Measurements in Micromechanics Based on Computer Vision

Miguel Hernandez Acosta, Tetyana Baydyk, Ernst Kussul,
Karen L. Roldán Serrato, Omar Olvera Tapia

Centro de Ciencias Aplicadas y Desarrollo Tecnológico, Universidad Nacional Autónoma de México, Circuito Exterior S/N, Ciudad Universitaria, AP 70-186, CP 04510, México D. F. México
tbaidyk@unam.mx

Abstract

We work in area of micromechanics and its applications. One of the tasks of our interest is applications of micromechanics in renewable energy, especially in solar concentrator production and assembly. We have developed several prototypes of solar concentrators with triangular flat mirrors that approximate the parabolic surface of solar dish concentrator. To convert the heat energy to electric energy it is necessary to develop heat engines, for example, Stirling engine. It contains many small components. If we want to automate the manufacture and assembly processes we need to have a computer vision system to measure the dimensions of these engine components. We propose to apply neural classifier to measure the microcomponent dimensions.

1. Introduction

The micromechanics is one of the most important engineering sciences. There are many articles and books devoted to this theme [1]-[8]. The microcomponents are used in mobile telephone, in video and foto cameras, and so on. We use small components for solar concentrators. In Fig.1 we show the first prototype of parabolic dish solar concentrator with triangular flat mirrors to collect the solar energy.

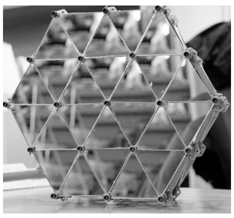

Figure 1: The first prototype of solar concentrator

For transformation it to electricity it is possible to use a Stirling engine [9]. The structure of Stirling engine is presented in Fig. 2.

We do not explain the principles of Stirling engine work because they are not the theme of this work. For us it is important that the Stirling engine contains many microcomponents and we want to recognize and to measure some mechanical components of these engines, for example, pistons, that are used in cold and hot cylinders (Fig.2).

This work contributes to the solution of problems in computer vision applications based on advanced algorithms, such as artificial neural networks, and, in addition, to reducing the costs of manufacturing of microcomponents. Our approach does not use very expensive commercial equipment such as smart cameras Cognex or LabView vision systems.

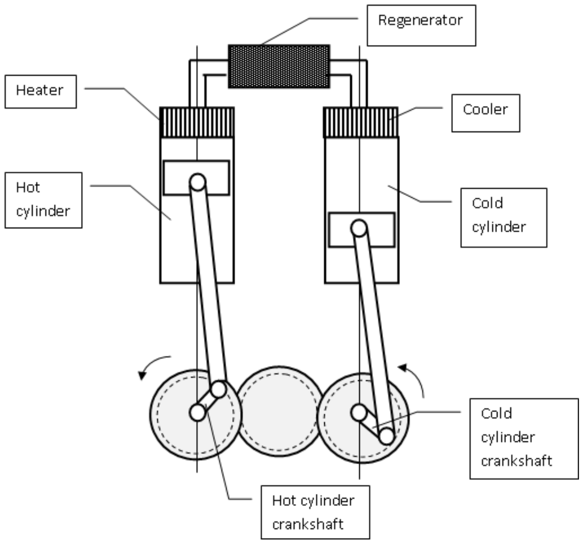

Figure 2: Stirling engine

To measure the piston diameter or piston length we need to obtain the images of the piston. Our principal method in computer vision use neural networks [4]. Using neural networks or neural classifiers we need to extract the object contours, and realize the edge detection. Different methods of contour detection are developed with subsequent application of neural networks [10] – [17]. Below we present several methods of contour detection, for example, Sobel and Roberts operators Prewitt and Schwartz algorithms.

Research and development of the computer vision system were carried out in such a way that complies with the following requirements:

1. To create a measuring method for microcomponent manufacturing.
2. To recognize three classes within the digital images (object, background, and object borders).
3. To configure the parameters of the neuronal classifier (number of neurons, number of training cycles, image size and window size within the image, etc.).

