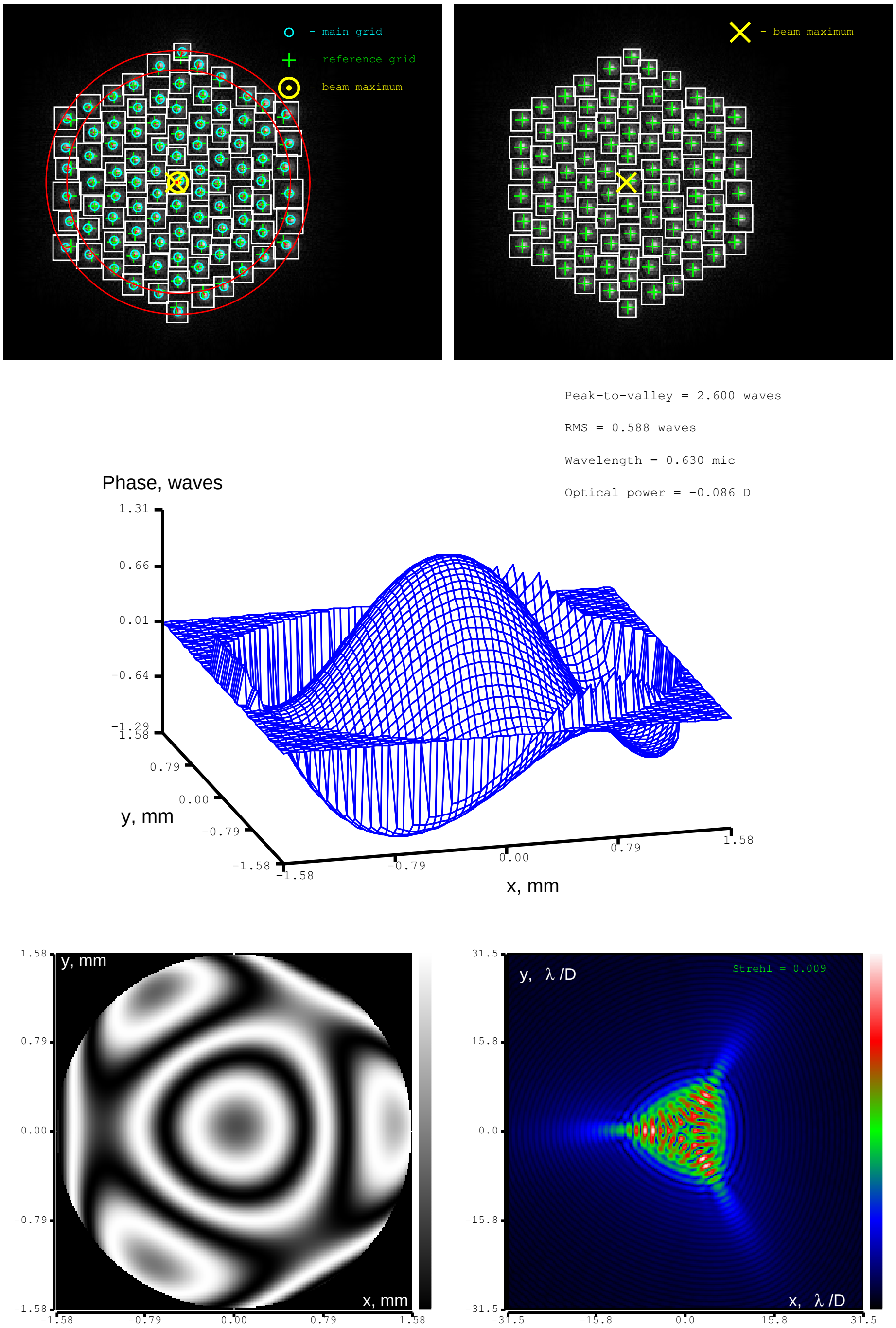


Figure 3.15: *Erroneous wavefront reconstruction with irregular hartmann mask.*

To correct for the mask irregularity, we need to switch to the reference mode and use a correct reference grid. In a real experiment, it can be obtained by illuminating the mask with an ideal (collimated) light beam.

To switch to the reference mode, go to “Options” “Parameters” and check “get from picture” button. Go to “File”, “Open reference” and load the file “irr_ref.bmp”. Alternatively, it can be done using the toolbar button marked by “2”. This file corresponds to “reference” mis-formed Hartmann mask. Now go to “File” “Open main” and load file “irr_main.bmp”, corresponding to the aberrated beam. Push Blue arrow processing button. The result shown in Fig. 3.16 reveals mixture of trifoil and spherical aberration, exactly as in the previous example with internally generated grid.



Peak-to-valley = 2.600 waves

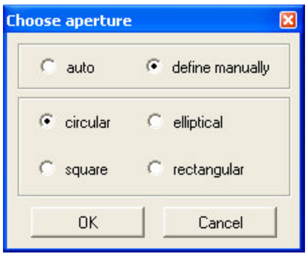
RMS = 0.588 waves

Wavelength = 0.630 mic

Optical power = -0.086 D

Figure 3.16: *Reconstruction using a reference grid generated from irregular mask.*

3.6.3 Reconstruction with manually defined aperture



Choose aperture

○ auto ● define manually

● circular ○ elliptical

○ square ○ rectangular

OK Cancel

Figure 3.17: *"Choose aperture" dialog box.*

As mentioned in Section 3.3.2), the area of interest (aperture) in a hartmanngram can be defined manually. To switch to this mode, go to "Options" "Parameters", go to the section "Aperture" and click the button "Change". In the dialog box "Choose aperture" (Figure 3.17), select the options "define manually" and "circular". Press "OK".

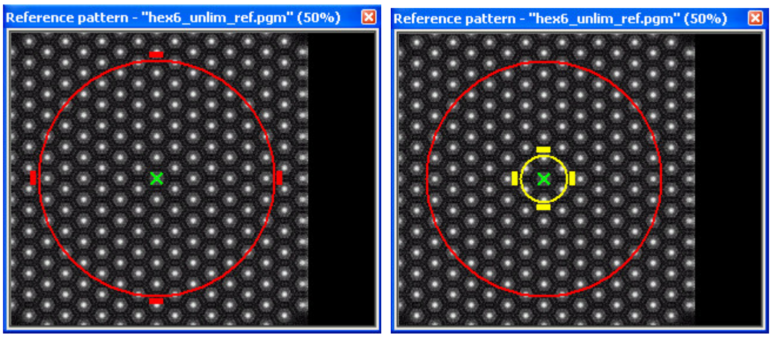


Figure 3.18: *Manually defined aperture (left) and central obscuration (right).*

Go to "File", "Open reference" (or use button "2" of the toolbar) and open the file "man_aper_ref.bmp". Click on the reference picture and define the aperture by dragging the cursor. It will be displayed as a red circle. After the aperture is defined, it can be edited. Dragging the cross mark in the aperture center allows moving the whole aperture. To modify the radius, first select it by double-clicking inside the aperture, and four marks will appear at its edge (see Figure 3.18, on the left). Now you can change the radius by dragging the marks. To delete the aperture, select it and press "Delete" button. In a similar way, you can create and modify an elliptical, square or rectangular aperture.

You can also define a circular obscuration in the center. Click inside the aperture and drag the cursor. The central obscuration will be displayed as a yellow circle. It can be edited in a similar way as the aperture (see Figure 3.18, on the right).

Go to "File", "Open main" (or use button "1" of the toolbar) and open the file "man_aper_main.bmp". For your convenience, the aperture and central obscuration will be also displayed on the main picture; they can also be edited on the main picture. Press the blue arrow button to proceed. The program will locate spots

