Neural Network Based Vision System for Micro Workpieces Manufacturing

BAIDYK T., KUSSUL E.
Center of Applied Science and Technological Development,
National Autonomous University of Mexico (UNAM),
Cd. Universitaria, A.P. 70-186, C.P. 04510, Mexico, D.F.
MÉXICO

Abstract: - A neural network based vision system for micro workpieces manufacturing is developing. The system has to recognize the shapes of the micro workpieces and to give the information for correction of the workpieces manufacturing process. The system prototype was proved off-line on the task of 3 mm screw shape recognition. The shape distortions were made with the incorrect position of the cutter in CNC lathe. The developed vision system recognizes correctly 92.5% of images with different cutter positions. This information could be used in the future to correct the cutter position in the process of micro workpieces manufacturing.

Key-Words: - vision system, neural network, adaptive manufacturing, micromachine tool, microfactory.

1 Introduction

A computer vision system permits one to provide the feedback, which increases the precision of the manufacturing process. It could be used in the low cost micromachine tools and micromanipulators for microdevices production. A method of sequential generations was proposed to create such microequipment [1].

According to this method the microequipment of each generation has the sizes smaller than the sizes of the equipment of the previous generations. This approach would allow us to use low cost components for each microequipment generation and to create the microfactories capable to produce the low cost microdevices [2].

To preserve a high precision of the microequipment it is necessary to use adaptive algorithms of micropieces production. The algorithms based on the contact sensors were proved and showed good results [2]. The neural network based vision system could provide much more extensive possibilities to improve the manufacture process.

A special project for microfactory creation based on miniature micromachine tools was initialised in Japan [3]. The Mechanical Engineering Laboratory developed a desktop machining microfactory [4], [5]. The microfactory consists of machine tools such as a lathe, a milling machine, a press machine and assembly machines such as a transfer arm and a two-fingered hand. This portable microfactory has external dimensions 625x490x380 mm^3.

The idea of microfactory creation is also supported in other countries. In Switzerland, the details of precision motion control and microhandling principles for future microfactories have been worked out [6]. One of the main problems in such microfactories is the problem of automation on the base of vision systems. There are different approaches to construct the computer vision systems [7] - [9] for these purposes.

In this paper we communicate the preliminary results of neural network based vision system investigation. We have proved the system in off-line mode. For this purpose we have manufactured four groups of screws with the different positions of the machine tool cutter. The images of these screws were used for training and testing of the developed system.

The task of shape recognition is well known [10]. The recognition of screw images in our case is based on the recognition of the screw shape or profile. We detected the screw contours and use this presentation as input of the recognition system.

The vision system was based on the neural network with permutation coding technique described in [11], [12]. This type of neural network showed good results in handwritten and face recognition tasks. Now we prove it for adaptive algorithms in manufacture.

2. The problems of adaptive cutting processes

To increase the precision of micromachine tools it is possible to use adaptive cutting processes [2].

Let us consider a lathe equipped with one TV camera, which could be moved automatically from the position 1 to the position 2 (Fig.1).

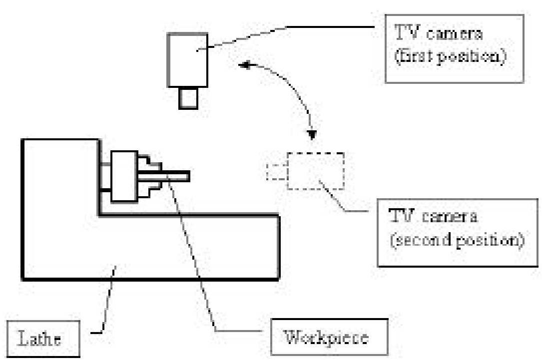


Fig.1. The lathe equipped with TV camera

The images from the TV camera in the first position could be used to measure the sizes of partially treated workpieces and to make needed corrections in the cutting process. The images from the TV camera in the second position could be used to correct the position of the cutting tool relatively to the workpieces (Fig.2).

In the both positions the TV camera can give useful information about the passing of the cutting process, for example, the chips formation, the contact of the cutter with the workpiece, etc.

