

# Autofocus System Implementation for Optical Inspection of Superconducting Cavities

Raquel Gómez Ambrosio  
UAM – SSP 2010  
Supervisor: Sebastian Aderhold

### 1)Abstract:

The aim of this project is to find an optimal autofocusing procedure, implemented in the Kytoto camera system which is responsible for the study of the inner surface of niobium resonant cavities.

The optical inspection of the cavities is of great importance, in order to localize and study the sources of thermal breakdown (quench) that limit the maximal achievable accelerating gradient.

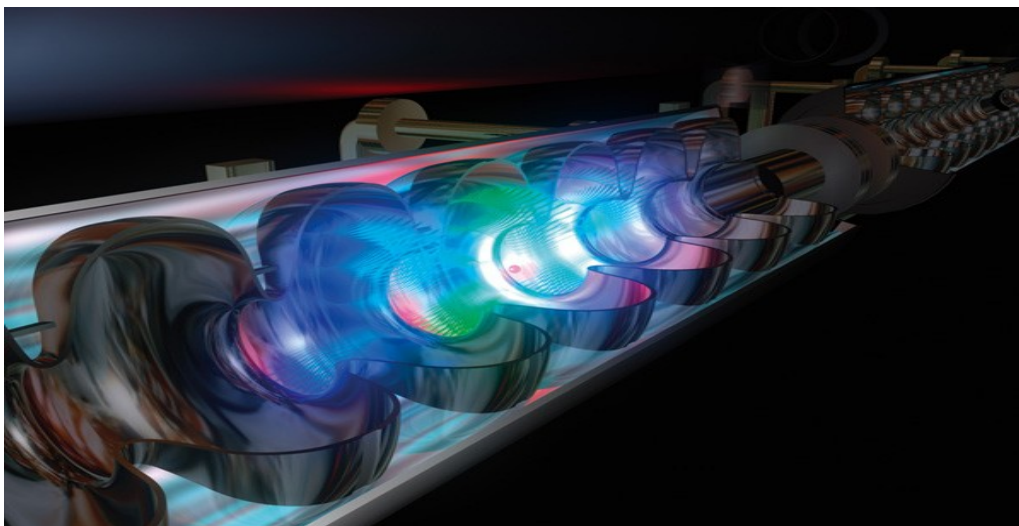
In the end, two MATLAB programmes were developed: The first one, capable to find the sharpest image from a bunch of them. The other one, simulating the real situation in which the programme runs in parallel to the data acquisition, deciding when and where should each picture be taken.

### 2)Introduction:

The FLA-ILC group at DESY works in the development of the SRF (Superconducting-Radio Frequency) Cavities. The aim is to develop a number of tools to automatise the localization and treatment of the defects that limit the accelerating gradient achievable in them, and to understand the mechanisms which lead to quenches.

RF cavities are one of the most important components in an accelerator. They are responsible for the acceleration, and they are most important in linear accelerators. There, the maximum accelerating gradient is not limited by the synchrotron radiation but by the surface quality. The amount of them is also much bigger a linear accelerator than in a circular one:

- LHC: **Eight cavities per beam**, each delivering 2MV (an accelerating field of 5 MV/m) at **400 MHz**. The cavities operate at 4.5K
- HERA-e: 16 cavities, operating at 500MHz
- XFEL: Eight hundred cavities, at 1.3GHz delivering a max. of 30 MV/m, at 2K
- ILC: 18 thousand cavities, yielding an operational field gradient of 31.5 MV/m



*Figure 1: Simulation of the EM filed inside a RF cavity*

In FLA the work is oriented to the superconducting niobium (1 metre distance, 9 cell) cavities. It is known that the maximum accelerating gradient achievable in them is limited by two phenomena:

- Field emission: Tunneling effect of the electron from the metal surface into the vacuum.
- Quench: Transition from superconducting to normal conducting state, due to surface defects.

