IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 25, 2019, accepted September 10, 2019, date of publication September 13, 2019, date of current version October 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2941239

SAS: Painting Detection and Recognition via Smart Art System With Mobile Devices

ZHENYU WANG[1], JIE LIAN[1], CHUNFENG SONG[2], ZHAOXIANG ZHANG[2], (Senior Member, IEEE), WEI ZHENG[1], SHAOLONG YUE[1], AND SENRONG JI[1]

[1]School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China
[2]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China

Corresponding author: Zhenyu Wang (zywang@ncepu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61976090 and Grant 61573139, and in part by the Fundamental Research Funds for the Central Universities under Grant 2018ZD05.

ABSTRACT Artwork recognition is an important research direction in the field of image processing. However, most of the current proposed methods are not designed for the demand of real-time analysis with mobile devices. Moreover, existing methods usually rely on high quality images and require large amounts of computing consumption. Based on the deep learning technology, in this paper, we propose a Smart Art System (SAS) with mobile devices. Our SAS mainly consists of two parts, i.e., painting detection unit and recognition unit. The detection module adopts a new painting detection algorithm called Single Shot Detection with Painting Landmark Location (SSD-PLL). SSD-PLL can effectively eliminate the influence of complex background factors on recognition. Considering the limited computing capacity of the mobile devices, our recognition module adopts a new ultra-light painting classifier. The classifier adopts MobileNet as the backbone and owns extra operation for Local Features Fusion (LFF). With our SAS, users can use mobile phone to take a photo of any paintings, then SAS would analyze the paintings and report the relevant information in real time. In order to validate the effectiveness of the proposed method, we have established two large scale image databases. The databases include 7,500 Traditional Chinese paintings (TCPs) and 8,800 Oil paintings (OPs), respectively. We evaluate our method and compare with the relevant algorithms, and our method achieves the highest performance and better real-time performance. Extensive experimental results on these databases show the effectiveness of the proposed algorithm.

INDEX TERMS Mobile devices, deep learning, painting detection and recognition.

I. INTRODUCTION

With the improvement of people's living standards, there is a growing number of painting's exhibitions and auctions. In such conditions, people always want to perceive the information about the paintings in real time. Our proposed Smart Art System (SAS) (Figure 1) makes it possible to meet the needs for users. As a convenient mobile device with camera and powerful computing functions, mobile phone is well suited as the platform for our system. SAS allows users to shoot any paintings they want to recognize. After shooting, our system would analyze the paintings automatically and return the relevant information to users in real time. In particular, detecting and recognizing the paintings in the captured images is the key of our system.

The associate editor coordinating the review of this manuscript and approving it for publication was Huiyu Zhou.

In the domain of artificial intelligence, there are few works about painting recognition. Existing works [1], [4]–[7], [9], [10] mainly focus on the attributes classification of the paintings, which is not designed for the single image recognition. On the other hand, some work [2] still use traditional methods [3], [8] for painting analysis without applying deep learning technology. More importantly, the practical application value of above works is limited, while our system is oriented to practical application.

Our system is based on mobile computing platform, when selecting model, we need to give consideration to both accuracy and latency.

In this paper, we propose a new painting detection algorithm called Single Shot Detection with Painting Landmark Location (SSD-PLL). The real application scenarios is complex and different users also have their own shooting habits. It is difficult to recognize painting by using captured image

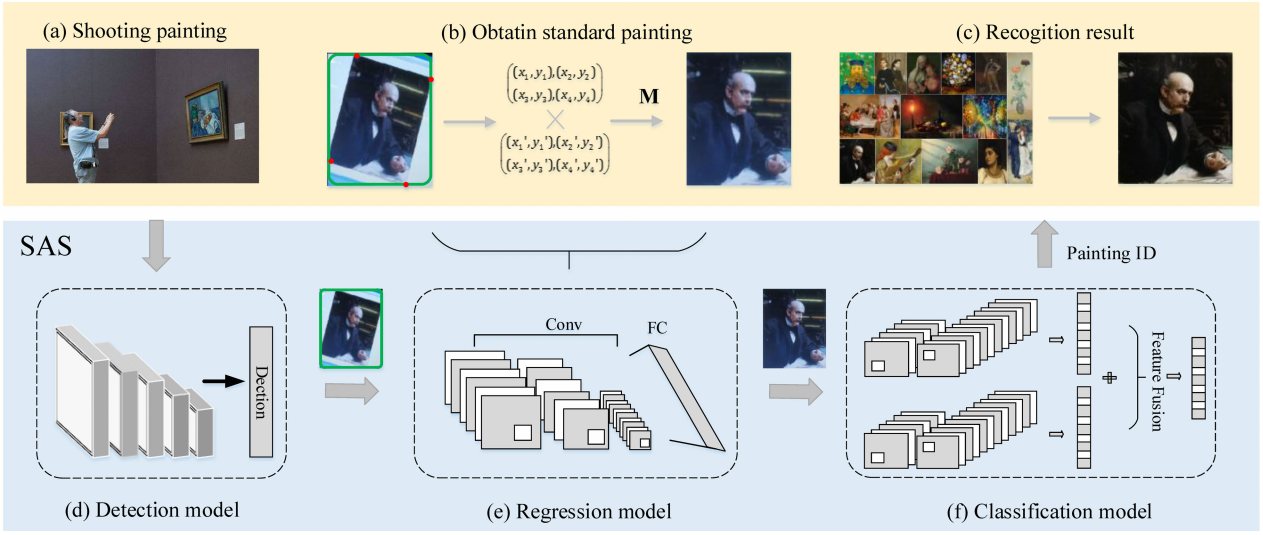


FIGURE 1. Pipeline of the proposed method. Shooting the painting we want to recognize (a), then the detection model (d) detects the painting from the captured image. Next, the regression model (e) locates the landmarks of the painting in the predict box, then rectifies the painting by using landmarks (b). *M* is the transformation matrix. Finally, the processed painting is feed into the classification model (f) for recognition (c).

directly. So, we first detect the painting from the shooting scene as to reduce the difficulty of recognition. We will obtain a predict box that contains the painting. It is worth noting that varied shooting angles may affect the detection results. Compared to the standard painting, the painting in the predict box may have varying degrees of deformation. Based on the detection network, we build a regression network to locate the landmarks of the painting in the predict box. Finally, we use the landmarks to rectify the painting and obtain the standard version.