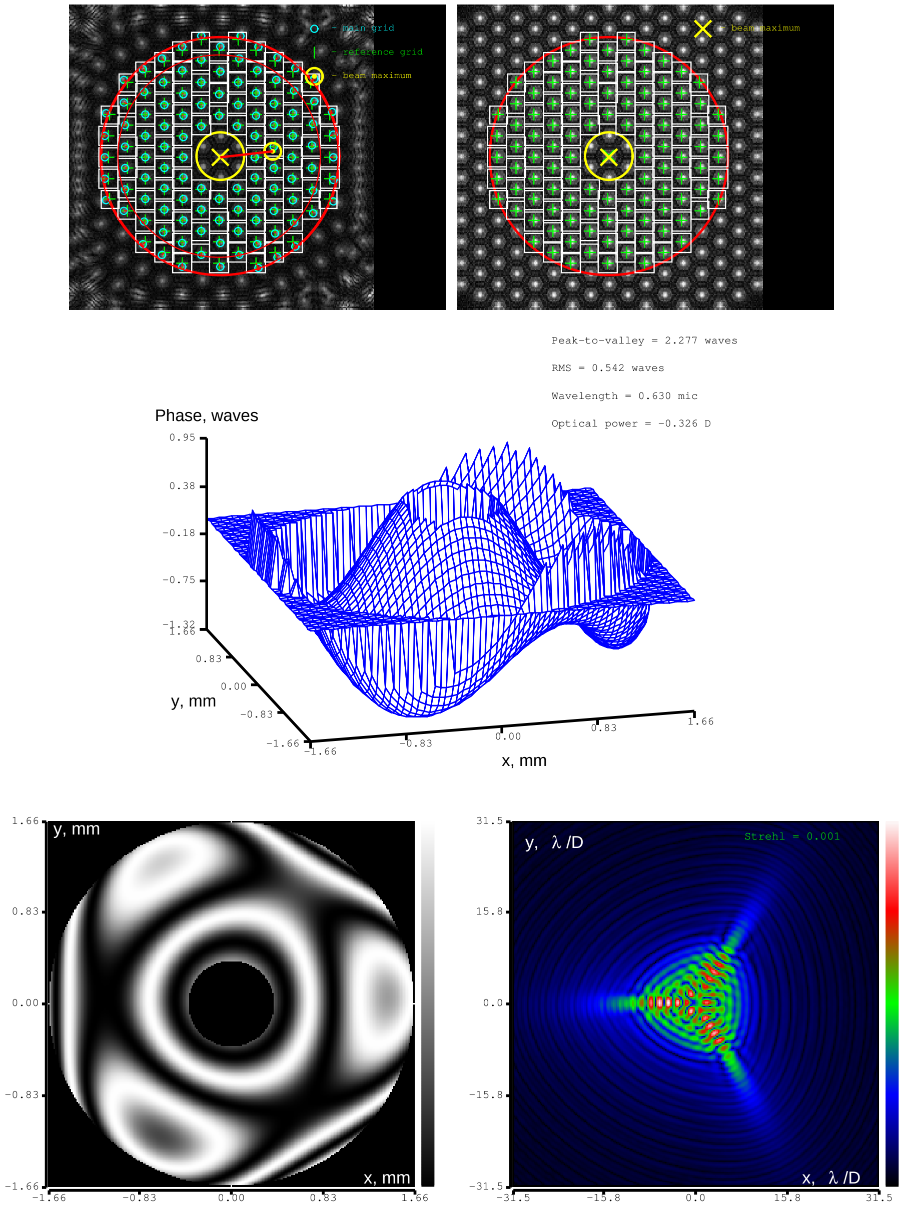


Figure 3.19: *Wavefront reconstruction with manually defined aperture and central obscuration.*

inside the aperture but outside the central obscuration. The reconstruction result is shown in Figure 3.19.

To reduce the reconstruction errors in the peripheral area when increasing the number of Zernike polynomials, the wavefront can be reconstructed over an aperture of smaller size. Relative size of the wavefront reconstruction area with respect to the manually defined area of interest is defined in “Options” ⇒ “Parameters” ⇒ “Sizing of the reconstruction area”.

3.7 Alignment mode

The sensor can be used for alignment of optical setups. In this mode, low-order aberrations and displacement of the center of the intensity distribution will be dynamically displayed. In case the reference grid is generated from the reference image, the program will use the last reference acquired before the alignment mode is initiated.

No other measurements is possible in the alignment mode.

To start the alignment mode:

1. Go to the processing menu.

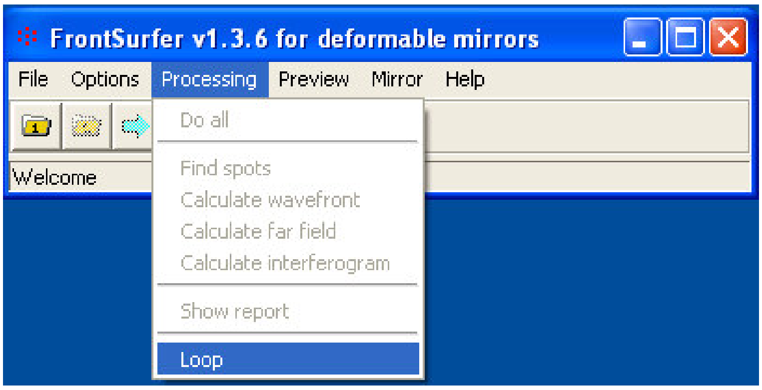
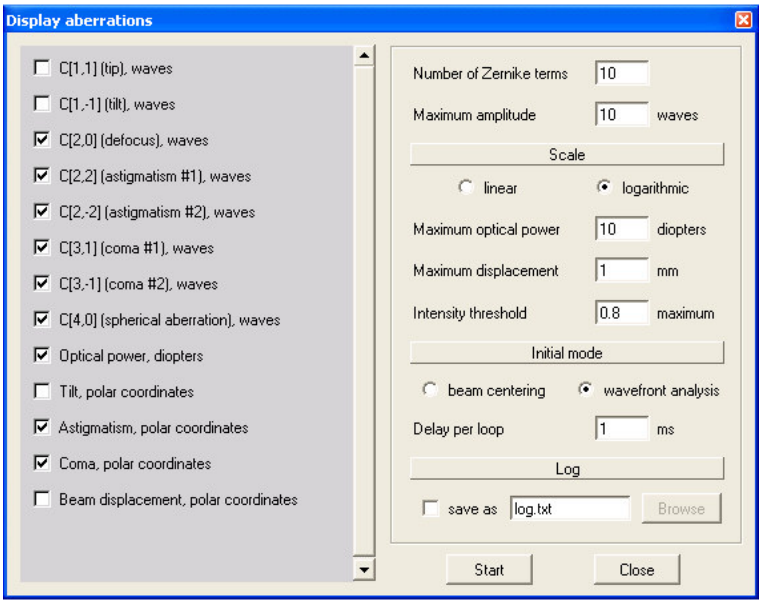


Figure 3.20: *The processing menu*

2. Choose the “Loop” option - the display menu will pop up:


Display aberrations

☐ C[1,1] (tip), waves
☐ C[1,-1] (tilt), waves
☑ C[2,0] (defocus), waves
☑ C[2,2] (astigmatism #1), waves
☑ C[2,-2] (astigmatism #2), waves
☑ C[3,1] (coma #1), waves
☑ C[3,-1] (coma #2), waves
☑ C[4,0] (spherical aberration), waves
☑ Optical power, diopters
☐ Tilt, polar coordinates
☑ Astigmatism, polar coordinates
☑ Coma, polar coordinates
☐ Beam displacement, polar coordinates

Number of Zernike terms 10
Maximum amplitude 10 waves

Scale
○ linear ● logarithmic

Maximum optical power 10 diopters
Maximum displacement 1 mm
Intensity threshold 0.8 maximum

Initial mode
○ beam centering ● wavefront analysis

Delay per loop 1 ms

Log
☐ save as log.txt

Figure 3.21: *Aberration menu in the alignment mode*

3. Mark the aberrations you wish to display and adjust the parameters of display.

Zernike aberrations (up to Z[4,0]) and optical power are displayed in bar-graph format. Tilt, astigmatism, coma and displacement of the intensity center relative to the reference picture ("Beam displacement") can be displayed in polar coordinates.

The value of **"Number of Zernike terms"** defined in the alignment mode can be different from those defined for the single measurement mode (see page 66). For higher speed it can be set to 10; however, for higher precision we recommend to make it equal to the value defined for the single measurement mode.

"Maximum amplitude" defines the scale limits for Zernike and Seidel aberrations.

The **scale** type for the bar-graph display can be chosen between **"linear"** and **"logarithmic"** using the corresponding radio buttons.

"Maximum optical power" defines the scale limits for the optical power.

"Maximum displacement" defines the scale limits for the beam displacement.

"Intensity threshold" defines the threshold level relative to the intensity maximum. When the beam displacement is calculated, intensities lower than the intensity threshold are not taken into account.

If you set the **"Initial mode"** option as **"beam centering"** then the only characteristic evaluated in the alignment mode will be the beam displacement, and evaluation will be performed with maximum possible speed. This option is useful for beam centering. You may easily switch between faster "beam centering" mode and slower "wavefront analysis" mode using radio buttons.

By adjusting the **"Delay per loop"** option you may control the value of time delay added at the end of each measurement loop.

If you select the "log" options, the measurement results will be also saved in a log file.

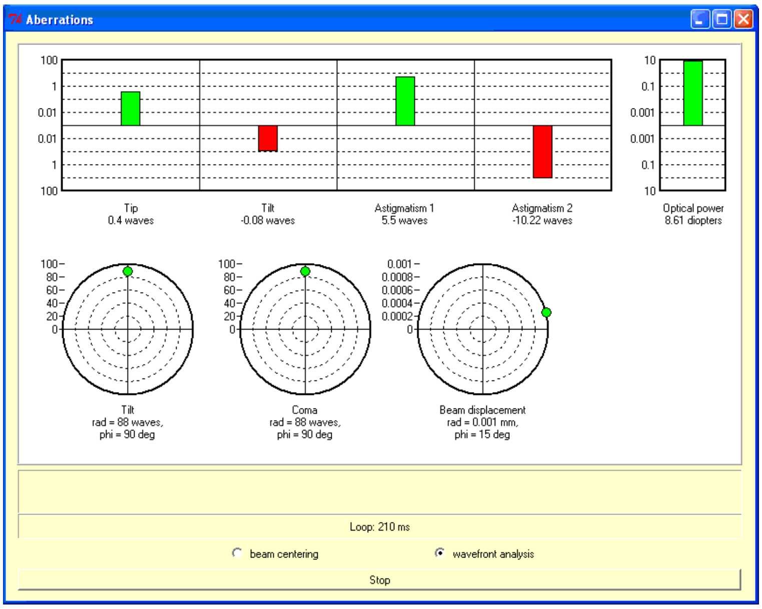
Push the "Start" button. The aberration chosen will be dynamically displayed.

The speed of the dynamic display depends on:

1. The speed of computer used

2. The frame resolution. 320x240 capture mode will produce higher refresh rate than 640x480 capture window. To change the capture mode go to "Options" menu and choose " camera", then "properties" and choose the resolution you wish. Note that you need to recalibrate the sensor for each resolution used.

