IOP Conf. Series: Materials Science and Engineering **209** (2017) 012067 doi:10.1088/1757-899X/209/1/012067

Designing Pattern Recognition-Based Method for Fast Visual Inspection of the Bucket Wheel Excavator Lattice Structure

M Risteiu[1*], A Lorincz[2], R Dobra[1], P Dasic[3], I Andras[4] and M Roventa[4]

[1] Computer Science and Engineering Department, "1 Decembrie 1918" University of Alba Iulia, Gabriel Bethlen Str. No.5, 510009 Alba Iulia, Romania

[2] Automation Department, University of Petrosani, University Street, No. 11, Petrosani Romania

[3] High Technical Mechanical School of Professional Studies, Str. Radoja Krstića 19, 37240 Trstenik, Serbia

[4] Machinery Department, University of Petrosani, University Street, No. 11, Petrosani Romania

E-mail: mristeiu@uab.ro

Abstract. The proposed paper shows some experimental results of a research in metallic structures inspection by using a high definition camera controller by high processing capabilities. The dedicated ARM Cortex-M4 initializes the ARM Cortex-M0 system for image acquiring. Then, by programming options, we are action for patterns (abnormal situations like metal cracks, or discontinuities) types and tuning, for enabling overexposure highlighting and adjusting camera brightness/exposure, to adjust minimum brightness, and to adjust the pattern's teach threshold. The proposed system has been tested in normal lighting conditions from the typical site.

1. Introduction

The investigation of the bucket wheel excavator health is one of the most important activities into a complex industry like energy production from coal. Everyone can understand this when the full chain is analyzed.

The mined coal must be correlated with the energy demand and the energetic content of the coal. This coal is then transported by conveyer belts and/or rail to the main deposit of the power plant. The amount of the coal in this deposit must be correlated with the demanded coal to be fired, and the capacity of the coal to solidify under its own weight. So, again, the mass flow of the coal is influenced by the excavator production [1,5].

When the coal fired thermal power plant works as a part of the National Energetic Dispatching System, any deviations from the energetic delivery contract (quantity and quality of the energy) are penalized. These penalties are influencing the entire energetic chain- including the excavator.

The bucket wheel excavator is a structure complicated from mechanical and electrical point of view. Main parts of the excavator are mechanical connected each other through lattice frames and supporting cables (figure 1).

Figure 1. A bucket wheel excavator in the marching position.

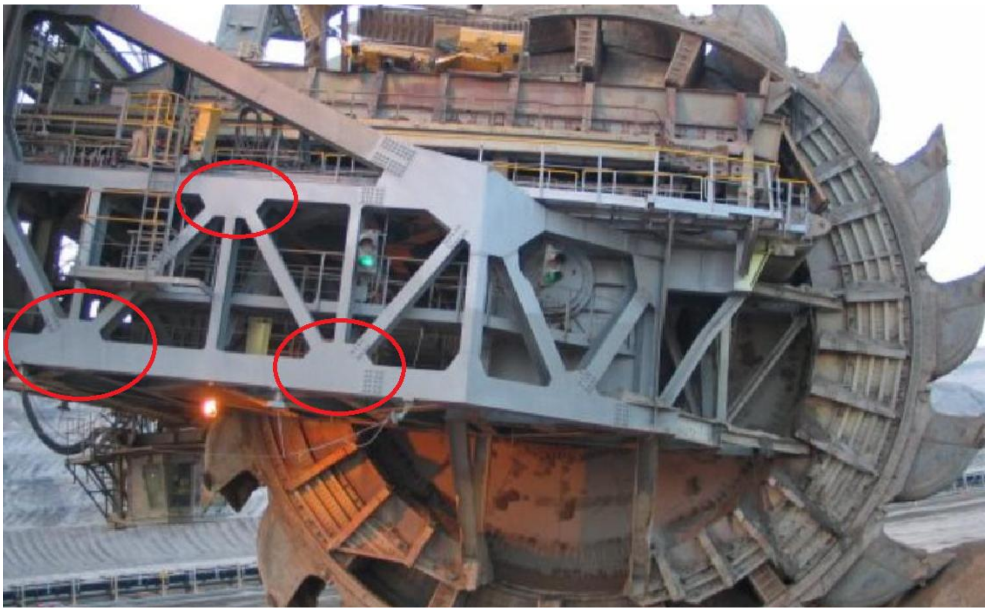

Figure 2. The lattices are sustaining entire structure. Some of them are welded.

