J. Parallel Distrib. Comput. 117 (2018) 218–227

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

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

Fast auto-clean CNN model for online prediction of food materials

Hao Chen, Jianglong Xu, Guangyi Xiao *, Qi Wu, Shiqin Zhang

College of Computer Science and Electronic Engineering, Hunan University, Changsha, China

HIGHLIGHTS

- A fast auto-clean CNN model for online prediction of food materials is proposed.
- The model relies on adapting learning of auto-clean task and multiclass prediction task.
- The time cost of combined two tasks constraint with high precision should be fast for online prediction.

ARTICLE INFO

Article history:
Received 30 April 2017
Received in revised form 19 June 2017
Accepted 16 July 2017
Available online 29 July 2017

Keywords:
Food image detection
Multi-class prediction
Fast auto-clean
CNN model

ABSTRACT

Online food image detection is a key issue for intelligent food materials receiving and food supply chain applications, how to efficiently, accurately and quickly detect the image of food materials is a challenging research topic. A fast auto-clean convolutional neural network (CNN) model for online prediction of food materials is proposed, which is aiming at problems of the complex characteristics of the food images such as the complexity of the food materials, the focus of the dislocation and the uniformity of illumination. Firstly, a new approach of the auto-clean CNN models is proposed for automatic image cleaning and classification, which starting from original images and ending with multi-class prediction of clean images. Given a vocabulary of K classes, and a *Yes/No* clean label, two CNN models will learn a class label and a clean label respectively. Secondly, after the forward pass of two CNN models, the joint features generated from the last convolutional layers will be fed into our two loss layers. Combined with multi-class classification method, it classifies and optimizes the image dataset intelligently. Finally, an online prediction algorithm is proposed to improve the image recognition efficiency. Experimental results show that the proposed model and algorithm have good efficiency and accuracy, and the results of this study have significance to optimize the efficiency of the food supply chain industry and food quality evaluation.

1. Introduction

Consumers always expect and demand high quality and safety food products, leading to the introduction of food safety legislation and food mandatory testing. In order to ensure the safe production of food during processing operations, as well as the correct labeling of products related to quality, safety, authenticity and compliance. It is important for the food industry to develop an accurate, fast and objective quality inspection system throughout the food process. For example, the weighing of food materials in supermarkets, the supply of food materials in restaurants and so on. Currently, it is widely used in human visual inspection to evaluate food products, which however is laborious, tedious, costly and subjective. Increased demands for accuracy, reliability, efficiency and objectivity required the introduction of computer-based image processing techniques, which include charge-coupled device camera [32], magnetic resonance imaging [43], ultrasound [27], electrical tomography [38] and computed tomography [25] for image acquisition; pixel [49] and local pre-processing approaches [33] for image pre-processing; thresholding-based [14], region-based [34], gradient-based [35], and classification-based methods [21] for image segmentation; size, color, shape, and texture features for object measurement [15,31,39]; and fuzzy logic [22], statistical [6], and neural network methods [46,52] for classification. Image processing systems provide application flexibility and can be a reasonable alternative to the human visual decision-making process.

At present, there is no datasets for food material images, and for these massive amounts of data, there is a need for an efficient and fast identification method to identify the food material in the picture. Traditional methods based on principal component analysis [5] and local binary features [36] are limited by hardware performance and cannot cope with massive data and have large error in recognition effect. On the contrary, the concept of Deep Learning (DL) is through the study of massive data, to imitate the

* Corresponding author.
E-mail address: gyxiao@hnu.edu.cn (G. Xiao).

http://dx.doi.org/10.1016/j.jpdc.2017.07.004

Clean: Yes
Category:Carrot

Clean: No
Category:Cabbage

Fig. 1. Illustration of online prediction of food materials. Examples from our online food material supply system. Yes/No indicates the clean/dirty of the image of food material. Label of category indicates a correct label from 100 food materials.

establishment of the human brain's visual neural network, which coincides with the large data processing. However, there is no relevant research on real-time classification [26], so the use of DL to identify the food material in the picture based on the cloud of food material supply system, both for commercial applications or scientific research has a real value.

Recently, we proposed a fast auto-clean convolutional neural network (CNN) model for online prediction of food materials, and collected the datasets from the existing camera with an intelligent scale, and then we proposed that a user could take a picture of a food material by an intelligent scale with a camera and then upload the image in a simple User Interface (UI). The user provided data would then be sent into the cloud to be passed through a pre-trained hierarchically structured CNN. The application UI would then return images, labels and names of the Top-5 likely predicted category, through this method to shorten the time of online classification. Fig. 1 shows that two examples from our online food material supply system, where both images have different labels, indicates the clean or dirty of the food material images, and the label of category indicate a correct label from food materials. The user would then be able to visually compare the target category in front of it with the 5 labels to help positively identify the food material.

This research explores the technical feasibility of high-resolution classification for food materials from a variety of categories. Once designed, built and trained, our predictive system could serve as a prototype 'back-end' for a deployable food material classification application. The goal of this project is to experiment with applying CNN techniques to this particularly challenging datasets we collected. Specific challenges include these points, first of all, the food material is sorted in real time. Secondly, the datasets are difficult to clean, because the quality of the images are quite different, such as light, shooting location, food material stacking and so on. Then, the classification of food materials, such as shooting the carrot, if the picture is clean, the display should be shown this image clean and the category of carrot, otherwise, the display prompts the user to take a picture again. Finally, the multi-tasking model is used to improve the efficiency of classification and accuracy. Our approach borrows heavily from Krizhevsky et al.'s 2012 work on the 8-layer CaffeNet network. They achieved state of the art results on the Imagenet-1000 dataset on 2012 (37.5% Top-1, 17.0% Top-5 misclassification errors) [23].

The rest of the paper is organized as follows. Section 2 introduces related work in online prediction for food supply chain, attributes and multi-labeling, multi-task learning and deep CNN research. Section 3 presents the proposed the fast auto-clean CNN models for food material images recognition. Section 4 describes the collection of datasets, the implementation details and the evaluation results of our proposed algorithms. We make concluding remarks in Section 5.

2. Related work

Since this work is mainly related to the topics of Online Prediction for Food Supply Chain, Attributes and Multi-labeling, Multi-task learning and Deep CNN, we briefly review the most recent literature on these approaches including the following.

2.1. Online prediction for food supply chain

Image Processing Techniques for Food Materials: The traditional food materials image recognition algorithms have many defects [13,30,47]. Bolle et al. [3] used the extracted image texture statistics, color and other characteristics, for the first time to achieve the classification of food materials identification, and can be any number of free and placed, and developed an intelligent food materials recognition system (Veggie Vision). But the fruit and vegetables on the "black box", are vulnerable to the impact of light. To alleviate this, Zhang and Wu [57] proposed that shape features can be used in the recognition feature, and the use of support vector machine (SVM) to identify food materials. Tao et al. [47] proposed a full local binary pattern texture feature extraction algorithm, using hue, saturation, value (HSV), color histogram and external point/interior point histogram to extract image color characteristics, through the nearest neighbor classifier to achieve the classification of food materials, but the experimental use of the fruits and vegetables only consider the light intensity changes. Paul et al. (2014) proposed a method of three dimensional (3D) color histogram combined with CNN, which classifies the food materials with changes in viewing angle and light. However, the number of experimental images is only one, and most of the viewing angles are less obvious.