In addition, we propose a new ultra-light painting classification pipeline through combining MobileNet backbone with Local Features Fusion (MobileNet-LFF). The MobileNet-LFF consists of two branches which both adopt the MobileNet (1.0 MobileNet-224 and 0.75 MobileNet-128) as the backbone. Two branches extract the global and local features of the painting, respectively. Finally, these features are fused at the fully connected layer.

The main contribution of this article are summarized as follows:

- We propose SSD-PLL, which can effectively eliminate the influence of interference factors on the detection results. Traditional detection algorithms usually provide a rigid bounding box as output, and the painting in the predict box may have different degrees of deformation. While our SSD-PLL directly outputs standard painting.
- We propose MobileNet-LFF. The design of this network structure allows the model to learn more diverse features. Moreover, the model is optimized to remove redundant parameters. This operation can effectively reduce the computational cost and let model achieve real-time recognition.
- Extensive experimental results show the effectiveness of the proposed algorithm. Our SAS have good overall performance and can be used in engineering applications.

The rest of the paper is organized as follows. In section II we summarize the related work in the field of detection and classification. Then in section III we introduce the new painting detection algorithm SSD-PLL, and we introduce the new painting classification model MobileNet-LFF in section IV. In section V, we present the experimental results. Finally we conclude this paper in section VI.

II. RELATED WORKS

Object detection is an important subject in the field of computer vision [11]–[17]. In general, the main task of object detection is to locate the interested target from the image and return the bounding box of each target [18]. The recently proposed object detection algorithms are mainly based on the deep learning model, which can be roughly divided into two categories, i.e., two-stage detector and one-stage detector. (1) Two-stage methods divides the detection problem into two stages [19], [20]. These methods first generate candidate regions, and then make regression and classification to predict whether a candidate contains an object. The representation of two-stage detectors are FPN [21] and R-CNN series. (2)One-stage method removes the process of producing candidate regions, while generating the classes and position coordinates of objects directly [22]. One stage detectors mainly represented by SSD [23] and YOLOv3 [24].

R-CNN [11] applied the deep learning theory to object detection for the first time, which can reach the industrial application level. After the development of R-CNN and Fast R-CNN, Girshick et al. proposed Faster R-CNN [25] in 2016. Faster R-CNN integrates region proposal, bounding box regression and classification into a network. Compared to RCNN and Fast-RCNN [26], Faster-RCNN greatly improves the comprehensive performance. Afterwards, He et al. proposed a new algorithm named R-FCN [27]. R-FCN can thus naturally adopt fully convolutional image classifier

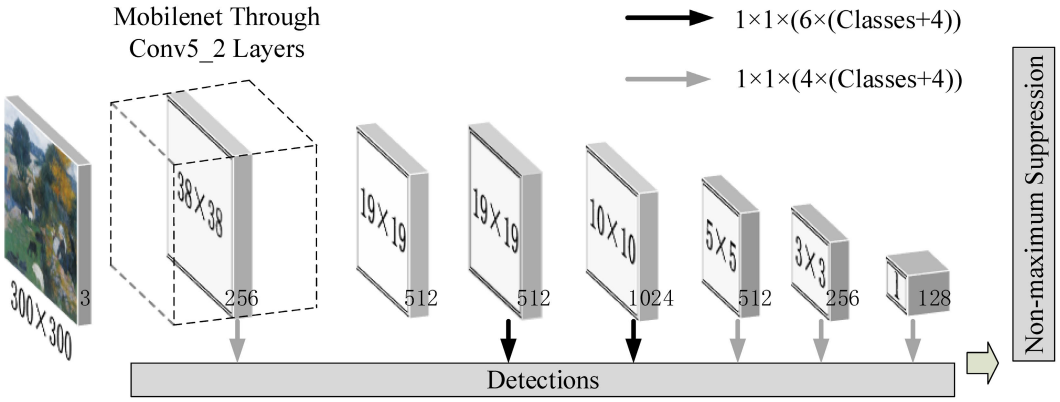


FIGURE 2. Structure of MobileNet based SSD. Our model mainly contains two parts. (a) MobileNet pre-trained from ImageNet dataset is adopted as the backbone for the early network layers. For this part, we use the feature map after Conv5_2, which is 8 times down sampled. (b) extra feature layers are added to the end of the truncated base network. These added layers decrease in both size progressively and allow predictions of detections at multiple scales [37].

backbones, such as the Residual Network. SSD is proposed by Liu et al., which completely eliminates the proposal generation and feature resampling stages. It encapsulates all computation into a single network and uses large numbers of anchor boxes to improve the recognition accuracy.

Deep convolutional neural networks (CNNs) have led to a series of breakthroughs for image classification [28]–[32]. Using CNNs for classification has become a popular research topic in the field of artificial intelligence. VGGNet [33] is developed by Visual Geometry Group, University of Oxford. By using small (3×3) convolution filters to increase the depth of network, VGGNet achieved the state-of-the-art accuracy on ILSVRC classification and localization tasks. GoogLeNet (Google Inception Net) [34] first appeared in the competition of ILSVRC 2014 and won the 1st place by a large margin. It replaces the final fully connected layer with a global average pooling layer and achieves excellent classification performance. ResNet [35] was proposed by He et al. of the Microsoft research institute. The authors trained residual nets with a depth of up to 152 layers, which is 8 $\times$ deeper than VGGNet. Note it still has a lower complexity. ResNet achieved 3.57% error on the ImageNet test set and won the 1st place on the ILSVRC 2015 classification task. The residual unit allows the raw input information to be transmitted directly to the later layer to protect the integrity of the information. MobileNet [36] is a class of efficient models for mobile and embedded vision applications. It is based on a streamlined architecture that uses depth-wise separable convolutions to build lightweight deep neural networks. MobileNet shows strong performance on ImageNet classification task even compared with above popular models.

Our SAS is designed for mobile devices. It is necessary to give consideration to latency and accuracy when selecting the model. Compared with the R-CNN series, SSD not only has a good performance on speed and accuracy, but also has a strong generalization ability. The SSD model trained on natural images still has good effects for detecting artworks. Given the limited resources of mobile devices, we need a base network with low latency and high precision. By comparing the structure and advantage of above networks, we find that MobileNet is the best choice for our task.

