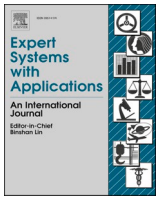
Expert Systems With Applications 255 (2024) 124834

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Multi-layer adaptive spatial-temporal feature fusion network for efficient food image recognition

Sirawan Phiphitphatphaisit [1], Olarik Surinta [2,*]

Multi-agent Intelligent Simulation Laboratory (MISL) Research Unit, Department of Information Technology, Faculty of Informatics, Mahasarakham University, Mahasarakham 44150, Thailand

ARTICLE INFO

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

ABSTRACT

Numerous deep learning methods have been developed to tackle the challenges of recognizing food images, including convolutional neural networks, deep feature extraction, and deep feature fusion methods. This research proposes a new architecture called ASTFF-Net that uses deep feature fusion to tackle various challenges in food recognition, including similarity patterns between two categories, multi-object problems, light conditions, camera position, noise objects, and blurred images. ASTFF-Net is a robust and adaptive spatial–temporal fusion network designed to address these challenges effectively. The ASTFF-Net architecture consisted of three networks. In the spatial feature extraction network, the ResNet50 architecture was used to extract robust spatial features, and the reduction operation was utilized to minimize parameter size. Subsequently, the spatial features were passed through a 1D convolution (Conv1D) to fit the features into the recurrent neural networks. In the temporal feature extraction network, the spatial features were given to the long short-term memory, allowing the network to learn from various long sequence patterns. In the adaptive feature fusion network, the robust spatial and temporal features were fused and assigned to the Conv1D, followed by the softmax function. The ASTFF-Net architecture is also intended to decrease the number of network parameters and prevent overfitting problems. Experimental results on four benchmark food image datasets: Food11, UEC Food-100, UEC Food-256, and ETH Food-101, demonstrate that the proposed ASTFF-Nets, particularly ASTFF-NetB3, were more competitive compared with other existing methods.

1. Introduction

Nowadays, people care about their health and ensure that they live a fit and good life. Many food image recognition applications, such as dietary, personal food logging, nutrition assessment, and social media applications (Liu et al., 2016; Sahoo et al., 2019; Dong, Sun, & Zhang, 2019; Nordin, Xin, & Aziz, 2019; Jiang et al., 2020), were invented to satisfy users' requirements. In order to use the program to its full potential, many applications were then built as mobile applications on smartphones. They allowed people who use a smartphone to take food photos and measure nutrition themselves.

To make the food image recognition applications achieve more accurate results in classification, the artificial intelligence algorithms should deal with uncontrolled photos taken by the users with variations such as brightness, orientation, noise, and other objects in the food images. Fig. 1(a) shows some different orientations of spaghetti. The Peking duck, as shown in Fig. 1(b), is decorated in different styles. Furthermore, Fig. 1(c) shows other objects in the food images, such as glasses, plates, forks, spoons, and knives. Many techniques have been proposed to address these challenges.

Many convolutional neural network (CNN) architectures are currently proposed for food image recognition systems to facilitate effective to analysis and classification of real-world food images. CNNs have also shown state-of-the-art performance on food image recognition. The fine-tuned models of AlexNet and InceptionV3 architectures were used to recognize the real-world food images on the benchmark food image datasets: ETH Food-101, UEC Food-100, and UEC Food-256 (Yanai & Kawano, 2015; Hassannejad et al., 2016). In their experiments, Yanai & Kawano (2015) obtained a recognition accuracy of 78.77 % and 65.57 % on UEC Food-100 and Food-256, respectively. In comparison,

* Corresponding author.
E-mail addresses: sirawan.ch@rmuti.ac.th (S. Phiphitphatphaisit), olarik.s@msu.ac.th (O. Surinta).
[1] https://orcid.org/0009-0003-0188-6528.
[2] https://orcid.org/0000-0002-0644-1435.

https://doi.org/10.1016/j.eswa.2024.124834
Received 6 November 2023; Received in revised form 8 May 2024; Accepted 18 July 2024
Available online 24 July 2024

Hassannejad et al. (2016) achieved an accuracy of 88.28 % on ETH Food-101, 81.45 % on UEC Food-100, and 76.17 % on UEC Food-256.

The concept of the ensemble CNNs network, called Ensemble Net, was proposed by Pandey et al. (2017). In their ensemble net, the input images were first changed to HSV color space and then histogram equalization was applied to only the brightness channel. Second, the food images were sent to fine-tuned CNNs consisting of AlexNet, GoogLeNet, and ResNet. Third, the feature maps that had been extracted from three CNNs were concatenated and sent to the fully connected layers. Finally, their proposed network was classified using the softmax function. Ensemble net performed with a recognition accuracy of 72.12 % on the ETH food-101 and 73.5 % on the Indian food database.

The deep feature extraction technique became the popular method that extracted the robust deep features based on the convolutional neural networks (CNNs). The CNN architecture emphasizes that it computes the weighted parameters from the input images and then creates unique spatial features. Şengür et al. (2019) extracted deep features using two CNN architectures: VGG16 and AlexNet. The deep features were then concatenated and sent to classify using the support vector machine (SVM) technique. Phiphitphatphaisit & Surinta (2021) extracted both spatial and temporal features. First, the spatial features were extracted using ResNet50 and spatial features were subsequently transferred to the Conv1D-LSTM network to extract the temporal features. Finally, the deep features were classified using the softmax function.

In recent research, convolutional neural network (CNN) architectures have commonly been used for deep feature extraction, known as spatial features. In our proposed network, initially the spatial feature extraction network was mainly used for extracting the deep features using the CNN architectures only for extracting the spatial features. The convolution operation allows for capturing pixel information in an image in relation to its neighboring pixels (Kunhoth et al., 2023). Hence, the correlation between the pixels is computed and extracted. It ensures that every region of the input image is processed. The convolution operation not only extracts the spatial features but also reduces the size of the input matrix (Rodriguez-Martinez et al., 2024). In our experiment, we used ResNet50 as the backbone network. Furthermore, we transfer knowledge and fine-tune the ResNet50 model to reduce training time and improve recognition performance.

Second, the spatial features were transferred to the temporal feature extraction network according to the extraction of the temporal features. The spatial features had a size of $M \times N \times L$, where $M \times N$ represents the size of feature maps and L represents the number of feature maps. Further, the LSTM network is employed in this process. The most significant advantage of the LSTM is that it has feedback connections and enables the LSTM to learn the long-term sequence (van Houdt, Mosquera, & Nápoles, 2020; Pereira-Ferrero, Valem, & Pedronette, 2022; Prabhakar & Lee, 2022). Assume L is an enormous number, called a long-term sequence. After extraction, it guarantees that all patterns are preserved and never lost while training the model.

In the third step of the adaptive feature fusion network, third, the deep features between the spatial features and temporal features (after applying the Conv1D) are concatenated and then sent to the global average pooling (GAP) layer, followed by the batch normalization (BN) layer. The GAP operation is recommended to replace fully connected layers in the CNN architectures (Xia, Huang, & Wang, 2020; Wang et al., 2021). The purpose of the adaptive feature fusion network is to combine the spatial and temporal information provided by the spatial feature extraction network and temporal feature extraction network, respectively, in order to represent real-world food image characteristics and improve the efficiency of a given model.

Recognizing food is more complex than recognizing general objects, as food dishes have sets of spatial structures that distinguish them from one another (Feng et al., 2023). Therefore, learning from food images presents several challenges. For instance, food categories may have similar patterns, while different patterns may exist within a single food category. Additionally, images may contain multiple objects, including two or more food dishes, side dishes, and other noise objects. The matter becomes more complicated due to differences in camera perspectives and varying light conditions (Zhang et al., 2023).

Contribution. To better extract the unique deep features from real-world food images. The significant contributions of this paper are summarized in the following.

- We introduce a CNN-based network for encoding food images to extract robust deep features, namely an adaptive spatial–temporal feature fusion network (ASTFF-Net).
- The ASTFF-Net architecture included three main networks: spatial feature extraction network, temporal feature extraction network, and adaptive feature fusion network.
- We enhanced the overall performance of the proposed ASTFF-Net architecture.
- The experimental results showed that ASTFF-Net significantly outperformed existing state-of-the-art deep learning techniques on four real-world food image datasets: Food11, UEC Food-100, UEC Food-256, and ETH Food-101.

Paper Outline. The remainder of this paper is structured as follows: Section 2 summarizes the overview of related work. Section 3 describes the proposed ASTFF-Net. The real-world food image datasets are explained in Section 4. The experimental results and discussion are presented in Section 5. The conclusion and future work are given in Section 6.

2. Related work

Recently, many approaches have been proposed to address the challenge of real-world food image recognition. The related works are described in this section, including convolutional neural networks, deep feature extraction methods, and deep feature fusion methods.

2.1. Convolutional neural networks (CNNs)

CNN architectures are popular and have been proposed to address the recognition problems in many domains. Many CNN architectures were proposed to recognize food images, such as VGG16, GoogLeNet, InceptionV3 (Hassannejad et al., 2016; Liu et al., 2016; Ege & Yanai, 2017; Vijayakumar & Sneha, 2021). Ng et al. (2019) proposed to use several state-of-the-art CNN architectures comprising MobileNetV2, ResNet50, InceptionV3, InceptionResNetV2, Xception, and NASNet-Large for food image recognition. In their experiments, they evaluated the performance of the CNN architectures on several parameters,

(a) (b) (c)

Fig. 1. Illustrated food images (a) similarities in different food types (b) different decoration and (c) non-food items.

including the impact of the training images, data augmentation techniques, class imbalance, and image resolutions. The results showed that the Xception perform better than other CNNs on UEC Food-100, ETH Food-101, and Vireo-Food 172 datasets.

Martinel, Foresti, & Micheloni (2018) invented wide-slice residual networks (WISeR) based on a residual network. The WISeR architecture contained two parts: residual network and slice network. In the first part, the residual network was employed. In the second part, the slice convolution kernel was proposed. The slice convolution kernel was designed using the rectangle kernel. The width of the rectangle kernel was the same size as the width of the input image. This was different from the standard convolution kernel in that the kernel of the standard convolution was designed as the square kernel. Further, two parts were concatenated and given to fully connected layers. The WISeR architecture achieved an accuracy of 89.58 % on the UEC Food-100, 83.15 % on the UEC Food-256, and 90.27 % on the ETH Food-101 datasets.

Moreover, Tasci (2020) proposed ensemble CNNs using voting combination rules, called voting based CNNs. For the CNN architectures, five CNNs, comprising VGG16, VGG19, GoogLeNet, ResNet101, and InceptionV3, were experimented with. In the ensemble method, six voting methods (minimum, average, median, max, product, and weighted probabilities) were evaluated. The voting-based CNNs yielded 84.28 %, 84.52 %, and 77.20 % accuracy rates on ETH Food-101, UEC Food-100, and UEC Food-256, respectively.

2.2. Deep feature extraction methods

Deep feature extraction methods aim to extract the spatial features from the input images. They are designed to extract features from different layers of deep CNN architectures to enhance accuracy performance. Hence, the deep features are transferred to the recurrent networks and other machine learning techniques to train and create a robust model. Further, the deep features can also be assigned to the LSTM network to extract the temporal features.

Ragusa et al. (2016) used AlexNet, VGG, and Network-in-Network models to extract the deep features from food images. The deep features were then given to classify using the support vector machine (SVM) techniques. The results showed that extracted deep features using AlexNet architecture and classified using the binary SVM outperformed extracted deep features using other CNNs. As a result, training the binary SVM technique on the deep features performed approximately 8 % better than classification using only the CNN technique.

Aguilar, Bolaños, & Radeva (2017a) proposed to use GoogLeNet architecture as the feature extraction method. In their method, first, the deep features were transformed and the best discriminant components selected using principal component analysis (PCA). Second, the best components were trained using the SVM technique. Moreover, in SVM, the grid-search method was used to find the best hyperparameters: cost and gamma. Finally, the optimal SVM model was trained on the best components with the best hyperparameters, then the input images were classified as the food or non-food images. This achieved an accuracy of 94.86 % on the RagusaDS and 99.01 % on the FCD datasets.

