Comparative Analysis of Image Threshold Algorithms for VHDL Real Time Applications

Nikolay Nenov[1], Todor Djamiykov[2], Volker Zerbe[3], Dimitar Alexiev[4]

Abstract – **This paper presents comparative analysis of various threshold methods. After the analysis, the right algorithm is selected. This algorithm is appropriate for real-time applications and implementation in FPGA. It is used in an optoelectronic system for calculation of laser spot position. This system uses CMOS image sensor, which brings additional restrictions to choose a proper algorithm.**

Keywords - **Grayscale Thresholding, FPGA, Real-time Applications, CMOS image sensor.**

I. Introduction

Optoelectronic systems are widely used in industry. The goal of the described comparative analysis is to choose the proper threshold algorithm. This algorithm will be used in optoelectronic system for laser spot position measurement. The basic application of this system is: Measurement of decentre of lens and the bend of rails. The block diagram of this system is shown if Figure 1.

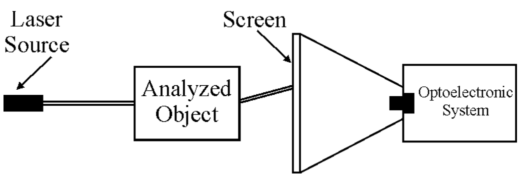


Fig.1 Block diagram of the optoelectronic system

Laser beam is used for determination of etalon direction. The beam is passed though the object and the measured parameter provokes deflection of the beam proportional to the measured value. The laser beam is projected at the transparence diffuse screen of the system and it is viewed like light spot. On the other side of this screen there is a camera of the optoelectronic system, which scans the screen with its CMOS image sensor. The output from the CMOS image sensor is processed by FPGA.

The Grayscale Image Threshold algorithm is the first step of series of calculations in the optoelectronic system for measurement of laser spot position. For the proper determination of its coordinates the spot must be separated from the background.

[1]Nikolay Nenov is with the Technical University of Sofia - Bulgaria, Faculty of Electronic Engineering and Technology

[2]Todor Djamiykov is with the Technical University of Sofia - Bulgaria, Faculty of Electronic Engineering and Technology

[3]Volker Zerbe is with the Technical University of Ilmenau - Germany, Faculty of Electronic Engineering and Technology

[4]Dimitar Alexiev is with the Technical University of Sofia - Bulgaria, Faculty of Electronic Engineering and Technology

After that the system makes calculations with all pixels belonging to the object, in order to calculate centre of mass (COM). The COM coordinates are coordinates of laser spot. The Decision of this task is connected with specific problems.

II. Problem Statement

In image processing applications the gray level values assigned to an object are different from gray level values of the background [1]. The pixel which belongs to the laser spot has grater intensity than background. The output of a threshold process is a binary image which is obtained by assigning pixels with values less than the threshold with zero and the remaining pixels with ones.

Let us consider that image **ε** is a digital image from the CMOS sensor, with M rows and N columns and L gray levels for each pixel. The pixel with coordinates (i,j) is denoted by **ε**(i,j). The threshold level T, is a value in the range $T \in (0,L-1)$. Let b(i,j) is a binary image and each pixel from b corresponds to the pixel from **ε** [1]. The threshold process is defined by Equation (1):

$$b(i,j) = \begin{cases} 0, \varepsilon(i,j) < T \\ 1, \varepsilon(i,j) \geq T \end{cases} \quad (1)$$

For the proper calculation of the coordinates of the object, an important factor is the precise definition of T.

The ideal laser spot is a circle with Gaussian distribution of energy. There are various effects which deform the laser spot. The photons from the laser beam have great value of intensity and they can saturate some pixels in the CMOS sensor. Because of internal effects in laser source and optical distortion, the spot may not be a circle and don't have Gaussian distribution of energy. The angle, in which laser beam gets the screen, also distorts the spot. Calculation of the centre of mass (COM) uses coordinates and the intensity of each pixel belongs to the object. For this reason the result of the threshold process is not a binary image – see Equation (2).

$$g(i,j) = \begin{cases} 0, \varepsilon(i,j) < T \\ \varepsilon(i,j), \varepsilon(i,j) \geq T \end{cases} \quad (2)$$

Used CMOS image sensor has VGA resolution. The laser spot is consisting of 100 to 500 pixels. This is less than 0.2% of the entire image. The laser spot is quite little and this makes the task for its separation hard. Figure 2 shows a real image from the laser beam over a screen of the optoelectronic system.

