



Intelligent detection of edge inconsistency for mechanical workpiece by machine vision with deep learning and variable geometry model

Xiankun Lin¹ · Xin Wang¹ · Li Li¹

Published online: 28 February 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Inconsistent edges of mechanical workpieces in the same batch are one of the main reasons that lead to different machining performance. A full recognition of the inconsistent features has significant impact on enhancement of their intelligent machining ability. An intelligent hybrid strategy is proposed for edge inconsistent feature detection by machine vision, in which deep learning is combined with variable geometric model together to conduct the function. A deep convolutional neural network based feature classification model is established with K-Means clustering tactic. Supported on the classification model, a variable geometric model for specific edge inconsistency is given with an inconsistency evaluation function to investigate the match degree between the geometric model and the actual detected edge, and then particle swarm optimization algorithm is applied to find the solution of this geometric model. Detection experiments are carried out on a domestic servo-driven vision measuring platform to verify the performance of the proposed approach. The results show that the combined scheme can classify the different type of geometric contour of edge features with 100% correctness, and better evaluation performance by dice similarity index and Hausdorff distance in comparisons with other recent candidate methodologies. It is also indicated that the presented method provides a good recognition of the geometrical shape with less than 0.06mm maximum error for workpiece with 142 × 119mm size in the visual field.

Keywords Inconsistent features · Deep learning · Contour edge · Variable parametric geometric model

1 Introduction

Due to constraints of industrial production conditions [1], edges of mechanical parts in bulk processing often appear inconsistent in dimension. A full recognition of the inconsistent features of the edges plays an important roles in enhancement of the intelligent machining level for batch-processed workpieces [2]. The edge inconsistent features are with characteristic of various geometric contour and randomly-located positions. Although conventional contour

measuring methods with detection by machine vision or measuring on CMM, can acquire the dimensions of the edge profile, it is difficult for them to recognize the edge features and obtain their basic dimension [3]. Hence, the research on how to improve the detection precision of the edge inconsistent features has gained more and more attention [4].

Detection by machine vision is with advantages for its low equipment cost, fast detection speed, and high recognition precision has been employed in more and more fields [5]. To realize intelligent detection of features by machine vision, two technical problems are in solution. One problem is how to intelligently recognize the edge inconsistent features, and the other is how to accurately measure the size of edge inconsistent features [6]. Fan et al. [7] present a two-phase fuzzy clustering algorithm to segment the remote sensing image from polarimetric synthetic aperture radar. Kusakunniran et al. [8] identify the hard exudates from retinal image with MLP supervised learning based iterative graph cut. Tsai et al. [9] apply maximum expectation algorithm (EM) to give weight to

✉ Xiankun Lin
linxk333@126.com

Xin Wang
870318668@qq.com

Li Li
756587155@qq.com

¹ School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

each edge points, which immensely reduces computation time of image matching. Han et al. [10] construct weighted least square model with Gaussian mixture algorithm to model the background of object to obtain the defects. Michal et al. [11] realize the inspection for micro-milling tool by wavelet-based method with reconstruction of depth image. Guo et al. [12] establish a shape model for reconstruction of parts by visual localization with high precision. Shanhabi et al. [13] obtain surface roughness and dimensional deviation data on a machined workpieces by establishment of 2-D images of cutting tools to improve the detection speed. Cano et al. [14] present a precise measurement method for prediction of the elastic deformations of machine parts based on efficient mass-spring-damper model. The above-mentioned researches on edge contour detection and recognition are almost conducted by designing feature-processing algorithms on low-level image semantic features from manual supply. Moreover, in the actual edge features recognition, it is difficult to establish comprehensive image semantic expressions, and there are many factors that influence the measurement precision of the edge inconsistent features. Meanwhile, it is also difficult to process illumination disturbance in real time [15]. Consequently, there exist some space to improve the level of intelligent detection of edge inconsistent features on these contributions.

In intelligent feature recognition [16], deep learning has been widely applied in many fields, e.g. medical diagnosis [17], mechanical fault diagnosis [18], etc. for its ability of automatic extraction of high-level semantic features with strong robustness and without manual-construction target features [19]. Because of this merit, the technology is commonly utilized to obtain high-level semantic features of images by mixing other intelligent optimization algorithms [20, 21]. Quite a few contributions have been done on edge inconsistent features detection with this strategy [22]. Diao et al. [23] establish a deep belief network (DBN) on the sample features to realize edge inconsistent features detection of the sample. Chowdhury et al. [24] apply deep learning for automatic recognition of materials with different compositions and orientation of microstructural features. Liu et al. [25] point out that deep learning with multistage convolutional networks has the best overall performance in local binary features description. Sarkar et al. [26] propose an automatic method combined with Holistically Nested edge detection algorithm (HED) to recognize the closed contour of image with 17% reduction in mismatching rate. Zhang et al. [27] present a deep-space-varying convolutional neural network model for detection of image boundary features and inconsistencies with lower memory usage and computational complexity. Hoang et al. [28] raise an intelligent method on improved convolutional neural network for road surface crack identification. Zhu et al. [29]

design 2D/3D sketch-based 3D retrieval methods with the Convolutional Neural Network to represent 3D objects with good performance.

Inconsistency detection of edge features for mechanical workpiece is investigated with deep learning based intelligent hybrid scheme in this work. Firstly, a deep convolutional neural network-based model is established for inconsistent feature classification with K-Means clustering tactic. And then on support of the classification model, a variable geometry model for specific edge inconsistency detection is given with an inconsistency evaluation function to evaluate the match degree between the geometric model and actual edge, in which an optimization algorithm is applied to acquire the solution of match process. As a contribution, a novel machine vision-based intelligent method is proposed in detection of the edge inconsistent features of mechanical parts to improve the automatic detection level in intelligent machining for large-batch workpieces.

The remainder of this paper is organized as follows. Section 2 presents the edge inconsistent feature classification model on the basis of deep convolutional neural network model. In Section 3, a variable geometry model for specific edge inconsistency detection is established. Section 4 describes the parameter solution of the geometric model for inconsistent features. In Section 5, experimental verification is introduced with the scheme of image data set establishment and experiment strategy. Section 6 deals with the experimental data analysis, discussion of the performance of the proposed approach. Finally, the conclusions are summarized in Section 7.

2 Deep learning based architecture for inconsistent feature classification

2.1 Construction of classification model

In accordance with whether labels for image data set are available, deep learning models can be classified into unsupervised deep belief networks (DBN) [30] and supervised deep convolutional neural networks (CNN) [31]. In consideration that the convolution layer in CNN processes stronger ability to mine local spatial correlation of image than DBN in image recognition, the supervised network is applied in detection of the edge inconsistency. The classification model for edge inconsistent feature detection is constructed on deep learning network, which is basically a multi-layer model including input layer, convolutional layer, pooling layer, hidden layer, K-Means layer and output layer [32]. Its major differences to conventional neural network lie in local perceptual component, sharing weights and pooling operation [33].

