# Laser beam monitoring system for future application for 3D ellipsoidal laser system at PITZ

*DESY summer student program 2014*

Ashkhen Nalbandyan
*Yerevan State University, Armenia*

Supervisors
Tino Rublack, Dr.
James Good

1st September 2014

## Abstract

The **P**hoto **I**njector **T**est Facility at the DESY, **Z**euthensite (PITZ), is mainly orientated on the development of high quality electron sources for FLASH and European XFEL. One of the ongoing projects at PITZ is the generation of 3D quasi ellipsoidal laser pulses. A basic requirement is laser beam positional stability at different locations within the laser system and on the photo cathode. An algorithm to find the laser pulse center of mass and possibilities of controlling beam position with mirrors shall be discussed in this work. This algorithm provides the basic understanding of future laser beam position control.

## Contents

# 1. Introduction

Photo Injector Test Facility at DESY, Zeuthen site (PITZ), was built for optimization and development of high brightness electron sources for free electron lasers (FELs), such as FLASH and European XFEL. PITZ is meeting the challenge to hold its leading position in this area by advancing photo cathode laser system development. Research at PITZ has shown that laser pulse shaping is one of the key-points for the photo injector performance and one of the ways to minimize the electron beam emittance. There were done experiments based on the PITZ photo injector layout in order to demonstrate the advantage of using 3D ellipsoidal laser system and is was proofed that using this system it is possible to reduce the laser beam emittance significantly [1].The key point of the implementation of 3D ellipsoidal pulse shape is to get high brightness and small emittance of the laser beam.

Small beam size and small energy spread are very important for operating FELs. Also a precisely stabilized laser beam is needed for operation. Usually at facilities like PITZ various type of cameras (mostly CCD) are used to monitor beam position. Using a movable mirrors a beam position can be controlled in respect to keep it precise during operation time. In this work a MATLAB supported code will be discussed. This code should be able to calculate laser beam center of gravity coordinates. Also it would be discussed a MATLAB function which can create serial Comport object. This will be useful for a code which can move mirrors automatically.

## 1.1 Problem description

For effective operation of a high quality electron beam it is required to have stable laser beam. The stabilization can be in time or in space. This problem consists therefore of two main parts: determination of the center of gravity of the laser beam, and positional correction using piezoelectro-actuated mirrors. A MATLAB code was used for both parts.

## 1.2 Theory

Some mathematical equations for calculating laser beam center of gravity are provided. The method provided below is a general way to find the x and y coordinates of center of gravity of the laser beam.

### 1.2.1 Laser beam center of gravity

The center of gravity is the best way to represent the center of the beam as there is an equal amount of power in all directions from the center of gravity. The center of gravity of a laser beam is the same as the average value of the projection on the X and Y axes of the image. This is not necessarily the same point as the maximum power in the beam (maximum pixel on the image).

Let us assume that E(x,y) is the beam energy in the pixel (x,y),P is the total intensity (power) in the beam, <x> and <y> is the position for the center of gravity[1]. All integrals are done over the entire image. To calculate the center of gravity the following three equations are used [1,2]:

$$P = \int E(x,y)dxdy$$

$$<x>= \frac{1}{P}\int (E(x,y)\cdot x)dxdy,$$

and

$$<y>= \frac{1}{P}\int (E(x,y)\cdot y)dxdy\ .$$

All integrals are done over the entire image.

## 2 Experimental setup and MATLAB code logic

In this section the experimental setup is described, MATLAB code logic is explained, and the results are discussed. It is described what was used for experiment and why.

### 2.1 Experimental setup

A test bench was set up as an approximation to a real positional feedback system, this was then used to develop and evaluate a controller algorithm.(Figure. 1).