2. Methods of Contour Detection

This chapter presents the state of the art on how to extract the contours and the edge of objects in digital images. To obtain contours we use the conventional methods and to classify them we apply algorithms of artificial neural networks. In our investigation we focused mainly on the Sobel operator and Schwartz algorithm [12]. These methods were selected due to the fact that other methods are based on them and the principle is the same, for example, for the Canny, the Prewitt, and the Roberts operators.

2.1. Sobel operator

The Sobel operator allows the detection of contours on the image. This procedure is useful to distinguish objects from the background or objects from other objects in the image. The Sobel operator is based on the principle that there is an abrupt change in the neighbor pixels brightness and thus it is possible to define a contour. The contours are extracted with four different maps: the horizontal, vertical, and two diagonal maps [11]. The Sobel operator is technically a discrete differential operator in image processing, which computes an approximation of the gradient of the image intensity function. The result of the Sobel operator in each pixel of the image corresponds to the gradient vector indicating the direction of change in the intensity or color of the image, as shown in Fig.3.

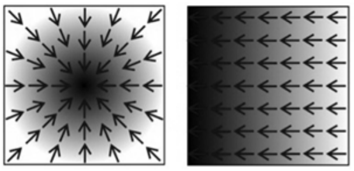

Figure 3: Gradient of the image intensity function

For the function $f(x,y)$ the gradient is expressed as:

$$\Delta f(x,y) = [G_x, G_y] = \left(\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right), \quad (1)$$

its magnitude is:

$$|\nabla f(x,y)| = [G_x^2 + G_y^2]^{\frac{1}{2}}, \quad (2)$$

and orientation is:

$$\theta(x,y) = tan^{-1} \frac{G_y}{G_x}. \quad (3)$$

But an image is seen as a discrete function, due to which a digital image intensity function is only known through discrete points. For the gradient of a discrete function a convolution filter gives an approach. The discrete convolution on an image is defined as in the expression 4.

$$(A * M) = \sum_i \sum_j A_{ij} M_{ij} \quad (4)$$

where A is the image, M is the filter that is applied, A_{ij} is the brightness of each pixel in the image A, and M_{ij} is each filter element.

The gradient of the image is generated with the convolution of the image and four filters separately, one in the horizontal direction, another vertically and two others on diagonals. With the resulting images of the application of filters, "OR" operation is used to create the final image with the contours in the desired directions (vertical, horizontal and diagonals). The filters consist of four masks 3x3. One example is presented in Fig.4.

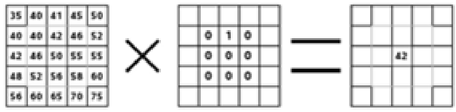

Figure 4: Example of convolution on an image.

2.2. Roberts Cross operator

The Roberts Cross operator, usually referred to as Roberts operator, is a differential operator, which is used as a detector of edges. It is based on the same principle as the Sobel operator, do one approximation of the magnitude of the gradient but with adjacent diagonal pixels, indicating if there is a point of edge without its orientation. The Roberts operator can be applied on images that can be binary or gray scale.

2.3. Prewitt operator

Other detector of edge based on the calculation of the gradient is the Prewitt operator, with which the abrupt changes in the image are highlighted. The Prewitt operator is very similar to the Sobel operator, since it takes into consideration adjacent pixels to make it more immune to noise; the difference lies in the coefficients of the masks, which do not emphasize the line or column that is processed. The Prewitt operator masks consist of arrays of 3x3, which is convolution with the original image to calculate approximations of the derivative, one in horizontal direction with matrix (5) and the other is in the vertical direction with matrix (6).

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad (5)$$

$$G_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad (6)$$

Masks of the Prewitt operator coefficients are generated in the same way that the coefficients of the Sobel operator masks, but do not emphasize the line or column that is processed.

2.4. Schwartz algorithm

The Schwartz algorithm describes how to get information about contours, to define angle of contour orientation, and used them as features of the image. The algorithm of Schwartz is presented as patent [12]. The method calculates not only the contour position but the angle of the contour orientation. The region of analysis includes 4 pixels with their brightness, defined as a square area of 2x2 pixels.