The main need is to find the precise location of the defect that produces the instabilities, and if it were possible to find its origin, to avoid it in future manufacture. This defects are of the order of  $\mu m$  :

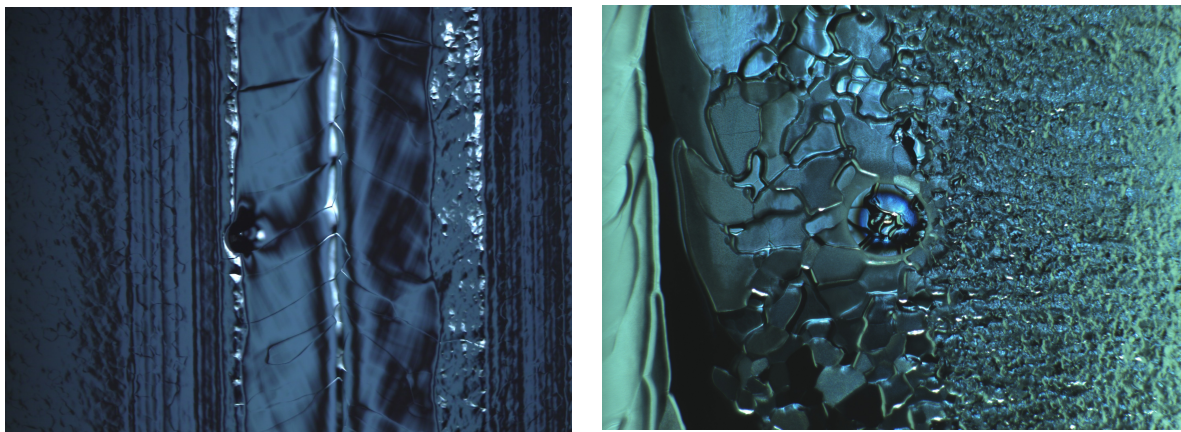


Figure 2: Surface defects in niobium

The usual procedure to detect the source of this problems is to do a temperature mapping of the cavity, which shows where the thermal instability is produced exactly, but does not allow us to see the defect, because it operates in the outside of the cavity. This procedure is actually being substituted by the “second sound” procedure, that consists in a thermal wave created through a heat source in superfluid Helium (the wave is triangulated and the heating spot detected).

After the defects are spatially located, we proceed to an optical inspection of the inner surface that will bring much more information about them. The system used for it is the Kyoto camera set-up (developed by Kyoto University):

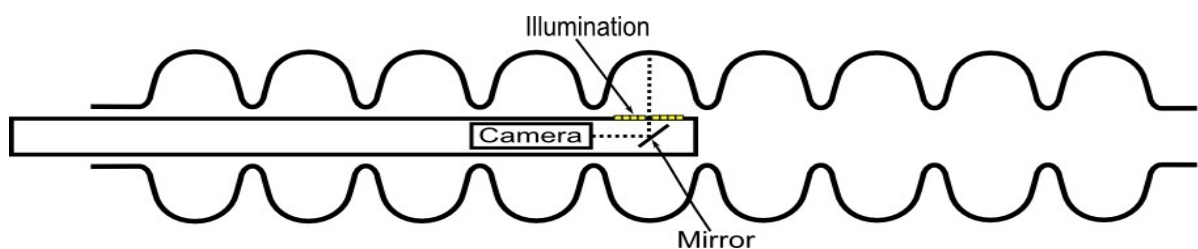


Figure 3: Kyoto Camera set up

The focusing of the camera is controlled by varying the distance between the lens and the mirror. The aim of this project is to find an appropriate autofocus function, with which one wouldn't need to operate the camera by hand. Instead, an algorithm would decide which picture is the sharpest, and therefore the camera would be controlled automatically. The main difficulty arises from the complex pillbox shape of each of the cells in the cavity.

### Autofocusing:

Autofocusing is one of the most important topics for the subject of microscopy, being something indispensable for the success of any research that uses microscopes or photo cameras to acquire data.

However, it is still a quite unknown field, it doesn't exist any "general theory" of it and every time that autofocus is needed for an experiment, a new research on the field is done, depending on the experimental conditions. The most important facts to take on account are:

- Type of microscope, or in this case, camera
- Lighting used
- Characteristics of the sample: large or small information content, fluorescent or dark...

When looking for an autofocus procedure, the first step to take is to define what is the difference between a sharp and a blurred image. This is done usually by defining a "focus function". This means, a function of the intensity of each pixel, that is maximum for a sharp image and decreases as we move away from it.