Online Prediction for Food Supply Chain: online prediction may help in resolving the food safety problem because it can offer more agile and more convenient management of merchandise, including foods [29,37,50]. In the work [50], a framework is developed that evaluates supermarket food safety and exposes its relative strengths and weaknesses. It can assist managers in comprehending the present strengths and weaknesses of their food safety. Pang et al. [37] proposed a three-tier information fusion architecture, which can speed up data processing, self-learning shelf life forecasting and re-planning real-time supply chain. Liu et al. [29] introduced a pilot project in China: the Internet of Agricultural Things (AIoT for short), which integrates state-of-the-art technologies to provide a method to easily track and trace the supply processes of foods.

2.2. Attributes and multi-labeling

Definition of Attribute: a visual property that appears or disappears in an image. Different image features such as colors, textures, and shapes may be described by different properties [12]. Some recent researches have focused on how to link human-interaction applications through these intermediate attributes, where a consistent alignment should occur between computer interpretations of query attribute phrases and the human query expressions.

Global vs. Local Attributes: If the global properties in the image are described, e.g., 'soil celery'. In general, global attributes do not involve specific object parts or locations [4,19,44,55]. Localized attributes are used to describe one or more locations of an object, e.g. 'Little red pepper'. Both types are not easily inferred, because the performance of the undersampling classifier may be reduced if the classifier is trained only on advanced tags that do not have spatial information such as bounded frames. However, some of the work in [19], Zhou et al. [58] show that sharing visual knowledge can counteract the effects of the lack of training samples.

Attributes and Multi-labeling: The image multi-labeling is simply learning to assign multiple labels to an image [10,40,55]. If the problem is adapted as is, challenges will arise when the number of labels increases and the potential combination of output labels becomes difficult to handle [48]. To alleviate this, a common conversion approach is achieved by breaking down the problem into a single set of binary classifiers [42]. Predicting co-occurring attributes can be viewed as multi-label learning. On the other hand, most of the related works [19,55], Sandeep et al. [44] tend to apply multi-task learning to allow a priori to share or use some of the label relational heuristics [10]. Another work is to use the deep CNN ranking function to rank the label scores [17].

2.3. Multi-task learning

Multi-Task Learning (MTL): MTL has recently been applied to computer vision problems, especially when certain tasks are under-sampled [1,7]. MTL is designed to enhance knowledge sharing while addressing multiple related tasks. It has been shown that this sharing can improve the performance of some or sometimes all tasks [7].

Task and Feature Correlations: Many shared strategies have been explored; the first is to design different ways to discover the relationships between tasks [18], the other takes into account a method of finding common features or mine the related features for all task sharing [41]. Recent frameworks such as Max-margin [56], Bayesian [53] and their co-expansion [28], try to discover either or both task and feature relevance. While Max-margin is known by its discriminative power, Bayesian is more flexible and therefore more suitable for any prior or performance reasoning [28]. In contrast to these researches, the work of Gong et al. [16] shows that, in the typical case, the dimensions of the data are high; therefore, it is unreasonable that all tasks should share a set of common features. They solve these assumptions by capturing the shared features in the task at the same time and by identifying the exception values by introducing the exception matrix [16]. In other works, [24,58], the authors further relax the constraint naturally by decomposing the model parameters into shared latent task matrices and linear combinatorial matrices; thus, it encourages all tasks to choose what content to share all the tasks are encouraged to select what to share through this potential matrix to learn more about localization. However, in all these techniques, they rely on the implementation of such a shared fashion by applying various rules to the model parameters.

2.4. Deep CNN

CNN for feature learning: CNN was generated in the DL era [2,8,23]; its goal was to model high-level abstractions of visual data by using multiple non-linear transformation architectures. In the DL model, CNN exhibits extraordinary performance, especially in image classification and object recognition applications [8,23]. The two troublesome questions about training CNN are the number of training images and the time required to complete the training network. This means that to have an effective CNN model, training datasets and time should be large enough for the CNN to gain the ability to perform its task well [23,46]. The learning features generated by CNN have proven to be powerful, generic, and more efficient than hand-crafted features [11,45]. Fortunately, the popular implementation of Donahue et al. [11], Krizhevsky et al. [23] and the use of pre-trained CNN models on the ImageNet dataset make it easier to fine tune the various CNN architectures of many visual datasets.

CNN between single and multi-labels: using CNN for single label prediction is intensively studied [2,23]. Multi-labels have many challenges, as described in the ''Attributes and Multi-labeling'' section. Therefore, training CNN directly is infeasible and impractical. However, a recent work has proposed a solution for multi-label issues [51]. In this work, the shared CNN is fed with any number of image batches, which are extracted or generated by some techniques, such as the binary specification gradient [9]. All of these hypothetical final CNN outputs are aggregated by maximum summation to give the final format for multi-labels prediction. Unlike their approach, our proposed model is to annotate the essence of an image with multi-labels through a multi-task CNN model, which is trained simultaneously by MTL to allow sharing of visual knowledge. Sharif Razavian et al. [45] proposed another direction of multi-labels, where CNN was mainly used to generate shelf activity characteristics; then they applied SVM to subsequent categories. In our approach, when the number of attributes is large, we fine-tune many CNN models, each of which is dedicated to learning the attribute-specific representation. These representations are used as off-the-shelf features of the late stages of MTL because we freeze their training while optimizing the multi-task loss function.

Compared with the previous method, we proposed a method to train the multi-task classifier model on the deep feature of attribute prediction and use the shareable latent task matrix, which can be very helpful in generating the complete input image in terms of attributes description. Exploring the importance of this potential matrix is a topic of interest for the future.

3. Fast auto-clean CNN model

In this section, we will explain the details of the proposed approach of the auto-clean CNN models. Fig. 2 shows the overall structure of the proposed approach, starting from raw images and ending with multi-class prediction of clean images. Given a vocabulary of K classes, and a *Yes/No* clean label, two tasks CNN model will learn a class label and a clean label respectively. After the forward pass of two tasks CNN model, the joint features generated from the last convolutional layers and first full connect layer will be fed into our two loss layers. Cleanly, the weight parameters matrix learned from two loss layers will be decomposed into bottom shared layers. Meanwhile, the combination matrix of bottom layers till the first fully connected layer has different information of two tasks CNN model.

This model can also be viewed as the specific softmax loss of clean task with a fully connected layer and a softmax loss of multiclass with two fully connected layer shared the same bottom layers. After optimizing the joint loss layer and sharing the visual features, two tasks CNN model will propagate back their specific parameters in the backward pass. By well collected images that are annotated as clean/dirty label and a multiclass label, we iteratively train the whole network and parameters until convergence.

We used the famous network structure proposed by Krizhevsky [23], which consists of 5 convolutions, followed by 2 fully-connected layers and finally the softmax and the loss. In addition, some pooling, normalization, and ReLU are applied between some of these layers. Many works have studied and analyzed the nature of this structure and identified some import aspects. For example, the work in [54] shows that the parameters of the fully-connected layers occupy almost 70% of the total network capacity, which consumes a great deal of error while training the network. However, given the expense of trading between 'good-but-fast' and 'perfect-but-slow', the work in [2] shows that the performance will drop slightly when removing the fully-connected layers. Because our model requires two tasks CNN model, for the clean CNN model we remove the last fully connected layer, in order to make fast the clean and multiclass prediction, we connect the first fully-connected layer with the softmax layer directly, depending on the weight parameter matrix learned with.