The idea of extracting the deep features from various convolution layers was proposed by Farooq & Sazonov (2017). In their method, the deep high-level features were extracted from convolution layers 6, 7, and 8 of the AlexNet architecture. It extracted 4,096, 4,096, and 1,000 features from the images, respectively. Consequently, the SVM classifier trained deep features from layers 6, 7, and 8 separately. As a result, the extracted deep feature from layer 6 achieved the highest accuracy with 70.13 % on the Pittsburgh fast-food image dataset. Furthermore, McAllister et al. (2018) extracted the deep features using ResNet-152 and GoogLeNet architectures from food image datasets. The deep features were then classified using four classifiers consisting of SVM, random forest, neural network, and Naive Bayes. The experimental results showed that it achieved a very high accuracy of 99.4 % on the Food-5 k dataset. Subsequently, it attained an accuracy above 90 % on Food11 and RawFooT-DB datasets. However, it achieved only 64.98 % on the ETH Food-101 dataset.

2.3. Deep feature fusion methods

The previous research mentioned above has shown that deep CNN features achieve high performance in classifying food images. In this section, we will discuss deep feature fusion for food image recognition. Pandey et al. (2017) presented a fusion of three deep CNN features consisting of AlexNet, GoogLeNet, and ResNet to classify benchmark food datasets. In the first layer, three fine-tuned CNNs were used for feature extraction, and the output was concatenated before being passed to ReLU activation followed by a fully connected layer and fed into the softmax function for classification. The experimental result on the ETH Food-101 dataset achieved 72.12 % accuracy. Aguilar, Bolaños, & Radeva (2017b) proposed the CNN fusion method based on inception modules and residual networks. The first step involved separately training two CNN models. Second, the best results in the validation dataset were used in the fusion step using the decision template scheme. The method achieved an accuracy of 86.71 % with the ETH Food-101 dataset.

In addition to the featured fusion methods, adaptive feature fusion has also been introduced for image classification. For example, Li et al. (2020) proposed multi-exemplar images and adaptive fusion of features to enhance blind face restoration. Kumar, Namboodiri, & Jawahar (2020) used the adaptive feature aggregation to recognize a person. The method was to combine the pooled features from multiple locations of the shared feature maps with adaptive weights produced by the attention module. Zhao et al. (2021) introduced a tracking algorithm with a multi-level adaptive feature fusion method. From all the research mentioned above, it was found that the adaptive feature fusion approach increases the efficiency of image recognition. In our study, we used the deep feature technique to extract the feature of the food image and fused the feature with the adaptive spatial–temporal feature fusion method, descript as follows in section 3.

3. Adaptive spatial-temporal feature fusion network (ASTFF-Net)

Overview of the Network. The adaptive spatial–temporal feature fusion network, called ASTFF-Net, is proposed to improve the robustness of the deep features extracted using deep learning methods. The proposed network has the capability to overcome challenges in recognizing various types of food dishes, problems in identifying similar patterns between two categories, multi-objects appearing in a single image, different light conditions, different perspectives in photography, and blurred images. The proposed network contains three schemes: a spatial feature extraction network, a temporal feature extraction network, and an adaptive feature fusion network. The schematic framework of the proposed ASTFF-Net architecture is shown in Fig. 2. The following are the brief details of the ASTFF-Net.

3.1. Spatial feature extraction network

In this network, we propose a spatial feature extraction network, as shown in Fig. 3, to extract the spatial features from various food images. According to experimental results given by Phiphiphatphaisit & Surinta (2020), we chose the ResNet50 architecture (He et al., 2016) which had achieved the best performance on the benchmark ETH Food-101 dataset.

A brief account of ResNet architecture would highlight that ResNet has very deep layers, but the residual block shortcuts the connection from the current layer to one or more layers. The residual block follows two simple rules. – 1) when the input from the previous residual block and output of the current residual block are presented as the same dimension, called identity mapping, it takes outputs from the previous

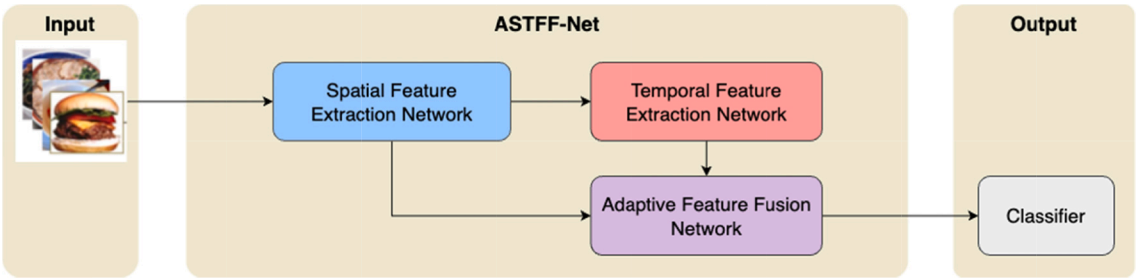


Fig. 2. Overall of the proposed ASTFF-Net.

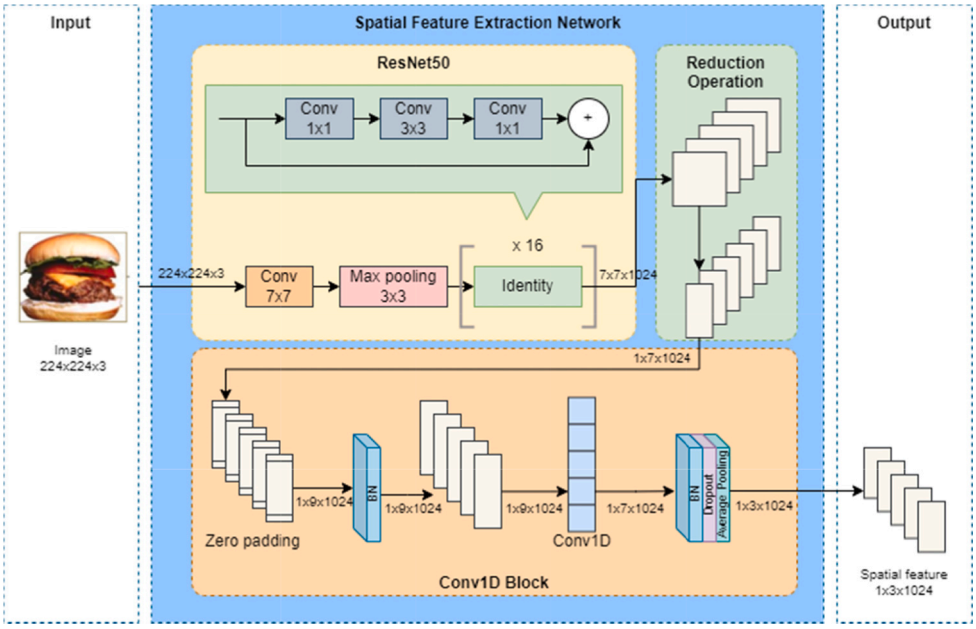


Fig. 3. Illustration of the spatial feature extraction network.

block and adds the output from the skipped layers, as shown in Fig. 4(a). 2) When the input of the previous residual block and output of the current residual block are not the same size, the projection shortcut is implemented to ensure that the output of the residual block is the same size after applying the addition operation, as shown in Fig. 4(b).

In our experiment, we trained the network using a pre-trained model of ResNet50 to speed up the training process. However, we removed the fully connected and extracted the spatial from the last layers of the

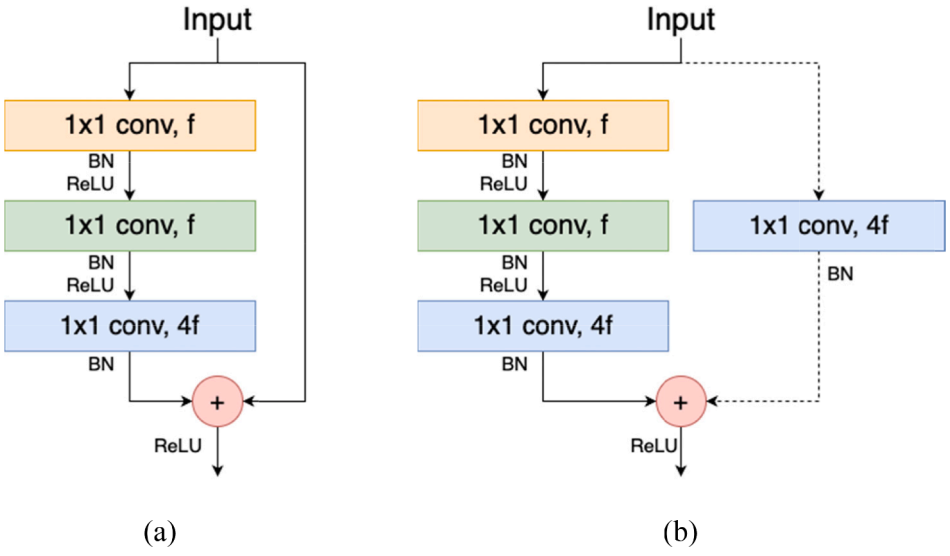


Fig. 4. Bottleneck block for ResNet50: (a) identity shortcut, (b) projection shortcut. (*f* denotes the number of filters).

ResNet50.

We implemented a reduction operation that aimed to adjust the size of the feature maps. The size of the feature maps that were extracted using the ResNet50 was defined by the three dimensions (width × height × number of feature maps). Hence, the input layer of the convolutional 1D block should be in the form of two dimensions. In our study, the reduction operation was installed between the ResNet50 and Conv1D block.

In the proposed Conv1D, the spatial features extracted using the ResNet50 architecture were first given to the reduction operation to transform the feature maps into one dimension. Second, we computed the zero-padding operation to the spatial features, followed by the batch normalization (BN) operation (Ioffe & Szegedy, 2015). Then, the Conv1D operation with a filter size of 1×3 and a stride of 1 was calculated through the spatial features after applying zero padding. Third, three operations: BN, dropout, and average pooling, were attached to the network. Finally, the robust spatial features were obtained from the spatial feature extraction network, as shown in Fig. 3.

3.2. Temporal feature extraction network

This section investigated the long short-term memory (LSTM) network (Hochreiter & Schmidhuber, 1997) to extract the robust temporal features. The LSTM network was proposed to learn patterns in long sequence data by combining cell state and three gates: input, output, and forget. In the LSTM network, the cell state function is to provide relevant sequence information into gates. The gates in the LSTM network choose which information is allowed and which information is related to keep or forget while training.

In this research, a sequence of spatial features is computed using the spatial feature extraction network and directly transferred to the temporal feature extraction network. In order to create a sequence of spatial features, we extract the feature maps from food images using the ResNet50 architecture. Subsequently, the feature maps are sent to the reduction operation to transform the feature maps, followed by the Conv1D operation. As a result, the sequence of spatial features with a size of $1 \times 3 \times 1024$ is then transferred to the temporal feature extraction network, where 1×3 is the input dimension and 1024 is the time steps. This study applied the LSTM network to learn the sequence data extracted using the spatial feature extraction network described in Section 3.2. The temporal feature is the output of the LSTM network, as shown in Fig. 5.

3.3. Adaptive feature fusion network

We propose an adaptive feature fusion network that combines robust spatial–temporal feature networks extracted from the spatial feature extraction network (see Section 3.2) and the temporal feature extraction network (see Section 3.3), as shown in Fig. 6; the details of the adaptive feature fusion network are as follows.

First, the Conv1D block employed in the adaptive feature fusion network differs from the Conv1D in Section 3.1. In this network, the input of the Conv1D block was the temporal feature extracted using the LSTM network. The filter size of 1×3 and a stride of 1 was computed in the Conv1D operation. We trained the network by the rectified linear unit (ReLU) (Nair & Hinton, 2010) activation function followed by the dropout layer (Srivastava et al., 2014). Hence, the neural nodes and their connections were randomly dropped during the training of the model to avoid the overfitting problem that may occur while training the network.

Second, two robust features obtained from the Conv1D block and spatial feature extraction network were fused using concatenation operation. Further, the robust features were given to the global average pooling (GAP) layer (Lin, Chen, & Yan, 2014), followed by the BN layer.

Finally, the robust feature vector was classified using the softmax function.

4. Real-world food image datasets

We evaluated our proposed adaptive feature fusion network (ASTFF-Net) on four benchmark food image datasets: Food11, UEC Food-100, UEC Food-256, and ETH Food-100. The details of each food image dataset are as follows:

4.1. Food11 dataset

Singla, Yuan, & Ebrahimi (2016) established the Food11 dataset that consisted of 16,643 food images of 11 categories that were bread, dairy products, egg, dessert, meat, fried food, pasta, seafood, rice, vegetables/fruit, and soup, as shown in Fig. 7.