The basic assumption that is made when treating to describe the degree of focusing of an image is that sharp images have more high frequency content than unfocused ones. This can be seen easily in an intuitive way:

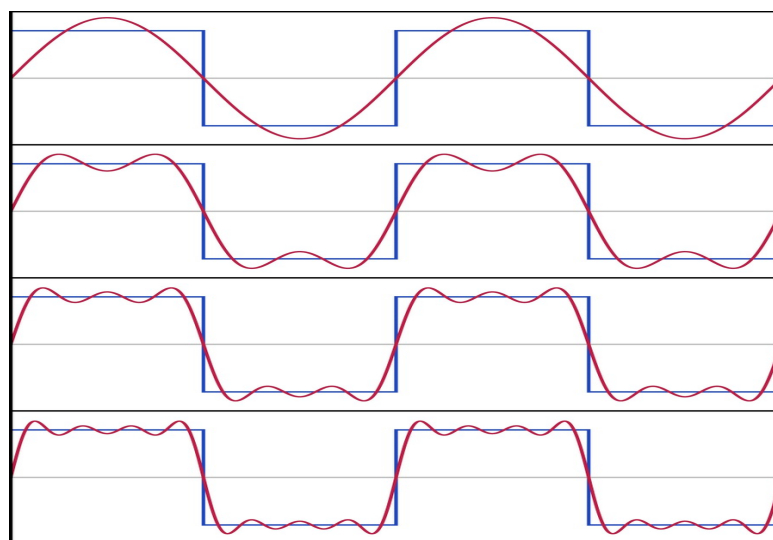


Figure 4: Fourier series representation of a square function. Taking 1,2,3 or 4 harmonics

If we try to represent the original image in the frequency space, through a Fourier series, the result will be more accurate if we have more and more higher order (higher frequency) terms.

In a more formal way, we have from Fourier optics the PSF (Point Spread Function) which is characteristic of each optical system, and it represents its deviation from an ideal one. The observed image is formed by calculating the convolution of this PSF with the original object. The consequence of this is a decrease in the high frequency content of the resulting image.

The different focusing algorithms are based on the grayscale intensity of each pixel of the image, seen as a matrix. They can be classified into five groups, depending on the mathematical model they are using:

- 1) **Derivative-based algorithms:** They take on account the intensity differences between neighbouring pixels. They are the most intuitive ones: in a sharp image the changes in brightness are better delimited, which means that this gradient should be maximal. In particular this two were evaluated

Absolut tresholed gradient:

$$F_{th} = \sum_N \sum_M |g(i, j+1) - g(i, j)| \quad \text{while} \quad |g(i, j+1) - g(i, j)| \geq v$$

Where  $v$  is an ad hoc introduced treshold, in particular  $v=0$  was chosen, thinking forward on the execution time of the code.

Squared gradient: The larger the gradient, the bigger the final effect on  $F$ .

$$F_{sq} = \sum_N \sum_M |g(i, j+1) - g(i, j)|^2 \quad \text{while} \quad |g(i, j+1) - g(i, j)| \geq v$$

Each of this algorithms were afterwards rewritten to take on account the differences both in vertical and horizontal directions. But this only resulted on a larger execution time. It's obvious from the point of view that we expect some symmetry from the original picture. (In this particular case we expect the image to be sharp or blurred all along the vertical axis, in which the distance to the camera is constant. That's why we apply the algorithm in the horizontal dimension).

- 2) **Depth of peaks and valleys:** As well as the preceding ones, they centre on the brightness or darkness of the images. Summing up all the intensities above or below some treshold. This algorithms were tried but didn't work for our sets of pictures. Probably due to an error in the original formulas.

$$F_{above\,tres.} = \sum_M \sum_N g(i, j) \quad \text{while} \quad g(i, j) \geq v \text{ (treshold)}$$

$$F_{below\,tres} = \sum_M \sum_N S[g(i, j)] \quad \begin{array}{ll} S=0 & \text{if } g(i, j) \geq v \text{ (treshold)} \\ S=1 & \text{if } g(i, j) < v \end{array}$$

- 3) **Statistical algorithms:** They are very similar to the first group, but instead on focusing in the difference between neighbour pixels, they compare to the average intensity of the image (they measure the contrast). Belonging to this group, the algorithm “normalized variance” was tried, based on the good results found in some microscopy papers.

