



# **A MATLAB Program for Beam Spot Evaluation at the PG2 Beamline Diagnostics Port at FLASH**

## **Summer Student Project**

Steffen Richter, Universität Leipzig, Germany

July 19 - September 8, 2011

### **Abstract**

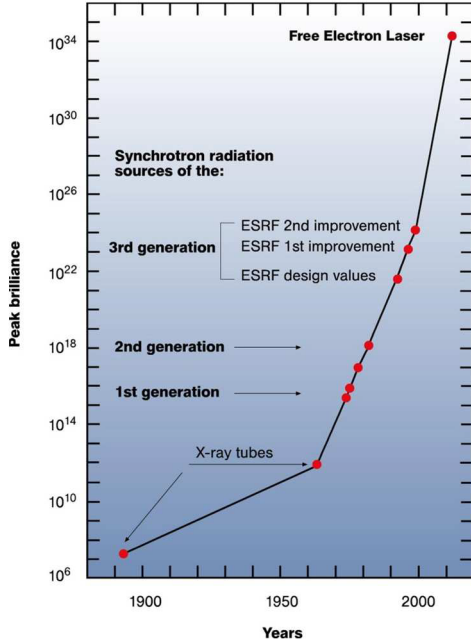
One of the most challenging procedures when working with a free-electron laser is the fluctuation of the beam pulse. Knowledge about size and position of the focused beam is not only important for the lateral resolution but also to estimate the peak brilliance. Furthermore, knowledge about the average beam size is necessary to perform pump-and-probe experiments where two laser beams must match exactly.

In this project, images of a beam spot on a fluorescent Ce:YAG crystal were evaluated. For this purpose, a MATLAB program was written that allows to evaluate spot widths and position fluctuations from such beam spot images. The program can be used to find out about the position of focal planes and also helps to determine the lateral precision and stability of the beam spot.

# Contents

<b>1. Introduction</b>	<b>3</b>
<b>2. Theory of FEL Facilities</b>	<b>3</b>
2.1. Classical Synchrotron Radiation - Bending Magnets, Wigglers, Undulators	4
2.2. FEL Principles . . . . .	6
2.2.1. SASE Effect . . . . .	6
2.2.2. Seeding . . . . .	8
2.3. X-Ray Optics . . . . .	9
<b>3. Setup at FLASH</b>	<b>9</b>
3.1. Overview about FLASH . . . . .	10
3.2. Beam Diagnostics at PG2 . . . . .	11
3.2.1. Methods of Beam Diagnostics . . . . .	12
<b>4. The MATLAB Program: beamspotdiagnostics</b>	<b>12</b>
<b>5. Data Evaluation</b>	<b>15</b>
5.1. Data from May 2011 . . . . .	16
5.2. Data from August 2011 . . . . .	20
5.3. Conclusion . . . . .	23
<b>APPENDIX</b>	<b>24</b>
<b>A. Source Code</b>	<b>24</b>
A.1. Extract of GUI_beamspotdiagnostics.m . . . . .	24
A.2. Extract of spotevaluation1.m . . . . .	32
<b>B. References</b>	<b>34</b>

# 1. Introduction



**Figure 1:** Historic development of peak brilliances [3]. Brilliance is the number of photons per time, area and solid angle within 0.1% of the spectral bandwidth.

absolute importance to optimize the FEL beam properties in terms of focusing and point stability. Since every FEL pulse is different from another one (what will be explained in chapter 2.2.1), diagnostics and evaluation of the beam properties need somewhat more effort. To find out about spotsizes and positions, a plenty of single shot images need to be evaluated. For this purpose, a program with graphical user interface has been written during this project.

## 2. Theory of FEL Facilities

Free-electron lasers (FEL) are the latest generation synchrotron radiation sources, i.e. the fourth generation. In contrast to all the generations before, they use a self amplification effect caused by the interaction between the electron and photon beams. To achieve this, long undulators are necessary and linear accelerators become more favorable. In the following, the production and properties of synchrotron radiation and the different generations of sources shall be explained. Those principles are illustrated in figure 2.

Since 2005, FLASH - the **F**ree-electron **L**ASer in **H**amburg has been working as a unique facility providing a laser beam in the EUV and soft x-ray regime. As figure 1 shows, it's peak brilliance exceeds the one of previous synchrotron radiation sources by far. Thus it is possible to gain complete diffraction patterns from only one shot. Furthermore, the length of the x-ray pulses produced at FLASH is around 100fs. This offers new possibilities to understand how chemical bondings and reactions work by time-resolved measurements.

As most of the samples are getting destroyed while using such intense pulses, data recording is a sophisticated task. Pump-and-probe experiments use two laser beams whereof one is probing while the other one is used for recording information like diffraction patterns. In a proof-of-principle experiment, it has been shown that it is even possible to record a diffraction pattern before the sample is destroyed by the same beam pulse [2].

To be able to use those brilliant x-ray pulses, it is of absolute importance to optimize the FEL beam properties in terms of focusing and point stability. Since every FEL pulse is different from another one (what will be explained in chapter 2.2.1), diagnostics and evaluation of the beam properties need somewhat more effort. To find out about spotsizes and positions, a plenty of single shot images need to be evaluated. For this purpose, a program with graphical user interface has been written during this project.

## 2.1. Classical Synchrotron Radiation - Bending Magnets, Wigglers, Undulators

Whenever charged particles are accelerated they emit radiation. The total power emitted follows the *Larmor formula*. Its Lorentz-invariant representation is ( $\vec{p}$  and  $t$  in the lab frame; CGS system) [3]:

$$P = \frac{2q^2}{3m_0^2c^3}\gamma^2 \left[ \left( \frac{d\vec{p}}{dt} \right)^2 - \frac{1}{c^2} \left( \frac{dE}{dt} \right)^2 \right] \quad (1)$$

If the particles' energy  $E$  is kept constant and they are accelerated constantly in a way that they move on a circle with radius  $R$ , classical synchrotron radiation (SR) is produced. For the total emitted power, it follows [3]:

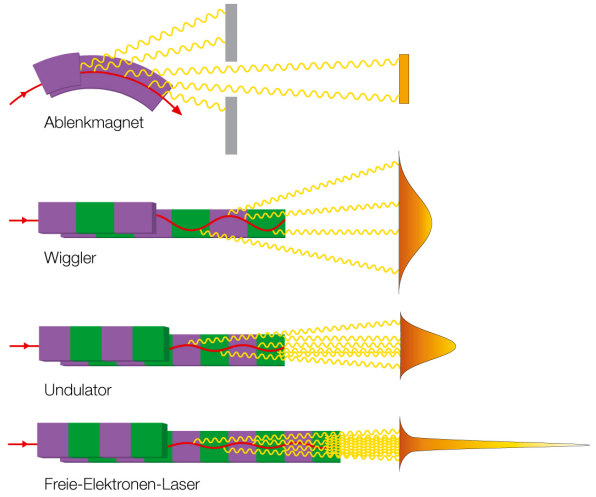
$$P_{circular} = \frac{2q^2}{3m_0^4c^7} \frac{E^4}{R^2} \quad (2)$$

Caused by Lorentz transformation from the particle to the lab system, the radiation power emitted in a narrow cone of angle (with Lorentz factor  $\gamma = \frac{E}{m_0c^2}$ ):

$$\theta_{bending\ magnet} \approx \frac{1}{\gamma} \quad (3)$$

The first generation of SR sources used this through ordinary bending magnets, as they are needed to control the path of charged particles in accelerator or storage rings like synchrotrons. Since the radiated power scales with  $m_0^{-4}$ , bunches of electrons or positrons are used.

For bending magnet radiation, the photon spectrum is determined by the critical energy  $\epsilon_c = \frac{3\hbar e B}{2m_e} \gamma^2$ . This is the photon energy where half of the radiation power is emitted by photons below and half of it by photons above [4]. The spectral photon density can be expressed by integrals involving modified Bessel functions. However, using Heisenberg's uncertainty principle, the photon energy spread can be approximated from the emission pulse duration of one single particle emitting bending magnet radiation when traveling by. This approach yields  $\Delta E \geq \frac{2\hbar e B \gamma^2}{m_e}$  [4]. In fact, bending magnet radiation reveals a broad energy spectrum with a maximum slightly below  $\epsilon_c$  (cf. figure 3, right).



**Figure 2:** Scheme of the different generations of SR sources [1].

To produce higher flux SR it is useful to bend the particle beam several times. This is done with a row of magnets, forming a so called wiggler - the second generation of SR sources. A wiggler produces bending magnet radiation as well, but the outcoming flux scales with the number of magnet poles  $N$ . Thus its total radiated power is much higher while the brilliance does not increase since the radiation cone has a significantly greater angle.

To improve the brilliance, the magnetic periods can be made shorter, resulting in an undulator (3rd generation SR source). Here, the radiation cone angle becomes much smaller than  $\frac{1}{\theta}$  because the electron path is less curved. However, constructive and destructive interference due to the periodic undulator length  $\lambda_u$  become now more important. This yields the fundamental wavelength of the outcoming photons referred to the so called *undulator equation*:

$$\lambda = \frac{\lambda_u}{2\gamma^2} \left( 1 + \frac{K^2}{2} + \gamma^2 \theta^2 \right) \quad (4)$$

where  $K$  is the *undulator constant*:

$$K = \frac{\lambda_u e B}{2\pi m_e c} \quad (5)$$

$K$  is a dimensionless representation of the magnetic strength for a periodic magnet. If  $K \leq 1$ , the source is called undulator, otherwise wiggler. Figure 3 shows the transition between wiggler and undulator.

The fundamental wavelength of an undulator is emitted within a central cone that is smaller than that of a bending magnet:

$$\theta_{central} \approx \frac{1}{\gamma \sqrt{N}} \quad (6)$$

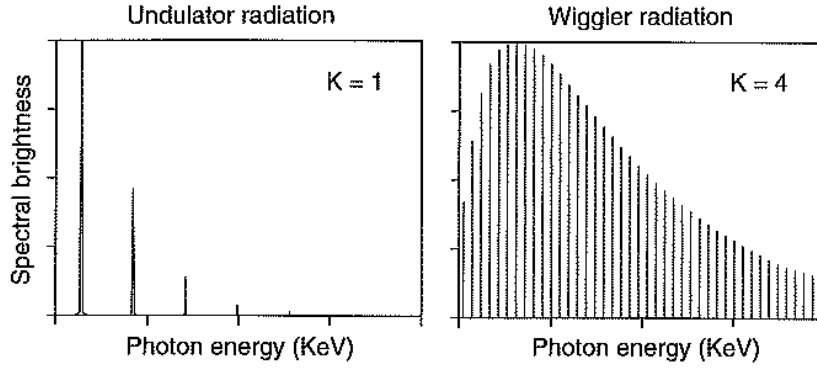
For this reason, the flux (emitted power per area) scales not only with  $N$  (as a wiggler does) but with  $N^2$ .

The natural bandwidth of the fundamental wavelength is  $\frac{\Delta\lambda}{\lambda} = \frac{1}{N}$ . The power emitted by one electron within this central cone is given by [4]:

$$P_{central} = \frac{\pi e^2 c \gamma^2}{\epsilon_0 \lambda_u^2 N} \frac{K^2}{(1 + \frac{K^2}{2})^2} \quad (7)$$

In fact, an undulator spectrum contains not only the fundamental wavelength but also higher harmonics  $\lambda_n = \frac{\lambda}{n}$ . Those result from different projection directions of the electron path fluctuations such as in the beam direction. Consequently, their spatial properties are different. For even harmonics, the intensity becomes 0 for  $\theta \approx 0$ . Hence they can only be seen off-axis and not on-axis (cf. figure 3). However, considering electron bunches (as used in storage rings), there is always a certain divergence of the electron paths. This leads up to the fact that also even harmonics can occur.