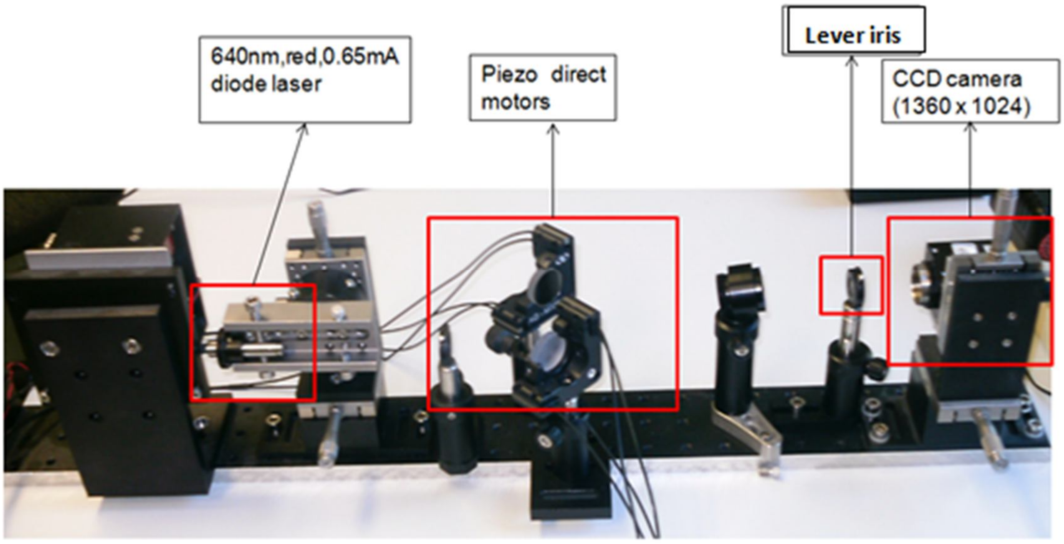

**Figure 1: Experimental setup**

The setup consisted of a 635nm, 1mW diode laser, a lever iris, piezoelectro-actuated mirror mounts(AG-M100L) with <THORLABS PF-10-30-G01> mirrors, a grey filter, and a 1360x1024 monochrome CCD camera(ALLIED Vision Technologies, AVT GC1350).
A laser diode(635 nm, 1 mW) was used as a laser source.
The iris was used to collimate and diffract the spatially broad laser beam into small airy-disc on the camera.

Piezo driven motors were used for moving mirrors. These mirrors must move due to control beam position. If for some reasons the position of the center of gravity changes the piezo motors must change the angle between the mirrors in order to keep the position stable.

In between the mirrors and the camera – a 1360x1024 pixel monochrome CCD camera (ALLIED Vision Technologies, AVT GC1350) has been used – a second iris was placed to cut out the higher order airy-rings.

The piezo motors are controlled via a USB controller.

ASCII commands may be sent directly to the COM port to move the mirrors. Due to some unclearness in the syntax for this problem a MATLAB function was written which allows creating serial port and connecting to the controller.

Camera interfacing is achieved via the functions provided by Matlab's Image Acquisition Toolbox. This permits control of camera properties such as region of interest (ROI), gain, or exposure time. These characteristics are very important. Specifying the ROI on the image reduces calculation time and excludes useless data. Exposure control is required for correction of saturation or the signal-to-noise ratio.

## 2.2 MATLAB code logic

The MATLAB code for calculating the laser beam center of gravity coordinates first checks if the image is saturated. If the array of the indices of the brightest pixels on the image is greater than one the image is saturated and the code stops with the error “image is saturated”. In the next step it would be better to improve code in this case to return to camera settings and automatically reduce exposure time.

If the image is not saturated the code proceeds to the “while”-loop. It detects the indexes of the brightest pixel on the image and looks for the best Gaussian fit in a banded area near the brightest pixel. Once the location of the brightest pixel is known the program takes 10 slices in both X and Y directions near the brightest pixel such as we have a band and starts to make Gaussian fits. For finding the best fit program checks ‘gof’ (goodness of fit parameter). This method is used for finding the center of gravity of the laser beam which is not necessarily at the same location as the brightest pixel of the image. The location of the center of gravity can obviously change due to some random noises. That is why program plots its value as a rolling buffer so one can easily follow the position changes. Figure 5 shows the final result of the code.

Once the best orthogonal fits are found the program generates a visualization of best fits and center of gravity position .The best fits and the center of gravity position for the last several iterations are then plotted. Figure 2 shows the flow chart of the code.