3. The number of sub-apertures in the hartmann mask. It is not recommended to use high-resolution masks in the alignment mode - they have smaller dynamic range and large number of spots will slow down the processing.



Tip
0.4 waves

Tilt
-0.08 waves

Astigmatism 1
5.5 waves

Astigmatism 2
-10.22 waves

Optical power
8.61 diopters

Tilt
rad = 88 waves,
phi = 90 deg

Coma
rad = 88 waves,
phi = 90 deg

Beam displacement
rad = 0.001 mm,
phi = 15 deg

Loop: 210 ms

○ beam centering ● wavefront analysis

Figure 3.22: *"Aberrations" window in the alignment mode*

3.8 Operation with adaptive optics correction loop

The features described in this section are available only in FrontSurfer version for deformable mirrors (see page 9). This version can be interfaced with deformable mirrors and can be used for their open-loop and closed-loop control. Dynamic correction of optical aberrations using mirrors from OKO Technologies can be implemented with loop frequency up to 50 Hz and higher, depending on configuration.

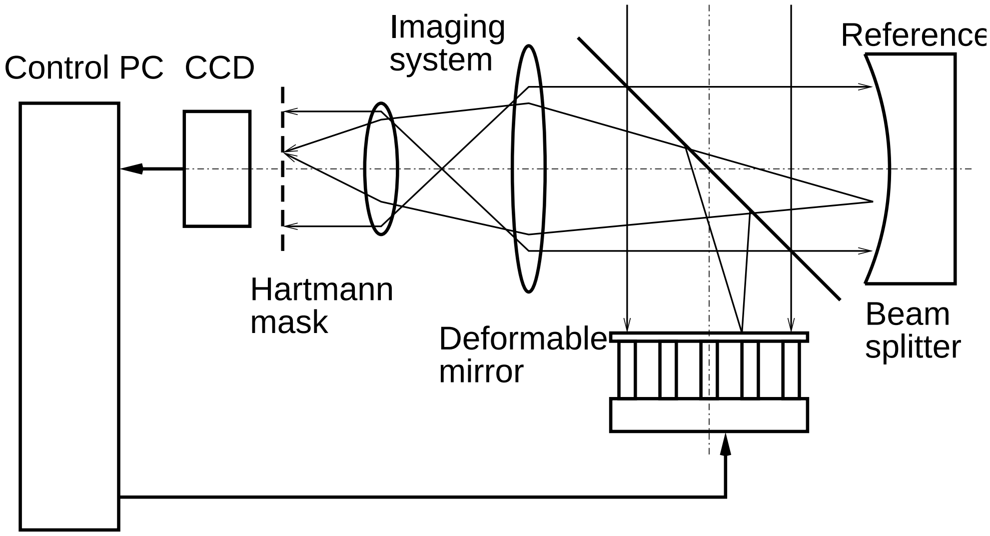


Figure 3.23: *Scheme of typical adaptive optics setup.*

A typical setup for functional feedback loop is shown in Fig. 3.23. Flat or spherical reference mirror is used for calibration purposes. Fig. 3.23 gives only an example including all essential elements, in practice the reference can be removed from the system after reference image is grabbed and saved to disk.

Please refer to “Deformable mirror technical passport” for more information about operation and setup of the deformable mirror.

3.8.1 Mirror configuration

To set up the deformable mirror interface go to “Mirror” and then “Configuration”, as shown in Fig. 3.24.

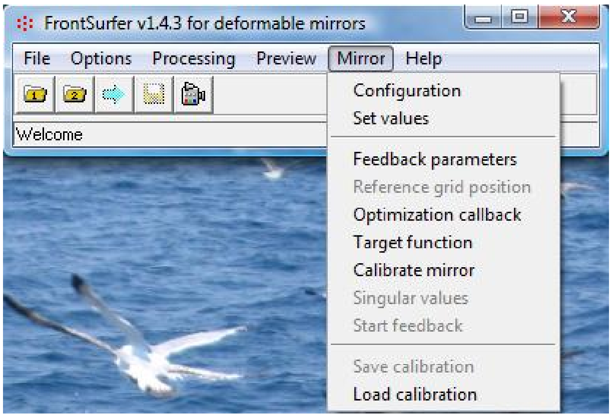


Figure 3.24: *Deformable mirror menu*

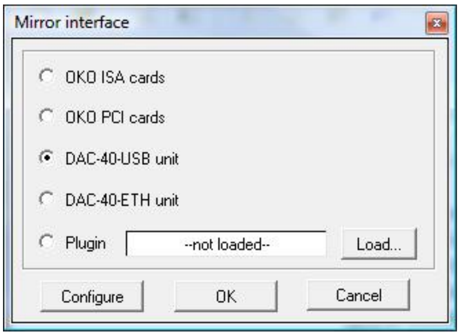
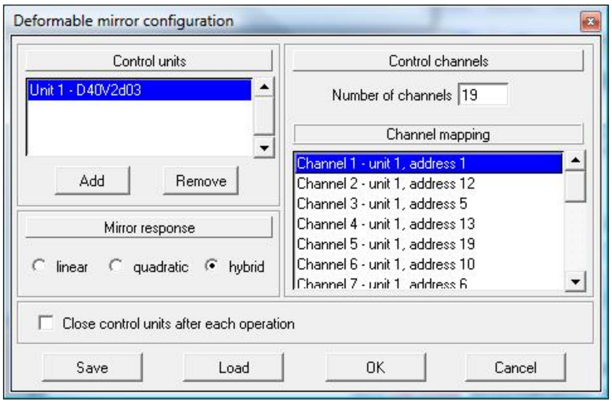


Figure 3.25: *Deformable mirror interface configuration*

In the “Mirror interface” dialog box (Fig. 3.25) you can choose the interface type for operation with deformable mirrors. Five interfaces are supported, namely, 8-bit digital ISA and PCI boards, 12-bit DAC40USB (USB) control units, 16-bit DAC40ETH (Ethernet) control units and custom plugin interface.

PCI and ISA interfaces

In the “Deformable mirror configuration” dialog box shown in Fig. 3.26 you can define base I/O addresses of the control boards involved and set the correspondence



Unit 1 - D40V2d03

Number of channels 19

Channel 1 - unit 1, address 1
Channel 2 - unit 1, address 12
Channel 3 - unit 1, address 5
Channel 4 - unit 1, address 13
Channel 5 - unit 1, address 19
Channel 6 - unit 1, address 10
Channel 7 - unit 1, address 6

○ linear ○ quadratic ● hybrid

☐ Close control units after each operation

Figure 3.26: *Deformable mirror channels configuration*

between board outputs and mirror actuators. Please use the pin-out tables from the mirror reference manual for setting these values. If FrontSurfer is shipped with a deformable mirror, configuration can be loaded from the mirror configuration file, which is normally provided on the installation CD.

Option “Mirror response” should be set to “quadratic” for mirrors with quadratic response (thermal and membrane mirrors), “linear” for ones with linear response (mirrors with piezoelectric control), and “hybrid” for MMDMs integrated with a piezo tip-tilt stage. In a hybrid configuration, the last two channels correspond to piezo actuators.

USB interface

Configuration of USB interface (40-channel 12-bit DAC40USB units) is similar to that of ISA/PCI interfaces; the only difference is that the serial number is used instead of the base I/O address to identify the control unit. The serial number can be found on a sticker attached to the USB unit. An additional option is provided, which allows to close the device handle after each operation. It may be needed for the case when the operation of the USB unit causes interaction with other devices, such as Firewire cameras connected to the same USB/Firewire interface card. Normally, this option should be off.

Ethernet interface

Configuration of Ethernet interface (40-channel 16-bit DAC40ETH units) is similar to that of ISA/PCI interface; the only difference is that the MAC address of the unit is used instead of the base I/O address to identify the control unit. The MAC address can be found on a sticker attached to the USB unit or detected with the test program supplied with DAC40ETH.

Custom plugin interface

Source code of the mirror plugin template (file “mirror_plugin.zip” in FrontSurfer installation directory) is now supplied with FrontSurfer. In Linux it can be compiled with GNU C, in Windows - with Microsoft Visual C++. Corresponding dynamically loaded library should export 5 functions declared as follows.

```
void properties();

int get_n_chan();

int init_mirror();

int close_mirror();

int set_mirror(double* volt_array);
```

Function “properties” is used to display a dialog box to change configuration of the mirror.

Function “get_n_chan” returns number of channels for the current mirror configuration. Returned zero or negative value indicates an error.

Function “init_mirror” and “close_mirror” are used for initialization and release of the mirror interface. They should return 0 if the operation was successful, otherwise they should return -1.

