Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 16, No. 2, 2026

Research on Chinese Cuisine Image Recognition Using Semi-Supervised Convolutional Neural Networks

Yuyang Lei

School of Digital Economy and Business, Chongqing College of Mobile Communication, Chongqing 401420, China

Abstract: With the rapid development of image recognition and deep learning, food image classification has achieved remarkable results. Chinese cuisine image recognition faces challenges due to diverse dishes and similar ingredients. This paper constructs a dataset containing both labeled and unlabeled Chinese food images. To address slow training and inference of traditional CNNs, we adopt lightweight networks. MobileNetV3-small reduces training time by 41.1% and improves inference speed by 21% compared to ResNet-34.To better capture correlated features, we propose MobileNetV3-small-sa with a self-attention mechanism, which improves accuracy by 2.2% over the base model. Given the high cost of labeling, we apply semi-supervised learning. MixMatch uses both labeled and unlabeled data. With only 1/7 labeled data plus 10,952 unlabeled images, it achieves 75.9% accuracy, while supervised learning on the full labeled set yields 80.1%. MixMatch improves accuracy by over 10% when labeled data is scarce. Experiments show that the self-attention model outperforms traditional networks, achieving 3.8% higher accuracy than VGGNet16 and 3.2% higher than ResNet34.

Keywords: Chinese Food Image Recognition; Lightweight Networks; Self-Attention; Semi-supervised Learning.

1. Introduction

Food image recognition is an artificial intelligence technology closely related to our daily lives. With the gradual refinement and advancement of deep learning techniques, it has gradually become a high-interest topic in academic research. Benefiting from this technology, we can accomplish tasks that were originally labor-intensive for human vision and the brain, such as recognizing and classifying specific foods in images.

Recognition oriented toward dishes has significant value in the catering industry, food apps, and other fields. Currently, most dish-oriented recognition studies focus on single-label recognition, i.e., only one dish per image, as in the ChineseFood dataset used in this paper. Xin Chen et al. released a large-scale food dataset named ChineseFood[1], and proposed a hybrid CNN model, achieving 81.55% accuracy on the test set.

Traditional food image recognition involves two key steps: food image feature extraction and classification model training. Among these, food image feature extraction is the core step. Nguyen et al. employed Non-Redundant Local Binary Pattern (NRLBP) to capture local visual features of dishes and used their proposed shape descriptor to obtain global structural features[2]. Finally, they combined these two features for food image recognition. Jiang et al. proposed a multi-scale model, MSMVFA, achieving over 90% accuracy on public datasets[3].

Before the maturity of deep learning, researchers primarily extracted low-level and mid-level features manually. Representative methods include SIFT, HOG, and SURF. With the development of deep learning, extracting deep features from images using CNNs has become the main research approach. In 2014, Simonyan K et al. proposed VGGNet, which replaced large kernels in AlexNet with multiple 3×3 convolutional kernels, deepening the network and improving model generalization and recognition accuracy[4]. The same year, Szegedy C et al. proposed GoogLeNet, which introduced the Inception structure while increasing network depth, significantly reducing parameters and achieving first place in the ImageNet competition with 93.3% Top-5 accuracy[5]. He K et al. pioneered ResNet in 2015, with network depth reaching up to 152 layers, yet model complexity was lower than VGGNet, and network performance was greatly improved[6]. The residual module proposed in ResNet largely solves the degradation problem in deep networks. ResNet achieved first place in accuracy in competitions such as ImageNet and can be considered a milestone in deep learning development. In 2016, Huang G et al. proposed DenseNet[7]. The core idea of DenseNet is to solve the vanishing gradient problem in deep neural networks through dense connections and to improve feature reuse.

The traditional neural network models mentioned above generally suffer from large parameter sizes and high computational costs. With the proliferation of mobile devices such as smartphones, improving the practicality and efficiency of neural networks has become increasingly important. Forrest N. Iandola et al. proposed SqueezeNet based on AlexNet[8]. SqueezeNet uses 1×1 convolutional kernels instead of 3×3 kernels and designs Fire modules, reducing the parameter count to 1/50 of AlexNet while maintaining the same accuracy. Andrew G. Howard et al. proposed MobileNetV1[9], which decomposes standard convolutions into depthwise convolutions and pointwise convolutions, using depthwise separable convolution to reduce parameter count and computational cost. This design reduces the model burden while maintaining relatively high accuracy, making it ideal for resource-constrained environments.