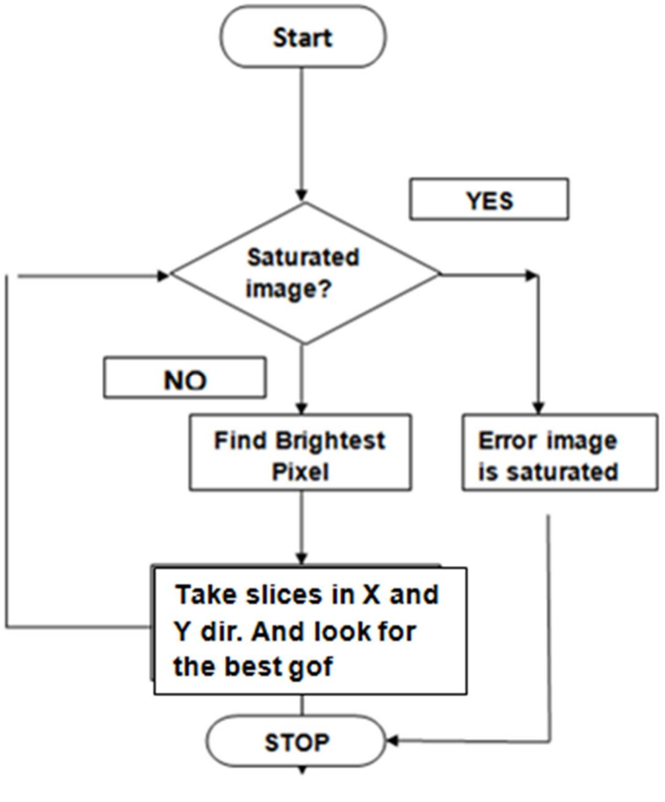

**Figure 2 MATLAB code flow chart**

The second part of the problem was figuring out how to move mirrors to control beam position. Therefore a function was written which allows to create serial port and change the angle between the mirrors in respect to the incidence angle of the beam on it. Figure 3 shows the code flow chart.

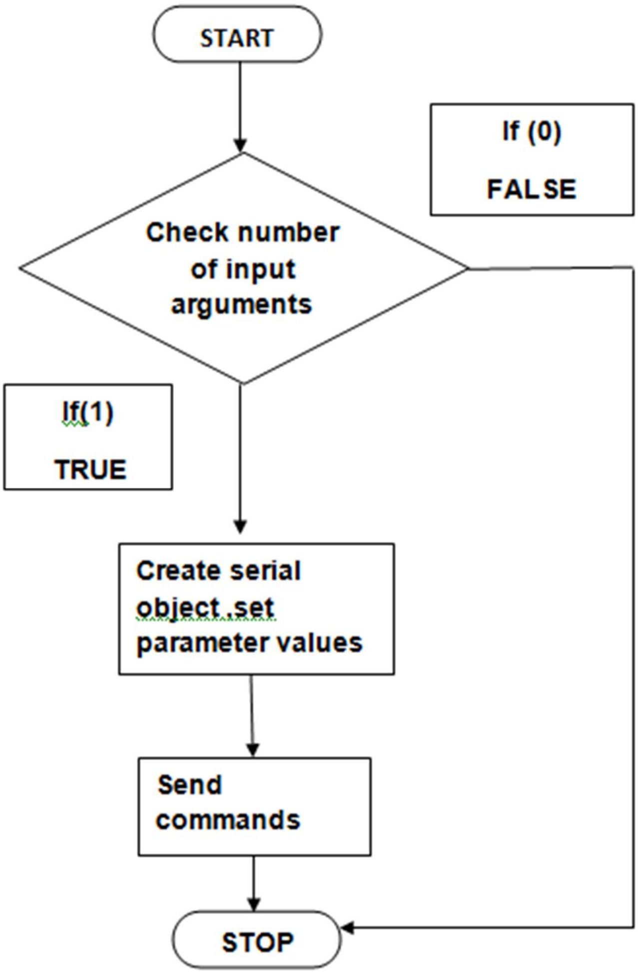

**Figure 3 Function flow chart**

## 3. Results

The script takes live, monochrome (greyscale) feed from the CCD camera (figure 4) and converts it into a false color image (figure 4).