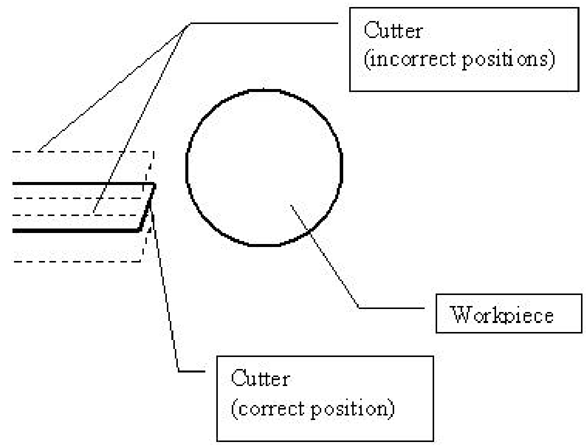


Fig.2. The position of the cutting tool relatively to the workpieces

All such images must be treated with the image recognition system. We propose to create such recognition system on the base of the neural network with permutative coding.

3. Permutative coding neural classifier

A Permutative Coding Neural Classifier (PCNC) was developed for the recognition of different types of images. It was proved on the MNIST database (handwritten digit recognition), ORL database (face recognition) and showed good results [11], [12]. Here we examine this classifier in the micromechanical applications.

The PCNC structure is presented in Fig.3. The image is input to the feature extractor. The extracted features are applied to the encoder input. The encoder produces the output binary vector of large dimension, which is presented to the input of one-layer neural classifier. The classifier output gives the recognized class.

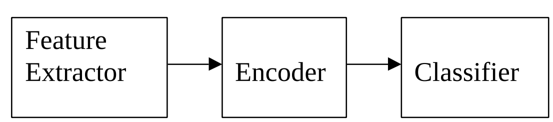
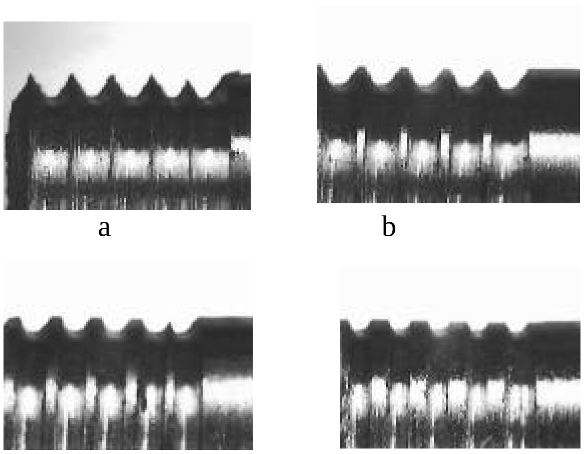


Fig. 3. Structure of the Permutative Coding Neural Classifier

3.1. Feature extractor

An initial image (Fig.4) is to be input to the feature extractor. The feature extractor begins with the selection of the specific points in the image. In principle, various methods of the specific points selection could be proposed. For example, the contour points could be selected as the specific points.

The rectangle of $h•w$ size is formed around each specific point (Fig.5).



a b

c d

Fig. 4. Initial images

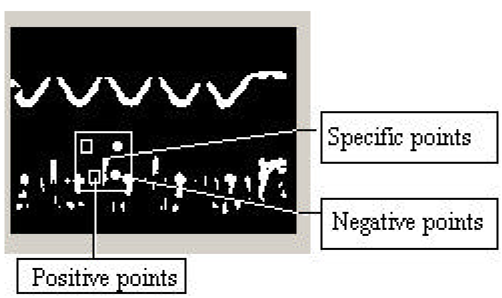


Fig. 5. Specific points selected by the feature extractor

The specific points were selected using the following procedure. For each set of four neighboring pixels we calculate the following expressions:

$$d_1 = \left| br_{ij} - br_{i+1\,j+1} \right| ,$$

$$d_2 = \left| br_{ij+1} - br_{i+1\,j} \right| , \qquad (1)$$

$$\Delta = \max(d_1, d_2) ,$$

where br_{ij} is the brightness of the pixel (i,j).

If ($\Delta > B$), then the pixel (i,j) corresponds to the selected specific point on the image, where B is the threshold for specific point selection.

