Computers in Industry 95 (2018) 15–21

Contents lists available at ScienceDirect

Computers in Industry

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

Multi-source data fusion using deep learning for smart refrigerators

Weishan Zhang[a,*], Yuanjie Zhang[a], Jia Zhai[b], Dehai Zhao[a], Liang Xu[a], Jiehan Zhou[c], Zhongwei Li[a,*], Su Yang[d]

[a] Department of Software Engineering, China University of Petroleum, No. 66 Changjiang West Road, Qingdao 266031, China
[b] Science and Technology on Optical Radiation Laboratory, No. 52 Yongding Road, Beijing 100854, China
[c] University of Oulu, FI-91004, Finland
[d] College of Computer Science and Technology, Fudan University, Shanghai 200433, China

ARTICLE INFO

Article history:
Received 12 June 2017
Accepted 8 September 2017
Available online xxx

Keywords:
Deep learning
Smart refrigerator
Fruit recognition
Multi-source data fusion

ABSTRACT

Food recognition is one of the core functions for a smart refrigerator. But there are many challenges for accurate food recognition due to reasons of too many kinds of food inside the refrigerator which tends to obscure each other, and they may look very similar. This paper proposes a fruit recognition approach that fuses weight information and multi deep learning models. The proposed approach can remarkably improve recognition accuracy. We have extensively evaluated the proposed approach for its performance and accuracy, which demonstrate the effectiveness of the proposed approach.

1. Introduction

Smart refrigerator is an essential appliance in a smart home. As one of the core functions of a smart refrigerator, recognizing food in a refrigerator is very important for, e.g. knowing whether the food is fresh and the amount of storage left in the refrigerator. However, recognizing food (e.g. fruits as demonstration approach in our paper) efficiently in a refrigerator is challenging due to the fact that food may be piled together, food may look very similar, and different food may obscure each other.

There exists quite some work on fruit recognition. Bolle et al. [1] proposed Veggie vision, which can identify fruits by extracting the characteristics of color, texture and density. However this work is sensitive to illumination changes. Zawbaa et al. [2] explored fruit shape and color to identify fruits. Naskar et al. [3] extract color, sharp and texture features, using artificial neural network to identify the fruit. Patel et al. [4] proposed an algorithm for fruit detection based on multi-feature. Shebiah et al. [5] proposed an efficient fusion of color and texture features for fruit recognition.

Deep learning provides new possibilities for effective recognition of objects, as illustrated by the classification results for the ImageNet problem with deep convolution neural network (deep CNN, or DCNN) [6]. It performs better in object recognition than most of traditional approaches [7]. But as fruits may have the same color, shape, and texture, like orange and tangerine, these existing methods are hardly effective to recognize food.

Therefore in this paper, we propose an integrated data fusion approach where multi convolution neural models together with weight information are combined to conduct fruit recognition. Through this multi-source data fusion approach, the accuracy of fruit recognition is improved significantly when handling similar fruits with the same color, shape and texture, where three SSD [8] models are introduced to identify the fruit.

Besides the data fusion approach, two different architectures, running remotely or locally, are proposed for realizing this multi-source data fusion approach. The first one is using RPi (raspberry pi)[1] which carries camera and weighing sensors to collect data. RPi will send the data to the server. The server will use HBase[2] to store data. Then the result of recognition will feedback to RPi. This method is called the cloud mode. Another architecture uses TX1[3] which collects data and identify the fruits by itself, this method is called local mode. The performance of the cloud mode is compared with the local mode.

* Corresponding authors.
E-mail address: zhangws@upc.edu.cn (W. Zhang).

[1] RPi is a micro-computer based on ARM which the size is as small as a credit card. https://www.raspberrypi.org/.
[2] http://hbase.apache.org/.
[3] NVIDIA Jetson TX1 can provide good computing performance which up to 1 T-Flops, and support the NVIDIA CUDA technology. http://www.nvidia.com/object/jetson-tk1-embedded-dev-kit.html.

http://dx.doi.org/10.1016/j.compind.2017.09.001
0166-3615/

The contributions of this paper are:

- A novel fruit recognition approach is proposed, where we design the recognition algorithm using neural network with multi-model fusion, combining with weight information. Through multi-source based data fusion, the accuracy of fruit recognition is improved significantly, when handling fruits with the same color, shape and texture.
- A data set for fruits is built for the refrigerator environment, which contains ten kinds of fruits with different shooting angels and different lighting environments.
- Two architectures of the smart refrigerator are proposed, running locally or remotely, using RPi plus HBase, and using TX1 respectively.