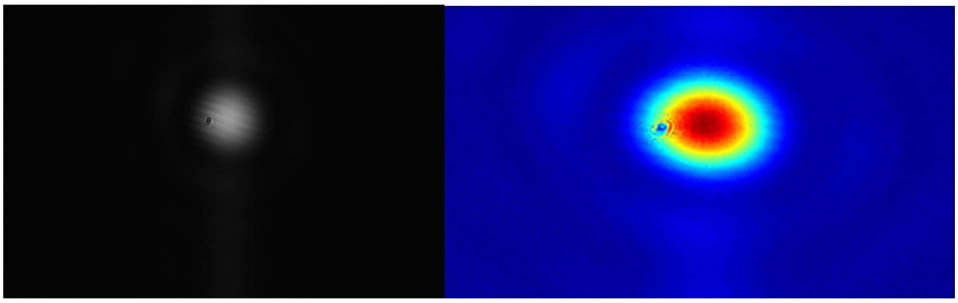

**Figure 4 Grayscale image and false color image**

Figure 5 shows the mask of the code

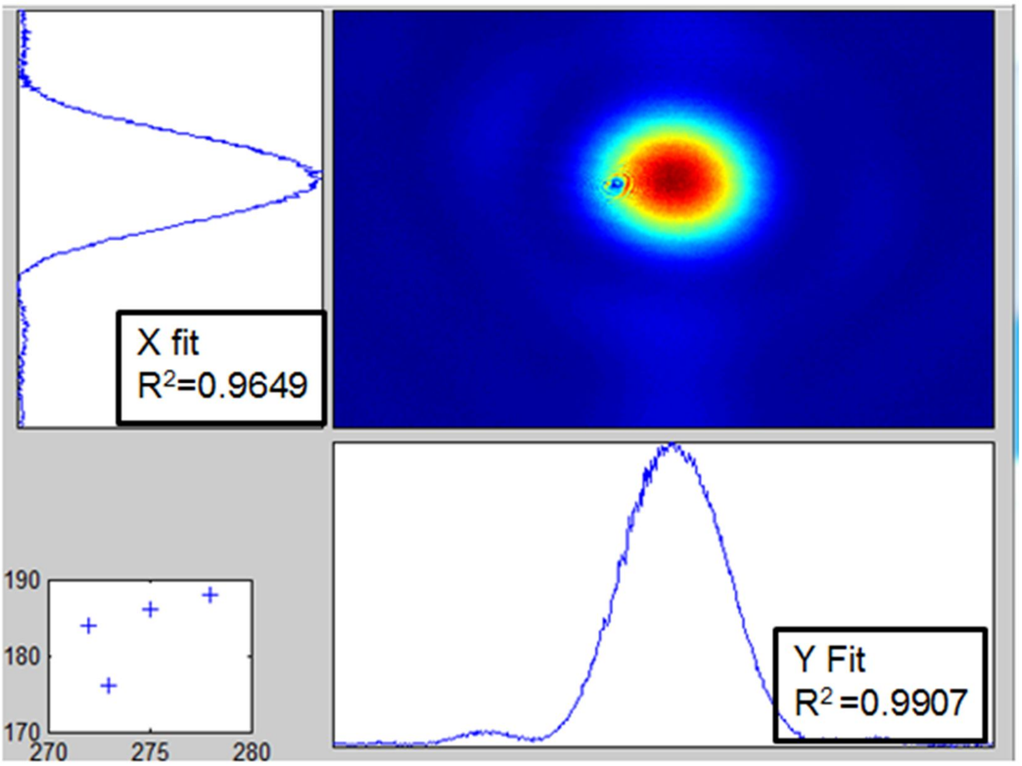

**Figure 5 X and Y fits, Center of Gravity position history**

On figure 5 we can see the false color image the X and Y direction best plots and at the bottom left corner the position of the center of gravity of the laser beam. Also the goodness of fit values are provided for both fits.

## 3.1 Conclusion and outlook

The aim of this work was the development of a code which allows the stabilization of a laser beam. To reach this goal the problem was subdivided into two parts: writing a Matlab code (1) to detect the center of gravity position of a laser beam and (2) for monitoring the beam position using the center of gravity position as a reference and automatically tracking the laser beam to a defined position using piezoelectro-mirrors.

The first part was successfully completed. Also a first step to complete the second part has been done: a MATLAB function has been introduced, which allows a defined movement of the piezo driven mirrors.

The next step must be to connect these two parts together in order to have a script on MATLAB for laser beam stabilization.

## References

[1]The monitoring of a laser beam, Ingemar Eriksson