Photons (no matter what harmonic number) beyond the central radiation cone cause a



**Figure 3:** Wiggler and undulator spectra in comparison [4]. Due to longer periodic magnet length, the wiggler spectrum contains more harmonics than the undulator one. At the same time, the bending magnet spectrum with its critical energy can clearly be recognized. It is important to note that both spectra only contain odd harmonics. This is idealized for a single electron (positron), only considering on-axis radiation.

smearing of the peaks due to off-axis Doppler effects [4]. Therefore, the spectrum would show very strongly broadened peaks.

Another important property of SR is its polarization. Since the polarization directly follows the electron motion, SR is polarized totally horizontally within the plane of the electron path. Above and below, off-axis Doppler effects influence the polarization, yielding elliptically polarized light. This applies only to bending magnet and fundamental undulator radiation. For higher undulator harmonics, the polarization is different, depending on the underlying electron motion.

## 2.2. FEL Principles

The key to even more intensive SR is stimulated emission as it is known from lasers (*light amplification by stimulated emission of radiation*). However, a 'closed FEL' with mirrors before and behind an undulator making it a resonator is not practical for x-rays [7] (cf. section 2.3). Instead, stimulated emission has to occur during a single pass of an electron through the undulator. The effect is known as *self-amplification*.

### 2.2.1. SASE Effect

The usual working mode of an FEL is the so called *Self-Amplification of Spontaneous Emission*, SASE. To understand the SASE process, it is necessary to consider the interaction between an already existing light wave and an electron moving through the undulator:

The energy transfer from an electro-magnetic wave to an electron is simply given by  $\Delta W = -e \int \vec{E}_{\text{photon}} d\vec{s} = -e \int \vec{v}_{\text{electron}} \vec{E}_{\text{photon}} dt$  [6]. The electric field of the undulator radiation is polarized in horizontal direction. Whenever the transverse electron movement

and the electric field vector of the light show in the same direction, the electron is decelerated while the light is amplified. This works vice versa if electric field vector and electron movement are anti-parallel. Thus, amplification of the undulator radiation is possible, if the phase of the light wave matches exactly with the transverse electron movement. I.e., the maximum transverse electron speed is overlapped with maxima of the electric field of the photon wave in the same direction. Otherwise the effect cancels out.

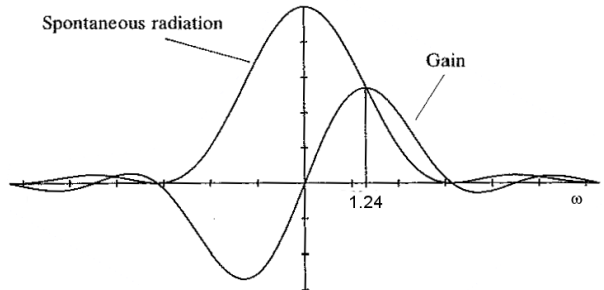
Since light travels faster than the electrons, a phase slippage appears from one to the next undulator pole. However, if this phase slippage between electron and photon wave is constant  $\psi = \pi$ , the condition for amplification is fulfilled. Otherwise the resonance condition for undulator radiation (eq. 4) is still valid.

For gain calculation, the exact electron movement that results must be calculated. From phase relations between electron and photon wave, a dependence of the gain on the Lorentz factor or photon wavelength, respectively, can be derived. In case of low gain, the laser field is assumed as constant. With this assumption, the dependence simplifies: One can show for the intensity behavior of an undulator radiation peak that the Intensity behaves as  $I(\Delta\gamma) \propto \left(\frac{\sin \omega}{\omega}\right)^2$  with  $\omega \propto \frac{\gamma - \gamma_r}{\gamma_r}$  where  $\gamma_r$  is the resonant Lorentz factor for the undulator (eq. 4). *Madey's theorem* now tells that the gain curve for a low-gain FEL follows the intensity derivation [6]:

$$g(\omega) \propto -\frac{dI}{d\omega} \propto -\frac{d}{d\omega} \left( \frac{\sin \omega}{\omega} \right)^2 \quad (8)$$

Figure 4 illustrates this. The maximum gain appears at  $\omega \approx 1.24$  [8]. This is, the photon wavelength is smaller than the fundamental undulator wavelength that starts the process as spontaneous radiation at a given electron energy.

The appearance of electron bunches improves the lasing in an FEL undulator through small energy differences. Thus, the fundamental wavelength and that one of maximum gain are spread wider than the bandwidth according to equation 8. This broadening allows a better overlap of undulator emission and FEL amplification but contains more free parameters on the other site. This means depending on the exact electron energies and positions given within a bunch, FEL pulses arise randomly with slightly different wavelengths. Moreover, the pulse intensities can fluctuate dramatically. Therefore one can say, an FEL pulse is growing from noise.



**Figure 4:** Overlap plot of the fundamental undulator radiation peak and FEL low gain as functions of  $\omega \propto \frac{\gamma - \gamma_r}{\gamma_r}$ . Modified from [9].

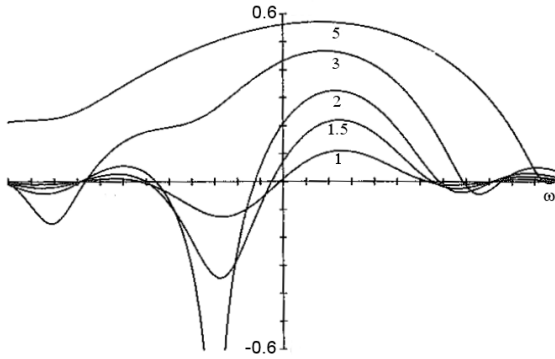
Another effect during the energy transfer from the electrons to the laser wave is so called micro-bunching: According to figure 4, electrons faster than the resonant energy are decelerated while slower electrons get accelerated. This causes a bunching sub-structure in longitudinal direction, yielding an even higher electron density at energies

corresponding to each lasing wavelength. In fact, this high electron number  $N_e$  'per wavelength' leads to a radiation power dependence  $P \propto N_e^2$  [8, 11].

At full micro-modulation the gain saturates.

The absolute magnitude of the FEL gain is depending on the electron and photon density and thus on the undulator length  $L$ . The maximum gain scales with  $L^3$  [9]. This leads to a different working regime where the laser field can not longer be assumed as constant. Calculation of the gain functions gets more complicated and is often only possible numerically [6]. Resulting high gain functions are shown in figure 5. For high FEL gain, there is also better amplification of the resonant wavelength, spreading the difference between individual FEL pulses even more.

To get use of high gain FELs, their undulators are usually much longer than those used as 3rd generation SR sources. Here is also the reason, FELs mostly use linear accelerators.



**Figure 5:** Transition from low to high gain function. The numbers next to the 5 curves represent the length of the undulator, where 1 is the undulator length corresponding to the low gain in figure 4. Modified from [9].

brilliance of a FEL, those diffraction patterns can be obtained even from single complex molecules without need to grow e.g. protein crystals.

The length of a laser shot is determined by the bunches to the order of 100fs. This is the limit to observe dynamics in chemical reactions and atomic bonds.

An FEL spectrum looks in principle like an undulator spectrum. As  $\theta \approx 0$  (exact overlap of laser and electron beam), there appear basically only odd harmonics (cf. section 2.1). As mentioned above, the exact wavelength and intensity differs from pulse to pulse. But also beam position and time structure (pulse length and intensity behavior) fluctuate. Despite these problems, FEL pulses show up very high peak and average power. Due to the micro-bunching, an FEL shot shows up full transverse coherence. This opens the opportunity to obtain a whole diffraction pattern with one shot. Thanks to the high

### 2.2.2. Seeding

To start a self-amplification process, a sufficiently intensive light field is required to overlap exactly with the electron beam. For a SASE effect, first substantial coherent radiation occurs spontaneously at one random site within the undulator. This can happen sooner or later resulting in different pulse intensities and slightly different positions. Further, the amplified wavelength is not exactly the same for different pulses.

To avoid the SASE fluctuations and improve shot-to-shot repeatability, one approach is to *seed* the self-amplification. The principle behind it is to pre-modulate the electron beam. This can be done e.g. using a high intense optical laser. A further advantage in



that case would be that the FEL is perfectly synchronized with an external optical laser for pump-and-probe experiments [12].

However, it is a big effort to align the seeding laser sufficiently accurate to create conditions for self-amplification. The phase of the seeding light must match exactly with the undulator structure and its resulting electron bunch movement.

### 2.3. X-Ray Optics

Caused by the short wavelength and refraction indices  $n \approx 1$ , optics for x-rays must use different principles than usual optics. To reflect x-rays, one can use either total external reflection or Bragg diffraction. The first one is a result of shallow incidence angles on materials where  $n < n_{\text{vacuum}} = 1$ , as it appears for x-rays. The latter principle is used with single crystals or multilayer mirrors. In the multilayer case, radiation is reflected by alternating films of different refraction indices with thicknesses matching to the used photon wavelength.

Using these principles, one can build mirroring and focusing devices. They again reveal opportunities for monochromatization of white beams.

The simplest way to focus a beam is to use toroidally shaped mirror. However, this can lead to a so called astigmatism, meaning the focal plane in horizontal and vertical direction are different. To solve this problem, so called Kirkpatrick-Baez optics uses a set of two adjustable mirrors where each of them independently focuses the beam in one direction.

Another way of focusing uses a hastate tube. Pointing to a focus spot, x-rays are reflected under shallow angles on its inner surface.

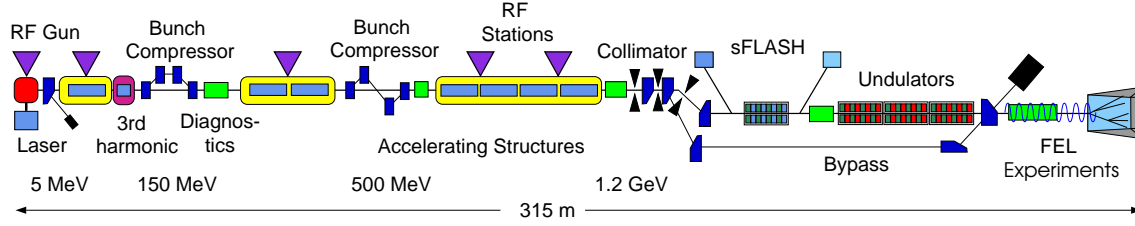
A different approach are Fresnel zone plates which use interference for focusing. They can be applied in reflection or transmission.

Refractive lenses are difficult to realize because the refractive indices for x-rays are too near at 1 for nearly all materials. Thus, rows of many concave lenses are needed.

It is important to mention that SR optics devices generally need to be cooled. This is due to the heat load from penetration of the intense radiation into the material. This applies especially for slits which are introduced to cut the beam size in order to get a well defined beam shape. Materials used for slits should also have high absorption (e.g. tungsten).

## 3. Setup at FLASH

FLASH is an around 300m long FEL in Hamburg. It produces pulses in the EUV and soft x-ray regime. Pulse lengths down to 10fs have been achieved [1]. FLASH was upgraded several times and the construction of an extension called FLASH II is in progress.



**Figure 6:** Longitudinal structure of FLASH [1].

### 3.1. Overview about FLASH

FLASH consists mainly of an electron source, linear accelerator and a long undulator. Figure 6 gives an overview about the flash setup.