$$F_{norm.var} = \left( \frac{1}{M \cdot N \cdot \bar{g}} \right) \cdot \sum_M \sum_N |g(i, j) - \bar{g}|$$

- 4) **Histogram:** These algorithms assume that focused images have more grey levels than the blurred ones. If we understand that more information content means a better defined image, this means having more entropy in the image, so a more 'spread' histogram. None of this algorithms were tried because they seem to enlarge the execution time.
- 5) **Correlation Measurements:** Are based on the autocorrelation function (cross correlation of the image with itself) and on the standard deviation function. In particular, this one was tried: (Vollath, 1987)

$$F_{autocorr} = \sum_M \sum_N g(i, j) \cdot g(i+1, j) - \sum_M \sum_N g(i, j) \cdot g(i+2, j)$$

- 6) **Fourier Transform-based:** This algorithms are very accurate in the theory but the result useless in the practical sense, due to the large computation time needed to calculate the fourier transform of the image.

When choosing the algorithms that were going to be tried, the first criterion to be applied was the simplicity of it, this way we can ensure a relatively small computation time and also take on account the possibility of using more than one program simultaneously.

All the focus algorithms implemented were written and run in MATLAB, with the aid of the “Image Processing Toolbox”.

Every focus algorithm has been written in two different ways, the first one using loops and the second one using matrix algebra, the second one being always much faster than the first one. The object of programming them twice is that the loop algebra is more transparent, in the sense that the original algorithms are presented like this (in summations) and applying both codes and checking the agreement in the results ensures us the validity of them.

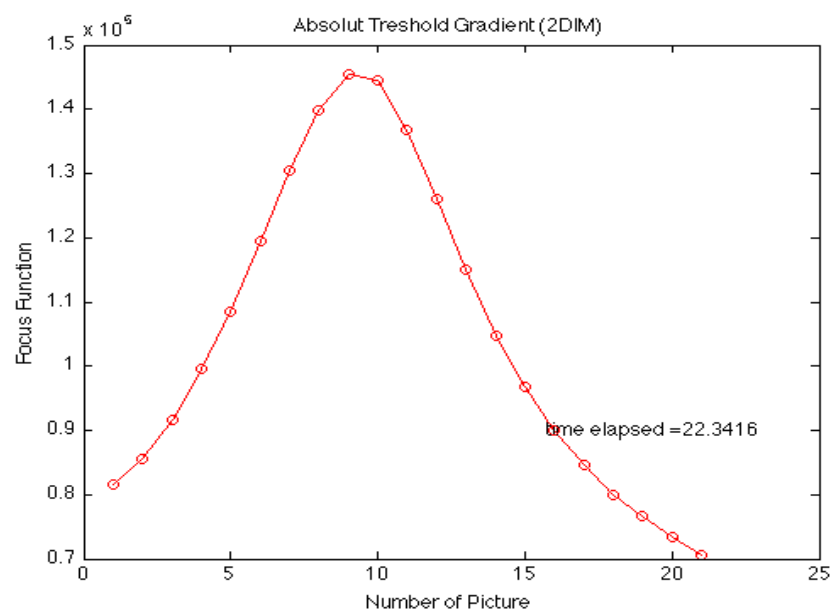
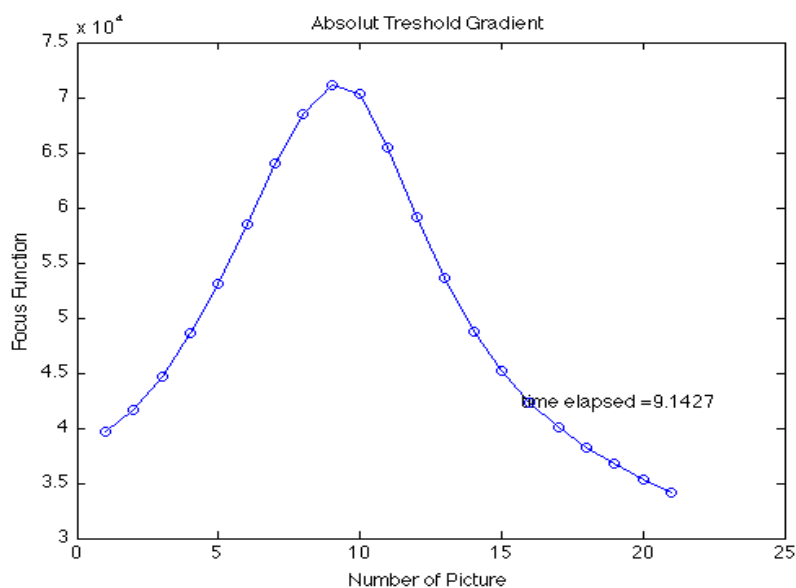
The most important facts that have been considered when choosing the best algorithm are:

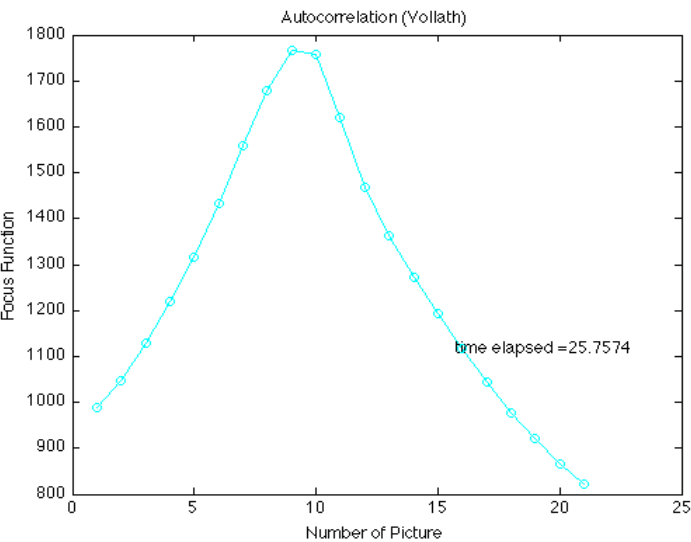
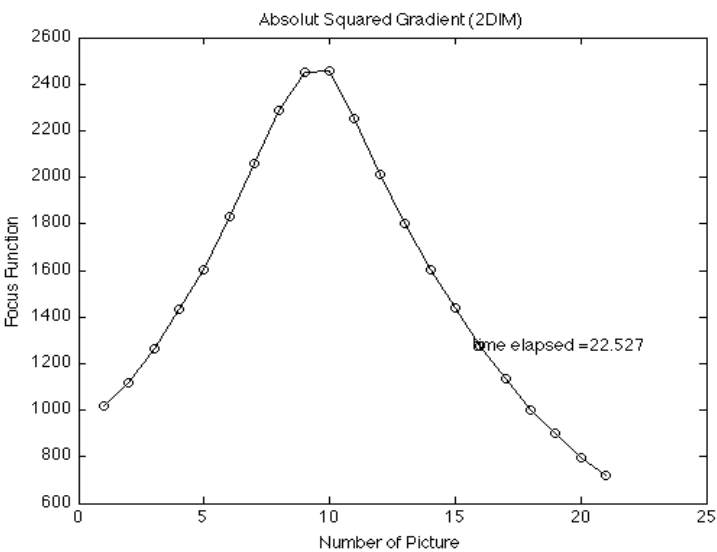
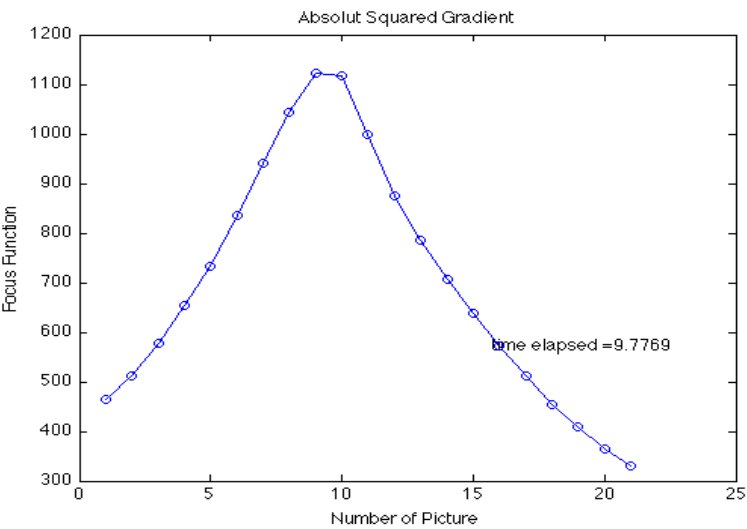
- The execution time of the algorithm for each picture.
- The number of pictures needed to find the optimal one; obviously, the less the images needed, the faster the procedure will be. In our experimental conditions, in which we always had the same amount of pictures, this property was represented by the narrowness of the maximum peak.

