EASR

Engineering and Applied Science Research

https://www.tci-thaijo.org/index.php/easr/index

Published by the Faculty of Engineering, Khon Kaen University, Thailand

Deep feature extraction technique based on Conv1D and LSTM network for food image recognition

Sirawan Phiphitphatphaisit and Olarik Surinta*

Department of Information Technology, Faculty of Informatics, Mahasarakham University, Maha Sarakham 44150, Thailand

Received 16 December 2020
Revised 19 February 2021
Accepted 8 March 2021

Abstract

There is a global increase in health awareness. The awareness of changing eating habits and choosing foods wisely are key factors that make for a healthy life. In order to design a food image recognition system, many food images were captured from a mobile device but sometimes include non-food objects such as people, cutlery, and even food decoration styles, called noise food images. These issues decreased the performance of the system. Convolutional neural network (CNN) architectures are proposed to address this issue and obtain good performance. In this study, we proposed to use the ResNet50-LSTM network to improve the efficiency of the food image recognition system. The state-of-the-art ResNet architecture was invented to extract the robust features from food images and was employed as the input data for the Conv1D combined with a long short-term memory (LSTM) network called Conv1D-LSTM. Then, the output of the LSTM was assigned to the global average pooling layer before passing to the softmax function to create a probability distribution. While training the CNN model, mixed data augmentation techniques were applied and increased by 0.6%. The results showed that the ResNet50+Conv1D-LSTM network outperformed the previous works on the Food-101 dataset. The best performance of the ResNet50+Conv1D-LSTM network achieved an accuracy of 90.87%.

Keywords: Food image recognition, Deep feature extraction method, Long short-term memory, Convolutional neural network, Spatial temporal features

1. Introduction

Overweight and obesity are the most significant factors for chronic diseases such as diabetes and cardiovascular diseases. The easiest way to avoid chronic diseases is to monitor and control people's dietary behavior. The advancement of artificial intelligence might help people to monitor and estimate daily calorie intake. Hence, food recognition systems are the most straightforward solution. Many systems can recognize several foods based on images. However, when people take a photograph several food characteristics (e.g. the shape and decoration of food, brightness adjustment, and non-food objects, called noise food images) are sent to the system to compute and predict the food type and calorific content. These issues can be a cause of weaknesses of food imaging systems.
Computer vision and machine learning techniques are proposed to address the problems mentioned above. Many researchers employ computer vision techniques to generate hand-crafted visual features and send robust features to the novel machine learning techniques, such as support vector machine (SVM), multilayer perceptron (MLP), random forest, and Naive Bayes techniques [1-3] to classify objects [4, 5].

Furthermore, many studies have extracted the robust features, called the spatial features, using convolution neural network (CNN) architectures. The greatest benefit of this technique is that we can extract robust features with various CNN architectures. The robust features, however, are sent to be classified using traditional machine learning techniques. Additionally, the CNN architecture combined with a long short-term memory (LSTM) network has been applied for classification tasks. Nevertheless, a few researchers have invented CNN architectures and LSTM networks for food image recognition. In this research, we focus on improving the accuracy performance of the food image recognition based on CNN architectures and LSTM networks.

The significant contributions of this research are summarized in the following:

1. We propose the deep learning framework that combines state-of-the-art ResNet50, which is the convolutional neural network (CNN) and long short-term memory (LSTM) network, called ResNet50+Conv1D-LSTM network. This framework can extract robust features that are spatial and temporal features, from the food images. Mixed data augmentation techniques are also involved while training the CNN model. The data augmentation technique insignificantly increases the performance of food image recognition.

2. In these experiments, LSTM and Conv1D-LSTM networks were proposed to create robust temporal features. For the Conv1D network, various layers were combined, including zero padding, batch normalization, Convolution 1D, ReLU, batch normalization, dropout, and average pooling layers. In the training scheme, batch size, which was the number of training examples, were applied as 16, 32, and 64. The LSTM network results showed that a batch size of 32 provided a better result than batch sizes of 16 and 64.

*Corresponding author. Tel.: +66 4375 4359
Email address: olarik.s@msu.ac.th
doi: 10.14456/easr.2021.60

Paper Outline. This paper is organized as follows. Section 2 briefly explains deep learning researches in food image recognition systems and describes the different deep learning techniques. Section 3 describes the proposed approach for the food image recognition system. In Section 4, the experimental settings and the results of the deep learning methods are presented. The conclusion and directions for future work are given in Section 5.

2. Literature review

In this section, we review the research that has applied different techniques to solve image recognition, especially food images.

In previous studies, many researchers have proposed using feature extraction methods based on handcrafted methods to extract features from images. Novel feature extraction methods such as local binary patterns (LBP) [6], the scale-invariant feature transform (SIFT) [7] the histogram of oriented gradients (HOG) [8], the speed-up robust features (SURF) [9] and a bag of visual words (BoVW) [10, 11] methods became popular and were proposed in many applications. Also, they achieved high accuracy performance. Secondly, the robust features extracted from the novel methods, are then given to machine learning algorithms such as support vector machine (SVM) [12], K-nearest neighbor (KNN) [13], and multi-layer perceptron (MLP) for a task of classification.

The food image recognition, Anthimopoulos et al. [4] proposed an automatic food recognition system to recognize 11 different central European foods. In the food recognition system, the features, namely visual words, are computed from the bag-of-features method and the k-means clustering algorithm. Then the linear SVM is used as a classifier. This method obtained a recognition performance of 78%. Furthermore, Martinel et al. [14] introduced an extreme learning committee approach. This approach was divided into three parts; feature extraction methods, extreme learning committee, and supervised classification. First, various feature extraction methods were proposed to extract color, shape, texture, local, and data-driven features. Second, each feature vector was given to the extreme learning machine (ELM). Finally, the output from each ELM was sent to the SVM algorithm for classification. The extreme learning committee outperformed the state-of-the-art methods on four benchmark food image datasets.

Deep learning techniques are becoming increasingly popular in food image recognition. In this section, we describe the research that has applied deep learning to solve the image recognition problem, including 1) deep learning for food image recognition and 2) deep feature extraction methods.

2.1 Deep learning for food images recognition

Convolution Neural Networks (CNNs) have been extensively used in food image recognition research. In 2016, Hassannejad et al. [15] and Liu et al. [16] used Google's image recognition architecture Inception. Hassannejad et al. [15] proposed a network composed of 54 layers with fine-tuned architecture for classifying food images from three benchmark food image datasets: Food-101, UECFOOD100, and UEC-FOOD256. On these datasets, the achieved accuracy was 88.28%, 81.45%, and 76.17%, respectively. Liu et al. [16] invented the DeepFood network that modified the Inception module by introducing a 1×1 convolutional layer to reduce the input dimension to the next layers. It allows a less complicated network. The accuracy achieved was 77.40% with the Food-101 dataset, 76.30%, and 54.70% with UEC-FOOD100, and UEC-FOOD256, respectively. In addition, the Inception architecture, the ResNet architecture is widely popular for food image recognition. Pandey et al. [17] used ResNet, AlexNet, and GoogLeNet to propose an ensemble network architecture. The network consisted of three fine-tuned CNN in the first layer. All of the output was concatenated before being fed into ReLU nonlinear activation and passed to a fully connected layer followed by a softmax layer for image classification. Aguilar, Bolaños, & Radeva [18] proposed the CNN Fusion methodology, which is composed of two main steps. First, training with state-of-the-art CNN models consisting of ResNet and Inception. Second, fusing the CNN outputs using the decision template scheme for classifiers fusion. The two proposed methods achieved accuracies of 72.12% and 86.71% with the Food-101 dataset, respectively. Table 1 summarizes different food classification approaches. The accuracies reported along with the food databases used in the evaluation and the underlying CNN architecture.