The electron source consists of a  $\text{Cs}_2\text{Te}$  photocathode where electron bunches are injected by an UV laser. The electrons are then accelerated by radio-frequency fields. This is why the electron source is called RF gun. The charge of one bunch is 0.5 and 1 nC [1].

A linear accelerator follows directly. Six modules of superconducting Nb-cavities raise the electron energies up to 1.25 GeV. This is done by high-voltage radio-frequencies. The accelerator modules are interrupted by so called bunch compressors: During the acceleration in the cavities, the bunch velocity is determined by the radio frequency but the energy gain is different for electrons at different longitudinal positions within the bunch. This makes, leading electrons in a bunch possess smaller energies than trailing ones. Bunch compressors now are magnetic chicanes where the electrons are guided to travel a short distance with transverse offset. Thus the leading, smaller energy electrons are deflected stronger, travel a longer way and the bunch is compressed [1]. High electron densities and short pulses become possible.

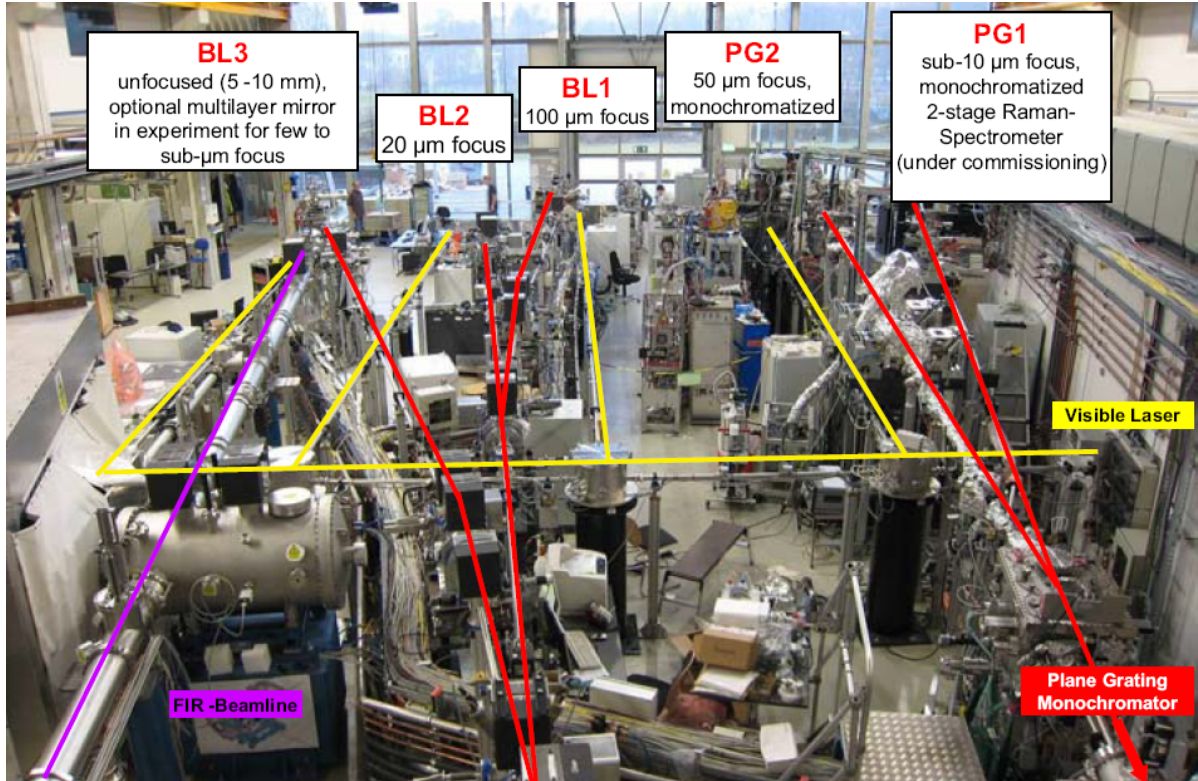
Finally, the FEL beam is produced in a 27m long undulator line. Afterwards, the electron beam is guided to a dump.

Typical running parameters are listed in table 1.

**Table 1:** Typical operation parameters during the 2nd user period from Nov 26, 2007 to Aug 16, 2009 [12].

parameter	value
wavelength (fundamental)	6.8 - 47 nm
average single pulse energy	10 - 100 $\mu\text{J}$
pulse duration (FWHM)	10 - 70 fs
peak power (from average)	1 - 5 GW
average power for 500 pulses/sec	$\approx 15$ mW
spectral bandwidth (FWHM)	$\approx 1$ %
peak brilliance	$10^{29} - 10^{30} \frac{\text{photons}}{\text{s mmrad}^2 \text{mm}^2 0.1\% \text{bw}}$

The undulators of FLASH have a fixed gap of 12mm. A change of the photon energy is



**Figure 7:** Overview of the FLASH experimental hall [13]. Additionally to the FEL beam, an optical laser or a far IR beam is available at some beamlines.

done by changing the electron bunch energy through the accelerator. In fact, the extension FLASH II will contain undulators with changeable gap which is a more convenient way to change the x-ray energy.

The FEL beam can be guided to different beamlines. In principle, there are 5 of them differing in focusing, monochromatization and combination opportunities with laser or IR, respectively. Figure 7 illustrates those. A more detailed description of the user facility is given in [14].

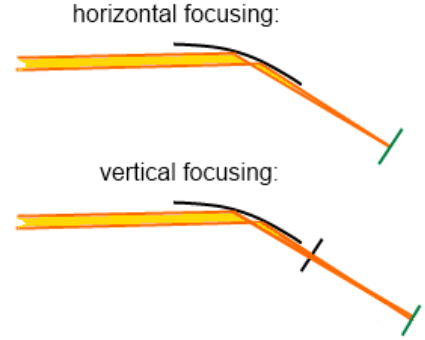
### 3.2. Beam Diagnostics at PG2

PG2 is a monochromator beamline based on a reflection grating. Focusing is done with only one mirror. In contrast to Kirkpatrick-Baez optics this can yield a so called astigmatism, i.e., the focal plane in horizontal and vertical direction differ from each other. To find out about this, a special diagnostics port can be mounted. Doing so, cross-section diagnostics are performed.

Additionally, apertures and slits can be introduced. The horizontal spotshape corresponds to focusing by the mirror in this direction. Thus it is assumed to be presented by a gaussian intensity curve. The vertical shape is expected to arise from a slit cutting the beam on its top and bottom. At the current setup, this slit is projected 1-to-1, i.e.

the spotsize should be equal to the length of the slit. This setup principles are shown in figure 8.

Further information about the PG2 beamline



### 3.2.1. Methods of Beam Diagnostics

Beam spot diagnostics can investigate spectral content, intensities, time structure or lateral information such as beam spot positions and cross-sections. Spectral and intensity diagnostics is done e.g. online using a gas monitor detector. This is important to get exact information for each pulse independently. In this project, cross-section and lateral position diagnostics is done directly, i.e. invasive.

**Figure 8:** Sketch of the focusing principles at the PG2 setup. Green represents the Ce:YAG crystal, black the mirror and exit slit.

For continuous beams, scans over slits, wires or grids are common for this purpose. However, for FEL pulses this is not applicable as every shot is different from another one. Thus, mainly two other ways are common: One is a e.g. Ce:YAG crystal as fluorescent screen and recording of the beam spot replica by a CCD camera. In this case, the screen prevents saturation and damage of the CCD and furthermore allows to increase the spatial resolution by optical lens systems. Another way to obtain a footprint of the FEL beam is to investigate ablation craters in PMMA. However, this is more wavelength-sensitive [8].

In both cases, the focused spotsize and transverse intensity distribution are observed. Furthermore, information about beam divergence can be obtained. For those kind of beam diagnostics, special attention has to be paid to the difference between several FEL shots. Therefore, an averaging mode or shot-to-shot diagnostics are applied. While the first one tells about stability and main parameters of the FEL, the latter one is also very important for experimental interpretation. Especially the fluorescent crystal method is well suited to discover the so called pointing stability, i.e. the relation between lateral shot-to-shot positions. Achieving a high pointing stability is very important for experimental alignment.

## 4. The MATLAB Program: beamspotdiagnostics

As summer student project, a program with graphical user interface (GUI) was written. It consists of the main program `beamspotdiagnostics.m` with GUI binary file `beamspotdiagnostics.fig`<sup>1</sup> as well as a sub-program which exists in two different versions, `beamspotevaluation1.m` and `beamspotevaluation2.m`, respectively.

<sup>1</sup>The binary `.fig` file is not necessarily needed. There is another version of `beamspotdiagnostics.m` that already contains the GUI appearance information.

The purpose of the program is to analyze camera images from beam replica on a fluorescent crystal. For that, an arbitrary number of images can be chosen and evaluated at one run. Figure 9 shows the GUI which was created with the MATLAB tool GUIDE. The upper part of the window is used to choose the files and region of interest. In the middle, evaluation parameters and information for output data are set. Finally, in the lower part, evaluated data is plotted.

In the upper listbox on the left, files are chosen from the local directories. Each chosen file appears in the listbox below where a file list is collected for an evaluation run. (Filenames can be deleted from this list again using the 'delete' button.) It is important, that all selected files are saved in the same folder. Whenever a file is chosen using double mouse click or return key at any listbox, it is plotted at the same time on the right. To define a region of interest (where the beam spot appears), the user must click on the image plot. Then a crop tool will appear allowing to define a rectangle within the image. Afterwards, this region is plotted separately below.

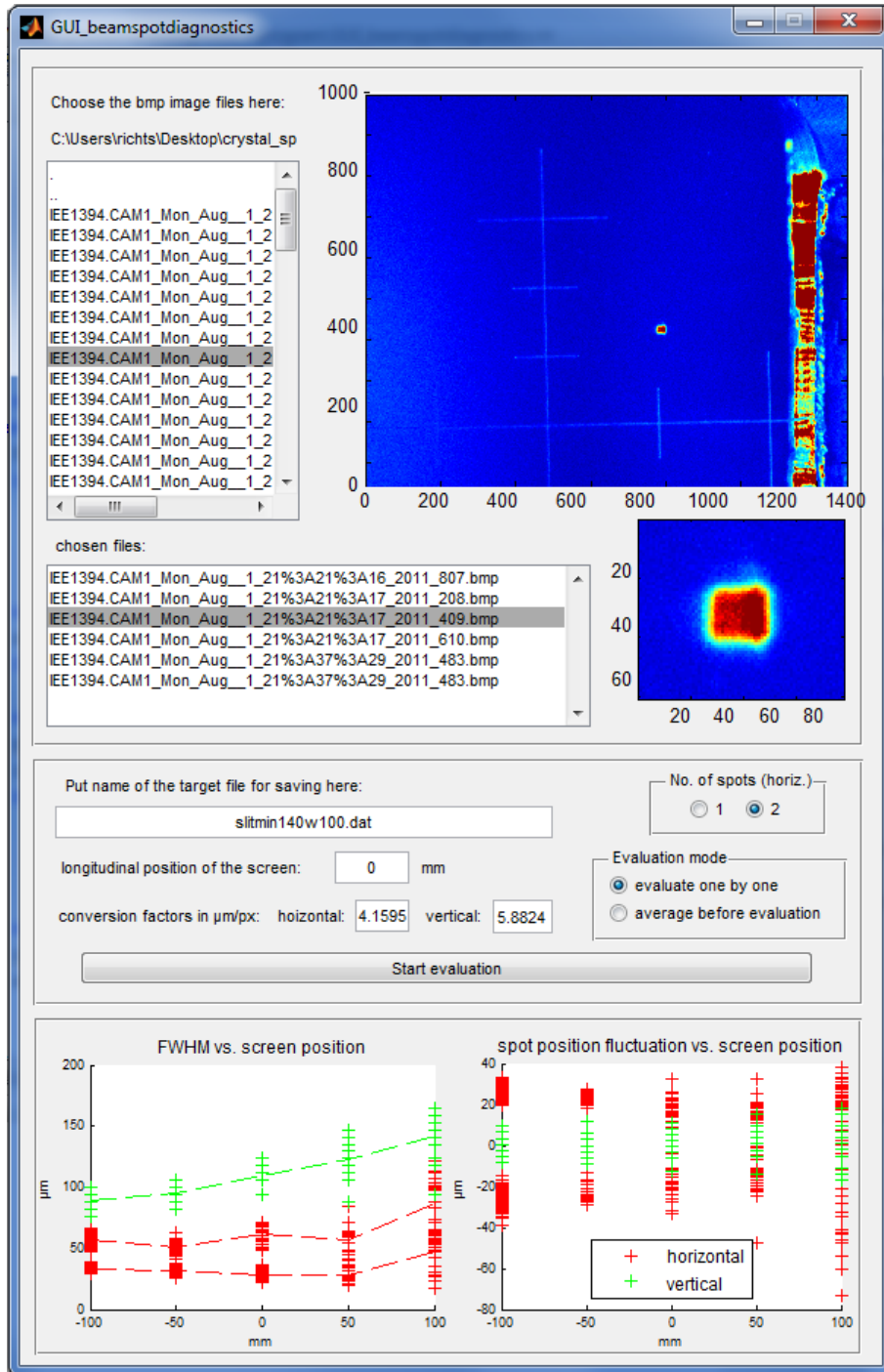
There are two pairs of radio buttons to select the mode of evaluation: First, it is possible to define if the data from all chosen images shall be averaged before running the evaluation or if single images should be evaluated one by one. In the latter case the average values of these single evaluations are also calculated and plotted afterwards. The second selection is if one or two spots are assumed in horizontal direction. The reason for this option is that often a slightly mis-aligned FEL leads to two laterally separated spots instead of one as it should be the case.