4.2. UEC Food-100 dataset

Matsuda & Yanai (2012) collected the UEC Food-100 dataset. It contains 14,361 images from 100 categories of famous Japanese foods, such as sushi, eels on rice, pilaf, beef curry, fried noodle, and tempura. The UEC Food-100 dataset consists of multiple food items in one image (see Fig. 8(a)) and a single food item in one image (see Fig. 8(b)).

4.3. UEC Food-256 dataset

Kawano & Yanai (2014) proposed the UEC Food-256 image dataset, which is the extended version of the UEC Food-101 dataset. First, all the images were collected from Flickr, Bing, and Twitter, using a specific query. Second, the downloaded images were classified using the Foodness method and categorized as food or non-food images. Finally, the

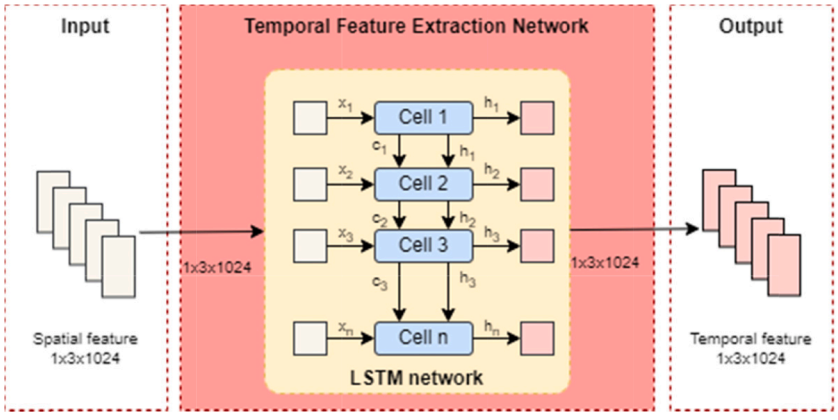


Fig. 5. Illustration of the LSTM network proposed to extract the temporal features.

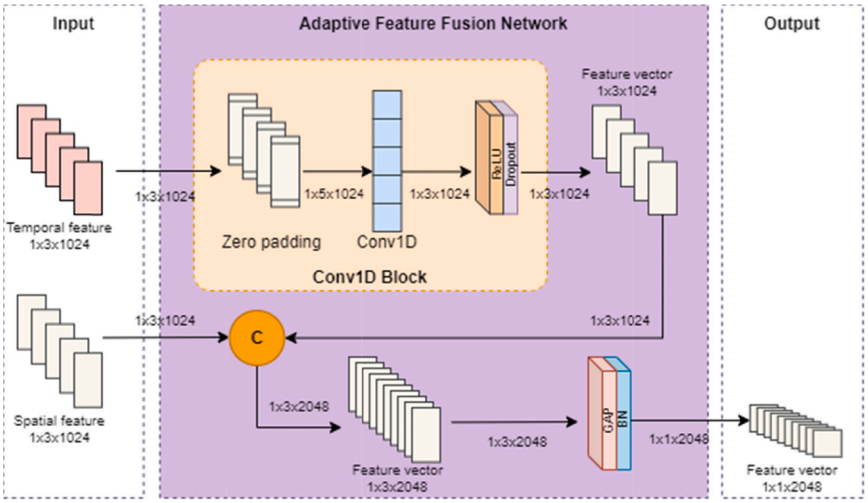


Fig. 6. Illustration of the adaptive feature fusion network.

Fig. 7. Some example images of the Food11 dataset.

UEC Food-256 dataset contained approximately 32,000 food images and comprised 256 categories with more than 600 food images in each category after removing noise images. Examples of the UEC Food-256 dataset are shown in Fig. 9(a).

4.4. ETH Food-101 dataset

The ETH Food-101 dataset was proposed by Bossard & Gool (2014), which consists of the real-world food images downloaded from the website foodspotting.com. It contains 101,000 food images and has 101 food image categories. The examples of the ETH Food-101 dataset are shown in Fig. 9(b).

The summary details of four benchmark food image datasets are shown in Table 1.

5. Experimental results and discussion

In this section, we implemented the adaptive feature fusion network with the TensorFlow platform running on Google Colab with GPU support for all the experiments. The proposed adaptive spatial–temporal feature fusion network (ASTFF-Net) was evaluated on the benchmark food image datasets, comprising Food11, UEC Food-100, UEC Food-256, and ETH Food-101. We divided the food image datasets into training and test sets. The accuracy of the ASTFF-Net was evaluated on the test set. Moreover, we employed 5-fold cross-validation (CV) over the training set to find the significance of the proposed network and prevent

(a)

(b)

Fig. 8. Example of the UEC Food-100 dataset that contains (a) multiple food items and (b) single food item in one image.

overfitting problems.

We evaluated the experimental results using average accuracy, standard deviation, precision, recall, and F_1-score (Fränti & Mariescu-Istodor, 2023). These metrics were calculated as follows.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F_1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

where true positive (TP) is the number of positive instances that were correctly classified, true negative (TN) is the number of negative instances that were correctly classified, false positive (FP) is the number of positive instances that were incorrectly classified, and false negative (FN) is the number of negative instances that were incorrectly classified.

Additionally, we analyzed the floating-point operations per second (FLOPS), the number of floating-point operations the system can execute in one second, to estimate the performance capability of computing systems, including processors and hardware (Cerar, Bertalanič, & Fortuna, 2023). The FLOPS is calculated as follows.

$$FLOPS = \frac{(NumberofFloating - PointOperations)}{TimeinSeconds} \tag{5}$$

In the ASTFF-Net, we used only the pre-trained model of the ResNet50 architecture with pre-trained weights from the ImageNet dataset. However, other parts of the framework do not transfer from the pre-trained model. We trained the ASTFF-Net with the SGD optimizer to optimize the loss function. The adaptive learning rate was proposed with the initial value of 0.01 and then reduced to 0.0001 when the loss value did not decrease after five epochs. The momentum value was set to 0.9 and the weight decay was updated based on the learning rate value and the number of epochs. The ASTFF-Net was trained for only 50 epochs. The hyperparameters and the best settings for the spatial feature extraction network, temporal feature extraction network, and adaptive feature fusion network are shown in Table 2.

To study the efficiency of the ASTFF-Net, we designed four different experiments. First, we combined spatial and temporal features, called the ASTFF-NetB1 model, as shown in Fig. 10(a). Second, the spatial features were sent to the Conv1d block before combining with the temporal features, called the ASTFF-NetB2, as shown in Fig. 10(b). Third, the temporal features were sent to the Conv1D block before combining with the spatial features, called the ASTFF-NetB3, as shown in Fig. 10(c). Finally, both spatial and temporal features were given to the Conv1D block before combining, called the ASTFF-NetB4, as shown in Fig. 10(d).

5.1. Experiments on the Food11 dataset

We trained four ASTFF-Nets on the Food11 dataset on the training data based on five-fold cross-validation (5-CV) and evaluated ASTFF-Net models on a separate test set. The results obtained are presented in Table 3.

From Table 3, we observed that ASTFF-NetB3, in which the temporal features were sent to the Conv1D block before combining with the spatial features, outperformed other ASTFF-Nets on the Food-11 image dataset. The ASTFF-NetB3 achieved 96.08 % accuracy on the training set using 5-CV and 95.04 % accuracy on the test set, which was the best network. On the other hand, ASTFF-NetB2 had the worst performance on both training and test sets. However, performance was only approximately 1.8 % below that of ASTFF-NetB3. Furthermore, we determined testing time to measure the computation time of the ASTFF-Nets. During the testing phase, all ASTFF-Nets achieved similar computational performance. It spent approximately 5 min to process the entire test set, approximately 77.8 ms (ms) per food image.

Fig. 11 illustrates the confusion matrix of four ASTFF-Nets. It was

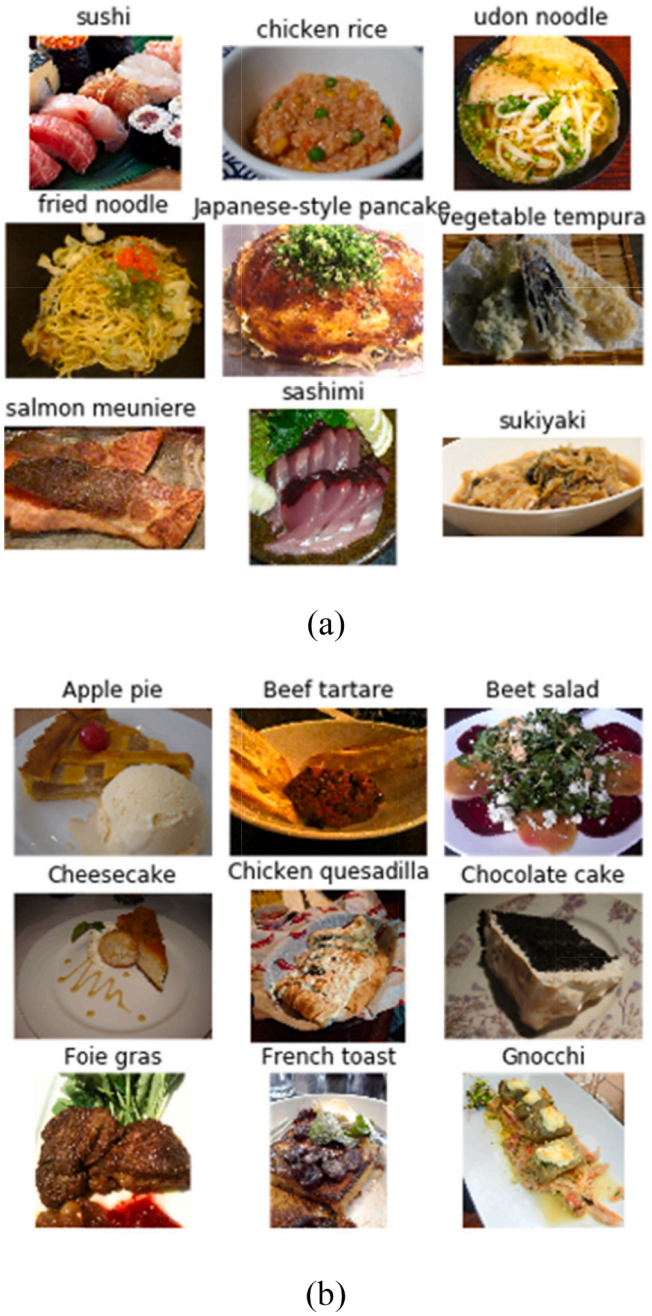


Fig. 9. Illustration of (a) the UEC-Food256 and (b) the ETH Food-101 datasets.

Table 1
Illustration of the details of the benchmark food image datasets.

Dataset	Category	No. of Images	No. of Training Images	No. of Test Images	Images per Category
Food11	11	16,643	12,483	4,160	Imbalanced
UEC Food-100	100	14,361	10,771	3,590	Imbalanced
UEC Food-256	256	31,395	23,547	7,848	Imbalanced
ETH Food-101	101	101,000	75,750	25,250	1,000

Table 2
The best hyperparameter settings of spatial feature extraction, temporal feature extraction, and adaptive feature fusion networks.

Hyperparameter	Parameter Setting of		
	Spatial Feature Extracted Network	Temporal Feature Extracted Network	Adaptive Feature Fusion Network
Epoch	50	50	50
Batch size	32	32	32
Optimizer	SGD	SGD	SGD
Learning rate	Schedule 0.01 to 0.0001	Schedule 0.01 to 0.0001	0.0001
Momentum	0.9	0.9	0.9
Weight decay	Learning rate/ epoch	Learning rate/ epoch	0.0001
Dropout	0.2	0.2	0.2

found that the ASTFF-NetB3 (see Fig. 11(c)) reduced the misclassified number of images from category egg to bread. It reduced the misclassified images from 17 images to only two images. Also, the rice category that was misclassified to the fruit/veg category was reduced from 4 images to zero.

Fig. 12 shows the probability of the egg (see Fig. 12(a)) and rice (see Fig. 12(b)) categories that were corrected classified using ASTFF-NetB3, but other ASTFF-Nets misclassified it.

In Table 4, we present extensive comparisons of our ASTFF-Nets on the Food11 dataset with existing state-of-the-art methods. The experimental results confirmed that our ASTFF-Nets increased the accuracy performance. Additionally, our ASTFF-Nets showed much better results than extracting the deep features using CNN architectures and combining them with machine learning techniques, such as artificial neural networks and support vector machines (McAllister et al., 2018; Şengür, Akbulut, & Budak, 2019). In conclusion, the ASTFF-NetB3 resulted in the highest accuracy performance of 95.04 %.