The Sobel operator and Schwartz algorithm were programmed and applied to images of pistons.

3. Recognition task

In Fig.5 we present several examples of original piston images. Below the images we present in extended scale the upper left border of the pistons (the board is marked with white color).

To use these images in recognition based on LIRA (Limited Receptive Area) neural network we needed to mark them. In Fig.6 we demonstrate three steps of image marking. The first image is original (Fig.6a), in the second image (Fig.6b) the background is demonstrated with grey color, in third image (Fig.6c) the object is white, and black line demonstrate the border.

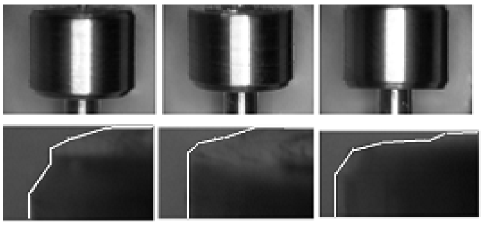

Figure 5: Piston images.

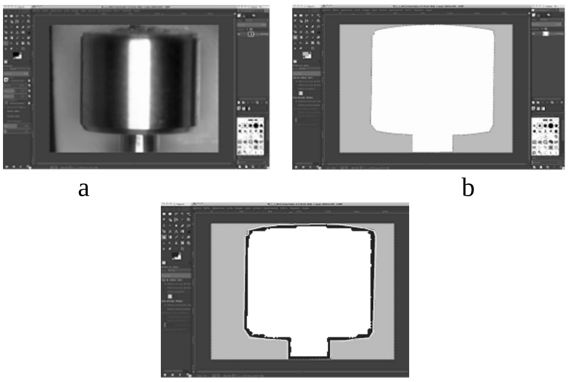

Figure 6: Image marking.

For piston images presented in Fig.5 we obtained the following marking images (Fig.7).

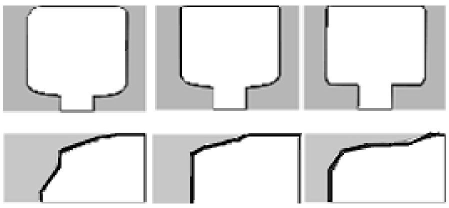

Figure 7: Piston images with three classes (grey – background, white – object, black line – object contour*)*

These segmented images we can use to train our neural network. 15 pistons were manufactured, which were classified into 5 groups, every group has 3 pistons. Every group included pistons with almost the same diameter (slight variations due to the inaccuracy of the equipment with which were manufactured). We had five groups of pistons with diameters: 8.00 mm, 8.1mm, 8.2 mm, 8.3 mm, 8.4 mm. The captured images, as shown in the examples in Fig.5, have a resolution of 1600 x 1200 pixels in BMP format.

In the following section we describe the neural network structure.

4. Neural Network

We propose to use LIRA neural network to classify three classes (object, background and contour) [4], [18], [19]. The structure of the LIRA neural classifier is presented in Fig.8.

This LIRA neural classifier was programmed with *Visual C#* from *Visual Studio 2010 Professional*®. We used two computers (computer characteristics are presented in Table 1).

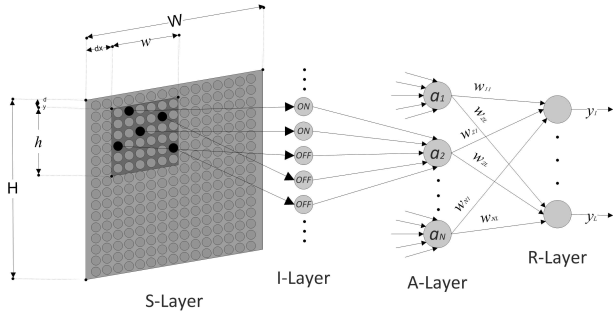

Figure 8: Structure of LIRA neural classifier

Table 1: Computer characteristics

Characteristics	System 1	System 2
Processor Memory OS	i7 @ 2.8GHz 4GB Windows® 7 Professional @ 32 bits	i7 @ 2.66GHz 8GB Windows® 7 Professional @ 64 bits

The first step was LIRA neural classifier training. To the sensor *S*-layer we presented samples of three classes. In Fig.9 we present one sample of background in point with coordinates (*X*,*Y*). This sample is the window (in blue color) with size of 100x100 pixels (in several experiments we used 150x150 pixels). In this window the ON and OFF neurons were randomly selected for intermediate level or *I*-layer.