As it can be seen, the lattice structure transmits the mechanical efforts in entire structure, when there is an excitation. Along its structure, the lattice is stressed differently in different points of suspension.

The purpose of this paper is to investigate in the fast way the health of the structure by implementing a pattern recognition method over pictures taken from the focused parts of the

IOP Conf. Series: Materials Science and Engineering **209** (2017) 012067 doi:10.1088/1757-899X/209/1/012067

excavator. We use a high computational system (built in around a Single Board Computer- SBC) with camera that is processing the taken pictures for a fast diagnose, after some learning and calibration process. In this paper we are focused on electrical wended mechanical assembles- that are parts of excavator lattices.

2. Infrastructure and Methods

For the fast inspection of the stress in lattice structures, we are using hardware adequate: SBC- Raspberry Pi Model B 756-8308 Motherboard, a microcontroller- based camera, packed as an embedded system- Pixy.

The major advantage of the SBC is related to its processing power capacity (CPU is an ARM 1176JZF, 700MHz- RISC, RAM- 512MB, many communication ports- video, USB, WiFi), together is an important interface with external processes [9].

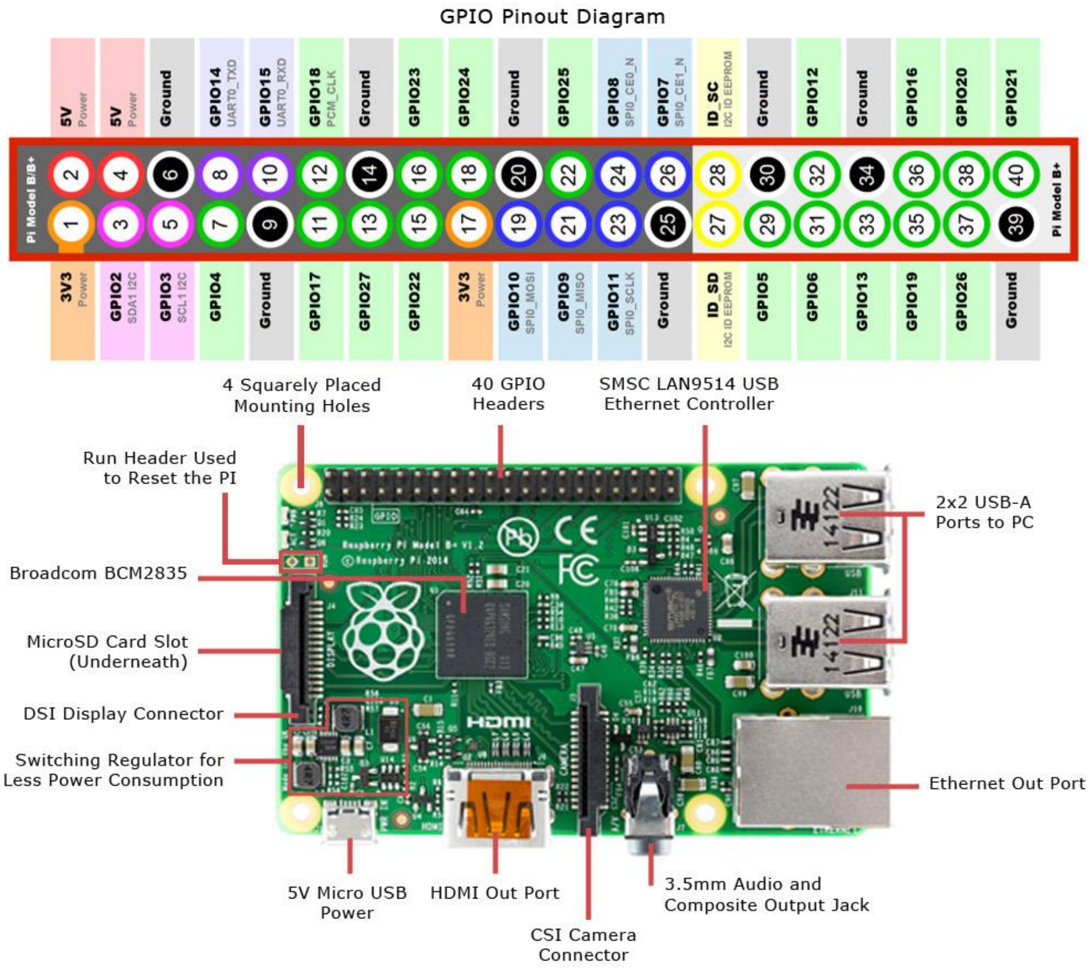

Figure 3. From its reference guide- the full GPIO map [8].

The inspection camera usually is designed around a high density CMOS image sensor. When it is integrated into an embedded system, the output of it is already a high quality processing result. The Pixy works with an NXP microcontroller: an M4 cortex at a frequency of 204MHz and a M0 cortex all at the frequency of 204 MHz, are quite powerful, running on 32 bits both.