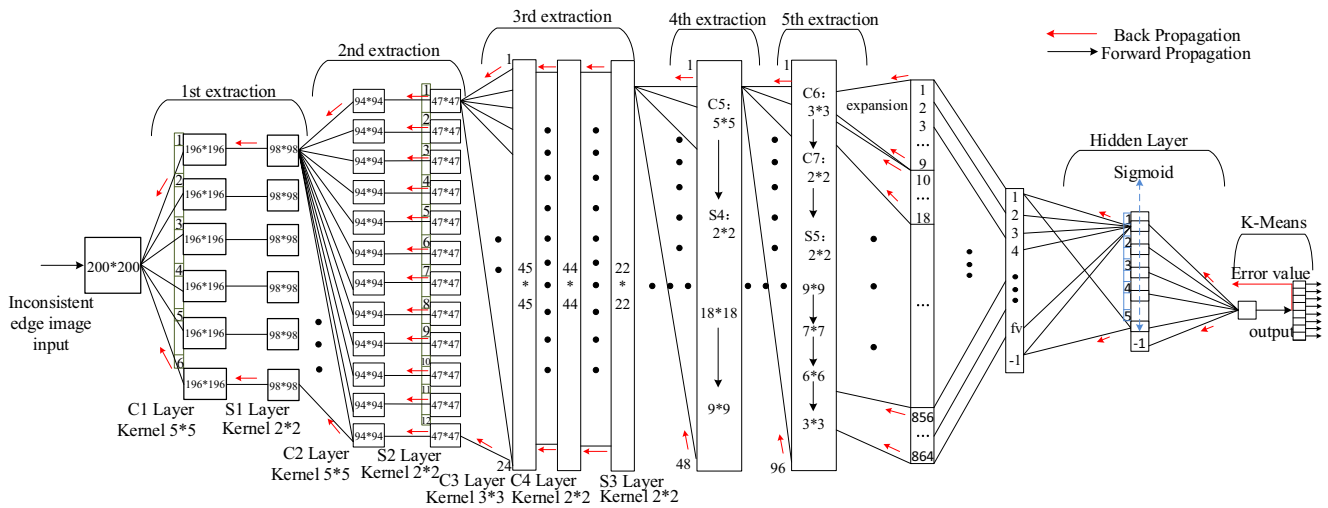


Fig. 1 Model architecture with deep learning network for edge inconsistent feature classification

An inconsistent feature classification model, illustrated in Fig. 1, is constructed on deep convolutional neural network with K-Means clustering tactic, in which there are five order feature extractions applied for convolutional pooling calculations from the input edge image. With expansion of the last extraction results, hidden layer and K-means clustering layer are introduced to realize the classification function.

It is indicated in Fig. 1 along with Table 1 listing the parameters used in the CNN structure that multiple layer extractions are adopted to perform superposition operation with the results, transforming of the 200×200 pixel edge inconsistent feature image into 96 pieces of 3×3 feature matrices. These tiny matrices are expanded into the form with one column vector containing 864 elements. By means of Gaussian operation through hidden layer neurons, the vector is transformed into a floating regression value. In the end, the classification result is obtained by K-means clustering algorithm for the inconsistent feature input [34].

There are two principles for determining the size of input image, in which one is that the image size contains at least a single part edge inconsistent feature, and the other is that

the image size containing necessary information should be small enough to avoid a sharp increase in computation. In addition, the appropriate input image size range is considered between 32×32 and 300×300 pixels [35], the size of 200×200 pixels image is selected as the input data. Considering the size of image and the number of feature extraction, seven convolutional layers C1-C7 are applied to extract image features and reproduce the image in the form of feature matrix as input of the activation function of layers. For the seven-layer network structure, the convolutional kernels are arranged as follows, C1, C2 and C5 layers with size of 5×5 convolutional kernel, C3, C6 layers with size of 3×3 convolutional kernel, and C4, C7 with size of 2×2 convolutional kernel. Different component weights and biases of each convolutional kernel of the layer generate various characteristic matrices for the next layer. There are five down sampling layers S1-S5 with 2×2 size designed to cooperate with the convolution sections for data dimension reduction and over fitting avoidance by data redundancy. Embedded hidden layer neurons are served to increase the fitness of the model and -1 node is utilized to limit too big input value for activation function.

Table 1 Parameters used in CNN structure

Parameter Item	Value
Size for the input image	200×200 pixels
Convolutinal layer number	5
5×5 layers	C1, C2, C5
3×3 layers	C3, C6
2×2 layers	C4, C7
Learning rate	0.2
Number of epochs for learning	500
Neurons number of hidden layer	5

2.2 Classification principle by deep learning

In training, the classification model, stochastic gradient descent algorithm is utilized to update connection weighted values and convolution kernels of the model by back propagation with minimal loss function. The whole process of intelligent classification involves two processes, training process and classification process, in which the classification procedure is illustrated in Fig. 2. At the beginning of the process, the sample collections are from comparisons between the measured workpiece and the standard one by machine vision detection in gray image.

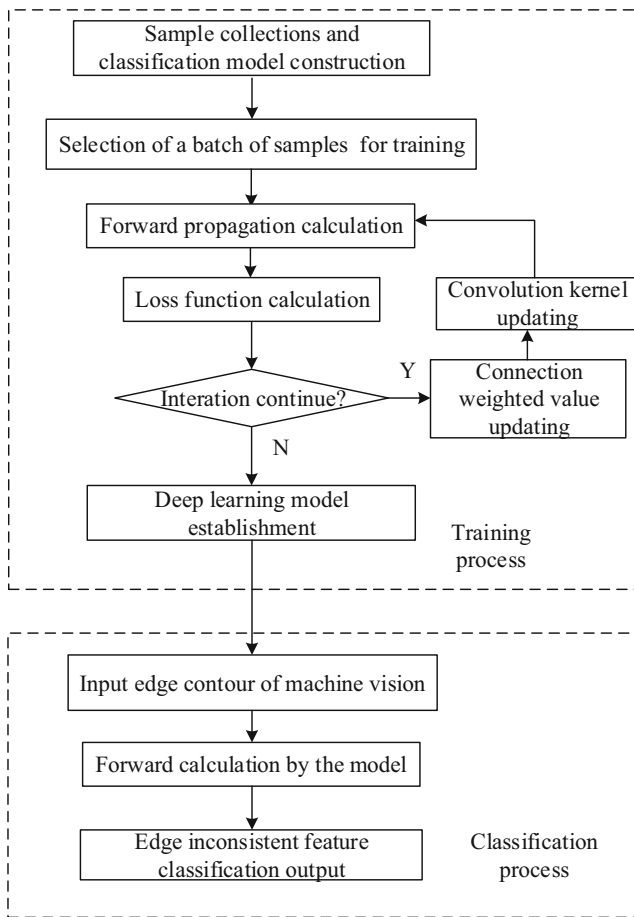


Fig. 2 Process of edge inconsistent feature classification by deep learning

With the compared inconsistent edge image as input for the network, forward propagation calculation by convolution deep learning mechanism is for classification process and backward propagation calculation by weight updating in gradient for training process.

In convolutional layer, suppose ω_{ij}^n denote the weights of a 2-D convolutional kernel associated with n -th layer edge inconsistent feature map, where i is component index of j -th kernel in matrix. Convolution sum calculation is the sum of products of the weights of kernel and elements of the map that are spatially coincident with the kernel. The n -th layer feature map can be expressed by the equation as,

$$x_j^n = f\left(\sum_{i \in M_j} x_i^{n-1} * \omega_{ij}^n + b_j\right), \tag{1}$$

where x_j^n is the j -th feature component of n -th layer and M_j is the j -th input feature map. The convolution sum between feature map x_i^{n-1} and kernel matrix ω_{ij}^n is $\sum_{i \in M_j} x_i^{n-1} * \omega_{ij}^n$. b_j is a scalar for input adjustment of $f(\cdot)$ function. In the

equation, $f(\cdot)$ is activation function for the network, which is also used to transform input of the neurons in the model, with the expression as,

$$f(x) = \text{sigmoid}(x) = \frac{1}{1 + e^{-x}}, \tag{2}$$

where x is input variable of the sigmoid function.

Pooling layers are used to form a unique down-sampling feature map on the output, x_j^{n-1} , of the convolutional layers. The node output expression of the layer is defined as follows:

$$x_j^n = f(\beta_j^n \text{down}(x^{n-1} j) + b_j), \tag{3}$$

where β_j^n represents multiplicative bias of the output feature map for n -th layer, b_j denotes additive bias of the corresponding output of feature map.

Function $\text{down}(\cdot)$ is selected to decline the dimensions of edge inconsistent feature images for salient features and avoidance of overlearning, in which interval point extraction is conducted in every rows and columns from the feature map for transformation of feature size from original size to its half.