The research focus of this paper is to optimize CNN models for practical Chinese food image recognition to improve performance. The main research contents are as follows: (1) To address the large parameter size and slow recognition speed of traditional CNNs, this paper investigates how to improve training speed, maintain recognition accuracy, and increase recognition speed; (2) The lightweight CNN MobileNetV3 proposed by Google incorporates a channel attention mechanism, which can only extract channel-wise

importance. Since Chinese food images contain rich contextual information, this paper introduces a self-attention mechanism into MobileNetV3-small to extract correlations between local and global features, naming the model MobileNetV3-small-sa; (3) Labeling Chinese food images is labor-intensive and time-consuming, and manual labeling is difficult for similar dishes. This paper investigates how to effectively utilize a large number of unlabeled samples for training to achieve recognition accuracy close to that of supervised learning.

The main research method of this paper is controlled experiments. First, a dataset is constructed by selecting 30 different categories of Chinese dish images from the ChineseFood dataset, which are divided into training set, validation set, test set, and unlabeled dataset. Then, using the Pytorch framework, different CNN models are trained on the dataset. Based on model performance on the validation set, hyperparameters such as learning rate, decay rate, batch size, and data augmentation methods are tuned to find the optimal combination, and the recognition efficiency and accuracy of different CNNs are compared. Finally, semi-supervised learning is used for training, comparing the performance of supervised and semi-supervised methods, as well as the effect of using semi-supervised methods with different amounts of labeled data.

2. Proposed Method

2.1. MobileNet with Self-Attention

Salman Khan et al. made a comprehensive review on the application of transformer in the field of vision[11], and proposed the future research direction and possible work. Therefore, in this paper we attempt to embed the self attention mechanism of transformer into the lightweight convolutional neural network mobilenet.The proposed MobileNetV3-small-sa architecture is based on MobileNetV3-small[10], adding a self-attention[12] module on top of the bneck structure to extract global contextual information, named bneck-sa. The overall structure of the bneck-sa architecture is shown in Figure 1.