When the evaluation is run, for each image (or for the average of the chosen images, respectively) the sub-program `beamspotevaluation1.m` (one spot in horizontal direction) or `beamspotevaluation2.m` (two spots) is executed. These programs integrate the intensities of the images within the chosen region of interest along horizontal and vertical directions. Then, for the horizontal intensity curve a gaussian function (or two respectively) is fitted to obtain the full width at half maximum (FWHM) and absolute spotposition. For the vertical intensity distribution, the programs directly search for half maximum values (cf. figure 10).

When the evaluation is done the results for FWHM and spot position are plotted in the bottom area of the user window. As abscissa, the longitudinal screen position is taken. This longitudinal position of the crystal must be set in the respective textfield above. The scale of FWHM and position are plotted in  $\mu\text{m}$ , based on the conversion factors the user sets in the respective textfields. Default values are  $4.1595\mu\text{m}/\text{pixel}$  in horizontal and  $5.8824\mu\text{m}/\text{pixel}$  in vertical direction (due to a  $45^\circ$  tilted crystal). For the FWHM, the meanvalues for the longitudinal positions are calculated and plotted as line, for the positions, the meanvalues are subtracted from each data point in the plot.

There are several output files produced by the beam diagnostics program: The evaluated data is saved in file '`separately_filename`' or '`averaged_filename`', respectively, depending on the evaluation mode chosen. Here, *filename* is the string that has to be defined by the user in the corresponding textfield. These files contain the following seven columns:

position [mm]	fwhmx1 [px]	fwhmx2 [px]	positionx1 [px]	positionx2 [px]	fwhmy [px]	positiony [px]
---------------	-------------	-------------	-----------------	-----------------	------------	----------------



**Figure 9:** Screenshot of the graphical user interface.



x is ment to be horizontal, y vertical direction. If the 'one spot'-mode is chosen instead of two spots the 3rd and 5th column are filled with '-1'.

A second output file with the name 'meanvalues\_<filename>' is produced. It contains four columns:

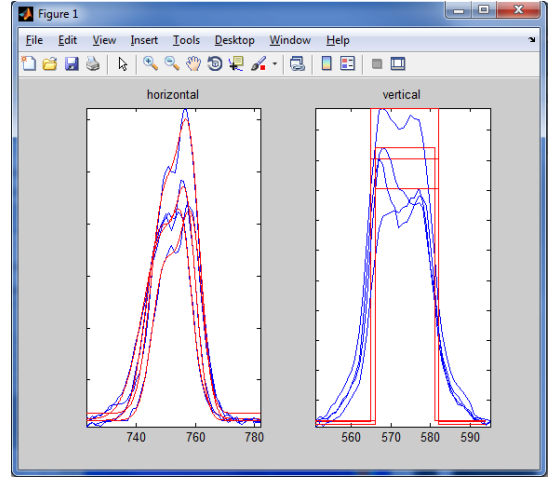
```
position [mm]    fwhmx1 [μm]    fwhmx2 [μm]    fwhmy [μm]
```

Here, the average values from the first output file are saved. Note that the values have been converted to  $\mu\text{m}$ .

If the 'average before evaluation' mode is selected, the an average image is also saved. Its name is 'averaged\_<filename>.bmp'.

If the 'run evaluation' button is pressed, all files visible in the second listbox are included to the evaluation. After running the evaluation for the whole list the evaluation data is saved in the corresponding file. Plotting and average value calculation is done from this file every time again. This is to make sure that already existing data is not neglected. If the respective output files already exist, the meanvalue file is overwritten while the evaluation data file is extended with the new data. The plots in the main window are held until the program is closed.

Finally it is important to mention that after each run of an evaluation, an additional plot of data and fit to the intensity curves appears. That gives the user the possibility to make sure that everything worked fine. An example plot window is shown in figure 10. If it gets obvious from this plot that the evaluation failed, saved data should be deleted by hand from the output files.



**Figure 10:** Plot of integrated intensities along horizontal and vertical direction within the chosen region of interest for four different shots/spot images. The fit curves have been plotted as well.

## 5. Data Evaluation

During the summer student project, the developed program has been applied to several sets of data. Two measurement campaigns from May and August 2011 contain series with different longitudinal crystal positions to find out about focused beam size and longitudinal position of the focal plane. Further, the pointing stability has been investigated.

The individual series mainly differ from each other in slitwidths and partly also positions. Between May and August, the angle of the focusing mirror has been changed slightly.

For all measurements evaluated here, FLASH was driven in 'single shot mode', i.e. no trains of electron bunches but single bunches are used in the FEL to produce pulses at a

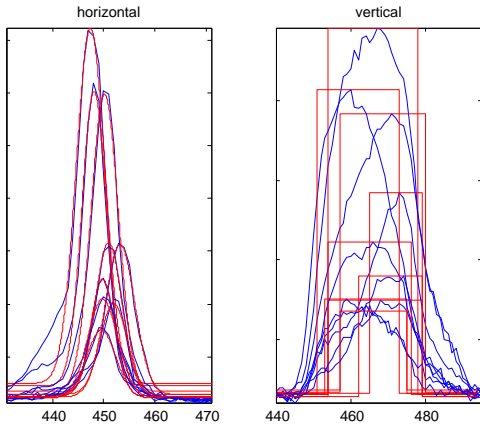
rate of e.g. 10Hz. As described above, the FEL beam is treated with a monochromator at the PG2 beamline. The single pulse energy was in the order of 5 $\mu$ J for all measurements. However, the dependence of the beam properties on the pulse energy is not examined here.

For the pixel-to- $\mu$ m conversion factors for the camera/lens system, 4.1595 $\mu$ m/pixel in horizontal and 5.8824 $\mu$ m/pixel in vertical direction have been used. Errors are in the order of 1 pixel.

For the longitudinal position, '-' means closer to the mirror while '+' is further away from it.

## 5.1. Data from May 2011

For the measurements in May, images from single shots at wavelength  $\lambda = 19$ nm were recorded. They have been evaluated in both modes, averaged and single evaluation. Since the FEL alignment was obviously quite good, only one beam spot appears (cf. figure 19). However, a first view on a series of images reveals that the spot positions change from shot to shot. This is illustrated in figure 11. Possibly, these fluctuations arose from vibrations of the diagnostics port. Those can be caused by the stage scaffold or by a slightly swinging sample holder.



**Figure 11:** The plot of data and fit for a measurement from the series 'new SMU' (exit slit 50 $\mu$ m) illustrates especially the horizontal position change between several shots.

The measure campaign from May contains two main series differing in the lateral and longitudinal position of the exit slit system shaping the beam in vertical direction. The series 'slit in focus' is self-explaining, for the series 'new SMU', the switchable mirror unit (SMU) was aligned to achieve a 1:1 image of the slitwidth on the crystal screen (cf. figure 8).

Both series contain sub-series for different exit slitwidth, namely 20, 50, 100 and 200 $\mu$ m. For each of them again three different longitudinal screen positions were used.

Additional changes in a further aperture unit were done for the series 'new SMU' with 100 $\mu$ m exit slit. Table 2 gives an overview about the available measurements and changed parameters. Each series consists of a number between 9 and 30 images.

Results of the spot width evaluation are shown in figure 12 and 13, respectively. For the horizontal spot width, the focal plane was obviously matched very well in the case of the 'slit in focus' series. The exact focal plane can be assumed a bit further away from the exit slit. That could decrease the focused spot width of 20-30 $\mu$ m even more. For the 'new SMU' series, the result is similar but the exact focal plane would be expected a little closer to the exit slit. Thus, the zero position is a good approximation of the



**Table 2:** Parameters for the different measurement series from May 2011.

exit slitposition	exit slitwidth	aperture unit	screen position
in focus	20 $\mu$ m	-	-100, 0, 100mm
	50 $\mu$ m	-	-100, 0, 100mm
	100 $\mu$ m	-	-100, 0, 100mm
	200 $\mu$ m	-	-100, 0, 100mm
new SMU	20 $\mu$ m	-	-100, 0, 100mm
	50 $\mu$ m	-	-100, 0, 100mm
	100 $\mu$ m	-	-100, 0, 100mm
	100 $\mu$ m	WAU 1	-100, 0, 100mm
	100 $\mu$ m	WAU 0.7	-100, 0, 100mm
	200 $\mu$ m	-	-100, 0, 100mm

exact focal plane. As expected, the vertical exit slit does not influence the horizontal spot width.