5.2. Experiments on the UEC Food-100 dataset

This section showed that our ASTFF-Nets also achieved the best accuracy performance on the UEC Food-100 dataset, which has 100 food categories. The results attained throughout the testing process are shown in Table 5.

From Table 5, it is seen that ASTFF-NetB3 significantly outperformed other ASTFF-Nets on the UEC Food-100 dataset (t-test, $p < 0.05$). We observed that the ASTFF-NetB3 performed with higher than 4 % accuracy on the 5-CV and higher than 5 % accuracy on the test set when compared with other ASTFF-Nets. Another observation is that the ASTFF-NetB3 achieved an F_1-score of more than 0.90, which means that the ASTFF-NetB3 successfully classified food images over a specific strength with a low false-positive rate. Moreover, all the ASTFF-Net architectures were rapid with the test set, requiring approximately 77.80 ms per food image.

We illustrated the food images that were correctly classified when using the ASTFF-NetB3 model, as shown in Fig. 13(a). All the food images contained only one dish, which means only one food category appeared in the image. On the other hand, the mostly misclassified food images, as shown in Fig. 13(b), always included many objects in one image. For example, the rice dish appears in sauteed vegetables and ganmodoki categories.

Table 6 compares the performance of our approach architectures on the UEC Food-100 dataset with existing deep learning techniques. The accuracy performance of the previous deep learning techniques did not achieve very high scores, even using the ensemble CNNs method (Tasci, 2020). The highest accuracy was 90.20 % with the visual aware hierarchy method (Mao et al., 2021). However, the ASTFF-NetB1, B2, and B4 did not achieve higher performance than the WISeR method. Consequently, the proposed ASTFF-NetB3 network, that directly gives

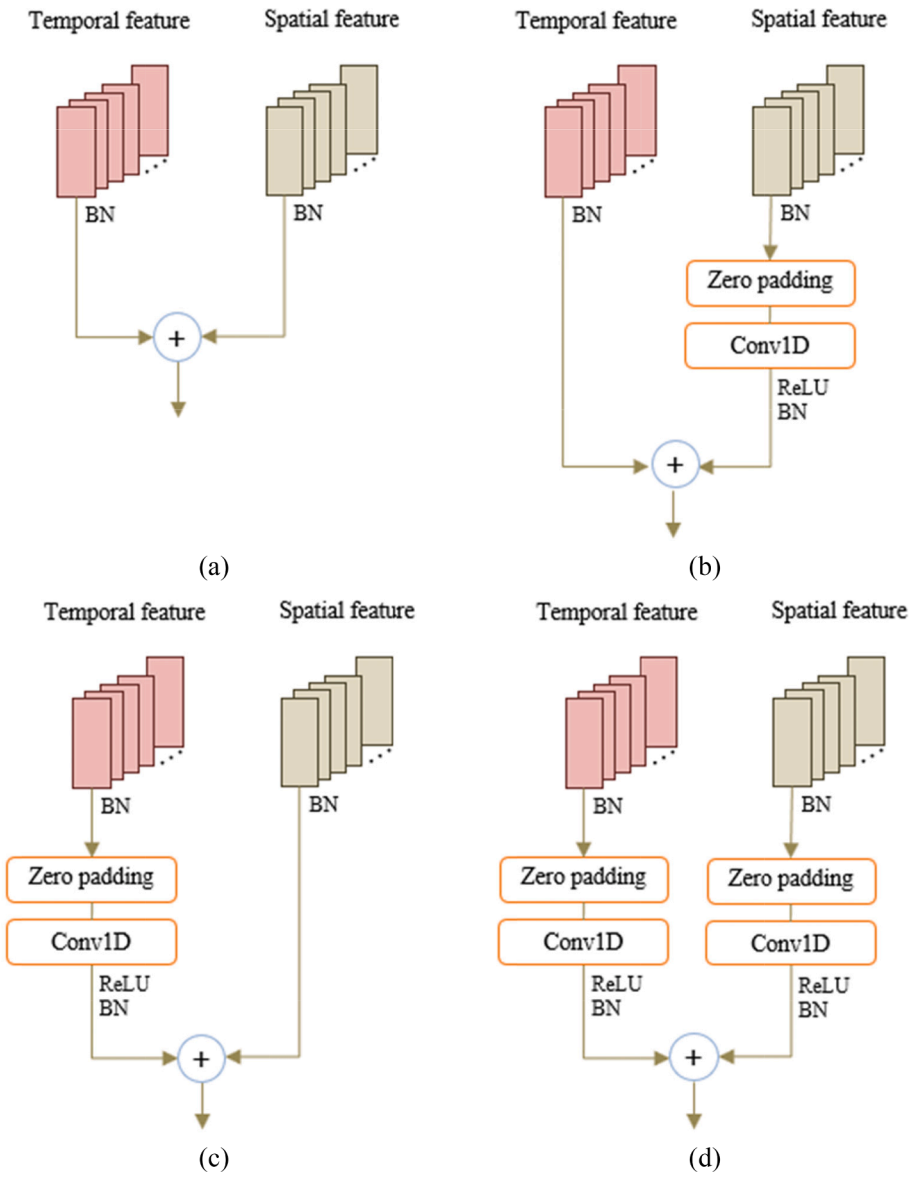


Fig. 10. Illustration of four ASTFF-Nets. (a) The ASTFF-NetB1, (b) ASTFF-NetB2, (c) ASTFF-NetB3, and (d) ASTFF-NetB4.

Table 3
Evaluation performances (average accuracy, ± standard deviation, test accuracy, recall, and F_1-score) of the ASTFF-Nets on the Food11 dataset. The bold numbers represent the best ASTFF-Net model.

Model	5-CV	Test Accuracy (%)	Recall	F_1-score	Testing Time (ms/image)
ASTFF-NetB1	94.26 ± 0.177	93.47	0.935	0.935	77.4
ASTFF-NetB2	94.17 ± 0.291	93.16	0.932	0.932	77.8
ASTFF-NetB3	**96.08 ± 0.330**	**95.04**	**0.950**	**0.950**	77.8
ASTFF-NetB4	95.54 ± 0.369	94.63	0.946	0.946	78.2

the temporal feature to the Conv1D block and then combines it with the spatial features, demonstrated the highest performance with 91.35 % accuracy.

5.3. Experiments on the UEC Food-256 dataset

In this section, we evaluated the proposed adaptive network on the UEC Food-256 dataset in terms of 5-CV, test accuracy, recall, and F_1-score. It has a huge category with 256 menus from Japan and other countries. The proposed ASTFF-Nets were evaluated on 23,547 training images and 7,848 test images.

Table 7 shows the evaluation performance of the ASTFF-Nets. We observed that the ASTFF-NetB3 consistently achieved the highest accuracy and significantly outperformed other ASTFF-Nets (*t*-test, $p < 0.05$) on both 5-CV and test sets. The ASTFF-NetB2 slightly decreased the performance on the UEC Food-256 dataset. Consequently, our proposed ASTFF-Nets achieved above 90 % accuracy. It spent approximately 10 min on the whole test set (approximately 77.8 ms per food image).

As illustrated in Fig. 14, we discovered that some food images have similar texture, color, and pattern characteristics that could harm the proposed ASTFF-Nets leading to misclassification.

In Table 8, it can be seen that the existing deep learning methods did not show high accuracy. The WISeR method (Martinel, Foresti, & Micheloni, 2018) and the visual aware hierarchy method (Mao et al.,

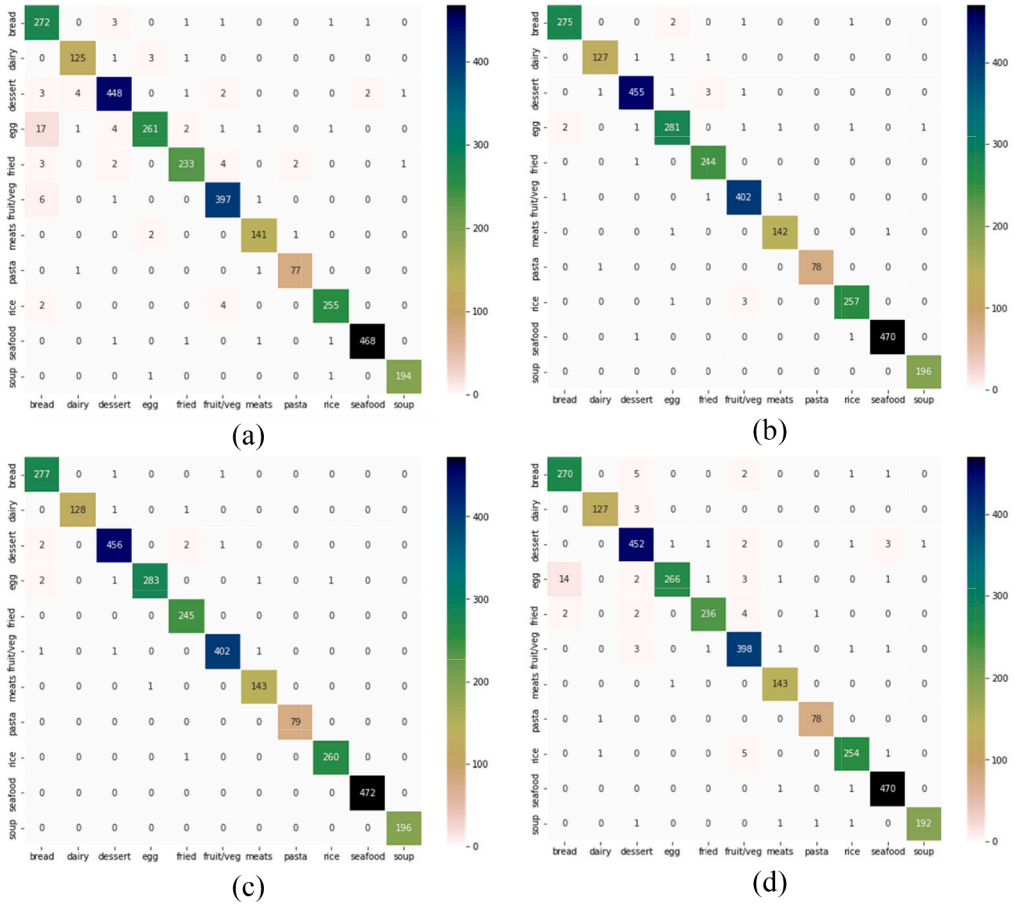


Fig. 11. Illustration of the confusion matrix of ASTFF-Nets on the Food11 dataset: (a) ASTFF-NetB1, (b) ASTFF-NetB2, (c) ASTFF-NetB3, and (d) ASTFF-NetB4.

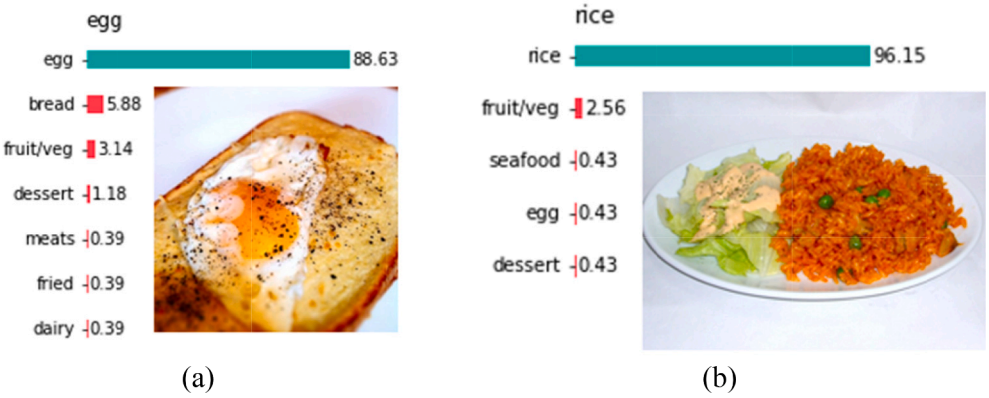


Fig. 12. Examples of similar categories between (a) egg and bread and (b) rice and fruit/veg categories classified using ASTFF-NetB3.

Table 4
Recognition performance on the Food11 dataset when compared with existing methods.