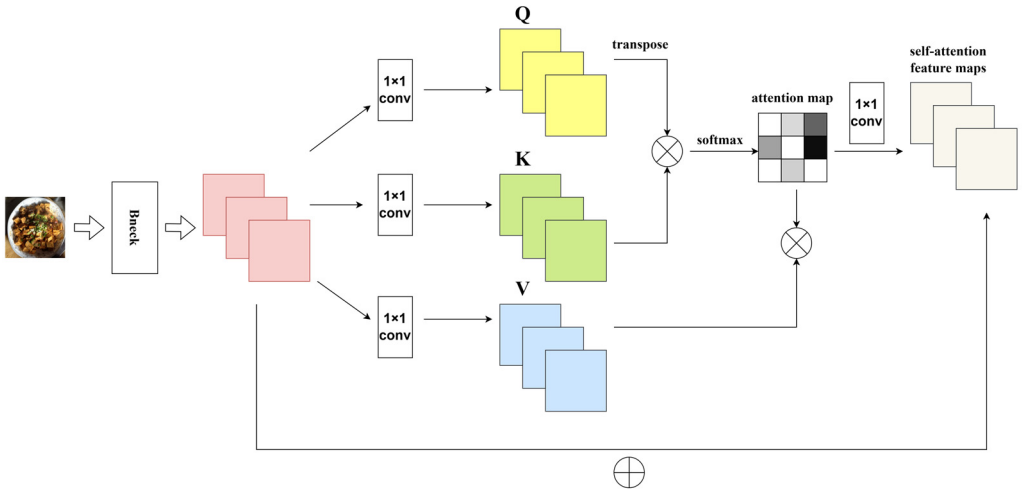

Fig 1. Structure of bneck-sa

Table 1. MobileNetV3-small-sa structure

Input	Operator	Exp size	#out	SE	SA	NL	s
$224^2 \times 3$	*conv2d*,3×3	-	16	-		HS	2
$112^2 \times 16$	*bneck*,3×3	16	16	✓	✓	RE	2
$56^2 \times 16$	*bneck*,3×3	72	24	-	✓	RE	2
$28^2 \times 24$	*bneck*,3×3	88	24	-	✓	RE	1
$28^2 \times 24$	*bneck*,5×5	96	40	✓	✓	HS	2
$14^2 \times 40$	*bneck*, 5×5	240	40	✓	✓	HS	1
$14^2 \times 40$	*bneck*, 5×5	240	40	✓		HS	1
$14^2 \times 40$	*bneck*,5×5	120	48	✓		HS	1
$14^2 \times 48$	*bneck*,5×5	144	48	✓		HS	1
$14^2 \times 48$	*bneck*,5×5	288	96	✓	✓	HS	2
$7^2 \times 96$	*bneck*, 5×5	576	96	✓	✓	HS	1
$7^2 \times 96$	*bneck*, 5×5	576	96	✓	✓	HS	1
$7^2 \times 96$	*conv2d*,1×1	-	576	✓		HS	1
$7^2 \times 576$	*pool*,7×7	-	-	-		-	1
$1^2 \times 576$	*conv2d*,1×1,*NBN*	-	1024	-		HS	1
$1^2 \times 1024$	*conv2d*,1×1,*NBN*	-	*k*	-		-	1

The overall flow of the bneck-sa architecture is as follows: For an input image, features are first extracted through the standard bneck architecture. Then, three 1×1 convolutions (output channel number generally set to 1/k of the original

channel number C, typically k=1,2,4,8; in this paper we use k=4) are applied to the extracted feature map to generate Query, Key, and Value for the self-attention mechanism. The dot product of the Query and Key feature maps is computed to obtain attention weights (in practice, the feature maps are flattened for matrix multiplication). The weights are then applied to the Value feature map, followed by a 1×1 convolution to obtain a feature map with the same dimensions as the input feature map. Finally, a residual connection is added. Depthwise separable convolutions make the bneck architecture much less computationally expensive than traditional CNNs, while adding the self-attention mechanism endows the bneck-sa architecture with the ability to extract feature correlations. Neural Architecture Search (NAS) is used to find the optimal structure for the dataset. The specific model structure is shown in Table 1.

1) Exp size indicates the channel number after the first 1×1 convolution in bneck (dimensionality expansion). Output channels refer to the channel number after the last 1×1 convolution in bneck. SE indicates whether channel attention is used, NL indicates the nonlinear function (activation function type), RE indicates ReLU, HS indicates h-swish, and SA indicates whether self-attention is used.

2.2. MobileNet with MixMatch

Semi-supervised learning is a machine learning method that utilizes unlabeled data to improve model performance. Traditional supervised learning relies on large amounts of labeled data for training, but acquiring labeled data is often expensive in terms of manpower and time. Semi-supervised learning reduces the dependence on labeled data by fully utilizing unlabeled data, enabling more efficient model training.

Many recent semi-supervised learning methods introduce an additional loss term computed on unlabeled data to encourage the model to generalize better on unseen data. This additional loss term is often called a semi-supervised loss or self-supervised loss. It can be defined based on intrinsic properties of unlabeled data or relationships between data points.

The goal of semi-supervised learning is to jointly use labeled and unlabeled data for model training to improve model performance and generalization ability. Unlabeled data act as a regularization mechanism during training, providing additional information to guide the model's learning process. By leveraging the distributional characteristics of unlabeled data, the model can better capture the underlying structure and representation of the data.

However, semi-supervised learning requires substantial additional computational power. Traditional CNNs such as VGGNet and ResNet have large parameter sizes, and using semi-supervised learning algorithms would lead to excessively long training times. Therefore, this paper proposes using a "lightweight neural network + semi-supervised learning" approach to reduce model training time. The MixMatch algorithm, proposed by David Berthelot et al. in 2019, which integrates various semi-supervised learning methods[13], is chosen.

3. Experimental Results and Analysis

3.1. Dataset

The dataset used in this paper is the ChineseFood dataset (Figure 2). ChineseFood is a collection of a large number of labeled Chinese dish images. The dataset was captured by users of Douguo and labeled by Midea employees. It contains 145,065 images for training, 20,253 images for validation, and 20,310 images for testing. The dataset includes 208 different categories of Chinese food images, such as Mapo Tofu, Sour and Spicy Shredded Potatoes, and Fish-Fragrant Eggplant. There are different dishes made from the same ingredient, e.g., Mapo Tofu, Pan-fried Tofu, and Stinky Tofu made from tofu, or Fish-Fragrant Eggplant, Garlic Eggplant, and Minced Pork Eggplant made from eggplant. This characteristic of small inter-class differences increases recognition difficulty, requiring neural networks to distinguish different cooking methods of the same ingredient based on local features.

