Automatic Pearl Classification Machine Based on a Multistream Convolutional Neural Network

Qi Xuan, Binwei Fang, Yi Liu, *Member, IEEE*, Jinbao Wang, Jian Zhang, Yayu Zheng, and Guanjun Bao, *Member, IEEE*

***Abstract*—In this paper, we design an automatic pearl classification machine, composed of four parts: feeding mechanism, delivering mechanism, vision-based detection device, and classification mechanism. Pearls can be delivered to the detection device one by one, where multiview images of each pearl can be collected. A novel multistream convolutional neural network (MS-CNN) is developed to cope with these multiview images, with each stream processing an image of particular viewing angle and different streams sharing part of weights to fuse high-order features without losing too much diversity. Using the machine, we collect 52 500 multiview images for 10 500 pearls, i.e., each pearl has five images of top, left, right, main, and rear views. These pearls were labeled by the experienced professionals in advance, and grouped into two classes with rough rules and seven classes with fine rules. Experimental results show that, compared with the support vector machine and backpropagation neural network, our MS-CNN behaves much better in both classification tasks, obtaining 92.14% and 91.24% accuracies. Moreover, the visualization of activations of convolutional kernels suggests that MS-CNN, imitating the manual process, can indeed recognize relatively complex features. These results indicate the potential value of our machine in the pearl industry.**

***Index Terms*—Convolutional neural network (CNN), deep learning, fine-grained classification, machine learning, pearl classification machine, textural feature.**

I. Introduction

PEARLS are the products of some mollusks, and they are very popular organic gems. The value of the pearl is related to many attributes or features, such as size, luster, shape, and texture. To the best of our knowledge, most pearl-producing companies rely mainly on manual classification after they harvest a large number of cultured pearls. However, multiple features of pearls should be considered simultaneously for such a manual classification task, which is relatively difficult in reality. Moreover, manual work is repetitive, monotonous, and inefficient. Therefore, it is considered to be urgent to extract the classification criteria automatically.

Machine learning is good at learning from a particular scene, i.e., learning a specific pattern and applying the pattern in the scene [1], [2]. For a traditional supervised learning method, the solution typically consists of two steps. First, discovering useful features from raw data such as images, since raw data are usually complex, redundant, and highly variable. Second, taking these features as input and constructing a classification model using some method, such as support vector machine (SVM) [3], backpropagation neural network (BPNN) [4], and so on. However, designing traditional hand-crafted features often requires expensive human labor and often relies on expert knowledge. Recently, deep learning [5] is becoming a hot topic in the area of machine learning, leading a series of breakthroughs in both academia and industry. One typical structure achieving great success in computer vision is a convolutional neural network (CNN) [6]. The major advantage of CNN is that it can automatically extract features, from images or videos, to be used for classification.

For image classification, hand-crafted features are useful and easy to design only if the classification rules are simple. However, for pearl classification, it is quite difficult to obtain clear classification rules, let alone to design the effective hand-crafted features. Thus, using CNN to automatically generate pearl features and further realize the pearl classification will be beneficial for the pearl industry, in terms of saving considerable labors and improving the classification performance. Generally, pearl classification can be considered as a fine-grained recognition task [7]–[9], which is quite challenging since the visual differences among the categories are very small and can be overlooked without checking the full views of pearls. Therefore, it is necessary to get the images of different viewing angles for the same pearl. This also means that the collected data will be more complex and informative, increasing the difficulty to design classification algorithms. Traditional CNN is often designed to deal with a single image. To make full use of the collected images from different viewing angles for the same pearl, it is necessary to construct a network architecture, which can deal with multiple views.

Manuscript received July 16, 2017; revised October 24, 2017; accepted November 20, 2017. Date of publication December 18, 2017; date of current version April 2, 2018. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61572439 and Grant 61273212 and in part by the NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization under Grant U1509212. *(Corresponding author: Guanjun Bao.)*

Q. Xuan, B. Fang, J. Wang, J. Zhang, and Y. Zheng are with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: xuanqi@zjut.edu.cn; fangbinwei@qq.com; jinbaowang_zjut@yeah.net; zj_1994@outlook.com; yayuzheng@zjut.edu.cn).

Y. Liu and G. Bao are with the College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou 310014, China (e-mail: yliuzju@zjut.edu.cn; gjbao@zjut.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2017.2784394

0278-0046 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

In this study, we focus on designing a pearl classification machine using the technologies including machine vision and deep learning. The machine consists of four parts: feeding mechanism, delivering mechanism, vision-based detection device, and classification mechanism. The feeding and delivering mechanisms are used to automatically deliver the pearls one by one to the detection device; the detection device is a monocular multiview image acquisition system containing a camera and four plane mirrors. Due to the existence of mirrors, a full-view image can be obtained for each pearl in proceeding, based on which we can get five individual images of different viewing angles, i.e., top, left, right, main, and rear, by using certain segmentation algorithm. We classify the pearls using a novel multistream CNN (MS-CNN), taking these multiview images as input, and then deliver the pearls of the same class into the same box by utilizing the classification mechanism.

A number of pearls are collected with their labels provided by the experienced professionals, i.e., they are classified into two classes by rough rules, and also classified into seven classes by fine rules. These sampled pearls are then put into our machine to collect their images of five different viewing angles. These images as well as the corresponding labels are used to train the MS-CNN model, making it capture the manual classification criteria. In application, when a new pearl is in proceeding, the delivering mechanism delivers it into the detection device, where the multiview images of the pearl are collected. These images are then fed into the MS-CNN model and the label of the pearl is generated as the output. Finally, the classification mechanism delivers the pearl to the corresponding box based on the label. Note that, to process the multiview images of the pearl simultaneously, we propose a novel CNN architecture containing five streams. Images of five views are taken as the inputs to the respective streams. One element-wise layer and two fully connected layers are then used to fuse the feature maps learned by the five streams. The output of the last fully connected layer is fed into a softmax layer, producing a distribution over the classification labels.

We evaluate our newly proposed network architecture based on the labeled pearls. Since the pearls can be either classified into two classes by rough rules or seven classes by fine rules, we train two MS-CNN models to classify pearls into two classes and seven classes. Our MS-CNN models are trained on GeForce GTX TITAN X. We find that our method achieves 92.14% and 91.24% test accuracies for two-class and seven-class classification tasks, respectively, both of which outperform the SVM and BPNN methods. Moreover, these accuracies even increase to 93.90% and 92.57%, respectively, when a weighted combination strategy is adopted to pay more attention to the left and right views.

