



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

University of West Bohemia in Pilsen
Faculty of Applied Sciences
Department of Computer Science and Engineering
Univerzitni 8
306 14 Pilsen
Czech Republic

Pattern Recognition, Classification and Simulation of Laser Beam Impact

The State of the Art and Concept of Doctoral Thesis

Ing. Jana Hájková

Technical Report No. DCSE/TR-2008-03
April, 2008

Distribution: public

Pattern Recognition, Classification and Simulation of Laser Beam Impact

Ing. Jana Hájková

Abstract

This work deals with the problem of laser simulation. At the beginning it gives a broader overview of the project of laser simulation which is processed at the University of West Bohemia. The simulation is described in several fundamental steps to gain basic concept of the process. Several techniques of data sets obtaining, their processing and usage for the simulation and also the importance of simulation system verification is highlighted to understand the whole approach well. Several methods for automatic pulse detection are described in detail. Pulse detection is the main part of pulse extraction, which is one of the most important data processing steps. The main idea of each described method is explained and their problems and possible ways of their elimination are discussed. All methods are tested for several selected samples which were chosen because of some typical feature. As another important part of the work methods for system verification are outlined and results are discussed. At the end of the paper future plans for the project are introduced.



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

University of West Bohemia in Pilsen
Faculty of Applied Sciences
Department of Computer Science and Engineering
Univerzitni 8
306 14 Pilsen
Czech Republic

Pattern Recognition, Classification and Simulation of Laser Beam Impact

The State of the Art and Concept of Doctoral Thesis

Ing. Jana Hájková

Pattern Recognition, Classification and Simulation of Laser Beam Impact

Ing. Jana Hájková

Abstract

This work deals with the problem of laser simulation. At the beginning it gives a broader overview of the project of laser simulation which is processed at the University of West Bohemia. The simulation is described in several fundamental steps to gain basic concept of the process. Several techniques of data sets obtaining, their processing and usage for the simulation and also the importance of simulation system verification is highlighted to understand the whole approach well. Several methods for automatic pulse detection are described in detail. Pulse detection is the main part of pulse extraction, which is one of the most important data processing steps. The main idea of each described method is explained and their problems and possible ways of their elimination are discussed. All methods are tested for several selected samples which were chosen because of some typical feature. As another important part of the work methods for system verification are outlined and results are discussed. At the end of the paper future plans for the project are introduced.

Table of Content

Table of Content	1
1 Project Description	2
2 Data Description.....	4
3 Pulse Extraction	9
3.1 Pulse Representation	9
3.2 Methods of Pulse Detection	10
3.2.1 Method Starting from Global Extremes	11
3.2.2 Centre of Mass Method	12
3.2.3 Spiral Method	13
3.2.4 Image Processing Methods Application.....	15
3.3 Actual Results of Pulse Detection.....	17
3.4 Pulse Completion	21
4 Simulation	23
4.1 Reasons for simulation.....	23
4.2 Technique of Simulation.....	23
4.3 Analytical methods	23
4.4 Application methods	24
5 System Verification.....	27
5.1 Samples Comparison	27
5.1.1 MSE.....	29
5.1.2 Difference Image.....	30
5.1.3 Logarithm Operator Adaptation of Difference Image.....	30
5.1.4 Principal Components Analysis	32
5.2 Samples Overlapping	34
6 Results	36
6.1 Sample Viewing.....	36
6.2 Simulation of Burning.....	41
6.3 Result Comparison.....	43
7 Future Plans.....	46
7.1 Simulation Improvement	46
7.2 Automation and Program Self-activity	46
7.3 The Ways of Pulse Description.....	47
7.4 Others	48
8 Conclusion.....	49
Literature	50

1 Project Description

The whole Ph.D. work is prepared as a part of the multi-departmental project. The project started at the beginning of 2007 in a cooperation of three departments of University of West Bohemia – the Department of Computer Science and Engineering, the Department of Physics and the Department of Cybernetics. Except these three university departments, also a hi-tech company – Lintech [wLin], participates in this project and supports it. The global overview of the project is described in [Háj&Her].

The aim of the project is to develop real laser equipment (HW device) for burning any image into any kind of material. This device must have several SW parts, which control the laser and simulate its function. Single parts of the whole project are depicted in Fig. 1. components processed by responsible workgroups are also marked. Partial rectangle border means cooperation of the departments.

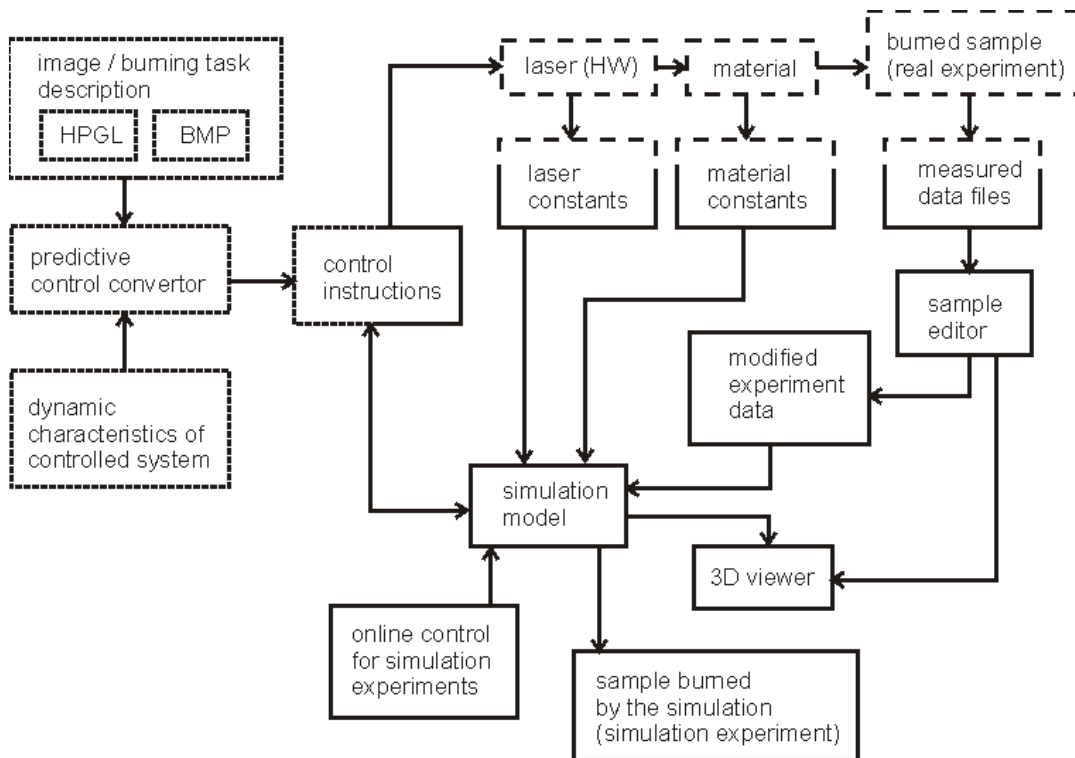


Fig. 1: Detail view of single modules of the project and their solvers: the Department of Computer Science and Engineering (solid part), the Department of Physics (dashed part) and the Department of Cybernetics (dotted part).

The laser itself services the Department of Physics (the dashed part). Experts from this group have big experience with lasers and they need the laser system for really sophisticated physical experiments. The second group is from the Department of

Cybernetics (the dotted part), who is responsible for direct controlling of the laser device. Of course, somebody could think that such devices already exist for commercial use. That is true, but they do not enable affecting all burning parameters. Moreover, they do not enable to use simulation for various kind of optimization. This is task for our group from the Department of Computer Science and Engineering (the solid part). We were asked to provide IT support and to add software which will be able to simulate burning process and to explore results in detail. The last partner of the project is the commercial firm Lintech who will need the system for common technical task. This is the second usage of the system.

After approximately one year of the existence of the project, we started also to cooperate with the next partner from the University of West Bohemia. Our team has spread for experts from the Department of Mathematics. The aim of the cooperation between the Department of Mathematics and the Department of Computer Science and Engineering is to find optimal algorithms for processed samples parts recognition and their comparison. These methods will be used during measured data modification (it means in the part of sample editor) and in the phase of system verification.

The whole system enables to burn predefined patterns into the chosen material. The basic function of system modules shown in Fig. 1 and the sequences of the device operations are described in this chapter.

The laser device enables burning raster and vector graphics, text or any other described experiment into any kind of material. Anything that should be burned has to be described in HPGL (Hewlett-Packard Graphics Language) language or in the form of BMP image. This experiment description goes to predictive control convertor. The convertor uses knowledge of the dynamic characteristics of controlled system and creates from the description the set of control instruction for the laser. Instructions contain information where to burn or which intensity to use. These instructions are sent to the laser to control the burning process into real material – the real experiment.

To provide simulation of the burning process, characteristics of the laser and material have to be known. They are described in the form of laser and material constants. Data sets from the real samples have to be measured and the result saved into the data file. The process of data obtaining is introduced in the Chapter 2. File with the sample description is loaded into the sample editor, where it is processed (e.g. overflowing parts of the sample are clipped). When all files with necessary data are loaded into the simulation model, the simulation system has to be verified. After the system verification the simulation of any experiment can be operated. Experiments can be burned individually or they can be processed in batches. The simulation of the burning process can run offline or it can be online controlled to get the best results. During the offline simulation all results are saved and logged so the best results can be chosen when the simulation finishes. During the online simulation the burning process can be optimized concurrently. Criteria for the optimization can be selected as the speed of burning or quality of result. The simulator also enables to view all samples (the real and the simulated ones) in 2D and 3D viewer. It is also possible to optimize control instruction for the laser during the simulation (e.g. intensity of the laser or time periods used for burning). Finally at the end of simulation process the burned sample (the simulation experiment) is obtained.

During the system verification both real experiment and simulation experiment are compared. Because the simulator should produce as realistic results as possible, it has to be verified.

2 Data Description

At first, the way of data obtaining should be described. To get data for the simulation input, real samples have to be burned by existing laser equipment into the real material and measured. To get enough information for setting the simulation system, plenty of samples have to be measured.

The set of samples is called experiment. One experiment contains data for one concrete laser device and a given material. It means that for each combination of laser device and material which are used for simulation a special experiment has to be measured. Experiment consists of samples with pulses burned by the laser into one point into the material. The count of pulses goes in sequence, e.g. from 1 to 100 (in Fig. 2 each pulse from the sequence is burned on a single line). Because after reburning the same pulse several times with keeping the same conditions the results differ a little, each pulse count is repeated several times, to get an average result. These samples are places side by side in one line (one line in Fig. 2a corresponds to the similar pulses). Results from one real experiment burned by the laser BLS-100 (Nd:YAG solid-material, lamp-pumped laser with wavelength 1064nm) [wSHT] into a steel are visible in the 3D views in Fig. 3. Parameter of burning are: laser power 100W, current 28A, width of the ray 0.01mm, diaphragm 1.8. This type of input data is used for pulse extraction and system verification.

For the verification also another data sets which represent other way of pulse application are planned to be used. Pulses are not burned into one point, but the laser moves during the burning and burns pulses with the determined frequency. Also for these samples experiments are prepared. Such experiment can be seen in Fig. 2b. Parameters of the laser are the same, the speed or laser movement is 100 mm/s.

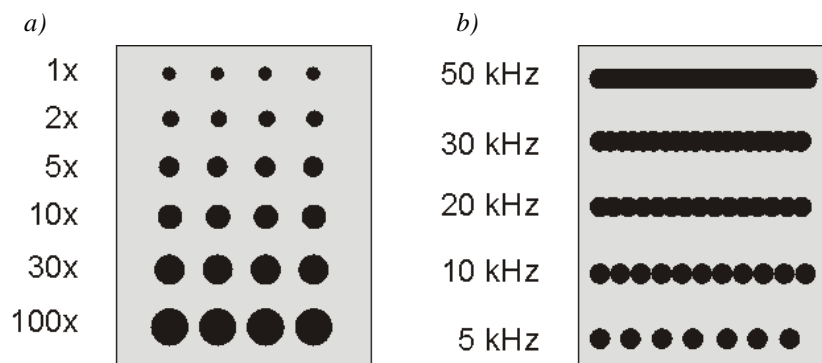


Fig. 2: Experiment description for a) pulses burned into one point into the material – lines correspond to samples with the same count of pulses, repeated samples are placed side by side. Sequences of the pulse counts go from 1 to 100 pulses burned into one place; b) pulses are burned during the laser motion with the frequency from 5 to 50 kHz.

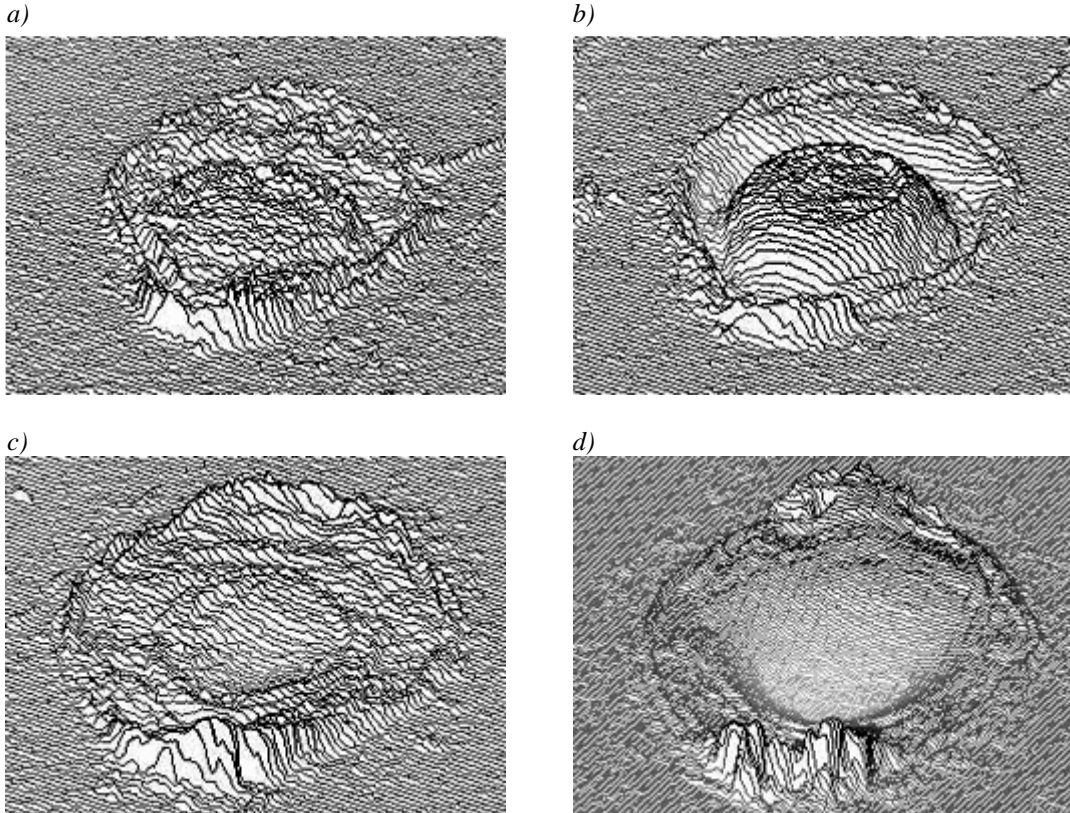


Fig. 3: 3D views of samples burned with lamp-pumped laser BLS-100 into one point. steal a) 1x, b) 2x, c) 5x, d) 10x into one point.

After burning of the real samples, they have to be measured. For this purpose the confocal microscope Olympus LEXT OLS3100 [wOlym] is used. During the measurement just a part of the material with burned pulses is focused, scale and zoom are chosen and the material sample is scanned and saved. As the output from the confocal microscope text format CSV (comma separated values) file is used. If the smaller zoom is used, the measured sample has worse resolution, it is not so precise, but it needs less time for processing and less memory for saving the sample. With better resolution the sample description is more exact but it needs more time for processing. The selection of optimal scale is another question to be solved.

To be more concrete, it would be suitable to describe data in more detailed way. The CSV file contains several head lines with descriptions of type of measured data, units for all three basic axes of the coordinate system and the step size that was used during measurement.

The head lines are followed by the matrix of floats which represents the sample surface. Values express heights in corner points of the uniform rectangular grid. This grid represents a height map which describes the surface of the sample (see Fig. 4).

Now the expressions of sample and pulse should be distinguished and explained. Data loaded into the simulation system represents the sample. It means the surface of the part of material measured by the confocal microscope. The pulse symbolizes the part of the sample that was burned by the laser (in Fig. 5 it is highlighted by the dashed rectangle). According to the scale selected during the measuring of the real material, sizes of similar pulses can differ. That is why the system computes all distances in real units.

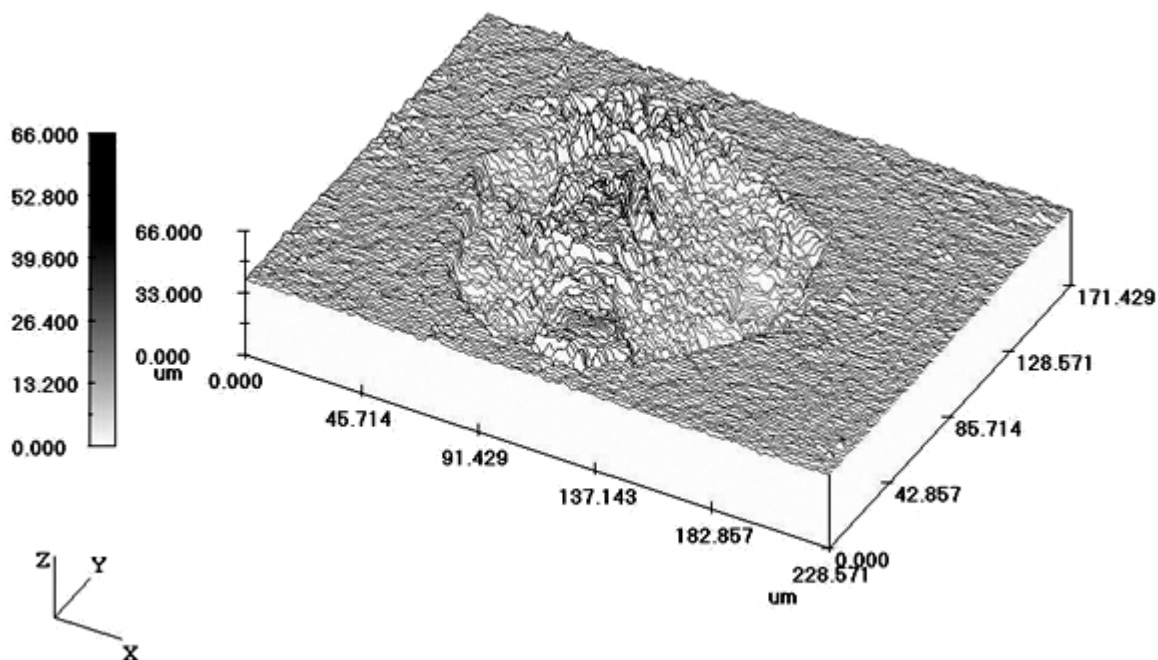


Fig. 4: Height map of the measured sample. The sample of 100 pulses burned into one point into cermet.