For this experiment, 30 dish categories were selected. The training set contains 100 images per category, totaling 3,000 images; the validation set contains 3,112 images, approximately 100 per category; the test set contains 2,483 images. There are 10,952 unlabeled images, with roughly the same number (about 400) of each dish category in the unlabeled dataset (the unlabeled images were taken from a subset of the original training set, which originally contained 22,800 images).

Fig 2. Examples from ChineseFood dataset

3.2. Supervised Learning Experiments

3.2.1. Experimental Procedure

The images in the dataset used in this paper are of varying sizes. During training, all images were resized to 224×224. To enhance model robustness, images were first randomly rotated, color-jittered, grayscaled, and randomly cropped, and finally resized.

The Adam algorithm was chosen as the training optimizer. Adam has hyperparameters such as learning rate, decay rate, β^1, β^2 (exponential decay rates), and a constant eps to prevent division by zero. β^1 and β^2 are similar to the weight controlling historical gradient momentum in Momentum and the weight controlling historical squared gradients in RMSProp, respectively. Using validation set accuracy for hyperparameter tuning revealed that pretrained models do not require a large learning rate; convergence was poor when the

learning rate exceeded 0.001. Through experimentation, the learning rate was set to 0.0003, the decay coefficient to 0.0008, and other parameters to default values. The cross-entropy loss function was used, and the batch size was set to 40.

Furthermore, to accelerate the convergence of model training, all training used pretrained models. Pretrained models significantly reduce training time and often achieve better performance. Pretrained models converge without excessive iterations, so the number of iterations was set to 200. Training was performed on GPUs. For testing, the best training model was selected as the test model, and the recognition efficiency and Top-1 accuracy of each model were compared. All parameter settings are shown in Table 2.

Table 2. Table of training parameters

Parameter	Value
input size	224×224
batch size	40
learning rate	0.0003
weight_decay	0.0008
β^1	0.9
β^2	0.999
eps	1×10^{-8}

3.2.2. Comparative Experiments with Different CNNs

To evaluate the performance of the improved MobileNet-V3, comparisons were made with traditional VGGNet and ResNet models on the Chinese food image dataset. Since the sample sizes are balanced, accuracy was chosen as the evaluation metric. For computational cost, model parameter size, training time consumption, and recognition speed were examined.

As seen in Table 3, MobileNet has far fewer parameters than traditional CNNs like VGGNet and ResNet, by orders of magnitude. During actual training, the VGGNet16 network required too much time per iteration (over 12 minutes). The recognition speed of VGGNet16 on the test set was much slower than other models; its recognition accuracy was 62.9%, slightly lower than other models. Compared to ResNet34 (a shallower ResNet model), the MobileNetV3-small model reduced training time by approximately 40% and recognition time per image by 21%, requiring only 4.5 milliseconds per image. Its accuracy on the test set also exceeded that of ResNet34, reaching 64.5%.

The MobileNetV3-small-sa model, which adds a self-attention mechanism, essentially builds dependencies or correlations between different regions in an image. It adds only 0.1 million parameters, and training time and recognition speed are nearly unchanged compared to the original model, but accuracy on the test set improves by 2.2%. Experimental results show that adding a self-attention mechanism to MobileNet can effectively extract correlation information in food images, thereby enhancing model generalization.

Table 3. Model Performance Comparison

Model	Arguments (million)	Training time (s)	Recognition speed(ms)	Accuracy (%)
VGGNet16	138.36	754.4	49.2	62.9
ResNet34	21.80	32.6	5.7	63.5
MobileNetV3-small	2.54	19.2	4.5	64.5
MobileNetV3-small-sa	2.64	20.5	4.7	66.7

1)Parameter count refers to the total number of parameters when the output category is 1000; training time refers to the average time per iteration; recognition speed refers to the time taken to classify each image on the test set; accuracy is Top-1 accuracy.