The main contributions of this paper are summarized as follows. First, we design a pearl classification machine, which can collect pearl images of five different viewing angles and realize pearl classification automatically. The wide application of this machine may help the pearl industry save a large number of labors and formulate a unified classification standard. Second, we propose a novel MS-CNN for pearl classification, which achieves relatively high accuracy and outperforms SVM and BPNN in both two-class and seven-class classification tasks. This novel structure of CNN may also be used in other applications where multiview images need to be handled, e.g., three-dimensional (3-D) CAD models.[1]

II. Related Work

Due to the peculiar nature of the pearl industry, there are few studies about pearl classification. The key to pearl classification is the extraction of features, which typically can be realized by two approaches in machine learning.

Traditionally, many computer vision tasks used hand-crafted features, such as shape and texture features. For instance, the shape features can be described as Fourier descriptor (FD) [10] or Zernike moments [11]. FD can be used as a representation of two-dimensional closed shapes, independent of its location, scaling, rotation, and starting point. The low-order Zernike moments can represent the whole shape of the image and the high-order Zernike moments can describe the details. The texture features can be obtained through a gray-level co-occurrence matrix (GLCM) [12], i.e., various statistical indices of the matrix can be used as texture features. At present, most of the studies on pearl classification just used one kind of features, e.g., pearl shape recognition [13] and pearl luster classification [14]. However, such tasks are quite different from the pearl classification required in the industry, which involves different kinds of features simultaneously and thus is much more complicated. Moreover, hand-crafted features always need expert knowledge, the extraction of which, thus, is relatively expensive and inefficient.

Recently, deep learning provides an effective framework for various tasks in many areas. In particular, CNN can be used to extract very expressive features in an adaptive manner on visual recognition tasks. Due to large public image datasets and the advances in hardware, CNN is becoming the mainstream in visual recognition. It is worth mentioning that ImageNet Large Scale Visual Recognition Competition (ILSVRC) [15], [16] plays an important role in creating advanced deep visual recognition architectures, and many successful network architectures were proposed in ILSVRC, such as AlexNet [17], VGG [18], GoogleNet [19], and ResNet [20]. Due to the effectiveness of deep learning methods, more and more researchers apply these methods in their own domains. Gao *et al.* [21] applied a deep belief network to crude-type classification. Shang *et al.* [22] proposed a novel soft sensor modeling method based on a deep learning network. Ince *et al.* [23] proposed an early fault-detection system using one-dimensional (1-D) CNN, which has an inherent adaptive design to fuse the feature extraction and classification phases of the motor fault detection into a single learning body.

For normal image classification, due to the large gap between the images of different categories, it is always enough to take only one single-view image as the input of CNN. However, fine-grained recognition tasks, such as identifying the species of a bird or the model of an aircraft, are much more challenging. A common approach for fine-grained recognition tasks is

[1] [Online]. Available: http://modelnet.cs.princeton.edu/.

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

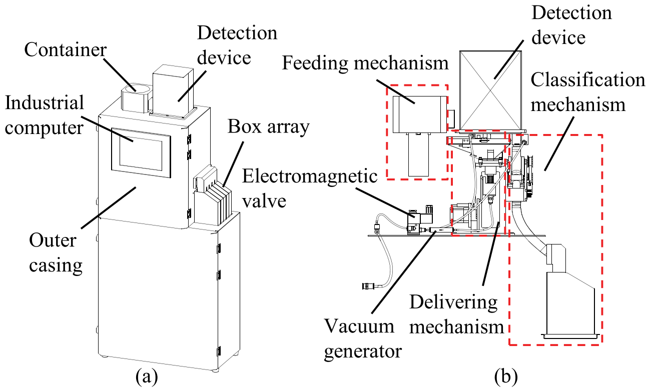

Fig. 1. Overall structure of the pearl classification machine: (a) the external shape and (b) the internal mechanical structure.

using part-based models such as bilinear CNN [24] models and two-stream contextualized CNN [25]. Besides, incorporating multiple-instance learning into a deep learning framework [26] also performs well.

These studies inspire the current work, and we propose a novel CNN architecture with five streams and take multiview images of a pearl as the input. Our work is relatively similar to a recent study performed by Laptev *et al.* [27], but there are still significant differences between the two. First, they used multiple streams in the training stage and used only one stream in the test stage, since the parameters are all shared among the streams, which is similar to Siamese network [28]. In our work, however, the parameters among streams are only partially shared. Second, they generated multiple instances of the image according to transformations and passed these instances through initial layers, so that the network can learn the transformation invariant features. However, our work takes multiview images as the input of the network, in order to overcome the problem that the visual differences among different categories are too small to be overlooked due to the inappropriate viewpoint.

III. Mechanical Design

We design a pearl classification machine to collect pearl images of different viewing angles and realize pearl classification automatically. In this section, we will introduce the mechanical design in detail.

A. Overall Structure

The external shape and the internal mechanical structure of the pearl classification machine are shown in Fig. 1(a) and (b), respectively. Its mechanical structure mainly consists of feeding mechanism, delivering mechanism, vision-based detection device, and classification mechanism, as well as the framework and the outer casing. The framework is used for installing the mechanical components and electronic equipment, and the outer casing encloses the framework and is used for protecting the internal structure. The role of the feeding mechanism is to store the pearls to be detected and send them into an inverted pyramid-shaped container. The role of the delivering mechanism is to deliver the pearls into the detection device one by one from the inverted pyramid-shaped container. The role of the classification mechanism is to deliver the pearls into the corresponding boxes according to the classification results obtained by our MS-CNN model.

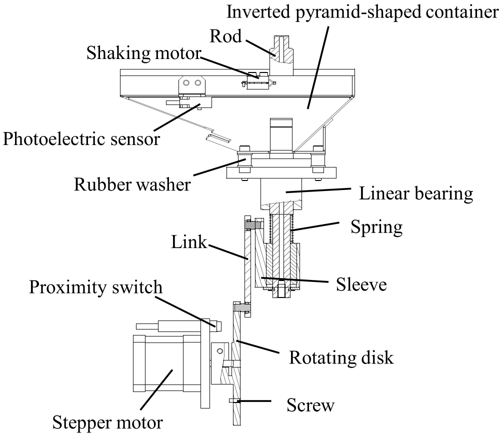

Fig. 2. Structure of the delivering mechanism.