Each feature is extracted from the rectangle of $h•w$ size (Fig. 5). The p positive and the n negative points determine one feature. These points are randomly distributed in the rectangle. Each point P_{rs} has the threshold T_{rs} that is randomly selected from the range:

$$T_{min} \le T_{rs} \le T_{max}, \qquad (2)$$

where s stands for the feature number; r stands for the point number.

The positive point is “active” if in initial image it has brightness:

$$b_{rs} \ge T_{rs}. \qquad (3)$$

The negative point is “active” if it has brightness:

$$b_{rs} \le T_{rs}. \qquad (4)$$

The feature under investigation exists in the rectangle if all its positive and negative points are active. In other cases the feature is absent in the rectangle.

3.2. Encoder

The encoder transforms the extracted features to the binary vector:

$$V = \{v_i\} \ (i = 1, \ldots, N),$$

where v_i = 0 or 1. For each extracted feature F_s the encoder creates an auxiliary binary vector:

$$U = \{u_i\} \ (i = 1, \ldots, N),$$

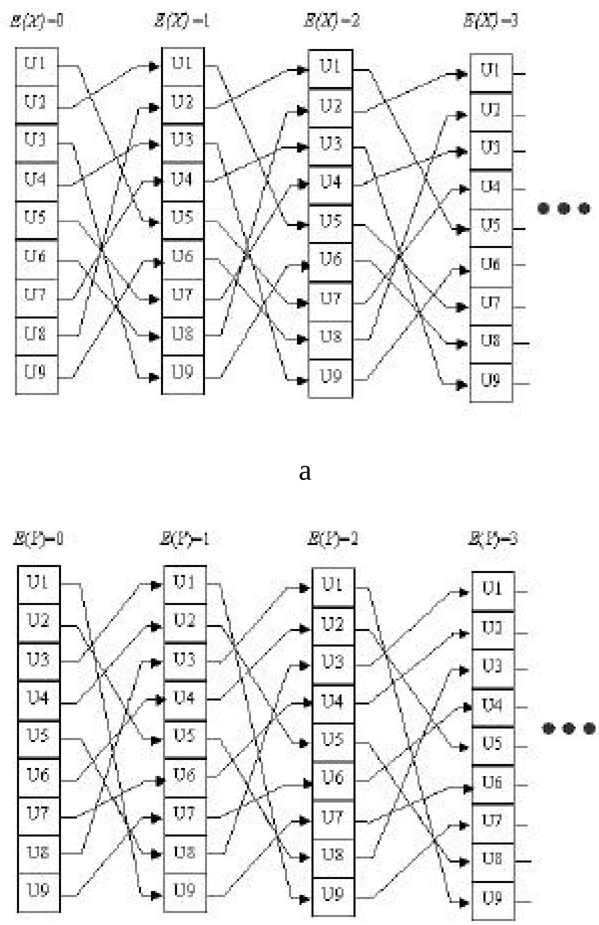
where u_i = 0 or 1.

A special random procedure is used to obtain the positions of ones in the vector U_s

for each feature F_s. This procedure generates the list of the positions of "1" for each feature and saves all such lists in the memory. The vector U_s we term "mask" of the feature F_s. To create this vector it is necessary to take the positions number form list and filled them with "1" and other positions are filled with "0".

In the next stage of encoding process it is necessary to transform the auxiliary vector U to the new vector U^* which corresponds to the feature location in the image. This transformation is made with permutations of the vector U components (Fig. 6).

The number of permutations depends on the feature location in the image. The permutations in horizontal (X) (Fig.6, *a*) and vertical (Y) (Fig.6,*b*) directions are different permutations.



a

b

Fig. 6. Permutation pattern.

Each feature can have different locations in the image. The feature will have different binary code for each location. For two locations of the same feature the binary codes must be strongly correlated if the distance between the feature locations is small and must be weakly correlated if the distance is large. Such property could be obtained with the following procedure.