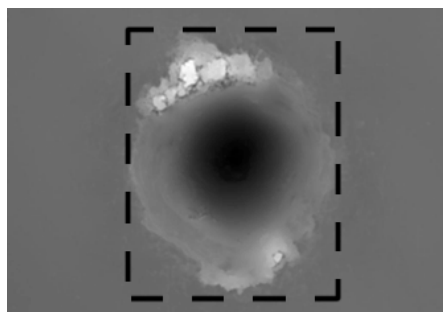


Fig. 5: Definition of expressions sample and pulse – the area of pulse is highlighted in the sample by a dashed rectangle.

The majority of samples which we use have the same size as the sample in Fig. 4. The burned pulse fills preponderance of the sample surface; the real dimension of the sample is $256 \times 192 \mu\text{m}$. The grid of height map is really fine; most common grid step in used data is $0.25 \mu\text{m}$. It means that the surface of such sample is described by 1024×768 values. We dispose with samples burned into two materials: steel and cermet (a composite material composed of ceramic and metallic materials) [wikiCer].

In addition to data from samples with pulses burned into one point which are used for pulse extracting and system verification, we work also with another type or real data. These are samples with pulses burned into the material along any trajectory with defined laser frequency. For each laser and material the sequences of samples with various laser frequencies used for burning are measured to get enough samples for simulation verification. More pieces of information about system verification are described in Chapter 5.

The way of data descriptions obtaining is the same as in the case of single pulses, it means data files are real samples measured by the confocal microscope and saved in CSV files. Both types of input data (both single pulses and trajectories) are for better comparison shown in Fig. 6. Fig. 6a shows the sample with 10 pulses burned into one point, Fig. 6b figures sample burned along the trajectory with frequency 10 kHz. In both cases the same laser and material (steel) were used.

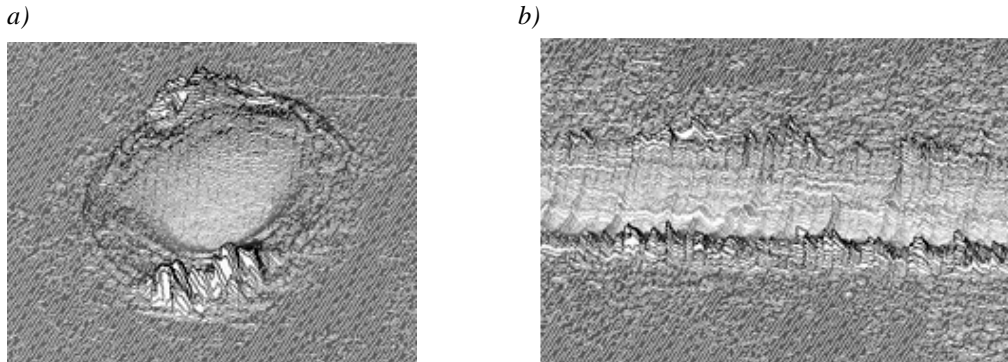


Fig. 6: Both types of input data burned with the same type of laser into steel a) sample with 10 pulses burned into one point, b) sample burned along the trajectory with laser frequency 10 kHz.

Let's explore the laser pulse burned into the material in a detail, at first with a little bit simplification. In an ideal situation the surface of the material would be perfectly smooth without any roughness. When the laser burns one pulse into such material, it will modify the surface. In the point of burning a hollow will be created and around the hollow the melted material will make a bulge (as seen in Fig. 7a). By these conditions if we place two laser pulses at almost one place, we can use a principle of summation. Result is shown in the Fig. 7b where two pulses next to each other are placed.

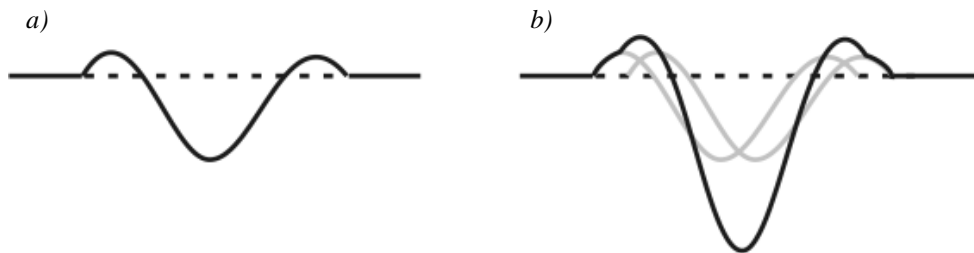


Fig. 7: Simplified result of burning into ideally smooth material a) one pulse b) two pulses next to each other.

But in the real environment we can hardly expect ideal conditions. Because we use real data which are measured with high zoom, roughness of the material plays an important role. Fortunately, the dimension of burned pulse is higher than material surface roughness. The difference depends on the material. In spite of that the surface roughness causes inconsiderable problems during sample processing. Cross-sections of real pulse (one pulse burned into the cermet) can be seen in Fig. 8. In the horizontal (Fig. 8a) and vertical (Fig. 8b) cross-sections of the pulse the differences in the pulse shape are visible. The differences can be seen both in the shape of the pulse cross-section and also in the pulse shape from the top view.



Fig. 8: Cross-section of real pulse in a) vertical and b) horizontal direction.

Such pulses cannot be represented by several parameters as it could be in the case of ideal situation. And this is the reason of our decision for using height maps for the representation of samples.

Of course the process of data receiving is money and time consuming, size of files with saved data is also not small, in the case of several tens of measured samples for each material all measured samples of the experiment can reach the size of GB. One of our tasks is to find the optimal experiment which would represent the behaviour of the material during the laser burning. We have to decide, how many burnings have to be done to obtain enough full-range and representative spectrum of pulses. Of course, we try not to do many burnings uselessly. For different materials the experiments can differ. With data from well defined experiment, the simulation model would be able to realize arbitrary kind of simulation experiment.

3 Pulse Extraction

The phase of pulse extraction includes several jobs which have to work in a perfect cooperation. It concerns pulse auto-detection in the sample, its saving in the proper format, possible additional loading, modification and finally creation of the average pulse which can be used for burning simulation itself. The topic of the pulse extraction was used as the main topic in [Háj].

3.1 Pulse Representation

At the beginning of our work it was important to choose any suitable form of pulse representation. At first we planned to represent the pulse by several parameters or by its cross-section. The whole pulse should be gained as a surface of revolution [wikiSoR]. It means that the surface of the sample would be obtained by rotating the curve of cross-section in space about an axis passing through the middle (see Fig. 9a). But after closer research of real samples we have founded that the real pulses for some materials are far too asymmetric to use any parametrization. Also cross-section curves in various directions differs so much, that representation of the pulse in such way would mean unacceptable distortion of data. In Fig. 9b-d real pulse cross-sections for the direction inflected from horizontal for 0° , 45° and 90° are shown (the scale used for all cross-sections is the same).

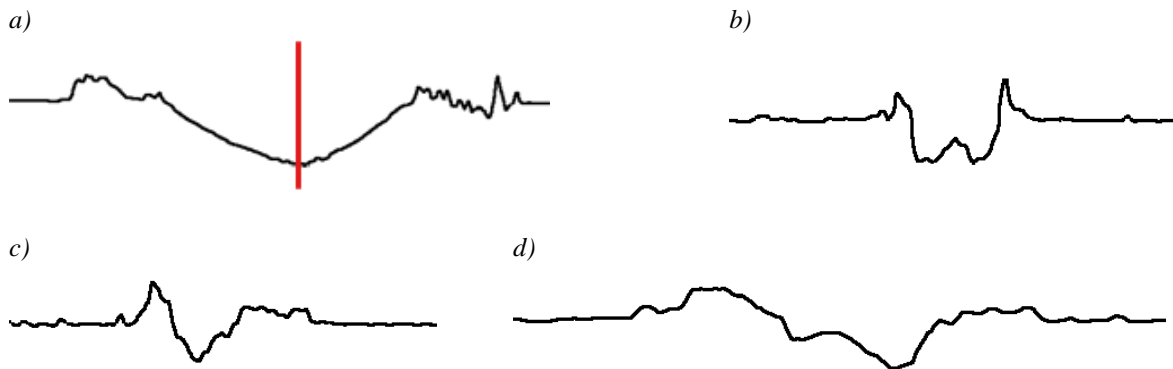


Fig. 9: a) Cross-section curve with axis passing through the middle round which the curve would be rotated to get surface of revolution representing surface of the pulse; b-d) real pulse cross-sections for the direction inflected from horizontal for 0° , 45° and 90° (the scale used for all cross-section is the same).

That is why we have to search for another way of pulse representation. Finally we decided for the same form as it is used for the representation of measured samples – for the height map representation of the whole pulse surface. The question was how to describe the

values of the pulse surface to be used for the burning simulation in the simplest way. From several experiments we decided to represent the level of the material by the value of zero. All points of the sample height map are during pulse extraction recounted and saved as difference of the surface of the material and their original height. It means that the final saved pulse consists of positive and negative real values. The positive values represent material upon the basic material level; on the contrary the negative ones represent material which has vaporized. Such a pulse can be seen in Fig. 10. The white color represents basic material zero level, red parts of the pulse are positive values and blue parts represent negative values. The higher is the distance of the value from zero, the brighter is the color.

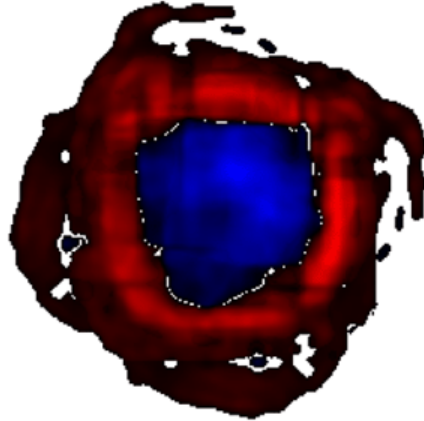


Fig. 10: Pulse representation – the white color represents basic material zero level, red parts of the pulse are positive values and blue parts represent negative values.

The format of height map is exact enough, but its disadvantage embodies in amount of data files which have to be saved for various pulses for different materials. The size of the file is not very high but we still speculate about the redundancy of this format. That is why we would like to try another method of pulse representation (as it is described in the future plans in chapter 7.3). Preliminary investigations show that for pulses with similar pulses the shapes of the holes are basically the same. The greatest divergences appear in the area of the pulse border, where melted material is localized. From this point of view it could be possible to describe the whole pulse by several parameters and the divergences of the pulse border could be generated with partial usage of randomization.

3.2 Methods of Pulse Detection

Pulse detection is used as a part of data preprocessing. The task of detection is to define the area of the material surface which was affected by the laser burning as exactly as possible. Pulse detection can be done of course manually but for the speedup and simplification of the whole preparation process, the system has to prepare maximum of parts self-containedly. But the precision and accuracy has to be preserved as well with the automation process. That is why we have to develop appropriate algorithm for pulse detection.

The main problem is the roughness of the basic material surface. As it can be seen in Fig. 11 for some materials, such as for a steel in Fig. 11a, the surface is quite smooth. Another situation comes in the case of cermet (Fig. 11b), where the protrusions on the material surface are more noticeable. Moreover, for all types of material also local roughnesses which are not specific for the material can appear. Such defects need not to be

visible on the material surface by naked eye, but thanks to the height resolution of real sample scanning, they are included in the description of the sample and are the source of problems during the pulse auto-detection. Local roughness can be seen in Fig. 11a on the upper side of the sample.

The user is able to distinguish roughness or defects of the material during the manual pulse detection, but for the automatic method it is very difficult to differentiate these inaccuracies from the border of the pulse. Chosen methods have to be precise enough, but very precise methods are already slow. Quickly working methods are unfortunately inaccurate. That is why we have to find a compromise and create a new method designed right for this task.

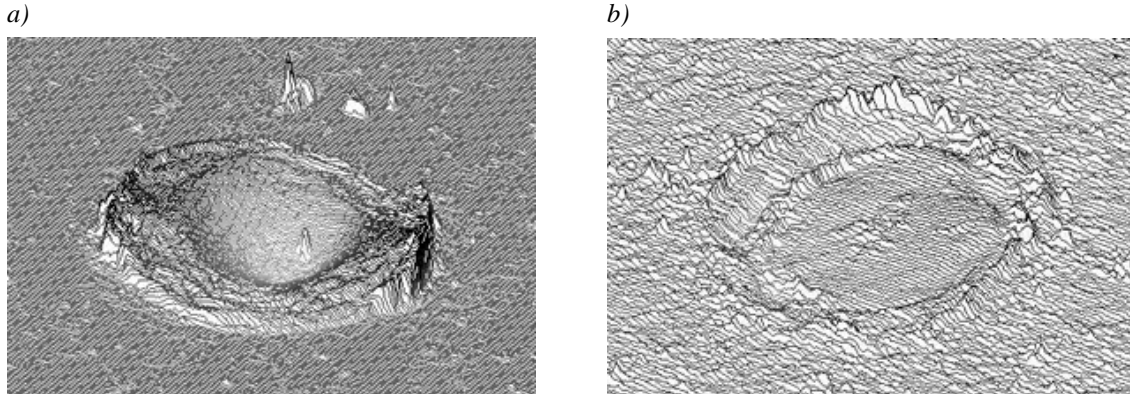


Fig. 11: a) Surface of steel with relatively smooth surface; local defect can be seen on the upper side of the sample. b) Surface of cermet sample with globally higher roughness of the surface.

3.2.1 Method Starting from Global Extremes

The first version of detection algorithm goes from unmodified sample surface. The procedure is in a simplified way shown in Fig. 12. The algorithm starts from points with the minimal and maximal height (represented as white and black cross). These points are supposed to be in the area of the burned pulse that should be detected. From their position columns of height map to the left and to the right side are inspected and the height difference of points in each single column (it means the difference between the maximal and minimal value in the column) is counted. If the value does not exceed given height limit, an inspection in the direction is finished. After cutting of columns on the left and right side of the pulse, the same process of border searching is started for rows. From the position of the minimum and maximum the method goes through rows and looks for the first line above and under the pulse which height difference is lower than the value of given height difference limit. Horizontal borders (showed as horizontal lines) are appointed.

There is a question how to gain the value of difference height limit. This is one of problems of this algorithm. If we do not want to set the constant manually we have to explore the sample automatically, for example during its loading into the system. We can suppose that borders of the sample are not modified by the burning and that is why they represent the original material surface and so the difference constant can be for most samples precounted, for example as the minimal height difference counted from several border columns. But this approach may not work well if the used border of the sample is damaged by any local defect of the material. In such case the difference constants is counted too high.

The worse problem that causes wrong results of this method is brought by local defects of large height or depth. The defect can move the location of global minimum or maximum from the area of the pulse to the area of local defect. This problem can be solved by exploring of the surroundings of the minimum and maximum position. If the volume of the shape around the starting point is too small, it can be labeled as a local defect and another starting position has to be searched for. Another way how to solve wrong starting position determination is to change the algorithm of its searching.

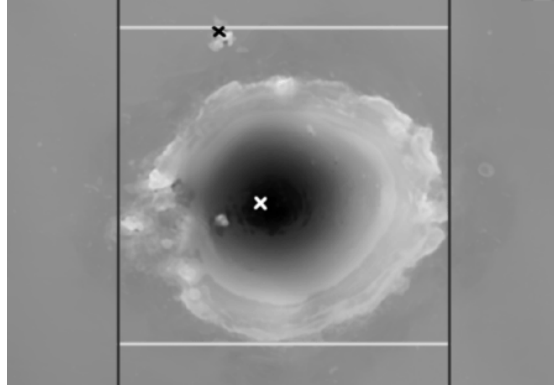


Fig. 12: Borders founded during the algorithm of pulse detection. Crosses represent minimum and maximum of the surface. At first vertical borders are found, and then upper and bottom edge is determined.

3.2.2 Centre of Mass Method

In the previous chapter a relatively simple method for pulse detection was described. Its main problem was finding of starting point. Another way how to get the starting point for automatic pulse detection is to find the position of centre of mass in the sample. The typical procedure of the center of mass computation has to be adapted for the sample representation. Basic expression for calculation of the center of mass x_c of a system of particles is defined as the average of the particle positions x_i , weighted by their masses m_i (4.1) [wikiCoM].

$$x_c = \frac{\sum m_i \cdot x_i}{\sum m_i} \quad (4.1)$$

Let us label the point of the sample as $f_i = f[x, y]$. Because the whole sample is represented by positive values (the values representing the material level in the sample are given by the height of the material into which the sample was burned), we could define $m_i = f_i$. The simplified cross-section curve of the sample can be seen in Fig. 13a. The result is shown in Fig. 14a. If the pulse does not take the majority space of the sample and the pulse is not placed in the centre of the sample, the rest of the surface overbalances the pulse area and the centre of mass is moved from the centre of the pulse partly to the side of plain surface.

That is why it is necessary to shift the whole sample so that the material level is represented as a zero value (see Fig. 13b). By this shifting some parts of the sample are represented by the negative value. Such sample cannot be used for center of mass

computation and so the negative values have to be converted to positive ones. That can be done by using power function with even and positive exponent. To stress values of the pulse from small values in the neighbourhood of the zero level, we decided to use exponent 4 (Fig. 13c). Finally the weights for center of mass computation are defined as in the expression (4.2), where *basicLevel* represents the height of the basic material. The result of such calculation can be seen in Fig. 14b.

$$m_i = (f_i - \text{basicLevel})^4 \quad (4.2)$$

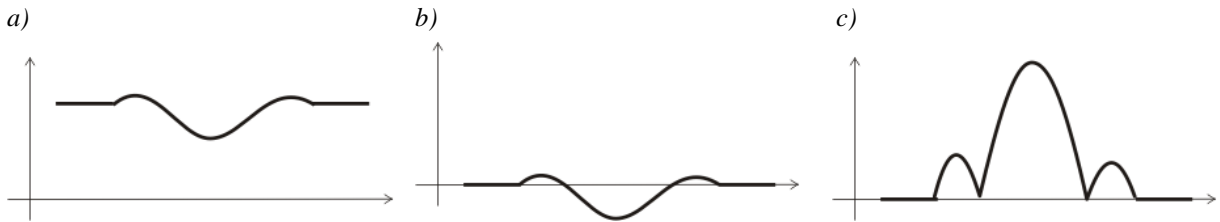


Fig. 13: Simplified cross-section of the sample in all three phases of computation; a) the original sample, b) the situation after shifting material surface to the zero level, c) the sample after application of power function.