B. Details of the Main Mechanism

1) Feeding Mechanism: The feeding mechanism consists of a container and a dc motor. The bottom of the container is a rotatable disc, which is driven by a dc motor. When the dc motor is rotating, the pearls in the container will fall into the inverted pyramid-shaped container through the round hole, which is on the lower side of the container, due to the centrifugal force and gravity.

2) Delivering Mechanism: The delivering mechanism consists of an inverted pyramid-shaped container, a mechanism similar to a crank-slider mechanism, and a stepper motor, as shown in Fig. 2. The inverted pyramid-shaped container is fixed on the framework with rubber washers, which are used for preventing vibration from being transmitted to the framework. One side of the container is a movable board. When inserting the board, pearls can be stored in the container. When pulling the board out, pearls can flow out of the container through the gap. Two shaking motors and a photoelectric sensor are installed on the two sides of the inverted pyramid-shaped container. The role of shaking motors is to distribute pearls evenly in the container. And the photoelectric sensor is used for detecting the level of the pearls in the container to judge whether the amount of pearls meets the requirement or not.

The mechanism similar to the crank-slider mechanism consists of a linear bearing fixed on the framework, a rod in the linear bearing (equivalent to the slider in the crank-slider mechanism), a rotating disk (equivalent to the crank in the crank-slider mechanism), and a link, which connects the rotating disk and the rod. The rod is hollow and can move up and down between the bottom of the inverted pyramid-shaped container and the bottom of the detection device, whose function is to deliver the pearls into the detection device one by one from the container. The top of the rod is concave and the bottom of the rod connects with a pipe. A pearl can be held on the top of the rod when the pipe is under negative pressure. The rod and the link are connected

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

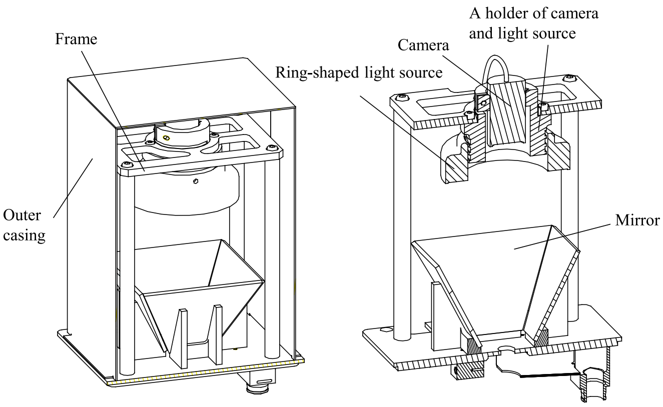

Fig. 3. Structure of the vision-based detection device.

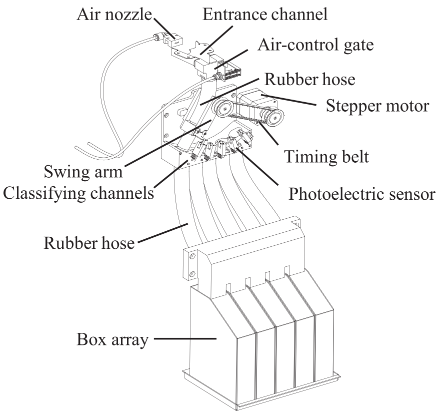

Fig. 4. Structure of the classification mechanism.

indirectly with a sleeve and a spring. This structure ensures that the pearl to be detected can get to the expected position in the detection device without precise size match and stay there for a short time. The rotating disk is driven by a stepper motor, so that the movement of the rod can be controlled by the stepper motor. And there is a screw on the specific point of the rotating disk, which cooperates with a proximity switch to eliminate the error caused by step loss in the operation of the stepper motor.

3) Vision-Based Detection Device: The structure of the vision-based detection device is shown in Fig. 3. The core components of the detection device include an HD camera, a ring-shaped light source, and four mirrors. The four mirrors are combined into an inverted pyramid shape, which allows the camera to capture the images of the pearl to be detected from five different viewing angles by one shot. The camera and ring-shaped light source are fixed on the framework, just above the four mirrors, by a holder, which can be adjusted up and down. The surfaces of all the parts in the outer casing are coated with light absorption material. And there is a hole on the bottom of the detection device, which allows a pearl to get in and out.

4) Classification Mechanism: The classification mechanism consists of an entrance channel, an air-control gate, a swing arm mechanism, and a box array, as shown in Fig. 4. The entrance channel is installed under the detection device. And there is an air nozzle fixed opposite to the entrance and the air-control gate is connected to the other side of the channel. The pearl at the entrance can be blown into the air-control gate through the entrance channel by the air nozzle. The exit of the air-control gate is connected to a rubber hose, which is able to swing driven by the swing arm. And the swing arm is driven by a stepper motor with a timing belt. Therefore, the exit of the rubber hose is able to be connected to any entrance of the classification channels controlled by the stepper motor. There is a photoelectric sensor that is used for obtaining the position information of the swing arm to ensure it be connected to the expected classification channel. And the classification channels are connected to the array of boxes, which are used for storing the classified pearls with rubber hoses.

C. Working Principle of the Whole Machine

We need to pour the pearls to be detected into the container before the pearl classification machine starts working. With the disc at the bottom of the container rotating, the pearls in the container fall into the inverted pyramid-shaped container. When the amount of pearls in the inverted pyramid-shaped container meets the requirement judged according to the test result of the photoelectric sensor, the disc stops rotating so that the pearls do not fall anymore. When the pearls in the inverted pyramid-shaped container become fewer, the disc rotates again, and the pearls in the container are added.

While the delivering mechanism is working, the shaking motor installed on the side of the inverted pyramid-shaped container is always in operation, which distributes pearls evenly in the container. When the amount of pearls in the container meets the requirements, the stepper motor of delivering mechanical starts working. At the beginning, the top of the rod is at the bottom of the inverted pyramid-shaped container, and a pearl is on the top of the rod.

While the vacuum generator is working, the pipe is under negative pressure so that the pearl can be held on the top of the rod. Then, the rod rises up and delivers the pearl into the detection device with the rotating of the stepper motor. When the pearl arrives at the specific position, the rod is blocked by the bottom of the detection device and does not rise anymore, but the sleeve continues to rise for a period of time and the spring is compressed. In this process, the pearl is stationary in the detection device, which provides the detection device enough time to detect the pearl. After the sleeve gets to the highest point, it goes down and the spring is released as the motor rotates. When the rod goes down, the vacuum generator stops working and the air nozzle starts working. When the top of the rod goes down to the entrance of the classification mechanism, the pearl will be blown into the air control gate through the entrance channel by the air nozzle. Then, the rod goes down to the bottom of the inverted pyramid-shaped container and a delivering operation is finished.