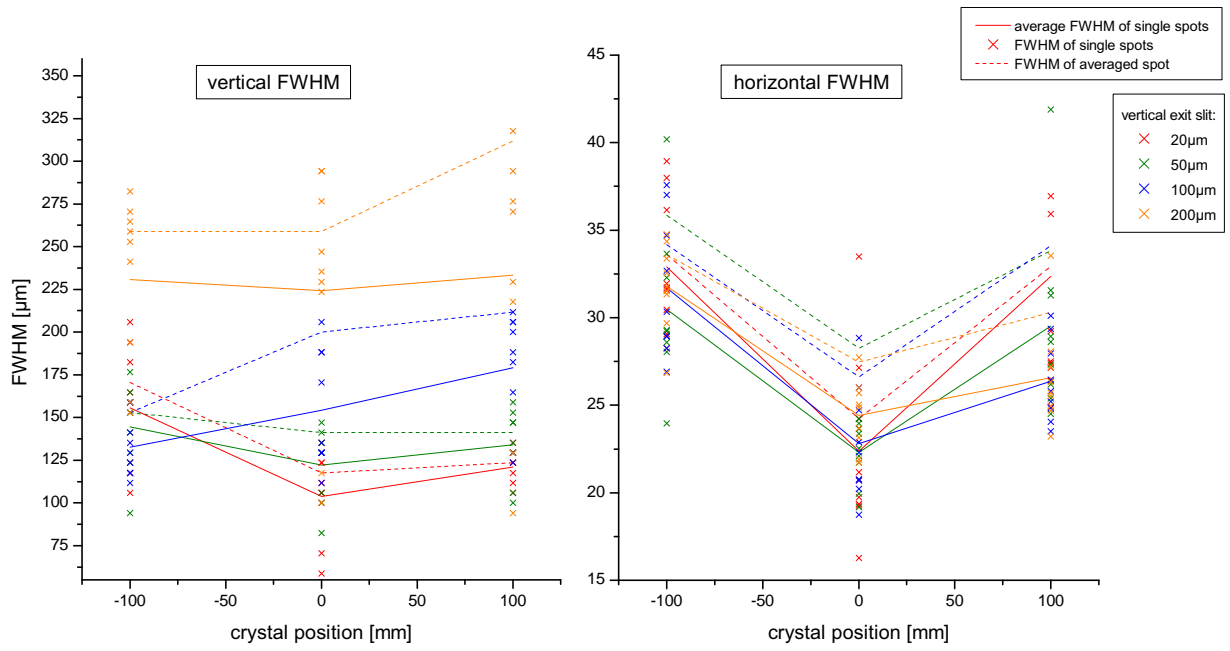
In vertical direction, the dependence of the spot width on the slitwidth is clearly visible. However, the exact behaviour is far away from 1:1 imaging of the slitwidth. At zero screen position, the behaviour of the meanvalue for the single spot width can roughly be described as  $spotwidth = 0.65 * slitwidth + 100\mu m$ .

The dependence on the screen position seems to be widely dominated by fluctuations even if minima appear around the zero position. It is not clear why the series for 100 $\mu$ m exit slit shows a different behavior regarding to the vertical spot width. One reason can be a lateral mis-alignment of the slit system. That could also explain the difference of the curve slopes, i.e. for 20 and 50 $\mu$ m the smallest vertical spot width tends to appear at slightly positive screen positions (further away from the exit slit) while for 100 and 200 $\mu$ m it behaves vice versa.

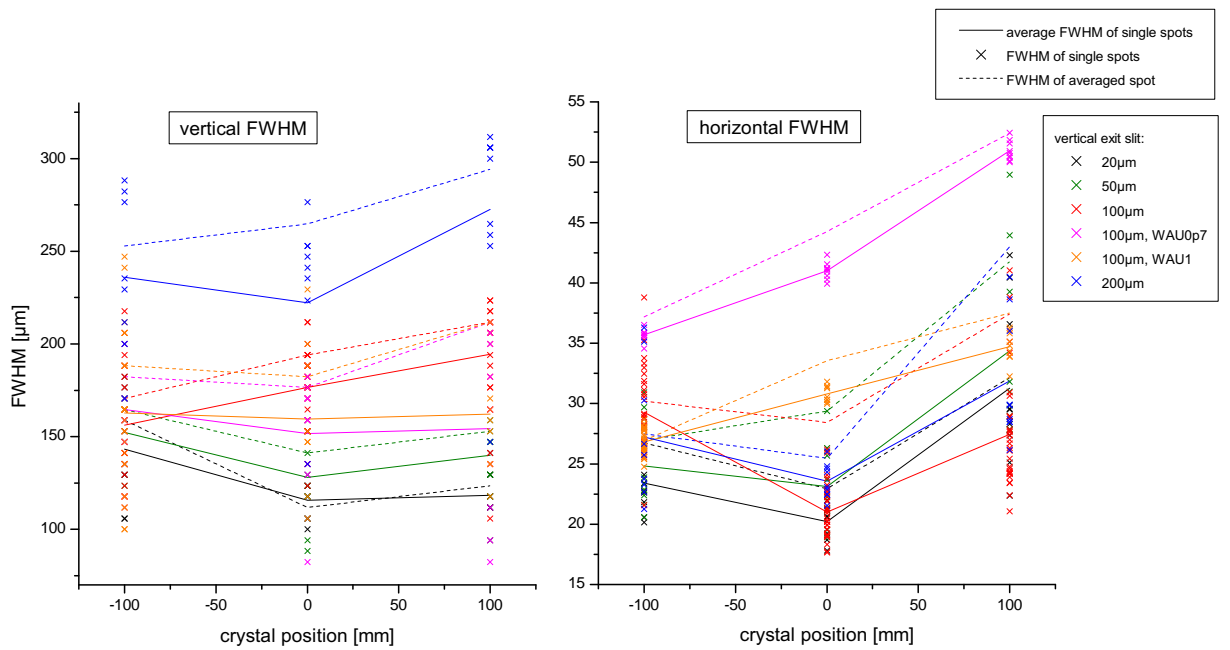
Comparing the results from the evaluation of averaged data and the averages of single shot evaluations, the averages FWHM tend to be 10-40 $\mu$ m larger in vertical and in the order of 5 $\mu$ m larger in horizontal direction. This was expected due to spot position fluctuations (cf, figure 11). The shot-to-shot fluctuation broadens the spot more than 15% in vertical and more than 10% in horizontal direction.

For the series 'new SMU', there is also a dependence of the spot width on the change of a water-cooled aperture unit (WAU): While the vertical FWHM curves are similar for all WAU settings, the horizontal spot width is dramatically changed from approximately 40 to 30 to 20  $\mu$ m (at zero screen position) from WAU 0.7 to WAU 1 to the usual setting. Thus, the setting without aperture changes can be assumed to correspond to WAU x with  $x > 1$ .

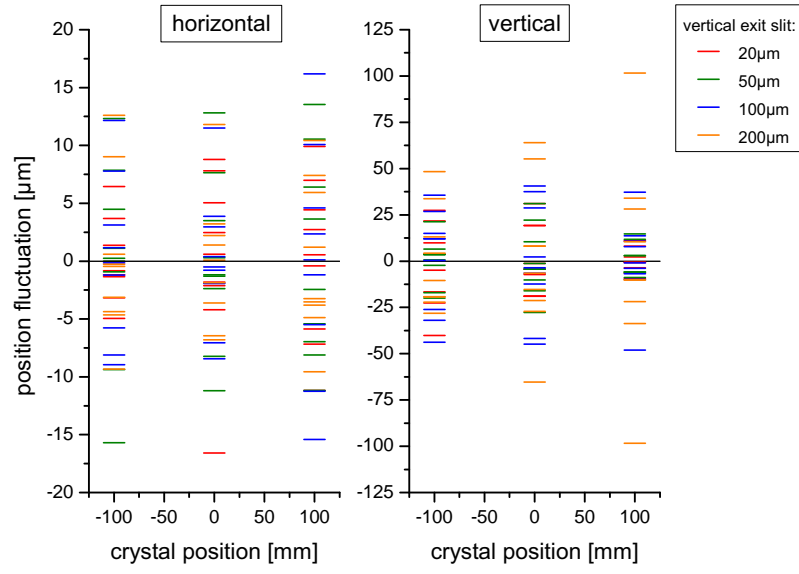
A closer look on the pointing stability (single shot evaluation) is done in the plots 14 and 15. The horizontal position fluctuations from one shot to another are spread up to  $\pm 15\mu m$  (at zero crystal position). That corresponds to about 50% of the spotwidth and verifies once again that the main reason for the difference between single spot and



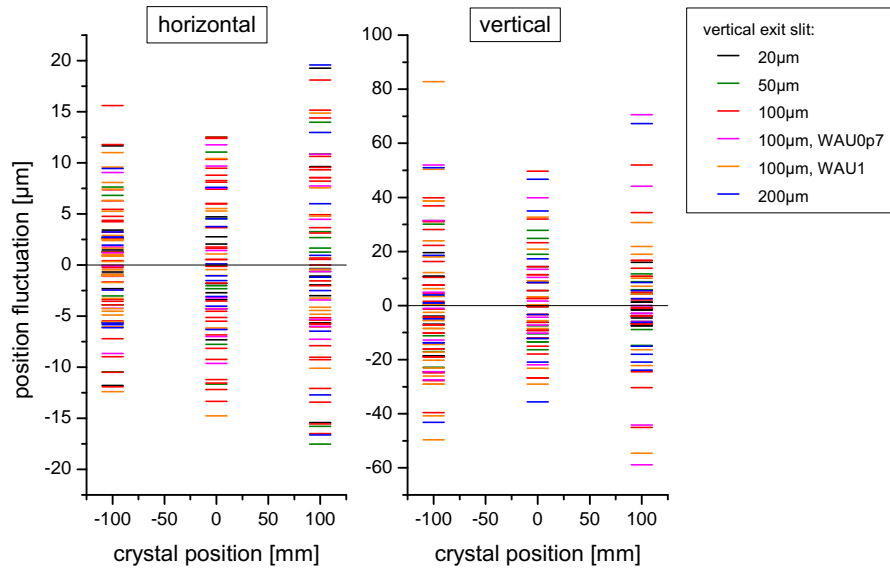
**Figure 12:** Lateral width of the focused beam on the fluorescent crystal. Measurement series 'slit in focus', May 2011.



**Figure 13:** Lateral width of the focused beam on the fluorescent crystal. Measurement series 'new SMU', May 2011.



**Figure 14:** Pointing fluctuation of the FEL beam for measurement series 'slit in focus', May 2011.



**Figure 15:** Pointing fluctuation of the FEL beam for measurement series 'new SMU', May 2011.

average evaluation is a bad pointing stability. The horizontal pointing stability is independent on the vertical exit slit.

The vertical pointing fluctuation can reach more than  $\pm 50\mu\text{m}$  (again at zero crystal position). It scales with the exit slit roughly like  $fluctuation = \pm 0.5 * (0.45 * slitwidth + 30\mu\text{m})$ . The additional aperture does not affect the pointing stability. Position fluctuations increase also if the focal plane is not matched by the longitudinal crystal position.

## 5.2. Data from August 2011

For the measurements from August, the angle of the focusing mirror was changed by 1mrad. A wavelength of  $\lambda = 6.3\text{nm}$  was used. The diagnostics port scaffold was made more stable in order to avoid vibrational influences. However, two different problems arose: First, due to a slightly mis-aligned FEL, two laterally separated beam spots appear and show up strong intensity fluctuations from shot to shot. Second, all images are the sum of two shots as the camera took data with 5Hz rate, while the FEL worked at 10Hz pulse rate.