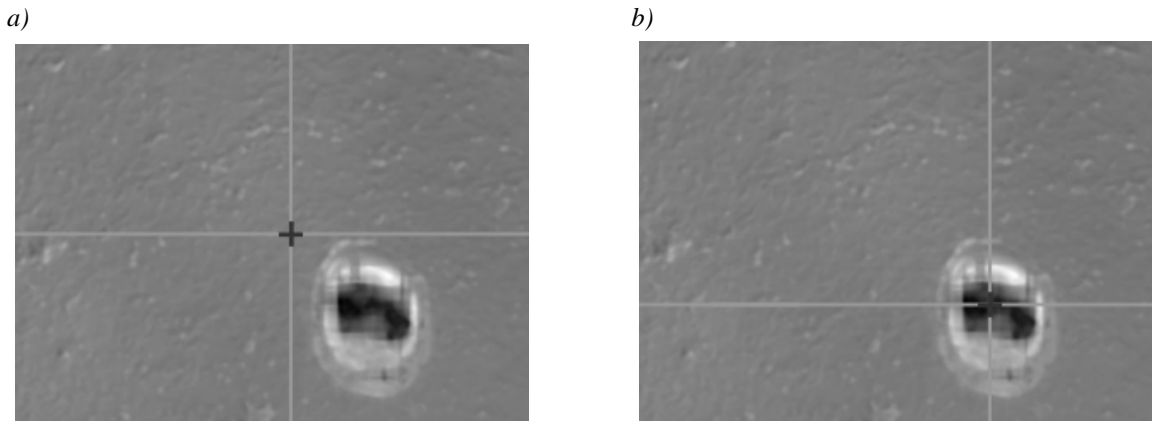


Fig. 14: The result of center of mass counting with the weight a) $m_i = f_i$ and b) $m_i = (f_i - \text{basicLevel})^4$.

By this method we get the starting position which is placed in the area of burned pulse and is affected by local material defect in a minimal way. The other steps of algorithm described in Chapter 3.2.1 can be used in this time or we can choose another technique which is described in Chapter 3.2.3.

3.2.3 Spiral Method

We have tried the centre of mass computation for all samples we dispose with and the location of starting point was always found correctly in the area of burned pulse. So we can try to use another approach for finding borders of the pulse (previously described algorithm of pulse borders detection is sensitive to local defects on the material surface). We can start in the starting point and then inspect the surroundings up to finding the basic level of the material.

In the ideal case, the pulse has circular or ellipsoidal shape and the centre of mass is placed exactly in the centre of the pulse. For such pulse we can find the bounding rectangle simply. If we put through the centre of mass two lines parallel with axis x and y (as can be seen in Fig. 15a), we can make cross-sections of the sample along these lines (the vertical and the horizontal ones). The cross-section curve is optimally in the area of the pulse more diverted than in the part of the unburned material. That is why we can determine two points of the curve, where the pulse finishes and to define borders there (Fig. 15b).

Searching of border values starts at the beginning and at the end of the curve and continues in the direction to the centre of mass. First, we are in the area of unburned material and that is why values of the curve do not differ from the average material height a lot. When the values start to differ more we have found the border of the pulse. To prevent mistakes caused by roughness of the material, the same height limit as in the algorithm described in previous methods (gained during the sample loading) is used. While searching for the border we inspect always a sequence of three values of the curve to prevent mistakes. By this approach we get left and right border of the pulse from the horizontal cross-section curve and the top and bottom border from the vertical cross-section curve.

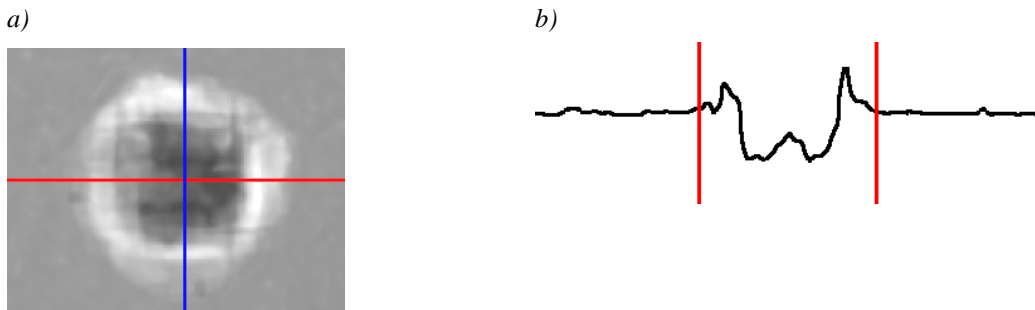


Fig. 15: a) Ideal sample with nearly circular shape; two lines parallel with axis x and y are going through the centre of mass. b) Cross-section curve with defined borders.

In the real cases the method described above is not sufficient, but we can use its result as a starting state for the next processing. If the shape of the pulse is irregular, the centre of mass is shifted from the middle of the pulse (as in Fig. 16a). Moreover, the irregularity of the pulse from the top view deflects the borders (as in Fig. 16b). That is why the previous procedure gives only a rectangle that borders a part of the pulse.

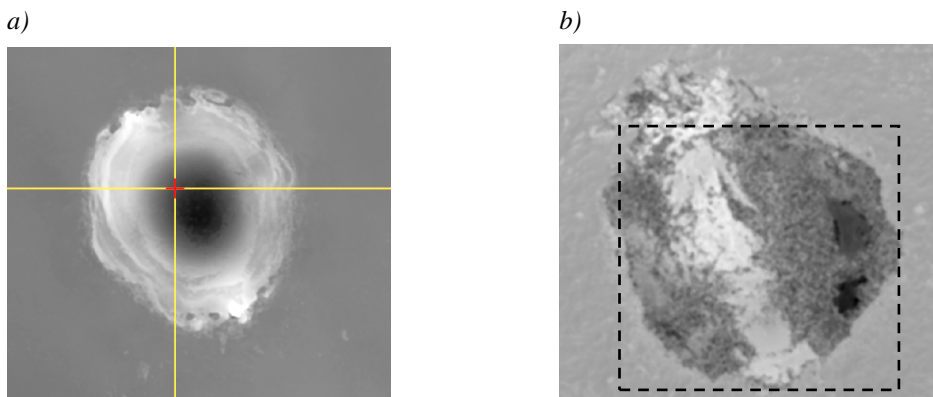


Fig. 16: a) Shifted position of the centre of mass location. b) Asymmetry of the pulse shape in the top part of the pulse from the top view that will cause top border shifting. Dashed rectangle presents the border determined by the algorithm.

3.2.4 Image Processing Methods Application

All methods for pulse detection described above use similar principles and that is why they face the similar problems. So we decided to try to test another approaches. One of the tested alternatives are methods used for image processing and recognition such as thresholding, erosion, dilatation, edge detection, pattern recognition, various types of image filtration etc. [Hlav], [Hlav&Sedl]. These methods are in the process of testing for the present and they are not used for the detection itself yet. Methods are tested in combination with different variants of resampling of the image. The results look well so far, the main problem is the automaticity of the detection process, because for each sample the setting is different.

So far we have tested two approaches. The first one uses thresholding together with image resampling and compression. In Fig. 17 the results of thresholding applied on the sample of 60 pulses burned into the same point into cermet are shown. Firstly, the original image was thresholded by two treshes so the image was converted into three colors – white for points above the material surface, gray for the material itself and black for points under the material basic level. Fig. 17c-d show the same sample modified by the maximal JPEG compression [Blell] before and after thresholding. The compression is used just for the simplification of the implementation because we need to remove high frequencies (which causes roughness of the material and the discrete cosine transformation used by the compression ensures that).

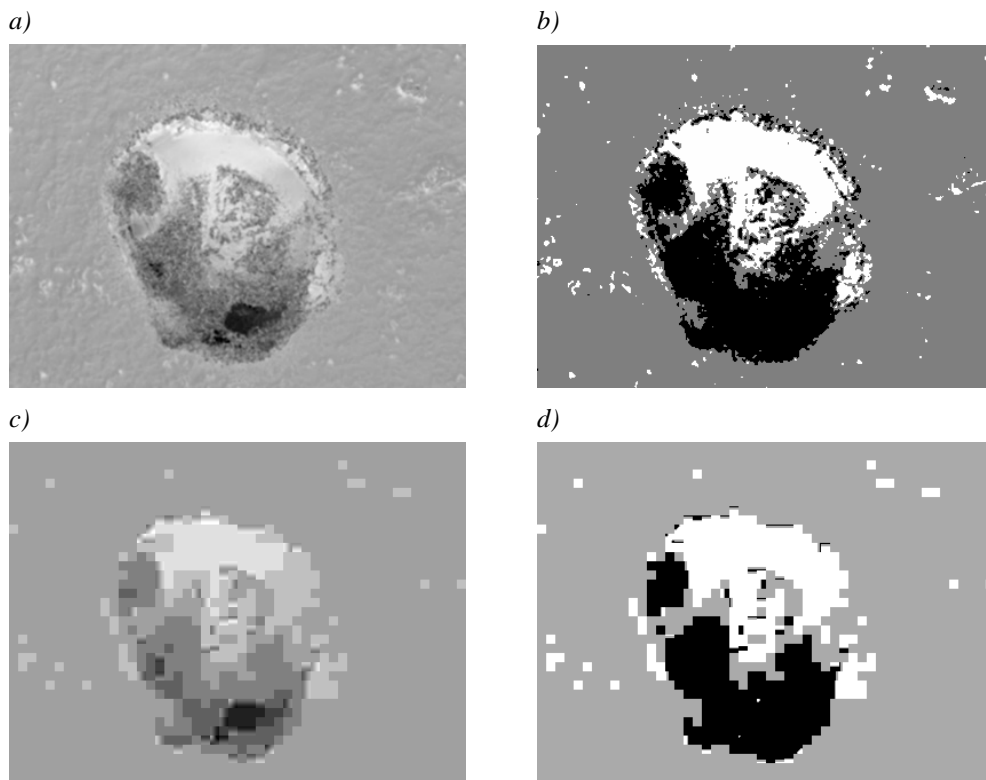


Fig. 17: a) Original sample – 60 pulses burned into the same point into cermet; b) the sample thresholded by two threshes. c) The original sample resampled by the JPEG compression; d) after application of two threshes.

The second way for automatic pulse detection using the image processing methods might be the methods using any kind of filtration in the combination with isolines searching. Also this approach is in the phase of testing. We did not want to implement all

used methods into our system before determination of the applicability of the methods. For speeding-up of the testing process we decided to test the method so far in the Wolfram Mathematica® [wWM] computational environment. We needed to bring data into the computational system in a simple way and so why we decided to use grayscale image representation of sample height maps. For the purpose of algorithm testing the format of grayscale image was easy to load and the exactness of surface description was for the method verification sufficient. Thereafter we were able to apply predefined functions of the Mathematica system to the form the algorithm.

At first the surface of the material had to be filtered. For the filtration the fast Fourier transform [Brigh] was used. By the filtration the surface of the material was smoothed out. In Fig. 18 the original and filtered sample can be seen. The smoothed image is clipped because of the filtration.

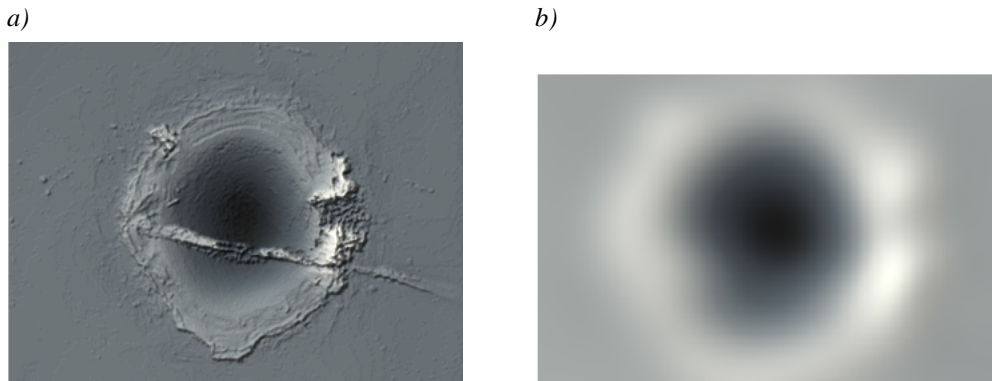


Fig. 18: Images representing the sample surface in the Mathematica®: a) the original one; b) image after the filtration.

After the smoothing of the surface we search isolines. They are created by the values tightly above and under the basic material level level which are connected into the continual curve. Finally the isolines are smoothed out. The result can be seen in Fig. 19. The blue line shows the isoline tightly under the basic material level, the red one represent the isoline above the unburned material surface. In Fig. 19a isolines in the smoothed image are shown, in Fig. 19b isolines are transferred into the original image.

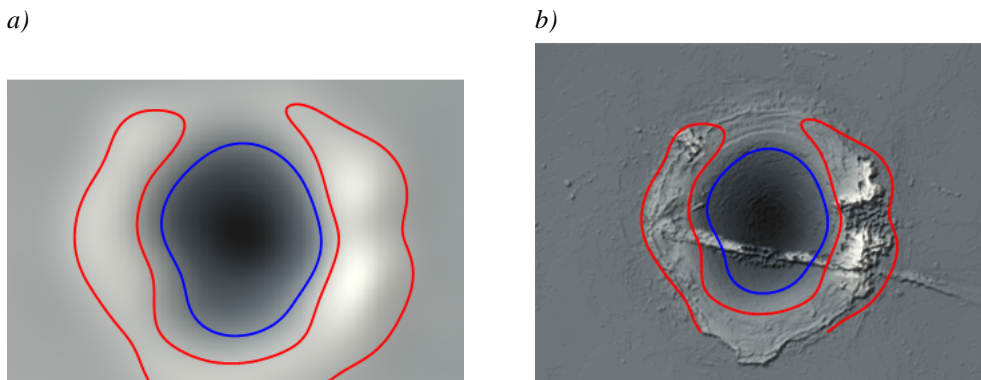


Fig. 19: a) The smoothed and b) the original images representing the sample surface with isolines which were found tightly under (blue) and above (red) the basic material level.

The successfulness of the algorithm depends on several features. One important standpoint is the choice of parameters (the size of the filter mask or the difference between the basic material level and the level of each single isoline). A significant role plays of course also the sample surface itself. Very important is the height symmetry of the pulse, especially its outer border. The lower isoline is in most cases found correctly, the worse situation is during the upper isoline searching. As can be seen in the pictured example, if the material is melted around the pulse hollow irregularly, the filtration is affected by the height difference of the material surface and the upper isoline is not found completely. Moreover, the filtration clips the image by the size of the filtration mask. Because the area of burn pulse takes the majority of the sample related especially to the sample height, the isolines are not computed for the upper and bottom part of the pulse very often. Some of these problems (e.g. the last one) can be solved by the individual adaptation of the filtration for the sample surfaces in the filtration implementation into the simulation system.

Unfortunately, this method is also very slow. The processing of one sample of the original size (1024×768 pixels) needs more than ten minutes. The speed has been partly increased by resampling of the original sample to the size 400×300 pixels, but it is still not optimal. The next speeding-up of the computation could be reached for example by the decreasing of the isoline shape.

Although the approach of surface smoothing and isolines searching is not the optimal solution yet, it gives to us another point of view on the sample and its processing. Except of the automatic pulse detection this method seems to be a good way to get more detailed description of the sample which would enable to compute better statistics over the sample or to compare them from the different point of view. In this approach also linear and non-linear diffusion filtering [Nishi] could be used. We want to test these methods and to search for their best usage in our future work.

3.3 Actual Results of Pulse Detection

Several methods for pulse detection were described. They differ in precision and reliability but also in computational speed. Some of them were implemented to the simulation system and some methods were so far tested in Wolfram Mathematica® [wWM] so that we get preliminary imagination of their applicability. All methods were tested on the same samples which were chosen because of any specific feature. The task of the algorithm has been to detect the pulse in the most perfect way.

In this chapter, problematical samples which are graphically shown in the pictures are described and discussed. Tested samples are shown in Fig. 20. Some of them are burned into cermet (Fig. 20a-d), where the high roughness of the material influences the detection process and some are burned into steel (Fig. 20f-h) which has much smoother surface. Samples with various counts of laser pulses burned into one point of the material were chosen. All samples except the first one were measured with the same scale. Surfaces of several samples are influenced by the local defects of the material and shapes of pulses are in some cases more and in some less asymmetric. In the following text the samples are for the simplification labeled by letters A-H.

A feature which is problematical for methods described in chapters 3.2.1 (Method starting from global extremes), 3.2.2 (Centre of mass method) and 3.2.3 (Spiral method) is the determination of height difference limit. The way of its precomputation as the minimal height difference calculated from several border columns works quite well for the samples without any areal local defect on the material surface. But the areal defect around the pulse can produce incorrect pulse detection as it is shown in Fig. 21.

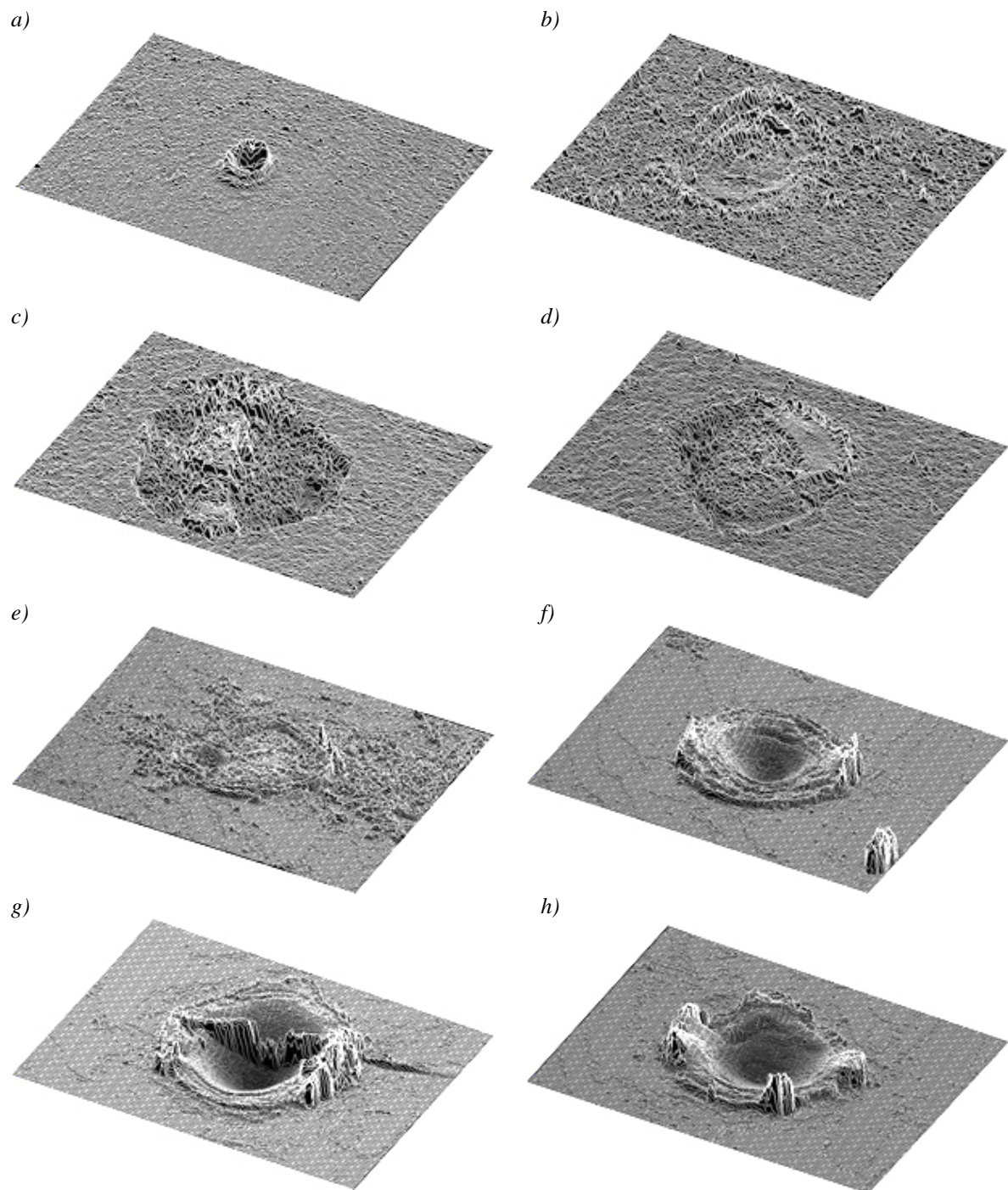


Fig. 20: Samples used for testing of pulse detection algorithms (material of the samples: a-d cermet, e-h steel). a) 1 pulse measured in smaller zoom, b) 2 pulses where the area of pulse is not strictly bounded, c) 70 pulses and d) 100 pulses with the non-symmetric shape of the pulse, e) 1 pulse with the areal local defect of the material surface, f) 10 pulses with the local defect in which the global height maximum is located, g) 50 pulses with a defect in the area of pulse, h) 100 pulses.