After the detection device completes detecting the pearl, the classification result is sent to the controller. And the controller controls the rotation of the stepper motor in the classification mechanism according to the result. When the swing arm is driven to the corresponding entrance, the light of the

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

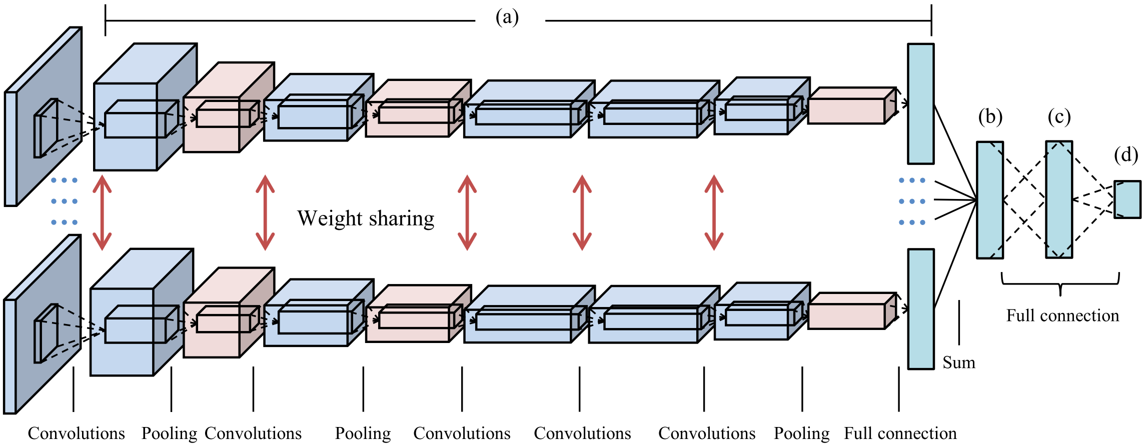

Fig. 5. Architecture of MS-CNN. Five input images correspond to the five views of a pearl. (a) For each input image, we use an AlexNet similar network to process it, consisting of five convolutional layers, three pooling layers, and one fully connected layer. (b) After that, the element-wise layer is applied on the feature vectors of the five streams to obtain a unified vector of feature, which contains information of multiple views of the pearl. (c) The unified vector of feature then serves as the input of a fully connected layer, and further propagates to the last fully connected layer. (d) The output of the last layer is fed to softmax, which produces a distribution over the classification labels. Moreover, the weights of the corresponding convolutional layers are shared among the five streams through a weight-sharing mechanism.

photoelectric sensors is blocked and the sensor generates a feedback signal. When the signal is received, the controller controls the air-control gate to open and the pearl falls into the corresponding box through the rubber hose, thus realizing the classification of the pearl.

IV. Method Description

Recently, CNN has attracted a lot of attention from both academia and industry, and has been used extensively in machine vision applications. CNN is effective in extracting and representing high-level features from big complex data. Due to its great success in many applications, and also due to the fact that we can collect numerous pearl images using our machine, we adopt CNN here to establish the classification model for pearls.

A. CNN Notation

CNN consists of alternatively stacked convolutional layers and pooling layers, and the feature maps are vectorized and fed into fully connected layers followed by a softmax layer. Let x be the input image, and $f^l(x)$ be the output of the lth layer. We refer to the computation of the lth fully connected layer as operator $F^l(i, w^l)$, where w^l represents the weight of the lth layer, and i is the input of the layer. If there is a pooling layer following a convolutional layer, we consider them as a single layer.

B. MS-CNN Architecture

Pearls, as a kind of gems, the classification of which is even more difficult than many fine-grained classifications, such as dogs and cats. In fact, dog lovers may tell the breed of a dog when they see a picture of it with whatever viewing angle. However, for a pearl, even the experienced professionals need to look very carefully around it to determine its classification. This is because some features of pearl, such as texture, are quite tiny and local, one cannot detect them from only a single viewing angle. Therefore, it requires us to use multiview images of pearls, and further construct an MS-CNN model to process them simultaneously, as shown in Fig. 5. The five streams can be considered as five feature extractors, which take the pearl images of five viewing angles, including top, left, right, main, and rear, as inputs.

The dimension of each input layer is $300 \times 300 \times 3$, i.e., we consider RGB images. And each stream, similar to the AlexNet, includes five convolutional layers, three pooling layers, and one fully connected layer.

The output of each stream is the feature of an image from one viewing angle. According to our notations, it can be expressed as follows:

$$f^6(x) = F^6(f^5(x), w^6). \tag{1}$$

Given the output of each stream $f^6(x)$, we, thus, construct the new feature $g^6(x)$ that contains information of multiple views of a pearl. We use an element-wise layer and two fully connected layers after the five streams, as shown in Fig. 5, and the element-wise layer formulates these features in the following manner:

$$g^6(x) = \sum_{k \in \Phi} \langle f^6(x) \rangle_k \tag{2}$$

where k is the index of streams, and Φ is the index set $\{1, \ldots, 5\}$.

Besides, we can integrate these features through a weighted combination

$$g^6(x) = \sum_{k \in \Phi} w_k \langle f^6(x) \rangle_k \tag{3}$$

where w_k is the weight to control the tradeoff among features of five streams, $w_k \in [0, 1]$.

Softmax loss function is finally used to tune the weights of the network with backpropagation in the training stage. The detailed architecture of our MS-CNN is shown in Table I. Although this architecture may not be optimal and could be further improved, we find that it can do an impressive job in our classification

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

TABLE I
SIZE FOR EACH LAYER OF THE MS-CNN

Stream	Name	Type	Filter size /stride	Output size
1–5	Conv1	Convolution	11 × 11/4	73 × 73 × 96
	Pool1	Max pooling	3 × 3/2	36 × 36 × 96
1–5	Conv2	Convolution	5 × 5/1	36 × 36 × 256
	Pool2	Max pooling	3 × 3/2	18 × 18 × 256
1–5	Conv3	Convolution	3 × 3/1	18 × 18 × 384
1–5	Conv4	Convolution	3 × 3/1	18 × 18 × 384
1–5	Conv5	Convolution	3 × 3/1	18 × 18 × 256
	Pool5	Max pooling	3 × 3/2	9 × 9 × 256
1–5	Fc6	Full connection		4096
1–5	Dropout [29]	Dropout (50%)		4096
	Element-wise	Sum		4096
	Fc7	Full connection		4096
	Dropout	Dropout (50%)		4096
	Fc8	Full connection		2/7

tasks and slight changes seem not to have a significant impact on the classification results.