The remainder of the paper is organized as followed: Section 2 presents the smart refrigerator framework; based on this platform, Section 3 presents the approach of multi-source based data fusion for fruit recognition. Section 4 discusses the evaluations of the fruit recognition. Section 5 shows some related work. Conclusion and future work end the paper.

2. Algorithm of fruit recognition using multi-source data fusion

Intuitively, different deep neural network models may extract different features for recognition, and the combination of these models can lead to a higher recognition accuracy than a single model. There, our first attempt is using neural network (BP in our case) with multi-model fusion, based on three SDD models, namely ResNet [9], VGG16 and VGG19 [7]. Although there are many CNN based recognition approaches can be used for fruit recognition, such as Faster R-CNN [10], YOLO [11]. Due to its performance, and accuracy, we choose SSD models instead as the basis for our approach.

2.1. Multi-model fusion

The multi-model fusion method for fruit recognition is shown in Fig. 1. The same data set is used to train SSD models, three outputs of these three models are used as the input of BP neural network.

The outputs of these three models serve as the input for the BP neural network, where we show the structure of it in Fig. 2. The node number of the input layer is 3, and the node number of hidden layer is 10, while the output layer only has one node. The neuron between the node of hidden layer and output layer uses a linear transfer function. The relationship between the state of neuron X_i and output Y_i is a linear transformation,

$$Y_i = f(X_i) = X_i. \tag{1}$$

We assume D_1, D_2, D_3 is the input of the neural network, the output is

$$E_i = D_I, \quad i = 1, 2, 3. \tag{2}$$

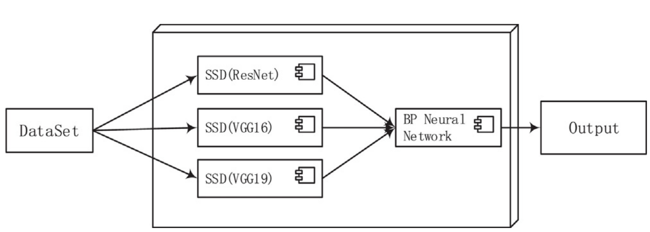

Fig. 1. The structure of multi-model fusion.

For the hidden layer, the input of node j is

$$F_j = L_{j1} \cdot E_1 + L_{j2} \cdot E_2 + L_{j3} \cdot E_3 + M_j; \tag{3}$$

the output is

$$H_i = f(X_j), \quad j = 1, 2, 3 \cdots, 10. \tag{4}$$

For the output layer, the input is

$$k = L^2_{1j} H_j + M^2; \tag{5}$$

the output is

$$Y = f(k) = k. \tag{6}$$

L_{ji} and L^2_{1j} are the linked weights, M_j and M^2 are the constant bias.

2.2. Multi-source data fusion

The usage of multi-model fusion can improve the recognition accuracy to some degree, but it is still hard to differentiate similar fruits. In our approach, we use the weight information of a fruit to help the recognition process. We build a priori knowledge data set for each kind of fruit including name of the fruit, weight range and a list of similar fruits. After obtaining the name of the fruit using multi-model of deep learning, and its weight range from the knowledge base, the decision on fruit type will be made based on the combination of these information.

Fruits that cannot be measured by weight are also taken into consideration, such as a bunch of grapes, bananas and a box of strawberries, etc. For such kind of fruit, it usually has unique features, which makes weight information useless. The fusion method is given in Algorithm 1.