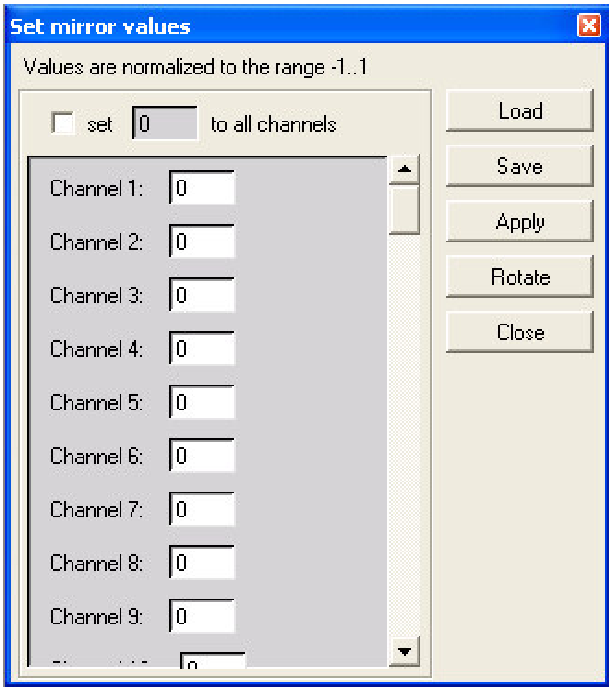
Function “set_mirror” sets values from array “volt_array” to the mirror. Number of items in the array is determined by the result returned by “get_n_chan” function. Control values should be normalized to the range $-1 \ldots 1$. The function should return 0 if the operation was successful, -1 if not.

3.8.2 Mirror control

The algorithms used by FrontSurfer are insensitive to the order in which the mirror channels are addressed, but make sure that all channels are addressed because each missing channel will strongly influence the quality of correction.

After the mirror is configured, you can go to “Set values” to set the voltages on the mirror actuators (Fig. 3.27). Setting “0” to all channels corresponds to bias of the mirror, which is the starting point for optimization of its shape. The range of control values is limited to [-1...1].

“Rotate” button will apply maximum voltage to all actuators sequentially, one by one, for 0.5 s per actuator. This is useful for diagnostics of the mirror and control electronics.



Channel 1: 0
Channel 2: 0
Channel 3: 0
Channel 4: 0
Channel 5: 0
Channel 6: 0
Channel 7: 0
Channel 8: 0
Channel 9: 0

Figure 3.27: *Mirror voltages dialog box*

3.8.3 Feedback loop operation

In feedback loop operation mode FrontSurfer uses a set of measured influence functions of the mirror for fitting of the desired phase aberration. This aberration is specified relative to the reference grid as a sum of Zernike polynomials; all types of the reference grid - hexagonal, square and those obtained from a reference picture - can be used.

FrontSurfer performs wavefront correction in a series of iterations. If the residual aberration ϕ_n at the n-th iteration corresponds to the set of actuator signals $\mathbf{X}_n$ then the actuator signals at the next step $\mathbf{X}_{n+1}$ will be determined by the expression

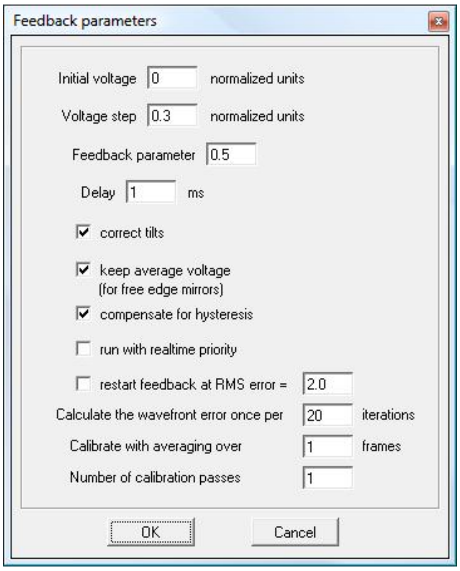
$$\mathbf{X}_{n+1} = \mathbf{X}_n - g\mathbf{A}^{-1}\phi_n,$$

where g is the feedback coefficient with a value in the range (0..1], $\mathbf{A}$ is the influence matrix of the mirror, $\mathbf{A}^{-1}$ is its pseudo-inverse given by

$$\mathbf{A}^{-1}=\mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T,$$

$\mathbf{U}$, $\mathbf{S}$ and $\mathbf{V}$ are the singular value decomposition of $\mathbf{A}$ which is $\mathbf{A}=\mathbf{U}\mathbf{S}\mathbf{V}^T$ [6]. The columns of the matrix $\mathbf{U}$ make up orthonormal set of the mirror deformations (modes), and the values of the diagonal matrix $\mathbf{S}$ represent the gains of these modes. Discarding those modes having small singular values may improve controllability of the system.

To adjust parameters of the feedback loop operation mode, select the “Feedback parameters” from the “Mirror” pull-down menu, and a dialog box (Fig. 3.28) will appear.



Feedback parameters

Initial voltage 0 normalized units

Voltage step 0.3 normalized units

Feedback parameter 0.5

Delay 1 ms

☑ correct tilts

☑ keep average voltage
(for free edge mirrors)

☑ compensate for hysteresis

☐ run with realtime priority

☐ restart feedback at RMS error = 2.0

Calculate the wavefront error once per 20 iterations

Calibrate with averaging over 1 frames

Number of calibration passes 1

Figure 3.28: *Feedback parameters dialog box*

"Initial voltage" defines the initial state of actuator voltages during the calibration and in the beginning of the closed-loop control. Its range is limited to [-1..1]; normally, it should be zero.

"Voltage step" defines the voltage step, which is used for measurement of the mirror's influence functions during the calibration. The influence function for the k-th actuator is measured for "Initial voltage"+"Voltage step" applied to the k-th actuator and "Initial voltage" applied to other ones.

"Feedback parameter" allows to adjust the value of g.

"Delay" sets the interval of time which is required for the mirror to turn on completely.

"Correct tilts" should be turned on if the deformable mirror is supposed to compensate for tip and tilt. If this option is off, the tilts could be adjusted by the optimization procedure to provide the optimum compensation for higher order aberrations.

"Keep average voltage" option should be set if the free-edge mirror is used; it provides that the average value of voltage calculated over all actuators is kept the same.

"Compensate for hysteresis" option is used to provide correct calibration of mirrors with hysteresis, such as those with piezoelectric actuators. When this option is set then the shape of the mirror is measured twice during each step of the calibration - once with "Initial voltage" applied to all actuators and once with "Initial voltage"+"Voltage step" applied to one actuator and "Initial voltage" to other actuators.

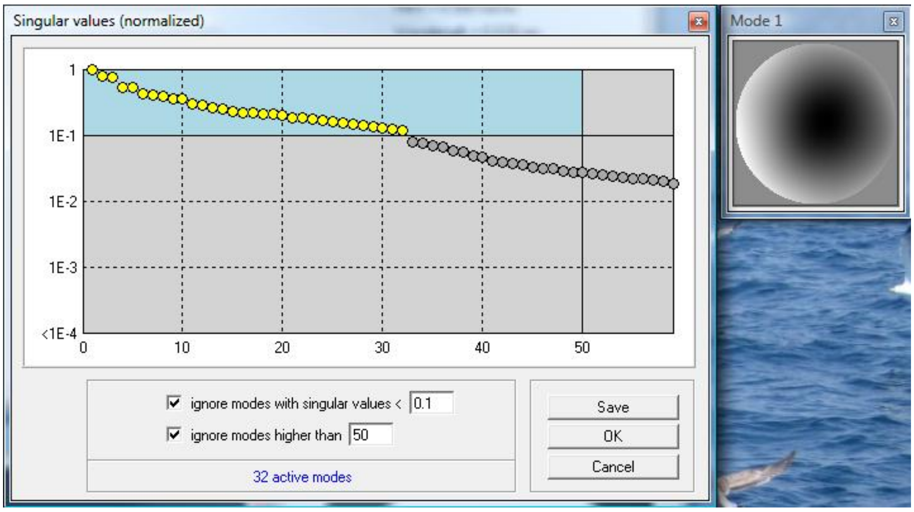
"Run with realtime priority" option allows to execute the correction loop in in a Windows thread with real-time priority class and highest priority. It prevents other processes from interrupting the correction loop. Multiprocessor or multiple-core PCs are recommended for higher performance. In such a system, the interface thread will be running on the 1st processor, and the correction loop on the 2nd one.

Option “Restart feedback at RMS error = ...” allows to restart feedback loop automatically as soon as the residual error reaches a certain threshold. It prevents the iterative process from divergence in case there is an obstruction in the beam or very strong turbulence.

“Calculate the wavefront error once per ... iterations” is used for diagnostics of the iteration process. It allows to specify how often the system will display Shack-Hartmann pattern and calculate the residual wavefront error. Doing it on every step would slow down the iteration process; we recommend setting this parameter to a reasonably high value depending on the available correction rate. Normally, RMS should be calculated 1 or 2 times per second.

“Calibrate with averaging over ... frames” is used for calibration in presence of the atmospheric turbulence. Averaging over 1000 and more frames allows to reduce its effects. In addition to this, the parameter “Number of calibration passes” allows to calibrate the system in several passes and average the results.

Before the adaptive correction can be started, the system should calibrate the mirror. This process consists of automatic measurement of the influence functions of the mirror and singular value decomposition (SVD) of the influence matrix. To initiate the process select the “Calibrate mirror” from the “Mirror” pull-down menu.

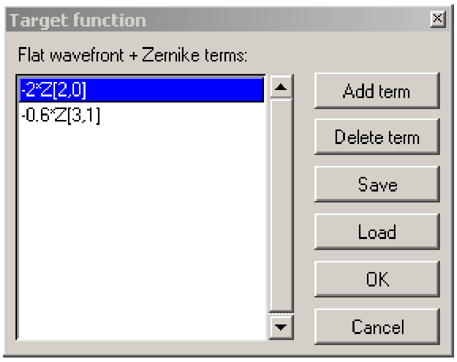


☑ ignore modes with singular values < 0.1
☑ ignore modes higher than 50
32 active modes

Figure 3.29: *“Singular values” dialog box*

To choose which modes should be used in the feedback loop operation, go to the menu “Mirror” and choose “Singular values”; this option becomes available after calibration. You may limit both the number of modes to use and the range of singular values to discard - see Fig. 3.29. By double-clicking on points corresponding to the singular values on a graph one can display the corresponding modes.