ReLU neuron [30] is used for all convolutional layers and all fully connected layers except the last fully connected layer. It is worth mentioning that the weights of convolutional layers are shared among the five streams. Therefore, the actual model requires smaller memory than the model without weight-sharing. Considering that the five streams correspond to the five different views of the pearl, to reflect the difference between them, the weights of the fully connected layers in the five streams are not shared.

C. Implementation

We train our MS-CNN using stochastic gradient descent with a batch size of 20 and use some tricks during training, such as momentum and weight decay.

We initialize the weights of convolutional layers following a zero-mean Gaussian distribution with standard deviation 0.01. The weights of fully connected layers are initialized in the same way but with standard deviation 0.005. And we initialize the neuron biases in both convolutional and fully connected layers with constant 0.1.

V. EXPERIMENT

Using our machine, we collect the image dataset of pearls labeled by experienced professionals in advance. All the pearls came from Zhuji, Zhejiang Province, China. Since the target pearls mainly have light color, we set black background so that the multiview images of a pearl can be obtained easily by threshold segmentation [31].

A. Pearl Dataset

We collected a large number of pearls, which were labeled by the experienced professionals in a moderate-scale pearl-producing corporation. Basically, the pearls are classified into two or seven classes, determined by the further usage of pearls and the market. Specifically, the pearls can be classified into two classes with rough rules, one for flat-shape or deeply blemished pearls and the other for slightly or nonblemished pearls. On the other hand, these pearls can also be classified into seven classes with fine rules. Compared with the rough rules, the fine rules for pearl classification have more strict requirements in the shape and texture of pearls. In particular, the flat-shape or deeply blemished pearls can be further classified into three classes according to fine rules: 1) the pearls with multiple flat faces; 2) the pearls with symmetric shape; and 3) the rest of the pearls. On the other hand, the slightly or nonblemished pearls can be further classified into four classes: 1) the pearls with the ratio of short to long radius approximately higher than 0.7; 2) the rest of the shallow blemished pearls; 3) the rest of the implicit blemished pearls; and 4) the rest of the pearls.

TABLE II
SUMMARY OF HAND-CRAFTED FEATURES

Index	Features	View
1–8	$F(0)$–$F(7)$	Top
9–16	$F(0)$–$F(7)$	Left
17–24	$F(0)$–$F(7)$	Right
25–32	$F(0)$–$F(7)$	Main
33–40	$F(0)$–$F(7)$	Rear
41	Average contrast	
42	Average correlation	
43	Average energy	
44	Average homogeneity	

We finally collect 10 500 pearls and 52 500 multiview images with label information. We split the dataset into three parts, according to the ratio of $6:2:2$, including 6300 training instances, 2100 validation instances, and 2100 test instances. The labels of these instances depend on the classification rules. We use training and validation sets in the training stage and use a test set to evaluate the models.

B. Classification Results

To test the performance of the proposed CNN, we train two models for rough and fine classification tasks. Specifically, we note *task #1* for classifying pearls into two classes and *task #2* for classifying pearls into seven classes. The element-wise layer combines features by (2) in this part. To compare with the traditional methods, we also design hand-crafted features to classify the pearls. For shape features, we use 1-D FD mentioned in [10] and [13]. Specifically, eight Fourier coefficient values $F(0)$–$F(7)$ are computed as the shape features. After getting the GLCM of pearl images, we use the metrics, including contrast, correlation, energy, and homogeneity, as the texture features.

Corresponding to the images of five viewing angles for a pearl, we have five groups of shape and texture features. We obtain the final texture features by averaging over five groups of texture features. Considering that the average operation may not be appropriate for the integration of shape information, five groups of shape features are concatenated as the final shape features of the pearl. Finally, we get a 44-dimensional feature vector, as shown in Table II. Based on these features, we then use SVM and BPNN to classify the pearls. Radial basis function

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

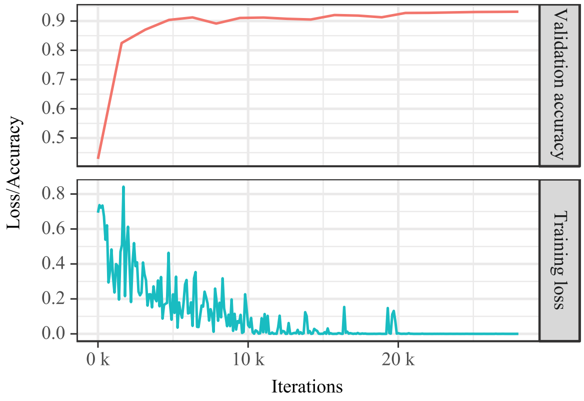

Fig. 6. Training loss and validation accuracy for task #1, which aims to classify pearls into two classes.

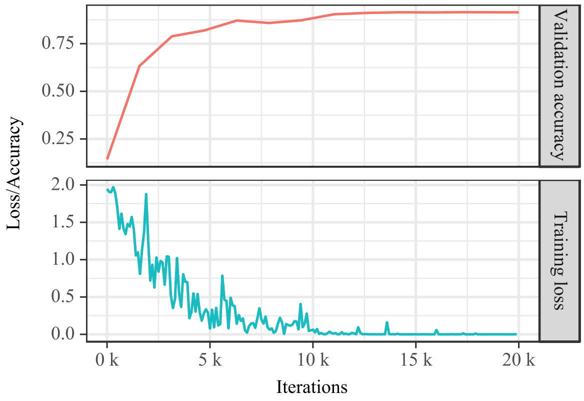

Fig. 7. Training loss and validation accuracy for task #2, which aims to classify pearls into seven classes.

(RBF) kernel is used in the SVM model. Training and validation data are used to select the hyperparameters of the SVM model by a grid searching technique. More specifically, fourfold cross-validation is used to evaluate the hyperparameters and determine the optimum ones. We get the optimal RBF kernel parameter $g = 8$ and the optimal penalty factor $c = 2$ for both task #1 and task #2. The BPNN model is developed using a single hidden layer with the sigmoidal activation function. The number of hidden neurons is also determined by the same method described above. For task #1, the optimal number of hidden neurons is 36, while for task #2, the optimal number is 42. These are the models for comparison.