Table 1 Performance evaluation of classification results on the food datasets using deep learning techniques.

Datasets	Architectures	Accuracy	References
UEC-FOOD100 [19]	DeepFood	76.30	Liu et al. [16]
	InceptionV3	81.45	Hassannejad et al. [15]
	WISeR	89.58	Martinel et al. [20]
UEC-FOOD256 [21]	DeepFood	54.70	Liu et al [16]
	GoogLeNet	63.16	Bolanos and Radeva [22]
	InceptionV3	76.17	Hassannejad et al. [15]
	WISeR	83.15	Martinel et al. [20]
Food-101 [23]	Inception	77.40	Lie et al. [16]
	GoogLeNet	79.20	Bolanos and Radeva [22]
	InceptionV3	88.28	Hassannejad et al. [15]
	Ensemble Net	72.12	Pandey et al. [17]
	CNNs Fusion	86.71	Aguilar et al. [18]
	ResNet152	64.98	McAllister et al. [2]
	WISeR	90.27	Martinel et al. [20]

2.2 Deep feature extraction methods

Many researchers have focused on extracting features using several CNN architectures, called deep feature extraction [24, 25] that have been applied in many image recognition systems. With the deep feature extraction method, the pre-trained models of the state-of-the-art CNN architectures are employed to train a set of images. Then, the deep features are extracted from the layer before the fully connected layer. After that, we can use the deep features as the input vector to a traditional machine learning algorithm, such as SVM, KNN, and MLP. Indeed, the state-of-the-art CNN architectures, such as VGG, ResNet, and Inception, have been proposed and widely used in the food image recognition system [2, 15].

Table 2 Performance evaluation of classification results on the food datasets using deep feature and machine learning techniques

Datasets	Classes	Deep feature methods	Classifiers	Accuracy	References
PFID	7	AlexNet	SVM-linear	94.01	Farooq et al. [1]
PFID	61	AlexNet	SVM-linear	70.13	Farooq et al. [1]
UNICT-FD889	2	AlexNet	SVM-sigmoid	94.86	Ragusa et al. [3]
Food-5K	2	ResNet152	SVM-RBF	99.4	McAllister et al. [2]
Food-11	11	ResNet152	ANN	91.34	McAllister et al. [2]
RawFooT-DB	46	ResNet152	ANN	99.28	McAllister et al. [2]
Food-101	101	ResNet152	SVM-RBF	64.68	McAllister et al. [2]

To classify the food and non-food images, Ragusa et al. [3] proposed to use three deep feature methods called the Network in Network, the AlexNet, and the VGG-s models to extract features and then use a support vector machine (SVM) as a classifier. The best performance result was the AlexNet model combined with a binary SVM classifier on the Food-5k dataset. For multi-class food images, Farooq & Sazonov [1] proposed the deep feature method called AlexNet to extract features from the PFID food image dataset. This method extracts the feature of 4,096, 4,096, and 1,000 channels from three fully connected (FC) layers; FC6, FC7, and FC8. Also, the linear SVM technique is applied as a classifier. The results showed that the features extracted from FC6 outperformed features from other FC layers. Moreover, McAllister et al. [2] applied ResNet152 and GoogLeNet for deep feature methods performed on five datasets consisting of Food-5k, Food-11, RawFooT-DB, and Food-101 dataset. The deep features were then classified using traditional machine learning comprising SVM, artificial neural networks, Random Forest, and Naive Bayes. The experimental result with these methods had accuracies above 90% on food image datasets, except for the Food-101 dataset that obtained only 64.68% accuracy. A summary of food classification using the deep feature methods is shown in Table 2.

3. Proposed approach for the food image recognition system

This section explains the framework of food image recognition. Two main architectures, convolutional neural network (CNN) and long short-term memory (LSTM) network, are proposed to extract the robust features from the food images. Hence, the robust spatial and temporal features are extracted from state-ofthe-art ResNet architecture and the LSTM network. The temporal features extracted from the LSTM network are transformed into a probability distribution using the softmax function.

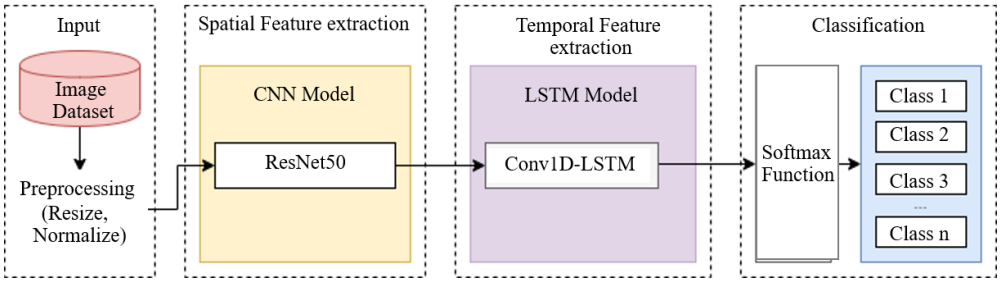

Figure 1 Architecture of our proposed framework for food image classification

According to our framework, as shown in Figure 1, we examine the transfer learning strategy to train the ResNet architecture. Hence, this architecture considers only the color image and the resolution of the images is decided to be 224x224x3 pixels. We also normalize all food images to the values between 0 and 1 by dividing the pixel values with 255, which is the maximum value of the RGB color. Other schemes are described in the section of the spatial feature extraction method using CNN architecture and temporal feature extraction method using LSTM network, as follows.

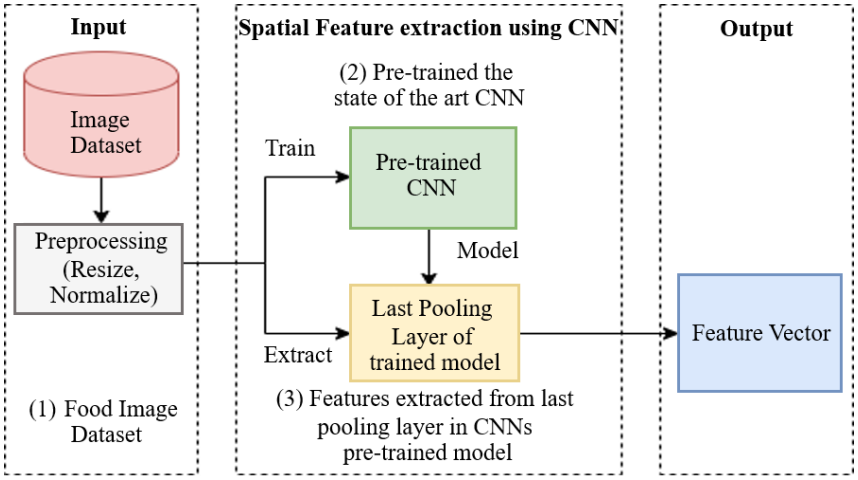

Figure 2 Diagram of the deep feature extraction technique. (1) food images are fed to the pre-processing step to resize and normalize. In the spatial feature extraction process, (2) food images are trained using state-of-the-art CNN architectures to find weights with low validation loss. Then, (3) the spatial features of the food images are extracted according to the best CNN model.

In this section, we propose an effective CNN architecture to extract a robust spatial feature. According to the computation power and time, the transfer learning approach is applied in the training scheme, then the pre-trained models of CNN architectures are trained on the food image and then examined to discover the best robust spatial feature. As a result, the last pooling layer of the CNN model is employed as the spatial feature, as shown in Figure 2. We can also call this method a deep feature extraction technique.