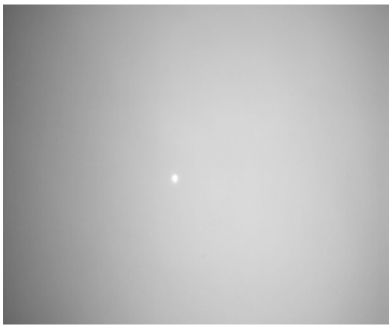


Fig. 2 Image of the laser spot over the screen of the system

III. Algorithm Survey

Because of the mentioned special features of the laser beam not all of the threshold algorithms can be used. The choice of right threshold algorithm has to be made according to the following criteria:

1. Possibility to work in real time – simple and fast algorithm, simple optoelectronic system, minimal delays, ability for work without memory. Used algorithm has to be fast and simple.
2. Possibility for work with CMOS image sensor – The information is obtained from the sensor pixel by pixel and row by row. Algorithm doesn't have to use neighbor pixels, because this requires additional memory.
3. Possibility for implementation in programmable logic device (PLD) – PLD can execute only simple mathematical operations like addition, subtraction and multiplication. The other operations require additional algorithms. With simple algorithms it is possible to perform division and calculate square root [2]. PLD's also work easy with integer numbers. Using integer numbers doesn't have to decrease the precision of used algorithm.
4. Possibility for finding small objects – separation of the image to small regions can make algorithm to find more than one object, because of local artifacts in the image. The used algorithm can not use local methods.

In the literature the threshold techniques are categorized into six groups as follows [4]:

- Histogram shape based methods. The histogram of the image is viewed as a mixture of two Gaussian distributions associated to the object and background classes: Sezan, Carlotto, Tsai, Olivo. Figure 3 is a histogram of the spot from Figure 2. It is clearly seen that the histogram has a lot of picks, due to noises and differences in illumination of the screen. MathCAD simulation shows that the pixels from the spot have level more than 230. Because small size of the spot, it do not provoke pick in the histogram. The methods based on the histogram are not applicable to solve the problem, because they do not correspond to criterion 4.

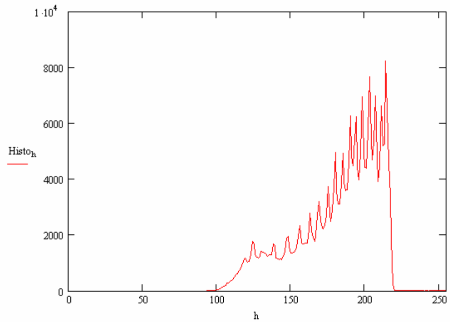


Fig. 3 Image of laser spot and its histogram

- Clustering based methods. The pixels are clustered in two classes as either background or foreground. Some of these methods are iterative: Riddler, Yanni and Otsu. They are slow methods. Some of them use complicate for realization functions like *log* or *exp*, which are hard for implementation in FPGA: Jawahar, Kittler and Lloud. These methods do not correspond to criteria 1 and 3.
- Entropy based methods. They use the difference in entropy between the foreground and background regions: Kapur, Brink, Yen and Shanbag. All of them require complicate calculations, which are difficult for implementation in programmable logic devices – criterion 3.
- Object attribute based methods. They find a measure of similarity between the grey level and the binarized images: Tsai, Hertz, and Leung. All of these methods also use complicated functions which are hard to implement in FPGA – see criterion 3.
- Spatial methods. They use higher-order probability distribution and/or correlation between pixels: Pal, Abutalep and Cheng. These methods also use hard to implement in FPGA mathematical functions and do not correspond to criterion 3.
- Local methods. They calculate the threshold value at each pixel based on the local image characteristics: Niblack, Sauvola, Bernsen, and Yasuda. These are statistic methods, making calculations over small regions from the image. They are fast methods, but not corresponding to criterion 4.