Results:

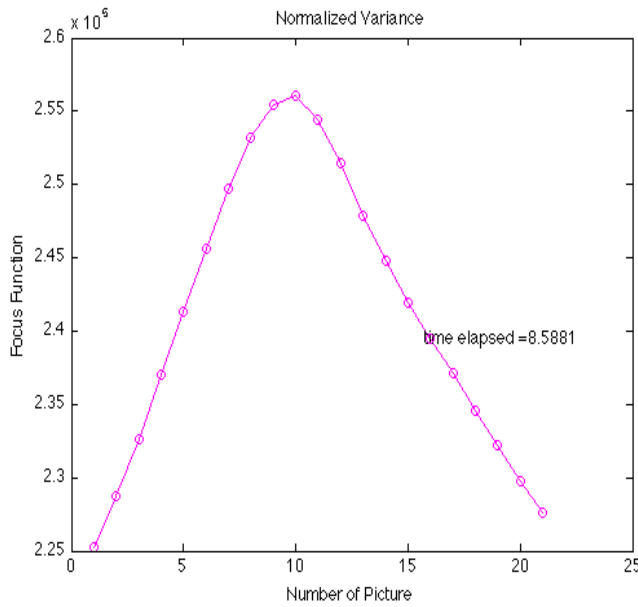
In the end, 14 scripts were evaluated, representing 5 different theoretical focus functions. Each of the algorithms was tested into 5 bunches of pictures, containing between 6 and 21 pictures each. In each of them there was one “focused” image (chosen by the human eye) and the others were taken moving the camera some steps further and backwards from this position (in steps of 0,2mm or 0,3mm). The minimal stepsize achievable with Kyoto camera is 4  $\mu m$  but for our purposes the stepsize of 0,3 mm (75 steps for the step-motor) appeared to be more than enough. Indeed, in one of the cases all the algorithms pointed to a “human error” choosing the best image as the one preceding the image chosen by human eye. Some examples of the pictures used can be seen in the appendix.

The results appeared to be homogenous for all the bunches of pictures. Below we present the results for one of them, plotting “focus” versus number of picture (the picture chosen by human eye is the one in the middle, in this case, number 11 out of 21)









As it can be seen, all the functions are reasonable, and they all guess the correct sharp image. One important fact is that they don't have any secondary (false) maxima, which is a very important property, because when searching for the sharp function with the Kyoto camera, we don't have any clue of where the expected focal position shall be.

The most narrow peaks appear for the autocorrelation function and the best execution time is achieved for "normalized variance". However, the optimal commitment between narrowness and time appeared for of the most simple ones: "Squared Gradient" (without any threshold).

$$F_{sq} = \sum_N \sum_M |g(i, j+1) - g(i, j)|^2$$

Nevertheless, all algorithms appeared to work properly, so they can also be tested in future calculations.

### Sampling:

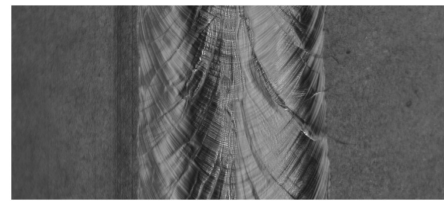
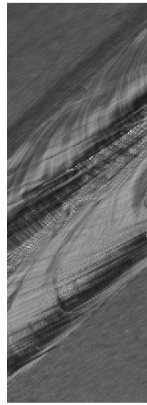
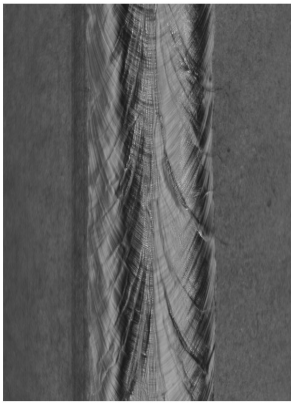
The next step forward in the search for the minimal execution time is the sampling of the image, this means reducing it's size in a small time and consequently reducing the number of pixels to be evaluated but without loss of information.