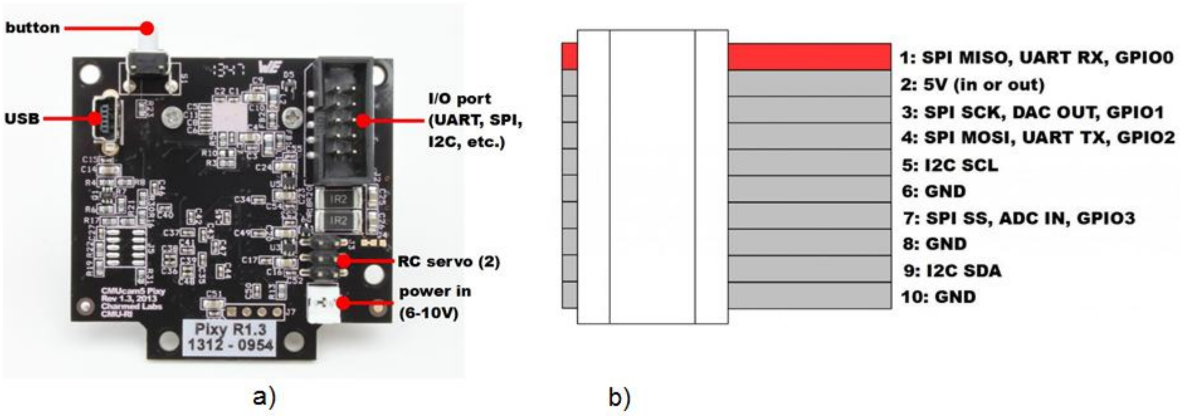

Figure 4. The camera-based embedded system proposed for fast inspection- product view (a), pinout map (b) [3].

The embedded system with its memory 1MB (flash), where it stores its code, protocol, patterns offers some special features, and in the same time some constrains like:

- Preferable patterns should be with a color code (it cannot really distinguish the shades of gray). Theoretically through the lens it has, it may even focus very well.
- Creating some contrast through thermal imaging rooms, the higher the temperature on a body, the color is of intense yellow and the low temperature is blue.
- With memorized patterns, it creates blocks, in a significant way.

The sensor used is a CMOS WXGA (1-megapixel) HD Sensor with OmniPixel3-HS Technology OV9715. It is able to ensure 720p HD video performance at 30 frames per second (fps), with high sensitivity - 3 x 3 µm pixels and low-light sensitivity of 3.3 V/lux-sec. The OV9715 provides imaging in virtually every lighting condition from bright daylight to nearly complete darkness.

OV9715 sensor provides full-frame, sub-sampled or windowed 8-bit/10-bit images in RAW RGB format via the digital video port and with complete user control over image quality, formatting and output data transfer. The OV9715 offers a chief ray angle (CRA) of 0°

2.1. Fast inspection of the welding stress considerations

Let's analyze next figure form the quality of the welding point of view.