Figure 3 shows the loss function during training. All three models initially reduced loss quickly and gradually converged after about 100 iterations. On the training set, the cross-entropy loss of MobileNetV3-small was smaller than the other two models, at 0.37.

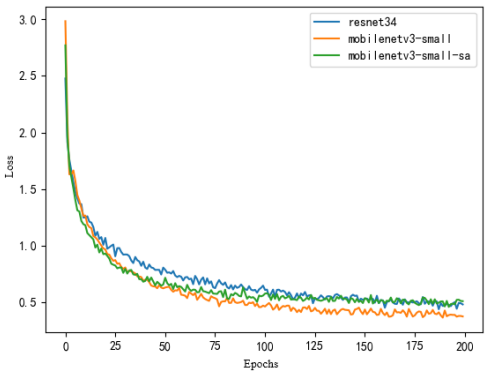

Fig 3. Training loss curves

Figure 4 shows the accuracy on the training and validation sets over iterations (left: validation set, right: training set). On the training set, accuracy continuously increased, with all three models exceeding 80%. On the validation set, MobileNetV3-small and MobileNetV3-small-sa performed well in the first few tens of iterations, surpassing ResNet34 and approaching 70%. However, as training progressed, validation accuracy did not increase but rather decreased, and ResNet34's accuracy fluctuated, indicating overfitting: the models focused too much on features of the training set images, reducing generalization. In the dataset we used, the training and validation sets have the same number of images, and some features in the validation set may not be present in the training set. To address this, the best approach is to improve data quality or increase the training set size. When no more labeled data is available, adding more unlabeled data can also alleviate this issue.

3.3. Experiments Based on MixMatch

3.3.1. Experimental Procedure

Based on the MobileNetV3-small-sa model used in Section 3.2, the MixMatch algorithm was added, and the dataset was augmented with 10,952 unlabeled images. Similar to the experiments in Section 3.2, data augmentation was also applied for the MixMatch-based experiments. However, it

was observed that using too many augmentation methods led to a multiplicative increase in training time because the unlabeled dataset requires K augmentations, which consumes substantial computational resources when a large amount of unlabeled data is present. Moreover, in the MixMatch algorithm, mixup operations are performed on both labeled and unlabeled data, which significantly affects recognition performance[14]. No significant difference in recognition accuracy was observed between using extensive data augmentation and using only random rotation and center cropping.

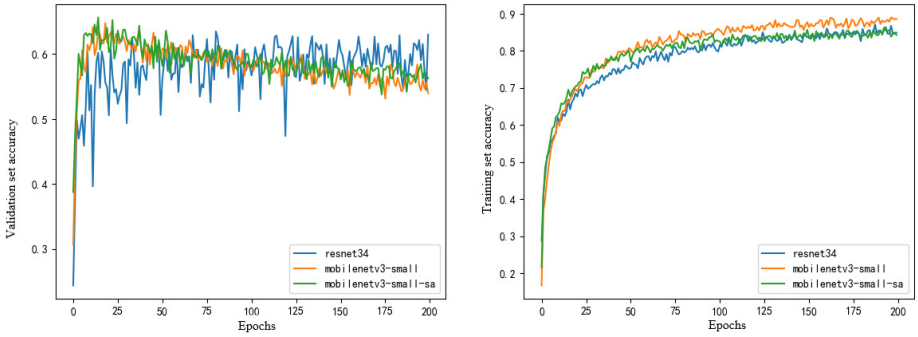

Fig 4. Training and validation accuracy curves

For other hyperparameters, the classification accuracy on the validation set with different combinations was used for selection. The number of augmentations K for unlabeled data was set to 2. Increasing K not only increases training time but also does not significantly improve model performance; when K=4 or K=5, it even reduces recognition performance. The sharpen temperature T was set to 0.5, the beta distribution parameter α was set to 0.5, the loss function parameter λ_u was set to 75, and λ_u was linearly ramped up as follows:

$$\lambda_u = 75 \times iter_{now} / 16000 \quad (1)$$

where iter_now is the current step (each step indicates one parameter update, i.e., completing training on one batch of data), and 16000 steps is an empirically derived value.