Algorithm 1. Multi-source data fusion

```
Type Fruit
  Dim names As STRING
  Dim weights As STRING
  Dim SimilarList As LIST
HashMap ⟨string, Fruit⟩ weightMap ← new HashMap ⟨string, Fruit⟩
For every fruits
  fruit.names ← db.names
  fruit.weightranges ← db.weights
  fruit.similiar ← db.similarlist
  WeightMap.put ⟨fruit. name, Fruit⟩
done
objects ← detect(pretrain model)
addFruit ← Objects-LastObjects
lastObject ← Objects
weight ← addfruit.weight
name ← addfruit.name
fruit ← WeightMap.get(name)
weightRange ← fruit.weightranges
similar ← furit.similar
Dim maxName As STRING
Dim maxProb As INTEGER ← 0
For similarFruit: similar
  If (similarFruit.Prob > maxProb) and (similarFruit.weight in range)
    maxName = similarFruit.name
    maxProb = similarFruit.Prob
  done
done
return maxName
```

3. Multi-source based data fusion for smart refrigerators using deep learning

Considering the requirement of running fruit recognition locally or remotely, we have designed two architectures. The first one uses RPi (raspberry pi) to send sensed data to a remote

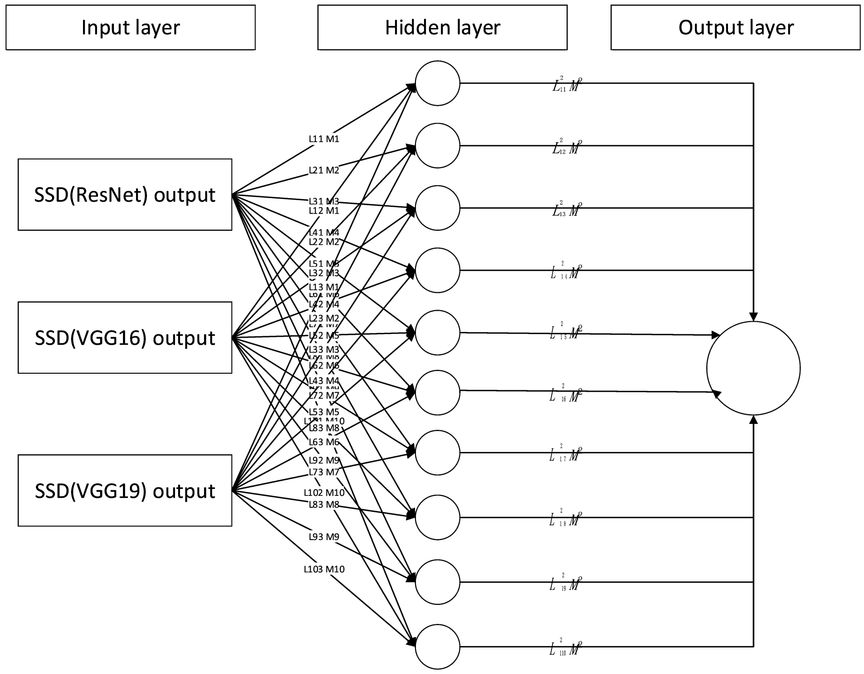

Fig. 2. The structure of neural network.

HBase[4] server. Then the result of recognition will feedback to RPi. This method is called the cloud mode. Another architecture uses TX1 which collects data and identify the fruits by itself, this method is called local mode.

3.1. Architecture of the cloud based mode

In the cloud mode, a RPi is connected with a camera and weighting sensors. When the RPi detects weight changes, it will trigger the camera to take a picture. The RPi will send the image, weight data, time information and user id to the server through a Socket connection. The HBase server uses a big table to store the fruit image, its weight, time information and user id at the same time. HBase will use user id combined with time information as the row key, then it will use the proposed multi-source data fusion approach to recognize fruits. The results will be stored in HBase and send a feedback to RPi, then the result is shown in the refrigerator. This architecture is divided into sensing layer, transport layer and application layer as shown in shown in Fig. 3.

For the sensing layer, RPi 3 is used which carries ARM Cortex-A53 1.2 GHz 64-bit quad-core ARMv8 CPU, 1 GB of memory. RPi 3 joins with weighing sensors and a camera. Four weighting sensors are installed under tempered glass. The tempered glass is placed in the refrigerator to measure the weight of fruits.

The transportation layer is using a socket for data transmission. The client is the RPi, and the server is a cluster with 4 high performance computers. RPi sends data to the server. After finishing the recognition, the server will feedback the data to RPi. The server cluster contains 4 computers with Nivada 1080 GPU and i7 6700K CPU.