To extract the robust spatial features, in this study, we propose state-of-the-art CNNs, VGG16, VGG19, ResNet50, DenseNet201, MobileNetV1, and MobileNetV2. An overview of each CNN will now be described.

3.1 Spatial feature extraction using convolutional neural network architecture

3.1.1 VGGNet Architecture

Simonyan and Zisserman [26] proposed a network to increase the stack of convolutional networks into 16 and 19 weight layers by using an architecture with a size of 3x3 pixels convolution filters, called VGGNet. With this network, the input images are the color image and are resized to 224x224 pixels resolution. The convolutional layers are downsized from 224x224 pixels to 7x7 pixels. Nevertheless, the number of feature maps is increased from 64 to 512 layers. The rectified linear unit (ReLU) is used as the activation function. Also, spatial pooling is computed by the max-pooling method with the size of a 2x2 pixel window. Three fully connected (FC) layers follow VGGNet. The first two FC layers have 4,096 channels and the last FC layer contains 1,000 channels. The VGGNet is designed as a plain network, but still obtained the best performance on many image classification applications, such as remote sensing classification [27], and plant recognition [28-30].

3.1.2 ResNet Architecture

According to the plain network, the deeper convolutional layers were performed from 34-Layer until 152-layer plain networks [31]. Firstly, the color image is resized to 224x224 pixels resolution and employed as the input of the deeper network. Secondly, the convolutional layers are divided into five convolutional blocks, namely building blocks. Remarkably, the output of each building block is always decreased by half of the input. For example, the output of the first, second, and fifth building blocks are 112x112, 56x56, and 7x7 pixels resolution, respectively. Finally, the average-pooling method is applied to the last building block and followed by the FC layer with 1,000 channels and the softmax function. As a result, the deeper plain network gave a higher error rate on the CIFAR-10 dataset.

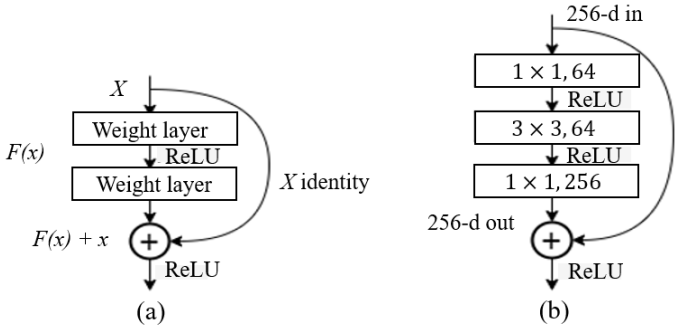

Figure 3 Illustration (a) a building block and the residual function and (b) a sample of bottleneck network for ResNet 50, 101, and 152.

According to the higher error rate, He et al. [31] proposed to add the residual network, which is the shortcut connection, to train the deeper network, called ResNet. Hence, the shortcut connections are computed using the residual function that allows the network to skip two convolutional layers, as shown in Figure 3a). The residual function is calculated by $F(x) = H(x) - x$ when the feature maps of the input and output have identical dimensions. The original function changes to $F(x) + x$. Furthermore, bottleneck architectures are presented when the deeper convolutional layers are implemented as 50, 101, and 152 layers. The bottleneck architectures allow the network to skip three convolutional layers, as shown in Figure 3b). Consequently, ResNet obtained a top-5 error rate of 3.57% on the ImageNet validation set and showed fast computation compared to the plain network. The ResNet also won the ILSVRC-2015 classification task.

3.1.3 DenseNet Architecture

Huang et al. [32] proposed a dense network called DenseNet architecture. The different depth convolutional layers were experimented with consisted of 121, 169, 201, and 264. The result showed that the DenseNet with 264-layer provided the lowest top-1 error rate on the ImageNet validation set and yielded a better error rate than the ResNet architecture. Also, the parameter of the DenseNet is approximately 3-time less than the ResNet. According to the connection of the DenseNet, the network can connect to other layers in a feed-forward method. The number of direct connections can be computed using L(L+1)/2, where L is the number of layers.

To further improve the DenseNet architecture, the convolutional layers are divided into four blocks, namely dense blocks. In each dense block, the bottleneck layers with a size of 1x1 and 3x3 convolution are used to reduce the number of input feature maps. The transition layers are combined with the dense blocks 1-3 to reduce the size of the feature maps to the half size of the convolutional layer in the dense block. The output size of each block is decreased from 112x112 to 7x7 pixels. As for the classification layer, the global average-pooling, FC layer, and softmax are applied. The differences between ResNet and DenseNet architectures are shown in Figure 4.

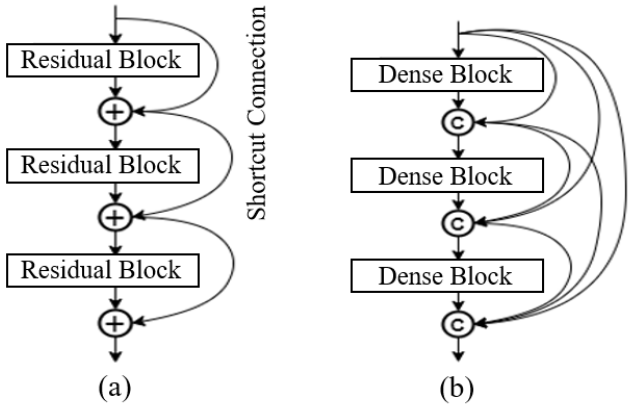

Figure 4 Illustration of the difference of the connections between (a) the ResNet and (b) the DenseNet architectures.

3.1.4 MobileNet Architecture

The lightweight CNN architecture called MobileNet is proposed for mobile and embedded devices [33]. In order to reduce the size of the model, the depthwise separable convolution layer, a core layer of the MobileNet, is designed to factorize the standard convolution into 3x3 depthwise convolutions and then factorize the depthwise convolution layer into 1x1, called pointwise convolution. Due to MobileNet architecture, the depthwise and pointwise convolution layers are always followed by batch normalization (batchnorm) and ReLu, as shown in Figure 5a).

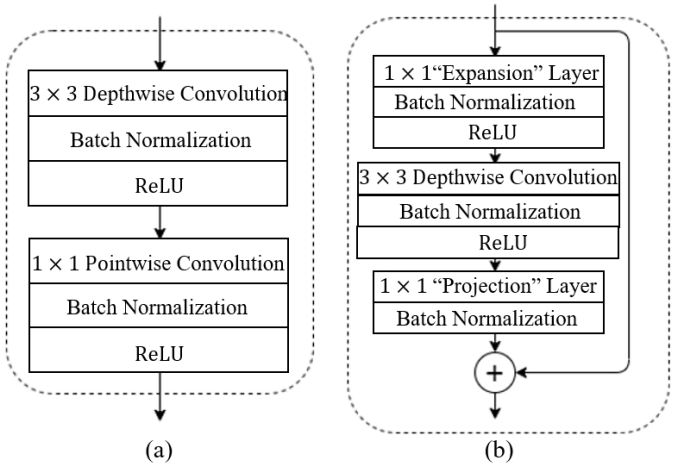

Figure 5 Network architectures of MobileNet. Examples of (a) the depthwise separable convolution and (b) inverted residual and linear bottleneck.

Furthermore, Sandler et al. [34] proposed MobileNetV2 architecture. The new mobile architecture, called inverted residuals and linear bottlenecks, is combined with the linear bottleneck layer and inverted residual network. The inverted residuals and linear bottlenecks block consist of three layers. First, 1x1 convolution combined with batchnorm and ReLU. Second, depthwise convolution combined with batchnorm and ReLU. Third, 1x1 convolution combined with batchnorm and without non-linearity, as shown in Figure 5b). In MobileNetV2 architecture, the number of operations is decreased, so that is was of small size and low memory usage. A summary of the state-of-the-art CNN architectures is presented in Table 3.