3.3.2. Experimental Results and Analysis

Training the model using the MixMatch algorithm requires more computational resources due to the several-fold increase in data volume. As shown in Table 4, each parameter update takes about 3 minutes, but compared to the original method, the recognition accuracy on the test set improves by 9.8%. This indicates that the model effectively utilizes image information from the unlabeled dataset and learns more features of Chinese food images. Both methods use the same MobileNetV3-small-sa model, so recognition speed on the test set is identical.

Table 4. Comparison of the performance of the MixMatch algorithm with the original algorithm

Method	Training time(s)	Accuracy(%)
MobileNetV3-small-sa	20.5	66.1
MobileNetV3-small-sa+Mixmatch	182.1	75.9

Figure 5 shows the training set accuracy during training. In the early stage of training, the original method achieved slightly higher accuracy than the MixMatch method, but the original method converged quickly. The model using MixMatch gradually converged after about 150 iterations and achieved higher accuracy on the training set than the original method, reaching approximately 87%.

On the other hand, as shown in Figure 6, when using the original training method, the model's accuracy on the validation set initially improved but then gradually declined as training progressed. This phenomenon typically indicates overfitting: the model over-adapts to the training set data and loses predictive power on new data.

However, when the MixMatch algorithm was introduced into the training process, overfitting was greatly alleviated. The model using this method showed gradually increasing accuracy on the validation set over time, reaching up to 77%. Clearly, in our task, the MixMatch algorithm effectively utilizes unlabeled data to expand the scale and diversity of training data. This process significantly improves the model's adaptability to new data and prediction accuracy, largely avoiding overfitting.

Overfitting is common when training deep learning models, and the MixMatch algorithm is an effective solution. Experimental results confirm the feasibility of using unlabeled data to improve model generalization in Chinese food recognition tasks.

To investigate the effectiveness of the MixMatch algorithm on larger or smaller labeled dish image datasets, additional controlled experiments were conducted by setting the number of images per category to 25, 50, 200, and 500.

As shown in Table 5, "ALL" indicates training with all labeled data (22,800 images as the training set), achieving 80.1% recognition accuracy. The less labeled data available, the more significant the improvement brought by the MixMatch algorithm. When there were only 25 labeled images per category, the supervised method achieved only 49.8% accuracy. Using MixMatch improved accuracy by 14.5%, reaching 64.3%. When there were 100 labeled images per category, MixMatch achieved 75.9% accuracy, already close to the supervised method trained with all data. Adding more labeled data beyond this point yielded only marginal improvements. Even increasing the labeled data fivefold improved recognition accuracy by only 2.6%.

Experimental results show that when labeled data is scarce, supervised learning performs poorly, and the MixMatch algorithm can significantly improve model performance, though still lagging behind models trained with all labeled images. Increasing the amount of labeled data at this stage rapidly improves model performance. However, once the labeled data reaches a certain amount, further increases yield diminishing returns, as features in the added data may have already been learned from unlabeled data. The MixMatch

method achieves recognition accuracy comparable to supervised learning while requiring less than one-seventh of the labeled data.

Table 5. Comparison of training effect with different amount of labeled data

Method \ Labels	25	50	100	200	500	ALL
MobileNetV3-small-sa	49.8%	58.8%	66.1%	69.1%	77.4%	80.1%
MixMatch+MobileNetV3-small-sa	64.3%	69.2%	75.9%	76.5%	78.5%	-

4. Conclusion

With the development of deep learning technologies in computer vision and people's increasing demand for healthy eating, research on Chinese food image recognition has gradually increased. After constructing a dataset containing both labeled and unlabeled data, this paper conducted experiments on an improved MobileNetV3 network and achieved promising results.