With convolutional and down sampling calculation, the input inconsistent image has been transformed to be a vector with extraction features. Forward propagation of the vector is also activated by neurons with sigmoid function,

$$x^n = f(u^n), u^n = \omega^n x^{n-1} + b_j, \tag{4}$$

where n is component number of the input vector, x^n output after calculation of the n -th neurons, u^n total input of the n th layer neurons and ω^n connection weight of the neuron. The connection weight updating is conducted in the gradient direction,

$$\Delta \omega^n = -\eta \frac{\partial E}{\partial \omega^n}, \tag{5}$$

where η is learning rate, E is loss function value, which is also used to describe the deviation degree between the output of networks and of the learning patterns as equation,

$$E = \sqrt{\frac{\sum_{i=1}^n (x_i - y_i)^2}{n}}, \tag{6}$$

where n is number of images in one batch, x_i is output value during the forward propagation for i -th image and y_i is label value for i -th image.

The updating of convolution kernel in learning process is derived from back propagation of loss function value in gradient by equation,

$$\frac{\partial E}{\partial k_{ij}^n} = \delta_j^n \Theta p_i^{n-i}, \tag{7}$$

where k_{ij}^n is j -th kernel matrix for the i -th feature map of the n -th layer, p_i^{n-1} patch of i -th feature map of $(n - 1)$ -th layer. Δ is convolution operation for elements of the arrays.

δ_j^n is deviation sensitivity of n -th convolution layer of j -th feature map, which can be expressed by equation,

$$\delta_j^n = \beta_j^{n+1} (f'(u_j^n) \Theta up(\delta_j^{n+1})), \tag{8}$$

where $up(\cdot)$ function denotes the up-sampling operation to matrix for expansion.

The gradient of the convolutional layer training error to additive bias b_j is obtained from the sum of elements on the deviation sensitivity of the j -th feature map, which can be defined with equation,

$$\frac{\partial E}{\partial b_j} = \sum_{u,v} (\delta_j^n)_{uv}, \tag{9}$$

The gradient of the convolutional layer training error to multiplicative bias β_j is obtained from down-sampling of the feature map by back propagation with convolution operation to the deviation sensitivity of the current layer, which is described by equations,

$$d_j^n = down(x_j^{n-1}), \frac{\partial E}{\partial \beta_j} = \sum_{u,v} (\delta_j^n \Theta d_j^n)_{uv}. \tag{10}$$

3 Variable Geometry model for inconsistent features

On support of the classification model, a variable parametric geometry model for specific edge inconsistency feature detection is established to describe different geometric contours with a parametric vector variable. Indicated in Fig. 3, a general geometric model is established on this parametric variable for detection of the edge inconsistency. Parameter vector P is introduced to express the variables,

$$P = (x, y, l_x, l_s, H, \theta, b_z, b_y, \beta_z, \beta_y, \bar{l}_x, \bar{l}_s), \tag{11}$$

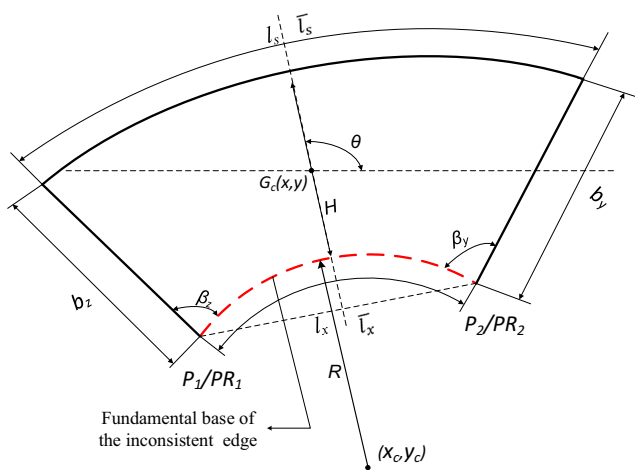


Fig. 3 General parametric geometry model for edge inconsistent feature

where l_x denote arc length of the model base and l_s top of the edge inconsistent feature contour. H stands for height of the contour. θ is azimuth of the geometric model, b_z width of the left side and b_y right side of the contour. β_z , represent include angle between the left side and the bottom side, β_y include angle between the right side and of the contour respectively, \bar{l}_x, \bar{l}_s is base curvature and top curvature of the contour respectively.

The purpose of designing left and right parameters respectively with 12 parameters in P is to better fit edge with inconsistent features in arbitrary geometric contour, and precisely recognize the geometric features of actual contour. The dashed segmental line marked in Fig. 3 is specified as the fundamental base of the inconsistent edges, also called leading edge, which is located on the mechanical workpiece and its geometric parameters can be obtained from the counterpart contour.

In accordance of the provided geometric parameters, there are 8 kinds of sub-models that can be transformed from the general geometric model with different distribution in profile, which are illustrated in Fig. 4. In the model, Azimuth θ and geometric center $G_c(x, y)$ are introduced to assist positioning the feature.

4 Inconsistent feature parameter acquisitions

4.1 Acquisition of azimuth

Azimuth θ is a variable that describes the rotational degree of geometric contour with respect to horizontal position of the image. It is applied to reduce the variation range of parameters in P with faster fitting speed. The angle is obtained by the principle of leading edge estimation provided by [36], which basically begins from determination of the geometric center of the feature. The geometric center $G_c(x, y)$ of the model can be acquired by collecting the coordinates of all contour pixel points, $A_n(x_n, y_n)$, by averaging calculation, which can be expressed with following equations,

$$x = \frac{x_1 + x_2 + \dots + x_n}{n}, y = \frac{y_1 + y_2 + \dots + y_n}{n}. \tag{12}$$

Suppose the double terminal points of the leading edge profile $P_1(x_i, y_i)$ and $P_2(x_j, y_j)$, the slope rate of straight line P1P2 can be define as,

$$K_z = \frac{x_i - x_j}{y_i - y_j}. \tag{13}$$

Then, azimuth θ can be expressed as,

$$\theta = \arctan(K_z) = \arctan \frac{x_i - x_j}{y_i - y_j}. \tag{14}$$

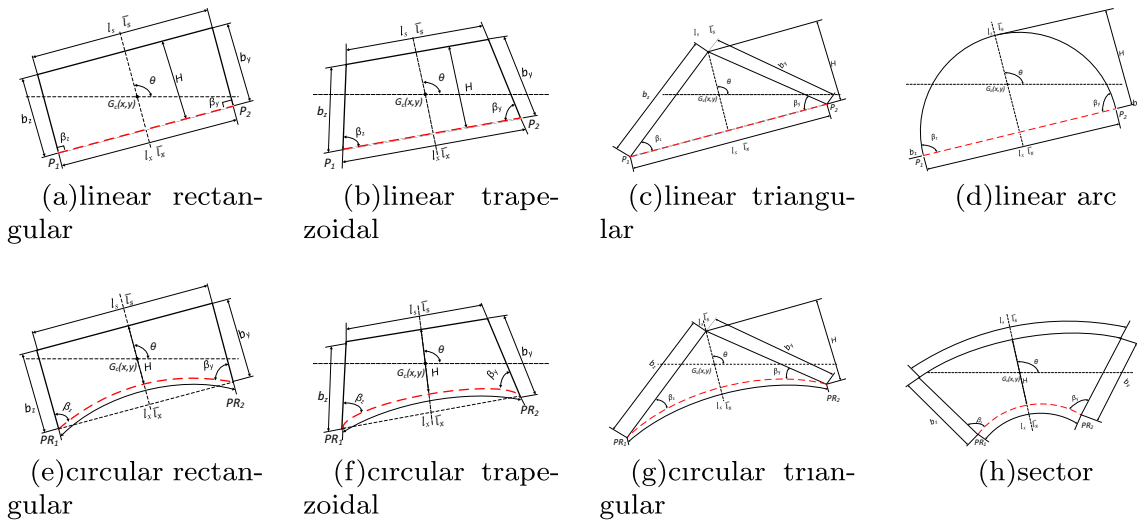


Fig. 4 Transformation of general model to different profile sub-models

4.2 Objective function construction

On the support of azimuth angle, the target feature detection can be derived from identification of the parameters of the geometric model. For this identification, Intersection-over-Union concept [37] is introduced to measure the fitting rate between candidate profile and edge profile from the vision. The fitting rate function is expressed as equation,

$$\max[F_{fitting}(P, S) = \frac{N_c}{N_m} + \frac{N_c}{N_s} - \frac{N_o}{N_s}], \quad (15)$$

where P is 12-parameter vector of the geometric model and S binary contour points set of actual inconsistent edge. N_c is number of pixels of the geometric model in image format, in agreement with the actual edge from the vision in coordinate distribution. N_m is number of pixels of the whole image-format contour points in the model. N_s stands for the number of pixels of actual edge contour and N_o relates to the number of pixels which position falls outside the positions of the target points of the model.