III. SSD-PLL

Our SSD-PLL mainly includes two parts, i.e., the painting detection unit and painting rectification unit. To get a standard painting for recognition, we first detect the painting from the shooting scene. Then, a regression network is used to locate the landmarks of the painting in the predict box. Finally, these landmarks are used to rectify the painting. Next, we will introduce the methods in more details.

A. PAINTING DETECTION

Our SAS is a very flexible framework, and each unit (detection, rectification and classification) can use different mainstream networks as the backbone. In this paper, we use SSD as the backbone of our detection unit.

SSD approach is based on a feed-forward convolutional network that adopts the regression strategy and anchors mechanism. It uses multiple anchors to extract features with different aspect ratios, which is more suitable than the global feature extraction method. Moreover, SSD predicts the positions and confidences of bounding boxes directly. This regression strategy can simplify the computational complexity and improve the real-time performance. In this paper, we adopt a novel detection framework named SSD_MobileNet (Figure 2). On the basis of maintaining high accuracy, this model structure can greatly reduce the redundant parameter.

Anchors Mechanism: SSD uses anchor boxes of different aspect ratios on the same feature maps, so as to enhance the anchor boxes' robustness on object shape changing. With each pixel as the center, we generate multiple anchor boxes of different shapes. Supposing that input image's height is h, width is w, scale $(s) = [0.25, 0.5, 0.75, 1]$, aspect ratio $(r) = [1, 2, 1/2, 1/3]$. The shape of anchor box is defined as

$$S = (w \cdot s \cdot \sqrt{r}, \frac{h \cdot s}{\sqrt{r}}) \tag{1}$$

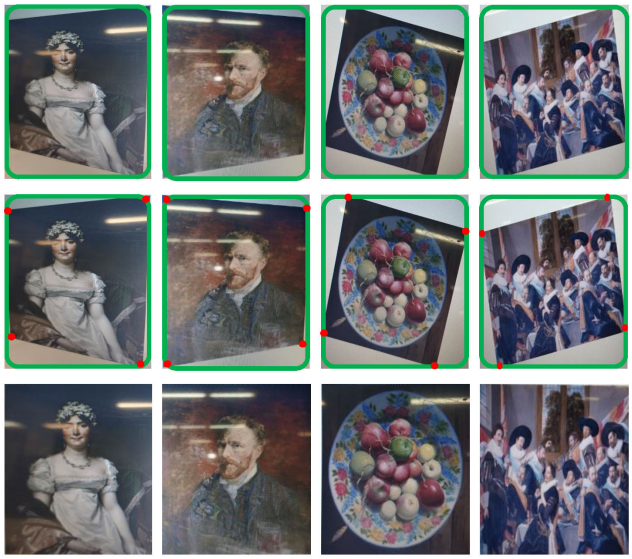


FIGURE 3. Some rectification results of different situations. The detection results, landmark localization results and rectification results are shown in the upper row, the middle row, and the lower row respectively. We can see that our approach can solve the problem of image distortion.

Loss Function: The SSD training objective can be extended to handle multiple object categories. Let $x_{ij}^{p} = \{1, 0\}$ be an indicator for matching the i-th default box to the j-th ground truth box of category p. The overall objective loss function is a weighted sum of the localization loss *(loc)* and the confidence loss *(conf)*.

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{2}$$

where N is the number of matched default boxes. The localization loss is a Smooth L1 loss between the predicted box (l) and the ground truth box (g) parameters. We regress to offsets for the center (cx, cy) of the default bounding box (d) and for its width (w) and height (h).

$$\begin{aligned} L_{\text{loc}}(x, l, g) &= \sum_{i \in Pos}^{N} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^{k} smooth_{L1}(l_i^m - g_j^{\wedge m}) \\ g_j^{\wedge cx,cy} &= (g_j^{cx,cy} - d_i^{cx,cy})/d_i^{w,h} \\ g_j^{\wedge w,h} &= \log(g_j^{w,h}/d_i^{w,h}) \end{aligned} \tag{3}$$

The confidence loss is the softmax loss over multiple classes confidences (c) and the weight term α is set to 1 by cross validation.

$$\begin{aligned} L_{conf}(x, c) &= -\sum_{i \in Pos}^{N} x_{ij}^{p} \log(c_i^{\wedge p}) - \sum_{i \in Neg} \log(c_i^{\wedge 0}) \\ c_i^{\wedge p} &= \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)} \end{aligned} \tag{4}$$

B. PAINTING RECTIFICATION

The actual scene is complex and varied, and different users have their own shooting habits. Traditional detection algorithms usually provide a rigid bounding box as output, the painting in the predict box may have different degrees of deformation, as shown in Figure 3 (a). The above situation will improve the difficulty of recognition. At detection stage, we prefer to get the standard painting directly. In this case, we propose a landmark location based idea, i.e., using a deep neural network to regress landmarks of the paintings, as shown in Figure 3(b). Finally, these landmarks are used to rectify the painting, as shown in Figure 3(c).

In order to locate the landmarks, we set up a regression neural network. The convolution layers are used to extract the features and the fully connected layers map the features from the representation space to the sample marker space. The output of the detection model (a predict box contains painting) serves as the input of the regression network. And the output of regression network is a group of landmark coordinates $(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$. The structure of the regression network is shown in Figure 4. When training, we use mean-square error (MSE) as the model loss function.

The main principle of painting rectification is perspective transformation. The equations of perspective transformations have eight unknowns, so a solution needs to find four sets of mapping points, four points that just define a three-dimensional space. By using landmark coordinates and boundary coordinates of predict box, we can get a transformation matrix M as follows.

$$M = \begin{pmatrix} a_{11} & b_{12} & c_1 \\ a_{21} & b_{22} & c_2 \\ a_{31} & b_{32} & c_3 \end{pmatrix} \tag{5}$$

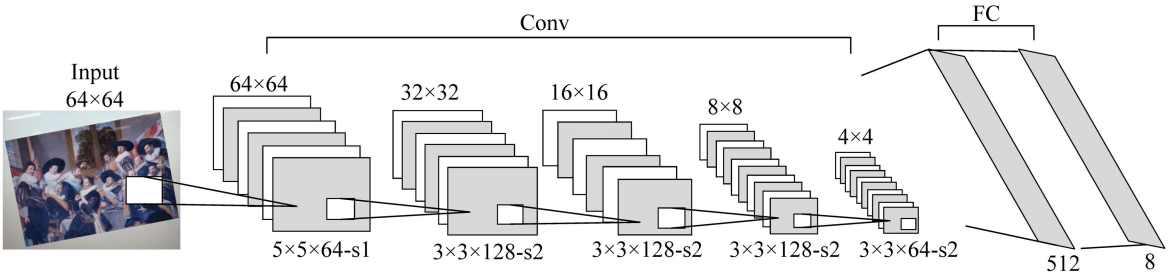


FIGURE 4. Structure of the regression network. More specifically, the input images are resized to 64×64. An appropriate input size can speed up the training convergence speed, and the training precision is better. −s1 and −s2 represent that the step size of convolution operation are 1 and 2 respectively.