For the application layer, there are two components. One is the storage module, and the other one is the recognition module. It uses HBase to store the sensed data, where one table uses user id

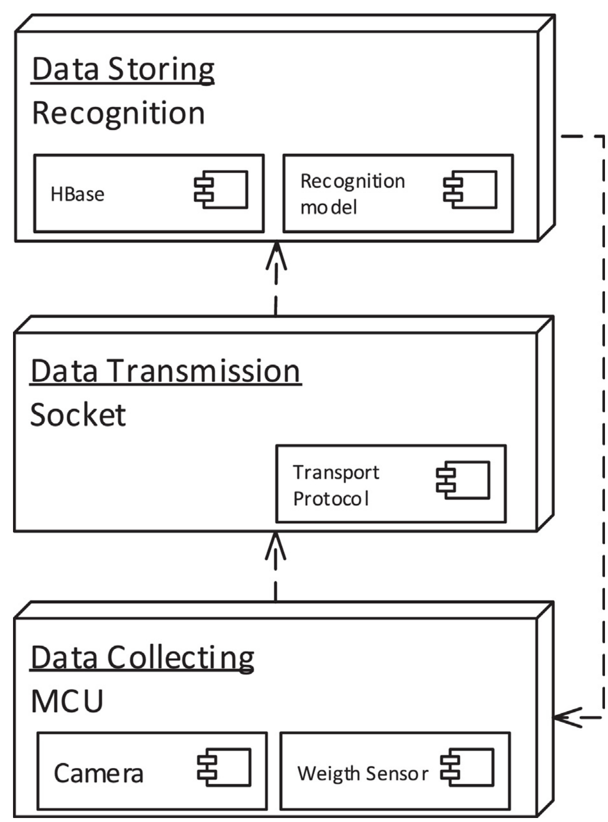

Fig. 3. Architecture of fruit recognition in the smart refrigerator using the cloud mode.

combined with time information as the row key. The table only has one column family, contains two columns, including the sensed image, and the weight data. The trained recognition model is running in a Hadoop cluster to identify the fruit in images.

[4] http://hbase.apache.org/.

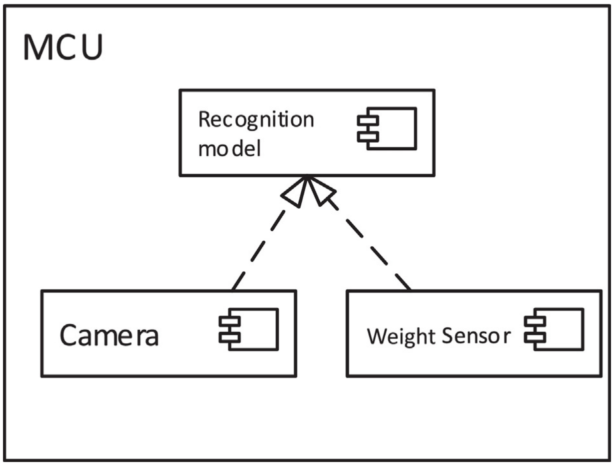

Fig. 4. Architecture of fruit recognition using TX1 for local mode.

3.2. Architecture of the local mode

Some users may worry about the problem of privacy, so we also propose an approach which do recognition jobs at local. We use TX1 development board to collect data and deploy the recognition model to TX1. The architecture is shown in Fig. 4.

3.3. System implementation

Four weighing sensors installed under the tempered glass, the data obtained by weighing sensor are analog signals. We use HX711 modules to make analog-digital conversion, then they send the converted digital signal to the RPi. A 720P HD camera is chosen to take photos, which is connected with RPi through a USB cable.

HBase can only use row key to query data, so the design of row key is very important. A good row key can improve the query efficiency. In our case, we combined user id and time information to be the row key. For example, Jack011 20170303093026, where Jack011 represents the user id, while 20170303093026 is a data accurate to seconds, and 2017 represents year, 0303 represents month and day, 093026 represents the time 9:30:26.

The UML class diagram of the implemented system architecture is shown in Fig. 5.

4. Evaluation

Efficiency experiments of overall process are taken and compare these two methods in order to verify the feasibility of method we proposed. Four kinds of methods' recognition accuracy of fruit recognition model are also compared by experiments.