Reference	Method	Test Accuracy (%)
McAllister et al. (2018)	ResNet152 + ANN	91.34
Şengür, Akbulut, & Budak (2019)	AlexNet + VGGl6 + SVM	88.08
Our Proposed	ASTFF-NetB1	93.47
	ASTFF-NetB2	93.16
	ASTFF-NetB3	**95.04**
	ASTFF-NetB4	94.63

Table 5
The experimental results of the proposed ASTFF-Nets on the UEC Food-100 dataset.

Model	5-CV	Test Accuracy (%)	Recall	F_1-score	Testing Time (ms/image)
ASTFF-NetB1	86.77 ± 0.231	85.70	0.857	0.857	77.4
ASTFF-NetB2	86.99 ± 0.267	86.05	0.861	0.861	77.7
ASTFF-NetB3	**92.55 ± 0.168**	**91.35**	**0.914**	**0.914**	77.7
ASTFF-NetB4	89.85 ± 0.344	88.85	0.889	0.889	78.2

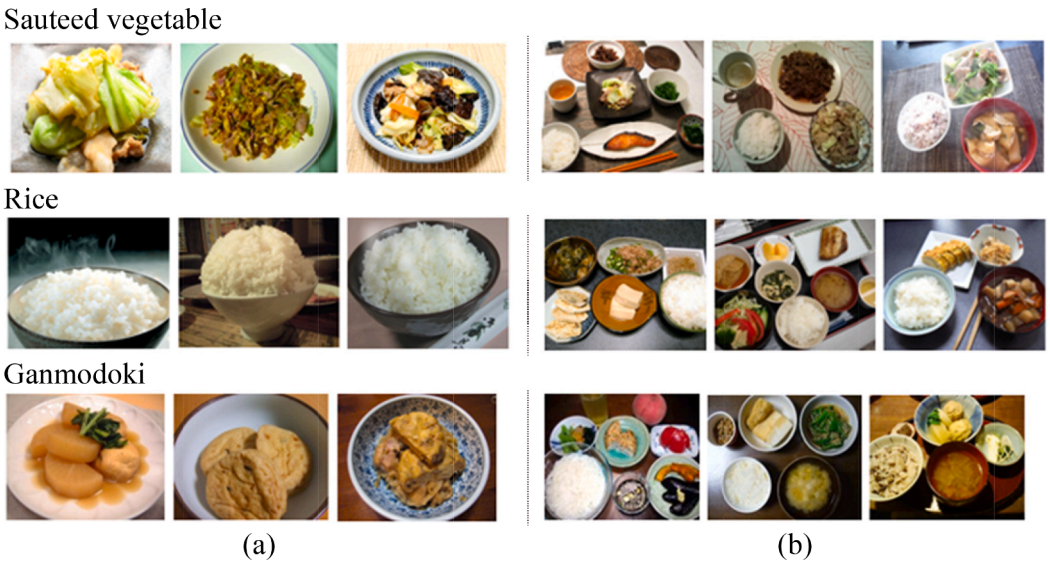


Fig. 13. Some examples of sauteed vegetables, rice, and ganmodoki images of the UEC Food-100 dataset were classified using the ASTFF-NetB3 model. The food images were (a) correctly classified and (b) misclassified.

Table 6
Recognition performance on the UEC Food-100 dataset when compared with existing deep learning techniques.

Reference	Method	Test Accuracy (%)
Liu et al. (2016)	DeepFood	76.30
Hassannejad et al. (2016)	InceptionV3	81.45
Martinel, Foresti, & Micheloni (2018)	WISeR	89.58
Tasci (2020)	Ensemble CNNs	84.52
Mao et al. (2021)	Visual Aware Hierarchy	90.20
Our Proposed	ASTFF-NetB1	85.70
	ASTFF-NetB2	86.05
	ASTFF-NetB3	**91.35**
	ASTFF-NetB4	88.85

Table 7
Recognition performance of the proposed ASTFF-Nets on the UEC Food-256 dataset.

Model	5-CV	Test Accuracy (%)	Recall	F_1-score	Testing Time (ms/image)
ASTFF-NetB1	92.16 ± 0.192	91.07	0.911	0.911	77.4
ASTFF-NetB2	92.05 ± 0.155	90.90	0.909	0.909	77.8
ASTFF-NetB3	**93.21 ± 0.324**	**92.15**	**0.921**	**0.921**	77.8
ASTFF-NetB4	92.40 ± 0.301	91.37	0.914	0.914	78.2

2021) achieved the performance with an accuracy of only 83.37 % and 83.15 % respectively. The proposed ASTFF-Nets performed much better than the previous methods and achieved greater than 90 % accuracy. Consequently, the ASTFF-NetB3 always achieved the best performance with an accuracy of 92.15 %, which is approximately 9 % more than with the WISeR method.

5.4. Experiments on the ETH Food-101 dataset

In this experiment, we tested the proposed adaptive network on the ETH-Food101 dataset, which has 75,750 training images and 25,250 test images. It is the largest food image dataset that we evaluated in our experiments. The results of the proposed ASTFF-Nets are shown in Table 9.

Table 9 reports that the ASTFF-NetB3 still achieved the best performance when compared with other ASTFF-Nets (*t*-test, $p < 0.05$, significant). It achieved a performance of 93.98 % accuracy on 5-CV and 93.06 % accuracy on the test set. Furthermore, we found that the ASTFF-NetB3 achieved the highest accuracy on four food image datasets: ETH Food-101, Food11, UEC Food-100, and UEC Food-256. The computational time with all the ASTFF-Net architectures was approximately 77.8 ms per food image on the test set.

We also observed that ASTFF-NetB3 achieved an F_1-score of 0.931 with a high true-positive rate. The illustration of the F_1-score, when classified using the ASTFF-Nets, is shown in Fig. 16. Moreover, for further investigation, we found noise and non-food objects in some food categories, such as apple pie and Peking duck. Examples of the noise and non-food objects are shown in Fig. 15.

Table 10 compares results with our proposed ASTFF-Nets with other methods. We observed that extraction the deep features using CNN, Conv1D, and LSTM performed better than training with only CNN architectures (Phiphitphatphaisit & Surinta, 2021) and even better than extracting the deep features and combined with machine learning techniques. The results in Table 10 show that our ASTFF-NetB1, B3, and B4 were given an accuracy above 90 %. These networks also outperformed various existing methods. Consequently, the ASTFF-NetB3 achieved an accuracy of 93.06 %, which is the highest performance on the ETH Food-101 dataset.

5.5. Discussion

In this research, we discussed several important issues that affect the performance of the CNN models.

5.5.1. Overfitting with robust network

Naturally, overfitting problems occur when very deep CNN layers are proposed to create the robust CNN model and trained with too many example images. With very deep CNN architectures, the CNN model needs to optimize many hyperparameters. To face this problem, we proposed the adaptive spatial–temporal feature fusion network, called ASTFF-Net, which was invented to combine both spatial and temporal feature extraction networks. The adaptive architectures were designed

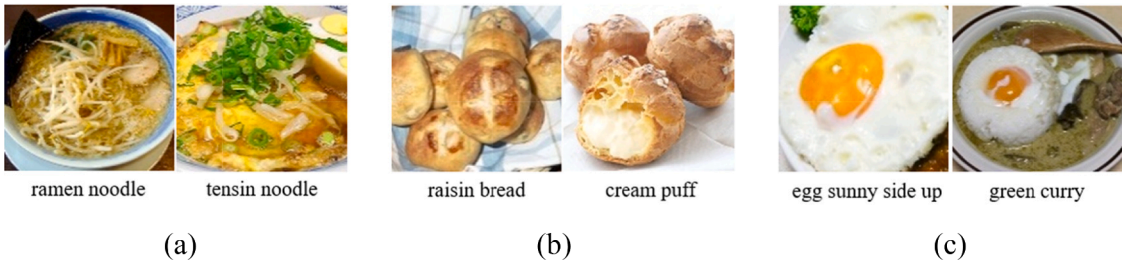


Fig. 14. Illustration of the similar food images between (a) ramen noodle and tensin noodle, (b) raisin bread and cream puff, and (c) egg sunny side up and green curry. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 8
Recognition performance on the UEC Food-256 dataset when compared with existing deep learning techniques.

Reference	Method	Test Accuracy (%)
Liu et al. (2016)	DeepFood	54.70
Hassannejad et al. (2016)	InceptionV3	76.17
Martinel et al. (2018)	WISeR	83.15
Tasci (2020)	Ensemble CNNs	77.20
Mao et al. (2021)	Visual Aware Hierarchy	83.37
Our Proposed	ASTFF-NetB1	91.07
	ASTFF-NetB2	90.90
	ASTFF-NetB3	**92.15**
	ASTFF-NetB4	91.37

Table 9
Recognition performance of the proposed ASTFF-Nets on the ETH Food-101 dataset.

Model	5-CV	Test Accuracy (%)	Recall	F_1-score	Testing Time (ms/image)
ASTFF-NetB1	91.88 ± 0.229	91.13	0.911	0.911	77.4
ASTFF-NetB2	90.16 ± 0.276	89.05	0.890	0.890	77.8
ASTFF-NetB3	**93.98** ± 0.247	**93.06**	**0.931**	**0.931**	77.8
ASTFF-NetB4	93.56 ± 0.224	92.81	0.928	0.928	78.2

to extract information on the spatial domain and ignore some insignificant information using the temporal network. We evaluated the proposed method using a five-fold cross-validation method (5-CV), as shown in Table 8, and we found that the ASTFF-Nets could learn well with many training examples and generalize well with the test set. There was no great difference between 5-CV and the test set results.

5.5.2. *Similarity patterns between two categories*

The real-world food images from the benchmark datasets were downloaded from the internet. Some of the images contained many noise objects (see Fig. 15(a)), some images had similar patterns (see Fig. 14(b)) and some images contained similar food objects (see Fig. 14 (c)) that appeared in many food categories. For example, the category of the bread dish was classified as the egg category because the bread is served with egg. We then presented the F_1-score to measure the precision of the ASTFF-Net architecture. Furthermore, the confusion matrix, as shown in Fig. 11(c), confirmed that ASTFF-NetB3 can address the similarity pattern between two classes: egg and bread.

5.5.3. *Multi-object problems*

The UEC Food-100 dataset usually contains the multi-object appearing in one image, as shown in Fig. 13(b). It is not easy to recognize as the correct category because many dishes are included in the image. As a result, it is misclassified. With the multi-object problem, we carefully checked the recognition results of the proposed ASTFF-Nets and found that the proposed network recognized one correct dish from many dishes that appear in one image. For example, the image contain fish, soup, rice, and sauteed vegetables in the sauteed vegetable category. So, the ASTFF-NetB3 classified it as rice, which was one

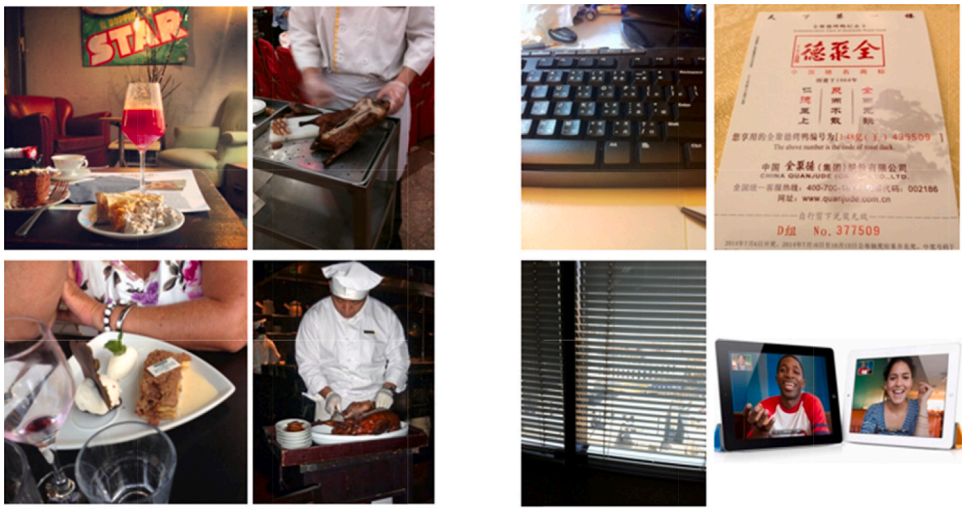


Fig. 15. Example of (a) noise and (b) non-food objects that appear in the ETH Food-101 images.