The most important parameters that are taken into account for quality of the welding analysis are [9]:

- Distribution of the weld material;
- The quantity of the waste (the slag after cooling);
- Porosity of the weld surface;
- Tightness should be constant and with a constant curve texture;
- Strength of the welding is required by design

All this parameters are influenced, during the arch life, by the length of the arch, by the speed of the welding head, and by the arch current [9]. The proposes three measurable parameters: the 3D trace of the weld (a); the quality of welding (eventually the speed of the weld generation) (b); the thickness of the welding cord (c).

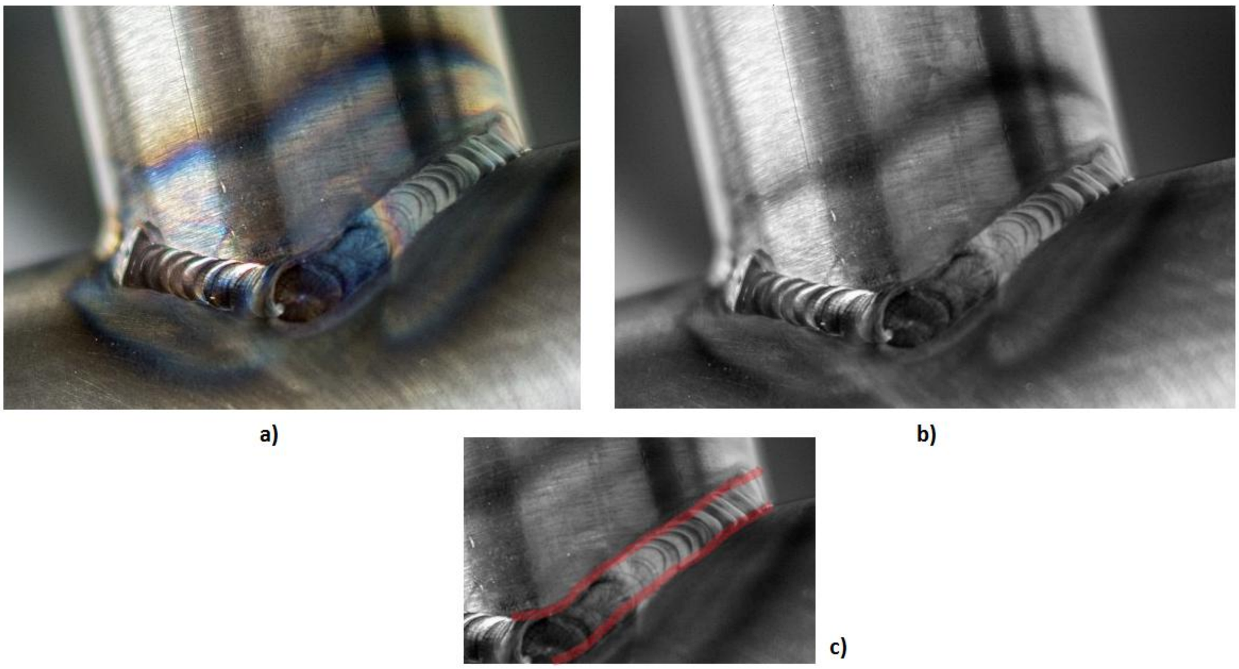

Figure 5. The evaluation of the quality of welding in a 3D exposal of the metallic assembles.

The first idea of visual inspection has started with a Matlab function designed for extracting contours, then evaluating the welding contour. The typical function is [2]:

figure;

imcontour(I,3)

In order to obtain focused contour (expected continuous contour) (figured in c) part of the figure), some extra processing are required (defining patterns according with many aspects- textures, positions). These actions are difficult to setup.

Another meet tool is developed by OpenGL programming environment [3,4,7]. We propose to analyze the *matchTemplate* function. They are developed in C++, Python. The typical pseudocode is shown for different programming environments:

C++: void matchTemplate(InputArray image, InputArray templ, OutputArray result, int method)

Python: cv2.matchTemplate(image, templ, method[, result]) → result

C: void cvMatchTemplate(const CvArr image, const CvArr* templ, CvArr* result, int method)*

Python: cv.MatchTemplate(image, templ, result, method) → None

The environment programming parameters are:

image – Image where the search is running. It must be 8-bit or 32-bit floating-point.

templ – Searched template. It must be not greater than the source image and have the same data type.

result – Map of comparison results. It must be single-channel 32-bit floating-point. If image is W \times H and templ is w \times h , then result is (W-w+1) \times (H-h+1) .

method – Parameter specifying the comparison method (see below).