To code the feature F_s location in the image it is necessary to select the correlation distance D_c and calculate the following values:

$$X = j / Dc,$$
$$E(X) = (int)X, \tag{5}$$
$$R(X) = j - E(X) \bullet Dc,$$
$$Y = i / D_c,$$
$$E(Y) = (int)Y \tag{6}$$
$$R(Y) = i - E(Y) \bullet D_c,$$
$$P_x = \frac{R(X) \bullet N}{D_c}, \tag{7}$$
$$P_y = \frac{R(Y) \bullet N}{D_c}, \tag{8}$$

where $E(X)$ is the integer part of X; $R(X)$ is the fraction part of X; i is the vertical coordinate of the detected feature; j is the horizontal coordinate of the detected feature, N is the number of neurons.

The mask of the feature F_s is considered as a code of this feature located at the left top corner of the image. To shift the feature location in the horizontal direction it is necessary to make its permutations $E(X)$ times and to make an additional permutation for P_x components of the vector. After that, it is necessary to shift the code to the vertical direction making the permutations $E(Y)$ times and an additional permutation for P_y components.

3.3. Neural Classifier

The structure of proposed recognition system is presented in Fig. 7.

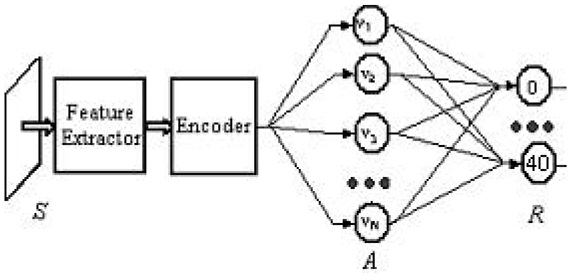


Fig. 7. Permutative Coding Neural Classifier.

The system contains the sensor layer *S*, feature extractor, encoder, the associative neural layer *A*, and the reaction neural layer *R*.. In the screw shape recognition task each neuron of the *R*-layer corresponds to one of the screw groups from database. The sensor layer *S* corresponds to the initial image.

The associative neural layer contains “binary” neurons having the outputs 0 or 1. The values of its neuron outputs are produced as a result of the work of the encoder. The neurons of the associative layer *A* are connected to the reaction layer *R* with the trainable connections having the weights w_{ji}. The excitations of the *R*-layer neurons are calculated in accordance with the equation:

$$E_i = \sum_{j=1}^{n} a_j * w_{ji} \tag{9}$$

where E_i is the excitation of the *i*-th neuron of the *R*-layer; a_j is the excitation of the *j*-th neuron of *A*-layer; w_{ji} is the weight of the connection between the *j*-th neuron of the *A*-layer and the *i*-th neuron of the *R*-layer.

The neuron winner having maximal excitation is selected after the calculations of the excitations.

We use the following training procedure. Denote the neuron-winner number as i_w, and the number of neuron, which really corresponds to the input image, as i_c. If $i_w = i_c$ nothing to be done. If $i_w \neq i_c$

$$\begin{aligned}(\forall j)\left(w_{ji_c}(t+1) = w_{ji_c}(t) + a_j\right)\\(\forall j)\left(w_{ji_w}(t+1) = w_{ji_w}(t) - a_j\right)\end{aligned} \tag{10}$$

$$\text{if } (w_{ji_w}(t+1) < 0) \quad w_{ji_w}(t+1) = 0,$$

where $w_{ji}(t)$ is the weight of the connection between the *j*-neuron of the *A*-layer and *i*-neuron of the *R*-layer before reinforcement, $w_{ji}(t + 1)$ is the weight after reinforcement.

4. Results

To examine the PCNC in recognition of the shape of micromechanical workpieces we have produced 40 screws of 3mm diameter with the CNC-lathe of the company “Boxford”. Ten screws were produced with correct position of the thread cutting cutter (Fig.4, b). Thirty screws were produced with erroneous positions of this cutter. Ten of them (Fig. 4, a) had the distance between the cutter and screw axis of 0.1mm less than it was necessary. Ten screws (Fig.4, c) were produced with the distance of 0.1mm larger than it was necessary and the rest ten (Fig.4, d) with the distance 0.2mm larger than it was necessary. We have made the image database from this screws using Web-camera Samsung mounted on the optical microscope.