The results from the survey are shown in Table 1.

Table 1 Compare between groups of threshold algorithms

	Criterion			
Groups of Algorithms	1	2	3	4
Histogram based	✓	✓	✓	X
Clusters based	x	✓	x	✓
Entropy based	✓	✓	x	✓
Object Attribute	x	✓	x	✓
Spatial based	✓	x	x	✓
Local	✓	✓	✓	x

The table shows, that no one of these groups of threshold algorithms satisfies all criteria.

The local method of Niblack calculates threshold level using mean value and standard deviation from regions with size 15x15 pixels. In [1] it is described a global threshold algorithm of Hamadani. This method is similar to Niblack's method. The modified method of Hamadani is used in optoelectronic system.

IV. ALGORITHM DESCRIPTION

The method of Hamadani is statistic method [1]. It uses mean value m and standard deviation σ of the entire frame. The threshold level is linear combination of m and σ – Equation (3).

$$T = k_1 m + k_2 \sigma \tag{3}$$

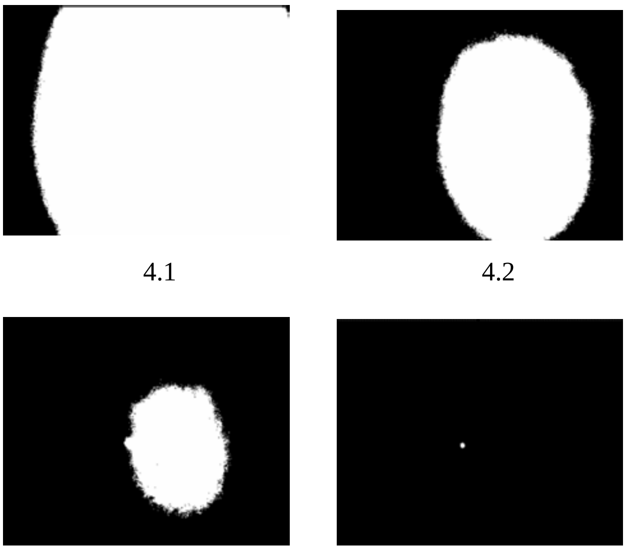
The weights k_1 and k_2 are pre-selected, based on image type in order to optimize performance. For a given image $\varepsilon \in R^x$. The mean value is given by Equation (4):

$$m = \frac{1}{M.N}\sum_{i=1}^{M}\sum_{j=1}^{N}\varepsilon(i,j) \tag{4}$$

M and N are numbers of rows and columns in the image. Standard deviation is given by Equation (5):

$$\sigma = \sqrt{\frac{1}{M.N}\sum_{i=1}^{M}\sum_{j=1}^{N}(\varepsilon(i,j)-m)^2} \tag{5}$$

Chosen algorithm is simulated in MathCAD using various values for k_1 and k_2. Tests with 20 images of laser spots are performed. Figure 4 shows the results for image from Figure 2.



4.1 4.2

4.3 4.4

Fig. 4 Image after thresholding

Table 2 shows threshold levels for various values for k_1 and k_2 and calculated threshold level T.

Table 2 Values for k_1, k_2 and calculated threshold level T

	Coefficients		
Image	**k_1**	**k_2**	**T**
4.1	0,5	2	147
4.2	1	0,5	196
4.3	1	1	210
4.4	1	2	237

The simulation shows that the best values for k_1 and k_2 are $k_1 = 1$ and $k_2 = 2$.

V. ALGORITHM IMPLEMENTATION

The basic applications of the optoelectronic system are measurement of decentre of lens and measurement of bending of rails.

A. Measurement decentre of lens

To measure the decentre of lens, laser beam passes through geometric axis of lens and it is projected at the screen of the optoelectronic system. When the lens routing around its geometric axis, the decentre drives to deviation of the laser spot over the screen. The deviation of the laser spot is proportional to the decentre of the lens. For good measurement it is needed a big number of images for one turn of lens. Used CMOS image sensor provides 30 frames per second (33.3 ms for one frame) and optoelectronic system must calculate coordinates of the laser spot for each frame. For adaptation to the changes in illumination, the threshold level is calculated in each frame. For this reason the algorithm of Hamadani is modified. The measurement is made during two consecutive frames in which spot can move, but the mean value and the standard deviation are constants in practice.