4.3 Function solving with particle swarm optimization

With the fitting function, the relationship among fitting degree of target and related parameters is established. It is indicated that the fitting degree varies with variation of parameters, and that the solution for P with maximum function value is regarded as an optimal solution. It is also shown that when geometric model is fully fitted with actual edge contour, the foregoing double item $\frac{N_c}{N_m} + \frac{N_c}{N_s}$ of the objective function can reach a maximum value, and that the third item, $\frac{N_o}{N_s}$, of the function, as penalty, is for enhancement of the fitting efficiency.

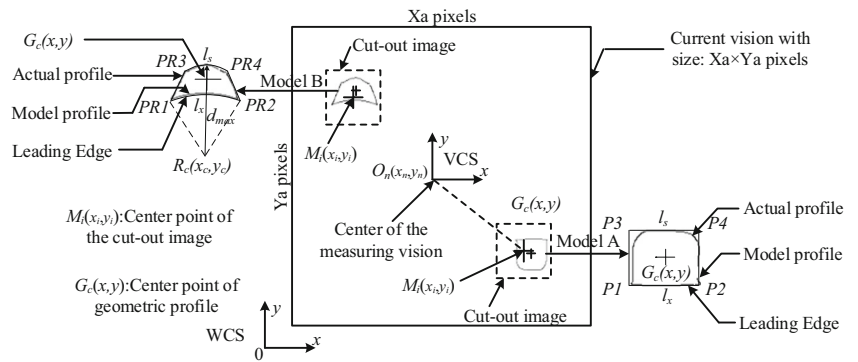
From (15), it is shown that the solution of the optimization problem is in a high complicated dimension space. It is known that particle swarm optimization algorithm(PSO) [38] is with good performance in solving the optimization with complicated solution combination. Thus it is introduced to solve this optimal fitting function. The particle positions are used to denote the parameters of P vectors and the velocities derived from the respective changing amplitudes of the vectors during the neighbouring iterations in the evolution. The fitness function derives from the rightside item of the Equation consisting of the three fractions. The engine of position and velocity updating of the particles by PSO is followed in solving this optimization.

4.4 Positioning of edge inconsistent features

As indicated in the geometric model, there are respectively four types of geometric contours with straight line and arc leading edges. The characteristic of these edge contours is with single-pixel width, which can be described by 2-D point set. According to the point set, geometric parameters of the contour can be used to calculate the dimensions and the position of different edge inconsistencies.

The relationship of pixel coordinate transformation between measuring system and edge feature positioning can be illustrated in Fig. 5. Suppose the current vision size is $X_a \times Y_a$ pixels, and point $O_n(x_n, y_n)$ is center point of the measuring system for the detection vision. The x direction and y direction of work-pieces coordinate system(WCS) are aligned to horizontal and vertical reference direction of the vision respectively. Suppose the coordinate of the center point of a cut-out image is expressed as $M_i(x_i, y_i)$, then the M_i point coordinates in WCS can be obtained as $((x_n +$

Fig. 5 Transformation of the coordinates for inconsistent edge features



$x_i - X_a/2$, $(y_n + y_i - Y_a/2)$). The distance between the center point of geometric profile $G_c(x, y)$ and M_i point can be utilized to determine the specific coordinate of $G_c(x, y)$ in global image coordinate system when the position of M_i point is available. An inconsistent edge feature with straight line as leading edge and a feature with arc line as leading edge are illustrated as model A and model B respectively in the figure for description of the position process.

In positioning the feature of Model A, the linear rectangular type sub-model, shown in Fig. 4 sub-model a, is introduced to fit it. When profile length l_s and l_x are available, the coordinates of point P1-P4 can be obtained. In positioning the feature of model B, fitting of the pixel points of the leading arc line is the first step with length l_x acquisition and center position $R_c(x_c, y_c)$ of circle determination. The sector type sub-model, shown in Fig. 4 sub-model h, is introduced to fit it. The maximal distance from pixel points of the outer arc to center point R_c is defined as the radius of outer arc d_{max} .

5 Detection experiment

5.1 Experimental platform

A domestic inspection experimental platform, illustrated in Fig. 6, is set up to verify the effectiveness of the intelligent recognition method for edge inconsistent features. The platform is constructed with XY motion mechanisms driven by Panasonic MSMJ022G1U servo amplifier and GTS-400-PG-PCI digital position control system, on which the vision detection components, BFLY-PGE-50H5M FLIR industrial camera and LM12HC lens, is installed with $1000mm/min$ maximal motion speed, $500mm$ stroke and $0.06mm$ positioning resolution. The environmental light source is from VL280-W type strip LED light source equipment, which can adjust the irradiation angle and the installation position according to the different test environment. The data transmission between camera and

host computer is based on Gigabit Ethernet interface with a maximum transmission rate of $10Gbit/s$.

The host hardware environment for the experiment is including a data acquisition and processing computer with CPU(Intel@Xeon@E5-2650 0 @ $2.00GHz$ $2.00GHz$ dual processor), $32GB$ RAM, Windows 7. The programming environment MATLAB R2015b is employed for deep learning, and image initial processing programming is conducted in Microsoft Visual Studio 2010 support by OpenCV library.

5.2 Image data set establishment

In image data set acquisition process, camera lens is set up with its measuring direction perpendicular to the XY plane of platform and $200mm$ distance from lens to the target workpiece. With these basic setups, the maximum image resolution of the camera is up to 2448×2048 pixels.

A cavity-type workpiece illustrated in Fig. 7 is adopted for investigation. The camera motion scheme for image data set acquisition is formulated to be along with the edge of a standard workpiece. In image data sampling, the pixel

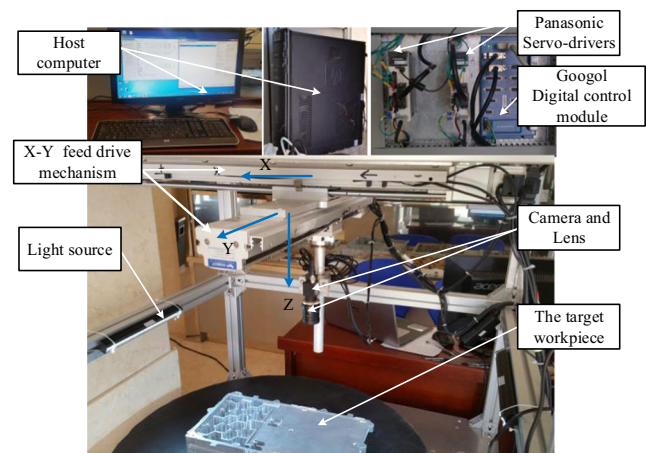


Fig. 6 Platform for machine vision under servo drive

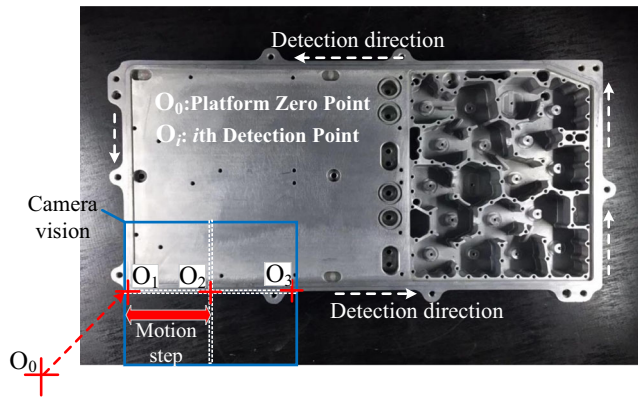


Fig. 7 Detection method for image data set establishment

accuracy reaches $0.06mm/pixel$, corresponding to a single image size $142 \times 119mm$.