Table 3 Summary of the state-of-the-art CNN architectures.

CNN architectures	Parameters					
	No. of Conv layer	**Filter Size**	**Stride**	**Pooling**	**No. of FC layers**	**No. of parameters**
VGG16	13	3	1	Max	3	138M
VGG19	16	3	1	Max,	3	143M
ResNet50	49	1, 3, 7	1, 2	Max, Average	1	25.6M
DenseNet201	200	1, 3, 7	1, 2	Max, Average	1	20.2M
MobileNetV1	13	1, 3	1, 2	Average	2	4.2M
MobileNetV2	13	3	1, 2	Average	1	3.2M

3.2 Temporal feature extraction

In this section, we propose two deep learning networks to extract temporal features, called long short-term memory and Conv1D-LSTM networks. The detail of deep learning networks is will now be described.

3.2.1 Long short-term memory

Hochreiter & Schmidhuber [35] invented a novel gradient-based method and developed the network based on a recurrent neural network (RNN) called a long short-term memory (LSTM) network, as shown in Figure 6. It proposed to address the computational complexity, error flow, constraints of the feedforward neural network, and sequence problems of time series data [36, 37]. The LSTM network comprised special units that connect to other units and are designed to cope with the sequence of data; video and speech data, called memory blocks. Each memory block contained the various functions consisting of the forget gate, input gate, update cell state, and the output gate.

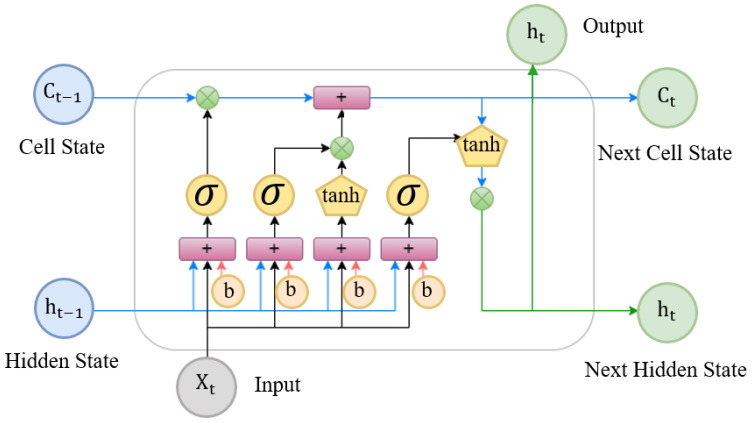

Figure 6 The architecture of the long short-term memory network [35].

The memory block presented in Figure 6 is calculated as follows;

$$f_t = \sigma\left(w_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\check{C}_t = tanh(w_c \cdot [h_{t-1}, x_t] + b_c)$$
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \check{C}_t$$
$$O_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_f \cdot tanh(C_t)$$

where f_t is forget gate's activation vector, i_t is input/update gate's activation vector, $\check{C}_t$ is cell input activation vector, C_t is current cell memory, O_t is output gate's activation vector, h_t is current cell output, , b and W denote the bias vector and weight matrices for the input gate (i), output gate (o), forget gate (f), and memory cell (c), h_{t-1} is previous cell output, C_{t-1} is previous cell memory, σ is sigmoid function, and ' · ' is the Hadamard product [35].

3.2.2 Conv1D-LSTM

In this study, we propose the Conv1D-LSTM framework to extract temporal feature from the spatial features, as shown in Figure 7. In the Conv1D block, the batch normalization layer was added so as to normalize the input data and speed up the process of learning. The dropout layer was implemented to prevent over-fitting, then some units were ignored during learning. After that, the average pooling layer which selected the average component from the sub-region of the feature map, was considered as the feature vector. The feature vector was sent to the LSTM Cells to learn and generate the temporal feature. Consequently, we again decreased the size of the feature using global average pooling layer (GAP) before giving the feature to the softmax function.

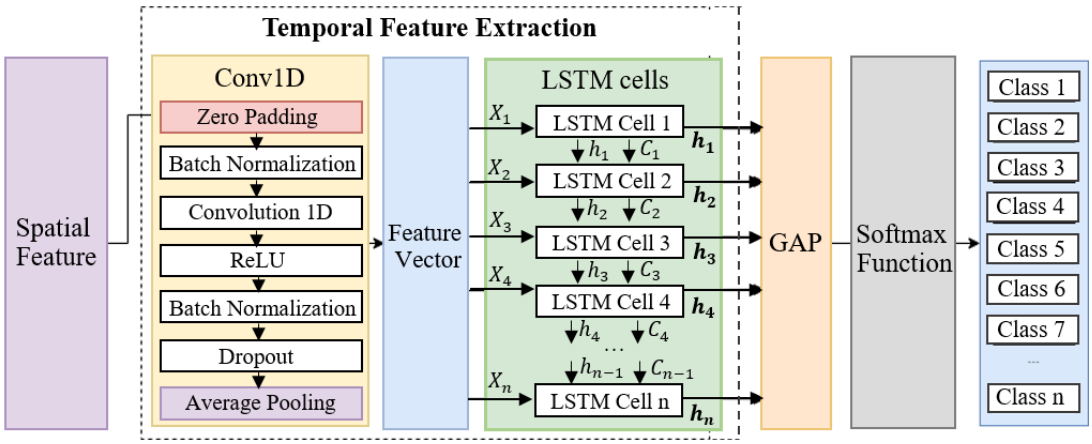

Figure 7 Illustration of extract temporal features using the Conv1D-LSTM network.

4. Experimental setup and results

4.1 Food image dataset

In this research, we focused on experimenting with the benchmark food image dataset, namely the Food101 dataset [23]. The training set contained the wrong labels and some noise images, such as food images taken from different camera angles that made other objects such as people, tables, and bottles, appear in the image. This dataset contains 101,000 real-world color images of 101 food categories. It consists of 75,750 training images and 25,250 test images. The sample images of the Food-101 dataset are shown in Figure 8. The challenge of this dataset is that the training set contained some noise images, such as food images taken from different camera angles that made other objects such as people, tables, and bottles, appear in the image, as shown in Figure 9(a) and similarities of shape, color, and decoration between two categories (chocolate cake and chocolate mousse), as shown in Figure 9(b). The researchers assume that computer vision can handle noise images and wrong labels.