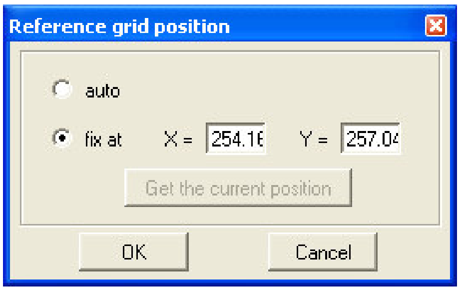
FrontSurfer can be used for generation of arbitrary phase aberrations represented as a sum of Zernike polynomials. These aberrations can be defined via menu command



Flat wavefront + Zernike terms:
-2*Z[2,0]
-0.6*Z[3,1]

Figure 3.30: *"Target function" dialog box*

"Mirror ⇒ Target function". In the dialog box "Target function" (Fig. 3.30) one may add, delete and modify certain Zernike terms by using the buttons and double-clicking items in the list. It is also possible to save the whole list in a file and load a previously saved target function.

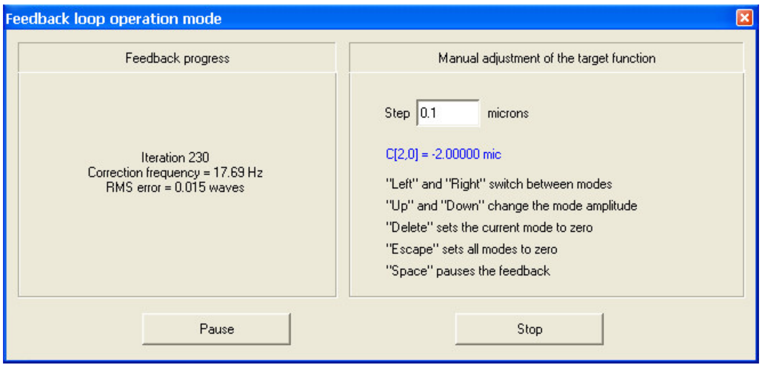


○ auto
● fix at X = 254.16 Y = 257.04

Figure 3.31: *"Reference grid position" dialog box*

If the option "remove tilts" (see Fig.3.28) is activated in the absolute measurements mode, the computed reference grid should be fixed to a certain position, which corresponds to zero tip and tilt. The menu command "Mirror ⇒ Reference grid position" allows to specify how the reference grid will be fixed. This command invokes the dialog box "Reference grid position" - see Fig.3.31. If the option "auto" is chosen then the reference grid position will be determined by the initial state of the mirror during the calibration. Alternatively, it can be fixed at a permanent position on the intensity pattern, which is expressed in pixel units (option "fix at…"). Pressing the button "Get the current position" will fix it to the reference position of a currently reconstructed wavefront (if it is available).

To start the adaptive correction loop, select "Start feedback" from the "Mirror" pull-down menu. The window shown in Fig. 3.32 will appear; it will display information about the average correction speed, the number of iterations passed and the residual error achieved. Besides, Shack-Hartmann pattern with superimposed subapertures will be displayed in a separate window. During the correction, the adaptive mirror minimizes the difference between the current and the target wavefront, even if changing aberrations are present.



Iteration 230
Correction frequency = 17.69 Hz
RMS error = 0.015 waves

Step 0.1 microns

C[2,0] = -2.00000 mic

"Left" and "Right" switch between modes
"Up" and "Down" change the mode amplitude
"Delete" sets the current mode to zero
"Escape" sets all modes to zero
"Space" pauses the feedback

Figure 3.32: *Feedback progress dialog box*

The target function can be adjusted during the feedback process through the computer keyboard; the "UP" key increases the amplitude of a certain Zernike mode, the "DOWN" key decreases the amplitude, and the "LEFT" and "RIGHT" keys are used to switch between the Zernike modes. "SPACE" button can be used to pause the feedback.

After the correction is stopped (it can be done using "Stop" button), you may measure and reconstruct the wavefront generated by the deformable mirror. The last set of the mirror voltages will be kept in the program. To see it, use the menu command "Mirror ⇒ Set values".

3.8.4 Callback plugin interface

Callback plugin interface allows monitoring of the closed-loop optimization process in FrontSurfer from an external DLL library and implementing custom optimization. Example source code for this kind of plugin (file "callback_plugin.zip" in FrontSurfer installation directory) is supplied with FrontSurfer. It can be compiled with Microsoft Visual C++. A callback plugin should export up to 5 functions declared as follows

```
void optimization_init();

void optimization_close();

void report_iter_result(double spots_shifts_rms, int bFeedbackOn);

void optimization_step(double spots_shifts_rms, double* voltages,
int n_actuators);

void report_rms_zernike(double rms, double* zernike_coefs,
int n_zernikes, int bFeedbackOn);
```

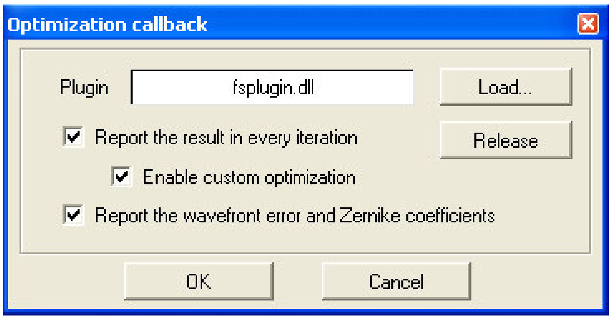
All these functions are optional.

The function "optimization_init" can be used for initialization of the user data; if it is present in the DLL, it will be invoked once before starting the feedback loop. "optimization_close" allows to release the data initialized in "optimization_init"; if it is present, it will be invoked once after the feedback is stopped.

The function "report_iter_result" allows monitoring the iteration process without external optimization. If this function is present and activated in FrontSurfer, it will be invoked in every loop, returning the rms deviation of the spot positions from target values (in millimeters). It is also reporthed whether the feedback is on or off.

The function "optimization_step" allows to implement custom optimization. Like "report_iter_result", it returns the rms deviation of the spot positions from target values in every loop. Additionally, it allows to adjust voltage values within the procedure and pass them to FrontSurfer, replacing FrontSurfer's internal optimization code.

The function "report_rms_zernike" allows monitoring the residual wavefront error and Zernike coefficients during the optimization. It will be invoked every time a new value of rms wavefront error value is available (by default, it is not calculated in every loop). The parameter "Calculate the wavefront error once per . . . iterations" (see Fig.3.28) allows to specify how often it will be calculated. It is also reporthed whether the feedback is on or off.



Plugin: fsplugin.dll

☑ Report the result in every iteration

☑ Enable custom optimization

☑ Report the wavefront error and Zernike coefficients

Figure 3.33: *"Optimization callback" dialog box*

To activate a custom callback plugin, go to the menu "Mirror ⇒ Optimization callback". The window "Optimization callback" - see Fig.3.33 - will pop up. Press the "Load. . . " button, locate the corresponding DLL and press "OK".

To activate the function "report_iter_result", select the option "Report the result in every iteration" and unselect "Enable custom optimization". To activate "optimization_step", select both "Report the result in every iteration" and "Enable custom optimization". Selecting "Report the wavefront error and Zernike coefficients" will activate the function "report_rms_zernike". If you wish to unload a previously loaded plugin, press the "Release" button.

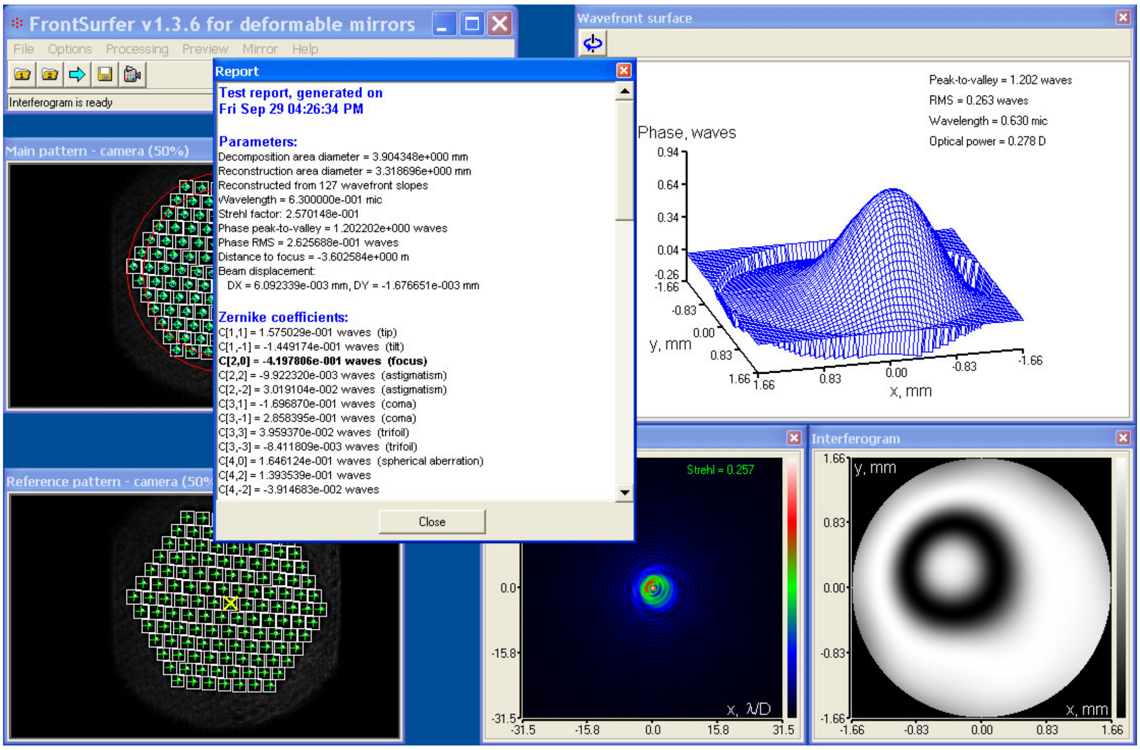
3.8.5 Loading and saving the calibration data