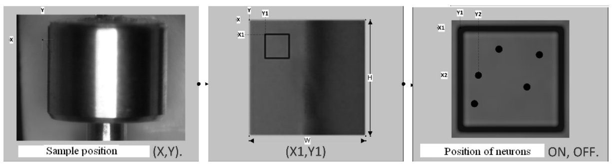

Figure 9: Sample formation for LIRA neural classifier

The number of ON and OFF neurons presents the parameter of investigation. Practically we used 2 ON neurons and 3 OFF neurons for codification of every sample.

The associative *A*-layer and the response *R*-layer have the connections "all neurons of *A*-layer are connected with all neurons of *R*-layer". These connections have the weights that are changed during the training process (we use the Hebb rule).

In the same manner as demonstrated in Fig.9 we prepared the samples of three classes (background, object and object border) to train and to test our neural network.

5. Results

For our experiments we selected the neural network with 1024 neurons in *A*-layer. Number of training cycles was 10.

In Fig.10 and Fig.11 we present the error number for 10 cycles of training process (for two different subset of training images, case I and II).

We had approximately 9,000-10,000 samples for every image. From them near 3,000 samples presented the border pixels of object.

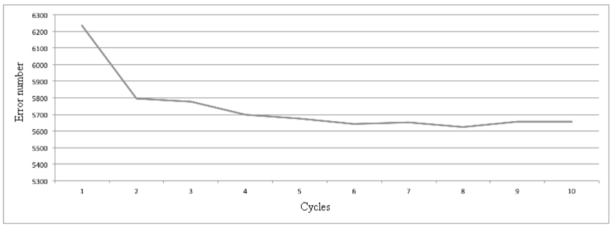

Figure 10: Error number for 10 cycles of training (case I).

We used seven images for training and the rest of images to test the system.

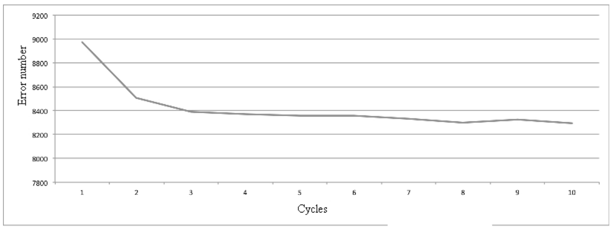

Figure 11: Error number for 10 cycles of training (case II).

The process of training demonstrated the possibility to improve the recognition rate.

To test the neural classifier we used the images that did not participate in training process. We obtained above 70% of recognition rate.

6. Conclusions

Different algorithms of contour detection (as Sobel and Roberts operators and Schwartz algorithm) generate false contours on the image that complicate the recognition process. We propose to use a neural classifier to recognize three classes as background, object and object border. When we obtain information about border we can use this information to measure diameter of the piston. The neural classifier permits us to recognize the border of pistons. The best recognition rate was obtained above 70%.

7. Acknowledgements

This work was supported in part by projects PAPIIT IN 102014, IT 102814.

8. References

[1] Trimmer, W. S., *Micromechanics and MEMS: Classic and Seminal Papers to 1990*, Wiley-IEEE Press, 1997.

[2] Kussul, E., Rachkovskij, D., Baidyk, T., and Talayev, S., “Micromechanical Engineering: a Basis for the Low-Cost Manufacturing of Mechanical Microdevices Using Microequipment”, *J. of Micromech. & Microeng.*, 6(4):410 – 425, 1996.