4.1. Experimental environment

For all the evaluations, the testbed configuration is shown in Table 1. The software packages include jdk-1.8, OpenCV-2.4.9, Hadoop-2.6.4, Hbase-1.2.1 and Caffe. The software packages in TX1 do not include Hadoop-2.6.4 and Hbase-1.2.1, because the results of TX1 would not be analyzed later.

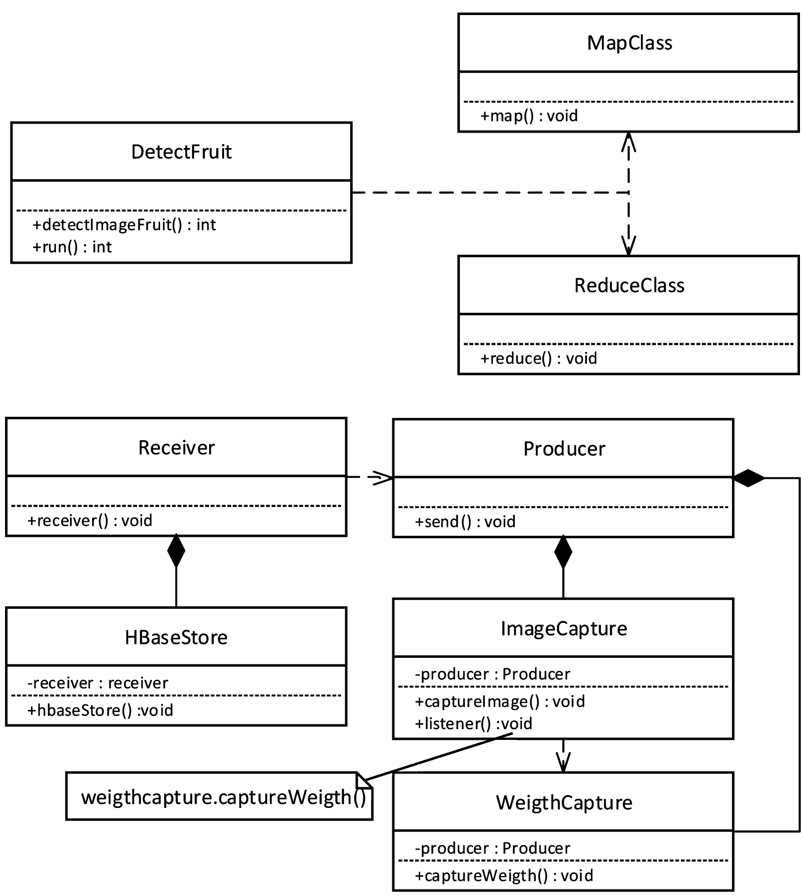

Fig. 5. The class diagram of the implemented system architecture.

Table 1
Experimental configurations.

Machine name	Hardware	Software	Number
1080	i7 6700K NVIDIA 1080	JDK1.8, Opencv2.4.9 Caffe, Hadoop2.6.4 Hbase1.2.1	4
TX1	ARM Cortex-A57 MPCoreCPU NVIDIA Maxwell GPU	JDK1.8, Opencv2.4.9 Caffe	8

4.2. Data set

Our data set includes fruit images from Internet and the data collected by ourselves with camera mounted on refrigerator. The ratio of them is about 1:3. The training set is made of 20,000 images and there are 5000 images in the test set.

The recognition accuracy is affected by light, camera angle, and so on. When shooting images, the light is limited, and it can be divided into dark environment and bright environment. Four kinds of shooting angles are considered, i.e. taking photos from four corners of the refrigerator. Therefore four data sets are made, and each one has a unique angle. The training set contains 4000 pictures, while the testing set contains 1000 pictures. Besides, considering the problem of personal privacy using the remote server cluster, the image we shoot can only show the inside of the refrigerator. Fig. 6 shows a part of images from data set.

4.3. Accuracy

The recognition accuracy for different models is shown in Table 2, showing the recognition results of our proposed multi-source data fusion based method, some single models and multi-model. All these methods use our own data set and are tested under the same experimental configurations.

It is obvious that our model which use the approach of multi-model fusion combined with weight data has a higher accuracy than that of other models.

Table 2
Recognition accuracy for different model of a pedestrian.