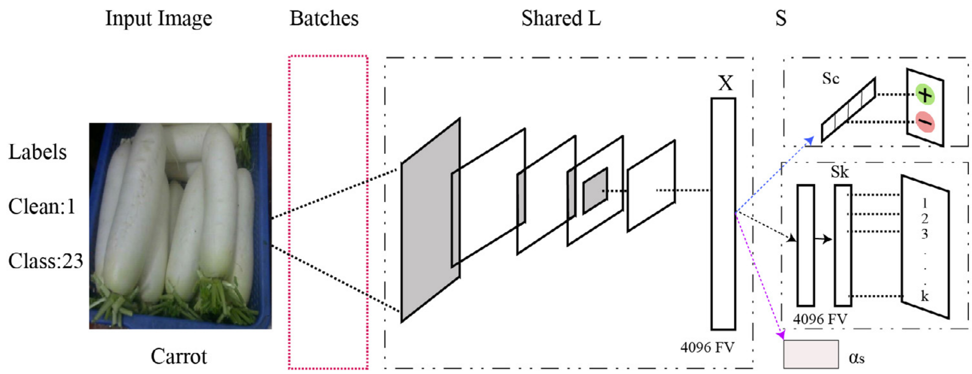


Fig. 2. Fast Auto-clean CNN models: the input image (in the left) with two labels information is fed into the model. Two loss layer will predict a binary clean value and a multiclass label. The shared layers L together with S layer from a weight matrix of the first fully connected layer followed by a softmax or a second fully connected layer followed by a softmax. The L layers are shared matrices between two tasks CNN model. The layers in S are task-specific weight matrix layers. Two labels information about clean or multiclass is utilized during the training of the network.

In the following subsection, we demonstrate the shared bottom layers, and we show how the feature sharing and completion is engaged. Then, we introduce our formulations, which we use to solve the clean and prediction tasks. Next, the total optimization procedure and adaptive weight learning used to train the whole network of fast auto-clean CNN model is described. Finally, the online prediction algorithm of applying auto-clean CNN model is described.

3.1. Sharing the bottom layers in fast auto-clean CNN learning

Given two labels, the goal is to learn a binary linear classifier for clean task and a multi-linear classifier for multiclass. Each classifier or task has model parameters, which we denoted by w^m and re dedicated to predict the corresponding labels. W is the total classifier weights matrix, which can also be considered a softmax weight matrix but stacked from all CNN softmax layers. Given N training images, each of them has a two labels C and K, such that C_m is either $\{1\}$ or $\{0\}$ indicating whether a training image i is clean or dirty, and L_m is in range of $\{0, \dots, |K|\}$ indicating the specific class of food materials, where K is total number of food material types. Suppose that the output from the last convolution layer in each CNN model forms an input feature vector, such that two CNN models will generate two-tasks training pool. Thus, we will have ${X_M}^{K+2}$ training examples aggregated from two CNN models.

Our assumption is inspired from the work of Zhou et al. [58], where each classifier can be reconstructed from two shared tasks and linear combination of these two tasks. Through this decomposition, simultaneous CNN models can share similar visual patterns, which learns more localized features. We denote L to be bottom layers till the first fully-connected layer, a binary softmax S_b, a second fully-connected layer F_c, and a multiclass softmax S_K.

Now, we have three matrices W_{Bf}, W_{Bb}, and W_{fK} for connecting (L, F_c), (L, S_b), (F_c, S_K) respectively, where b is a binary label of clean or dirty and K is a multiclass label of food materials.

Given two CNN models, we aim to learn the matrices of W_{Bf}, W_{Bb}, and W_{fK}, which is formed by stacking the parameter metrics of the softmax layers of two CNN. The key idea behind our model is the share L layers will fast our two predict tasks. By this shared L layers, two CNN can share visual patterns with the other CNN model, and two CNN models can collaborate together in the training stage to optimize this shared layer. Each CNN predicts whether the image contains the corresponding label. The benefit of learning the shared bottom layers through two-tasks is that each CNN can leverage the visual knowledge from learning other CNN models even if its training samples are not enough.

3.2. Formulations of the fast auto-clean CNN model

Given a set of N training images and their labels: $D = \{I_i, C_i, K_i\}_{i=1}^{N}$, where each binary label C_i is a bit indicating clean or dirty, and a class label L_i indicates the class of food material. As shown in Fig. 2, the proposed Auto-clean CNN extracts a high-level feature representation $x_i \in \mathbb{R}^{D_b \times 1}$

$$x_i = f\left(I_i, k, b, \beta, \gamma\right) \tag{1}$$

where $f(\cdot)$ represents the non-linear mapping from the input image to the shared features in first fully-connected layer. k and b are sets of filters and bias of all convolutional layers. β and γ are the sets of scales and shifts in all bottom layers. Let $\Theta = (k, b, \beta, \gamma)$ denote all parameters to be learned to extract the features.

The extracted features x_i, which is first fully-connected layer in our model, are shared among two tasks such as image clean and multiclass prediction. Ignored another fully-connected layer in our model, suppose $W^d \in \mathbb{R}^{D_b \times D_d}$ and $b^k \in \mathbb{R}^{D_d \times 1}$ are the weight matrix and bias vector in the fully connected layer for food material classification, where D_d is the number of different food material in D. The generalized linear model is as follows:

$$y_i^d = W^d . Tx_i + b^d \tag{2}$$

y_i^d is fed to softmax layer to compute the probability of x_i belonging to each food material in the training set.

$$softmax\left(y_i^d\right)_{K_n} = p\left(y_i^d = K_n | x_i\right) = \frac{\exp\left(y_{iK_i}^d\right)}{\sum_j \exp\left(y_{iK_j}^d\right)} \tag{3}$$

where y_{ij}^d is the jth food material in y_i^d. The $softmax(\cdot)$ function maps the model output y_i^d to the probability distribution over all food materials and the index selects the K_ith food material. Finally, the estimated food material $\hat{y}_i^d$ is obtained through:

$$\hat{y}_i^d = \underset{K_n}{argmax}\ softmax\left(y_i^d\right)_{K_n}. \tag{4}$$

Then the cross-entropy loss can be employed:

$$L\left(I_i, y_i^d\right) = -\log\left(p\left(\hat{y}_i^d = y_i^d | I_i, \Theta, W^d, b^d\right)\right). \tag{5}$$

Similarly, we formulate the losses of the clean task. Let $W = \left\{W^d, W^c, W^f\right\}$ represent the weight matrices for food material class, image clean, and the connection of two fully connected layers. The bias terms are eliminated for simplicity. Given the training dataset D, our Auto-clean CNN aims to minimize the combined loss of two tasks:

$$\arg\min_{\Theta,W} \sum_{i=1}^{N} \alpha_d L\left(I_i, y_i^d\right) + \alpha_c \sum_{i=1}^{N} L\left(I_i, y_i^c\right) \tag{6}$$

where α_d and α_c control the importance of two tasks respectively. It becomes a single-task model (s-CNN) when either α_d or α_c is 0. The loss drives the model to learn both the parameters Θ for extracting the shared features and W for the two classification tasks. In the testing stage, we first test whether the input image is clean or dirty. If we predict the image is dirty, the system will capture a new image to process until we find a clean image. After that the clean image will also get the predict class of food material simultaneously since the parallel predict model from our Fast Auto-clean CNN model.