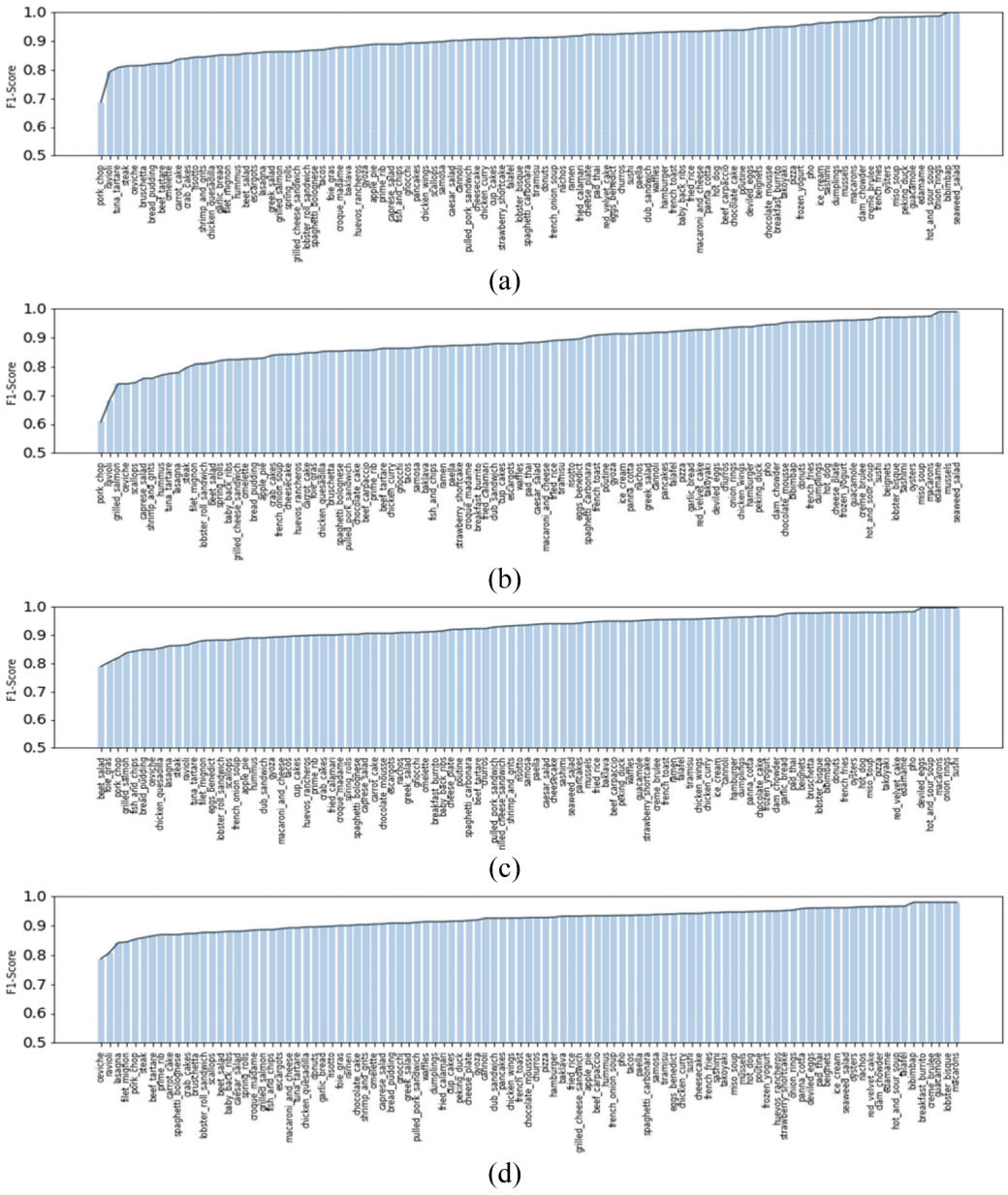


Fig. 16. Illustration of the F_1-score of each category when classified using (a) ASTFF-NetB1, (b) ASTFF-NetB2, (c) ASTFF-NetB3, and (d) ASTFF-NetB4 on the ETH Food-256 dataset.

category from many categories from the image. To address the multi-object problem, thus, we recommend applying object detection and classifying each object.

5.5.4. The effect of the lighting conditions, camera positions, noise objects, and blurred images

Different people and photographers generally capture real-world food images using various types of cameras and camera phones. Two major factors determine the quality of food images: the quality of the camera phone and the skills of the photographers. For the DSLR camera, adjusting various parameters, such as focus, flash, shutter speed, and ISO, can enhance food photography. For the camera phone, when taking photos in low light conditions with a low-quality camera phone, the resulting picture may contain noise and blur, resulting in a lower-quality image. Additionally, people may capture food dishes from different perspectives, including close-ups, bird-eye views, and even the composition of other things in the food dish (called noise objects), which impact recognition accuracy.

To address the issues mentioned above, we utilized ASTFF-NetB2, our top-performing food image recognition model, to demonstrate its ability to recognize food dishes in various light conditions, camera positions and noise objects, and in blurred images, as follows.

Light Conditions. To handle different light conditions, we provided illustrations for three conditions: natural and adequate lights (see Fig. 17 (a)), flashlight (see Fig. 17(b)), and low light (see Fig. 17(c) and (d)). The results indicate that ASTFF-NetB3 was accurately recognized with over 90 % confidence, as shown in Fig. 17(a)–(c). The performance of the ASTFF-NetB3 in recognition of the bread dish was affected by low light by decreasing the confidence value to 46.4 %, as illustrated in Fig. 17(d). Nevertheless, the proposed network was still correctly recognized in low-light conditions.

Camera Positions and Noise Objects. We worked with images of various meat dishes captured from different camera positions and noise objects, as shown in Fig. 18. The yield issues affected confidence by decreasing the confidence value to approximately 70–80 %. Furthermore, the ASTFF-NetB3 still provided accurate recognition.

Table 10
Recognition performance of the ETH Food-101 dataset when compared with different deep learning techniques.

Reference	Method	Test Accuracy (%)
Liu et al. (2016)	DeepFood	77.40
Hassannejad et al. (2016)	InceptionV3	88.25
Bolanos & Radeva (2016)	GoogLeNet	79.20
Pandey et al. (2017)	EnsembleNet	72.12
Aguilar et al. (2017b)	CNNs Fusion	86.71
Martinel et al. (2018)	WISeR	90.27
McAllister et al. (2018)	ResNet152 + SVM-RBF	64.98
Şengür, Akbulut, & Budak (2019)	AlexNet + VGGl6 + SVM	79.86
Tasci (2020)	Ensemble CNNs	84.28
Mao et al. (2021)	Visual Aware Hierarchy	87.82
Phiphiphatphaisit & Surinta (2020)	Modified MobileNetV1	72.59
Phiphitphatphaisit & Surinta (2021)	ResNet50 + Conv1D-LSTM	90.87
Our Proposed	ASTFF-NetB1	91.13
	ASTFF-NetB2	89.05
	ASTFF-NetB3	**93.06**
	ASTFF-NetB4	92.81

Blurred Images. The different types of cooked eggs and egg dishes with distinct decorations, along with blurred images, are shown in Fig. 19. These factors are crucial in identifying food images accurately. Although ASTFF-NetB3 correctly recognized them, it had a lower confidence value ranging from 60-75 %.

5.5.5. *Computational cost and model size*

We designed the ASTFF-Nets according to the advantage of extracting the spatial and temporal deep features. Further, three networks were included in the ASTFF-Nets: spatial feature extraction, temporal feature extraction, and adaptive feature fusion. Indeed, the ASTFF-Nets had a larger model size than the CNN and CNN-LSTM networks, as shown in Table 10. However, when we evaluated the proposed ASTFF-Nets on the test set, the computation cost of the ASTFF-Nets did not significantly increase. It increased only around four milliseconds and only 0.6 ms compared with the ResNet50 and CNN-LSTM respectively. A comparison of the FLOPS, testing time, and model size is shown in Table 11.

As shown in Table 11, we use the FLOPS to measure the computing performance of the ASTFF-Net architectures. ASTFF-NetB1 has the best FLOPS, only 4.61G. Additionally, when comparing the FLOPs of ASTFF-NetB1 and ResNet50, we found that the difference between the two architectures was only 0.8G. Note that in Table 11, G is 10^9 and M is 10^6.

Furthermore, we employed the testing time to measure ASTFF-Nets

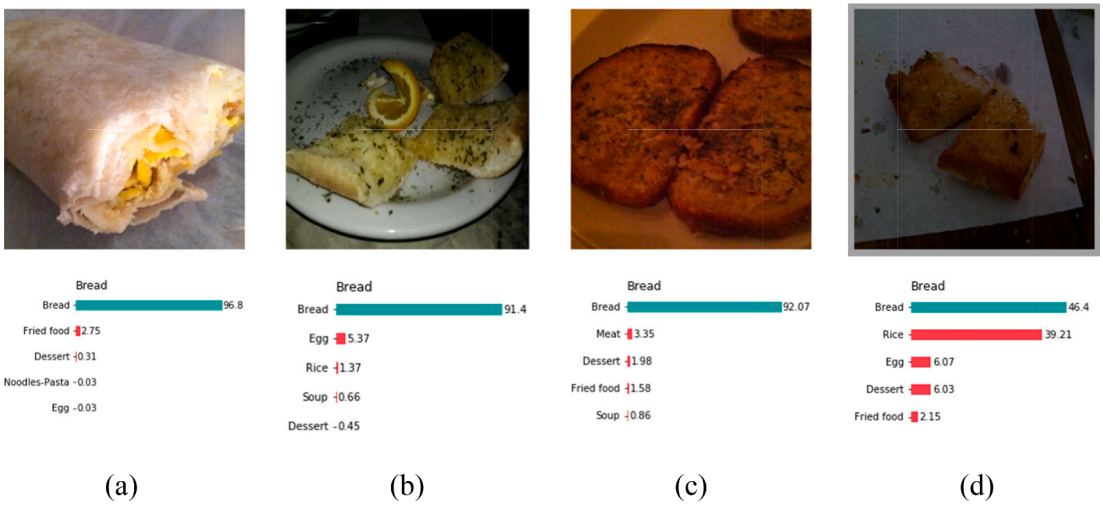


Fig. 17. Example images of bread dishes captured under various light conditions: (a) natural and adequate light, (b) flashlight, (c), and (d) low light.

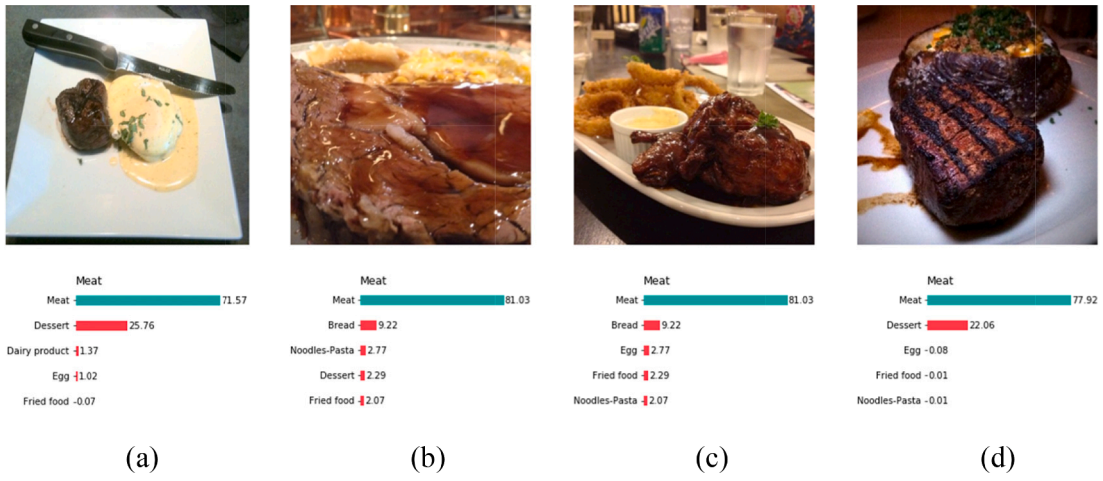


Fig. 18. Example images of meat dished captured under various camera positions: (a) bird-eye view and noise object, (b) close-up, (c), side angle and contain other food and noise object, and (d) side angle.

Fig. 19. Example blurred images of egg dishes presented in various types of cooked egg and decorations.