The function slides through image, compares the overlapped patches of size w \times h against templ using the specified method and stores the comparison results in result. Here are the formulae for the available comparison methods (I denotes image, T template, R result). The summation is done over template and/or the image patch: x' = 0...w-1, y' = 0...h-1 [4,7].

What does mean *templ*? It means defining a template for the acquired figure, template that covers many uncontrolled conditions.

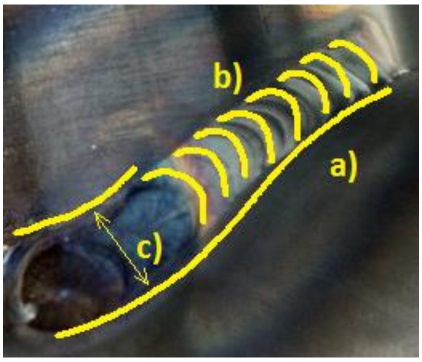

Figure 6. The defined characteristics that compose the *templ* parameter.

These parameters consist in branch of paths that are analyzed in an acquired picture inspection. These parameters are not influenced by position, contrast, and noise.

The literature offers more methods to calculate matches [4,7]. For our experiment we have used the basic method: *method=CV_TM_SQDIFF*. It uses next formula:

$$R(x,y)=\sum_{x',y'}(T(x',y')-I(x+x',y+y'))^2$$

After the function finishes the comparison, the best matches can be found as global minimums (when *CV_TM_SQDIFF* was used) or maximums (when *CV_TM_CCORR*) using the *minMaxLoc()* function.

$$R(x,y)=\sum_{x',y'}(T(x',y')I(x+x',y+y'))^2$$

In case of a color image, template summation in the numerator and each sum in the denominator is done over all of the channels and separate mean values are used for each channel (Red, Green, Blue, Hue). That is, the function can take a color template and a color image. The result will still be a single-channel image, which is easier to analyze.

2.2. Implementation pattern recognition using OpenGL and C++ for fast inspection of the welding cordon

The implementation uses two steps:

- extract focused contour;
- matches the processed contours with the templates with technical meanings.

Extracting contours of images is done through some typical steps:

```
/// Load source image and convert it to gray
src = imread( argv[1], 1 );
/// Convert image to gray and blur it
cvtColor( src, src_gray, CV_BGR2GRAY );
blur( src_gray, src_gray, Size(3,3) );
/// Create Window
char* source_window = "Source";
namedWindow( source_window, CV_WINDOW_AUTOSIZE );
imshow( source_window, src );
createTrackbar( " Canny thresh:", "Source", &thresh, max_thresh, thresh_callback );
thresh_callback( 0, 0 );
Mat canny_output;
vector<vector<Point> > contours;
```

```
vector<Vec4i> hierarchy;
/// Detect edges using canny
Canny( src_gray, canny_output, thresh, thresh*2, 3 );
/// Find contours
findContours( canny_output, contours, hierarchy, CV_RETR_TREE, CV_CHAIN_APPROX_SIMPLE, Point(0, 0) );
/// Draw contours
Mat drawing = Mat::zeros( canny_output.size(), CV_8UC3 );
for( int i = 0; i< contours.size(); i++ )
  {
    Scalar color = Scalar( rng.uniform(0, 255), rng.uniform(0,255), rng.uniform(0,255) );
    drawContours( drawing, contours, i, color, 2, 8, hierarchy, 0, Point() );
  }
```

For the second part of this implementation, the software is doing some actions:

- Loads an input image and a image patch (template);
- Perform a template matching procedure by using the OpenGL function matchTemplate with any of the method described before.
- Normalize the output of the matching procedure;
- Localize the location with higher matching probability;
- Draw a rectangle around the area corresponding to the highest match.