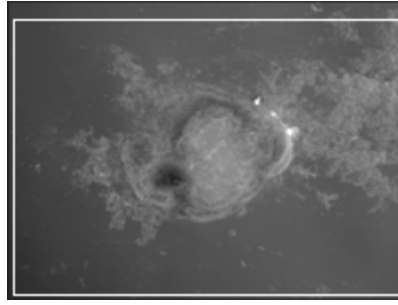


Fig. 21: Sample E with local defect of areal character around the pulse which causes incorrect pulse detection.

Another problem is the roughness of the material surface for example in the case of cermet. Each of methods copes with the problem in a little bit different way. Results of the Centre of mass method are shown in Fig. 22a and it can be compared with the result of the Spiral method in Fig. 22b. The value of height difference limit was computed too high because of the roughness of the material. In the first case where left and right border were found first, part of the pulse was not bordered. In the second case, thanks to the spiral approach, this did not happen at the cost of excessive expansion of bordering rectangle.

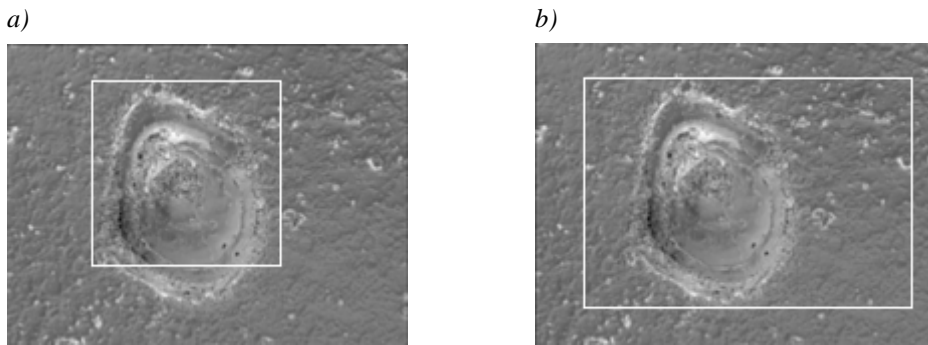


Fig. 22: Comparison of pulse detection for the sample B with the high material roughness – a) Centre of mass method; b) Spiral method.

The main disadvantage of the method starting from global extremes is the setting of the starting position. If there is any local defect of extreme height, the starting point for pulse border search is shifted into the position of local defect and the result is distorted. The sample corresponding to such situation can be seen in Fig. 23. The other two methods have solved this problem by another way of starting point computation.

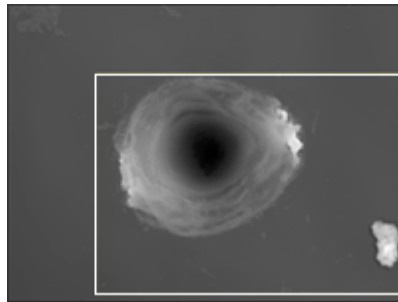


Fig. 23: Sample F with local defect of extreme height in the surrounding of the sample which causes wrong localization of starting position for the method starting from global extremes.

Till now only problematical samples were described. For the rest of samples the results gained by three fully described methods are satisfactory and so the methods could be used for the automatic pulse detection. To describe the category of unproblematic samples is not very complicated. The pulses are well bordered, on the material surface no defect occurs and the roughness of the material itself is minor.

For the other pulses we have to search for any better solution. The following description shows results of tested methods. The results are partial and they should lead to the pulse detection. That is why results for both problematic and well-processed samples are presented. Application of the image processing methods and smoothing and isolines searching methods are so far in the phase of testing and gained results are just partial for the present. Nevertheless, the results look very perspective to be used for the pulse detection in our system.

As one of the methods which could be possibly used for the pulse detection also thresholding was mentioned. In Fig. 24 the results of the thresholding by two threshes are shown. All samples have been processed manually, to get the best results. For these samples no resampling was used. As three examples, samples B (Fig. 24a), F (Fig. 24b) and G (Fig. 24c) were chosen.

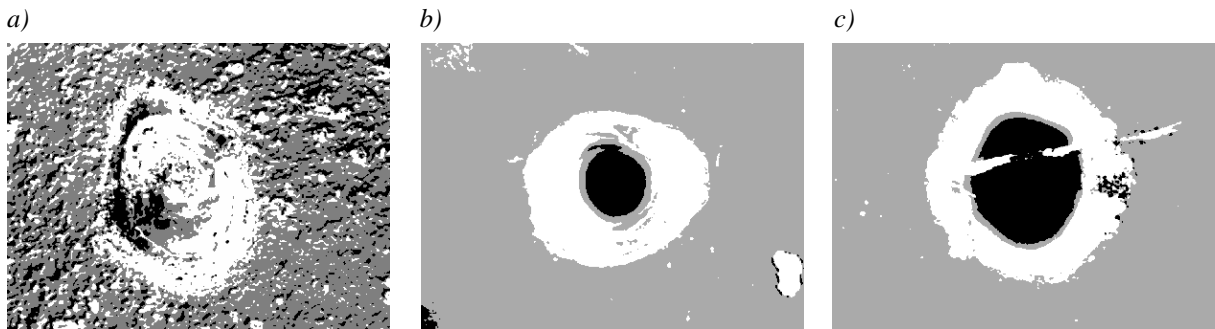


Fig. 24: The results of the manually thresholding by two threshes for the samples a) B, b) F, c) G.

For the method of surface smoothing and isolines searching the biggest problem is the setting of constants. The size of the filter and the value of difference which gives the level of the isoline above and under the basic material level have to be determined. The filter size has to be set to smooth the roughness of the material well, but not to smooth out the whole pulse too much. In Fig. 25 the results of using different size of filter masks is shown. The sample B was used because of high material roughness. In Fig. 25a the mask is too small, in Fig. 25b it is quite optimal and in Fig. 25c it is too large.

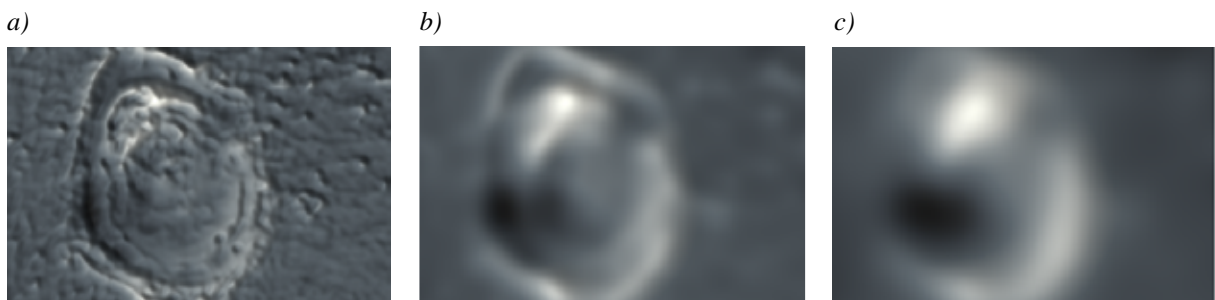


Fig. 25: The results of using different size of filter masks for filtration of sample B: a) too small filter mask, b) optimal filter mask, c) too large filter mask.

In Fig. 26 the same filter size (the optimal one) was used for the sample H and the value of difference between the basic material level and the level of isoline was increasing. In Fig. 26a the isoline was found on the basic material level. The result looks well, but there is a space between the isoline and the border of the pulse. In Fig. 26b the difference was increased and so the isoline was founded more tightly to the pulse. Finally, in Fig. 26c the upper isoline is separated into several parts which border the highest places in the pulse.

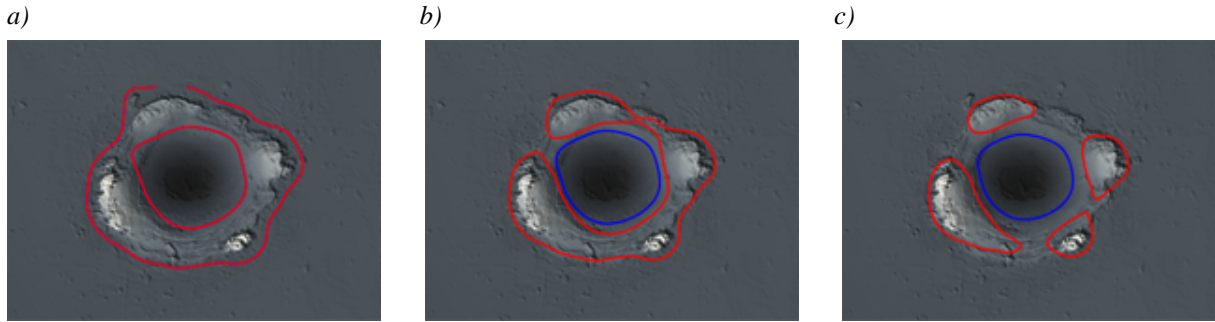


Fig. 26: The results of isoline searching with the same filter, the difference between the basic material level and the level of isolines. a) Isoline directly on the basic material level, b-c) increasing value of difference.

As it was written, this method seems to give good results, but its main problem is in the parameters which can be hardly set automatically. As can be seen in Fig. 27, also for this method the pulse detection for samples E (Fig. 27a) and B (Fig. 27b) is problematic.

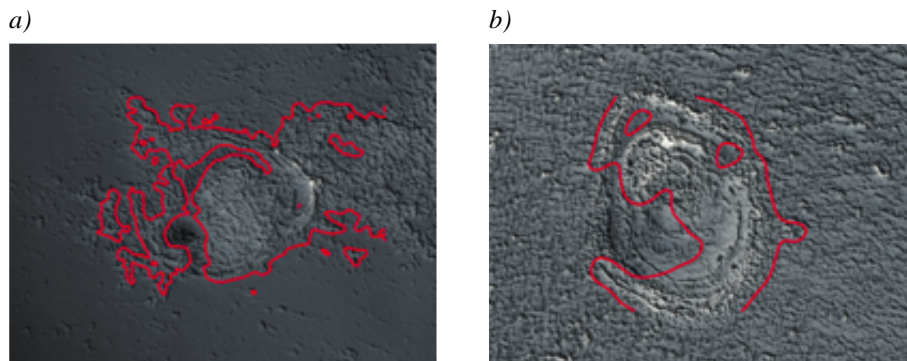


Fig. 27: Isolines computed for problematical samples a) E and b) B gives also not optimal results.

3.4 Pulse Completion

Let us assume we have detected pulses in all samples of the experiment. Each of them is saved in a separate file. The question arises how to combine pulses of the similar samples (it means e.g. all samples representing 50 laser pulses burned into one point into the steel) and to create one optimal pulse. This optimal pulse should not be influenced by the local defects of the material.

The final pulse can be created as an average from all measured and detected pulses, other possibility is to count median or to use any statistics of number of values incidence. Determination of the best method has to be done during verification and result comparing. In Fig. 28a-c three samples with 10 pulses burned into the steel can be seen. Fig. 28d represents an optimal pulse which was created as the average of all three pulses.

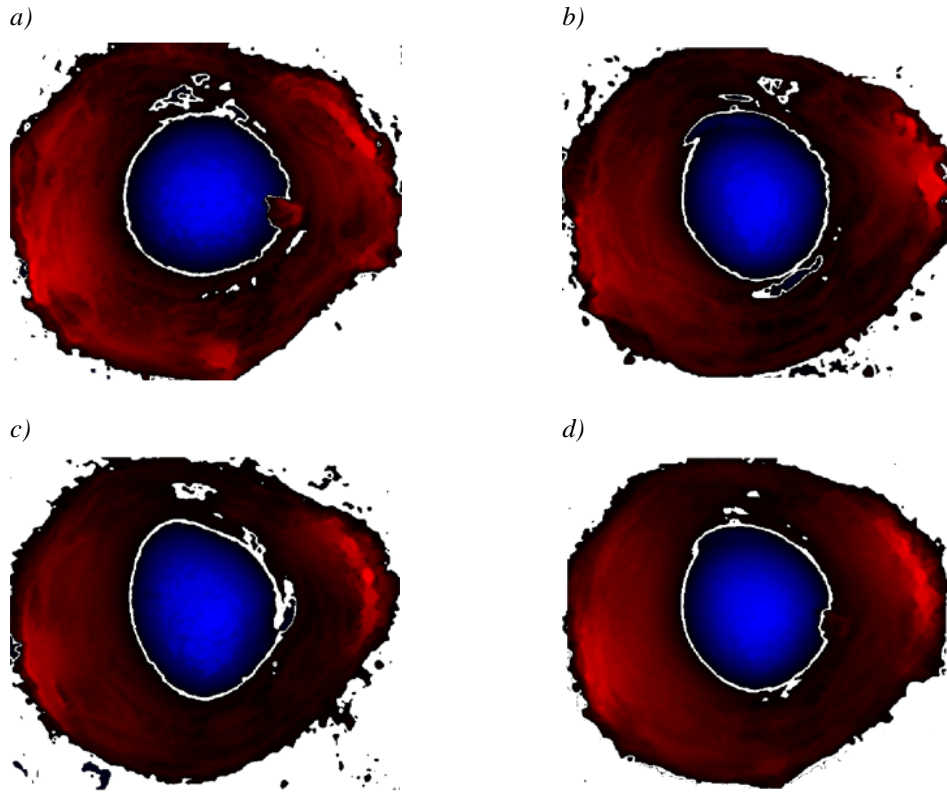


Fig. 28: a-c) Three samples with 10 pulses burned into steel; d) an optimal pulse which was created as the average of all three pulses.

4 Simulation

What leads us to simulation creation, which are our reasons and why it is useful to use simulation for processes like laser burning? General simulation methods, the reasons for the simulation and the concrete technique and method we have decided to use are described in the following chapters.

4.1 Reasons for simulation

The whole system of laser device should serve for miscellaneous scientific and commercial experiments. Results of these experiments are sometimes not fully deterministic, that is why they sometimes need to be reoperated several times to obtain optimal result. Repetitious burning of the same experiment is money and time consuming. That is why any software tool which would eliminate real burning of incorrect results is beneficial.

The simulation should provide experiments as quick and cheap as possible. It would also enable optimization from different points of view (speed, accuracy, etc.) and help to eliminate the unreasonable experiments. All parts of the simulation should be automatic in the maximal way so the simulation can run independently of the user. Moreover, in contrast to real burning where each experiment requires servicing, simulation creates a possibility of batch-oriented experiments executing. After the simulation finishes and all gained results are described by some log system, the best results can be selected.

As a part of the simulation there should be also implemented a tool for data 2D and 3D visualization. This tool would enable to explore real or simulated results and to interpret accuracy and optimality of the simulated sample.

4.2 Technique of Simulation

For the simulation we had to find which simulation method to choose. To be able to decide which method to choose, we have to learn as much as possible about the simulated system. We also have to consider possible cooperation with the real equipment for real data burning and measuring.

We had to decide between two basic ways of simulation. It is possible to simulate the real situation on the basis of analytical methods or to create simulation model with using an application approach.

4.3 Analytical methods

Simulation using analytical approach comes from knowledge of physical equations, mathematical descriptions, procedures and dependencies. Of course, the analytical methods can be discretized and results can be computed numerically. Simulation model which we get by this method can be general enough but for the case of concrete equipment it could be difficult to find the right combination of parameters to describe it exactly.

Nevertheless, the analytical approach is a good way how to model lasers. Not only lasers can be modeled this way, but also e.g. melting processes of the material can be described by analytical methods (e.g. the Stefan problem [Alex]).

4.4 Application methods

Because we have relatively good access to the real laser device and so we can get data from the real experiments, we decided to select another way to simulate laser burning – the application approach. We want to start from real experiments and make use of knowledge of concrete lasers and materials characteristics. Because we use measured real data burned by concrete type of laser in a given material, we can achieve more accurate results in comparison with analytical methods mentioned above. The format of used data is described in Chapter 2.

The pulses are extracted from input data for given combination of used material and laser setting. The basic technique of simulation of samples burning is to place selected pulses gained from input data on the surface of the unburned material. Pulses can be placed also over each other. The format of the pulse is designed for the simplest usage as possible and it offers the ability of direct application of the pulse on the surface of the material (as it was described in chapter 3).

The matrix of pulse height map is during the simulation simply added to the height map of the surface in given point. Negative values representing removed material are subtracted and positive values of melted material around the hole are summed to the original surface (as can be seen in Fig. 29).