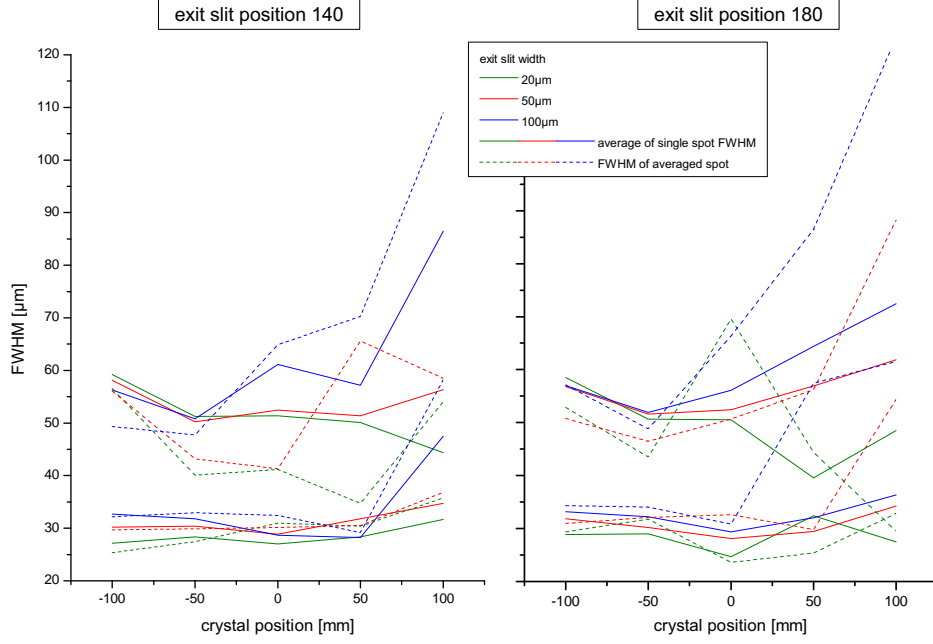
Again, measurements with different longitudinal crystal positions were performed for different slitwidths. There are two main series differing in the longitudinal position of the exit slit. The effect on the vertical spotsize can be investigated. An overview about the measurements series is given in table 3. Each series consists of at least 20, maximum 60 images.

**Table 3:** Parameters for the different measurement series from August 2011.

longitud. exit slitpos.	exit slitwidth	screen position
140	20 $\mu\text{m}$	-100, -50, 0, 50, 100mm
	50 $\mu\text{m}$	-100, -50, 0, 50, 100mm
	100 $\mu\text{m}$	-100, -50, 0, 50, 100mm
180	20 $\mu\text{m}$	-100, -50, 0, 50, 100mm
	50 $\mu\text{m}$	-100, -50, 0, 50, 100mm
	100 $\mu\text{m}$	-100, -50, 0, 50, 100mm

The plots in figures 16 and 17 contain the spot widths (FWHM) in horizontal and vertical direction for both exit slit positions. Again, the horizontal focus plane is matched quite well. At the same time strong fluctuations affect the curves. Those result from the two overlapping beam spots. Two spots can arise e.g. from wrong time structure due to not exact RF phase during electron bunch acceleration in the FEL. Such a disturbed bunch structure can be transferred into a changed lateral structure in the undulator. The fluctuations have even more effect than the difference between single shot evaluation and averaged one.

In vertical direction, the focal plane is obviously closer to the exit slit. This is expected since a 1:1 imaging of the exit slitwidth on the crystal was wanted for zero screen position (cf. figure 8) and represents the astigmatism of the system. However, the

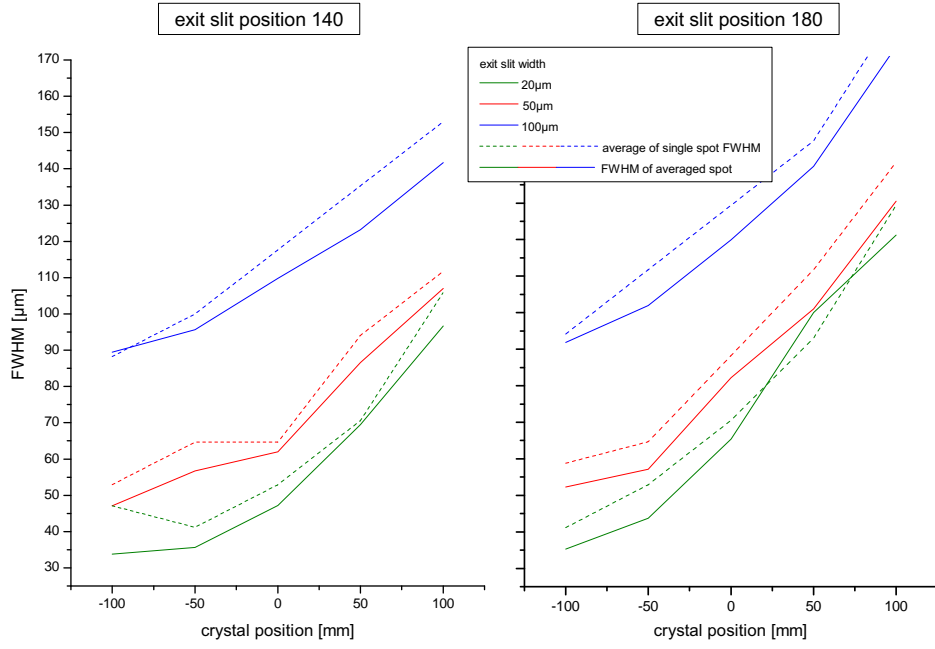


**Figure 16:** Horizontal spotsize from the measurements in August 2011. For each condition two spots appear - a smaller and a larger one.

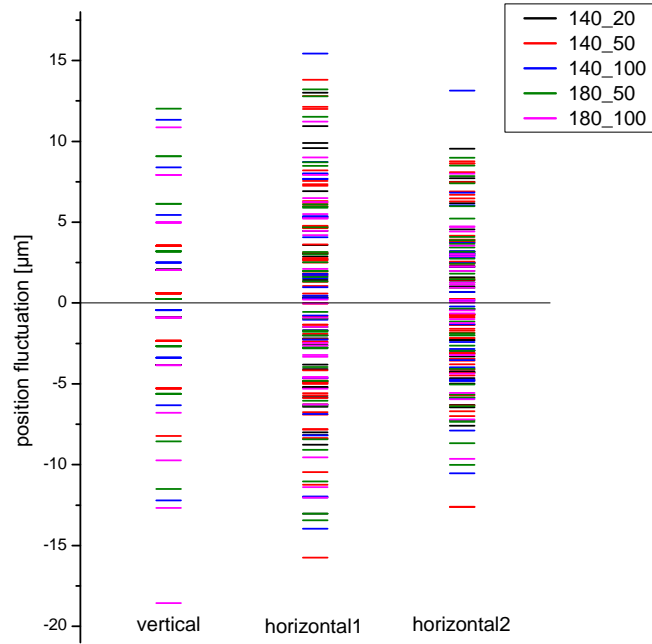
vertical spotwidth is larger than the exitslit width. The spotwidth follows approximately  $0.75 \cdot \text{slitwidth} + 30\mu\text{m}$  for slitposition 140 and  $0.60 \cdot \text{slitwidth} + 50\mu\text{m}$  for slitposition 180. The horizontal spotwidth is also slightly affected, probably due to unexact alignment. The two beam spots have horizontal width of about 30 and 50μm, respectively.

Even if the appearance of two laterally separated beam spots cause strong fluctuations in the corresponding spotwidths there is less vertical spotposition fluctuation in the August data than in May. Figure 18 confirms that vertical and horizontal pointing stability are in the same range of about  $\pm 15\mu\text{m}$ . Reasons for the vertical improvement can be the new additional stabilization of the diagnostics port or better exitslit alignment.

Nevertheless, there is a further notable fact about the intensity curves (cf. figures 10 and 11): Regarding to the machine setup we assume a gaussian shape in horizontal direction and a constant intensity vertically. However, the vertical intensity curve does not show a flat plateau (top of a broadened gaussian) with sharp edges from the exitslit. Instead, several maxima on changing positions appear. This is probably because of fluctuations, spatial irregularities or a slight misalignment.



**Figure 17:** Vertical spotsizes from the measurements in August 2011.



**Figure 18:** Pointing stability from the measurements in August 2011 at longitudinal crystal position 0. The legend information is *slitposition\_slitwidth*. The two horizontal spots are treated separately.

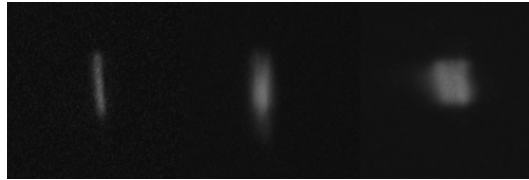
### 5.3. Conclusion

One of the biggest challenges when working with a free-electron laser is the fluctuation from one pulse to another one. Intensity (distribution) fluctuations become even more prominent after monochromatization, amplifying this fluctuation. Next to focused spotwidth and intensity especially the pointing stability is an important beam property factor. Thus, performing beam diagnostics is crucial.

In this project, the lateral behavior of the beam at FLASH has been investigated with respect to its width and position. The focused spot of a monochromatized beam at the PG2 beamline can be minimized down to about 20 $\mu\text{m}$  (May 2011) if the FEL alignment is good. In other cases a mis-alignment can limit the lateral pointing to several spots of this size or larger (August 2011). The position change from shot to shot was observed to be in the order of  $\pm 15\mu\text{m}$  in the focused direction. Thus the minimum effective spotsize that can be used in an experiment is limited not only by focusing. It is determined to the same extend by the pointing fluctuations. Figure 19 shows typical spot shapes recorded from the Ce:YAG crystal here.

Since focusing for a soft x-ray laser beam is a sophisticated task, astigmatism and alignment limitations restrict the mentioned focused spotsize to only one dimension while the other one is usually determined by aperture or slit systems. The exact alignment of such optics is a further challenge as has been seen by the effect of the exit slit on the vertical (and partly even on the horizontal) spotsize.

The knowledge about lateral FEL beam properties can help to plan and align experiments. Despite, the used method is direct and cannot be combined with an experiment at the same time. Therefore the improvement of non-invasive diagnostics methods is needed. An alternative way to get better control about the beam properties is seeding. This would prevent the pulses from arising from noise and make them more equivilant.



**Figure 19:** Typical spotshapes to illustrate the dependence on the alignment and lasing conditions. The spot on the left is a single shot from the measurements in Mai 2011. For the spot in the middle, it was integrated over such spots. The broadening results from position fluctuations. The spot on the right is a typical example of the measurements in August 2011 where the alignment was not as good as in May. For all shown spots, the vertical exit slit was set to 100 $\mu\text{m}$

# APPENDIX

## A. Source Code

The shown source code is only the part containing the most important functionality. To determine layout and graphics object handling there is a lot more code.