We train our MS-CNN on GeForce GTX TITAN X with Caffe [32]. The learning rate starts from $1e{-}3$ and is divided by 2 after every 10 000 iterations. We stop training when the validation accuracy stops increasing. The models for two different tasks tend to converge after 10 000 iterations: the validation accuracy for task #1 stops increasing after 28 000 iterations, and reaches about 93.19%, while the validation accuracy stops increasing after 20 000 iterations for task #2 and keeps about 91.44%. Figs. 6 and 7 show the trends of training loss and the trends of validation accuracy in the training stage, for task #1 and task #2, respectively. It takes about 72 s to test 2100 pearls using GPU. In other words, classifying a pearl only needs 0.034 s, leaving plenty of time for mechanical operations.

TABLE III
TEST ACCURACIES OBTAINED BY SVM, BPNN, AND OUR MS-CNN ON TASK #1 AND TASK #2

Task	Method	Features	Accuracy (%)
#1	SVM	Hand-crafted	85.19
	BPNN		81.57
	MS-CNN	Autoencoder-derived	**92.14**
#2	SVM	Hand-crafted	67.19
	BPNN		62.52
	MS-CNN	Autoencoder-derived	**91.24**

The accuracies obtained by SVM, BPNN, and our MS-CNN on the test data are presented in Table III. It can be seen that, in general, MS-CNN behaves much better than SVM and BPNN. For task #1, which is relatively easy, SVM and BPNN can achieve acceptable accuracies, i.e., higher than 80%, although they are still lower than the accuracy, equal to 92.14%, obtained by our method. However, for task #2, which is relatively difficult, SVM and BPNN, taking the 44-dimensional hand-crafted feature vector as input, rapidly lose their effectiveness, i.e., the accuracies are even lower than 70%, while in this case, the accuracy obtained by the MS-CNN still keeps high level and is equal to 91.24%. Such significant contrast may indicate that richer hand-crafted features need to be extracted to solve more complicated classification tasks based on SVM and BPNN, while CNN, on the other hand, can obtain expressive features automatically no matter for simple or complicated classification tasks, which is a huge advantage especially when the various hand-crafted features are difficult to describe and expensive to obtain.

Five streams are used in our MS-CNN model to fuse the information of different views of a pearl. In order to verify the effectiveness of using multiple streams, we also use a single-view image to realize pearl classification. In this case, only one stream is needed and the network architecture is just like the AlexNet. The size of the network is set to the same as one stream of the MS-CNN, as presented in Table I. We train the network by using the images of top, left, right, main, and rear views. After about 30 000 iterations in the training stage, the validation accuracy stops increasing. Then, we compare the test accuracy obtained by using single-stream CNN with that obtained by our MS-CNN. The comparison results are shown in Fig. 8, where we can see that our method indeed outperforms the single-stream CNN model, especially for the relatively difficult task #2.

C. Weighted Combination

We can find that a single-stream CNN model is more effective on both tasks #1 and #2 when it takes left or right view as input, as shown in Fig. 8. It seems that left and right views can provide more information and it is reasonable to assign a higher weight to them. Hence, we empirically set the weights w_k corresponding to left and right views to 1, and set the others to 0.8 in (3).

The accuracies obtained by our modified model are presented in Table IV. The results show that weighted combination of different streams can indeed improve the performance of our

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

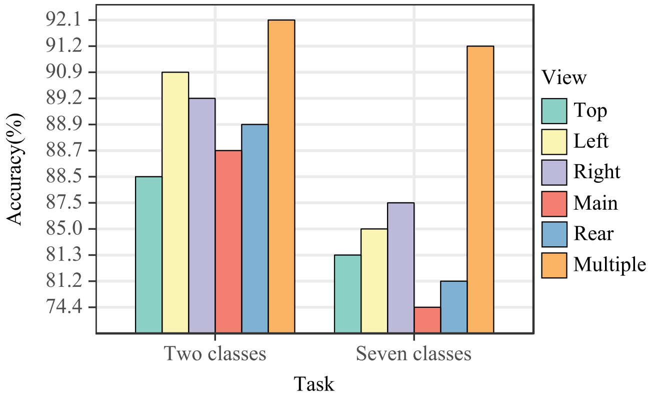

Fig. 8. Comparison between MS-CNN and the single-stream CNN taking the top, left, right, main, and rear views as input.

TABLE IV
COMPARISON BETWEEN MS-CNN AND MS-CNN WITH WEIGHED COMBINATION ON TASK #1 AND TASK #2

Task	Method	Accuracy (%)
#1	MS-CNN	92.14
	Weighted combination	**93.90**
#2	MS-CNN	91.24
	Weighted combination	**92.57**

TABLE V
COMPARISON AMONG DIFFERENT NETWORKS ON TASK #2

Method	Batch size	Training time (h)	Testing time (s)	Accuracy (%)
AlexNet	20	5.64	**15.25**	87.48
VGG19	5	18.30	74.35	**89.48**
GoogleNet	20	**2.87**	23.69	89.29
ResNet34	20	3.78	30.69	83.57

MS-CNN. Specifically, setting some weight to be 1 and the others to be 0 can make our MS-CNN degenerate to a single-stream model if the weights of streams are not shared. Although weighted combination can make the model more flexible, how to find optimal weight parameters still keeps an open problem.

D. Why AlexNet

It is well known that many successful networks were proposed recently. Besides AlexNet, we also use other networks to realize pearl classification. In particular, we compare AlexNet with VGG19, GoogleNet, and ResNet34 on task #2, since it is relatively difficult.

Right view is adopted in our experiment, as it seems that this view is more effective in classification task, as shown in Fig. 8. We use some indices, including training time, testing time, and accuracy, to evaluate different networks. It is found that VGG19 is difficult to converge in the training stage, so we add batch normalization [33] after convolutional layers in VGG19. Since training VGG19 with the batch size of 20 is memory-consuming, we decrease the batch size to 5 so that it can finish training in a single GPU. The experimental results are presented in Table V.