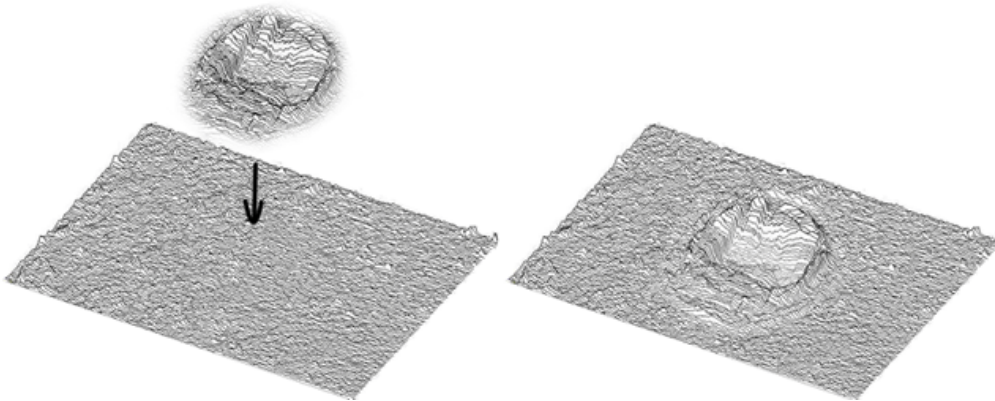


Fig. 29: Material surface before and after simulation burning of the selected pulse.

This simple technique is not able to catch changes of results caused by heating of the material during repetitive burning of laser ray into one point of the material. During one pulse burning this fact does not affect the result, but if more pulses are burned, material is heated and it gets different physical properties.

Another but a little bit similar problem is caused by the ray reflection. If the laser ray burns on the smooth surface it behaves in a little bit different way than if it impacts the surface curved by previous pulses. It can reflect and the material is evaporated in a different direction.

That is why real measured pulses not only for one burned pulse, but also for more pulses burned into one point are used. Using various input samples leads to more realistic simulation. E.g., if pulse with 10 pulses and subsequently pulse with 5 pulses is used for the

simulation, it corresponds more to the situation where the burned material is heated after burning of 10 pulses then it cools down and 5 pulses are burned. Of course, the real 15 pulses are burned without any pause, but for the simulation it can help to better result approximation. Let's assume a task of burning a sample with 15 pulses in one point. We dispose with real measured data for 1, 2, 5, 10 and 20 pulses. There arises a question how to combine existing input to get the best result. There are many possibilities, e.g. $5 + 5 + 5$, $10 + 5$, but also $5 + 10$, because how it is written the order of pulse burning is important. The reasons are heating of the material and especially different way of laser ray affecting the material at the curved surface of the material. The operation of adding pulses is not commutative.

The problems do not occur only during burning into one point. Very similar problem as in the previous case have to be solved – if we burn pulses next to each other during burning along a trajectory. Pulses are laid over each other and part of the pulse is applied on the original surface of the material and part is burned over neighbouring pulse. Heating of the material might not have been taken into account (the temperature of the material does not run noticeably after one pulse burning), but the laser ray would very probably reflect in a little bit different direction and dissimilarities in material vaporization should be covered by the simulation.

Other inaccuracies are usually caused by starting and finishing laser motion. As can be seen in Fig. 30a, the start of burning the shape of trajectory is not accurate at the beginning, also pulses are burned in different distances.

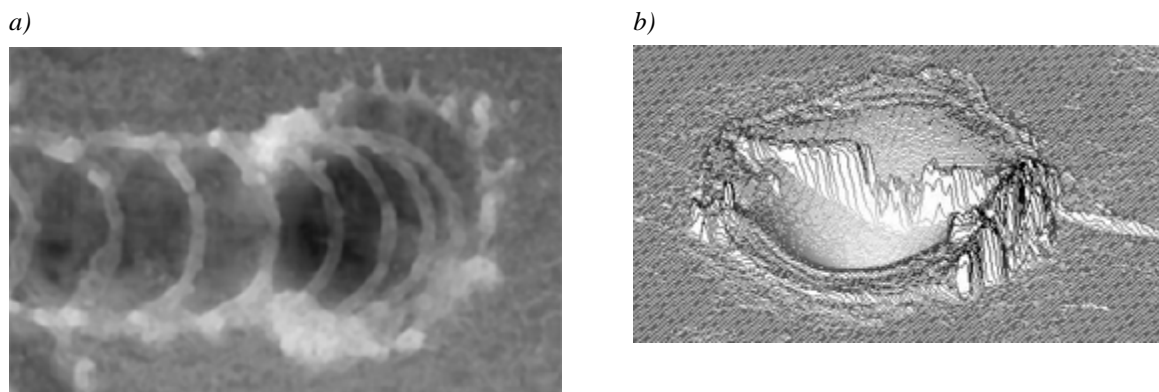


Fig. 30: a) Inaccuracies caused by starting the laser motion (the shape of trajectory is not accurate at the beginning; also pulses are burned in different distances). b) Local defect in the area of the pulse.

Also material which is used for burning can cause problems. Because of height resolution in which the simulation runs, there are visible inaccuracies of the material. These local defects cannot be seen by eye, but they can influence the result of the burning. The probability of error can be decreased by polishing the used material, but it cannot be done each time. There is also not technically possible to measure the material before the burning and to set the exact point where to burn. That is why we can suppose using of an ideal material without any local defects.

There are many factors which can influence the final result of burning and so they should be taken into account also during the simulation. For example surround and atmosphere during the burning itself belongs among such factors. Burning can be provided in the protective atmosphere or in presence of the air circulation. Condition of the surround can influence shape of the pulse, especially the equability of material melted around the pulse.

Because the simulation should reflect the real results as exactly as possible, all these variations should be taken into account. The final result of the simulation would be adapted by using several simulation coefficients. Of course, higher precision will be given to the simulation progressively and it will be also verified in the future.

5 System Verification

For the correct function of the simulation the system has to be verified. During the verification the burned samples are compared with the real ones. The real samples for verification are gained by the same method as it was described in Chapter 2. The system verification can be done manually by comparing and evaluating results visually but the application has to enable to evaluate results automatically. The process of system verification and data comparing serves for simulation system setting.

The manual evaluation is used for single samples. Selected sample is burned by the simulation and compared with another one that is really burned into the material and measured by the confocal microscope. In dependence on the result of comparison, the simulation is adjusted. The comparison is done for several combinations of material and several number of burned pulses or other type of burning sample (e.g. burning along the trajectory).

To test and evaluate the system in a more global way a broad range of samples has to be simulated and compared. An automatic verification is used for speeding it up. In this phase, whole experiments designed for verification can be evaluated at once. For each single case several parameters are computed. Based on these parameters the results can be marked as accurate or problematic. Problematic sample has to be reoperated or the simulation has to cover such problems so that the simulation becomes more precise. After the system is verified it is able to burn the samples as realistic as possible.

A question arises, which parameters to use to represent the difference well enough. Possible methods and approaches are described in following chapters.

5.1 Samples Comparison

Each time during the verification two samples are compared. At the beginning we should discuss possible variants of samples and location of the pulse in the sample. Pulses can be shifted towards each other, they can be scaled or they can have even different shape. Several basic possibilities are shown in Fig. 31 from top view and in Fig. 32 from the side view as the cross-section curve. First sample (a) is compared with other ones. The other sample can be, e.g., shifted (b), scaled (c) or can differ from the first one in the shape (d). Of course, all these differences can be combined together. A question arises, which one of the simulated pulses is more similar to the real one and what pulse characteristics influence the difference most.

Now it is a good time to think, which of described modifications is most problematic. The most uncomplicated situation is in the case of vertical shift (Fig. 32b). Both samples can be aligned to the same height by recounting the surface of the basic material. Solving of the horizontal shift (Fig. 31b) is a little bit more complicated, and it is described in chapter 5.2. The problem of scaled sample or pulse in the sample has to be solved by the simulation model modification. The worst situation arises in the case, when pulses differ in the shape as visible in Fig. 31d and Fig. 32d. This could be solved partly by

changing the input experiment used for the simulation model setting and its verification and also probably by modification of simulation itself.

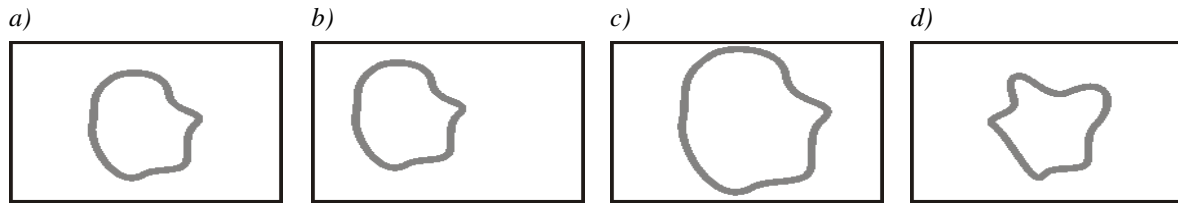


Fig. 31: Possibilities how can a sample a) differ from another one which is shifted (b), is scaled (c) or has different shape of pulse (d). All variants are examined from the top view.

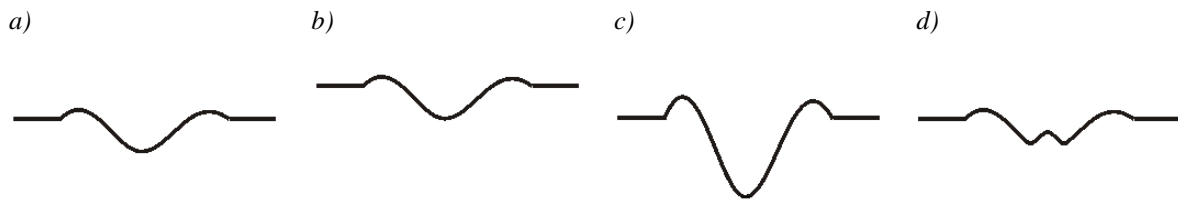


Fig. 32: Possibilities how can a sample a) differ from another one which is shifted (b), is scaled (c) or has different shape of pulse (d). All variants are in the form of cross-section curve examined from the side view.

Of course, to get correct comparison results, samples must be saved in the same precision and resolution or they have to be recomputed before the comparison itself. The recomputation is also done in the case that basic material levels of both materials are not in the same height. By this operation we eliminate shifting of both samples in the vertical axis.

On the surface of the sample the roughness of the original material is visible. Also local defect of the material can change the original surface unpredictably (as described in Chapter 3.2). That is why we would not get a zero difference after comparing two real samples of the unchanged material (samples without any pulses) measured in different places. It shows the fact that even more samples of the same basic material will differ. There can be seen that the roughness of surrounding material can increase rate of inaccuracy of compared samples. This happens also in the case of samples in which the pulse fills the majority of the sample surface. To increase the precision we should try to evaluate only the area of burned pulses and closest surroundings. For this purpose, the methods and techniques used for automatic pulse detection in the sample described in Chapter 3.2 can be used successfully.

Another question is what to do in the case of different dimensions of the samples (Fig. 31c). If the sample sizes differ, but whole burned areas are correctly detected, they can be resized and compared. In the case where only a part of the burned surface is presented in the sample, it is important to localize corresponding parts before the comparison. Localization of corresponding parts does not have to be simple in all cases.

Let's suppose that we detected burned areas in both samples and we want to compare them. Even though we expect minimal space around the burned area it is necessary to overlap both samples over each other so that we really compare only burned areas. Usable methods, possibilities and problems of samples overlapping are described in Chapter 5.2.

Samples are represented as height maps, so that the problem is about to compare two surfaces described in the form of uniform rectangular grid. This representation can be

transferred into the format of grayscale image. So that the problem of two burned samples comparison can be solved by methods and algorithms for image comparing [Zapl]. Methods based on image comparing are not the only possible way how to solve the problem of sample comparison. As a promising approach seems to be the usage of the Principal Component Analysis which is outlined in Chapter 5.1.4.

We have to decide which information we need to get as the result of samples comparison. Some methods give us one value. This value expresses total difference. Such methods are suitable for automatic verification. The other group of methods shows difference in the whole image (or image part) and enables better localization of problematic parts. However they are unsuitable for automatic processing.

In following three chapters, methods used for images comparing which can be used for samples comparison, are described and their usage are showed for concrete samples.

5.1.1 MSE

This method counts the mean square error (MSE) of both samples. It is defined as the difference of pixel heights of samples squared for the error highlighting. Differences for all pixels are summed and divided by the sample surface dimension. The resultant value expresses average error of each pixel. The calculation is given by expression (6.1), where W and H represents width and height of the samples, $real(i, j)$ and $sim(i, j)$ indicate single points at the given position in the sample height map grid.

$$MSE = \frac{1}{W \times H} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} (real(i, j) - sim(i, j))^2 \quad (6.1)$$

It is easy to see that for more identical samples the lower result of MSE method is computed. Let us compare two samples with the same pulse counts from the real experiment (5 pulses burned into one point in the steel; shown in Fig. 33a-b. Samples are very similar, but nevertheless they differ a little. The MSE for this case reached the value of 0.385. The other case is comparison of two absolutely different samples (sample with 5 and 50 pulses burned into one point in the steel; shown in Fig. 33b-c; MSE = 4.186.

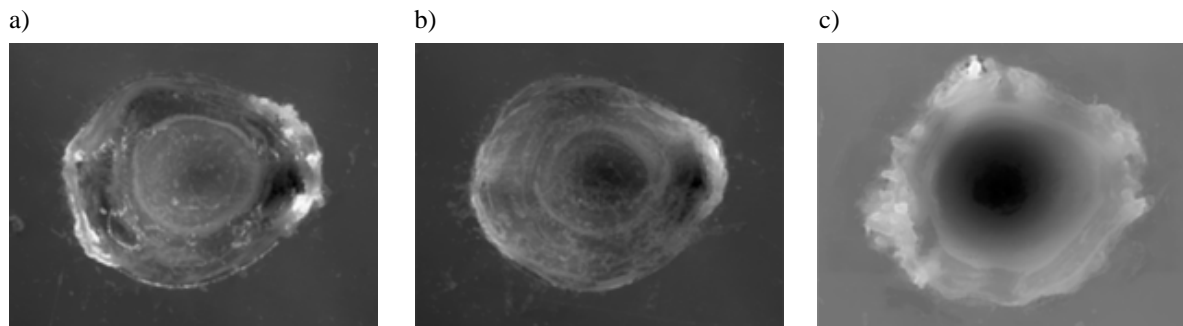


Fig. 33: a-b) Two similar samples with 5 pulses burned into one point into the steel; c) sample with 50 pulses burned into one point into the steel.

The results of MSE computation for sample height maps are much smaller than in the case of grayscale images. The reason is simple, gray scale images consist of values in the interval $\langle 0, 255 \rangle$ while values representing the sample surface reach the maximal values of about a few tens. Moreover, before the computation itself, basic levels of both

samples are shifted to the same height and so the differences among values of both samples come more near. After the sum of individual differences is divided by the total dimension of the sample, the result is typical between 0 and 10.

During the MSE computing for the whole sample a computational error arises. It results from among of the points on the basic material level where the zero difference is computed. These points do not add any increment into the total sum, but they are included into the sample dimension and so they distort the result. That is why to get the most accurate results it is important to compute the MSE only in the area of pulse. For the purpose of the automatic pulse detection algorithm described in Chapter 3.2 can be used.

5.1.2 Difference Image

Difference image is in principle visualization of the MSE. Color of the pixel is defined as the difference of heights of both samples at the corresponding position recounted to grayscale interval. The dependence can be simply described by expression (6.2), where *min* and *max* represent minimal and maximal values of difference. The method differs a little from the typical difference image counting, where the absolute value of difference is used. By this modification we preserve information about the order of sample heights (we can find out from the image which sample has the higher value representing the surface in the given pixel).

$$dif[i, j] = 255 * \frac{(real(i, j) - sim(i, j)) - \min}{\max - \min} \quad (6.2)$$

The color of the basic material level represents the level of zero difference; all points which are darker are in positions, where simulated sample surface was higher than the surface of the real burned sample. In the opposite cases where the simulated sample does not reach heights of the real sample, the pixel is lighter. The points with highest difference have white (real sample is higher) and black (simulated sample is higher) color.

Results of comparing of the samples described in the previous chapter are shown in following images. Two different samples with 5 pulses burned into one point were compared and its result can be seen in the first column of images. The comparison of sample with 5 pulses and the sample with 50 pulses burned into one point are placed in the second column. All samples are burned in the steel. Difference images are depicted in Fig. 34a. Together with difference images also horizontal cross-sections going through middle of the samples are shown. In Fig. 34b cross-sections of difference images are visualized and in Fig. 34c cross-sections of both simulated (black curve) and real (gray curve) sample are presented.

5.1.3 Logarithm Operator Adaptation of Difference Image

For difference images with broad range of values, small differences can not be distinguished well, because they are in the image represented by very similar or even the same gray color. If we are interested in these small difference values, we have to highlight them. It is possible by using the logarithm operator [wLogOp] and to use logarithmic function as a mapping function for difference image modification. During this operation each pixel value is replaced with its logarithm.

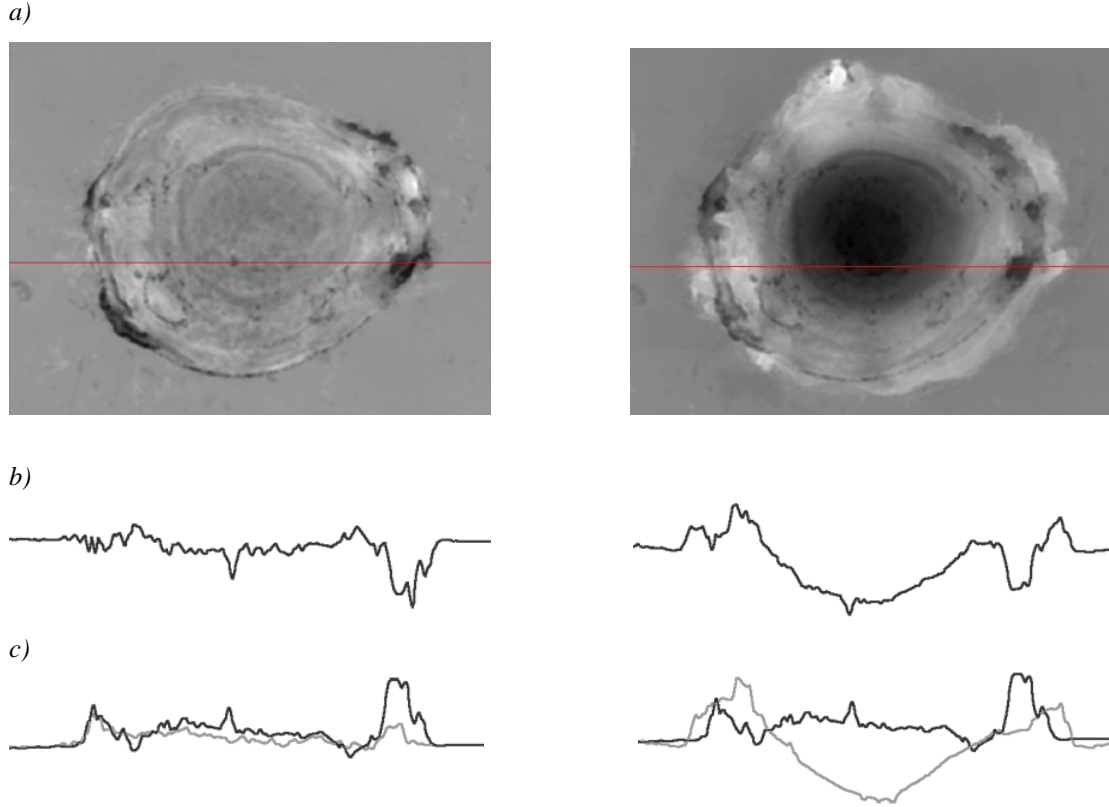


Fig. 34: Difference images of comparison of two similar samples (the left column) and two absolutely different samples (the right column). a) Difference images, b) horizontal cross-sections of difference images, c) horizontal cross-section with curves for both simulated (black curve) and real (gray curve) sample.