A complete state of the calibrated system can be saved using the menu command "Mirror ⇒ Save calibration". Please specify a catalogue where the files should be saved; preferably, it should be empty. FrontSurfer will generate a number of files in this directory, including the reference pattern, reference grid, matrix of singular modes, wavefront sensor and mirror configuration, aperture etc. The saved system state can be later restored using the command “Mirror ⇒ Load calibration”.

Chapter 4

Software reference

4.1 Program windows



```
Test report, generated on
Fri Sep 29 04:26:34 PM

Parameters:
Decomposition area diameter = 3.904348e+000 mm
Reconstruction area diameter = 3.318696e+000 mm
Reconstructed from 127 wavefront slopes
Wavelength = 6.300000e-001 mic
Strehl factor: 2.570148e-001
Phase peak-to-valley = 1.202202e+000 waves
Phase RMS = 2.625688e-001  waves
Distance to focus = -3.602584e+000 m
Beam displacement:
   DX = 6.092339e-003 mm, DY = -1.676651e-003 mm

Zernike coefficients:
C[1,1] = 1.575029e-001  waves  (tip)
C[1,-1] = -1.449174e-001  waves  (tilt)
C[2,0] = -4.197806e-001 waves  (focus)
C[2,2] = -9.922320e-003 waves  (astigmatism)
C[2,-2] = 3.019104e-002 waves  (astigmatism)
C[3,1] = -1.696870e-001  waves  (coma)
C[3,-1] = 2.858395e-001  waves  (coma)
C[3,3] = 3.959370e-002 waves  (trifoil)
C[3,-3] = -8.411809e-003 waves  (trifoil)
C[4,0] = 1.646124e-001  waves  (spherical aberration)
C[4,2] = 1.393539e-001  waves
C[4,-2] = -3.914683e-002 waves
```

```
Peak-to-valley = 1.202 waves
RMS = 0.263 waves
Wavelength = 0.630 mic
Optical power = 0.278 D
```

Figure 4.1: *Screen-shot of the FrontSurfer program. In the background clockwise starting from the top left corner: main window, wavefront wire-frame reconstruction, wavefront interferogram, far field distribution, reference hartmannogram, sample hartmannogram. In the foreground: the report window.*

The Figure 4.1 shows the output of FrontSurfer wavefront reconstruction:

- **Main window:** always present, contains links to all setup and reconstruction functions. Buttons:(left to right) Sample acquisition (1), reference acquisition (2), “process all” (blue arrow), “save data” (floppy), “preview” (camera).

- **Wavefront wire-frame:** presents information on the P-V and *rms* wavefront deviation from the nearest plane (sphere), measurement wavelength and the strength of the defocus component P (in diopters). The ROC of the wavefront can be obtained as $1/P$.

  The axes tick marks give the lateral scale of the reconstructed wavefront; it is defined by the Hartmann mask size. The top left button corresponds to a pop-up window with slide bars to rotate the wire-frame plot.

- **Interferogram window:** corresponds to the interferogram of the reconstructed phase profile. The interferogram depends on the reconstruction wavelength which can be arbitrary defined through the “Options” ⇒ “Parameters” menu.

- **Far field window:** plots color far field distribution in λ/D scale, where D is the diameter of the reconstruction area.

- **Reference and sample windows:** show the grabbed hartmannogram, superimposed with spot centroiding information, decomposition (outer red circle) and reconstruction (inner red circle) apertures.

- **Report windows:** includes the measurement date, measurement parameters, Zernike and Seidel aberration coefficients and the address of OKO Technologies.

4.2 “File” pull-down menu

4.2.1 Open main

Grabs the image from the camera if “Options ⇒ Parameters ⇒ Camera” is set. Reads the image from a file in **pgm**, **ppm** or **bmp** format if “Options ⇒ Parameters ⇒ File” is set. There is a shortcut button “1” for “Open Main”.

If only main image is opened and “Options ⇒ Parameters ⇒ Get from picture” is unset, the program will generate internal reference grid (in the hexagonal or orthogonal form) and all reconstruction steps will be conducted with respect to this ideal reference.

4.2.2 Open reference

Grabs the image if “Options ⇒ Parameters ⇒ Get from picture” is set. The acquired image will serve as a reference for all further processing steps, including wavefront reconstruction, alignment mode and deformable mirror operation. The image will be acquired from the camera if “Options ⇒ Parameters ⇒ Camera” is set. The image will be acquired from a file in **pgm**, **ppm** or **bmp** format if “Options ⇒ Parameters ⇒ File” is set. There is a shortcut button “2” for “Open Reference”.

4.2.3 Save output

Gives numerous options for saving of the reconstructed data. In particular “Wavefront grid” can be configured to save the wavefront data in three different types of units (wavelengths, micrometers or radians) in two possible formats: gnuplot and Excel Spreadsheet - see Fig. 3.14. “Everything” will save all data in a file having format “file_extension” with extensions:
***_far.bmp** for farfield bmp image (only if the wavelength is defined),

***_far.ps** for farfield postscript image (only if the wavelength is defined),

***_interf.bmp** for interferogram bmp image (only if the wavelength is defined),

***_interf.ps** for interferogram postscript image (only if the wavelength is defined),

***_main.bmp** for sample bmp hartmannogram,

***_main.ps** for sample postscript hartmannogram,

***_ref.bmp** for reference bmp hartmannogram (only in "reference" mode),

***_ref.ps** for reference postscript hartmannogram (only in "reference" mode),

***_wf.ps** for wireframe wavefront in postscript format,

***_wf.txt** for wavefront output as a text table, in gnuplot (one data point per line with empty line as a row separator) or excel format. The parameters of the table can be defined by "File ⇒ Save output ⇒ Wavefront, grid" - see Fig. 3.14.

***_params.txt** for measurement parameters, text file,

***_zernike.txt** for Zernike polynomials, text file,

***_report.txt** for the wavefront reconstruction report.

4.2.4 Exit

Closes the program with saving all options, including configuration of the sensor and deformable mirror.

4.3 “Options” pull-down menu

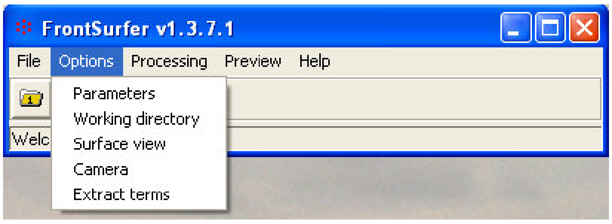
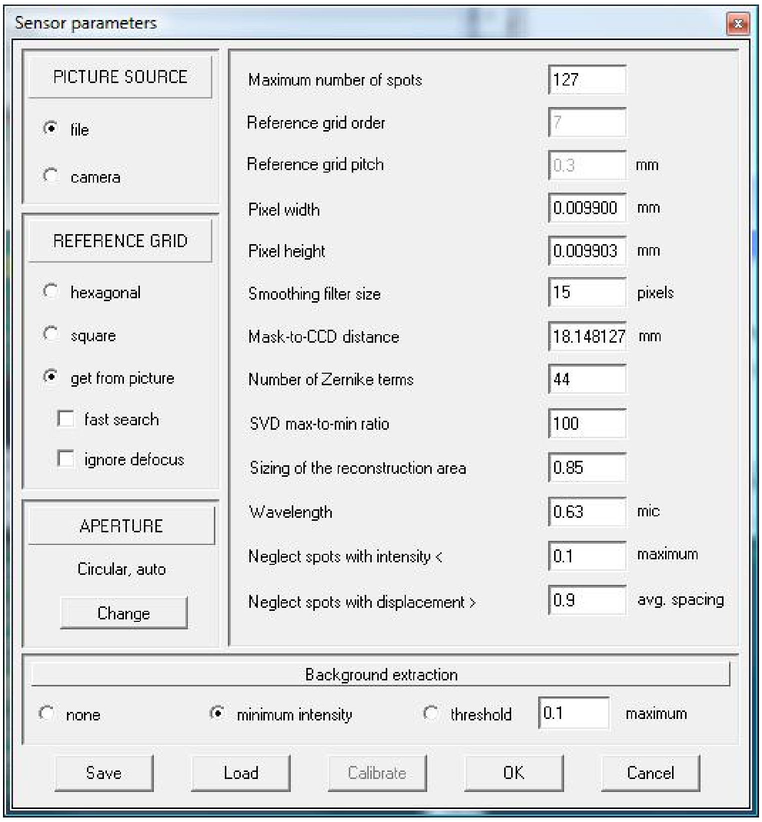


Figure 4.2: *“Options” menu*

4.3.1 Parameters

Calls the dialog box “Sensor parameters” - see Figure 4.3.