3.3. Adapted-weighting learning

In CNN-based multi-task learning, it is an open question on how to set loss weight for these two tasks. Prior work either treat all tasks equally or obtain weights via brute-force search. However, it is very time consuming to search for all weight combinations. Our automatically assign adapted weighting to two tasks is as follows.

Our fast Auto-clean CNN learns to allocate α_d and α_c with a vector of α_s to two tasks. As shown in Fig. 2, we add a softmax layer from first fully-connected layer. Let $\omega_s \in \mathbb{R}^{D_b \times 2}$ and $\varepsilon_s \in \mathbb{R}^{2\times 1}$ indicate the weight matrix and bias vector in the weighting softmax layer.

$$\alpha_s = softmax\left(\omega_s.Tx_i + \varepsilon_s\right) \tag{7}$$

where $\alpha_s = [\alpha_d, \alpha_c].T$ are the adapted loss weights for two tasks with $\alpha_d + \alpha_c = 1$. So Eq. (6) becomes:

$$\begin{aligned} \arg\min_{\Theta,W,\alpha_s} \quad & \sum_{i=1}^{N} \alpha_d L\left(I_i, y_i^d\right) + \alpha_c \sum_{i=1}^{N} L\left(I_i, y_i^c\right) \\ & \text{s.t. } \alpha_d + \alpha_c = 1. \end{aligned} \tag{8}$$

We use mini-batch stochastic gradient descent to solve the above optimization problem where the adapted weights are converged over the batch of samples.

3.4. Online predication algorithm

During the online prediction phrase, the procedure of prediction is descripted in *Algorithm 1*. The input of *Algorithm 1* is the model of our fast auto-clean model with optimized parameter learned from training phrase. When a food material is loaded in our system, the prediction procedure is as algorithm 1, the output of the algorithm is an integer number k. If k is -1, that indicates the online prediction system cannot capture a clean image for the food material to recognize. If k is in range of $\{0, \ldots, K-1\}$, that indicates the online prediction system has already recognized the image as a specific class of food material with the index k.

Algorithm 1: Online Prediction Algorithm

```
Input--     M: fast auto-clean CNN model with optimized parameters
Output --   k: an integer
            value of -1 denotes can't capture a clean image
            values in range of {0,…,K-1} denotes the class index of a food material
for max_iterations do:
    I = CaptureNewImage()
    b, k = Predict(M,I)
    if b is True, then return k
    else continue
return -1
```

The procedure of Algorithm is as follows: Step 1: Online prediction algorithm will capture a new image I from the client system. Given a new image I, Step 2: The fast auto-clean CNN model M will predict two variables b and k from two softmax functions. Step 3: If b is true, then this online prediction system will return the class index k to the client system immediately, otherwise go to *Step 1* until we have repeated *max_iteration* times. Step 4: return an error code -1 to the client system to tell user the camera cannot capture a clean image for the food material, and please select the class of food material manually.

This parallel prediction for two tasks is really fast and accurate in practice. For accuracy, since the limited data of training samples, the parameters of shared features can be tuned from the samples of both two tasks which will benefit the accuracy for both two tasks. For time cost, since the task for predicting the clean label is based on the CNN model by subtracting a fully-connect layer, plus the shared bottom features, the extra cost for clean label predicting is very small. The performance of both tasks have the same approximately time cost.

4. Experiments and results

In order to achieve a better recognition accuracy with less time consumption, famous deep learning neural networks like CaffeNet, AlexNet and VGG-16 are used in the first experiment (Single Task Comparison) to figure out which neural network is suitable here. And the second experiment (Auto Clean) is multi task experiment which do auto clean and food materials images recognition tasks at the same time. Both of these two experiments are used to recognize good images from bad ones and classify good images.

4.1. Datasets and environments

The dataset is collected by a large food supply chain platform in China (www.mealcome.com), it includes nearly 1000 restaurants and above 12 000 food supplies, the amount of restaurant and food materials images is keeping increase consistently, and this dataset is called Mealcome dataset (MLC dataset). Our dataset consists of two parts: clean part (MLC-CP) and dirty part (MLC-DP) and we get two parts manually. The original food materials image we acquire are arranged by date and one folder contains all images generated on that day. All the images are shot in the field, so the original images are mixed with dirty images because of overexposure, pose, illumination, plastic cover and so on, and we called these kinds of images bad ones.

Bad images are picked up manually to create the MLC-DP, and all the good images are gathered to create the MLC-CP. We also reorganize the MLC-CP, make sure that one kind of food materials image is in one folder and the folder name is the food material itself.

The dataset used in experiments contains 27 635 images and 41 classes, which are divided into three parts, 60% for training set, 20% for validating set and 20% for testing set. In order to achieve better performance, we set all the image size to 640*480 px.

In these experiments, the proposed algorithm is implemented on a desktop computer with an Intel(R) Core(TM)i5-6500 CPU(3.2 GHz), 8 GB RAM and ROG STRIX-GTX1060 GPU, which can impressively improve the computing speed of experiments. In order to build the basic experimental environment, we installed Ubuntu 16.04.1 Desktop LTS operating system on this computer, then installed Caffe framework as a deep learning framework that have advantage in expression, speed and modularity.

4.2. Experiments setting

Single Task Comparison of CaffeNet, AlexNet and VGG-16 is aimed to determine which neural network is most suitable for online fast food materials recognition and classification. These neural networks are different kinds of neural network, and each

Table 1
Left section is part of the file list for original Caffenet, the content has two columns, column 1 is the image relative path, and column 2 is the food materials index that means which kind of food materials the image belongs to. Right section is the special file list for modified Multi-task Caffenet, the content has three columns, column 1 and column 2 have the same mean as left section, while column 3 mean indicates another type, here 1 represents clean category and 0 represents bad category.