Table 11
The comparison of the computational cost and model size between the proposed ASTFF-Nets and other architectures.

Method	FLOPS	Testing Time (ms/image)	Model Size
ResNet50	3.8G	73.2	24.6 M
ResNet50 + Conv1D-LSTM (Phiphitphatphaisit & Surinta, 2021)	4.34G	77.2	38.3 M
ASTFF-NetB1	4.61G	77.4	38.4 M
ASTFF-NetB2	7.36G	77.8	41.5 M
ASTFF-NetB3	7.36G	77.8	41.5 M
ASTFF-NetB4	8.24G	78.2	78.2 M

performance. During the testing phase, all ASTFF-Nets achieved similar computational performance. Each image took 77.8 ms to compute. The ASTFF-Nets were slightly slower than ResNet50, which computed at 73.2 ms per food image. ASTFF-NetB3 achieved the best performance on UEC Food-100, UEC Food-256, and ETH Food-101 with a model size of 41.5 M, which is 16.9 M smaller than the ResNet50 architecture, while the speed observed during the testing phase is almost the same.

Compared to ResNet50, ASTFF-Nets have a larger model size due to the inclusion of three networks: spatial feature extracted network, temporal feature extracted network, and adaptive feature fusion network, that extract spatial and temporal features. Further, the ASTFF-NetB1 has the smallest model size of 38.4 M. The ASTFF-NetB4 is the biggest among the ASTFF-Nets, with a model size 78.2 M.

5.5.6. *The impact of training sizes and the quality of training samples*

We conducted two experiments based on the proposed ASTFF-NetB1 architecture using the UEC Food-100 dataset to evaluate the effect of training sample size and quality.

In the first experiment, we investigated the impact of training sizes by randomly selecting 20 percent of the test set, which consisted of 2,873 food images. We applied this strategy consistently across all subsequent experiments. Experimental results illustrated that using 80 % of the training set, which consisted of 11,488 images, resulted in the highest accuracy of 87.39 %. Therefore, it can be concluded that a large training set is essential for constructing an effective deep learning model. The impact of the training sizes is shown in Table 12 and Fig. 20.

However, a decrease in the training set size by 5–10 % slightly impacts test accuracy, resulting in a reduction of approximately 2 %. Additionally, we plan to employ instance selection methods to curate a robust training set (Branikas et al., 2019; Malekipirbazari et al., 2021) before training the model.

In the second experiment, we observed that within the UEC Food-100 dataset, a significant proportion of misclassifications occurred in food images containing multiple food items or a combination of food and other objects. To assess the impact of dataset cleaning, we compared the original UEC Food-100 dataset with a cleaned version. Initially, we removed 2,567 unnecessary images to address the issue of food images containing unrelated objects. As a result, the original dataset was reduced from 14,361 to 11,785 food images. Example images of the unrelated objects are shown in Fig. 21. Subsequently, we split the remaining food images into training and test sets using a 75:25 ratio. Finally, we trained ASTFF-Net models (B1-B4) on the 75 % training set

Table 12
Recognition performance of the proposed ASTFF-NetB1 when training with different sizes of training images.

Training Set		Test Accuracy (%)
Percentage (%)	No. of Training Images	
80	**11,488**	**87.39**
75	10,770	86.10
70	10,052	85.25
65	9,334	83.65
60	8,616	83.25
55	7,898	82.30
50	7,180	81.40
45	6,462	78.80
40	5,744	78.30

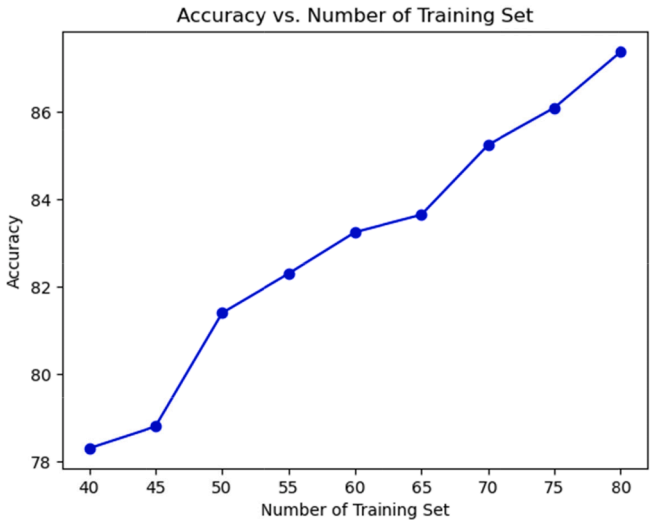


Fig. 20. Illustrated the impact of the training images with the accuracy of the proposed ASTFF-NetB1.

Fig. 21. Illustrated the non-food and unrelated objects that appear in the food images.

and evaluated their performance on the test set.

After removing non-food and unrelated objects, Table 13 illustrates that the recognition performance of ASTFF-Nets (B1-B3), when trained on the cleaned data of the UEC Food-100 dataset, increased by approximately 1 %. Additionally, we plan to apply food segmentation techniques to the images before training the model, aiming further to remove non-food and unrelated objects from the food images.

5.5.7. Comparison with the existing methods

Most existing food recognition methods focus on extracting spatial features using CNN architectures. Şengür, Akbulut, & Budak (2019) used two CNN architectures, VGG16 and AlexNet, to extract spatial features four times from food images. Hence, four deep features were concatenated and then classified using a support vector machine (SVM) algorithm. McAllister et al. (2018) extracted deep features using pre-trained deep CNN models of ResNet152 and GoogLeNet. The deep features were then trained again using machine learning algorithms, including naive Bayes, SVM, artificial neural network (ANN), and random forest. These two CNN-based techniques extract spatial features and train them with machine learning algorithms.

Additionally, to achieve better accuracy performance, Tasci (2020) fine-tuned multiple CNN models, including ResNet, GoogLeNet, VGGNet, and InceptionV3. Hence, the voting rule-based probabilities method was employed to determine the output class from the maximum probability. Moreover, Mao et al. (2021) proposed a two-step recognition that includes food localization using a faster R-CNN method and hierarchical classification using CNNs, called the visual aware hierarchy method. Their method mainly recognizes food in specific locations and contains various food categories in a single food image.

However, the proposed ASTFF-Net extracts robust deep features from real-world food images using three main networks: a spatial feature extraction network, a temporal feature extraction network, and an adaptive feature fusion network to extract spatial and temporal features. In our proposed network, we employed only ResNet50 and Conv1D architectures to extract spatial features. Concurrently, the LSTM network was used to extract long sequence patterns from food images. The network also responds to various challenges in food recognition, such as numerous food dishes, identifying similar patterns between two food categories, encountering objects appearing in one image, variations in light conditions and perspectives, and noise occurring in the food images.

Table 13
Comparison of the recognition performance of ASTFF-Nets trained on the original and cleaned data of the UEC Food-100 dataset.

Model	Test Accuracy (%) on the UEC Food-100 Dataset	
	Original Data	Cleaning Data
ASTFF-NetB1	85.70	**86.22**
ASTFF-NetB2	86.50	**87.16**
ASTFF-NetB3	91.35	**91.94**
ASTFF-NetB4	88.85	**89.22**

5.5.8. Integrate the concept of relevance feedback into the ASTFF-Net

Relevance feedback, one of the adaptive learning strategies, is achieved in content-based image retrieval (CBIR) systems. In relevance feedback, similarity measure algorithms are employed to measure relevance based on user interaction. This approach improves retrieval performance by dynamically updating queries and similarity measures according to user preferences. However, relevance feedback sometimes overlooks valuable historical data from other users, potentially resulting in a loss of useful information (Salton & Buckley, 1990; Zhou & Huang, 2003; Doulamis & Doulamis, 2006).

Indeed, the user plays an essential role in the recognition process by providing relevance feedback. When combining the relevance feedback into the proposed ASTFF-Net, relevance feedback will involve the user providing feedback on the accuracy of recognized food images. This feedback is used to refine the model's predictions over time and empowers the user to actively contribute to the system's performance and adaptability to their preferences.

For example, if the user identifies misclassified images, they can provide feedback to the system, indicating which images were incorrectly classified. The system can then adjust its parameters or training data accordingly to recognize similar images in the future. This iterative process of user interaction and feedback enhances the recognition capabilities of ASTFF-Net. It makes the user an integral part of the system, making it more effective in real-world applications.

6. Conclusions

In this research, an adaptive spatial–temporal feature fusion network, namely ASTFF-Net, was invented to improve the food image recognition performance. In other food recognition systems, a convolutional neural network (CNN) is usually proposed to extract the spatial features from the food images. However, real-world food images sometimes contain many noise and non-food objects, resulting in the CNN extracting deep features containing information of the object mentioned. Consequently, we proposed to use ResNet50 to extract the spatial features and directly send them to the convolutional 1D (Conv1D) block, followed by a long short-term memory (LSTM) network. The LSTM network has gate operations designed to learn sequence patterns from spatial information and allow which information to keep or forget during the training scheme.

The ASTFF-Net architecture is divided into three parts as follows. First, the spatial feature extraction network, we proposed to use the state-of-the-art CNN model, namely ResNet50, to extract temporal features. Then, the reduction operation was attached to the ResNet50 to minimize the size of the feature maps before sending them to the

Conv1D block. Second, the temporal feature extraction network, the sequence output of the Conv1D block was assigned to the LSTM network to create temporal features. Third, the spatial and temporal features from the first and second parts were combined using concatenation operation, then assigned to the Conv1D, called adaptive feature fusion network. As with the ASTFF-Net, the softmax function was connected to the ASTFF-Net as the recognition layer proposed to recognize real-world food images. The ASTFF-Net architecture was proposed to address the overfitting problems because we combined the global average pooling (GAP) and dropout layers to the architecture. The most benefit of the GAP layer is that the ASTFF-Net parameter was reduced. Additionally, the unnecessary connections between layers were dropped using the dropout layer.

In the experiments, we evaluated four ASTFF-Nets on four different real-word food image datasets: Food11, UEC Food-100, UEC Food-256, and ETH Food-101. The results show that the ASTFF-Nets achieved the highest accuracy on 5-CV and the test set. Furthermore, we found that the proposed ASTFF-NetB3 outperformed the existing methods on four food image datasets.

In future research, we will apply the ASTFF-Nets to address the challenge of unbalanced datasets (Aggarwal, Popescu, & Hudelot, 2020; Özdemir, Polat, & Alhudhaif, 2021). Another direction will be applying the instance selection methods (Branikas et al., 2019; Malekipirbazari et al., 2021) to reduce the training set. It may reduce the training set by more than 50 %; the computational time will decrease while training the ASTFF-Nets. We will consider segmentation techniques (Hafiz & Bhat, 2020; Ye et al., 2023) that can select the most relevant food region from real-world food images. To compare the efficiency of the ASTFF-Net, we will also apply it to other image recognition tasks, such as vehicle, plant leaf disease, and land use. Finally, we will delve into relevance feedback as a significant part of our research. The powerful adaptive learning - strategy is crucial in content-based image retrieval (CBIR) systems. This exploration could potentially enhance the efficiency and effectiveness of CBIR systems, leading to improved user experiences and more accurate image retrieval (Salton & Buckley, 1990; Zhou & Huang, 2003; Doulamis & Doulamis, 2006).

CRediT authorship contribution statement

Sirawan Phiphitphatphaisit: Conceptualization, Methodology, Software, Validation, Formal analysis, Visualization, Writing – original draft. **Olarik Surinta:** Supervision, Conceptualization, Methodology, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Olarik Surinta reports financial support was provided by Faculty of Informatics, Mahasarakham University, Thailand.

Data availability

I used public food image datasets that can be downloaded online.

Acknowledgment

This research project was financially supported by Faculty of Informatics, Mahasarakham University, Thailand.

References

Aggarwal, U., Popescu, A., & Hudelot, C. (2020). Active learning for imbalanced datasets. In *the IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 1428–1437). https://doi.org/10.1109/WACV45572.2020.9093475

Aguilar, E., Bolaños, M., & Radeva, P. (2017a). Exploring food detection using CNNs. In *the 16th International Conference on Computer Aided Systems Theory (EUROCAST)*, 339–347. doi: 10.1007/978-3-319-74727-9_40.

Aguilar, E., Bolaños, M., & Radeva, P. (2017b). Food recognition using fusion of classifiers based on CNNs. In *the 21th International Conference on Image Analysis and Processing (ICIAP)*, 1–12. doi: 10.1007/978-3-319-68548-9_20.