Sensor parameters

PICTURE SOURCE
● file
○ camera

REFERENCE GRID
○ hexagonal
○ square
● get from picture
☐ fast search
☐ ignore defocus

APERTURE
Circular, auto

Maximum number of spots 127
Reference grid order 7
Reference grid pitch 0.3 mm
Pixel width 0.009900 mm
Pixel height 0.009903 mm
Smoothing filter size 15 pixels
Mask-to-CCD distance 18.148127 mm
Number of Zernike terms 44
SVD max-to-min ratio 100
Sizing of the reconstruction area 0.85
Wavelength 0.63 mic
Neglect spots with intensity < 0.1 maximum
Neglect spots with displacement > 0.9 avg. spacing

Background extraction
○ none
● minimum intensity
○ threshold 0.1 maximum

Figure 4.3: *“Sensor parameters” dialog box*

- **PICTURE SOURCE:** choose the signal source here. It can be either camera or a file from disk.
- **REFERENCE GRID:** choose the reconstruction mode here. If **“hexagonal”** or **“square”** is checked, the program will design a best-fit ideal hexagonal or square grid to use it as a reference for all reconstructions.

If **"get from picture"** is checked, one needs to acquire a reference image (button "2"). The reference image can have any geometry, but the user has to define the number of spots in the image, since the program itself has no idea about how many spots it should look for.

The "FrontSurfer" program looks for spots positions in every hartmanngram. To find the spots, the program first defines the rectangular regions (white rectangles around spots) for spot localization. After the regions are defined, the program looks for precise position of each spot inside the defined region. Since the full search can take a long time, **"Fast search"** option will re-use the spot regions found in the reference patttern for spot centroiding in the main pattern. This option can significantly improve the speed of processing, but one has to be careful. For instance the spots can move completely out of the pre-defined regions resulting in erroneous results.

"Ignore defocus" option will ignore the defocus term in all reconstruction data. Very useful for measuring aberrations in relation to a spherical reference.

- **APERTURE:** choose whether the aperture (area of interest in the hartmanngram) is determined automatically (**"auto"** option) or defined by the user. If the option **"define manually"** is chosen, the user should manually define the aperture and (optionally) central obscuration. Manually defined aperture can be circular, elliptical, square or rectangular. See Section 3.6.3 for more information on this mode.

- **"Maximum number of spots"** defines the maximum number of spots to be searched in a hartmannogram. This number is user-defined in the reference mode, or calculated automatically from the **"Reference grid order"** in the absolute measurement mode. The **"Reference grid order"** is equal to the number of spots along the side of a square or hexagonal mask (lenslet array).

- **"Reference grid pitch"** should be defined by the user. It should be equivalent to the pitch of the microlens array, or Hartmann mask used.

- **"Pixel width, Pixel height"** is equivalent to the calculated CCD pixel width and height. These parameters are not equivalent to the physical pitch of the CCD sensor used. They take into account the grabber parameters and the effective frame size. For instance these parameters defined for 640x480 mode will be different from the same but defined for 320x240 mode. The best way to determine these parameters is to calibrate the sensor with a point source.

- **"Smoothing filter size."** This parameter controls the behavior of the spot searching algorithm. The higher the parameter, the longer the spot search will take and the lower the sensitivity to high spatial frequencies will be. The parameter has a meaning of smoothing filter window size (in pixels), applied to the image before processing.

- **"Mask-to-CCD distance"** is defined as a result of sensor calibration. Can be manually overwritten in the reference mode.

- **“Number of Zernike terms”** is a manually defined parameter. The higher the parameter, the better the high spatial frequencies will be presented in the wavefront reconstruction.

- **“SVD max-to-min ratio.”** FrontSurfer uses singular value decomposition (SVD) algorithm for orthogonalization of the basis during the wavefront reconstruction [4]. Discarding those of the modes having relatively low singular values allows to make reconstruction more steady, especially if the wavefront is approximated by a large number of Zernike terms. Normally, taking into account modes with singular values larger than 1/100 of the maximum one provides stable reconstruction.

- **“Sizing of the reconstruction area”** defines the relative size of the area of wavefront reconstruction with respect to the automatically or manually defined aperture (area of interest). Its purpose is to reduce the reconstruction errors at the periphery. The possible range is 0...1. If this parameter is set to “1”, the reconstruction area is identical to the aperture.

- **“Wavelength”** defines the reconstruction wavelength for the interferogram and the far field window. It also influences the calculated Strehl parameter. As the measurement principle is wavelength independent, this parameter can be set to any value. If the wavelength is not set, processing will produce only surface plot, omitting the wavelength-dependent interferogram and far field.

- **“Neglect spots with intensity < … maximum”** defines the threshold level for the spots intensity. The spots weaker than the threshold will not be taken into account.

- **“Neglect spots with displacement > … avg. spacing”** defines the maximum displacement allowed for the spots with respect to the average spacing between spots. If the displacement exceeds the allowed value, this spot will not be used for wavefront reconstruction.

- **“Background extraction”** specifies how an imported image should be preprocessed in order to extract the background. Strong background may introduce a significant error in the found positions of the spots centroids. If “none” is selected, no preprocessing is performed; that is why it is important to find a proper way to extract it. If “minimum intensity” is selected then the minimum pixel value over the whole image is extracted from all pixel values. If “threshold” option is selected, then all pixels with intensity below a certain threshold, which is specified with respect to maximum intensity over the whole image, are assigned a zero value.

4.3.2 Working directory

This menu defines the default directory for loading and saving files.

4.3.3 Surface view

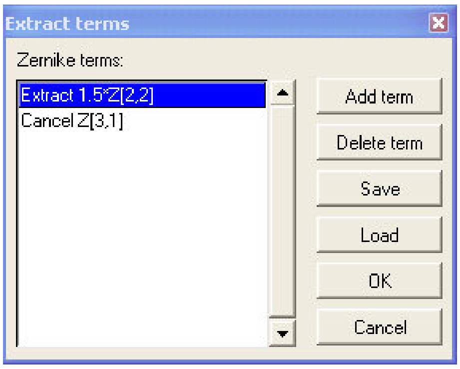
This menu defines the perspective view of the wireframe wavefront graph. Can be also addressed by pushing the button in the top left corner of the wireframe graph window.

4.3.4 Camera

Invokes the menu for setup and configuration of the video interface. For more information please see section 3.2, page 24.

4.3.5 Extract terms

Invokes the dialog box "Extract terms" - see Figure 4.4.



Extract 1.5*Z[2,2]
Cancel Z[3,1]

Figure 4.4: *"Extract terms" dialog box*

FrontSurfer allows to extract certain Zernike aberration terms from the reconstructed wavefront or cancel them by zeroing the corresponding Zernike coefficients. Using this dialog box one may add, delete and modify certain Zernike terms by using the buttons and double-clicking on the items in the list. It is also possible to save the whole list in a file and load a previously saved list of aberrations. It is also possible to load the list of Zernike coefficients saved from wavefront measurement results. It can be done using the menu command "File ⇒ Save output" and choosing the option "Zernike coefficients, mic".

The Zernike terms specified here are also extracted from the measurement results in the alignment mode and taken into account in the feedback loop mode.

4.4 "Processing" pull-down menu

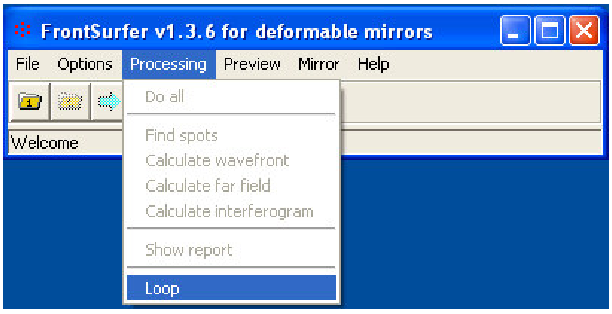


Figure 4.5: *"Processing" menu*

4.4.1 Do all

This is the most useful command in FrontSurfer. Because of its importance, it is linked to the large blue arrow in the button bar. Pushing this button will result in all results to be shown at once.

4.4.2 Find spots

4.4.3 Calculate wavefront

4.4.4 Calculate far field

4.4.5 Calculate interferogram

These commands allow to do step-wize processing of the hartmanogram. Normally they can be replaced by the single command "Do all", which executes all four commands sequentially. See Section 3.4 for more explanation.

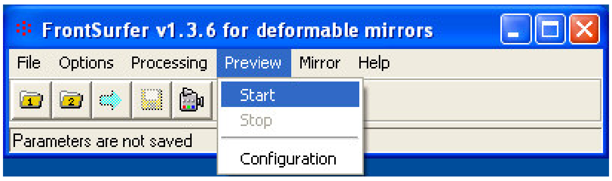
4.4.6 Show report

Displays the final report containing the summary results of reconstruction and wavefront analysis. For more information please see page 39.

4.4.7 Loop

Starts the alignment mode. For more information please see Section 3.7.