The expression (6.3) is typically used for logarithmic modification of images. It expresses logarithm operator mapped on the standard difference image with the absolute value of difference. The basis of logarithm does not influence the result; there can be used e.g. natural logarithm or the base of 2 or 10 logarithm. The multiplicative constant of the expression ensures scaling to values in the interval $\langle 0, 255 \rangle$ representing gray color. Because the logarithmic function is not defined for 0, the value of 1 is added to the parameters of logarithmic function used in the expression.

$$difLog[i, j] = \frac{255}{\log(1 + 255)} * \log(1 + Abs(real(i, j) - sim(i, j))) \quad (6.3)$$

As it was written in previous chapter, the modified difference image is used for the samples comparing. That is why also logarithm operator has to be used in a modified way. We have to distinguish points, where the height of simulated sample is higher than the surface of the real one (such points are in the difference image placed under the basic level of the material) from the opposite case. That is why we have to apply logarithmic operator separately on the points above and under the basic material level. The material level can be declared as zero, because both samples have been aligned before the sample comparison (the reasons of this operation is described in chapter 5.1).

The modified expression (6.4) goes from the value $diff[i,j]$ computed by the expression (6.2). The value M is counted for each of two cases separately as a number of gray colors between the material base level and maximal difference in the relevant direction. For example, if the basic material level is represented by the gray color with the intensity 160, all pixels above the material are spread into 95 levels of have the value in the interval $\langle 0, 159 \rangle$. Value of the fraction at the beginning of the expression differs for converting for the points above and under the level of basic material.

$$difLog[i, j] = \frac{M}{\log(1+M)} * \log(1 + dif[i, j]) \quad (6.4)$$

The results can be seen in the following images; in Fig. 35a the original difference image is shown, in Fig. 35b the difference image modified with the logarithmic operator can be seen. The difference of both images is visible well. The original difference image defines the color equally in dependence of the displayed color. From this representation we can get good imagination of heights distribution. On the other hand, from the logarithmical modified image we can better find out small differences in the neighbourhood of the material level.

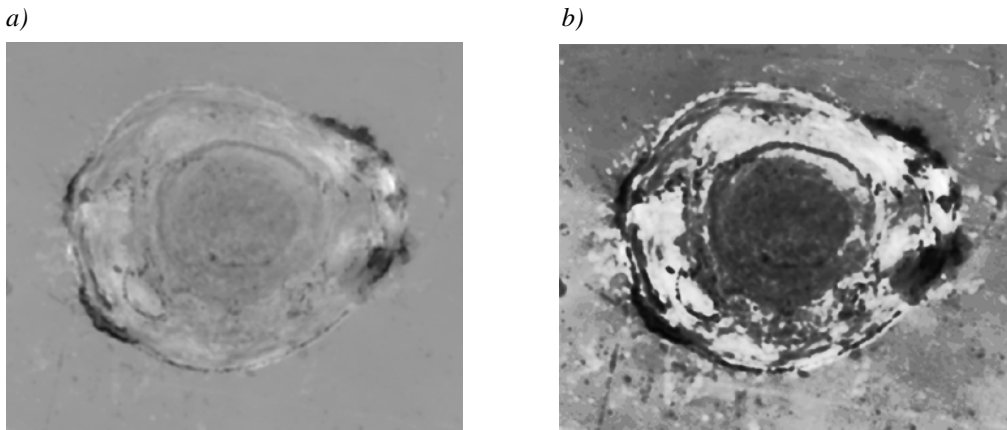


Fig. 35: Difference images of two compared samples a) the original difference image, b) difference image with logarithm operator modification.

5.1.4 Principal Components Analysis

Principal Components Analysis (PCA) can be used as a powerful tool for data analysis. It bases on computation of the eigenvectors and eigenvalues which help to identify patterns in data. The data set is expressed in such a way that similarities or differences are highlighted. PCA seems to be a promising approach for the comparison of samples. All peaces of information for this chapter were gained from [Smith], where the whole method is explained in a very understandable way. That is why we will not describe here the whole method in detail, but just several main features will be outlined.

The method runs in several steps. At the beginning data has to be loaded. An example of the original data set is visible in Fig. 36a. To get data set with the mean value of zero (which are used in PCA), all values have to be shifted in each dimension by the mean value (by subtracting the average in all dimensions). An example can be seen in Fig. 36b.

From the n -dimensional data set the covariance matrix of dimension $n \times n$ is calculated and n eigenvectors and eigenvalues are computed. All eigenvectors are orthonormal (i.e. they are perpendicular to each other and have the unit length) and one eigenvalue belongs to each eigenvector.

The computed eigenvectors represent patterns in the data set. The eigenvector with the highest eigenvalue (it is called the principle component) determines the main direction in which the data is placed. If we sort all eigenvectors by their eigenvalues, we get the sequence where the vectors at the beginning have a significant influence on data distribution while the last eigenvectors affect the data distribution minimally.

Computed eigenvectors are orthonormal and so they can form a base of a new coordinate system. At this time we can transform data into the new coordinate system. We can use all dimensions or we can ignore several eigenvector with the lowest eigenvalues and so reduce the number of dimension. The main reason is the elimination of tiny differences in dimensions with a low influence. The result of data from the example derived into the new coordinate system is shown in Fig. 36c, before the transformation the minor eigenvector was ignored. It returns the original data just in the direction we have chosen. In fewer dimensions it is easier to compare samples among each other.

Of course it is also possible to transform the data back into the original coordinate system. If we have ignored some eigenvectors, some information from the data is lost (as can be seen in Fig. 36d); otherwise we get the original data.

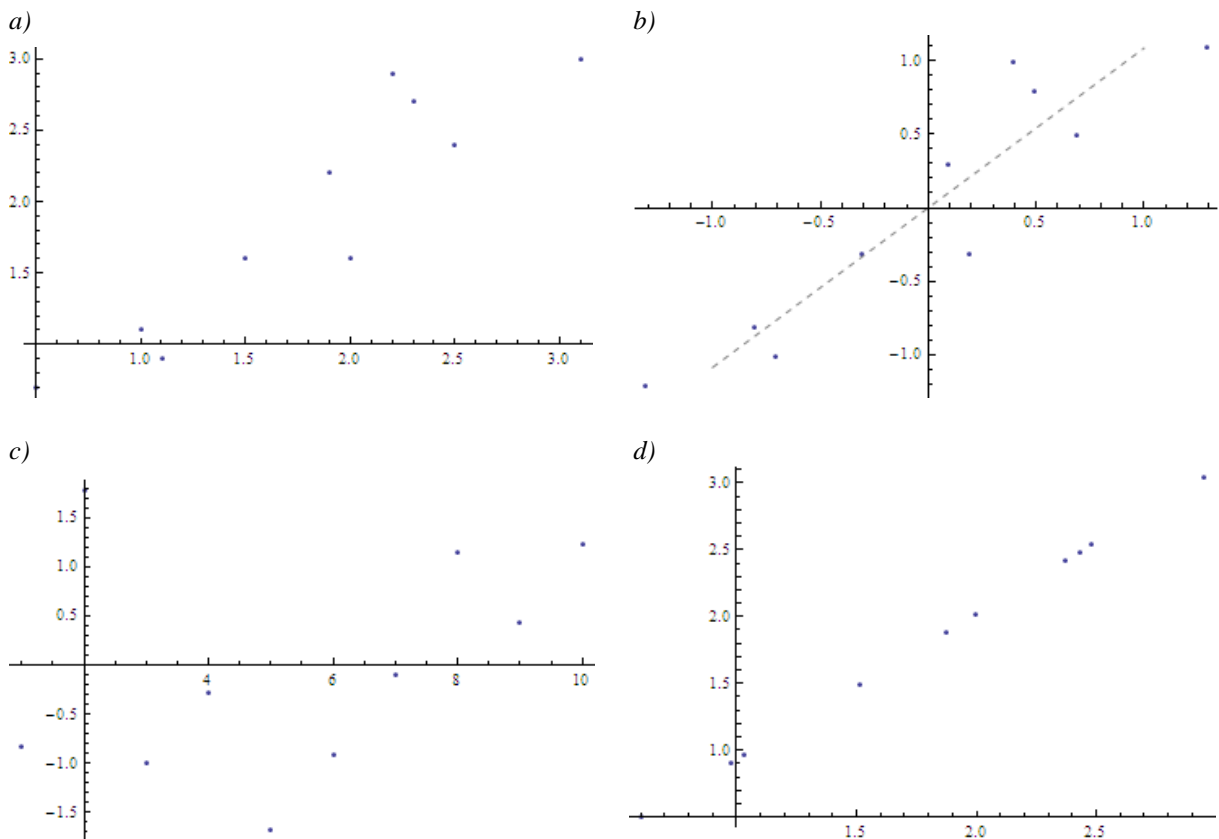


Fig. 36: Visualized data in various phases of the PCA method: a) the original data, b) data shifted by subtracting of the average in both dimensions into the position of zero mean, c) data transformed into the new coordinate system after ignoring the minor eigenvector, d) reconstructed data derived back into the original coordinate system after ignoring the minor eigenvector. [Smith]

For the representation of the height maps for the PCA processing each sample should be represented as one vector. It can be done by putting all sample rows after each another to create a vector of the length $M \times N$ (where M symbolizes the number of rows and N the number of sample columns). For all images we want to compare we have to create such a vector representation and from all these vectors a matrix is composed – each image vector creates one matrix row. Then the covariance matrix has to be computed and for the covariance matrix the eigenvectors and the eigenvalues are calculated. In the next step, the original images are recomputed into another axis representation derived from the PCA. After the PCA analysis the difference of images can be measured along the new axis.

The PCA method is successfully used for faces recognition [Zhang], so we would like to try the method for the pulses comparing and maybe also for the pulse recognition.

5.2 Samples Overlapping

To get optimal result for simulation verification it is necessary to overlay burned pulses as exactly as possible. Otherwise the error of comparison is enlarging in dependence on relative shift of burned area in the sample. The manual overlapping is possible, but it can not be used for automatic verification and results evaluating. That is why the system has to be able to find the best matching position independently.

One possible way how to match pulses automatically is to compute any parameter representing the difference of both samples (e.g. MSE) in a given position for all possible positions of two samples. It could be the optimal solution, if it were not for the problem of speed. E.g., for two samples of the size 1024×768 values would be the parameter computed 768432-times. If one calculation takes 1ms, the best position of samples would be decided after approximately 128 minutes. That is why we have to reduce number of repetitions as much as possible, but at the same time we have to preserve sufficient precision of the result.

First, we can eliminate all the positions, in which only borders are overlapped (as shown in Fig. 37). We can suppose that among these combinations there will be probably no optimum.



Fig. 37: Positions of overlapped samples which can be most likely excluded from searching for optimum.

We have to find the best starting position (a guess of optimal position of overlapped samples) and from there we have to start searching the real optimal position of overlaid samples in its surrounding. We can use, e.g., the centre of mass as the starting position (the method of centre of mass calculation is described as a part of Chapter 3.2.2). The centre of mass can be computed for both compared samples and samples can be shifted to overlay their centers of mass. Another approach offers the usage of the method of automatic pulse detection (possible approaches are outlined in Chapter 3.2), to detect pulses in both samples and to overlay the selected areas. In the second method we can directly take advantage of the pulse detection for the faster computation of difference only in the selected area.

After finding the starting position we just have to find the optimum by small shifting of samples over each other. The dimension of area has to be determined where the optimal position can be found. Further positions the step for shifting is rougher, in the close surroundings the shift step is chosen smaller. For each of these combinations the difference of the samples is computed, so the algorithm of difference parameter computation has to be quick enough so that we reach computation acceleration.

6 Results

The first thing to be solved during the project was the data preparation and processing. Because no tool for data exploring and its modification was available so far, we had to start development of a new software tool that would help us with data processing. Many functions for pulse detection and extracting, samples viewing, modification or comparing were required. Plenty of functions which are divided into several categories in dependence on the phase of data processing are designed and implemented. The most often used functions were matched with key short cuts for the operations acceleration. Thanks to the language localization the tool can be translated into another language without any problem.

The tool works in three modes of data processing. The first one which is described in Chapter 6.1 serves for sample viewing and exploring. In this mode pulses are extracted, statistics over the sample are computed etc. This part of the tool is used also separately by our colleagues from the Department of Physics for the exploring of real measured samples. The second mode serves for the simulation of burning, in which the extracted pulses are used. The burning process can run automatically after describing of the experiment and setting parameters of the simulation or single pulses can be made manually. All functions of the second mode are described in Chapter 6.2. The third mode is used for samples comparing and simulation verification. Details of this functionality are introduced in Chapter 6.3.

6.1 *Sample Viewing*

The sample viewing mode offers broad range of functions for sample exploring and data preprocessing. For the sample preprocessing a sample description is loaded from the file. Format of the used files is described in the Chapter 2.

The sample can be explored in many ways. The main window offers the 2D visualization. It means the sample is shown in three views in the direction of three basic axes of the coordinate system (it can be seen in Fig. 38). In the central part of the window the explored sample is represented by the grayscale image. Intensity of the gray color reflects the height of the surface at the given position. The highest point of the sample is represented by the white color, the lowest point is black. For viewing the sample from the orthogonal directions cross-sections of the surface in both directions are used. They are placed on the left and bottom side of the window. In the left part of the window the cross-section in the vertical direction is projected, in the bottom part horizontal cross-section curves is visible. Both cross-sections are in the top view represented by blue and red lines which show the column and row that are visualized in the form of cross-section curves. The cross-section curve reflects surface of one column or row of the sample. For both vertical and horizontal cross-section maximal and minimal value is marked. Position of the cross-section lines can be changed by the mouse dragging or by using the user interface of the control panel in the right part of the main program window. In the control panel there are also indicated both real and image units of placing the cross-section

lines. At the crossing of both lines their image position and the real height of sample at the crossing point are given.

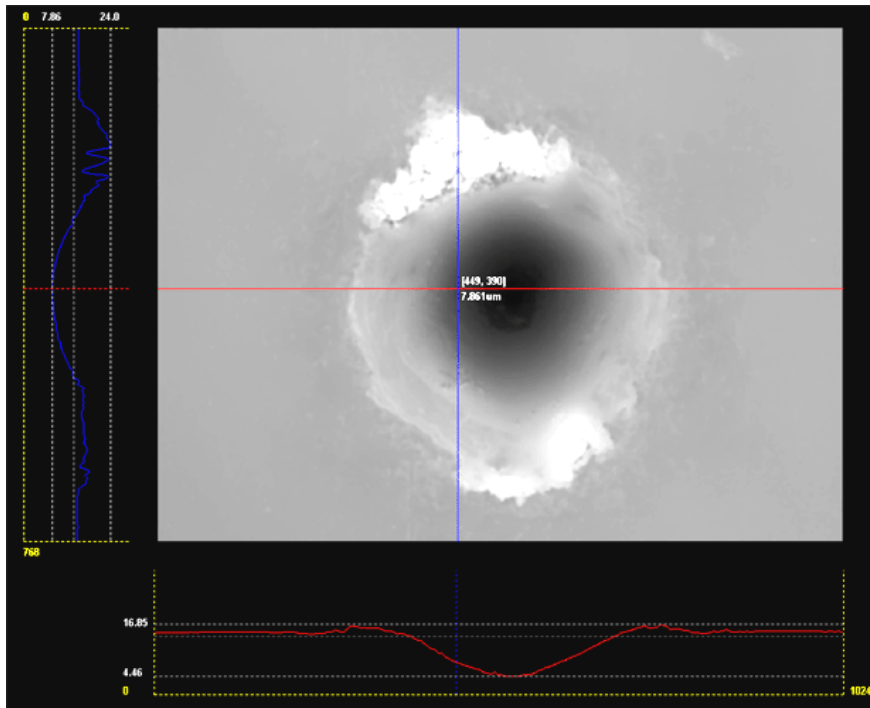


Fig. 38: 2D visualization of the explored sample – the top view represents gray scaly image of the sample, in the left and bottom part vertical and horizontal cross-sections can be seen.

Except of the cross-sections parallel with basic axis described above, the sample editor offers also the possibility of viewing the surface in any direction. After setting end points of the line segment, the curve of surface cross-section along this line is visualized in a stand-alone window (see in Fig. 39). The line segment between two defined points has to be computed and so we have to implement any algorithm for line segment rasterization. For this case, using DDA (digital differential analyzer) [Žára] algorithm was used.

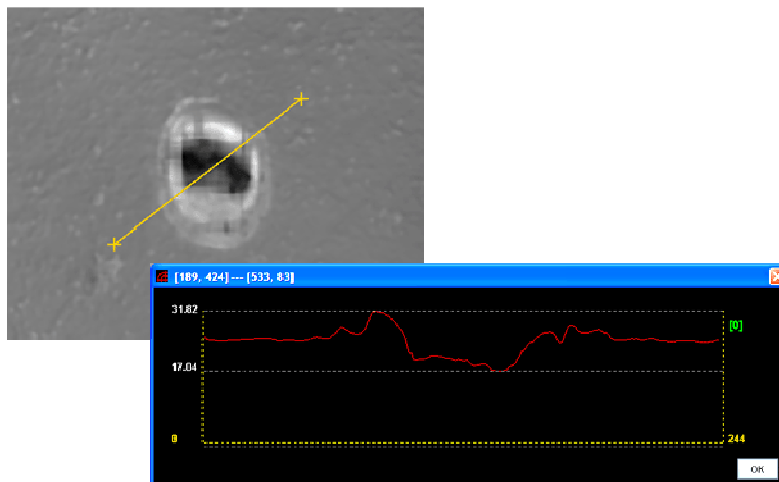


Fig. 39: Visualized curve of the sample surface cross-section along the line segment in a separate window.

For samples exploring it is very useful to dispose with any tool for lengths and depths measuring. That is why the tool of ruler is implemented in the system. It enables to measure distances in all views of the explored sample. Results of measurements are given in real units. Using of the ruler is simple and if the tool is activated, the user can measure distance by setting of two points with the right mouse button clicking. Size of the pulse (width and length) can be measured in the central top view (see Fig. 40). In the vertical and horizontal cross-sections the ruler can be used for measuring of the pulse depth. The result of measurement gives the difference between two points of the cross-section curve.