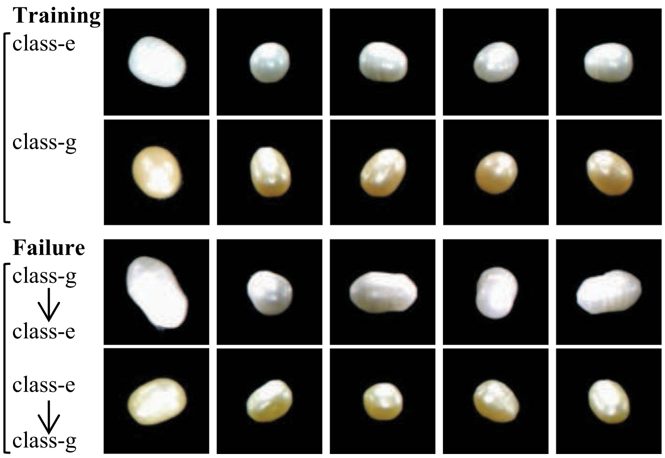

Fig. 9. First two rows are the examples of training pearls, belonging to class-e and class-g, with the classification criteria described in Section V-A. The last two rows are the failure cases. The pearl on the third row is classified as class-g by experienced professionals, but is classified as class-e by our model, while the pearl on the last row is classified as class-e by experienced professionals, but is classified as class-g by our model. The images are listed in the order of top, left, right, main, and rear views, from the left to the right.

We can see that training VGG19 is most time-consuming due to the smaller batch size and larger number of parameters. Compared with other networks, AlexNet has fewer layers, so that it has the least testing time, i.e., it takes only 15.25 s to test 2100 pearls with AlexNet, much faster than the other three networks. The accuracy of AlexNet is 87.48%, comparable with VGG19 and GoogleNet and much better than ResNet34. The simpler structure of AlexNet makes it easier to be extended to multistream networks. Moreover, the less testing time can help the classification machine process more pearls in unit time, which is important in industrial application. For these reasons, we adopt AlexNet to design MS-CNN.

E. Failure Cases

In order to better understand the performance of our model, we select some failure cases in both MS-CNN and MS-CNN with weighted combination, i.e., neither MS-CNN nor MS-CNN with weighted combination can classify these pearls correctly. The failure cases are shown in Fig. 9. The pearls of class-e and class-g are those with the ratio of short to long radius approximately lower than 0.7, in simple terms, it can be interpreted that these pearls are rice-shaped. Furthermore, the pearls of class-e are shallow blemished, while pearls of class-g are nonblemished.

We can find that the pearl on the third row is rice-shaped, and it is shallow blemished with high probability according to left, right, main, and rear views. From the viewpoint of classification criteria, the pearl on the third row could also belong to class-e. On the other hand, we can also see that the size of this pearl is relatively large, and thus may have a higher value than the other pearls of class-e. This might be the reason why the experienced professionals tend to classify this pearl into class-g as the more valuable class.

The pearl on the last row is also rice-shaped. By zooming in left, right, and rear views, we can find that there is a slightly

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

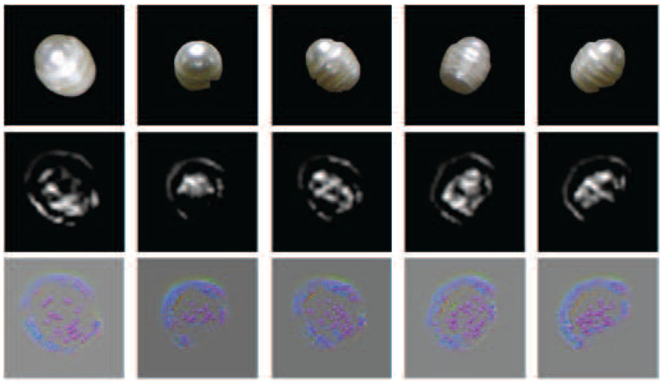

Fig. 10. Visualizations of the activations of conv2_{239} in all five streams using deconvnet for a pearl. The first row shows five views of the pearl, which are listed in the order of top, left, right, main, and rear views. The activations in the second row show that the channel does not only extract edges but highly respond to the textures on the surface. And the visualizations in the third row reflect part of the texture information of the pearl.

dent on the surface. Hence, this pearl is shallow blemished and it should belong to class-e. We argue that dent is an unusual blemish in our training data, so our models failed to classify it correctly, especially this pearl is very similar to pearls of class-g on color and luster. We believe this wrongly classified case can be corrected after our dataset is enlarged in the future.

VI. Feature Visualization

To explore what features extracted by our model so as to better understand the mechanism of the MS-CNN, we use the *deconvnet* [34]. The deconvnet is a novel way to map the activities of a chosen feature map back to the input pixel space, making it possible for us to build intuitions about how CNN works. Since each stream in our MS-CNN is similar to the AlexNet, the convolutional layers in the MS-CNN share similar functions as those in AlexNet, i.e., conv1 simply extracts features like edges and colors, while conv2 reflects their conjunctions. For example, one pearl texture detector, conv2_{239} (channel number 239 on conv2), as shown in Fig. 10, obviously captures the complex blemishes on a pearl.

Although part of the features may reflect the glisten on the surface of the pearl, the visualization results reveal that our model does not only simply extract features like shape and color, but also can recognize relatively complex features like texture. It coincides with the rules of manual pearl classification in reality.

VII. Conclusion

In this paper, we designed a pearl classification machine, based on which multiview images of massive pearls can be automatically collected, and the pearls can be classified with a relatively high accuracy utilizing a novel MS-CNN algorithm.

The machine can run smoothly. Our MS-CNN with five streams, taking the images of five views for each pearl as the input, can extract the features on the surface of the pearl and further fuse them through a weight-sharing mechanism and the following element-wise and fully connected layers. Experimental results showed that the proposed MS-CNN can overcome the problem caused by viewpoint to a certain extent and performed well in two classification tasks, much better than SVM and BPNN based on hand-crafted features, especially when a weighted combination strategy was adopted to pay more attention to the left and right views. It should be noted that here we chose AlexNet as a basis network to construct our MS-CNN, since the experiments validated that AlexNet has the least testing time and comparable accuracy, compared with VGG19, GoogleNet, and ResNet34. Through visualization of activations of convolutional kernels, we demonstrated that our model can recognize relatively complex features like texture. MS-CNN may also be used in many other applications where multiview images need to be handled, e.g., 3-D CAD models. Our machine can classify pearls effectively and efficiently, imitating the manual classification in reality, and thus is of value to the pearl-producing companies.

References

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[2] Y. L. Murphey, M. A. Masrur, Z. Chen, and B. Zhang, "Model-based fault diagnosis in electric drives using machine learning," *IEEE/ASME Trans. Mechatron.*, vol. 11, no. 3, pp. 290–303, Jun. 2006.

[3] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[4] R. Hecht-Nielsen *et al.*, "Theory of the backpropagation neural network," *Neural Netw.*, vol. 1, no. Suppl. 1, pp. 445–448, 1988.