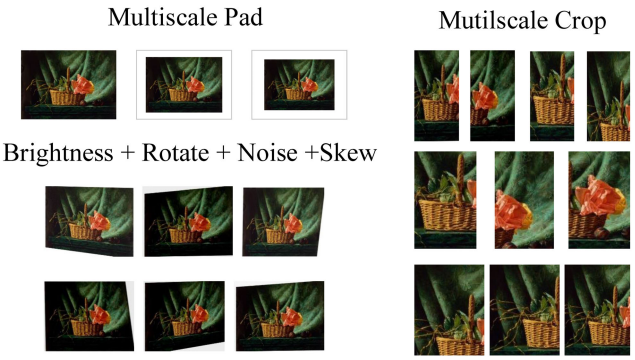


FIGURE 5. Examples of data augmentation. Note that the width/height ratio of the common mobile phones' screen are 9:16, 10:16 and 12:16.

Perspective transformation projects the image onto a new visual plane, via mapping from (x, y) to (X, Y, Z), then (X, Y, Z) to $\left(x^{'}, y^{'}\right)$, as shown in follows.

$$\begin{cases} X = a_{11}x + b_{12}y + c_1 \\ Y = a_{21}x + b_{22}y + c_2 \\ Z = a_{31}x + b_{32}y + c_3 \end{cases}, \quad x^{'} = \frac{X}{Z}, \; y^{'} = \frac{Y}{Z} \tag{6}$$

By using Equation (5) and (6), we can realize the perspective transformation and obtain the standard image. So far we have introduced the detection module. In next section, we will focus on the recognition module.

IV. MobileNet-LFF

We simplify the image recognition into a classification task and train the classifier by treating each painting as a separate class. In order to give the model better performance, we use the following tricks. (1) To make our model more robust to various situation, we use data augmentation for datasets processing. (2) Considering that sometimes local features can better reflect the differences between paintings which could contribute to the painting classification. In the classification model, we add the operation of local fusion. (3) Finally, in order to get better real-time performance, the model is optimized by an entropy-based metric loss function.

A. DATA AUGMENTATION

Unlike the usual data augmentation, all of our augmentation operations are for actual shooting scenes. Our process can be roughly divided into 4 steps. (a) The captured images may have different scales due to indeterminate shooting distance. To simulate this scene, we perform multi-scale pad on the original images. (b) To cope with the different lighting intensity of the shooting scene, we add extra operations such as brightness and noise. (c) Sometimes only a part of the painting is captured. At the same time, in order to adapt the screen width/height ratio of different mobile phone models, we also add multi-scale crop operation to the images. (d) Moreover, there are a few extreme, the regression model did not locate the landmarks of the painting accurately. The rectification results are not ideal. So we do the slight rotation and perspective transformation for the images.

In our process, each painting is treated as a separate class and subjected to a series of operations, the augmentation results are shown in Figure 5. The experiment results

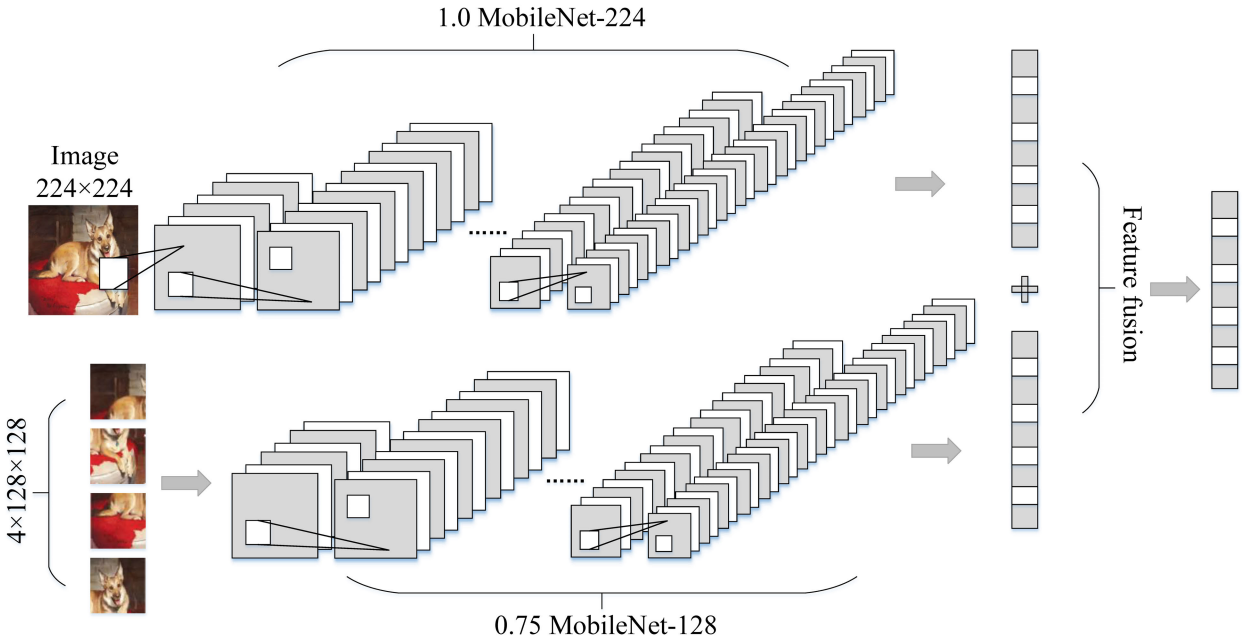


FIGURE 6. There are two streams in our MobileNet-LFF. The upper branch is the global stream to learn features from the whole image, and the lower branch is the part steam to learn local details form cropped parts. In particular, we choose 1.0 MobileNet-224 as upper branch, and choose 0.75 MobileNet-128 as lower branch. The size of input image of upper branch is 224×224. Then we cut the image into 4 different parts with the size of 128×128 and input them into lower branch.

demonstrate that the targeted data augmentation is helpful to improve the performance of our model.