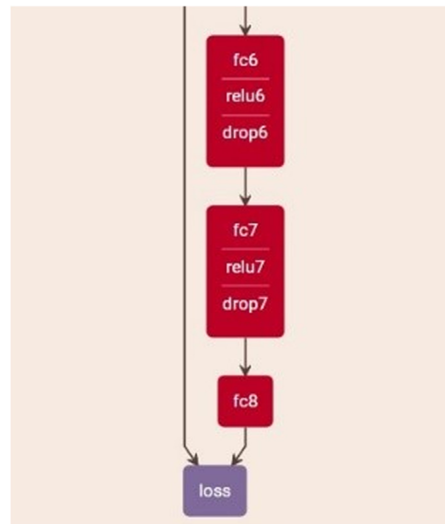
File list for original Caffenet	*File list for modified Multi-task Caffenet*
baiLuoBu/20170320074356439cdc21-o.jpg 0	woSunTou/20170227073949753 86d55-o.jpg 0 1
baoCai/20170211084028872fdb7e-o.jpg 1	xiangCai/201612010820571723c1e9-o.jpg 1 1
benDiZiYu/20161224074717 58917d89-o.jpg 2	jiuCai/20161222080454298320e4-o.jpg 2 1
daHongJiao/20170210083236901f3d57-o.jpg 3	all-others/20170413091331239d8fc0-o.jpg 41 0
……	……

has their own feature. Dataset in this experiment is MLC-CP, which are all the good food materials picked out manually. And for the purpose of validating the performance, we tested different neural networks for classification. In order to get a better experimental result and train the model faster, we use the pre-trained model to fine tune the parameters in these experiments. Hyper-parameter adjustment is also an important part of Single Task Comparison experiment, better iteration number and proper learning rate are fixed in this part. The detailed steps of Single Task Comparison experiment are as follows:

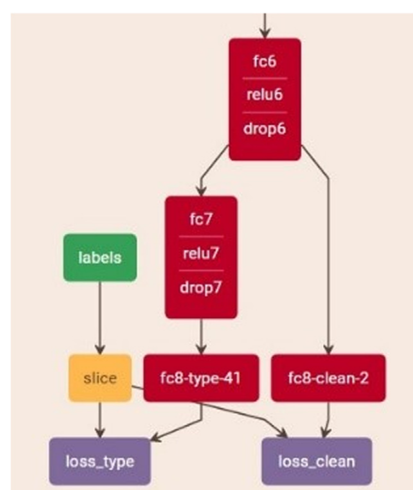
- Step 1: change images to *lmdb* format and calculate the mean file
- Step 2: design our neural network that to be trained.
- Step 3: configure the *train.prototxt* and *solver.prototxt* file and then train the model
- Step 4: choose a trained net with lowest loss and highest accuracy from the model snapshot and then use the testing dataset to test the chosen model.

In step 1, *lmdb* is a kind of image format easily read by Caffe framework, and by doing this, we can improve speed and accuracy in the training and testing process. We set a lot of hyper-parameter in *train.prototxt* file and *solver.prototxt* file at step 2 to get better recognition and use these hyper-parameter to make the comparison experiment.

After the most suitable neural network is chosen and proper parameters are decided, the network will be used to do Auto Clean experiment. Because CaffeNet is the most suitable network according to the experiment results part, we propose an Auto-clean CNN model based on CaffeNet in this experiment. And the dataset used in this are MLC-CP and MLC-DP. For Auto Clean experiment, we train a multi task model based on CaffeNet. Our philosophy is to extract image features layer 1 to layer 6 or 7, and the two tasks both share the same features provided by previous layers. The way MTL-CNN model reading the dataset is totally different because original caffenet which do not support multi task. The critical points of MTL-CNN model are modifications of last layer sharing the same image features, the method of data input and Softmax ratio for two tasks. Fig. 3 shows the difference of original and modified CaffeNet, and Table 1 shows the details of file list which original CaffeNet and Multi-task CaffeNet use for Caffe Framework, file list is an important file for Caffe Framework, by which Caffe can find where the images located and which kind a certain image belongs to.



(a) Original Caffenet.



(b) Modified multi-task Caffenet.

Fig. 3. Our modification of original Caffenet. (a) Original Caffenet fc8 takes advantage of features extracted by fc8. (b) In modified multi task Caffenet, fc7 and fc8-clean-2 both share the features provided by fc6, fc8-type-41 use features supplied by fc.

4.3. Experiments results and analysis

We evaluate the proposed method under three settings: (1) evaluate the different kinds of network performance for our online prediction application based on CaffeNet, AlexNet and VGG-16 neural network, this is a part of Single Task Comparison experiment; (2) optimize hyperparameter upon CaffeNet; (3) evaluate the performance results on double tasks model and single task model, and this is the Auto Clean experiment. We use Caffe [20] with our own modifications.

(1) Performance result on CaffeNet, AlexNet and VGG-16 are shown in Fig. 4. It illustrates the food material image Recognition time and accuracy of CaffeNet, AlexNet and VGG-16. As Fig. 4(a) shows, VGG-16 takes more time than other models. The time consumption of CaffeNet and AlexNet is similar, but CaffeNet only takes less than 300 ms to recognize an image in fact, which is faster than other models to recognize the image. As shown in Fig. 4(b),

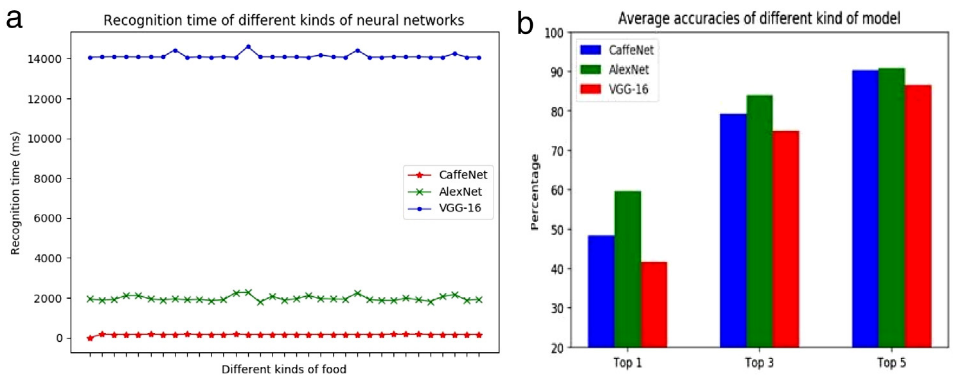


Fig. 4. Recognition time and accuracy of CaffeNet, AlexNet and VGG-16. (a) Recognition time of different kinds of food images using different kinds of neural network. (b) Top1, top3 and top5 food materials recognition accuracies of CaffeNet, AlexNet and VGG-16.

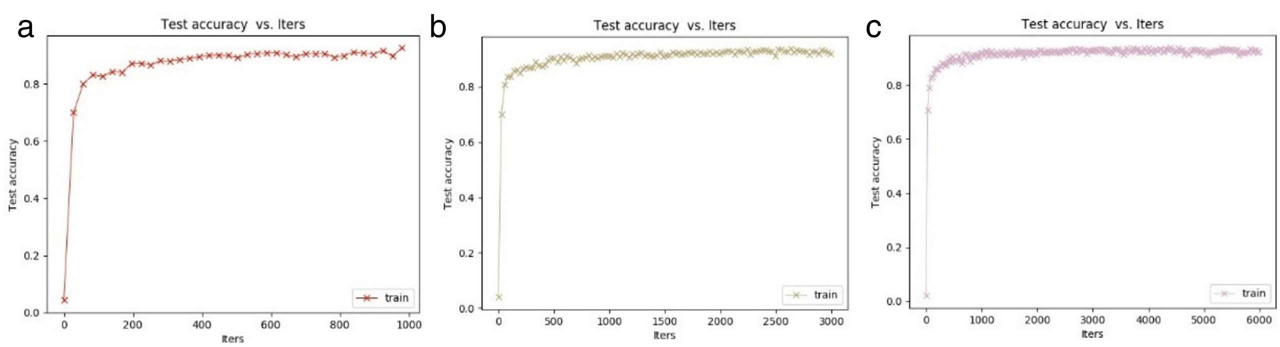


Fig. 5. Different kinds of Iters value upon CaffeNet. (a) Iters value equals 1000, the test accuracy is still increasing. For (b) and (c) iters value is equal to 3000 and 6000, test accuracy do not increase any more, and come close to stability.