### A.1. Extract of GUI\_beamspotdiagnostics.m

```
1 function varargout = GUI_beamspotdiagnostics(varargin)

9 % This GUI helps to evaluate a set of beamspot images (e.g. from a
10 % flourescent crystal) with respect to the spot sizes and positions in
11 % horizontal and vertical direction. For evaluation, it can be chosen
12 % between single evaluation for each image or averaging over all chosen
13 % images before evaluation. Furthermore it can be chosen whether one or
14 % two gaussian spots occure in horizontal direction. If two are selected,
15 % they are treated separately.
16 %
17 % Results are saved into an ASCII file containing 7 columns:
18 % 1: lateral position of the screen [mm] (to be typed in at the GUI)
19 % 2: width of the first gaussian in x-dimension [px]
20 % 3: width of the second gaussian in x-dimension [px] (only if 2 spots)
21 % 4: absolute position of the first maximum in x-dimension [px]
22 % 5: absolute position of the second maximum in x-dimension [px] (only
23 % if 2 spots)
24 % 6: width in y-dimension [px]
25 % 7: position in y-dimension [px]
26 % Depending on the chosen evaluation mode, the name of this file is either
27 % 'averaged_typedfilename' or 'separately_typedfilename'. If the file
28 % already exists, data is appended.
29 %
30 % Average values are saved in an additional ascii file containing 4 columns
31 % 1: lateral position of the screen [mm] (to be typed in at the GUI)
32 % 2: average width of the first gaussian in x-dimension [m]
33 % 3: average width of the second gaussian in x-dimension [m] (only if 2
34 % spots)
35 % 4: average width in y-dimension [m]
36 % The name of this file is 'meanvalues_typedfilename'. It is overwritten
37 % if
38 % already existing.

99 % Global variables
100 handles.roicordinates=[1 1 1 1];
101 handles.imagebuffer=[];

137 % CALLBACK FUNCTIONS TO GUI ELEMENTS:
138
139 % — Executes on selection change in listbox1.
```



```

140 function listbox1_Callback(hObject, eventdata, handles) % list of current
    folder content
141 % hObject    handle to listbox1 (see GCBO)
142 % eventdata  reserved – to be defined in a future version of MATLAB
143 % handles    structure with handles and user data (see GUIDATA)
144
145 set(handles.listbox1, 'HitTest', 'off');
146 if strcmp(get(handles.figure1, 'SelectionType'), 'open')
147     file_list = get(handles.listbox1, 'String');
148     index_selected = get(handles.listbox1, 'Value');
149     filename = file_list{index_selected};
150     if handles.is_dir(handles.sorted_index(index_selected))
151         cd (filename)
152         load_listbox(pwd, handles)
153     else
154         [path, name, ext] = fileparts(filename);
155         if strcmp(ext, '.bmp')
156             try
157                 entries = get(handles.listbox2, 'String');
158                 entries{end+1}=filename;
159                 entries=sort(entries);
160                 set(handles.listbox2, 'String', entries);
161                 handles.imagebuffer=imread(filename);
162                 guidata(hObject, handles);
163                 set(handles.figure1, 'CurrentAxes', handles.axes2);
164                 image(handles.imagebuffer, 'HitTest', 'off');
165                 set(handles.axes2, 'ButtonDownFcn', {@axes2_ButtonDownFcn,
                    handles});
166             catch ex
167                 errordlg(...
168                     ex.getReport('basic'), 'File_Type_Error', 'modal')
169             end
170         end
171     end
172 end
173 set(handles.listbox1, 'HitTest', 'on');
174
175 % — Executes on key press with focus on listbox1 and none of its
    controls.
176 function listbox1_KeyPressFcn(hObject, eventdata, handles) % list of
    current folder content
177 % hObject    handle to listbox1 (see GCBO)
178 % eventdata  structure with the following fields (see UICONTROL)
179 %      Key: name of the key that was pressed, in lower case
180 %      Character: character interpretation of the key(s) that was pressed
181 %      Modifier: name(s) of the modifier key(s) (i.e., control, shift)
    pressed
182 % handles    structure with handles and user data (see GUIDATA)
183 if strcmp(eventdata.Key, 'return') & get(handles.listbox1, 'Value')
184     set(handles.listbox1, 'HitTest', 'off');
185     file_list = get(handles.listbox1, 'String');
186     index_selected = get(handles.listbox1, 'Value');

```

```

187     directories=0;%check if only files selected
188     for a=1:length(index_selected)
189         if isdir(file_list{index_selected(a)})
190             directories=1;
191         end
192     end
193     if directories==0
194         for a=2:length(index_selected)
195             filename = file_list{index_selected(a)};
196             [path,name,ext] = fileparts(filename);
197             if strcmp(ext, '.bmp')
198                 try
199                     entries = get(handles.listbox2, 'String');
200                     entries{end+1}=filename;
201                     set(handles.listbox2, 'String', entries);
202                     handles.imagebuffer=imread(filename);
203                     guidata(hObject, handles);
204                     set(handles.figure1, 'CurrentAxes', handles.axes2);
205                     image(handles.imagebuffer, 'HitTest', 'off');
206                     set(handles.axes2, 'ButtonDownFcn', {@axes2.ButtonDownFcn,
207                                                         handles});
208                 catch ex
209                     errordlg(...
210                         ex.getReport('basic'), 'File_Type_Error', 'modal')
211                 end
212             end
213         end
214         set(handles.listbox1, 'HitTest', 'on');
215     end
216
217
218 % — Executes on selection change in listbox2.
219 function listbox2_Callback(hObject, eventdata, handles) % list of chosen
220 % hObject    handle to listbox2 (see GCBO)
221 % eventdata  reserved — to be defined in a future version of MATLAB
222 % handles    structure with handles and user data (see GUIDATA)
223
224 get(handles.figure1, 'SelectionType');
225 if strcmp(get(handles.figure1, 'SelectionType'), 'open')
226     index_selected = get(handles.listbox2, 'Value');
227     file_list = get(handles.listbox2, 'String');
228     filename = file_list{index_selected};
229     [path,name,ext] = fileparts(filename);
230     if strcmp(ext, '.bmp')
231         try
232             handles.imagebuffer=imread(filename);
233             guidata(hObject, handles);
234             set(handles.figure1, 'CurrentAxes', handles.axes2);
235             image(handles.imagebuffer, 'HitTest', 'off');

```

```

236         set(handles.axes2, 'ButtonDownFcn', {@axes2_ButtonDownFcn,
           handles});
237     catch ex
238         errordlg(...
239             ex.getReport('basic'),'File_Type_Error','modal')
240     end
241 end
242 end
243
244 % — Executes on key press with focus on listbox2 and none of its
      controls.
245 function listbox2_KeyPressFcn(hObject, eventdata, handles) % list of
      chosen files
246 % hObject    handle to listbox2 (see GCBO)
247 % eventdata  structure with the following fields (see UICONTROL)
248 %           Key: name of the key that was pressed, in lower case
249 %           Character: character interpretation of the key(s) that was pressed
250 %           Modifier: name(s) of the modifier key(s) (i.e., control, shift)
      pressed
251 % handles    structure with handles and user data (see GUIDATA)
252 if strcmp(eventdata.Key, 'delete') & get(handles.listbox2, 'Value')
253     file_list = get(handles.listbox2, 'String');
254     index_selected = get(handles.listbox2, 'Value');
255     file_list(index_selected) = [];
256     set(handles.listbox2, 'String', file_list);
257     set(handles.listbox2, 'Value', 1);
258     guidata(hObject, handles);
259
260 end
261
262 % — Executes on button press in pushbutton2.
263 function pushbutton2_Callback(hObject, eventdata, handles) % Button to
      start evaluation
264 % hObject    handle to pushbutton2 (see GCBO)
265 % eventdata  reserved — to be defined in a future version of MATLAB
266 % handles    structure with handles and user data (see GUIDATA)
267 hold on;
268 %evaluation and saving
269 if ~strcmp(get(handles.listbox2, 'String'), '')
270     samples=get(handles.listbox2, 'String');
271     figure;
272     filename='';
273     switch get(handles.radiobutton1, 'Value')
274     case 0
275         filename=['separately_', get(handles.edit2, 'String')];
276         if ~exist(filename, 'file')
277             fid = fopen(filename, 'wt');
278             fprintf(fid, '%s%s%s%s%s%s%s', ['_position_ [mm] ', '_
279                 fwhmx1_ [px] _', '_fwhmx2_ [px] _', '_positionx1_ [px] ', '_
280                 positionx2_ [px] ', '_fwhmy_ [px] _', '_positiony_ [px] _'])
281             ;
282             fprintf(fid, '\n');
283             fclose(fid);

```

```

315     end
316     switch get(handles.radiobutton3, 'Value')
317         case 1
318             for a=1:length(samples)
319                 oneevaluation=spotevaluation1({samples{a}, round(
                    handles.roicoordinates(1)), round(handles.
                    roicoordinates(2)), round(handles.roicoordinates(3)
                    ), round(handles.roicoordinates(4)))}; % {inputfile
                    , xstart, ystart, xwidth, ywidth}
320                 oneevaluation=[str2num(get(handles.edit1, 'String')),
                    oneevaluation];
321                 save(filename, 'oneevaluation', '-ascii', '-append');
322             end
323         case 0
324             for a=1:length(samples)
325                 oneevaluation=spotevaluation2({samples{a}, round(
                    handles.roicoordinates(1)), round(handles.
                    roicoordinates(2)), round(handles.roicoordinates(3)
                    ), round(handles.roicoordinates(4)))}; % {inputfile
                    , xstart, ystart, xwidth, ywidth}
326                 oneevaluation=[str2num(get(handles.edit1, 'String')),
                    oneevaluation];
327                 save(filename, 'oneevaluation', '-ascii', '-append');
328             end
329     end
330 case 1
331     filename=['averaged_', get(handles.edit2, 'String')];
332     if ~exist(filename, 'file')
333         fid = fopen(filename, 'wt');
334         fprintf(fid, '%s_%s_%s_%s_%s_%s_%s', ['__position__[mm]__' '__
                    fwhmx1__[px]__' '__fwhmx2__[px]__' '__positionx1__[px]__' '__
                    positionx2__[px]__' '__fwhmy__[px]__' '__positiony__[px]__'])
                    ;
335         fprintf(fid, '\n');
336         fclose(fid);
337     end
338     sampleaverage=imread(samples{1})/length(samples);
339     for a=2:length(samples)
340         sampleaverage=sampleaverage+imread(samples{a})/length(
            samples);
341     end
342     sampleaverage=sampleaverage;
343     imwrite(sampleaverage,[filename, '.bmp'], 'bmp')
344     switch get(handles.radiobutton3, 'Value')
345         case 1
346             oneevaluation=spotevaluation1({'[filename, '.bmp'], round
                (handles.roicoordinates(1)), round(handles.
                roicoordinates(2)), round(handles.roicoordinates(3)),
                round(handles.roicoordinates(4)))}; % {inputfile ,
                xstart, ystart, xwidth, ywidth}
347         case 0

```

```

348         oneevaluation=spotevaluation2([filename, '.bmp'], round
            (handles.roicoordinates(1)), round(handles.
            roicoordinates(2)), round(handles.roicoordinates(3)),
            round(handles.roicoordinates(4))}); % {inputfile,
            xstart, ystart, xwidth, ywidth}

349     end
350     oneevaluation=[str2num(get(handles.edit1, 'String')),
        oneevaluation];
351     save(filename, 'oneevaluation', '-ascii', '-append');
352 end
353 end
354 hold off;
355 %plotting
356 plotbuffer=plotpreparation(handles);
357 resultdata=plotbuffer{1,1};
358 averages=plotbuffer{1,2};
359 set(handles.figure1, 'CurrentAxes', handles.axes4);
360 hold on;
361 p1=plot(resultdata(:,1), resultdata(:,2), 'r+'); %fwhmx1
362 plot(averages(:,1), averages(:,2), 'r—')% average fwhmx1
363 if ~sum(resultdata(1,3)<0)>0
364     plot(resultdata(:,1), resultdata(:,3), 'r+') %fwhmx2
365     plot(averages(:,1), averages(:,3), 'r—')% average fwhmx2
366 end
367 p2=plot(resultdata(:,1), resultdata(:,6), 'g+'); %fwhmy
368 plot(averages(:,1), averages(:,4), 'g—')% average fwhmy
369 title( 'FWHM vs. _screen_position', 'FontSize', 12);
370 xlabel( 'mm');
371 ylabel( 'm');
372 hold off;
373 set(handles.figure1, 'CurrentAxes', handles.axes5);
374 hold on;
375 p3=plot(resultdata(:,1), resultdata(:,4), 'r+'); %position x1
376 if ~sum(resultdata(1,3)<0)>0
377     p4=plot(resultdata(:,1), resultdata(:,5), 'r+'); %position x2
378 end
379 p5=plot(resultdata(:,1), resultdata(:,7), 'g+'); %position y
380 legend([p3 p5], {'horizontal' 'vertical'}, 'Location', 'Best');
381 title( 'spot_position_fluctuation_vs._screen_position', 'FontSize', 12);
382 xlabel( 'mm');
383 ylabel( 'm');
384 hold off;
385 %save averages of fwhm
386 filename=[ 'meanvalues_', get(handles.edit2, 'String') ];
387 fid = fopen(filename, 'wt');
388 fprintf(fid, '%s_%s_%s_%s', [ '_position_[mm]_' '_fwhmx1_[ m]_ _' '_'
        fwhmx2_[ m]_ _' '_fwhmy_[ m]_ _' ] );
389 fprintf(fid, '\n');
390 fclose(fid);
391 save(filename, 'averages', '-ascii', '-append');
392
393