It is illustrated in the figure that the tiny-line cross stars are specified as the center position of the detection image, the rectangle frame indicates the size of vision, and the dashed cross in the vision is applied to align the contour edge of the workpiece. In image acquisition, the camera absolute motion begins from the absolute zero point of workpiece O_1 which is with absolute coordinate in the platform coordinate system. One step motion distance for detection is set to be an half of the size of the vision. Images from different workpieces are collected in the sequence according to the detection direction indicated by the dashed arrow in the figure.

Due to the influence of illumination and electricity characteristic of the equipment, the original image contains much noise. In order to effectively low down the influence

of noise on measurement of edge inconsistent features of the part, some preprocessing methods are employed to generate the inconsistent edge image data. Median filtering method [39] is applied to process the original images. Considering the local Reflection of metal material under illumination, Retinex method [40] is utilized to depress the effect of surface Reflection, and adaptive threshold segmentation method [41] is introduced to generate binary images. To cut out the redundant part of images, differential algorithm [42] is employed to extract the region with single inconsistent features of the edge compared with the standard workpiece. Then the contour of inconsistent feature is identified by canny operator. The whole processing steps and their results for two processing cases are illustrated in Fig. 8. In the figure, steps a1-f1 are for processing of edge inconsistent feature with linear leading edge, and steps a2-f2 for feature with arc leading edge.

In the image data set establishment, inconsistent feature contours are segmented into the size with $200 \times 200pixels$. In equalization of the samples there are 50 images of linear rectangular type, 50 images of linear trapezoidal type, 50 images of linear triangular type, 50 images of linear arc type, 50 images of circular rectangular type, 50 images of circular trapezoidal type, 50 images of circular triangular type, 50 images of sector type selected for original data source. Figure 9 shows 10 pieces of edge inconsistent feature images of each kind for indication of the data sample. Moreover, for the sake of enhancement of the generality of proposed model, rotations of these images with arbitrary angles and mirroring in horizontal or vertical direction are conducted to generate a dataset with 16800 images.

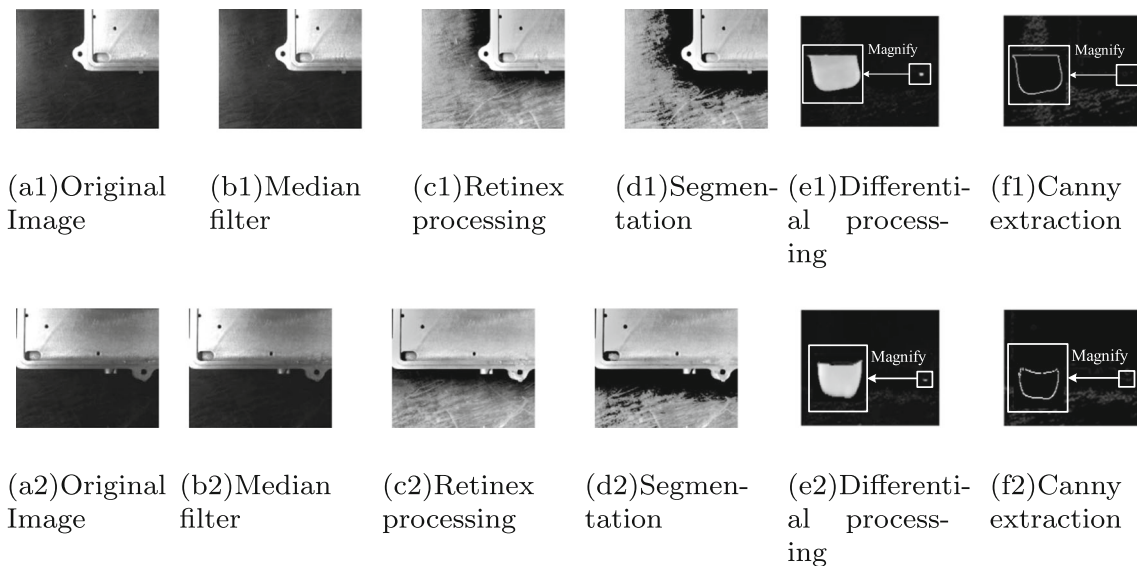


Fig. 8 steps of image preprocessing for the two cases

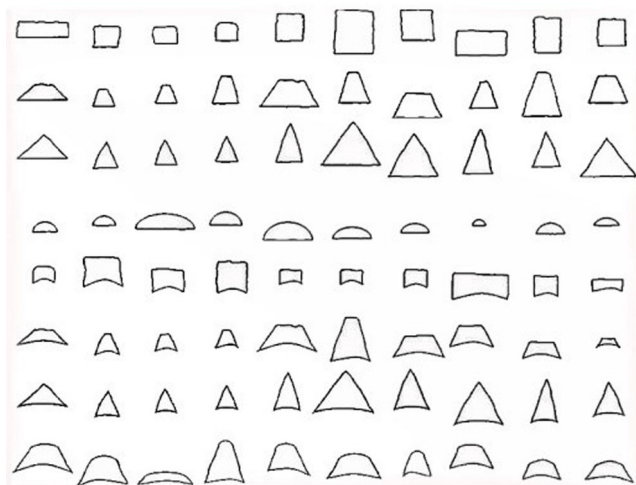


Fig. 9 illustration of part data set

5.3 Experiment strategy

For verification of the feasibility of the proposed detection scheme, experiment strategy, illustrated in Fig. 10, is employed with the cavity-type workpiece as detected target on the experimental platform. The experiment involves three steps, detection preparation step for image preprocessing, classification step for feature type detection and measurement step for feature geometric parameters acquisition. 50 detection images in per type of sub-models, 400 images in all, are selected as test sample for the verification. In the training process, 50 images as a learning batch are treated as one-time iteration.

6 Results and analysis

With the data sets, 500 iteration calculations are conducted for optimal loss function in machine learning. The relationship between the objective function and iterations is illustrated in Fig. 11. It is shown in the figure that the fluctuation of the mean square error in the training roughly experiences a growth-decline-stabilization process. In the beginning iterations, the objective function value updates in an opposite direction for the relatively single direction for adjustment of weights with batches of image inputs. With increase of the iteration number, regular transformation of

the weights is in effect. Before 150 times iteration the error converges with high speed and then it falls slower. It illustrated that when 500 times iterations are applied, the value reach nearly to zero. This process also shows that the built deep learning networks is with good training performance, and better in training performance than the CNNs with other try parameters combination as test. Some training results are listed out in Table 2 as comparisons with MSE loss at the 500th iteration.

The regression values derived from the better selected CNN based deep learning classification model of the different test datasets are illustrated in Table 3. Then the values are inputted into the K-means clustering algorithm to acquire the classification result about its geometric contour type with the consequence demonstrated in the table. It is shown that the maximum deviation of the output value among these 8 group results is 9.9e-7. With the k-means process, the classification accuracy can reach 100%.

To show advantages of the proposed combined classification model, comparisons to conventional softmax model [43] are conducted with the similar data set and test samples. The comparison results, which are illustrated in Fig. 12, shows that the average classification accuracy of the proposed model, 100%, is higher than the conventional model, 98.17%. The 1.83% difference indicates the possibility with incorrect classification by the conventional method.