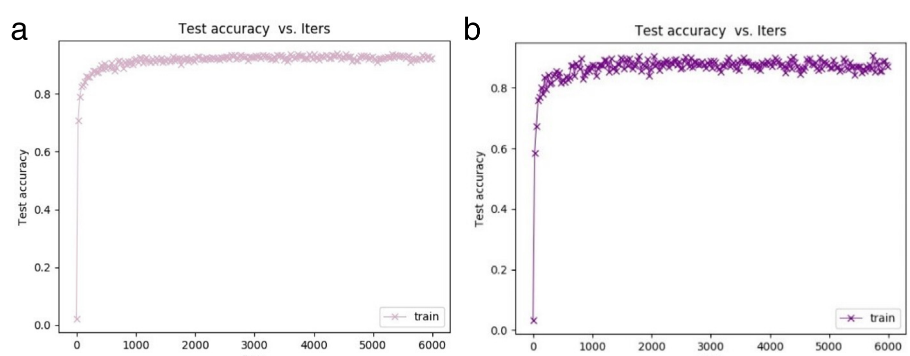


Fig. 6. Iters equals 6000 and the learning rate is adjusted from 0.001 to 0.005. (a) Learning rate equals 0.001, (b) Learning rate equals 0.005. And as shown in the figure, the higher the learning rate is, the more oscillatory the test accuracy will become.

we can see that both CaffeNet and AlexNet have higher average accuracy than VGG-16. When comparing CaffeNet with AlexNet, CaffeNet's top 1 and top 5 are better than AlexNet, while AlexNet's top 3 is slightly higher than CaffeNet, but almost the same.

When recognition time and accuracy are considered globally, CaffeNet is the best choice. And the next experiment part is the adjustment of hyper parameters based on CaffeNet, through which the best iteration number and learning rate will be determined.

(2) Hyper-parameter adjustment upon CaffeNet are shown in Figs. 5 and 6. The iteration number is the first hyper parameter we figure out, as shown in Fig. 5(a–c), when the iteration number increases, the test accuracy becomes more and more stable. 6000 is chosen as the iteration number finally. Then we move to finetune the learning rate, Fig. 6 shows that when iteration number equals 6000 the learning rate varies from 0.001 to 0.005. A lot of different learning rate value have been tried in this part, one phenomenon is the test accuracy will become more oscillatory if learning rate is increased. Fig. 6(a) shows us a more stable test accuracy than that in Fig. 6(b).

After a series of finetuning experiments, iteration number is set 6000 and learning rate is equal to 0.001. In the next Auto Clean experiment, these finetuned parameters will be used and we will move further to analyze how effective the Auto Clean model is.

(3) Auto Clean model is used to finish double tasks at the same time in this experiment, one is to pick out good images from bad ones, and the other is to recognize images. As a comparative experiment, we use two single task models to finish double tasks, which is noted as single task in figures of this part. For the Auto

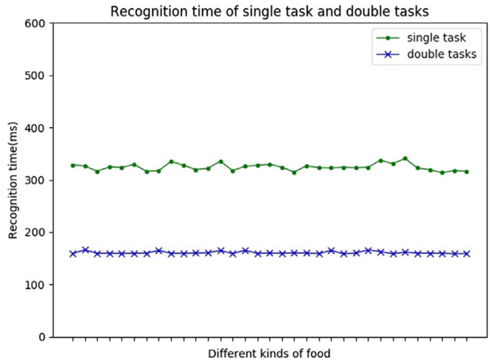


Fig. 7. Recognition time of all kinds food material image on double tasks model and single task model, double tasks auto clean model is better than single task.

Clean model, which is modified based on CaffeNet, both of these two tasks share the same features provided by fc7 layer. And the softmax loss ratio of these two is 0.5:0.5, because these two tasks are of the same importance.

Experimental results of single task and double tasks are shown in Fig. 7. It illustrates the recognition time of all kinds food material image on double tasks model and single task model, and it is very clear that Auto Clean model in double task is pretty better than single task. In addition to time consumption, it also has better accuracy, which is shown in Fig. 8.

5. Conclusion

Online food image detection is a key issue in the application of intelligent food materials applications and food supply chain, how to efficiently, accurately and quickly detect the image of food materials is a challenging research topic. In this study, a fast Auto-Clean CNN model for online prediction of food materials is proposed, which is aiming at problems such as the complex characteristics of the food images, the complexity of the food materials, the focus of the dislocation and the uniformity of illumination. The specific contributions of this paper include:

(1) A new approach of the auto-clean CNN model is proposed for automatic image cleaning, which starting from raw images and ending with multi-class prediction of clean images. Given a vocabulary of K classes, and a Yes/No clean label, two CNN model will learn a class label and a clean label respectively;

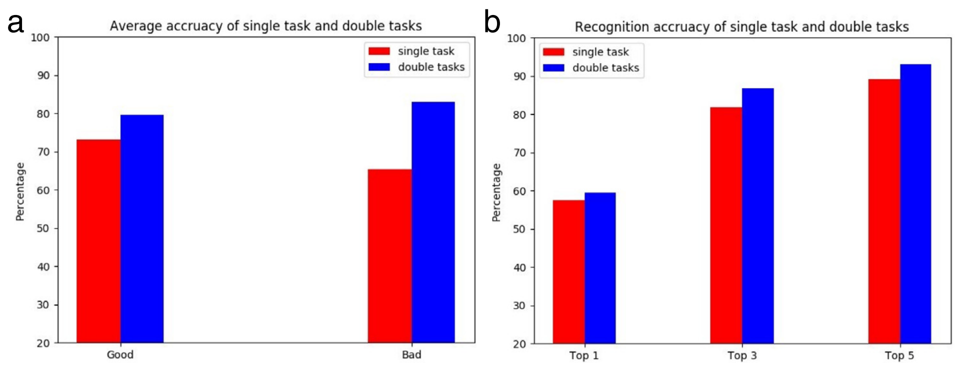


Fig. 8. Good and bad average accuracies of double tasks and single task; and top 1, top 3 and top 5 accuracy of these two models. (a) Good and bad in *x*-axis mean recognize good food materials images as good class and recognize bad food materials images as bad class. (b) Average top1, top3 and top5 accuracy of single task and double task, this is aim to recognize which kind of good images it is.