Figure 8 Sample images of the Food-101 dataset

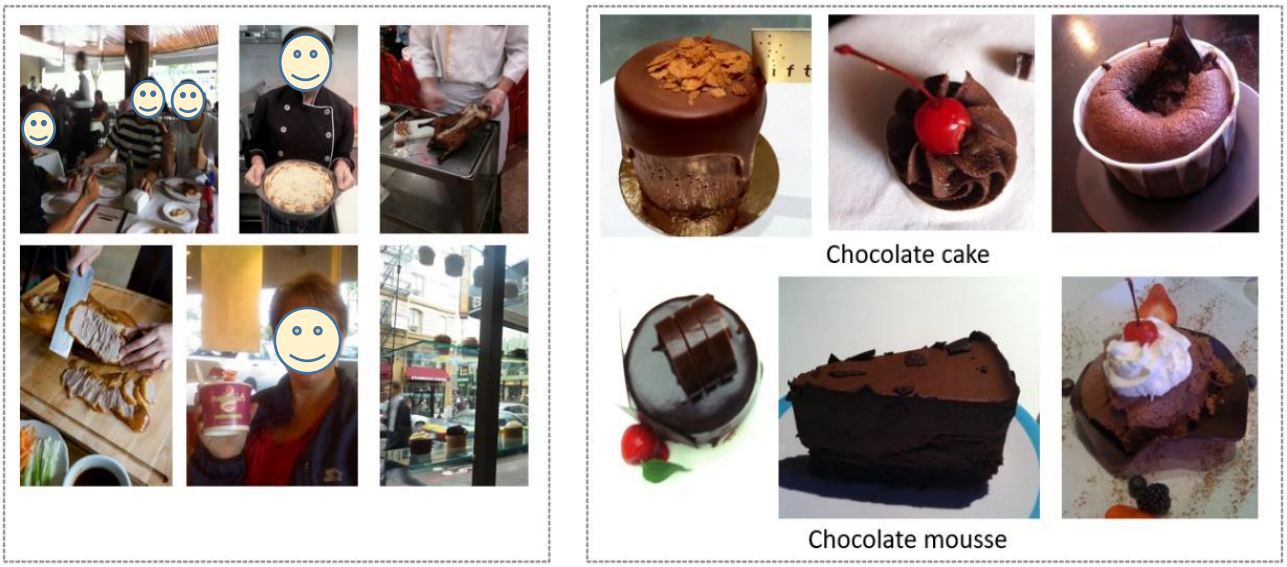

Figure 9 Some examples of the Food-101 dataset that containing (a) other objects (e.g., people, cake shelves, tables, and glasses of beer) and (b) similarities of chocolate cake and mousse.

4.2 Experimental setup

We implement the proposed framework with the TensorFlow platform. All experiments were performed on a Linux operating system with Intel(R) Core(TM) i7-4790 Processor 3.6GHz, 16GB DDR4 RAM. As explained in Section 3, we first used pre-trained models of six CNN architectures; VGG16, VGG19, ResNet50, DenseNet201, MobileNetV1, and MobileNetV2, to train and extract the spatial feature from food images. All CNNs were trained using the stochastic gradient descent (SGD) optimizer, rectified linear unit (ReLU) for activation function, and learning rate between 0.01 to 0.0001. Second, the spatial features were then sent to Conv1D-LSTM and LSTM networks to extract temporal features. In the LSTM network, the fraction of the units was employed to drop the linear transformation of the inputs. The initial weights were randomly selected by using a Gaussian distribution where the mean is zero.

We decided to train only 100 epochs to avoid overfitting when training the model. Figure 10 shows loss values while training the Conv1D-LSTM and LSTM model. According to loss values, better loss values were obtained after epoch 50 when they became stable values until epoch 100.

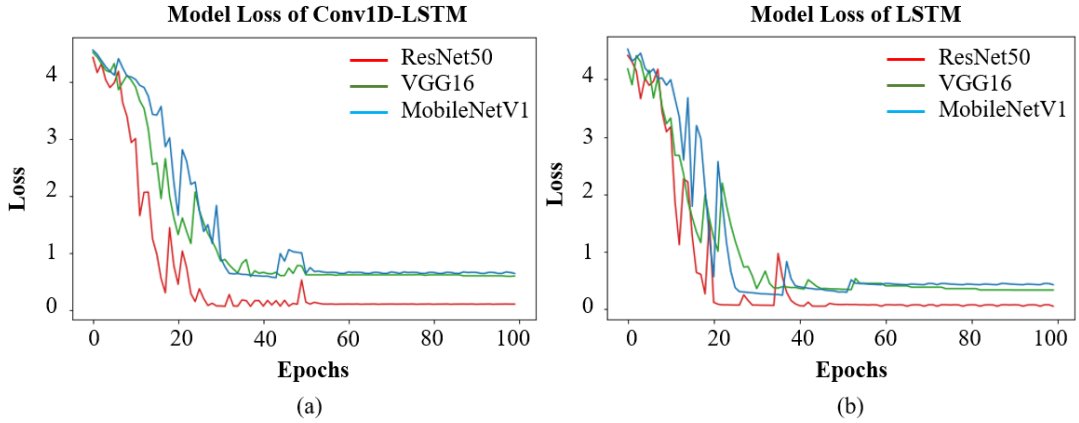

Figure 10 Illustration of loss values of (a) Conv1D-LSTM and (b) LSTM networks when using ResNet50, VGG16, and MobileNetV1 as a deep feature method.

4.3 Evaluation metrics

The evaluation metrics used for food image recognition were accuracy and F1-score. We used the accuracy score to evaluate the performance of the deep learning models on the test set and used the F1-score to examine the individual accuracy of each class. The accuracy and the F1-score were computed by Equations 2 and 3.

$$accuracy = \frac{TP_k + TN_k}{TP_k + TN_k + FP_k + FN_k} \tag{2}$$

$$F1 - score = 2 \times \frac{\left(\frac{TP_k}{TP_k + FN_k} \times \frac{TP_k}{TP_k + FP_k}\right)}{\left(\frac{TP_k}{TP_k + FN_k} + \frac{TP_k}{TP_k + FP_k}\right)} \tag{3}$$

Where TP_k called true positives, is the number of correctly classified images from class k.
FP_k called false positives, is the number of misclassified images from class k.
TN_k called true negatives, is the number of correctly classified image that does not belong to class k.
FN_k called false negatives is the number of misclassified images belong to class k.

4.4 Experiments with deep learning methods

In the experiments with deep learning methods, we first trained the Food-101 dataset using a pre-trained model of six state-of-the-art CNNs; VGG16, VGG19, MobileNetV1, MobileNetV2, ResNet50, and DenseNet201. Second, we proposed the deep feature method to extract the spatial feature from the last pooling layer of each CNN. The deep feature method extracted a high dimension of the spatial feature. The number of spatial features is reported in Table 4. It can be seen that ResNet50 provided 99,176 features. On the other hand, VGG16 produced only 25,088 features. Finally, we trained the high dimension of the spatial features using Conv1D-LSTM and LSTM networks.

Table 4 Illustration of the number of spatial features extract from different CNN architectures and size of each model

Deep feature methods	No. of parameters	No. of features
VGG16	14.7M	25,088
VGG19	20M	25,088
ResNet50	23.5M	99,176
DenseNet201	18.3M	94,080
MobileNetV1	3.2M	50,176
MobileNetV2	2.2M	62,720

Table 5 Evaluation of the classification results for the Food-101 dataset using different deep learning consisting of CNN, LSTM, and Conv1D-LSTM. The first column shows the deep feature methods that used to extract spatial features.

Model	CNN	LSTM		Conv1D-LSTM		
		No pooling layer	Global average pooling	No pooling layer	Average pooling	Max pooling
VGG16	67.40	78.55	80.44	75.94	85.91	84.61
VGG19	65.54	77.15	79.94	75.02	85.66	84.52
MobileNetV1	50.60	58.59	60.32	64.80	65.88	65.75
MobileNetV2	37.20	50.33	51.94	55.14	56.73	56.71
DenseNet201	39.29	38.08	38.98	42.25	42.87	38.11
ResNet50	79.86	88.90	88.92	86.83	89.82	89.01