B. LOCAL FEATURE FUSION

The image classification task is usually to extract the feature values of the fully connected layer for probability distribution estimation. Sometimes the final classification result is only semantically similar to the label image, while the local features are greatly different, because the high-level features have lost a lot of detail information. To solve this problem, we add features fusion operation to the classification network. By fusing the global features and local features, the obtained features contain both global abstract semantic information and local detail information.

Two branches of MobileNet-LFF are used to extract global features and local features respectively, then these features are fused at fully connected layer. The structure of MobileNet-LFF is shown in Figure 6.

Note that our classification model use MobieNet as the backbone. MobileNet is based on depth-wise separable convolutions which factorizes a standard convolution into a depth-wise convolution and a point-wise convolution. The depth-wise convolution applies a single filter to each input channel, and the point-wise convolution applies a $1\times$ 1 convolution to combine the outputs the depth-wise convolution [38]. This factorization has the effect of drastically reducing the computation and model size. Standard convolutions have the computational cost of

$$C_1 = D_K \times D_K \times N \times M \times D_F \times D_F \tag{7}$$

where the number of input channels is M, the number of output channels is N, the kernel size is $D_K \times D_K$ and the feature map size is $D_F \times D_F$. While depth-wise separable convolutions have the computational cost of

$$C_2 = D_K \times D_K \times M \times D_F \times D_F + N \times M \times D_F \times D_F \tag{8}$$

By expressing convolution as a two steps process of filtering and combining, we get a reduction in computation of

$$\frac{D_K \times D_K \times M \times D_F \times D_F + N \times M \times D_F \times D_F}{D_K \times D_K \times N \times M \times D_F \times D_F} = \frac{1}{D_K^2} + \frac{1}{N} \tag{9}$$

Depth-wise convolution is extremely efficient relative to the standard convolution. In addition, MobileNet uses two simple global hyper-parameters that achieve efficiently trade-off between latency and accuracy. The role of the width multiplier α is to thin a network uniformly at each layer. Note α has the effect of reducing computational cost. The second hyper-parameter to reduce the computational cost of a neural network is a resolution multiplier β. The computational cost for the core layers of network as depth-wise separable convolutions with width multiplier α and resolution multiplier β is

$$C_3 = D_K \times D_K \times \alpha M \times \beta D_F \times \beta D_F + \alpha N \times \alpha M \times \beta D_F \times \beta D_F \tag{10}$$

where $\alpha \epsilon$ (0, 1], $\beta \epsilon$ (0, 1]. Resolution multiplier is typically set implicitly so that the input resolution of the network is {224, 192, 160, 128}.

C. MODEL OPTIMIZATION

Although the base MobileNet is already lightweight, the application scenarios of SAS require the model to be smaller and faster. Based on this consideration, we need to further optimize the structure of the MobileNet-LFF.

In this paper, we adopt the entropy-based metric [39] to evaluate the importance of each filter. Entropy is a commonly used metric to measure the disorder or uncertainty in information theory. If a filter always produces similar values, we can believe that this filter contains less information, thus is less important. Then we can discard these unimportant filters to get a smaller model.

From another perspective, if a channel of activation tensor contains less information, its corresponding filter is less important, thus could be dropped. We first use global average pooling to convert the output of layer i, which is a $c \times h \times w$ tensor, into a $1 \times c$ vector, c is the channel number. For each channel j, we compute the entropy value.

To compute the entropy value of each channel, we divide it into m different bins, and calculate the probability of each bin. Finally, the entropy can be calculated as follows.

$$H_j = -\sum_{i=1}^{m} p_i \log p_i \tag{11}$$

where p_i is the probability of bin i, H_j is the entropy of filter j. A smaller score of H_j means channel j is less important in this layer, thus could be removed. In general, if some layers are weak enough, e.g., most of their activation are zeros, their entropy are relatively small. This strategy is illustrated in Figure 7.

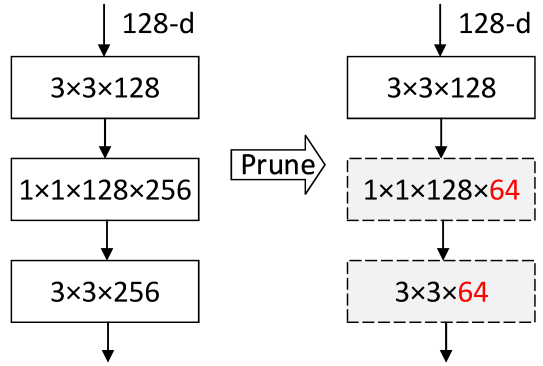


FIGURE 7. Illustration of our pruning strategy. We only prune the middle convolutional layer, which would simultaneously reduce the parameters of the next layer.

TABLE 1. The detailed distribution of the datasets.

Datasets	Sample size	After augmentation	Training set (%)	Validation set (%)	Test set (%)
Detection /Regression	500	/	80	0	20
TCPs	7500	1.17 million	70	15	15
OPs	8800	1.37 million	70	15	15
SAS	300	/	/	/	100

After the classification model training is completed, we use the above strategy to compress it. Then we integrate the classification model with the detection model and regression model, and transplant them to the mobile phone. It's worth noting that although the three models are trained separately, the testing is an end-to-end inference process, the same is true of actual usage.

V. EXPERIMENTS

A. DATASETS

In experiments, the datasets can be divided into three parts. (1) As shown in Figure 8, for classification model, we establish two large scale image databases, including 7,500 traditional Chinese paintings (TCPs) and 8,800 Oil paintings (OPs), respectively. Each painting is treated as a separate class. After data augmentation, 156 images will be generated in each class. The sample sizes of the datasets become 1.17 million and 1.37 million. 70% of them are taken as the training set samples, 15% of them are taken as the test set samples and the left are for validation. The data of training set, test set and validation set do not overlap each other.