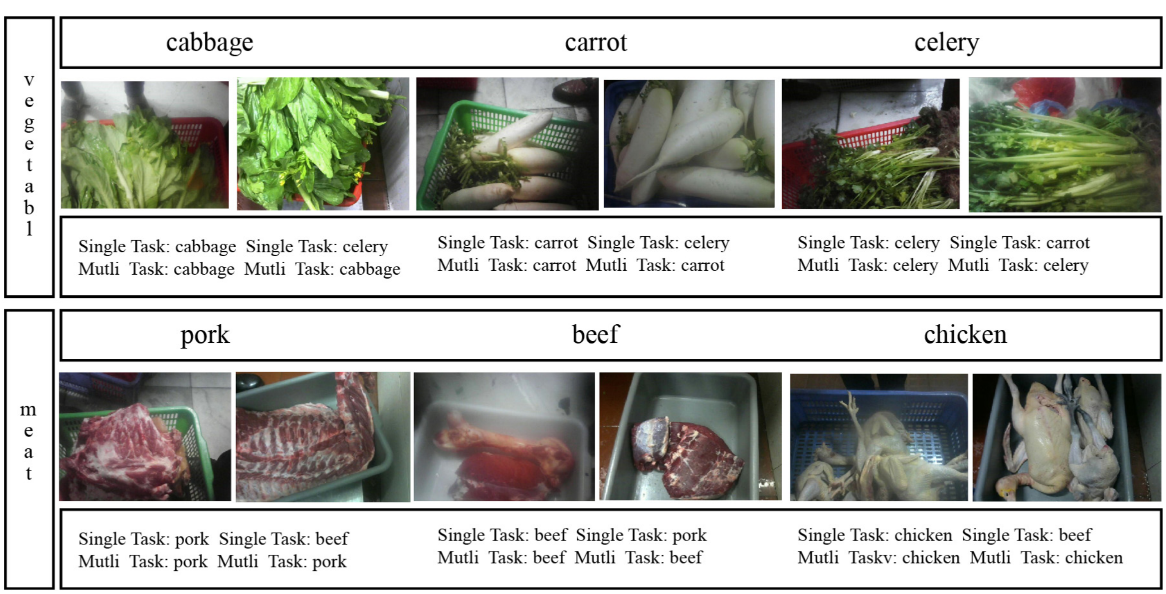


Fig. 9. Examples of misclassified samples from single-task classifiers results. The first row has samples from vegetables. The second row has samples from meats. Yes/No of Clean label indicates whether the image is clean or dirty. Multi-task classifiers are able to correctly classify these samples.

(2) Combined with multi-class classification method, an online prediction algorithm is proposed to improve the image recognition efficiency.

(3) We have verified our model and algorithm in real datasets we collected in our real cloud-based food supply system. Our experimental results show that the proposed model and algorithm have good operation efficiency and accuracy, and the results of this study have significance to the optimization of the efficiency in the food supply chain industry and food quality evaluation. (See Fig. 9.)

Acknowledgments

This work was supported by the National Science Foundation of China (Grant Nos. 61472132, 61472131, 61603420), Natural Science Foundation of Hunan Province (Grant Nos. 2015JJ2027 and 2014GK3028), Science and Technology Key Projects of Hunan Province (2015TP1004, 2016JC2012, 2015SK2087, 2015JC1001, 2016JC2015).

References

[1] A. Argyriou, T. Evgeniou, M. Pontil, Multi-task feature learning, Adv. Neural Inf. Process. Syst. 19 (2007) 41.
[2] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[3] R.M. Bolle, J.H. Connell, N. Haas, R. Mohan, G. Taubin, Veggievision: A produce recognition system, in: Applications of Computer Vision, 1996 WACV'96 Proceedings 3rd IEEE Workshop on, IEEE, 1996, pp. 244–251.
[4] L.D. Bourdev, US Patent No. 9,400,925 Washington, DC: US Patent and Trademark Office, 2016.
[5] R. Bro, A.K. Smilde, Principal component analysis, Anal. Methods 6 (9) (2014) 2812–2831.
[6] G. Camps-Valls, D. Tuia, L. Bruzzone, J.A. Benediktsson, Advances in hyperspectral image classification: Earth monitoring with statistical learning methods, IEEE Signal Process. Mag. 31 (1) (2013) 45–54.
[7] R. Caruana, Multitask Learning, in: Learning to Learn, Springer US, 1998, pp. 95–133.
[8] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: Delving deep into convolutional nets, 2014. arXiv preprint arXiv:1405.3531.
[9] M.M. Cheng, Z. Zhang, W.Y. Lin, P. Torr, BING: Binarized normed gradients for objectness estimation at 300fps, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 3286–3293.
[10] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, et al., Large-scale object classification using label relation graphs, in: European Conference on Computer Vision, Springer International Publishing, 2014, pp. 48–64.
[11] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, DeCAF: A deep convolutional activation feature for generic visual recognition, in: Icml, vol. 32, 2014, pp. 647–655.
[12] A. Farhadi, I. Endres, D. Hoiem, D. Forsyth, Describing objects by their attributes, in: Computer Vision and Pattern Recognition, 2009 CVPR 2009 IEEE Conference on, IEEE, 2009, pp. 1778–1785.
[13] D.A. Freedman, N. Peña Purcell, J.R. Hebert, Extending cancer prevention by improving fruit and vegetable consumption, Int. Fruit Vegetable Alliance Newslett. 96 (2015) 2.
[14] P. Ghamisi, M.S. Couceiro, F.M. Martins, J.A. Benediktsson, Multilevel image segmentation based on fractional-order darwinian particle swarm optimization, IEEE Trans. Geosci. Remote Sens. 52 (5) (2014) 2382–2394.
[15] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.
[16] P. Gong, J. Ye, C. Zhang, Robust multi-task feature learning, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 895–903.
[17] Y. Gong, Y. Jia, T. Leung, A. Toshev, S. Ioffe, Deep convolutional ranking for multilabel image annotation, 2013, arXiv preprint arXiv:1312.4894.
[18] B. Hariharan, L. Zelnik-Manor, M. Varma, S. Vishwanathan, Large scale max-margin multi-label classification with priors, in: Proceedings of the 27th International Conference on Machine Learning, ICML-10, 2010, pp. 423–430.
[19] D. Jayaraman, F. Sha, K. Grauman, Decorrelating semantic visual attributes by resisting the urge to share, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1629–1636.
[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, et al., Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM International Conference on Multimedia, ACM, 2014, pp. 675–678.
[21] S.C. Kheruka, L.M. Aggarwal, N. Sharma, U.C. Naithani, A.K. Maurya, S. Gambhir, Evaluation of single-photon emission computed tomography images obtained with and without copper filter by segmentation, Indian J. Nucl. Med. 31 (2) (2016) 114.
[22] M. Korytkowski, L. Rutkowski, R. Scherer, Fast image classification by boosting fuzzy classifiers, Inform. Sci. 327 (2016) 175–182.
[23] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[24] A. Kumar, H. Daume III, Learning task grouping and overlap in multi-task learning, 2012, arXiv preprint arXiv:1206.6417.
[25] C.A. Lascala, J. Panella, M.M. Marques, Analysis of the accuracy of linear measurements obtained by cone beam computed tomography (CBCT-NewTom), Dentomaxillofacial Radiol. (2014).
[26] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[27] C.H. Leow, E. Bazigou, R.J. Eckersley, C.H. Alfred, P.D. Weinberg, M.X. Tang, Flow velocity mapping using contrast enhanced high-frame-rate plane wave ultrasound and image tracking: methods and initial in vitro and in vivo evaluation, Ultrasound Med. Biol. 41 (11) (2015) 2913–2925.
[28] C. Li, J. Zhu, J. Chen, Bayesian max-margin multi-task learning with data augmentation, in: ICML, 2014, pp. 415–423.
[29] Y. Liu, W. Han, Y. Zhang, L. Li, J. Wang, L. Zheng, An Internet-of-Things solution for food safety and quality control: A pilot project in China, J. Ind. Inf. Integr. 3 (2016) 1–7.
[30] A. Luszczynska, K. Horodyska, K. Zarychta, N. Liszewska, N. Knoll, U. Scholz, Planning and self-efficacy interventions encouraging replacing energy-dense foods intake with fruit and vegetable: A longitudinal experimental study, Psychol. Health 31 (1) (2016) 40–64.
[31] L. Ma, L. Cheng, M. Li, Y. Liu, X. Ma, Training set size, scale, and features in geographic object-based image analysis of very high resolution unmanned aerial vehicle imagery, ISPRS J. Photogramm. Remote Sens. 102 (2015) 14–27.
[32] Z. Meng, V.V. Yakovlev, Precise determination of Brillouin scattering spectrum using a virtually imaged phase array (VIPA) spectrometer and charge-coupled device (CCD) camera, Appl. Spectrosc. 70 (8) (2016) 1356–1363.
[33] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P.T. Furgale, R. Siegwart, A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM, in: Robotics and Automation, ICRA, 2014 IEEE International Conference on, IEEE, 2014, pp. 431–437.
[34] S. Niu, Q. Chen, L. de Sisternes, Z. Ji, Z. Zhou, D.L. Rubin, Robust noise region-based active contour model via local similarity factor for image segmentation, Pattern Recognit. 61 (2017) 104–119.
[35] P. Ochs, R. Ranftl, T. Brox, T. Pock, Techniques for gradient-based bilevel optimization with non-smooth lower level problems, J. Math. Imaging Vision 56 (2) (2016) 175–194.
[36] A. Ouamane, B. Messaoud, A. Guessoum, A. Hadid, M. Cheriet, Multi scale multi descriptor local binary features and exponential discriminant analysis for robust face authentication, in: Image Processing (ICIP), 2014 IEEE International Conference on, IEEE, 2014, pp. 313–317.
[37] Z. Pang, Q. Chen, W. Han, L. Zheng, Value-centric design of the internet-of-things solution for food supply chain: value creation, sensor portfolio and information fusion, Inf. Syst. Front. 17 (2) (2015) 289–319.
[38] A. Perrone, V. Lapenna, S. Piscitelli, Electrical resistivity tomography technique for landslide investigation: A review, Earth-Sci. Rev. 135 (2014) 65–82.
[39] P. Pouladzadeh, S. Shirmohammadi, R. Al-Maghrabi, Measuring calorie and nutrition from food image, IEEE Trans. Instrum. Meas. 63 (8) (2014) 1947–1956.
[40] G.J. Qi, X.S. Hua, Y. Rui, J. Tang, H.J. Zhang, Two-dimensional active learning for image classification, in: Computer Vision and Pattern Recognition, 2008 CVPR 2008 IEEE Conference on, IEEE, 2008, pp. 1–8.
[41] P. Rai, H. Daumé III, Infinite predictor subspace models for multitask learning, in: AISTATS, 2010, pp. 613–620.
[42] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, Mach. Learn. 85 (3) (2011) 333–359.
[43] S.A. Reinsberg, E.D. Scurr, J.M. Brewster, G.S. Payne, Magnetic resonance imaging in prostate cancer: the value of apparent diffusion coefficients for identifying malignant nodules, Br. J. Radiol. (2014).
[44] R.N. Sandeep, Y. Verma, C.V. Jawahar, (2014) Relative parts: Distinctive parts for learning relative attributes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3614–3621.