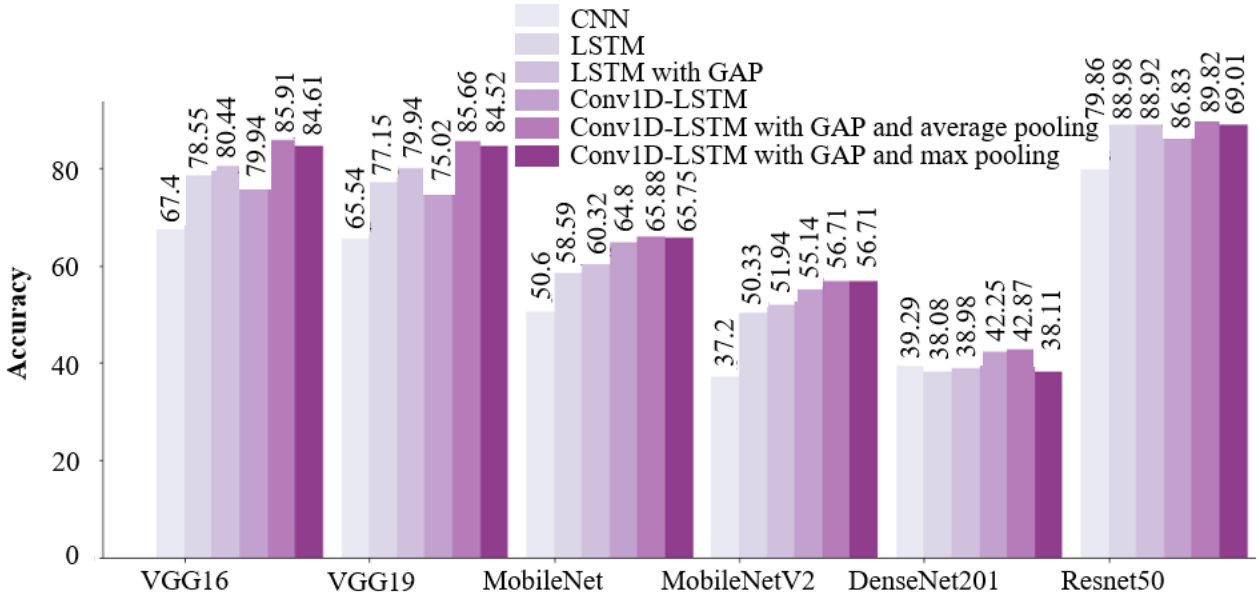

Figure 11 Performance evaluation of three classifiers consisted of CNN, Conv1D-LSTM, and LSTM architectures that extract features based on six different deep CNN architectures on the Food-101 dataset.

Table 5 and Figure 11 present the accuracy results on the test set of the Food-101 dataset for CNN, Conv1D-LSTM, and LSTM networks. The results show that the Conv1D-LSTM achieved the best performance with 89.82% accuracy when using a batch size of 32 and extracting features with ResNet50. As a result, the Conv1D-LSTM network with the batch size of 32 always showed better accuracy than other batch sizes. According to our experiments, however, the CNN architectures presented worse performance compared to the Conv1D-LSTM and LSTM networks. In terms of the deep feature methods, the ResNet50 outperforms all CNN architectures when training with the CNN, Conv1D-LSTM, and LSTM networks. The result of the CNN architectures shows that the ResNet50 provided 42.66% accuracy higher than the MobileNetV2. We concluded that the ResNet50 extracted the spatial feature with a high dimension and still provided higher accuracy when training with Conv1D-LSTM and LSTM networks. Hence, the ResNet50 combined with the Conv1D-LSTM, namely ResNet50+Conv1D-LSTM, performed best on the Food-101 dataset.

The experimental results show that the Conv1D-LSTM outperformed LSTM because we combined necessary layers toward the Conv1D network, such as batch normalization, ReLU activation function, and dropout. These layers produced the Conv1D network to normalize the inputs to each feature map and cope with the linear activation function. For Conv1D, we experimented with pooling layers; global average pooling and global max pooling to decrease the size of the feature vector before giving it to classified with the softmax function. The success of the pooling layer is no parameter to optimize and robust to perform the spatial feature.

To study the effect of the data augmentation techniques, we applied six data augmentation techniques; rotation, width shift, height shift, horizontal flip, shear, and zoom while training the CNN architecture because Phiphiphatphaisit & Surinta [38] reported that data augmentation techniques could increase the accuracy of CNN, especially for food image recognition. In this experiment, ResNet50+Conv1D-LSTM using the batch size of 32 was considered.

Table 6 The classification results for the Food-101 dataset using features that extracting from the ResNet50 architecture and data augmentation techniques.

Data augmentation	LSTM	Conv1D-LSTM
No	88.92	89.82
Yes	89.49	90.87

Table 6 showed that LSTM and Conv1D-LSTM perform better when data augmentation techniques were applied. The accuracy of the Conv1D-LSTM with the data augmentation technique was slightly increasing compared with the LSTM with the data augmentation technique. As a result, the ResNet50+Conv1D-LSTM network with the data augmentation technique provided an accuracy of 90.87% on the Food-101 dataset. The data augmentation can generate more food images while training, and then it increases the robustness of the model without decreasing the effectiveness.

The F1-score value of the ResNet50+Conv1D-LSTM network was computed according to Equation (3) and is illustrated in Figure 12. We found that only two categories, chocolate mousse and Filet mignon (see red bar) provided an F1-score of less than 80%. The F1-score also reported that 42 categories (see green bar) obtained a score above 90%. However, when we examined the ResNet50+Conv1D-LSTM network with non-food elements, called noise images, our proposed network could not classify these noise images correctly. Some noise images are shown in Figure 9a and the misclassified results of the noise images are shown in Figure 13. Also, misclassification of similar categories such as chocolate cake and chocolate mousse were found, as shown in Figure 14.

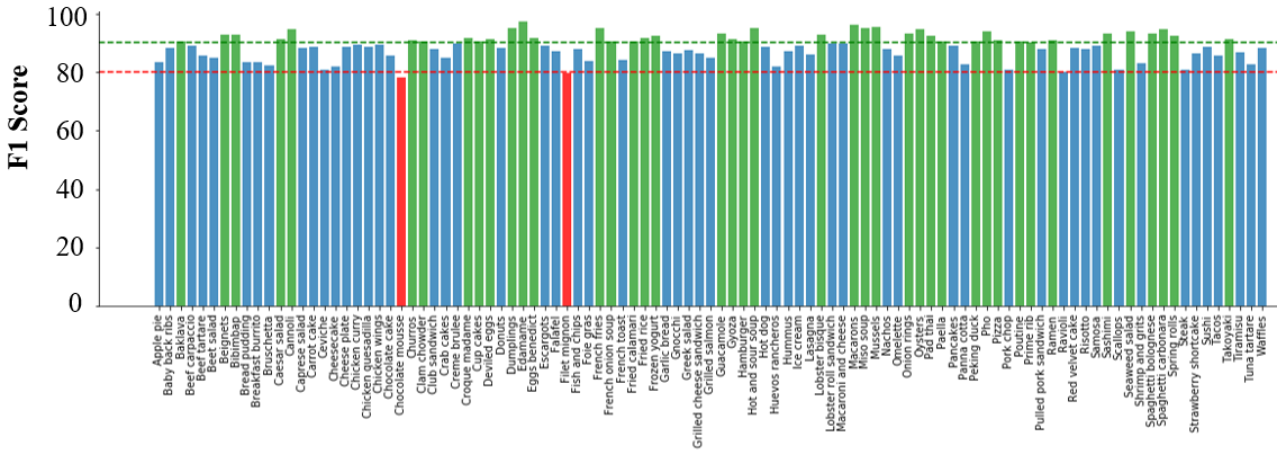

Figure 12 The result of the F1-score on the Food-101 dataset using the ResNet50 and LSTM architectures.