Minimum profiles enclosing edge inconsistent features of the cavity is gained by adjusting parameters in the variable geometric model with particle swarm optimization algorithm. Performance of the fitting degree between model and actual profile is evaluated by dice similarity coefficient (DM) [44] and Hausdorff distance (HD) [45] The dice similarity coefficient provides a close relationship between the actual profile and the model profile, and the DM value normally locates in the range of 0-1. A higher DM value indicates a better match condition between the two profile boundaries. The distance indicates a symmetric distance measure of the maximum discrepancy between the two profile boundaries. A smaller HD value indicates higher similarity of the shapes between the two profiles, which is expressed as,

$$DM = \frac{2|M \cap A|}{|M| + |A|}, \tag{16}$$

Fig. 10 Experiment strategy for intelligent detection with three steps

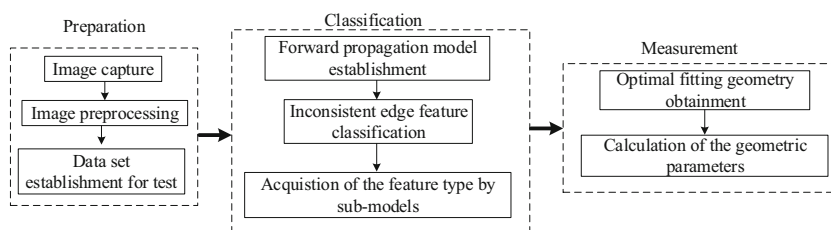


Fig. 11 Change of the Mean Square Error during training period

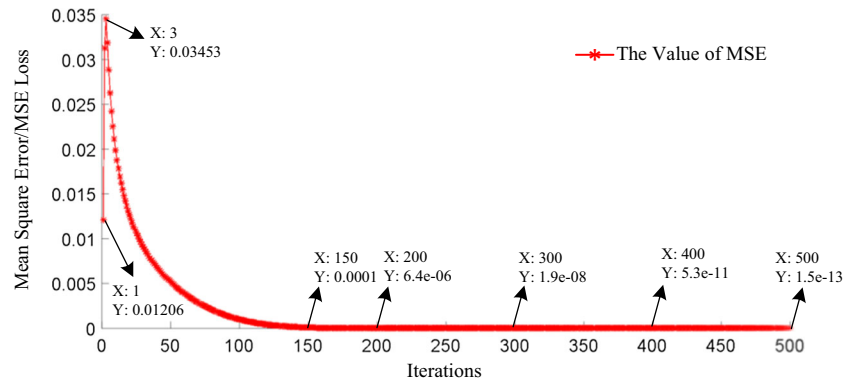


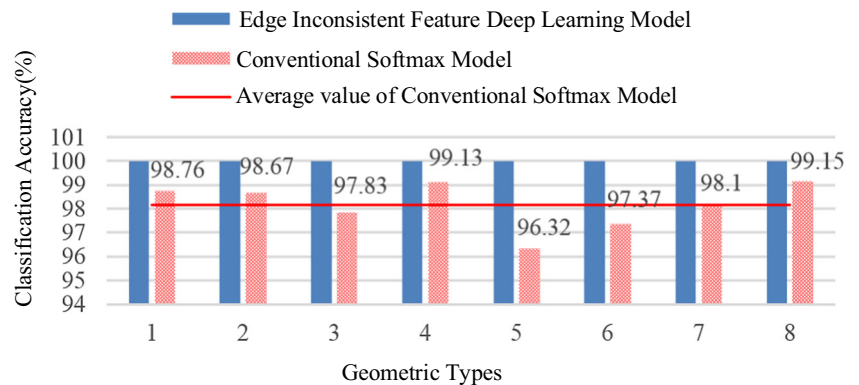
Table 2 Comparisons in training performance for some CNNs with different parameter combination

Some CNNs for try and test	MSE Loss at 500th iteration
The selected parameters with try-and-error combination	1.5e-13
Different learning rate(=0.1)	3.3e-2
Different size of convolution kernels(=19x19/5x5/3x3)	2.7e-2
Different number of convolution layers(=3)	2.5e-6
Different number of hidden neurons(=15)	3.0e-5

Table 3 Maximum deviation for regression value in classification model

Geometric type	Sample size	Image label value	Maximum deviation value
Linear rectangular	50	1	2.2e-7
Linea trapezoidal	50	2	9.9e-7
Linear triangular	50	3	9.9e-7
Linear arc	50	4	9.6e-7
Arc rectangular	50	5	9.6e-7
Arc trapezoidal	50	6	9.5e-7
Arc triangular	50	7	9.8e-7
Sector	50	8	9.9e-7

Fig. 12 Classification performance comparison of the edge inconsistent feature deep learning model and CNN



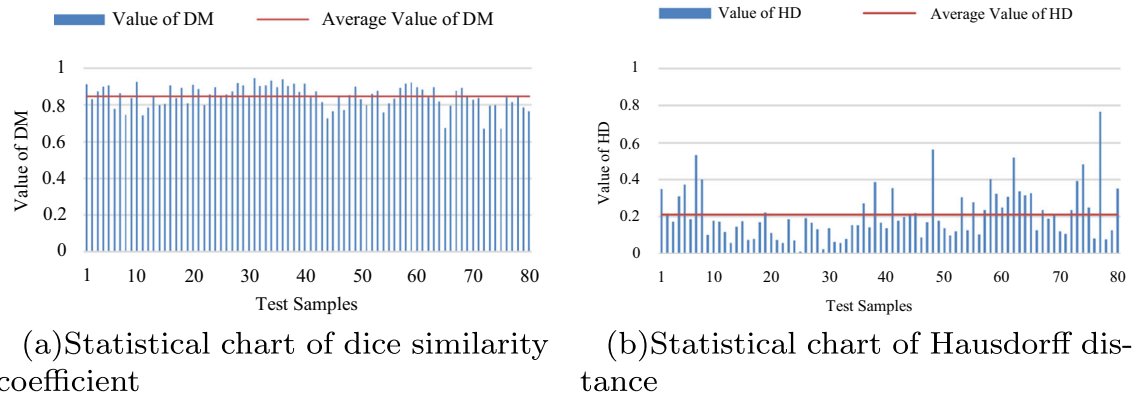


Fig. 13 Similarity evaluation of the actual contour and the fitted contour

where M is on behalf of a set of boundary points that fit the profile with a variable geometric model, A is the boundary point set for the actual profile of the inconsistent features of. $|M|$ represents the area enclosed by the points set of the fitted profile and $|A|$ stands for the area enclosed by points set of the actual profile.

For quantitative evaluation of the performance of the combined method, 10 arbitrary sample are selected from 8 type of features to generated the double index, which are demonstrated in Fig. 13 with addition of their mean value. It is shown in the figure that the average dice similarity coefficient between the model and actual profiles of edge inconsistent features of the target workpiece is up to 0.84, indicated by the horizontal solid line in Fig. 13a, and their average Hausdorff distance 0.21, indicated by the horizontal solid line in Fig. 13b. It is also found that the profiles of line-based model are more similar to their actual contours than profiles of arc-based model.

The detection performance for actual edge inconsistent feature of the workpeice is indicated by the double combined

evaluation index from the test sample set. The combined index is derived from the equation,

$$COMI = \frac{DM}{HD}. \tag{17}$$

The evaluation comparisons of all the test samples are illustrated in Fig. 14. It is shown in the figure that the minimal COMI of straight line-lead insistent contour features is at No.27 sample, which is indicated by Point PL with 1.62 COMI value. In similar way, Point PA indicates the minimal COMI of arc-lead insistent contour features at No.387 sample with 1.07 COMI value.

The detail fitting performance in detection of geometric inconsistent features of the two samples is illustrated in Table 4. It is shown in the table that the straight line-lead and the arc-lead edge inconsistent geometric features with maximum recognition error in width is 0.06mm and 0.04mm respectively, and in height 0.02mm and 0.04mm.