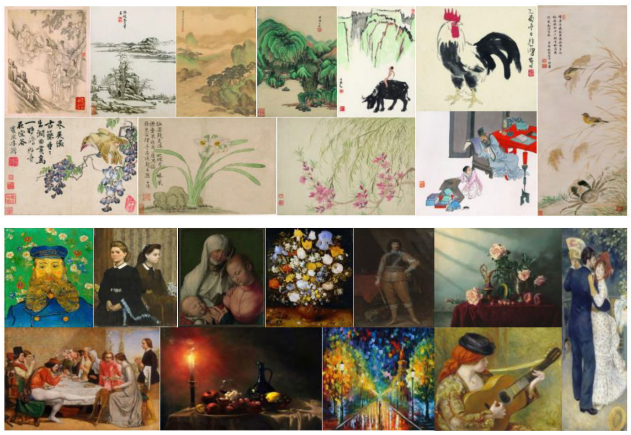


FIGURE 8. Examples of our painting datasets. The Traditional Chinese paintings (TCPs) and Oil paintings (OPs) are shown in the upper row and the lower row respectively.

(2) For detection model and regression model, we shoot 500 paintings from the museum and exhibition. These data are used for training and testing of detection model and regression model. 80% among them are taken as the training set samples, and 20% among them are taken as the test set.

(3) In order to validate the performance of the SASčí-models integration and end-to-end inference), we pick out 300 normal images form the TCPs and OPs and print out them. Five experimenters test these paintings in real environment (paintings on the wall etc.) respectively. Table 1 shows the details of above datasets.

B. MODEL ANALYSIS

In this experimental section, we compare our proposed method with the relevant methods. We implement all the experiments using Tensorflow, and the computing platform is Nividia GTX 1080Ti GUP. The backbones of detection model and classification model are both MobileNet, but the network configurations are different. Some configuration details are shown in Table 2.

TABLE 2. The configurations of MobileNets.

Model	Backbone	α	β
Detector	1.0 MobileNet-224 (Truncated at Conv5_2)	1.0	1.0
Classifier (Global Branch)	1.0 MobileNet-224	1.0	1.0
Classifier (Local Branch)	0.75 MobileNet-128	0.75	0.57

In our work, width multiplier α thin a network uniformly at each layer. Where $\alpha\epsilon$ (0, 1] with settings of 1 and 0.75. $\alpha = 1$ is the baseline MobileNet and $\alpha < 1$ is reduced MobileNet. Resolution multiplier β reduce the computational cost of the neural network. We apply it to the input image and the internal representation of every layer is subsequently reduced by the same multiplier. Where $\beta\epsilon$ (0, 1] with settings of 1 and 0.57, $\beta = 1$ is the baseline MobileNet and $\beta < 1$ is reduced computation MobileNet.

1) DETECTION MODEL ANALYSIS

We train and test different detection models on our detection dataset, respectively. Note that the input images' resolution of all models are 224×224. The test results of models are provided in Figure 9.

As shown in Figure 9, although the AP of SSD_MobileNet and YOLOv3 are slight lower than other models, their real-time performance are outstanding. In our SAS, recognition accuracy is determined by the classification model directly. The detection model only provides an auxiliary work to

eliminate the clutter backgrounds which may affect the recognition. Moreover, the adopted painting rectification algorithm can further improve the performance of the detector. Hence, we prefer the detection model have the lowest latency with an acceptable accuracy. Our SAS is a very flexible framework, and detection unit can use different mainstream networks as the backbone. Either SSD or YOLOv3 can be embedded directly into the system.

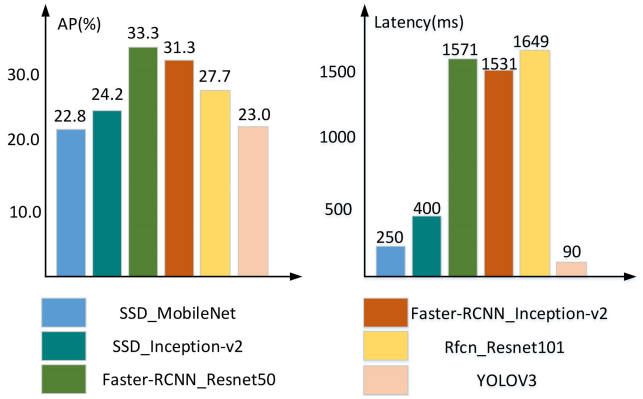


FIGURE 9. The test results consist of the AP (only one class: Painting) and latency. Latency is defined as the time consumption for the model detects a painting. We test 5 times and report the average values. AP refers to the evaluation criteria for COCO dataset (AP at IOU = 0.50:0.05:0.95) [40]. Note that the testing results are computed on the detection test set and are not strictly comparable to the official COCO test-dev data.

2) CLASSIFICATION MODEL ANALYSIS

In this section, we compare the proposed MobileNet-LFF with different classification models such as VGG16, Inception-v2 and base MobileNet. All models are pre-trained on ImageNet and with the last layer fine-tuned for the task of painting work recognition. We apply Adam with a weight decay of 0.0001 and momentum of 0.9 to optimize all models. The initial learning rate is 0.01, reduce two times ($\times 0.01$) after 6 and 11 epochs. We train 15 epochs with mini-batch size of 64 for all the models. The experimental results are illustrated in Figure 10. As we can see in Figure 10, although VGG16 can achieve 93.8% accuracy on the Ops dataset, it does not perform well on the TCPs dataset and only achieves 90.5% accuracy. Moreover, its overall real-time performance is poor. As for MobileNet and Inception-v2, MobileNet's real-time performance is much better, whereas the accuracy is roughly 1.4% lower than Inception-v2. The above three models cannot achieve the trade-off between the accuracy and latency. On the contrary, our MobileNet-LFF not only outperforms the Inception-v2 in terms of accuracy, but also achieves a similar latency with MobileNet. Our MobileNet-LFF has the best comprehensive performance for practical applications on mobile devices.

C. ABLATION STUDIES

To perform a detailed component analysis, we conducted the ablative experiments on SAS test set. Various modules are integrated into one system and implement an end-to-end inference. Xiaomi-6 is selected as the mobile platform, which owns the 2.4GHz CPU and 6GB RAM. The test simulates a real SAS usage scene. The experimental results are shown in Table 3.

TABLE 3. Ablation study based on different components of our method.

%	Backbone of Classifier	Augmentation Strategies	Painting Rectification	Accuracy
	MobileNet-LFF	×	√	82.2
	MobileNet-LFF	√	×	87.7
	Base MobileNet	√	√	88.5
SAS	MobileNet-LFF	√	√	**90.2**

The Effect of Data Augmentation: All of our augmentation strategies are for actual shooting scenes. We can see that data augmentation have prominent influence on our system. Our strategies restores the possible situations in reality and improves the adaptability of the model to the environment.