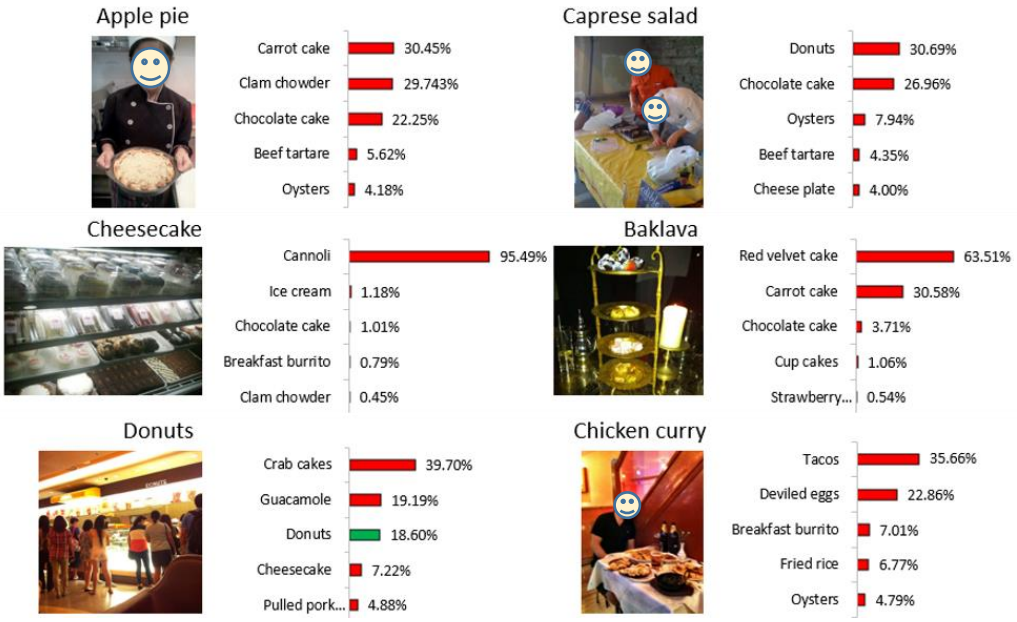

Figure 13 Examples of misclassified results according to the noise images.

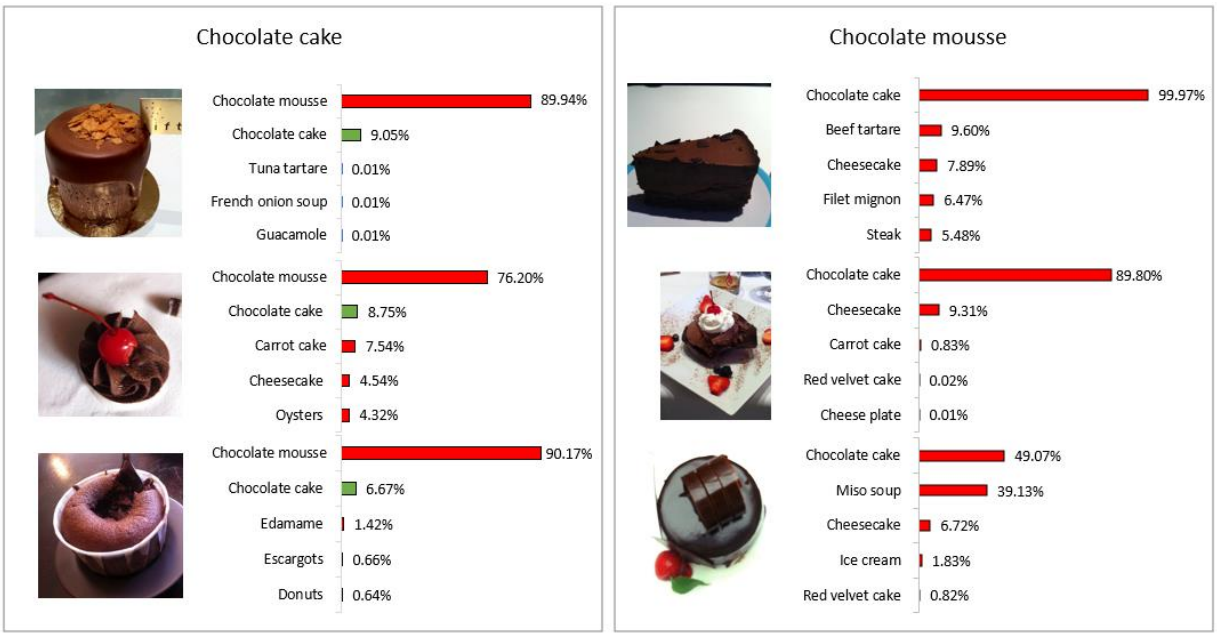

Figure 14 An example of the similarity categories between chocolate cake and chocolate mousse contains in the Food-101 dataset.

4.5 Comparison between ResNet50+Conv1D-LSTM network and previous methods

We made extensive comparisons between our ResNet50+Conv1D-LSTM network and existing state-of-the-art CNN architectures. The experimental results showed that our network performed better than all CNN architectures. The accuracy of 90.87% was obtained from the ResNet50+Conv1D-LSTM, while, the performance of the state-of-the-art WISeR architecture was 90.27% accuracy. The comparative results between the existing CNN architectures and our proposed architecture on the Food-101 dataset are shown in Table 7.

Table 7 Recognition performance on the Food-101 dataset when compared with different deep learning techniques.

Architectures	No. of training images per class	Accuracy	References
ResNet152	750	64.98	McAllister et al. [2]
EnsembleNet	750	72.12	Pandey et al. [17]
Modified MobileNetV1	400	72.59	Phiphiphatphaisit & Surinta [38]
DeepFood	750	77.40	Liu et al. [16]
GoogLeNet	750	79.20	Bolanos & Radeva [22]
CNNs Fusion	750	86.71	Aguilar et al. [18]
InceptionV3	750	88.28	Hassannejad et al. [15]
WISeR	750	90.27	Martinel et al. [20]
ResNet50+Conv1D-LSTM	750	90.87	Our proposed

From the experimental results shown in Table 7, it can be seen that the Conv1D-LSTM yielded better performance than other techniques. Our Conv1D network included many layers consists of batch normalization layer, ReLU activation function, and dropout layer. In our Conv1D, we used the batch normalization layer to normalize the input data to each feature map and this layer works better with the ReLU activation function. The dropout layer was attached to the Conv1D network to prevent the over-fitting, then it allows the network to ignored some units during training.

5. Conclusions

This study proposed the ResNet50+Conv1D-LSTM network for accurate food image recognition. First, our network took advantage of extracting the robust spatial feature using a state-of-the-art convolutional neural network (CNN), called ResNet50 architecture. Second, we used the robust feature as input data for the Conv1D combined with the long short-term memory (LSTM) network, namely Conv1D-LSTM. The primary function of the Conv1D-LSTM network was to extract a temporal feature. Finally, the softmax function was employed to transforms the output of the Conv1D-LSTM into a probability distribution.

In the experiments, we evaluated six CNNs; VGG16, VGG19, ResNet50, DenseNet201, MobileNetV1, and MobileNetV2 to extract the feature, then classify with Conv1D-LSTM and LSTM networks on the Food101 dataset. The results showed that the ResNet50 combined with the Conv1D-LSTM network, called ResNet+Conv1D-LSTM network, provided the best performance (see Table 5). Additionally, we experimented with mixed data augmentation techniques; rotation, width shift, height shift, horizontal flip, shear, and zoom. The result of the data augmentation also insignificantly increased accuracy by 0.27%. Our experiments presented better results than previous work (see Table 7). The best result of the ResNet+Conv1D-LSTM obtained 90.87% on the Food-101 dataset.

In future work, we will experiment on increasing the performance of the food image recognition. We will consider other novel data augmentation techniques, which could be more efficient in the noise food images. Also, the ensemble and parallel networks will be involved in future work.