Some recent intelligent methods based on machine vision processing for feature detection of other objects

Fig. 14 Combination Evaluation Index Comparison of test samples

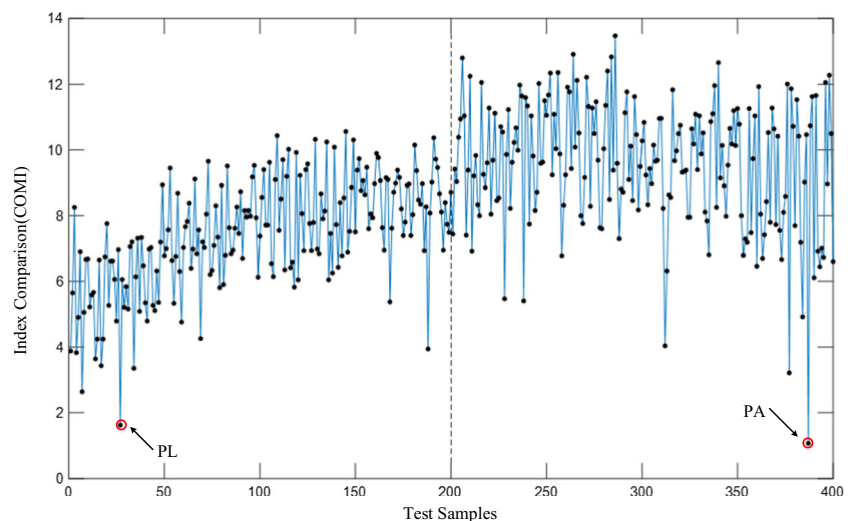
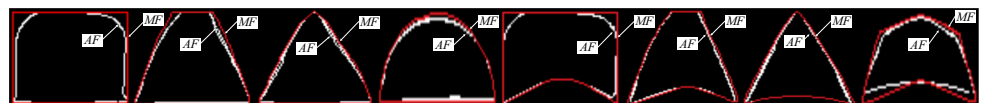


Table 4 Geometrical parameter of Inconsistent Feature by the approach(mm)

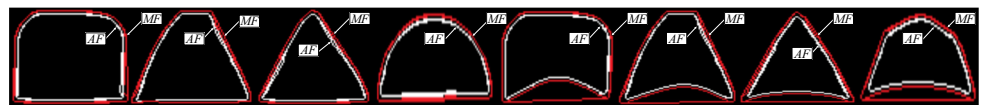
Sample	Width			Height			DM	HD	COMI
	Meas.	Actual	Error	Meas.	Actual	Error			
PL	4.14	4.20	0.06	3.30	3.28	0.02	0.86	0.53	1.62
PA	3.78	3.74	0.04	1.80	1.84	0.04	0.81	0.76	1.07

Table 5 Comparisons of recognition performance in profile fitting among candiated methods

No.	Method	Average dice similarity coefficient	Average hausdorff distance
1	The Proposed Method	0.84 ± 0.08	0.21 ± 0.09
2	Two-phase Segmentation [7]	0.81 ± 0.09	0.22 ± 0.12
3	Graph Cut method [8]	0.80 ± 0.16	0.50 ± 0.47
4	Sliding Comparison Window [46]	0.81 ± 0.03	0.34 ± 0.05
5	Adaptive Background template [47]	0.79 ± 0.12	0.28 ± 0.07
6	Single Rectangular Box [48]	0.78 ± 0.17	0.26 ± 0.04
7	Golden Image Subtraction [49]	0.80 ± 0.11	0.43 ± 0.21
8	Active Contour method [50]	0.81 ± 0.05	0.22 ± 0.08

Fig. 15 Performance comparison of the approach and active contour

(a) Edge inconsistent feature geometric contour fitting by the proposed method



(b) Edge inconsistent feature geometric contour fitting by the active contour method

are introduced into this research as candidate method with contour fitting. The comparisons of the performance among these methods are conducted with recognition performance evaluated by the double indexes. The comparison results, illustrated in Table 5, indicate the highest average DM and the lowest HD index of the proposed method. It also means the proposed method is with good performance in low mismatching and high efficiency in feature extraction.

Since the performance of active contour method is on the second level in these comparisons, the method is introduced to be further compared with the proposed method in fitting the actual profiles for some specific samples. Figure 15 indicates the fitting results of 8 type features in subfigure (a) by the proposed method, and subfigure (b) from active contour method. In the figure, MF is the fitting profile from methods and AF is actual profile of the target features. Although the fitting profile generated by active contour method is not with low similarity to the actual profile, it generates multiple boundaries in some areas, which results in no clear boundary for fitting profile. In addition, the comparison between the corresponding sub-figures in (a) and (b), illustrates that the proposed method is with better fitting performance.

As a result, the combined model can classify the different type of geometric contour of edge features with 100% correctness, with 0.84 average dice similarity index metric compared between the results of model and the actual edges, 3.70% higher than the result of active contour method. Meanwhile, the average Hausdorff Distance between the double compared object falls at 0.21 with 4.55% lower than that of active contour method. It is also shown that the presented method can provide a good recognition of the geometrical shape with less than 0.06mm maximum error for workpiece with $142 \times 119\text{mm}$ size in the visual field. Based on the above analysis, it can be concluded that it is feasible to apply deep learning and variable parametric geometric model to intelligent detection of edge inconsistencies with high accuracy of mechanical parts. As a result, the proposed method presents a contribution to improve the intelligent detection level in machining of batch workpieces.

7 Conclusion

A novel method applying deep learning model and variable geometric model is presented to improve the correctness and precision of intelligent detection of edge inconsistent features. In accordance with geometric characteristics of inconsistent edge features of the parts, a single pixel-based edge data set is established by image preprocessing. A hidden layer with -1 neuron is introduced to scale down the error between regression output of the network and the label value. The K-means clustering algorithm is

utilized to make the classification model with accuracy up to 100%. The variable parametric geometric model is designed to fit edge inconsistent feature contour of the classified results. The geometric parameters of the model are obtained by calculating the correlation parameters based on particle swarm optimization. Supported on the domestic experimental system, test experiments to a cavity-type mechanical workpiece are carried out. The results show that the combined model can classify the different type of geometric contour of edge features with 100% correctness, with 0.84 average dice similarity index metric compared between the results of model and the actual edges, 3.70% higher than the result of active contour method. It is also shown that the presented method can provide a good recognition of the geometrical shape with less than 0.06mm maximum error for workpiece with $142 \times 119\text{mm}$ size in the visual field. In the future work, automatic division of the whole detection vision with gray contrast in the contour for positioning inconsistency into small recognition patterns is still an important task to put the methodology into application.

Acknowledgments This work was supported by National Natural Science Foundation of China [grant number 51005158]; and “High-end Numerical Control Machine and Basic Manufacturing Equipment” Key Sic-tech Special Project [grant number 2011ZX04003-031].

References

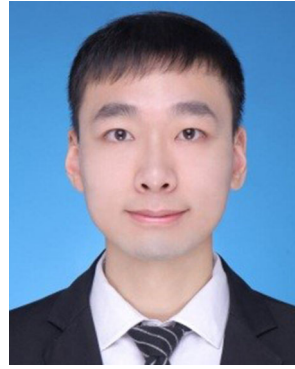
1. Qiu H, Li Y, Li Y, new method A (2001) device for motion accuracy measurement of NC machine tools part 2: device error identification and trajectory measurement of general planar motions. *Int J Mach Tools Manuf* 41(4):535–554
2. Zhang Y, Lefebvre D, Li QL (2017) Automatic detection of defects in tire radiographic images. *IEEE Trans Autom Sci Eng* 14(3):1378–1386
3. Thongkamwitoon T, Muammar H, Dragotti PL (2015) An image recapture detection algorithm based on learning dictionaries of edge profiles. *IEEE Trans Inform Forens Secur* 10(5):953–968
4. Chong Y, Song YH, Zhang YL (2016) Scene text localization using edge analysis and feature pool. *Neurocomputing* 175:652–661
5. Chen TJ, Wang Y, Xiao CY, Wu QMJ (2016) A machine vision apparatus and method for can-end inspection. *IEEE Trans Instrum Measur* 65(9):2055–2066
6. Liu HW, Yin JP, Luo XD, Zhang SC (2018) Foreword to the special issue on recent advances on pattern recognition and artificial intelligence. *Neural Comput Appl* 29(1):1–2
7. Fan J, Wang J (2018) A two-phase fuzzy clustering algorithm based on neurodynamic optimization with its application for PolSAR image segmentation. *IEEE Trans Fuzzy Syst* 26(1):72–83
8. Kusakunniran W, Wu Q, Ritthipravit P (2018) Hard exudates segmentation based on learned initial seeds and iterative graph cut. *Comput Methods Programs Biomed* 158:173–183
9. Tsai DM, Hsieh YC (2017) Machine vision-based positioning and inspection using expectation-maximization technique. *IEEE Trans Instrum Meas* 66(11):2858–2868