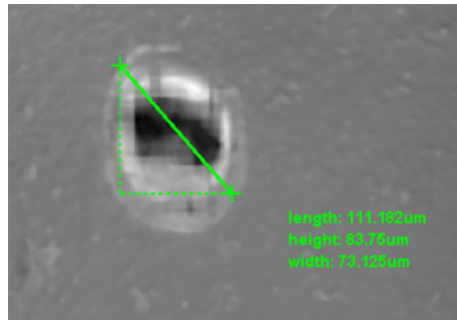


Fig. 40: Size measuring of the pulse in the top view – distance of two points is measured in the real units.

For better knowledge of the sample surface height, the gray scale indicator can be activated. It shows minimal and maximal height of the sample which are colored black and white and the gray scale between them (as shown in Fig. 41). The red line shows height of the point which is currently at the mouse cursor position.

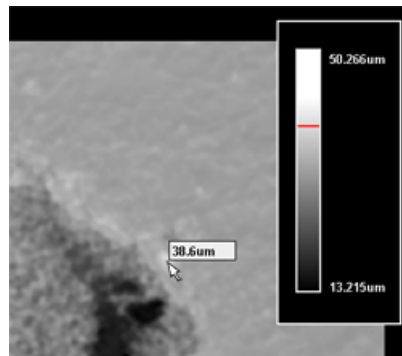


Fig. 41: Gray scale indicator shows by the red line actual height of the point at the cursor position in comparison with heights of the sample.

To have another view of the sample, it is possible to use a 3D visualization. Thanks to the mouse motion or using of arrow keys the viewing camera is rotated, moved or tilted to get the best position for exploring the sample. The spatial view offers the better imagination of the real sample surface. By the camera rotation and motion of the sample can be explored perfectly from all sides. Moreover, the sample can be explored in the form of parallel line cross-sections. This way of visualization can be better for some cases. Line spacing can be changed if needed. Both ways of 3D visualization are shown in Fig. 42. For the 3D visualization we use OpenGL [Wri], [wOGL] and because the whole simulation system is implemented in Java, for cooperating OpenGL and Java we use JOGL (Java for OpenGL) [wJOGL].

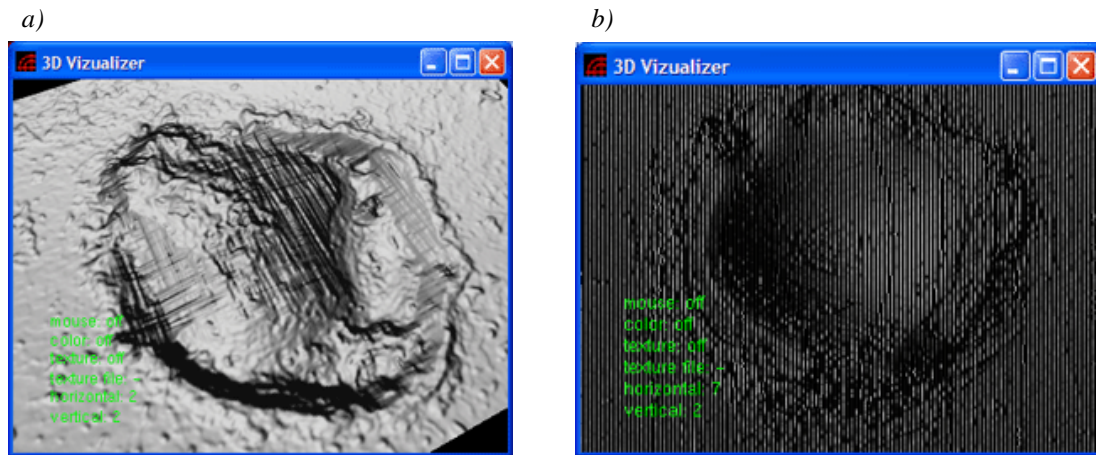


Fig. 42: 3D views of the sample in the form of a) full surface, b) line cross-section visualization.

Another very useful function for sample exploring is the statistics computation. Sample statistics information contains basic values of sample dimension, minimal and maximal height of the surface, but also volumes of the pulse parts under and above the basic level of the material are computed. The results represent the amount of material which has vaporized (the volume of the hollow measured under the basic material level) and the amount of material melted on the border of the pulse (the volume of material melted around the pulse measured above the material). Unfortunately, if the used material has the high roughness, it influences the computed result. That is why two values are counted, the first one for the whole sample and the second one only for the area of selection which would border most likely the area of pulse. Volumes are computed by applying numeric integration.

There is only one problem – locating the basic level of the material (it means the surface of the material before the pulse burning) and to identify it by one value. In Chapter 3.2.1 the possible method of a difference limit value computation is described. The middle value of the unburned material surface is computed in a very similar way. If the roughness of the material is not insignificant, the value is defined as an average of a part of the sample. If the typical form of the input samples (a pulse is placed in the middle of the sample and the borders of the sample correspond to the basic material) is processed, borders can be used for computing the value of the material level. The system enables using several boarding rows or columns, for untypical samples average of minimal and maximal height in the sample can be used. The sample statistics dialog can be seen in Fig. 43, the material basic level was defined as an average from several border rows.

The sample viewing mode serves also for pulse extraction. This task includes problems of automatic pulse detection, saving, reloading or modification. All these operations are in a detailed way described in Chapter 3.

Because the editor works with the measured real data, it is also necessary to assign to the loaded sample concrete material and laser setting that were used for burning of the real pulse. Various parameters of the laser can be set (e.g. frequency, width of the ray, diaphragm, the speed of the laser or the current). Also different patterns of signal behavior and maximal amplitudes can be determined. Material can be characterized by various parameters such as surface roughness, hardness, thermal resistance, etc.

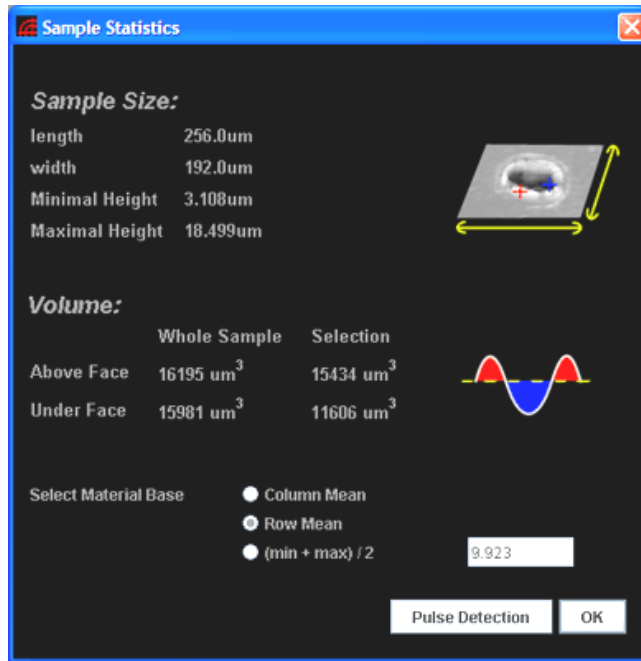


Fig. 43: Statistics dialog with all important values related to the explored sample.

For the operation of laser and material assigning to the sample a package of prepared materials and lasers was created as the separate function of the implemented tool. All created materials and lasers are saved into a XML file and reloaded after starting of the editor. Material and laser descriptions can be edited, deleted or extended by the user, but this part of the editor is not fully completed yet, because the requirements for all parameters have not been described completely yet. The present appearance of the tool for creating a new laser is shown in Fig. 44.

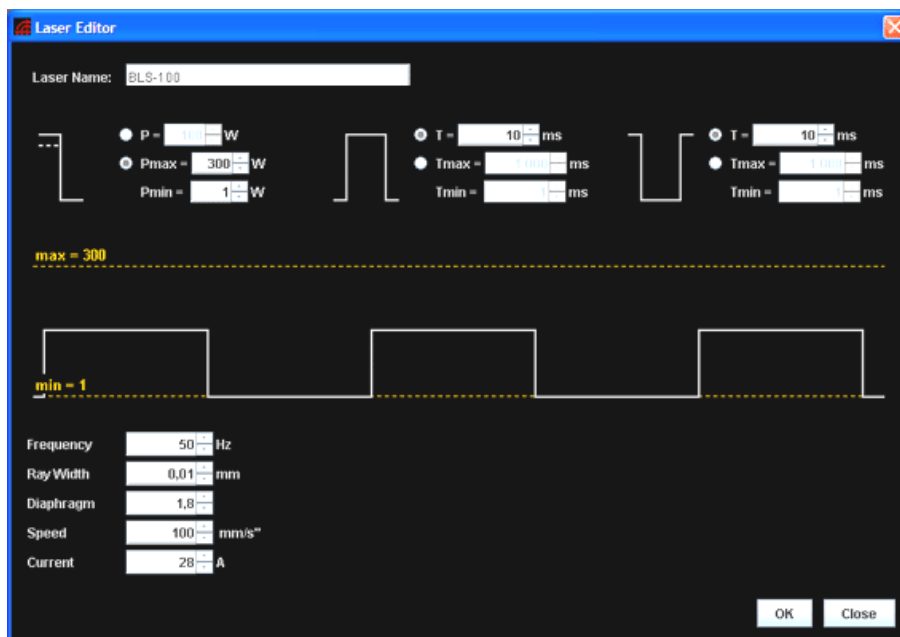


Fig. 44: Present appearance of the tool for creating a new laser description. Various parameters of the laser can be set.

Because in the sample viewing mode the sample can be clipped and pulses should be extracted, a function of sample part selection is needed. The selection function is used quite often, so it has to be manipulated in a direct and simple way. Borders of the selection can be changed manually by the mouse dragging in all 2D views. In the top view both width and height of selected area can be chosen, in the vertical view only height and in the horizontal view only width of selection is implemented. Another way how to set border position exactly is to set selection border values in the right control panel of the window. Selection borders are also changed automatically after the automatic pulse detection. The dimension of the selection is given also in real units, the information about the selection dimension is written in the right control panel.

An example of the visualized sample with detected pulse is shown in Fig. 45. All three views of the 2D visualization of the sample are shown; the pulse is bordered by the yellow rectangle of the selection.

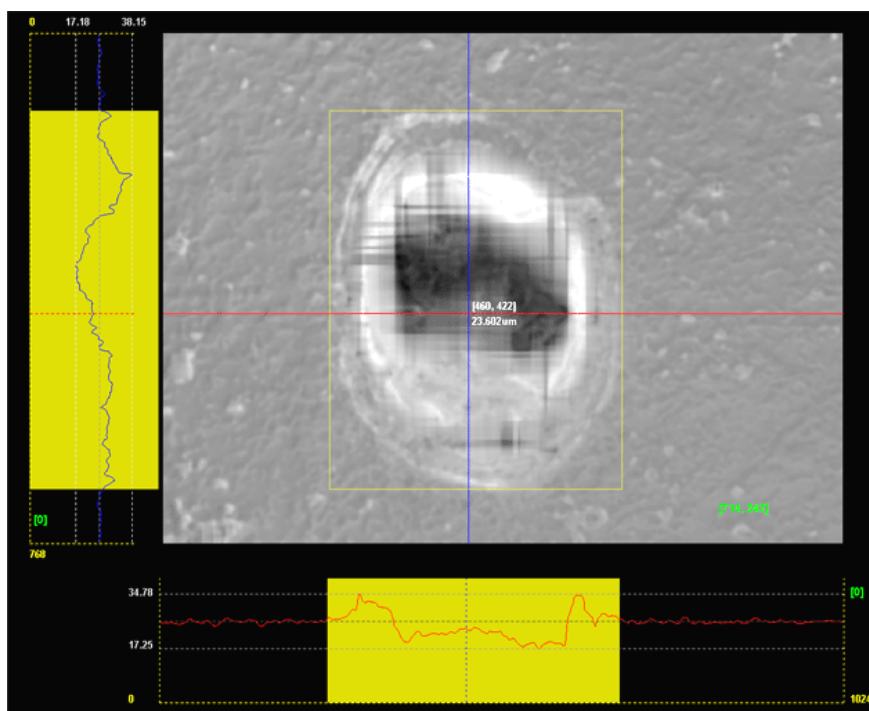


Fig. 45: The visualization part of the main window of the editor in the sample viewing mode. The detected pulse is bounded by the selection rectangle.

6.2 Simulation of Burning

This chapter introduces the created software tool and its functions which is used for the simulation of burning. The theoretical background, ideas and algorithms of the simulation are described in Chapter 4.

The pulses are created in the process of pulse extraction (the process of pulse extraction is described in Chapter 3). They are loaded to the system and they are used for the simulation of burning. Before or during the burning process the pulse can be modified or there can be loaded a new one. All modified pulses can be saved and used later or they can be used immediately after they completion without saving. For the pulse modification there was created a separate tool – the pulse editor. The editor can be launched

independently and any saved pulse can be processed in the editor. Also the actually loaded pulse can be modified and it can be used directly for the burning with its new shape.

The pulse editor offers several simple functions. It can load one pulse or several pulses together. In the case of several pulses, all pulses are loaded, centered and the average pulse is created. Its final width and height corresponds to maximum of all used files. The editor enables to rotate the pulse by 90° if needed. The level of the basic material level around the pulse can be changed globally or the eraser of selected size can be used for the pulse preparation by removing of small inaccuracies. The main window of the pulse editor is shown in Fig. 46, two pulses are loaded and small inaccuracies on the border are removed by the eraser.

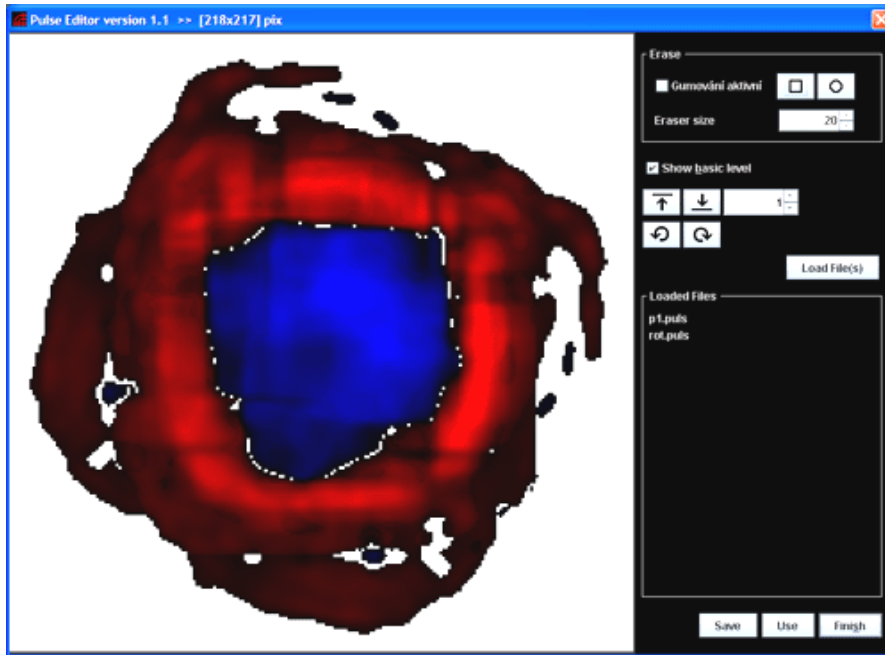


Fig. 46: The main window of the pulse editor – two pulses are loaded and small inaccuracies on the border are removed by the eraser.

For the simulation of the burning the pulse has to be loaded. After the pulse selection a target of the same size and shape as the loaded pulse is used instead of the cursor if the mouse moves above the sample. As the sample for burning an unburned material should be used. The unburned material surface can be generated as a new sample of selected material and given dimension or a real measured unburned material surface can be measured and loaded. If the material is selected and the pulse is loaded, the burning simulation can start. Three possible ways of burning can be used – pulses can be burned into one point or along a chosen trajectory or single pulses are burned manually.

If we want to burn several pulses into one point or along a trajectory, we have to choose the number of pulses to burn and the position of burning. For the case of burning into one point the coordinates of the point have to be set, in the case of trajectory there have to be determined end points of the line segment. The tool computes distance among single pulses automatically and at their positions are highlighted by the cross symbol. Such case is together with the target presenting the loaded pulse shown in Fig. 47. Because the laser burning process is not fully deterministic, the simulation system enables setting of the inaccuracy of burned pulses placing before the simulation starting. After starting the burning process the selected experiment is burned. The results of burning 8 pulses in

distance of 52 μm with inaccuracy 5% into cermet are shown in the form of top view and as the horizontal cross-section curve in Fig. 48.

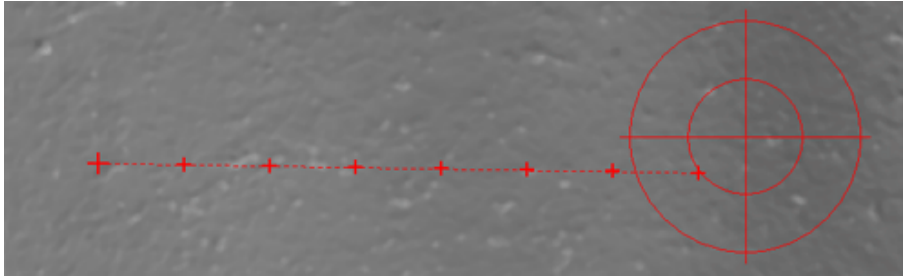


Fig. 47: The situation before 8 pulses burning along a line segment – the tool automatically computes distance among single pulses and shows crosses at their position. The target represents the loaded pulse.

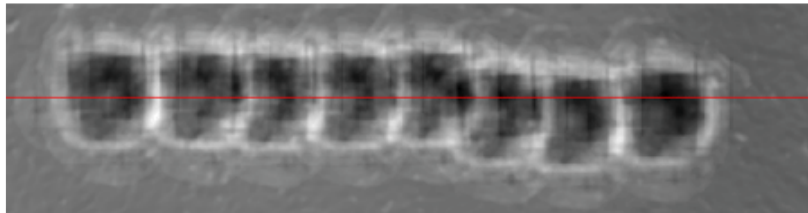


Fig. 48: The results of 8 pulses burning in distance 52 μm with the inaccuracy 10 μm into cermet in the top view and as the horizontal cross-section curve.

Nowadays, there can be used only line segments as a trajectory. In the next version of the simulation we plan to add other burning patterns (e.g. burning along any curves or burning of bitmap image).

Because the real burning process is not fully deterministic, it was required to bring soft inaccuracy into the simulation. The rate of inaccuracy is given by the percentage value which expresses the maximal possible difference of placement of the point on the sample surface. How higher the value is, so bigger is the random difference from the exact pulse placement. The final difference is randomly counted in the interval $\langle 0, \text{max difference value} \rangle$ for each direction.

In the case of any other testing of burning the single pulses can be placed on the sample surface manually only by using mouse. After the clicking the actually loaded pulse is applied in the position of the mouse cursor.

6.3 Result Comparison

This mode is used for the results exploring and comparing and for the system verification. It enables to compare any two samples and to determine the size of difference. Description of the system verification, its possibilities and problems are particularized in Chapter 5.