The typical pseudocode used is shown next:

```
using namespace std;
using namespace cv;
/// Global Variables
Mat img; Mat templ; Mat result;
char* image_window = "sudura_viteza_mare";
char* result_window = "sudura viteza mare window";
int match_method;
int max_Trackbar = 5;
/// Function Headers
void MatchingMethod( int, void* );
```

Then:

```
/// Do the Matching and Normalize
  matchTemplate( img, templ, result, match_method );
  normalize( result, result, 0, 1, NORM_MINMAX, -1, Mat() );
  /// Localizing the best match with minMaxLoc
  double minVal; double maxVal; Point minLoc; Point maxLoc;
  Point matchLoc;
  minMaxLoc( result, &minVal, &maxVal, &minLoc, &maxLoc, Mat() );
  /// For SQDIFF and SQDIFF_NORMED, the best matches are lower values. For all the other methods, the higher the better
  if( match_method  == CV_TM_SQDIFF || match_method == CV_TM_SQDIFF_NORMED )
    { matchLoc = minLoc; }
  else
    { matchLoc = maxLoc; }

  /// Show me what you got
  rectangle( img_display, matchLoc, Point( matchLoc.x + templ.cols , matchLoc.y + templ.rows ), Scalar::all(0), 2, 8, 0 );
  rectangle( result, matchLoc, Point( matchLoc.x + templ.cols , matchLoc.y + templ.rows ), Scalar::all(0), 2, 8, 0 );
```

```
imshow( image_window, img_display );
imshow( result_window, result );
```

Then, the technical inspection means to evaluate/ report the results.

3. Results and discussions

The preliminary tests have been done first to train our system when different welding regimes has been established:

3.1. Testing the length of the welding

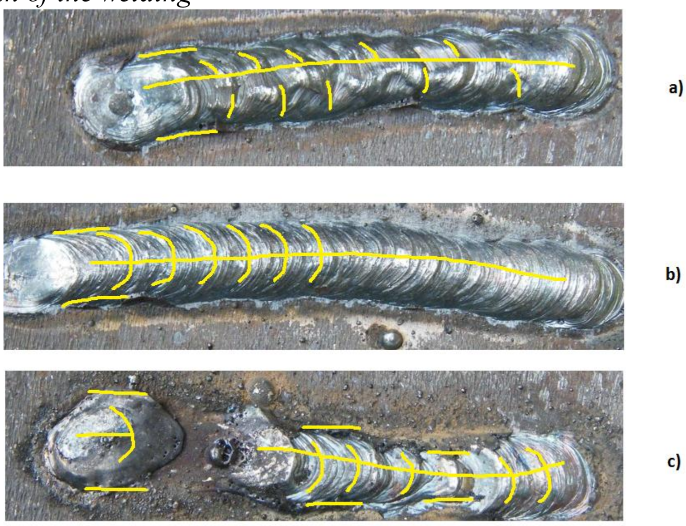

Figure 7. The patterns recognized by the *templ* parameter, according with arch length.

By running different tests (position, light, distance, zooming options), the length of the welding arch defines different patterns:

- the continuity of the weld is disturbed in short arch length (a);
- the trace and thickness of the weld is distorted (c).

3.2. Testing the speed of the welding head

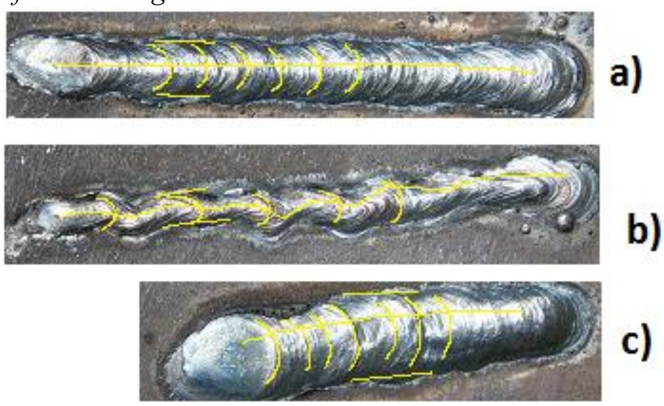

Figure 8. The patterns recognized by the *templ* parameter, according with the speed of the welding head.

By running different tests (position, light, distance, zooming options), the length of the welding arch defines different patterns:

- the thickness and the trace of the weld is disturbed(b);
- the thickness and the continuity of the weld is distorted (c).