During the first frame the mean value is calculated. In the second frame the standard deviation is calculated and at the end of the second frame the threshold level is calculated. The threshold level is used in the next (third) frame. In the second frame simultaneously with the standard deviation, the next mean value is calculated. The threshold level is calculated continuously. The algorithm is shown in Figure 5.

B. Measurement of bending of rails

For the measurement of the bending of rails, the laser source is located at the end of the rail. The laser beam gets into the screen of a measurement system. Optoelectronic system moves along the rail and makes photos at a definite distance. Each bend provokes radial moving of the system.

This is detected by the measurement system like deviation of the laser spot. The photos are not continuous and two following images can be quite different. The threshold level is calculated using one frame and this is made at each frame. The time between two following frames must be minimum 33.3 ms. In this case external SRAM memory is used. The algorithm is shown in Figure 6.

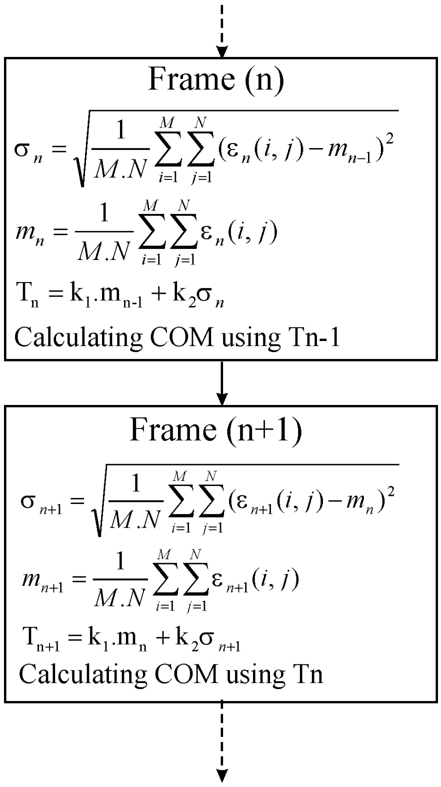


Fig. 5 Algorithm for decentre of lens measurement

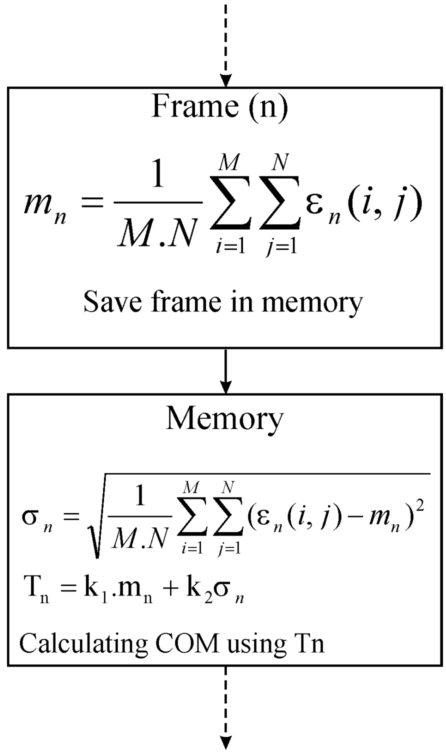


Fig. 6 Algorithm for definition the bend of rails

VI. CONCLUSION

In the paper it is made a comparison between various threshold algorithms according to offered criteria. After that the right algorithm for real-time applications is chosen. This algorithm was simulated in MathCAD. Figure 4 shows its ability to find small objects. It is implemented in FPGA like a part of complex optoelectronic system. Modified algorithms for work in real time are proposed. They are written in VHDL.

VII. REFERENCES

[1] Ritter, G. X. Computer Vision Algorithms in Image Algebra, New York, CRC Press, 2001.
[2] Roth, Ch. H. Digital System Design using VHDL, International Thomson Publishing, 1998.
[3] Parker, J. R. Algorithms for Image Processing and Computer Vision, New York, John Wiley & Sons Inc, 1997.
[4] Sezgin M. Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging 13 (1), pp. 146 – 165 (January 2004).