Before the comparing itself it is needed to recompute the basic level of both samples to the same value. During the samples comparison the MSE (Mean Square Error) is computed to give information about similarity of both surfaces. The computed value is written in the right control panel. The MSE is computed for the whole sample and for the area bordered by the selection rectangle. For the system verification both samples have to be compared in the position where burned pulses overlap over each other in the most precise way. The reasons and methods of samples overlapping are outlined in Chapter 5.2. In the created system both manual and automatic overlapping of samples is implemented. The manual sample shifting is provided simply by the mouse dragging.

Our tool enables comparing of two samples and to visualize them in three possible ways. The user can switch during results exploring among three ways of visualization – both compared samples together, the visualization of standard difference image or the logarithm operator adaptation of the difference image. The first visualization way shows both samples together in one window and enables to compare them directly. Each sample is highlighted by another color for better differentiation. The result of this visualization approach is shown in Fig. 49. The top view shows all parts of the sample which are higher than the surface of the second one; in the cross-section windows both cross-section curves are shown. The blue pixels in the top view show points where the simulated sample is higher than the real one. The cross-section curves of the simulated sample are red and blue, for the real sample the yellow and green colors were chosen.

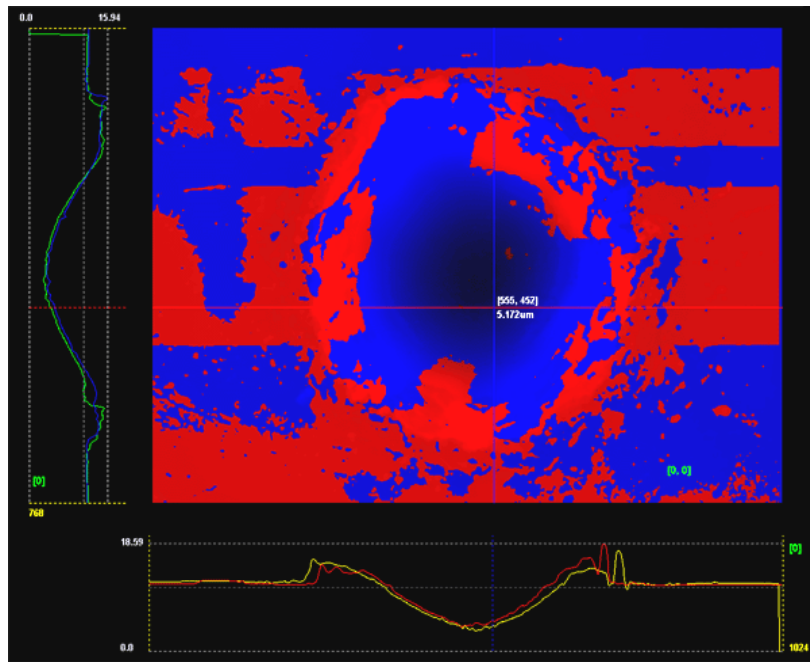


Fig. 49: Comparison of two samples visualized in the way of parallel visualization of both samples. Red and blue cross-section belong to the blue sample in the top view, yellow and green cross-section to the red sample.

The second way of visualization of comparison is the standard difference image. The computation of the difference image is described in Chapter 5.1.2. The result of comparison of the similar samples can be seen in Fig. 50. From standard difference image the distribution of height differences is visible well, but it is not easy to distinguish small differences, because they are displayed with the very similar or even the same color. These

disadvantages are removed in the logarithm operator adaptation of the difference image. Its computation is outlined in Chapter 5.1.3. As can be seen in Fig. 51, we can better find out small differences in the neighbourhood of the material level, but on the other hand we cannot see the difference distribution. Because each way of comparison visualization brings several advantages and disadvantages, the implemented tool offers the possibility of choice.

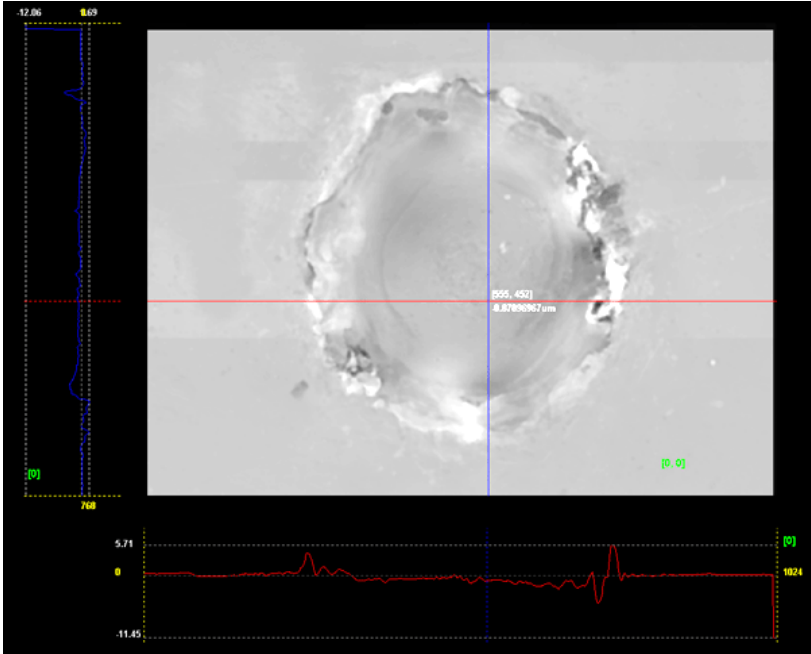


Fig. 50: Comparison of two samples visualized as a standard difference image.

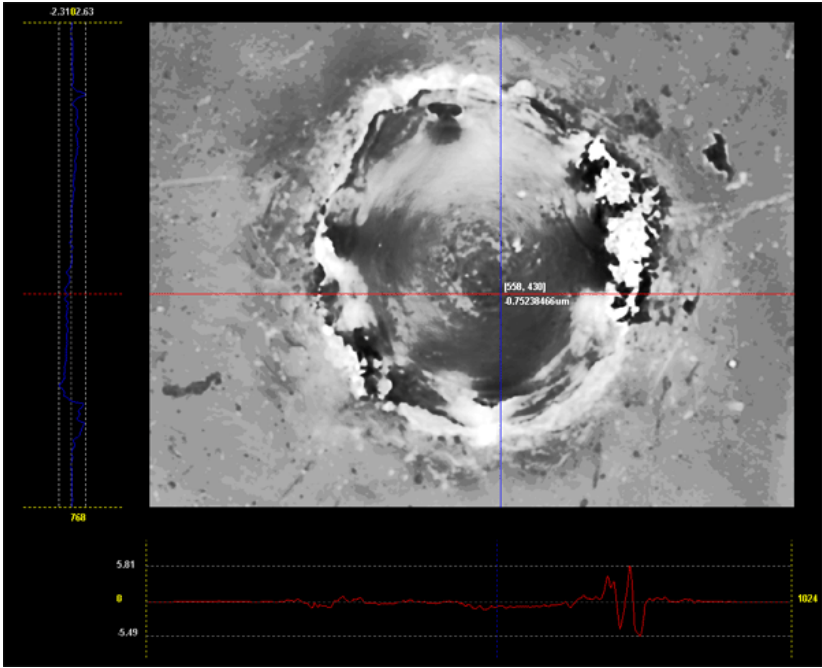


Fig. 51: Comparison of two samples visualized as the logarithm operator adaptation of the difference image.

7 Future Plans

Our future plans are divided into several groups in dependence to which activity it is related to. All described plans concern the part we are working on – data processing, software simulation and result visualization. The other (especially hardware) parts of the project have of course their own individual plans for the future.

7.1 Simulation Improvement

One of our most important aims is to improve the simulation to get as realistic results as possible. This task has several sides we have to make better to get the maximal accuracy of the simulation. Some of the tasks are connected with automation and program self-activity which are described in Chapter 7.2.

As it was written above gaining data from real experiment is time and money demanding and that is why we want to minimize the amount of the real experiment required for satisfactory system setting and verification. It is important to define minimal required experiment for the most often used materials and also to determine requirements and parameters for experiment design if any new material will be used.

The problems that were already mentioned in Chapter 4.2 are related to including the material heating into the simulation and to including the starting and finishing laser motion inaccuracies to the simulation (the inaccuracies caused by the laser motion starting were shown in Fig. 30a). This problem influences especially the simulation of burning pulses along a trajectory. Both problems are related to the used mechanic parts of laser which are not currently specified and will be solved in the future.

7.2 Automation and Program Self-activity

The whole system should work as automatic as possible. That is why all partial tasks of data processing should be self-acting in a maximal way. Of course, the reliability and speed of the system must be preserved despite the automation.

At first, before the simulation is provided we have to get input data. The basic level of the material has to be determined and then pulses are automatically extracted from the samples. All pulses from the experiment should be executed automatically at once. That is why a really precise method for automatic pulse detection must be used. Possible ways and its results are described Chapter 3.2, also several perspective approaches are outlined. We plan to test all methods at different types of samples and finally we will choose the best one.

Next task belongs partly to the previous chapter about the simulation itself. The problem is choosing the optimal combination of pulses for burning of given sample. The question like if the sample of 15 pulses can better achieved by combination of three samples with 5 pulses or by using sample with 10 and after that with 5 pulses (the topic was

outlined in Chapter 4.2) should be answered after more tests and result verifications. We also want to take advantage of all significant relations among single samples in the experiment we will be able to find.

Finally, in order to verify the system, the simulated sample must be compared with the real one. Because pulses are not placed at the same place in the sample (as it was described in Chapter 5), it is necessary to overlay pulses over each other in the most accurate way. This operation is very time demanding, we have to find any suitable algorithm which would quickly and exactly overlay both compared samples and determine similarity rate of the real and simulation burned sample.

7.3 The Ways of Pulse Description

As a next stage of the simulation there is a question if it is really necessary to use whole height map of pulses for its representation or if it is possible to describe it by using several parameters and to integrate features of randomization. Our idea for possible simplification of description comes out from the closer examinations of the samples of similar experiments. In Fig. 52a-b two samples of 50 pulses burned into steel and measured separately can be seen. In Fig. 52c difference image of overlapped samples is visible and in Fig. 52d-e the vertical and horizontal cross-section curves of difference image are shown.

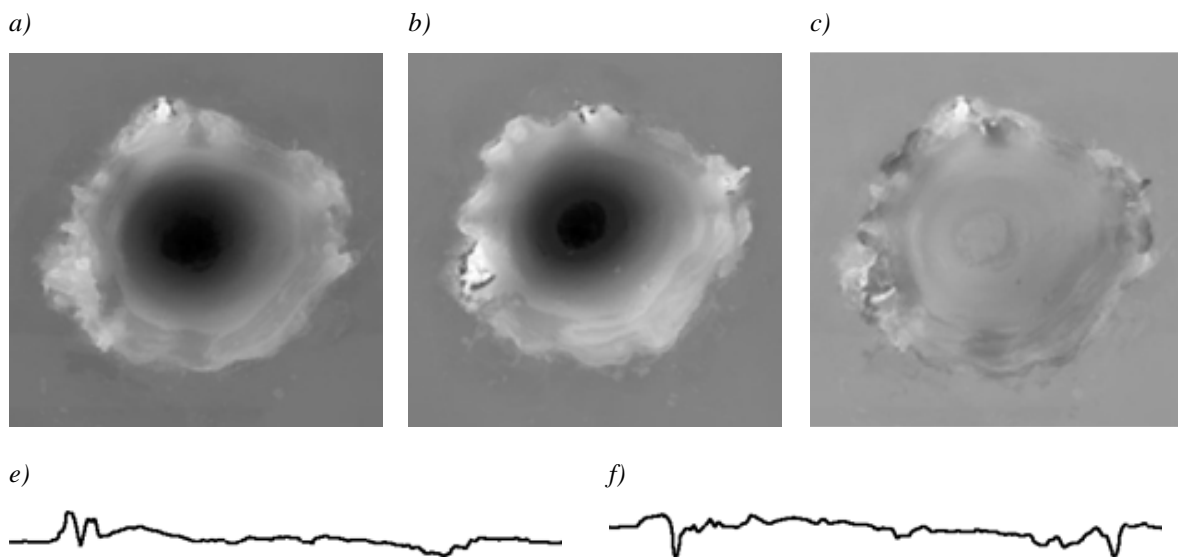


Fig. 52: a, b) Two samples of 50 pulses burned into steel and measured separately. c) Difference image, d, e) vertical and horizontal cross-section curves of the difference image.

From this concrete sample it is visible, that the shapes of similar pulses seem to differ minimally in the area of vaporized material (it means that the shapes of the hollows are basically the same). The most important divergences appear in the area of melted material around the pulse. That is why we presuppose that the similar parts of the pulses (hollows) could be parametrically described and for generating of the pulse border random surface modification could be used.

However, at this level of understanding we are not so far absolutely sure to be able to decide, if all samples can be described by this way and if it does not mean a too high distortion of processed pulses. It is next task for future examination and testing.

7.4 Others

Of course, there are other things and problems we will have to solve and it is not necessary to create a special group for them.

One of these problems relates to size of processed sample. As it is written in Chapter 2, data are processed in a high resolution. Sample of real size $256 \times 192 \mu\text{m}$ is described by 1024×768 height map. The real sample of this size is hardly visible. But let's imagine sample with size of several millimeters or even centimeters. The amount of values needed for such sample is far larger than it would be possible to process in the real time. So far we are working with comparatively small data, but in the future the necessity of big data processing will probably growth. That is why we will have to use either smaller resolution of the simulation or any algorithm for level of detail height maps processing [Lueb].

8 Conclusion

After one year of preparation the project works is in the phase of partial completion. We have prepared a tool for data preprocessing and groundwork for simulation tool. It enables us to provide various experiments. For this purpose we dispose with data sets from experiments burned and measured for two materials – cermet and steel. For both materials we obtained basic experiments, where each type of sample was burned and measured once. For steel we dispose moreover with data from extended experiment, where each type of sample was burned and measured several times. Unfortunately, the project is hindered by the lack of further real samples which are necessary for the testing and verification.

At this state we are able to experiment, verify and explore results. As it is described in the chapter 7.2, we plan to maximize the self-activity of the tool by finding suitable methods for image processing. Several methods for automatic pulse detection and system verification have been already designed, implemented and tested. The subsequent research is already realized in cooperation with the Department of Mathematics.

Of course, the simulator should be expanded. Functions for burning continuous surface of defined area, burning along any curve or collective defining of experiments for batch-oriented burning should be worked into the system. Also a tool for the complete experiment description is planned to be used. The described experiment will be used as the groundwork for the simulation of burning.

Literature

- [Alex] Alexiades, V., Solomon, A. D.: *Mathematical modelling of melting and freezing processes*. ISBN 1-56032-125-3. Taylor&Francis, 1993.
- [Blell] Blesloch, E. G.: *Introduction to Data Compression*, 2001. [online], [cit. 2008-03-28]. <<http://www.cs.cmu.edu/afs/cs/project/pscico-guyb/realworld/www/compression.pdf>>
- [Brigh] Brigham, E. O.: *The fast Fourier transform and its applications*, ISBN 0-13-307505-2. Englewood Cliffs, N.J.: Prentice Hall, 1988.
- [Háj] Hájková, J.: *Methods of Pulse Detection in Laser Simulation*. Submitted to the 3rd International Conference on Software and Data Technologies ICISOFT 2008. Portugal.
- [Háj&Her] Hájková, J., Herout, P.: *Laser Simulation*, Proceedings of the 7th International Conference APLIMAT 2008, ISBN: 978-80-89313-03-7, pp 823-828. Bratislava, 2008.
- [Hlav] Hlaváč, V., Šonka, M., Boyle, R.: *Image Processing and Machine Vision*, ISBN 0412455706. Chapman & Hall Computing, London, 1993.
- [Hlav&Sedl] Hlaváč, V., Sedláček, M.: *Zpracování signálů a obrazů*, ISBN 80-01-02114-9. Vydavatelství ČVUT. Praha, 2001.
- [Lueb] Luebke, D., Reddy, M., Cohen, J. D., Varshney, A., Watson, B., Huebner, R.: *Level of Detail for 3D Graphics*, ISBN 1-55860-838-9. The Morgan Kaufmann Series in Computer Graphic, Hardcover, 2002.
- [Smith] Smith, I. L.: A Tutorial on Principal Component Analysis, 2002. [online], [2008-03-20]. <http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf>
- [Nishi] Nishiguchi, H., Imiya, A., Sakai, T.: *Laplacian Based Non-Linear Diffusion Filtering*. Proceedings of 18th International Conference on Pattern Recognition ICPR 2006. 2006.
- [wikiCer] Wikipedia – Cermet [online], [cit. 2008-02-24]. <<http://en.wikipedia.org/wiki/Cermet>>
- [wikiCoM] Wikipedia – Center of Mass [online], [cit. 2008-03-19]. <http://en.wikipedia.org/wiki/Center_of_mass>, <<http://cs.wikipedia.org/wiki/Těžiště>>
- [wikiSoR] Wikipedia – Surface of Revolution [online], [cit. 2008-03-05]. <http://en.wikipedia.org/wiki/Surface_of_revolution>

- [wOGL] OpenGL [online], [cit. 2007-09-01]. <<http://www.opengl.org/>>.
- [wJOGL] Java OpenGL [online], [cit. 2007-04-21]. <<https://jogl.dev.java.net/>>.
- [wLin] Lintech Homepage [online], [cit. 2008-02-21]. <<http://www.lintech.cz/en/>>
- [wLogOp] Logarithm Operator [online], [2008-03-13].
<<http://homepages.inf.ed.ac.uk/rbf/HIPR2/pixlog.htm>>
- [wOlym] Olympus LEXT OLS3100 [online], [cit. 2008-02-21].
<<http://www.olympusmicroimaging.com/index.cfm/page/products.index.cfm/cid/1085/navid/206/parentid/182>>
- [wSHT] Laser BLS-100 (Software – Hardware Technik) [online], [cit. 2008-02-21].
<<http://www.sht-gmbh.de/webeng/530716975607ad305/5307169756083de22/5307169756084252b/index.html>>
- [Wri] Wright, R.S., Lipchak, B.: *OpenGL SuperBible (3rd Edition)*, ISBN 0672326019. Sams, 2004.
- [wWM] Wolfram Mathematica® [online], [cit. 2008-03-20].
<<http://www.wolfram.com/>>
- [Zapl] Zapletal, J.: *Application of radial Basic Functions*. Diploma Thesis, University of West Bohemia, Pilsen, 2007.
- [Zhang] Zhang, J., Yan, Y., Lades, M.: *Face Recognition: Eigenfaces, Elastic Matching, and Neural Nets*. Proceedings of the IEEE, Vol. 85, Issue 9, September 1997.
- [Žára] Žára, J., Beneš B., Felkl P.: *Moderní počítačová grafika*, ISBN 80-251-0454-0. Computer Press, 2005.