Compared to existing related research, the innovations of this paper are mainly reflected in the following aspects. First, the lightweight CNN MobileNet is introduced into Chinese food image recognition. Chinese food image recognition is often applied on terminal devices with limited computational power. To enable real-world applications, models need to be simplified. Compared to traditional deep neural networks, lightweight networks consume fewer computational resources and have higher recognition efficiency, creating conditions for deploying models on mobile devices. The lightweight network MobileNetV3 not only has far fewer parameters than traditional CNNs like VGGNet and ResNet but also, by incorporating channel attention mechanisms and inverted residual structures, achieves better training time and recognition speed on our dataset, as well as higher classification accuracy than other networks. Second, among current studies on food classification, some use spatial attention mechanisms to obtain salient regions in images, others use channel attention mechanisms to suppress noise, or hybrid attention mechanisms combining both. Inspired by the self-attention mechanism in Transformers, this paper proposes a lightweight network incorporating self-attention: the MobileNetV3-small-sa model. This model adds very few parameters and computational cost while extracting spatial correlation features in images, improving classification accuracy by 2.2% over the original model. Finally, to address the high cost of obtaining labeled Chinese food images, this paper proposes using a semi-supervised learning method, MixMatch, which requires only a small amount of labeled data and a large amount of unlabeled data to achieve performance close to that of supervised learning. A comparative experiment between supervised and unsupervised learning methods showed that recognition accuracy is significantly improved at the cost of an acceptable increase in training time.

However, this study has certain limitations. First, the recognition accuracy of the proposed method on the test set is limited to some extent by the quality of the dataset itself. In the dataset, some dish names do not match the actual images. For example, for stinky tofu and pan-fried tofu, many images show no obvious difference, and some stinky tofu images do not match common knowledge, which may cause the model to learn incorrect features and affect performance. Future work should focus on the dataset itself, improving data labeling and quality, which would further enhance model performance. Second, there are many hyperparameters in the model, such as the learning rate, decay rate of the loss function, and the temperature T for the sharpen operation. This paper experimented with many hyperparameter combinations and achieved good results, but further tuning these hyperparameters could improve model performance. Moreover, Chinese food images contain much noise, such as tableware and background. Future research could investigate image segmentation to remove noise interference and extract features more useful for recognition.

References

[1] Chen, X., Zhu, Y., Zhou, H., et al. (2017). ChineseFoodNet: A large-scale Image Dataset for Chinese Food Recognition. arXiv preprint arXiv:1705.02743. https://arxiv.org/abs/1705.02743.

[2] Nguyen, D. T., Zong, Z., Ogunbona, P. O., et al. (2014). Food image classification using local appearance and global structural information. *Neurocomputing*, *140*, 242–251. https://doi.org/10.1016/j.neucom.2014.03.019.

[3] Jiang, S., Min, W., Liu, L., et al. (2020). Multi-Scale Multi-View Deep Feature Aggregation for Food Recognition. *IEEE Transactions on Image Processing*, *29*, 265–276. https://doi. org/10. 1109/TIP.2019.2932258.

[4] Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/ 1409. 1556.

[5] Szegedy, C., Liu, W., Jia, Y., et al. (2015). Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1–9). IEEE. https:// doi.org/ 10.1109/CVPR.2015.7298594.

[6] He, K., Zhang, X., Ren, S., et al. (2016). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778). IEEE. https://doi.org/10.1109/CVPR.2016.90.

[7] Huang, G., Liu, Z., Van Der Maaten, L., et al. (2016). Densely Connected Convolutional Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2261–2269). IEEE. https://doi.org/10.1109/CVPR.2016.247.

[8] Iandola, F., Han, S., Moskewicz, M. W., et al. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. arXiv preprint arXiv:1602.07360. https://arxiv.org/abs/1602.07360.

[9] Howard, A., Zhu, M., Chen, B., et al. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv preprint arXiv:1704.04861. https:// arxiv.org/ abs/1704.04861.

[10] Howard, A., Sandler, M., Chen, B., et al. (2020). Searching for MobileNetV3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 1314–1324). IEEE. https:// doi. org/ 10.1109/ICCV.2019.01412.

[11] Khan, S., Naseer, M., Hayat, M., et al. (2022). Transformers in Vision: A Survey. *ACM Computing Surveys*, *54*(10s), 1–41. https://doi.org/10.1145/3505244.

[12] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, *30*. https://proceedings. neurips.cc/paper_ files/paper/ 2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract. html.

[13] Berthelot, D., Carlini, N., Goodfellow, I., et al. (2019). MixMatch: A Holistic Approach to Semi-Supervised Learning. arXiv preprint arXiv:1905.00546. https://arxiv.org/ abs/ 1905. 00546.

[14] Zhang, H., Cisse, M., Dauphin, Y. N., et al. (2017). mixup: Beyond Empirical Risk Minimization. arXiv preprint arXiv:1710.09412. https: //arxiv.org/abs/1710.09412.