The Effect of Painting Rectification: Painting rectification reduces the difficulty of recognition to some extent. Without rectification operation, the recognition accuracy of SAS is 87.7%. If we add the painting rectification, the accuracy can increase 2.5%.

The Effect of Local Feature Fusion: Combining the local and global feature, more reasonable feature expression can significantly improve the accuracy by 1.7%. It can be inferred that the detailed information of the painting plays an important role.

D. DISCUSSION

The proposed SAS is based on the framework of detection and classification. As stated in the subsection V.B, we have compared detection models and classification models respectively. Considering that the platform of the SAS is mobile terminal, the trade-off between the latency and accuracy is our major concern. Our MobilNet-LFF not only retains the low latency property of the base MobileNet, but also improves the accuracy by using local feature fusion.

Referring to subsection V.C, the adopted data augmentation and painting rectification enhance the robustness of the system and significantly improve the recognition accuracy. In the mobile terminal, the optimized classification model is improved greatly in terms of real-time performance, but it also has a certain accuracy loss. Painting rectification and augmentation strategies make up for this shortcoming and keep the accuracy and real-time performance at a high level.

In the process of experiments, we find that the bad result of predict box will increase the difficulty of landmark location, thereby influence the rectification effect. Some failed cases are shown in Figure11. Generally, as long as the shooting condition is not particularly harsh, we can obtain favorable predict box and ideal rectification result. In addition, as mentioned in the IV.A, we remedy this situation through targeted augmentation strategies.

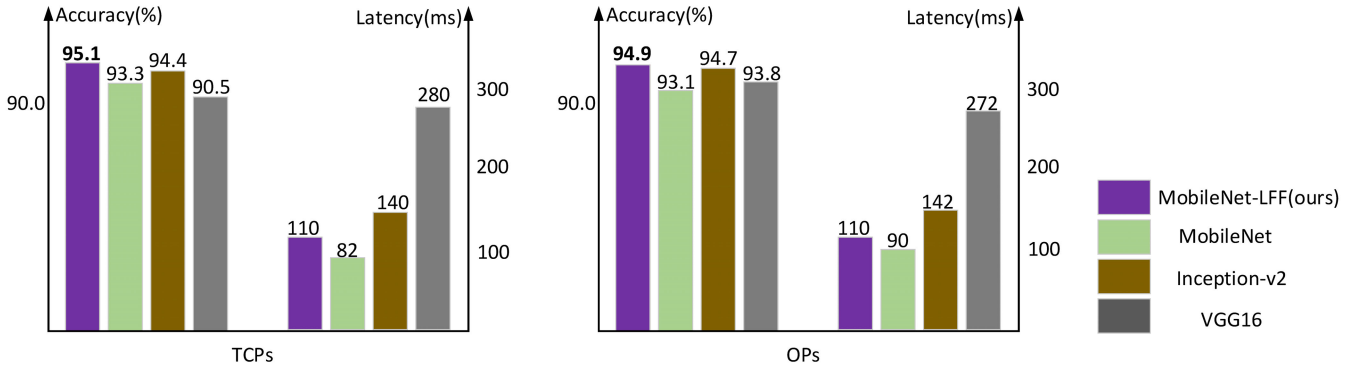


FIGURE 10. The test results consist of the accuracy and latency. We define the latency as the time consumption for the model classifies a painting. We test 5 times and report the average values. The accuracy is the final Top-1 accuracy of the classification model.

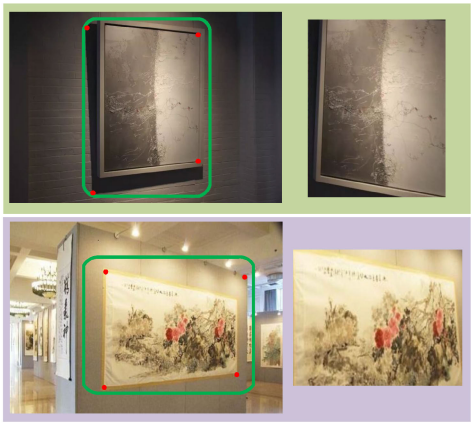


FIGURE 11. Bad cases of painting rectification.

VI. CONCLUSION

In this paper, we propose a painting recognition system SAS based on mobile device. SAS decomposes the painting recognition into the process of detection, rectification and classification. Painting rectification module and local feature fusion idea effectively improve the performance of the whole system. Our SAS can not only be used for painting recognition, but also has a good generalization in other artistic fields, such as calligraphy analysis. For future works, we consider using YOLOv3 as the backbone of our detection module. YOLOv3 has lower latency and satisfactory AP. In addition, for model optimization, we would like to adapt a dynamic pruning framework.

REFERENCES

[1] F. Gao, J. Nie, L. Huang, L.-Y. Duan, and X.-M. Li, "Traditional Chinese painting classification based on painting techniques," *Chin. J. Comput.*, vol. 40, no. 12, pp. 2871–2882, Dec. 2017.

[2] J.-C. Sheng and Y.-Z. Li, "Learning artistic objects for improved classification of Chinese paintings," *J. Image Graph.*, vol. 23, no. 8, pp. 1193–1206, 2018.

[3] V. N. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 2000.

[4] E. Gultepe, T. E. Conturo, and M. Makrehchi, "Predicting and grouping digitized paintings by style using unsupervised feature learning," *J. Cultural Heritage*, vol. 31, pp. 13–23, May/Jun. 2018.

[5] E. Levy, O. E. David, and N. S. Netanyahu, "Genetic algorithms and deep learning for automatic painter classification," in *Proc. Annu. Conf. Genetic Evol. Comput.*, 2014, pp. 1143–1150. [Online]. Available: https://www.researchgate.net/publication/266656414

[6] E. Cetinic, T. Lipic, and S. Grgic, "Fine-tuning convolutional neural networks for fine art classification," *Expert Syst. Appl.*, vol. 114, pp. 107–118, Dec. 2018.

[7] Q. Zou, Y. Cao, Q.-Q. Li, C.-H. Huang, and S. Wang, "Chronological classification of ancient paintings using appearance and shape features," *Pattern Recognit. Lett.*, vol. 49, pp. 146–154, Nov. 2014.

[8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[9] J. Zujovic, L. Gandy, S. Friedman, B. Pardo, and T. N. Pappas, "Classifying paintings by artistic genre: An analysis of features & classifiers," in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, 2009.