[3] Chollet, F., and Liu, H., Short Introduction to MEMS. At the website http://memscyclopedia.org/introMEMS.html, Oct. 2012.

[4] Kussul, E., Baidyk, T., and Wunsch, D.. *Neural Networks and Micromechanics*, Springer, Berlin Heidelberg, 2010.

[5] Kussul, E., Baidyk, T., Ruiz-Huerta, L., Caballero-Ruiz, A., Velasco, G., and Makeyev, O., *Techniques in the Development of Micromachine Tool Prototypes and Their Applications in Microfactories MET Technology*. In CorneliusT. Leondes, editor, *MEMS/NEMS*, Springer, 2006.

[6] Baidyk, T., Kussul, E., Makeyev, O., Caballero, A., Ruiz, L., Carrera, G., and Velasco, G., “Flat Image Recognition in the Process of Microdevice Assembly”, *Pattern Recognition Letters*, 25(1):107 – 118, 2004.

[7] Tanaka Makoto, “Development of Desktop Machining Microfactory”, *RIKEN*, 34:46 – 49, 2001.

[8] Kussul, E., Baidyk, T., Ruiz-Huerta, L., Caballero-Ruiz, A., Velasco, G., and Kasatkina, L., “Development of Micromachine Tool Prototypes”, *J. of Micromech. & Microeng.*, 12(6):795, 2002.

[9] Kussul, E., Makeyev, O., Baidyk, T., and Olvera, O., “Design of Ericsson Heat Engine with Micro Channel Recuperator”, *ISRN Renewable Energy*, (4):8, 2012.

[10] Duda, R.O., Hart, P.E., and Stork, D.G., *Pattern Classification and Scene Analysis: Pattern Classification*. Wiley-Interscience, 2000.

[11] Cruz, R.M.O., Cavalcanti, G.D.C, and Ren, T.I., “Handwritten Digit Recognition using Multiple Feature Extraction Techniques and Classifier Ensemble”, *17th International Conference on Systems, Signals and Image Processing*, 2010.

[12] Schwartz, R., “Method and a Circuit for Determining a Contour in an Image”, *Patent US4433912*, at the website http://www.google.com/patents/US4433912, February 1984.

[13] Terry, P.J. and Vu, D., “Edge Detection Using Neural Networks”, in Proc. of the *27th Conf. on Signals, Systems and Computers*, 1993, vol.1, pp. 391 – 395.

[14] Grassi, G., Vecchio, P., Di Sciascio, E., Grieco, L.A., and Cafagna, D., “Neural Networks for Image Processing: New Edge Detection Algorithm”, in *IEEE Intern. Conf. on Electro/Information Technology*, 2007, pp. 498 – 502.

[15] Damak, A., Krid, M., and Masmoudi, D.S., “Neural Network Based Edge Detection with Pulse Mode Operations and Floating Point Format Precision”, in Proc. of *3rd Intern. Conf. on Design and Technology of Integrated Systems in Nanoscale Era*, 2008, pp. 1 – 5.

[16] Mehrara, H., Zahedinejad, M., and Pourmohammad, A., “Novel Edge Detection Using BP Neural Network based on Threshold Binarization”, in *Second Intern. Conf. on Computer and Electrical Engineering*, 2009, vol. 2, pp. 408 – 412.

[17] He, Zh., and Siyal M.Y., "Edge Detection with BP Neural Networks”, in proceedings of the *Fourth Intern. Conf. on Signal Processing*, 1998, vol. 2, pp. 1382 – 1384.

[18] Baydyk T., Kussul E., Hernández Acosta M., “New Application of LIRA Neural Network”, In Proc. of the *16th WSEAS Intern. Conf. on Circuits*, Kos Island, Greece, 14-17 July 2012, pp.115-119.

[19] Baidyk T., Kussul E., Hernandez Acosta M., and Vega A., “Research and Development of a Recognition System for Measurements in Micromechanics”, *ICMEAE 2013*, Cuernavaca, México, pp.53-58.