Two kinds of sampling where proposed, the first one, more intuitive consists in taking only one of each 4 pixels of the image, in both vertical and horizontal directions. This means a reduction of 16 times in the number of pixels, without loss of information (at least for the human eye), as it can be checked comparing both images.

The second kind of sampling is also interesting. It takes on account the symmetry of

the images, due to the fact that all of them represent the same pattern: one homogeneous background with a vertical line in the middle representing the binding between cells (welding seam). Recalling the shape of the cavity we notice that the distance relative to the camera is constant in the vertical axis, while it decreases very fast from this central line to the borders, so the most “fair” sampling we could do to the original image without losing information of focused and unfocused zones would consist:

a) Taking a diagonal line from one corner to the opposite, and choosing the width of it, in agreement with our preferences, we can reduce by 4 the size of the original matrix, and if we combine it with the method described before we can reduce it in a factor  $16 \times 4 = 64$



*Fig. 5: Original Image*

*Fig. 6: "Diagonal"*

*Figure 7: Horizontal Sampling*

b) Taking a horizontal stripe in the centre of the picture. Due to the already mentioned “vertical symmetry” we are supposed not to lose any information with it. As well as in the preceding case, we can choose the reduction factor through the wideness of the stripe. We can combine this method once again with the one mentioned above.

In the end, the first kind of sampling (keeping only one of every 4 pixels) appeared to be very successful.

The second one, diagonal, appeared to be too slow in comparison with the computation time of the focus algorithm (combined with the first kind of sampling, it took  $\sim 0.8$  sec per picture, and on its own it took  $> 10$  sec to sample a  $2616 \times 3488$  pixel image).

The third sampling possibility, also combined with the first one, appeared to be optimal. This programme needed  $\sim 0.35$  sec to sample a  $2616 \times 3488$  pixel image, that in the end resulted in a reduction of 40% in the average computation time for the final programme. (From  $\sim 10$  sec for a bunch of 21 pictures to  $\sim 6$  sec).

Once the optimal autofocusing algorithms has been found, the next step involves “teaching” the camera how to distinguish which of two pictures is better, and move forward or backwards in the direction of it.

To simulate this situation, a code was written in which the camera is “blind” to the group of pictures that we have. It starts evaluating one random picture, without any information about the position from which it was taken.

This programme needed an average of 6 seconds to find the best picture of a bunch of 21. This 21 pictures were taken in a range of (-500,+500) steps relative to the sharp image. That is approximately  $\pm 2$  mm in the position of the camera with the step-motor.

This means that if we are able to position the camera in a range of 2 mm from the focal position, and assuming that the time needed to take the picture is smaller than the computation time, we could obtain a sharp image in a lapse of 6 seconds time.

#### Further Steps:

The next step to take for the implementation of the autofocus system is to integrate this code with the camera, so as we wouldn't need to have a bunch of pictures beforehand, but take them simultaneously with the computation.

This can be done using the data acquisition system "Labview" or the "Instrument control" and "Image Acquisition" toolboxes from MATLAB.

#### -References:

- "Evaluation of autofocus functions in molecular cytogenetic analysis"

A.Santos, C.Ortiz de Solórzano, J.M. Peña, N.Malpica.  
Journals of Microscopy 1997