[10] A. Lecoutre, B. Negrevergne, and F. Yger, "Recognizing art style automatically in painting with deep learning," in *Proc. JMLR, Workshop Conf.*, vol. 80, 2017, pp. 1–16.

[11] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[12] A. Raghunanda, M. Mohana, P. Raghav, and H. V. R. Aradhya, "Object detection algorithms for video surveillance applications," in *Proc. Int. Conf. Commun. Signal Process.*, Apr. 2018, pp. 563–568.

[13] W. Rakumthong, N. Phetcharaladakun, and W. Wealveerakup, "Unattended and stolen object detection based on relocating of existing object," in *Proc. 3rd ICT Int. Student Project Conf.*, Mar. 2014, pp. 115–118.

[14] Q. Hu, S. Paisitkriangkrai, C. Shen, A. van den Hengel, and F. Porikli, "Fast detection of multiple objects in traffic scenes with a common detection framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1002–1013, Apr. 2016.

[15] B. Singh, M. Najibi, and L. S. Davis, "SNIPER: Efficient multi-scale training," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 9310–9320.

[16] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. CVPR*, Jun. 2018, pp. 6154–6162.

[17] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proc. CVPR*, Jun. 2018, pp. 3588–3597.

[18] X. Li, M. Ye, T. Li, and R. Center, "Review of object detection based on convolutional neural networks," *Appl. Res. Comput.*, vol. 34, no. 10, pp. 2881–2891, Oct. 2017.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[20] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 354–370.

[21] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, Jul. 2017, pp. 936–944.

[22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. ECCV*, 2016, pp. 21–37.

[24] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 91–99.
[26] R. Girshick, "Fast R-CNN," in *Proc. ICCV*, Dec. 2015, pp. 1440–1448.
[27] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," 2016, *arXiv:1605.06409*. [Online]. Available: https://arxiv.org/abs/1605.06409
[28] W. Jiang, Z. Wang, J. S. Jin, Y. Han, and M. Sun, "DCT–CNN-based classification method for the Gongbi and Xieyi techniques of Chinese ink-wash paintings," *Neurocomputing*, vol. 330, pp. 280–286, Feb. 2019.
[29] H. Gao, Y. Yang, S. Lei, C. Li, H. Zhou, and X. Qu, "Multi-branch fusion network for hyperspectral image classification," *Knowl.-Based Syst.*, vol. 167, pp. 11–25, Mar. 2019.
[30] J. Zhao, X. Mao, and L. Chen, "Speech emotion recognition using deep 1D & 2D CNN LSTM networks," *Biomed. Signal Process. Control*, vol. 47, pp. 312–323, Jan. 2019.
[31] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, Jul. 2017, pp. 4700–4708.
[32] Y. Zhang, K. Lee, and H. Lee, "Augmenting supervised neural networks with unsupervised objectives for large-scale image classification," in *Proc. ICML*, 2016, pp. 612–621.
[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556
[34] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: https://arxiv.org/abs/1502.03167
[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*. [Online]. Available: https://arxiv.org/abs/1512.03385
[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861
[37] T. Cong, L. Yongshun, and Z. Kedong, "Object detection method of multi-view SSD based on deep learning," *Infr. Laser Eng.*, vol. 47, no. 1, pp. 290–298, Jan. 2018.
[38] J. Yang, M. Z. Chen, Z. Q. Wu, L. N. Chen, and Y. Lin, "Research on video target detection based on SSD convolution network," *J. Univ. South China (Sci. Technol.)*, vol. 32, no. 1, pp. 78–86, 2018.
[39] J.-H. Luo and J.-X. Wu, "An entropy-based pruning method for CNN compression," 2017, *arXiv:1706.05791*. [Online]. Available: https://arxiv.org/abs/1706.05791
[40] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," 2017, *arXiv:1611.10012*. [Online]. Available: https://arxiv.org/abs/1611.10012

ZHENYU WANG received the B.Sc. degree from the National Defense University of Science and Technology, China, in 1997, the M.Sc. degree from Xi'an Jiaotong University, China, in 2003, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2007. He is currently an Associate Professor with the School of Control and Computer Engineering, North China Electric Power University, Beijing, China. His research interests include pattern recognition, computer vision, and machine learning.

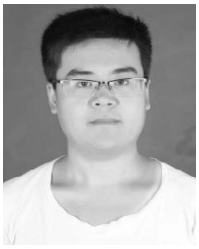
JIE LIAN received the B.S. degree from the College of Information Engineering, Southwest University of Science and Technology, Mianyang, China, in 2017. He is currently pursuing the M.S. degree with the School of Control and Computer Engineering, North China Electric Power University, Beijing, China. His research interests include deep learning, pattern recognition, and computer vision.

CHUNFENG SONG received the B.Sc. degree from the Qilu University of Technology, China, in 2012, and the M.Sc. degree from North China Electric Power University, China, in 2016. He is currently pursuing the Ph.D. degree with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China. His research interests include image segmentation, data clustering, person recognition, and deep learning.

ZHAOXIANG ZHANG received the bachelor's degree in circuits and systems from the University of Science and Technology of China, Hefei, China, in 2004, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2009, under the supervision of Prof. T. Tan. In 2009, he joined the School of Computer Science and Engineering, Beihang University, Beijing, as an Assistant Professor from 2009 to 2011, an Associate Professor from 2012 to 2015, and the Vice Director of the Department of Computer Application Technology from 2014 to 2015. In 2015, he returned to the Institute of Automation, Chinese Academy of Sciences, where he is currently a Professor with the Research Center for Brain-Inspired Intelligence. He specifically focuses on braininspired neuronetwork and brain-inspired learning. His current research interests include computer vision, pattern recognition, and machine learning.

WEI ZHENG received the B.S. degree from the School of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao, China, in 2017. He is currently pursuing the M.S. degree with the School of Control and Computer Engineering, North China Electric Power University, Beijing, China. His research interests include deep learning, pattern recognition, and computer vision.

SHAOLONG YUE received the B.Sc. degree from the Shandong University of Technology, China, in 2018. He is currently pursuing the master's degree with North China Electric Power University, China. His research interests include pattern recognition, computer vision, and machine learning.



SENRONG JI received the B.Sc. degree from Qufu Normal University, China, in 2018. He is currently pursuing the master's degree with North China Electric Power University, China. His research interests include pattern recognition, computer vision, and deep learning.