Structure of network	Accuracy
SSD (ResNet)	0.91
SSD (VGG16)	0.89
SSD (VGG19)	0.90
Multi-model fusion	0.92
Our model	0.97

The confusion matrix for the multi-model is shown in Fig. 7, and the confusion matrix for the multi-source data is shown in Fig. 8. The accuracy of Fig. 8 is better than Fig. 7. The probability of mis-recognition of orange and tangerine is very high in Fig. 7. The same situation also appears in distinguishing pear and yellow apple. But in Fig. 8, the probability of incorrectly recognize is decreased. Especially, the confusion of orange and tangerine, pear and yellow apple is decreased.

We also use 4 data sets with 4 different shooting angels to train our proposed model. The accuracy is shown in Table 3. Because the scale of each data set is smaller than the data set containing 4 angels, the accuracy is lower than the bigger data set. But the best angel can be calculated. Obviously, the best angel is in the front of the refrigerator.

In an early experiment, we only use single model SSD (VGG16) to test several images contains fruits shading each other. We focus on the identification of orange and tangerine, yellow apple and pear. These fruits are difficult to distinguish. As shown in Fig. 9, the accuracy of single model is not very high. In the middleware of the picture, pear even can be identified as yellow apple by mistake.

Our model has a much better performance as shown in Fig. 10. The orange and tangerine, yellow apple and pear both can be correctly identified and the accuracy is very high.

4.4. Performance

We have tested the time taken for image taken by the camera sending to the server cluster, then the time for recognition for the cloud based mode. We also show the local mode running by TX1.

Fig. 6. The image from our data set.

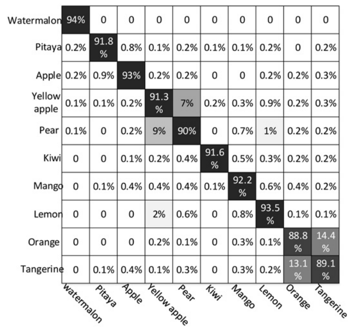

Fig. 7. Confusion matrix for the multi-model.

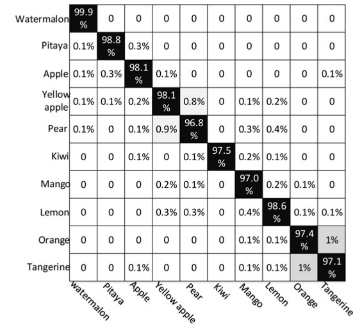

Fig. 8. Confusion matrix for the multi-source data.

Table 3
The accuracy of data sets with different angels which use our proposed model.

Camera angel	Accuracy
Right front	0.90
Right rear	0.89
Left front	0.92
Left rear	0.87

The performance is shown in Table 4. We can see that for the actual recognition on the cloud, it takes less than 50 ms, with the main cost of comes from the data transmission. The TX1 performs well and can be used locally as less than 1 s is acceptable for users.

5. Related work

There is scarce published work on food recognition for smart refrigerators, though there are some work on food recognition in general scenarios.

Sa et al. [12] presents a novel approach to fruit detection using deep convolutional neural networks. This paper combined the RGB and NIR, using Faster R-CNN to detect fruit. They evaluate their method and reach 83% accuracy for the detection of sweet pepper. We did not choose Faster R-CNN as we found that SSD performs better.

Pratap et al. [13] proposed an algorithm for fruits classification based on the shape, color and texture. Using artificial neural network to identify the fruit. The accuracy this paper evaluated is up to 96%. But the data set contains only five kinds of fruit. Our data set is more complex than this paper.

Tao et al. [14] proposed a novel texture feature extraction algorithms called color completed local binary pattern (CCLBP). This paper uses a HSV color histogram and border/interior pixel classification (BIC) color histogram to extract image color features. But this paper only considered the change of light intensity. Ijjina et al. [15] proposed a method for classification of objects that is invariant to illumination color, illumination direction and view-point based on 3D color histogram, and combined with CNN. But the test image only has only one object in one angel. Our data sets contain images with 4 different angels, and have considered light changes, and have more kinds of fruits.

Zhang et al. [16] proposed a hybrid classification method based on fitness-scaled chaotic artificial bee colony (FSCABC) algorithm and feedforward neural network (FNN). The experimental results of the 1653 color fruit images from the 18 categories demonstrated that the FSCABC–FNN achieved a classification accuracy of 89.1%. Song et al. [17] proposed a method of recognition and counting the number, they use the bag-of-words model to identify the single image, and then use a new method to estimate multiple images. These approaches have lower accuracy than ours.