Bossard, L., & Gool, L. Van. (2014). Food-101 – Mining discriminative components with random forests. In *the European Conference on Computer Vision (ECCV)*, 446–461. doi: 10.1007/978-3-319-10599-4_29.

Branikas, E., Papastergiou, T., Zacharaki, E., & Megalooikonomou, V. (2019). Instance selection techniques for multiple instance classification. In *the 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, 1–7. doi: 10.1109/IISA.2019.8900679.

Bolanos, M., & Radeva, P. (2016). Simultaneous food localization and recognition. In 23rd International Conference on Pattern Recognition (ICPR), 3140–3145. https://doi.org/10.1109/ICPR.2016.7900117.

Cerar, G., Bertalanič, B., & Fortuna, C. (2023). Resource-aware deep learning for wireless fingerprinting localization. In S. Tiku, & S. Pasricha (Eds.), *Machine Learning for Indoor Localization and Navigation* (pp. 437–490). Cham: Springer. https://doi.org/10.1007/978-3-031-26712-3_20.

Dong, T., Sun, Y., & Zhang, F. (2019). A diet control and fitness assistant application using deep learning-based image classification. In *the 8th International Conference on Natural Language Processing (NLP)*, 63–98. doi: 10.5121/csit.2019.91207.

Doulamis, N., & Doulamis, A. (2006). Evaluation of relevance feedback schemes in content-based in retrieval systems. *Signal Processing: Image Communication, 21*(4), 334–357. https://doi.org/10.1016/j.image.2005.11.006

Ege, T., & Yanai, K. (2017). Estimating food calories for multiple-dish food photos. In *the 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, 646–651. doi: 10.1109/ACPR.2017.145.

Farooq, M., & Sazonov, E. (2017). Feature extraction using deep learning for food type recognition. In *the International Conference on Bioinformatics and Biomedical Engineering (IWBBIO)*, 464–472. doi: 10.1007/978-3-319-56148-6_41.

Fränti, P., & Mariescu-Istodor, R. (2023). Soft precision and recall. *Pattern Recognition Letters, 167*, 115–121. https://doi.org/10.1016/j.patrec.2023.02.005

Feng, S., Wang, Y., Gong, J., Li, X., & Li, S. (2023). A fine-grained recognition technique for identifying Chinese food images. *Heliyon, 9*(11), e21565.

Hafiz, A. M., & Bhat, G. M. (2020). A survey on instance segmentation: State of the art. *International Journal of Multimedia Information Retrieval, 9*, 171–189. https://doi.org/10.1007/s13735-020-00195-x

Hassannejad, H., Matrella, G., Ciampolini, P., De Munari, I., Mordonini, M., & Cagnoni, S. (2016). Food image recognition using very deep convolutional networks. In *the 2nd International Workshop on Multimedia Assisted Dietary Management (MADiMa)*, 41–49. doi: 10.1145/2986035.2986042.

He, K., Zhang, X., Ren, S., & J., S. (2016). Deep residual learning for image recognition. In *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. doi: 10.1109/CVPR.2016.90.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *the 32nd International Conference on International Conference on Machine Learning (ICML)*, 448–456.

Jiang, L., Qiu, B., Liu, X., Huang, C., & Lin, K. (2020). DeepFood: Food image analysis and dietary assessment via deep model. *IEEE Access, 8*, 47477–47489. https://doi.org/10.1109/ACCESS.2020.2973625

Kawano, Y., & Yanai, K. (2014). Food image recognition with deep convolutional features. In *the ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication (UbiComp)*, 589–593. doi: 10.1145/2638728.2641339.

Kumar, V., Namboodiri, A., & Jawahar, C. V. (2020). region pooling with adaptive feature fusion for end-to-end person recognition. In *the IEEE Winter Conference on Applications of Computer Vision (WACV),* 2122–2131. doi: 10.1109/WACV45572.2020.9093631.

Kunhoth, J., Maadeed, S. A., Saleh, M., & Akbari, Y. (2023). CNN feature and classifier fusion on novel transformed image dataset for dysgraphia diagnosis in children. *Expert Systems with Applications, 231*, Article 120740. https://doi.org/10.1016/j.eswa.2023.120740

Li, X., Li, W., Ren, D., Zhang, H., Wang, M., & Zuo, W. (2020). Enhanced blind face restoration with multi-exemplar images and adaptive spatial feature fusion. In *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2703–2712. doi: 10.1109/CVPR42600.2020.00278.

Lin, M., Chen, Q., & Yan, S. (2014). Network in network. *arXiv*, arXiv:1312.4400v3, 1–10. https://arxiv.org/abs/1312.4400.

Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., & Ma, Y. (2016). DeepFood: Deep learning-based food image recognition for computer-aided dietary assessment. In *the Inclusive Smart Cities and Digital Health (ICOST)*, 37–48. doi: 10.1007/978-3-319-39601-9_4.

Mao, R., He, J., Shao, Z., Yarlagadda, S.K., & Zhu, F. (2021). Visual aware hierarchy based food recognition. In *the International Conference on Pattern Recognition (ICPR),* 571–598. doi: 10.1007/978-3-030-68821-9_47.

Malekipirbazari, M., Aksakalli, V., Shafqat, W., & Eberhard, A. (2021). Performance comparison of feature selection and extraction methods with random instance selection. *Expert Systems with Applications, 179*, Article 115072. https://doi.org/10.1016/j.eswa.2021.115072

Martinel, N., Foresti, G. L., & Micheloni, C. (2018). Wide-slice residual networks for food recognition. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 567–576. doi: 10.1109/WACV.2018.00068.

Matsuda, Y., & Yanai, K. (2012). Multiple-food recognition considering co-occurrence employing manifold ranking. In *the 21st International Conference on Pattern Recognition (ICPR)*, 2017–2020. https://ieeexplore.ieee.org/document/6460555.

McAllister, P., Zheng, H., Bond, R., & Moorhead, A. (2018). Combining deep residual neural network features with supervised machine learning algorithms to classify diverse food image datasets. *Computers in Biology and Medicine, 95*, 217–233. https://doi.org/10.1016/j.compbiomed.2018.02.008

Nair, V., & Hinton, G. E. (2010). rectified linear units improve restricted Boltzmann machines. In *the 27th International Conference on International Conference on Machine Learning (ICML)*, 807-814.

Ng, Y. Sen, Xue, W., Wang, W., & Qi, P. (2019). Convolutional neural networks for food image recognition: An experimental study. In *the 5th International Workshop on Multimedia Assisted Dietary Management (MADiMa)*, 33–41. doi: 10.1145/3347448.3357168.

Nordin, M. J., Xin, O. W., & Aziz, N. (2019). Food image recognition for price calculation using convolutional neural network. In *the 3rd International Conference on Digital Signal Processing (ICDSP)*, 80–85. doi: 10.1145/3316551.3316557.

Özdemir, A., Polat, K., & Alhudhaif, A. (2021). Classification of imbalanced hyperspectral images using SMOTE-based deep learning methods. *Expert Systems with Applications, 178*, Article 114986. https://doi.org/10.1016/j.eswa.2021.114986

Pandey, P., Deepthi, A., Mandal, B., & Puhan, N. B. (2017). FoodNet: Recognizing foods using ensemble of deep networks. *IEEE Signal Processing Letters, 24*(12), 1758–1762. https://doi.org/10.1109/LSP.2017.2758862

Pereira-Ferrero, V. H., Valem, L. P., & Pedronette, D. C. G. (2022). Feature augmentation based on manifold ranking and LSTM for image classification. *Expert Systems with Applications, 213*(Part B), Article 118995. https://doi.org/10.1016/j.eswa.2022.118995

Phiphiphatphaisit, S., & Surinta, O. (2020). Food image classification with improved MobileNet architecture and data augmentation. In *the 3rd International Conference on Information Science and Systems (ICISS)*, 51–56. doi: 10.1145/3388176.3388179.

Phiphitphatphaisit, S., & Surinta, O. (2021). Deep feature extraction technique based on Conv1D and LSTM network for food image recognition. *Engineering and Applied Science Research, 48*(5), 581–592. https://doi.org/10.14456/easr.2021.60

Prabhakar, S. K., & Lee, S.-W. (2022). Improved sparse representation based robust hybrid feature extraction models with transfer and deep learning for EEG classification. *Expert Systems with Applications, 198*, Article 116783. https://doi.org/10.1016/j.eswa.2022.116783

Rodriguez-Martinez, I., Ursua-Medrano, P., Fernandez, J., Takáč, Z., & Bustince, H. (2024). A study on the suitability of different pooling operators for convolutional neural networks in the prediction of COVID-19 through chest X-ray image analysis. *Expert Systems with Applications, 235*, Article 121162. https://doi.org/10.1016/j.eswa.2023.121162

Ragusa, F., Tomaselli, V., Furnari, A., Battiato, S., & Farinella, G. (2016). Food vs non-food classification. In *the 2nd International Workshop on Multimedia Assisted Dietary Management (MADiMa)*, 77–81. doi: 10.1145/2986035.2986041.

Sahoo, D., Hao, W., Ke, S., Wu, X., Le, H., Achananuparp, P., Lim, E., & Hoi, S. C. (2019). FoodAI: Food image recognition via deep learning for smart food logging. In *the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, 2260–2268. doi: 10.1145/3292500.3330734.

Salton, G., & Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science, 41*(4), 288–297. https://doi.org/10.1002/(SICI)1097-4571(199006)41:4%3C288::AID-ASI8%3E3.0.CO;2-H

Şengür, A., Akbulut, Y., & Budak, Ü (2019). Food image classification with deep features. In *the International Artificial Intelligence and Data Processing Symposium (IDAP)*, 1–6. doi: 10.1109/IDAP.2019.8875946.

Singla, A., Yuan, L., & Ebrahimi, T. (2016). Food/non-food image classification and food categorization using pre-trained GoogLeNet model. In *the 2nd International Workshop on Multimedia Assisted Dietary Management (MADiMa)*, 3–13. https://doi.org/10.1145/2986035.2986039.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15*(1), 1929–1958. https://jmlr.org/papers/v15/srivastava14a.html.

Tasci, E. (2020). Voting combinations-based ensemble of fine-tuned convolutional neural networks for food image recognition. *Multimedia Tools and Applications, 79*, 30397–30418. https://doi.org/10.1007/s11042-020-09486-1

van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review, 53*, 5929–5955. https://doi.org/10.1007/s10462-020-09838-1

Vijayakumar, D. S., & Sneha, M. (2021). Low cost Covid-19 preliminary diagnosis utilizing cough samples and keenly intellective deep learning approaches. *Alexandria Engineering Journal, 60*(1), 549–557. https://doi.org/10.1016/j.aej.2020.09.032

Wang, R., Zhang, W., Ding, J., Xia, M., Wang, M., Rao, Y., & Jiang, Z. (2021). Deep neural network compression for plant disease recognition. *Symmetry, 13*(10), 1769. https://doi.org/10.3390/sym13101769

Xia, K., Huang, J., & Wang, H. (2020). LSTM-CNN architecture for human activity recognition. *IEEE Access, 8*, 56855–56866. https://doi.org/10.1109/ACCESS.2020.2982225

Yanai, K., & Kawano, Y. (2015). Food image recognition using deep convolutional network with pre-training and fine-tuning. In *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 1–6. doi: 10.1109/ICMEW.2015.7169816.

Ye, W., Zhang, W., Lei, W., Zhang, W., Chen, X., & Wang, Y. (2023). Remote sensing image instance segmentation network with transformer and multi-scale feature representation. *Expert Systems with Applications., 234*, Article 121007. https://doi.org/10.1016/j.eswa.2023.121007

Zhang, Y., Deng, L., Zhu, H., Wang, W., Ren, Z., Zhou, Q., Lu, S., Sun, S., Zhu, Z., Gorriz, J. M., & Wang, S. (2023). Deep learning in food category recognition. *Information Fusion, 98*, Article 101859. https://doi.org/10.1016/j.inffus.2023.101859

Zhao, S., Xu, T., Wu, X. J., & Zhu, X. F. (2021). Adaptive feature fusion for visual object tracking. *Pattern Recognition, 111*, Article 107679. https://doi.org/10.1016/j.patcog.2020.107679

Zhou, X. S., & Huang, T. S. (2003). Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems, 8*, 536–544. https://doi.org/10.1007/s00530-002-0070-3