4.5 “Preview” pull-down menu



Parameters are not saved

Figure 4.6: *“Preview” menu*

4.5.1 Start

Starts the preview mode. When activated, the preview mode will be switched off for image grabbing (since two processes can not share one video input. For the same reason preview is de-activated in the alignment mode and when the program is operated with a deformable mirror.

The “Start” entry is linked to the “Preview” button in the button bar.

4.5.2 Stop

Stops the preview mode. When the preview mode is on, this entry is also linked to the “Preview” button in the button bar.

4.5.3 Configuration

Calls the dialog box “Camera preview configuration” to configure the preview mode (Fig.3.3).

FrontSurfer uses external viewer programs for standard video interfaces. “VidCon32” is mostly used for “Video for Windows” interface; it is included into the distribution. “xawtv” is used for standard video interface in Linux; this open-source utility is included in many Linux distributions and is available in Internet.

Preview for the plugin-supported video interfaces is usually implemented within the corresponding plugin; you need to turn on the option “internal viewer” when using these interfaces.

4.6 “Mirror” pull-down menu

This menu is available only in the version for deformable mirrors.

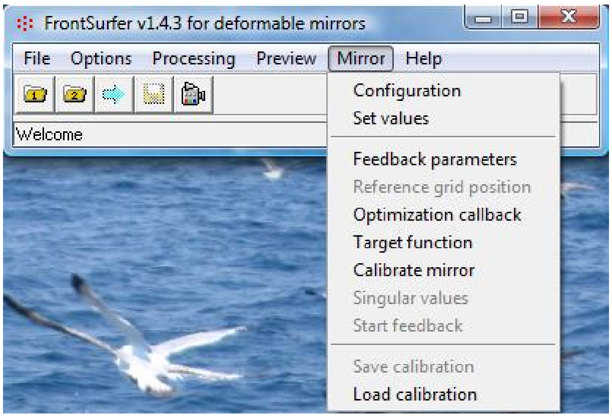


Figure 4.7: *“Mirror” menu*

4.6.1 Configuration

Calls the dialog box “Mirror interface” (Figure 3.25) to configure interface for deformable mirror. See Section 3.8.1 for more information.

4.6.2 Set values

Calls the dialog box “Set mirror values” (Figure 3.27) for manual control of the actuator values. See Section 3.8.2 for more information.

4.6.3 Feedback parameters

Calls the dialog box “Feedback parameters” (Figure 3.28) to adjust parameters of the feedback-loop operation mode. See page 55 for more information.

4.6.4 Reference grid position

Invokes the dialog box “Reference grid position” (Figure 3.31), which allows to define the position of the computed reference grid in the closed-loop mode. See page 57 for more information.

4.6.5 Optimization callback

Invokes the dialog box “Optimization callback” (Figure 3.33), which allows to specify a custom callback plugin for external monitoring of the closed-loop control and custom optimization. See the section 3.8.4 for more information.

4.6.6 Target function

Calls the dialog box “Target function” (Figure 3.30) to define a target aberration to be generated by the deformable mirror in the feedback-loop mode. See page 57 for more information.

4.6.7 Calibrate mirror

Starts calibration of the deformable mirror; it should be done once before starting feedback-loop correction. See Section 3.8.3 for more information.

4.6.8 Singular values

Calls the dialog box “Singular values” (Figure 3.29) to display singular values and SVD modes of the adaptive optical system. See page 56 for more information.

4.6.9 Start feedback

Starts the feedback-loop wavefront correction mode. See Section 3.8.3 for more information on this mode.

Bibliography

[1] J. Geary, Introduction to wavefront sensors, SPIE, 1995.

[2] D. L. Fried, Least-square fitting a wave-front distortion estimate to an array of phase–difference measurements, Journal of the Optical Society of America **67**, 370–375 (1977).

[3] R. H. Hudgin, Wave-front reconstruction for compensated imaging, Journal of the Optical Society of America **67**, 375–378 (1977).

[4] L. Baker, C tools for scientists and engineers, New York, McGraw-Hill, 1989.

[5] O. Soloviev, G. Vdovin, Hartmann-Shack test with random masks for modal wavefront reconstruction, Optics Express **13**, 9570–9584 (2005).

[6] C. Paterson, I. Munro, C. Dainty, A low cost adaptive optics system using a membrane mirror, Optics Express **6**, 175-185 (2000).

Index

Aberrations of relay optics, 14, 15
Absolute measurement mode, 28
Adaptive optics, 50
Adaptive optics loop frequency, 50
Alignment mode, 48, 69
Aperture, 66
Aperture, manual definition, 30, 45, 66
Area of interest, 30, 45, 66
Atmospheric turbulence, averaging, 56

B/W camera, 7
Background extraction, 67
Beam scaling, 15

Calculate far field, 69
Calculate interferogram, 69
Calculate wavefront, 69
Calibration with a point source, 32
Calibration, parameters, 34
Callback plugin interface, 58
Camera, 68
CCD to mask distance, 14

DAC40ETH, 51
DAC40USB, 51
Decomposition area, 40
Deformable mirror, 50
Deformable mirror, “Rotate”, 53
Deformable mirror, calibration, 72
Deformable mirror, configuration, 51, 71
Deformable mirror, custom interface, 53
Deformable mirror, Ethernet interface, 52
Deformable mirror, feedback-loop mode, 54
Deformable mirror, hysteresis, 55
Deformable mirror, interfaces, 51
Deformable mirror, ISA interface, 51
Deformable mirror, manual control, 53, 71
Deformable mirror, PCI interface, 51
Deformable mirror, plugin, 53
Deformable mirror, SVD modes, 56
Deformable mirror, USB interface, 52
Delay, 55
Do all, 69
Dynamic mode, 48

Errors of measurement, 13
Errors, image sensor geometry, 13, 14
Errors, image sensor noise, 14, 17
Errors, mask geometry, 13, 14
Errors, reconstruction algorithm, 14, 16
Errors, sampling, 14, 15
Ethernet, 51
Examples, 37
Examples, absolute measurement mode, 37
Examples, calibration, 32
Examples, manually defined aperture, 45
Examples, reference mode, 42, 45
Exit, 64
Extract terms, 68

Far field, 62
Fast search, 29, 66
Feedback loop operation mode, 54
Feedback parameter, 55
Feedback parameters, 54, 71
Fill factor, 15
Filter size, 66
Find spots, 69
Frame grabber, 24
Free edge mirrors, 55
FrontSurfer, data processing, 30
FrontSurfer, features, 8
FrontSurfer, installation, 23
FrontSurfer, introduction, 7

FrontSurfer, output files, 40
FrontSurfer, version for deformable mirrors, 9, 50

Gamma parameter, 17
Gnuplot format, 64
Graphical formats, 40
Grid order, 66

Hartmann mask, 7
Hartmann test, principle, 11
Hartmann-Shack test,principle, 11
Hexagonal grid, 28
How to calibrate, 34

Ignore defocus mode, 34
Ignore defocus option, 66
Image sensor, geometry, 14
Image sensor, noise, 14
Initial voltage, 55
Intensity threshold, 67
Interferogram, 16, 62

Loop, 69

Main grid, 30
Main window, 61
Mask geometry, 14
Mask size, 15
Mask to CCD distance, 66
Maximum spot displacement, 67
Measurement errors, 13
Measurement head, 7
Measurement modes, 27
Measurement modes, absolute, 28
Measurement modes, reference, 29
Microlens array, 7, 14
Mirror bias, 53
Mirror calibration data, loading, 59
Mirror calibration data, saving, 59
Mirror menu, 71
Mirror, calibration, 56
Modal reconstruction, 12, 31
Multiprocessor support, 56

Number of terms, reconstruction, 17
Number of Zernike terms, 66

Open main, 63
Open reference, 63
Optical shop testing, microlenses, 20
Optical shop testing, mirrors, 19
Optical shop testing, schemes, 18
Optical shop testing, transparent optics, 18
Optimization callback, 58, 72
Options, 65
Orthogonal grid, 28
Orthogonalization, 13, 31

Parameters of calibration, 34
Pixel width, height, 66
Preview, 70
Preview, configuration, 70
Preview, start, 70
Preview, stop, 70
Program windows, 61
Protection dongle, 23

Realtime priority, 56
Reconstruction area, 40
Reference and sample, 62
Reference grid, 30
Reference grid order, 66
Reference grid position, 57, 71
Reference mode, 29
Report file, 39
Report window, 62
Restart feedback, 56

Sample and reference, 62
Save output, 63
Save, parameters, 63
Saving the results, 40
Screen-shot, 61
Seidel aberrations, 40
Sensitivity, 7
Sensor parameters, 65
Show report, 69
Singular value decomposition, 13, 31, 54, 67
Singular values, 56, 72
Sizing of reconstruction area, 67
Smoothing filter, 30, 66

Spectral range, 7
Square grid, 28
Start feedback, 72
Surface view, 68
SVD, 13, 31
SVD max-to-min ratio, 67

Target function, 56, 72
Tip and tilt correction, 55

USB, 51

Video for Windows, 24
Video interface setup, 24
Video plugin, 25, 26
Video plugin, development, 26
Viewer, 24
Voltage step, 55

Wavefront correction, 54
Wavefront error calculation, 56
Wavefront file save, parameters, 64
Wavefront reconstruction, 12
Wavefront sensing, 11
Wavefront sensor calibration, 32
Wavefront tilts, 31
Wavefront wire-frame, 62
Wavelength, 67
Working deirectory, 67

Zernike polynomials, 13, 31, 40
Zonal reconstruction, 12