[5] L. Deng *et al.*, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 7, nos. 3–4, pp. 197–387, 2014.

[6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[7] X. Zhang, H. Xiong, W. Zhou, W. Lin, and Q. Tian, "Picking deep filter responses for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2016, pp. 1134–1142.

[8] H. Zhang *et al.*, "SPDA-CNN: Unifying semantic part detection and abstraction for fine-grained recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2016, pp. 1143–1152.

[9] S. Huang, Z. Xu, D. Tao, and Y. Zhang, "Part-stacked CNN for fine-grained visual categorization," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2016, pp. 1173–1182.

[10] D. Zhang and G. Lu, "Shape-based image retrieval using generic fourier descriptor," *Signal Process., Image Commun.*, vol. 17, no. 10, pp. 825–848, 2002.

[11] M. Liu, Y. He, and B. Ye, "Image zernike moments shape feature evaluation based on image reconstruction," *Geo-Spatial Inf. Sci.*, vol. 10, no. 3, pp. 191–195, 2007.

[12] R. M. Haralick *et al.*, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[13] G. Li, B. Li, Y. Wang, Z. Yu, J. Li, and H. Zhao, "Pearl shape recognition based on computer vision," *Trans. Chin. Soc. Agric. Mach.*, vol. 39, no. 7, pp. 129–132, 2008.

[14] G. Li, B. Li, Y. Wang, and J. Li, "Classification method of pearl luster degree based on HSL," *Trans. Chin. Soc. Agric. Mach.*, vol. 39, no. 6, pp. 113–117, 2008.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Comput. Vision Pattern Recog.*, 2009, pp. 248–255.

[16] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[19] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2015, pp. 1–9.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2016, pp. 770–778.

[21] X. Gao, C. Shang, Y. Jiang, D. Huang, and T. Chen, "Refinery scheduling with varying crude: A deep belief network classification and multimodel approach," *AIChE J.*, vol. 60, no. 7, pp. 2525–2532, 2014.

[22] C. Shang, F. Yang, D. Huang, and W. Lyu, "Data-driven soft sensor development based on deep learning technique," *J. Process Control*, vol. 24, no. 3, pp. 223–233, 2014.

[23] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-d convolutional neural networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7067–7075, Nov. 2016.

[24] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear CNN models for fine-grained visual recognition," in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 1449–1457.

[25] J. Liu, C. Gao, D. Meng, and W. Zuo, "Two-stream contextualized CNN for fine-grained image classification," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 4232–4233.

[26] J. Wu, Y. Yu, C. Huang, and K. Yu, "Deep multiple instance learning for image classification and auto-annotation," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2015, pp. 3460–3469.

[27] D. Laptev, N. Savinov, J. M. Buhmann, and M. Pollefeys, "Ti-pooling: Transformation-invariant pooling for feature learning in convolutional neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, 2016, pp. 289–297.

[28] J. Bromley *et al.*, "Signature verification using a "siamese" time delay neural network," *Int. J. Pattern Recog. Artif. Intell.*, vol. 7, no. 4, pp. 669–688, 1993.

[29] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR*, vol. abs/1207.0580, 2012. [Online]. Available: http://arxiv.org/abs/1207.0580

[30] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[31] S. S. Al-Amri, N. V. Kalyankar, and S. D. Khamitkar, "Image segmentation by using threshold techniques," *J. Computing*, vol. 2, no. 5, pp. 83–86, 2010.

[32] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[33] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[34] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 818–833.

Qi Xuan received the B.S. and Ph.D. degrees in control theory and engineering from Zhejiang University, Hangzhou, China, in 2003 and 2008, respectively.

He was a Postdoctoral Researcher with the Department of Information Science and Electronic Engineering, Zhejiang University, from 2008 to 2010, and a Research Assistant with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, China, SAR, in 2010 and 2017. From 2012 to 2014, he was a Postdoctoral Fellow with the Department of Computer Science, University of California at Davis, Davis, CA, USA. He is currently a Professor with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His current research interests include network-based algorithm design, social network data mining, social synchronization and consensus, reaction-diffusion network dynamics, machine learning, and computer vision.

Binwei Fang was born in Zhejiang, China, in 1993. He received the B.S. degree in automation from China Jiliang University, Hangzhou, China, in 2015. He is currently working toward the M.S. degree in control theory and engineering at the Zhejiang University of Technology, Hangzhou, China.

His current research interests include computer vision, machine learning, and deep learning.

Yi Liu (M'11) received the Ph.D. degree in control theory and engineering from Zhejiang University, Hangzhou, China, in 2009.

He was a Postdoctoral Researcher with the Department of Chemical Engineering, Chung-Yuan Christian University from February 2012 to June 2013. He is currently an Associate Professor with the Zhejiang University of Technology, Hangzhou, China. He has published about 30 international journal papers. His research interests include data intelligence with applications to modeling, control, and optimization of industrial processes.

Jinbao Wang was born in Zhejiang, China, in 1991. He received the B.S. degree in automation from the Zhejiang University of Technology, Hangzhou, China, in 2014, where he is currently working toward the M.S. degree in control science and engineering.

He is currently focusing on information dissemination modeling on social network. His research interests include computer vision and data mining.

Jian Zhang received the B.S. degree in automation from the Zhejiang University of Technology, Hangzhou, China, in 2017, where he is currently working toward the M.S. degree at the College of Information and Engineering.

His current research interests include computer vision, understanding of deep convolutional neural networks, and data mining.

Yayu Zheng was born in Zhejiang, China, in 1978. He received the B.S. and Ph.D. degrees in control theory and engineering from Zhejiang University, Hangzhou, China, in 2002 and 2008, respectively.

He is currently an Associate Professor with the Zhejiang University of Technology, Hangzhou, China. His major research interests embedded systems for artificial intelligence and video processing.

Guanjun Bao (M'18) received the B.S. degree in mechanical engineering from North China Electric Power University, Beijing, China, in 2001, and the Ph.D. degree in mechatronics from the Zhejiang University of Technology, Hangzhou, China, in 2006.

In 2006, he joined the College of Mechanical Engineering, Zhejiang University of Technology, as an Assistant Professor where he became an Associate Professor in 2009. He has authored or co-authored about 70 papers. His research interests include robotics, soft robotics, machine vision, and smart manufacturing.

Authorized licensed use limited to: Tsinghua University. Downloaded on November 14,2022 at 09:03:22 UTC from IEEE Xplore. Restrictions apply.