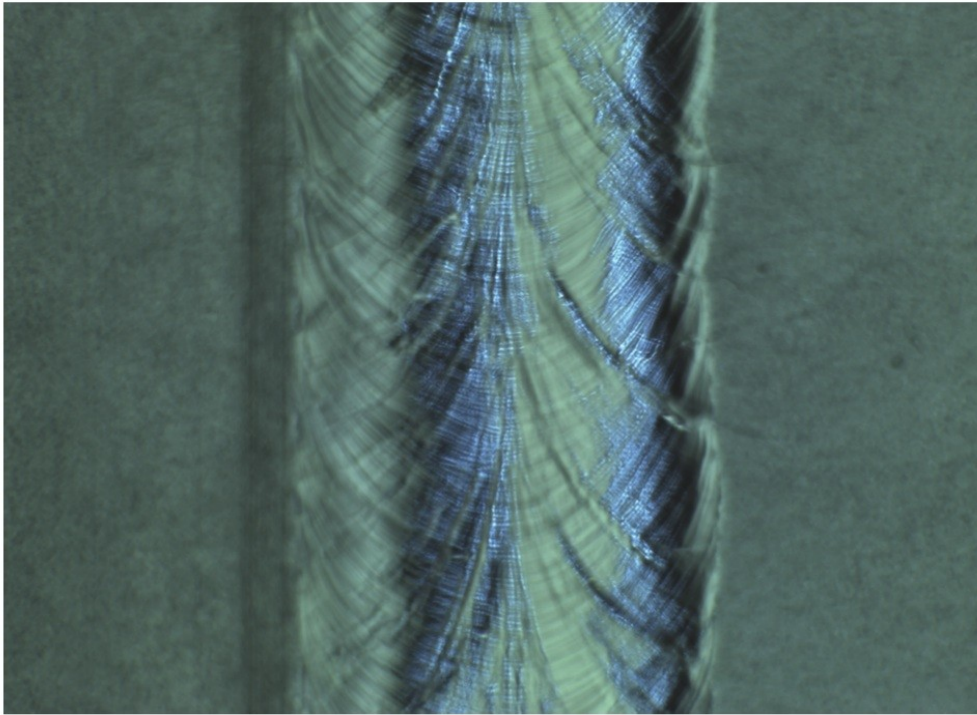
- "Autofocusing in Computer Microscopy: Selecting the Optimal Focus Algorithm"

Yu Sun, Stefan Duthaler, Bradley J. Nelson  
Microscopy Research and Technique 2004

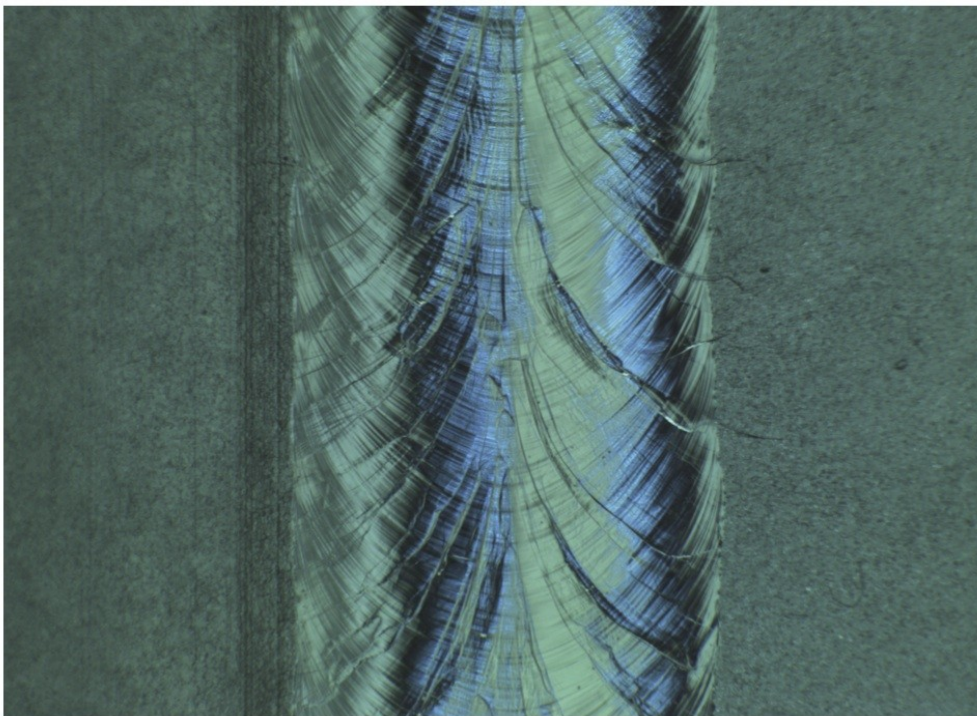
- "HIGH FIELD SC-CAVITIES" D. Proch, DESY

- "Progress on Diagnostic Tools for Superconducting High-Gradient Cavities"

F. Schlander, S. Aderhold, E. Elsen, D. Reschke, DESY



*Figure 1: unfocused image*



*Figure 2: sharp image*