```

```

394 % — Executes on mouse press over axes background.
395 function axes2_ButtonDownFcn(hObject, eventdata, handles) % image plot
396 % hObject      handle to axes2 (see GCBO)
397 % eventdata    reserved — to be defined in a future version of MATLAB
398 % handles      structure with handles and user data (see GUIDATA)
399     set(handles.figure1, 'CurrentObject', handles.axes2);
400     try
401         [roi rect] = imcrop(handles.axes2);
402         handles.roicoordinates=rect;
403         guidata(hObject, handles);
404         set(handles.figure1, 'CurrentAxes', handles.axes3);
405         set(handles.axes3, 'XLim', [0 rect(3)], 'YLim', [0 rect(4)]);
406         image(roi);
407     catch err
408         disp('try_again')
409     end

462 % FURTHER FUNCTIONS USED BY THE CALLBACKS :
463
464 % function to read the current directory and sort the names
465 function load_listbox(dir_path, handles)
466 cd (dir_path)
467 dir_struct = dir(dir_path);
468 [sorted_names, sorted_index] = sortrows({ dir_struct.name}');
469 handles.file_names = sorted_names;
470 handles.is_dir = [dir_struct.isdir];
471 handles.sorted_index = sorted_index;
472 guidata(handles.figure1, handles);
473 set(handles.listbox1, 'String', handles.file_names, ...
474     'Value', 1)
475 set(handles.text1, 'String', pwd)
476
477
478 % preparing data from the target file for plotting (incl. average
    calculation)
479 function plotdata=plotpreparation(handles)
480 switch get(handles radiobutton1, 'Value')
481     case 1
482         filename=['averaged_', get(handles.edit2, 'String')];
483     case 0
484         filename=['separately_', get(handles.edit2, 'String')];
485 end
486 resultdata = getfield(importdata(filename), 'data');
487 figure(handles.figure1);
488 % average value calculation for plot
489 resultdata=sortrows(resultdata,1);
490 screenposition = [];
491 averagesxw1 = [];
492 averagesxp1 = [];
493 averagesxw2 = [];
494 averagesxp2 = [];
495 averagesyp = [];
496 averagesyw = [];

```

```

497 c2=0;
498 c4=0;
499 a=1;
500 while a<=length(resultdata(:,1))
501     counter=1;
502     c1=resultdata(a,2);%xfwhm1
503     c3=resultdata(a,4);%xposition1
504     if resultdata(a,3)>=0
505         c2=resultdata(a,3);%xfwhm2
506         c4=resultdata(a,5);%xposition2
507     end
508     c5=resultdata(a,6);%yfwhm
509     c6=resultdata(a,7);%yposition
510     b=a+1;
511     while b<=length(resultdata(:,1)) & resultdata(a,1)==resultdata(b,1)
512         c1=c1+resultdata(b,2);
513         c3=c3+resultdata(b,4);
514         if resultdata(b,3)>=0
515             c2=c2+resultdata(b,3);
516             c4=c4+resultdata(b,5);
517         end
518         c5=c5+resultdata(b,6);
519         c6=c6+resultdata(b,7);
520         counter=counter+1;
521         b=b+1;
522     end
523     screenposition(end+1)=resultdata(a,1);
524     averagesxw1(end+1)=c1/counter;
525     averagesxp1(end+1)=c3/counter;
526     if resultdata(a,3)>=0
527         averagesxw2(end+1)=c2/counter;
528         averagesxp2(end+1)=c4/counter;
529     else
530         averagesxw2(end+1)=-1;
531         averagesxp2(end+1)=-1;
532     end
533     averagesyw(end+1)=c5/counter;
534     averagesyp(end+1)=c6/counter;
535     a=b;
536 end
537 %average position subtraction
538 b=1;
539 while b<=length(screenposition)
540     for a=b:length(resultdata(:,1))
541         if resultdata(a,1)==screenposition(b)
542             if resultdata(a,5)>=0
543                 resultdata(a,4)=resultdata(a,4)-0.5*(averagesxp1(b)+
544                     averagesxp2(b));
545                 resultdata(a,5)=resultdata(a,5)-0.5*(averagesxp1(b)+
546                     averagesxp2(b));
547             else
548                 resultdata(a,4)=resultdata(a,4)-averagesxp1(b);

```

```

547         end
548         resultdata(a,7)=resultdata(a,7)-averagesyp(b);
549     end
550 end
551 b=b+1;
552 end
553 %m-conversion
554 if str2num(get(handles.edit3,'String'))
555     factor=str2num(get(handles.edit3,'String'));
556     resultdata(:,2)=resultdata(:,2)*factor;
557     resultdata(:,3)=resultdata(:,3)*factor;
558     resultdata(:,4)=resultdata(:,4)*factor;
559     resultdata(:,5)=resultdata(:,5)*factor;
560     averagesxw1=averagesxw1*factor;
561     averagesxw2=averagesxw2*factor;
562 end
563 if str2num(get(handles.edit4,'String'))
564     factor=str2num(get(handles.edit4,'String'));
565     resultdata(:,6)=resultdata(:,6)*factor;
566     resultdata(:,7)=resultdata(:,7)*factor;
567     averagesyw=averagesyw*factor;
568 end
569 %return cell array
570 plotdata={resultdata, [screenposition', averagesxw1', averagesxw2',
    averagesyw']};

```

## A.2. Extract of spotevaluation1.m

This is the program doing the actual evaluation. There is another version where two spots are fitted in horizontal direction (instead of only one).

```

1 function results = spotevaluation1(input);
2 % input = {inputfile, xstart, ystart, xwidth, ywidth}
3 % This function reads in an image file and tries to find the position and
4 % width of a spot in x- and y-dimensions in an input-determined ROI.
5 % For vertical evaluation, the width is directly taken from the points
6 % with half maximum; for the horizontal evaluation, a gaussianfit is
   performed.
7 % The output is a vector containing the following 6 elements:
8 % 1: width of the gaussian in x-dimension
9 % 2: empty (entry = '-1')
10 % 3: absolute position of the gaussian in x-dimension
11 % 4: empty (entry = '-1')
12 % 5: width in y-dimension
13 % 6: position in y-dimension.
14 % Note: The spot must be in the center of the chosen ROI!
15
16 disp('begin_one_evaluation');
17
18 %> reading image file
19 imagebuffer=imread (input{1,1});
20 % image(imagebuffer);
21

```



```

22 %> setting ROI and integrate intensity
23 x=sum(imagebuffer(input{1,3}:input{1,3}+input{1,5}+1,input{1,2}:input
    {1,2}+input{1,4}+1));
24 y=sum(imagebuffer (input{1,3}:input{1,3}+input{1,5}+1,input{1,2}:input
    {1,2}+input{1,4}+1) ');
25
26 %> evaluation x
27 fitfunction1=fitttype('y0+(A1/(w1*sqrt(pi/2)))*exp_(-2*((x-xc1)/w1)^2)');
28 xoptions = fitoptions(fitfunction1);
29 xoptions.StartPoint = [5*max(x) 10 input{1,2}+.5*input{1,4} 0.5*(x(1)+x(
    end))];
30 xoptions.Lower = [0 0 input{1,2} 0];
31 xoptions.MaxIter = 1000;
32 xfit=fit((input{1,2}:1:input{1,2}+input{1,4}+1)',x',fitfunction1,xoptions)
    ;
33 xparameters=coeffvalues(xfit);
34 xfwlm=xparameters(2) % result
35 xposition1=xparameters(3) % result
36 %> evaluation y
37 yoffset=0.5*(y(1)+y(end));
38 ybuffer=find(y >= (max(y)-.5*(max(y)-yoffset)));
39 yfwhm=ybuffer(end)-ybuffer(1); % result
40 yposition=input{1,3}+ybuffer(1)+.5*yfwhm; % result
41
42 %> plotting x
43 subplot(1,2,1);
44 plot(xfit,input{1,2}:1:input{1,2}+input{1,4}+1,x,'-');
45 legend('off');
46 axis tight;
47 title('horizontal');
48 xlabel('');
49 ylabel('');
50 set(gca,'YTickLabel',{});
51 hold on;
52 %> plotting y
53 subplot(1,2,2);
54 plot(input{1,3}:1:input{1,3}+input{1,5}+1,y,[input{1,3} input{1,3}+ybuffer
    (1) input{1,3}+ybuffer(1) input{1,3}+ybuffer(end) input{1,3}+ybuffer(
    end) input{1,3}+input{1,5}], [yoffset yoffset max(y) max(y) yoffset
    yoffset'],'r');
55 axis tight;
56 title('vertical');
57 set(gca,'YTickLabel',{});
58 hold on;
59
60 %> saving results
61 a=-1;
62 results=[xfwlm a xposition1 a yfwhm yposition]; % results output = vector
    with the values input{1,2}, xfwlm, xposition, yfwhm, yposition
63
64 disp('end_one_evaluation');

```

## B. References

- [1] FLASH. The Free-Electron Laser in Hamburg, *Deutsches Elektronen-Synchrotron DESY*, May 2007
- [2] Femtosecond diffractive imaging with a soft-X-ray free-electron laser, *H.N. Chapman et al.*, Nature Physics 2 (2006) 839-843
- [3] Synchrotron Radiation - Production and Properties, *Rainer Gehrke*, DESY Summerstudents Lectures 2011
- [4] Soft X-Ray and Extreme Ultraviolet Radiation, *David Attwood*, Cambridge University Press, 2000
- [5] [http://hasylab.desy.de/science/studentsteaching/primers/synchrotron\\_radiation/index\\_eng.html](http://hasylab.desy.de/science/studentsteaching/primers/synchrotron_radiation/index_eng.html) (2011-09-02)
- [6] Physik der Teilchenbeschleuniger und Synchrotronstrahlungsquellen, *Klaus Wille*, B.G. Teubner, Stuttgart 1992
- [7] Accelerator X-Ray Sources, *Richard Talman*, WILEY-VCH, Weinheim 2006
- [8] A compendium on beam transport and beam methods for Free Electron Lasers, *A. Lindblad, S. Svensson, K. Tiedtke*, DESY/IRUVX-PP, Hamburg 2011
- [9] Undulators and Free-electron Lasers, *P. Luchini and H. Motz*, Clarendon Press, Oxford 1990
- [10] Beam-Wave-Interaction in Periodic and Quasi-Periodic Structures, *Levi Schächter*, Springer-Verlag, Berlin Heidelberg 1997
- [11] Free-Electron Laser, *Martin Dohlus*, DESY Summerstudents Lectures 2011
- [12] <http://flash.desy.de/>
- [13] Überblick über die DESY Beschleuniger, *Dirk Nölle*, [http://www.helmholtz-berlin.de/media/media/spezial/events/sei/Desy10/noelle\\_desy10.pdf](http://www.helmholtz-berlin.de/media/media/spezial/events/sei/Desy10/noelle_desy10.pdf)
- [14] The soft x-ray free-electron laser FLASH at DESY: beamlines, diagnostics and end-stations, *K. Tiedtke et al.*, New Journal of Physics 11 (2009) 023029
- [15] The Monochromator Beamline at FLASH - Assembly, Characterization and first Experiments, *Michael Wellhöfer*, dissertation, Universität Hamburg 2007