Five images from each group of screws selected randomly were used for the neural classifier training and the rest five were used for the neural classifier examination.

The experiments were made with different parameter *B* of specific point selection.

In the Table 1 the first column corresponds to the parameter *B*. Four different runs were made for each value *B* to obtain statistically reliable results. Each run differs from others by the set of samples selected randomly for the neural classifier training and by the permutation scheme structure. For this reason we obtained different error rates in different runs. The second column contains the error number for each run. The third column gives mean value of error number for each *B* value. The fourth column contains the mean percentage of correct recognition.

Table 1.
Results of system investigation

Threshold for specific point selection	Error number (four runs)	Average number of errors	% of correct recognition
20	3	3	85
	4		
	1		
	3		
40	2	2.25	88.75
	2		
	2		
	3		
60	3	1.5	92.5
	1		
	1		
	1		

4. Conclusions

A new neural classifier PCNC was developed for recognition of different types of images. It showed good results on MNIST and ORL databases. In this paper the application of PCNC to micro workpieces shape recognition was investigated. The results show that this application of PCNC has a good perspective. The future works will be connected with online experiments with the neural recognition system on the base of the micromachine tools and micromanipulators, which form the microfactory.

5. Acknowledgment

This work was funded by Projects CONACYT 33944-U, NSF-CONACYT39395A, PAPIIT IN 112102.

References:

[1] Kussul E., Rachkovskij D., Baidyk T. et al., Micromechanical engineering: a basis of the low cost manufacturing of mechanical microdevices using microequipment. *Journal of Micromechanics and Microengineering,* 6, 1996, pp. 410-425.

[2] Kussul E., Baidyk T., Ruiz-Huerta L., Caballero A., Velasco G., Kasatkina L. “Development of Micromachine Tool Prototypes for Microfactories”, *Journal of Micromechanics and Microengineering*, 12, 2002, 795-812.

[3] Some Micro Machine Activities in Japan. //Report ATIP96.021, 1996.

[4] Okazaki Yuichi, Kitahara Tokio. Micro-Lathe Equipped with Closed-Loop Numerical Control. *Proceedings of the 2-nd International Workshop on Microfactories,* Switzerland, Oct.9-10, 2000. – pp.87- 90.

[5] Naotake Ooyama, Shigeru Kokaji, Makoto Tanaka and others. Desktop Machining Microfactory. *Proceedings of the 2-nd International Workshop on Microfactories,* Switzerland, Oct.9-10, 2000. pp.14-17.

[6] Bleuler H., Clavel R., Breguet J-M., Langen H., Pernette E. Issues in Precision Motion Control and Microhandling. *Proceedings of the 2000 IEEE International Conference On Robotics & Automation*, San Francisco, pp.959-964.

[7] Jonathan Wu Q.M., M.F. Ricky Lee, Clarence W. de Silva. Intellihgent 3-D Sensing in Automated Manufacturing Processes. In: *Proceedings IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Italy, 2001, 366-370.

[8] Lee, S.J., K.Kim, D.-H. Kim, J.-O.Park, G.T. Park,. Recognizing and Tracking of 3-D-Shaped Micro Parts Using Multiple Visions for Micromanipulation. In: *IEEE International Symposium on Micromechatronics and Human Science*, 2001, 203-210.

[9] Kim, J.Y., H.S.Cho. A vision Based Error-Corrective Algorithm for Flexible Parts Assembly. In: *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, Portugal, 1999, 205-210.

[10] Grigorescu C., Petkov N. Distance Sets for Shape Filtres and Shape Recognition. IEEE Transactions on Image Processing, Vol.12, N.10, October 2003, pp. 1274-1286.

[11] Kussul E.M., Baidyk T.N. Permutative coding technique for handwritten digit recognition. *Proceedings of the 2003 IEEE International Joint Conference on Neural Networks (IJCNN’2003),* Portland, Oregon, USA, July 20-24, 2003, V.3, pp. 2163-2168.

[12] Kussul E., Baidyk T., Kussul M. Neural Network System for Face Recognition. *2004 IEEE Internatrional Symposium on Circuits and Systems*, ISCAS 2004, May 23-26, Vancouver, Canada, pp. 4. (accepted).