[45] A. Sharif Razavian, H. Azizpour, J. Sullivan, S. Carlsson, CNN features off-the-shelf: an astounding baseline for recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 806–813).

[46] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.

[47] H. Tao, L. Zhao, J. Xi, L. Yu, T. Wang, Fruits and vegetables recognition based on color and texture features, Trans. Chinese Soc. Agric. Eng. 30 (16) (2014) 305–311.

[48] G. Tsoumakas, I. Katakis, Multi-label classification: An overview, Int. J. Data Warehousing Min. 3 (3) (2007).

[49] G. Verhoeven, M. Nowak, R. Nowak, Pixel-level image fusion for archaeological interpretative mapping, in: 8th International Congress on Archaeology, Computer Graphics, Cultural Heritage and Innovation (ARQUEOLÓGICA 20), Editorial Universitat Politècnica de València., 2016, pp. 404–407.

[50] Y. Wang, W. Ji, S.S. Chaudhry, A hybrid approach for the evaluation of supermarket food safety, J. Manage. Anal. 1 (2) (2014) 156–167.

[51] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, CNN: Single-label to multi-label, 2014, arXiv preprint arXiv:1406.5726.

[52] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, Z. Zhang, The application of two-level attention models in deep convolutional neural network for fine-grained image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 842–850.

[53] M. Yang, Y. Li, Z. Zhang, Multi-task learning with gaussian matrix generalized inverse gaussian model, in: ICML, vol. 3, 2013, pp. 423–431.

[54] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: European Conference on Computer Vision, Springer, 2014, pp. 818–833.

[55] M.L. Zhang, Z.H. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (8) (2014) 1819–1837.

[56] Y. Zhang, J.G. Schneider, Learning multiple tasks with a sparse matrix-normal penalty, in: Advances in Neural Information Processing Systems, 2010, pp. 2550–2558.

[57] Y. Zhang, L. Wu, Classification of fruits using computer vision and a multiclass support vector machine, Sensors 12 (9) (2012) 12489–12505.

[58] Q. Zhou, G. Wang, K. Jia, Q. Zhao, Learning to share latent tasks for action recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2264–2271.

Hao Chen received the M.S. degree in software engineering from Hunan University, Changsha, China, in 2004, and the Ph.D. degree from the School of Information Science and Engineering, Changsha, China, in 2012.

He is currently an Associate Professor and Ph.D. Supervisor at the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests are Web mining, personalized recommendation and big data technology.

Jianglong Xu received the B.S. degree in medical informatics from Central South University, Changsha, China, in 2008, and the master degree in intelligence science from Central South University, Changsha, China, in 2010.

He is currently a Ph.D. student at the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests are deep learning, web mining, personalized recommendation and big data technology.

Guangyi Xiao received the B.Econ. degree in software engineering from the Hunan University, Changsha, China, in 2006, the M.Sc. degree in software engineering from the University of Macao, Macao, China, in 2009 and the Ph.D. degree in software engineering from University of Macao, Macao, in 2015.

He is currently an Assistant Professor in information engineering with the Hunan University, Changsha, China. His principal research is in the field of concept representation, semantic integration, interoperation, and collaboration systems, mainly applied to the fields of e-commerce, e-marketplace, e-banking, and virtual world.

Qi Wu received the B.Econ. degree in automation engineering from the Central South University of Forestry and Technology, Changsha, China, in 2017.

He is currently a graduate student in software engineering with the Hunan University, Changsha, China. His principal research is in the field of artificial intelligence, image processing and big data technology.

Shiqin Zhang received the bachelor degree in Electric Engineering from the Zhengzhou University of Aeronautics of China, Zhengzhou, in 2014, and he is a master student of College of Computer Science and Electronic Engineering, Hunan University now.