3.3. Testing the current through the welding arch

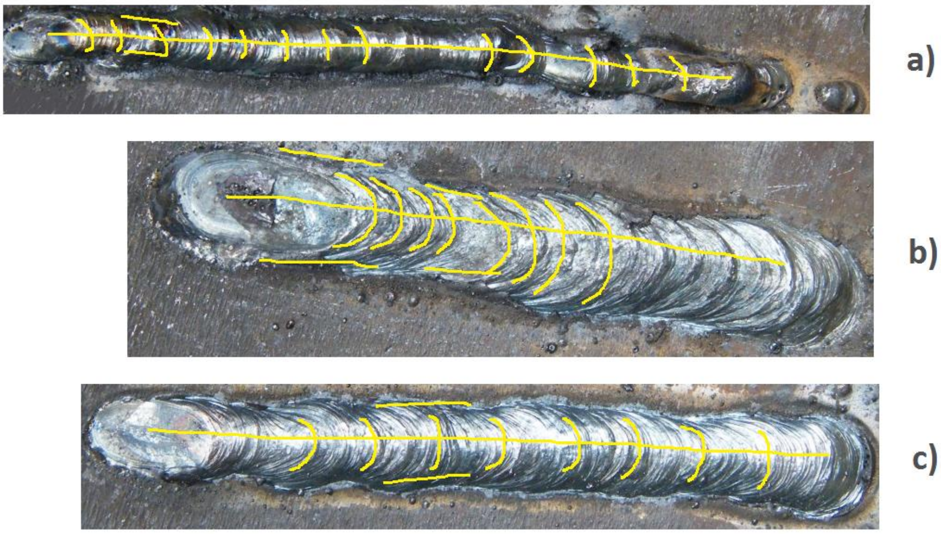

Figure 9. The patterns recognized by the *templ* parameter, according with the current of the welding arch.

By running different tests (position, light, distance, zooming options), the length of the welding arch defines different patterns:

- the thickness and the trace of the weld is not constant (b);
- the thickness is not (c).

Then, by combining different pictures we have been required to adjust to zooming facility in order to create detect the cause of some faults in different weld. This is required in order to have a calibration database for calibrating the system for weakness inspection during the in site testing.

4. Conclusions

In this paper we have investigated different types of welds cordons according with welding regime. The tests have been done based on patterns implemented in high computing systems.

The investigation camera is powerful, with acceptable results (89% of results' success).

We consider that an important improvement (increasing the results' success) can be done by:

- Investigating, in a continuous weld contour a weakness area like a cleave
- The inspection will be done by adding zooming facility (increasing the density of the acquired dots into one pictures),
- By adding light contrast to the picture by illuminating with laser source.

References

[1] Bishop C M 2014 *Pattern Recognition and Machine Learning 3rd Edition* ISBN-10: 0-387-31073-8

[2] Stenberg E L T 2011 Quality inspection and fatigue assessment of welded structures in cooperation with volvo construction equipment AB *Master of Science Thesis in Lightweight Structures Dept of Aeronautical and Vehicle Engineering Stockholm, Sweden*

[3] Theodoridis S 2010 *Introduction in Pattern Recognition in Matlab*, ISBN 978-0-12-374486-9

[4] *** http://cmucam.org/projects/cmucam5/wiki/Porting_Guide 2017
[5] *** Edge Detection and Image Overlay with Impaired Frame 2017
[6] *** Move into the future with reliable measurement 2017 *KYOWA technical report 2016*
[7] *** Kyowa strain gauges application note 5
[8] *** OpenCV, pattern matching, ttp://docs.opencv.org/2.4/doc/tutorials/imgproc/ histograms/template_matching/template_matching.html
[9] *** https://raspberrypi.stackexchange.com/questions/40318/raspberry-pi-2-can-gpio-pins-29-40-be-used-gpio-gen-input-output-configurable-in

Acknowledgments
The paper is based on research activity that is the subject of the invention brevet Installation for in-situ measuring and recording the forces acting upon the teeth of the cutting element in rock shattering machines, Patent number RO 123483 B1

The paper is a part of an European Union research project founded by the Coal and Steel Founding, RFCS agreement No. RFCR-CT-2015-00003 (BEWEXMIN)