6. References

[1] Farooq M, Sazonov E. Feature extraction using deep learning for food type recognition. International conference on bioinformatics and biomedical engineering; 2017 Apr 26-28; Granada, Spain. Berlin: Springer; 2017. p. 464-72.
[2] McAllister P, Zheng H, Bond R, Moorhead A. Combining deep residual neural network features with supervised machine learning algorithms to classify diverse food image datasets. Comput Biol Med. 2018;95:217-33.
[3] Ragusa F, Tomaselli V, Furnari A, Battiato S, Farinella GM. Food vs Non-Food classification. Proceedings of the 2nd International workshop on multimedia assisted dietary management; 2016 Oct 16; Amsterdam, Netherlands. New York: ACM Press; 2016. p. 77-81.
[4] Anthimopoulos MM, Gianola L, Scarnato L, Diem P, Mougiakakou SG. A food recognition system for diabetic patients based on an optimized bag-of-features model. IEEE J Biomed Health Inform. 2014;18(4):1261-71.
[5] Martinel N, Piciarelli C, Micheloni C. A supervised extreme learning committee for food recognition. Comput Vis Image Understand. 2016;148:67-86.
[6] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. Proceedings of 12th international conference on pattern recognition; 1994 Oct 9-13; Jerusalem, Israel. New York: IEEE; 2002. p. 582-5.
[7] Lowe DG. Distinctive image features from scale-invariant key points. Int J Comput Vis. 2004;60:91-110.
[8] Dalal N, Triggs B. Histograms of oriented gradients for human detection. 2005 IEEE Computer society conference on computer vision and pattern recognition (CVPR'05); 2005 Jun 20-25; San Diego, USA. New York: IEEE; 2005. p. 886-93.

[9] Bay H, Ess A, Tuytelaars T, Van Gool L. Speeded-up robust features (SURF). Comput Vis Image Understand. 2008;110(3): 346-59.
[10] Coates A, Carpenter B, Case C, Satheesh S, Suresh B, Wang T, et al. Text detection and character recognition in scene images with unsupervised feature learning. 2011 International conference on document analysis and recognition; 2011 Sep 18-21; Beijing, China. New York: IEEE; 2011. p. 440-5.
[11] Csurka G. Visual categorization with bags of keypoints. Workshop on statistical learning in computer vision, ECCV; 2004. p. 1-22.
[12] Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20:273-97.
[13] Altman NS. An introduction to kernel and nearest neighbor nonparametric regression. Am Stat. 1992;46(3):175-85.
[14] Martinel N, Piciarelli C, Micheloni C. A supervised extreme learning committee for food recognition. Comput Vis Image Understand. 2016;148:67-86.
[15] Hassannejad H, Matrella G, Ciampolini P, De Munari I, Mordonini M, Cagnoni S. Food image recognition using very deep convolutional networks. Proceedings of the 2nd International workshop on multimedia assisted dietary management; 2016 Oct 16; Amsterdam, Netherlands. New York: ACM Press; 2016. p. 41-9.
[16] Liu C, Cao Y, Luo Y, Chen G, Vokkarane V, Ma Y. Deep food: deep learning-based food image recognition for computer-aided dietary assessment. Lect Notes Comput Sci. 2016;9677:37-48.
[17] Pandey P, Deepthi A, Mandal B, Puhan NB. Food net: recognizing foods using ensemble of deep networks. IEEE Signal Process Lett. 2017;24(12):1758-62.
[18] Aguilar E, Bolanos M, Radeva P. Food recognition using fusion of classifiers based on CNNs. International conference on image analysis and processing (ICIAR); 2017 Sep 11-15; Catania, Italy. Berlin: Springer; 2017. p. 213-24.
[19] Matsuda Y, Yanai K. Multiple-food recognition considering co-occurrence employing manifold ranking. The 21st International conference on pattern recognition (ICPR); 2012 Nov 11-15; Tsukuba, Japan. New York: IEEE; 2012. p. 2017-20.
[20] Martinel N, Foresti GL, Micheloni C. Wide-slice residual networks for food recognition. 2018 IEEE Winter conference on applications of computer vision (WACV); 2018 Mar 12-15; Lake Tahoe, USA. New York: IEEE; 2018. p. 567-76.
[21] Kawano Y, Yanai K. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. Computer vision - ECCV 2014 workshops; 2014 Sep 6-7, Sep 12; Zurich, Switzerland. Berlin: Springer; 2015. p. 3-17.
[22] Bolanos M, Radeva P. Simultaneous food localization and recognition. 2016 23rd International conference on pattern recognition (ICPR); 2016 Dec 4-8; Cancun, Mexico. New York: IEEE; 2016. p. 3140-5.
[23] Bossard L, Guillaumin M, Van Gool L. Food-101-mining discriminative components with random forests. European Conference on Computer Vision (ECCV); 2014 Sep 6-12; Zurich, Switzerland. Berlin: Springer; 2014. p. 446-61.
[24] Chen Y, Jiang H, Li C, Jia X, Ghamisi P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. IEEE Trans Geosci Rem Sens. 2016;54:6232-51.
[25] Paul R, Hawkins SH, Balagurunathan Y, Schabath M, Gillies R, Hall L, et al. Deep feature transfer learning in combination with traditional features predicts survival among patients with lung adenocarcinoma. Tomo. 2016;2:388-95.
[26] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. The 3rd International Conference on Learning Representations (ICLR); 2015 May 7-9; San Diego, USA. p. 1-14.
[27] Liu X, Chi M, Zhang Y, Qin Y. Classifying high resolution remote sensing images by fine-tuned VGG deep networks. IEEE International geoscience and remote sensing symposium; 2018 Jul 22-27; Valencia, Spain. New York: IEEE; 2018. p. 7137-40.
[28] Abas MAH, Ismail N, Yassin A, Taib M. VGG16 for plant image classification with transfer learning and data augmentation. Int J Eng Tech. 2018;7:90-4.
[29] Habiba SU, Islam MF, Ahsan SMM. Bangladeshi plant recognition using deep learning based leaf classification. 2019 International conference on computer, communication, chemical, materials and electronic Engineering (IC4ME2); 2019 Jul 11-12; Rajshahi, Bangladesh. New York: IEEE; 2019. p. 1-4.
[30] Pearline SA, Vajravelu SK, Harini S. A study on plant recognition using conventional image processing and deep learning approaches. J Intell Fuzzy Syst. 2019;36:1997-2004.
[31] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. 2016 IEEE conference on computer vision and pattern recognition; 2016 Jun 27-30; Las Vegas, USA. New York: IEEE; 2016. p. 770-8.
[32] Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely connected convolutional networks. IEEE Conf Comput Vis Pattern Recogn. 2017; 2261-9.
[33] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: efficient convolutional neural networks for mobile vision applications. Comput Vis Pattern Recogn. 2017;1:1-9.
[34] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: inverted residuals and linear bottlenecks. IEEE Conf Comput Vis Pattern Recogn. 2018;45:10-20.
[35] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735-80.
[36] Jain S, Gupta R, Moghe AA. Stock price prediction on daily stock data using deep neural networks. 2018 International conference on advanced computation and telecommunication (ICACAT); 2018 Dec 28-29; Bhopal, India. New York: IEEE; 2018. p. 1-13.
[37] Yan J, Qi Y, Rao Q. Detecting malware with an ensemble method based on deep neural network. Secur Comm Network. 2018;2018(1):1-16.
[38] Phiphiphatphaisit S, Surinta O. Food image classification with improved MobileNet architecture and data augmentation. The 3rd international conference on information science and systems (ICISS); 2020 Mar 19-22; Cambridge, UK. New York: ACM Press; 2020. p. 51-6.