10. Han Y, Wu YB, Cao DH (2017) Defect detection on button surfaces with the weighted least-squares model. *Front Optoelectron* 10(2):151–159
11. Michal S, Bartosz P, Marcin M (2016) Machine vision micro-milling tool wear inspection by image reconstruction and light Reflectance. *Precis Eng* 44:236–244
12. Guo G, Wang Y, Jiang TT (2014) A shape reconstructability measure of object part importance with applications to object detection and localization. *Int J Comput Vis* 108(3):241–258
13. Shahabi HH, Ratnam MM (2010) Prediction of surface roughness and dimensional deviation of workpiece in turning: a machine vision approach. *Int J Adv Manuf Technol* 48(1-4):1213–1226
14. Cano T, Chapelle F, Lavest JM, Ray P (2008) A new approach to identifying the elastic behaviour of a manufacturing machine. *Int J Mach Tools Manuf* 48(14):1569–1577
15. Koroglu MT, Passino KM (2014) Illumination balancing algorithm for smart lights. *IEEE Trans Control Syst Technol* 22(2):557–567
16. Wang JJ, Ma YL, Zhang L, Gao RX, Wu DZ (2018) Deep learning for smart manufacturing: methods and applications. *J Manuf Syst* 48:144–156
17. Fujita H, Cimr D (2019) Decision support system for arrhythmia prediction using convolutional neural network structure without preprocessing. *Appl Intell* 49:3383–3391. <https://doi.org/10.1007/s10489-019-01461-0>
18. Zhou F, Yang S, Fujita H et al Deep learning fault diagnosis method based on global optimization GAN for unbalanced data. *Knowledge-Based Systems*, <https://doi.org/10.1016/j.knosys.2019.07.008>
19. Park JK, Kwon BK, Park JH, Kang DJ (2016) Machine learning-based imaging system for surface defect inspection. *Int J Precis Eng Manuf-Green Technol* 3(3):303–310
20. Zhou SF, Shen W, Zeng D (2016) Spatial-temporal convolutional neural networks for anomaly detection and localization in crowded scenes. *Signal Process: Image Commun* 47:358–368
21. Hu ZL, Tang JS, Wang ZM, Zhang K, Zhang L (2018) Deep learning for image-based cancer detection and diagnosis a survey. *Pattern Recogn* 83:134–149
22. Wen CB, Liu PL, Ma WB (2018) Edge detection with feature re-extraction deep convolutional neural network. *J Vis Commun Image Represent* 57:84–90
23. Diao WH, Xian S, Dou FZ, Yan ML, Wang H, Fu K (2015) Object recognition in remote sensing images using sparse deep belief networks. *Remote Sensing Lett* 6(10):745–754
24. Chowdhury A, Kautz E, Yener B (2016) Image driven machine learning methods for microstructure recognition. *Comput Mater Sci* 123:176–187
25. Liu L, Fieguth P, Guo Y (2017) Local binary features for texture classification: taxonomy and experimental study. *Pattern Recogn* 62:135–160
26. Sarkar S, Venugopalan V, Reddy K (2017) Deep learning for automated occlusion edge detection in RGB-D frames study. *J Signal Process Syst Signal Image Video Technol* 88:205–217
27. Zhang XS, Gao T, Gao DD (2018) A new deep spatial transformer convolutional neural network for image saliency detection. *Des Autom Embed Syst* 22(3):243–256
28. Hoang ND, Nguyen QL, Tran VD (2018) Automatic recognition of asphalt pavement cracks using meta heuristic optimized edge detection algorithms and convolution neural network. *Autom Constr* 94:203–213
29. Zhu ZX, Rao C, Bai S, Latecki LJ (2019) Training convolutional neural network from multi-domain contour images for 3D shape retrieval. *Pattern Recogn Lett* 119:41–48
30. Mao YH, Shen JJ, Gui XL (2018) A study on deep belief net for branch prediction. *IEEE Access* 6(99):10779–10786
31. Gu JX, Wang ZH, Jason K, Ma LY, Amir S, Bing S, Liu T, Wang XX, Wang G (2015) Recent advances in convolutional neural networks. *Comput Sci* 77:354–377
32. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444
33. Bengio Y (2009) Deep learning architectures for AI. *Found Trends Mach Learn* 2(1):1–127
34. Chen LH, Xu ZS, Wang H, Liu SS (2016) An ordered clustering algorithm based on K-means and the PROMETHEE method. *Int J Mach Learn Cybern* 9(6):1–10
35. Hu YC, Huan C, Nian FD, Wang Y, Li T (2016) Dense crowd counting from still images with convolutional neural networks. *J Vis Commun Image Represent* 38:530–539
36. Zhou W, Xie JH, Li GP, Yuan YS (2017) High-precision estimation of target range, radial velocity, and azimuth in mechanical scanning LFM CW radar. *IET Radar Sonar Navigation* 11(11):1664–1672
37. Li HL, Meng FM, Luo B, Zhu SY (2014) Repairing bad segmentation using its quality evaluation and segment propagation. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 23(8):3545–3559
38. Li S, Cheng C (2017) Particle swarm optimization with fitness adjustment parameters. *Comput Industr Eng* 113:831–841
39. Kuo YL, Tai CW (2015) A simple and efficient median filter for removing high-density impulse noise in images. *Int J Fuzzy Syst* 17(1):67–75
40. Gao YY, Hu HM, Li B (2018) Naturalness preserved nonuniform illumination estimation for image enhancement based on retinex. *IEEE Trans Multimed* 20(2):335–344
41. Khambampati AK, Liu D, Konki SK, Kim KY (2018) An automatic detection of the ROI using Otsu thresholding in nonlinear difference EIT imaging. *IEEE Sensors J* 18(12):5133–5142
42. Suresh S, Lal S (2017) Modified differential evolution algorithm for contrast and brightness enhancement of satellite images. *Appl Soft Comput* 61:622–641
43. Wang F, Liu WY, Liu HJ, Cheng J (2018) Additive margin softmax for face verification. *IEEE Signal Process Lett* 25(7):926–930
44. Sudan J, Son LH, Raghvendra K, Ishaani P, Florentin S, Long HV (2019) Neutrosophic image segmentation with dice coefficients. *Measurement* 134:762–772
45. Huttenlocher DP, Klanderma GA, Rucklidge WJ (1993) Comparing images using the Hausdorff distance. *IEEE Trans Pattern Anal Mach Intell* 15:850–863
46. Rong D, Rao XQ, Ying YB (2017) Computer vision detection of surface defeat on oranges by means of a sliding comparison window local segmentation algorithm. *Comput Electron Agric* 137:59–68
47. Lin HF, Li J, Zhou PY, Liang DC, Li DM (2017) Saliency detection using adaptive background template. *IET Comput Vis* 11(6):389–397
48. Zhu WB, Luo ZX, Lim A, Oon WC (2016) A fast implementation for the 2D/3D box placement problem. *Comput Optim Appl* 63(2):585–612
49. Jing JF, Chen S, Li PF (2016) Fabric defect detection based on golden image subtraction. *Color Technol* 133:26–39
50. Garrido L, Guerrieri M, Igual L (2015) Image segmentation with cage active contours. *IEEE Trans Image Process* 24(12):5557–5566

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Xiankun Lin received the B.Sc. and M.Sc. in fluid mechanical engineering in 1998 and 2001, respectively, both from Jiansu University, China. He received the Ph.D in mechanical engineering from Tongji University, Shanghai, China. He is associate professor with the School of Mechanical Engineering, University of Shanghai for Science and Technology. His research interest is high precision drive for machine tools, intelligent measurement for precision machining.



Li Li M.Sc. received the B.Sc in mechanical engineering in 2016 from Hubei University of Engineering. He is the M.Sc candidate in the School of Mechanical Engineering, University of Shanghai for Science and Technology. His research interest is intelligent measurement for precision machining.



Xin Wang received the B.Sc in mechanical engineering in 2016 from Changshu College of Technique and the M.Sc. in mechanical engineering in 2019 from University of Shanghai for Science and Technology. His research interest is intelligent measurement for precision machining.