Kwon et al. [18] proposed a sensor-equipped food container, smart refrigerator, which discriminates foods and monitors their status. Sandholm et al. [19] present a testbed for exploring novel

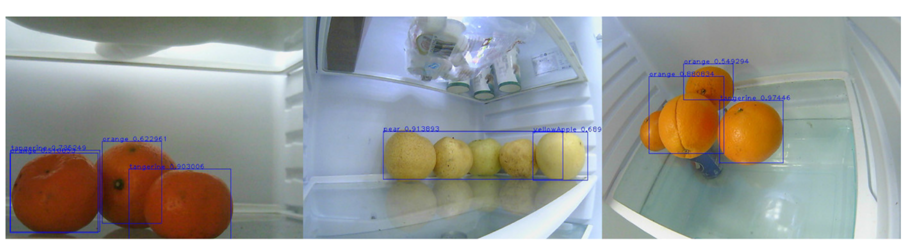

Fig. 9. The identify result by single model.

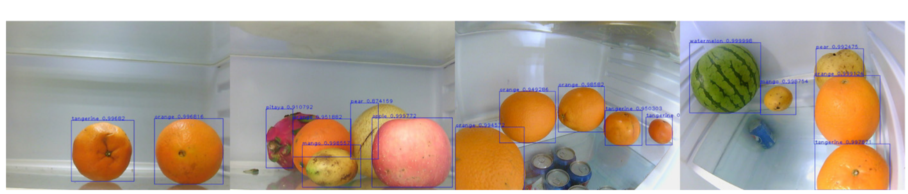

Fig. 10. Identified result.

Table 4
The complete of efficiency.

Mode	Data transmission	Recognition time	Total time
Server cluster	481 ms	42 ms	523 ms
TX1		603 ms	603 ms

smart refrigerator interactions. None of the above work uses big data platform, which plays a very important role for data mining later on [20].

6. Conclusion and future work

In this paper, we proposed a multi-source data fusion approach for smart refrigerator which combines weight data and multi deep learning models. Two architectures are proposed for data collection, data processing and fruit recognition. We also build a comprehensive data set with different shot angles, and considering light changes. The evaluations show that our approach can perform very well with good recognition accuracy.

In the future work, we will enrich the data set, and conduct more tests. We are working with Haier to make some ideas possible in their future prototypes.

Acknowledgements

The work presented in this paper is supported by the Key Research Program of Shandong Province (No. 2017GGX10140), National Natural Science Foundation of China (No. 61309024), the Fundamental Research Funds for the Central Universities (2015020031).

References

[1] R.M. Bolle, J.H. Connell, N. Haas, R. Mohan, G. Taubin, Veggievision: a produce recognition system, Proceedings IEEE Workshop on Applications of Computer Vision, 1996. WACV'96 (1997) 244–251.
[2] H.M. Zawbaa, M. Abbass, M. Hazman, A.E. Hassenian, Automatic Fruit Image Recognition System Based on Shape and Color Features, (2014) .
[3] S. Naskar, T. Bhattacharya, A novel fruit recognition technique using multiple features and artificial neural network, Int. J. Comput. Appl. 116 (20) (2015) 23–28.
[4] H.N. Patel, R.K. Jain, M.V. Joshi, Fruit detection using improved multiple features based algorithm, Int. J. Comput. Appl. 13 (2) (2011) 1–5.
[5] R.N. Shebiah, Fruit recognition using color and texture features, J. Emerg. Trends Comput. Inf. Sci. 1 (2) (2010) 90–94.
[6] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[7] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, (2014) . arXiv:1409.1556.
[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, Shot Multibox DSSD: Single Shot MultiBox Detector, European Conference on Computer Vision (2015) 21–37.
[9] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, (2015) , pp. 770–778.
[10] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. (2016) 1.
[11] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, Comput. Sci. (2016) 779–788.
[12] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, C. Mccool, Deepfruits: a fruit detection system using deep neural networks, Sensors 16 (8) (2016) 1222.
[13] B. Pratap, N. Agarwal, S. Joshi, S. Gupta, Development of ANN based efficient fruit recognition technique, Glob. J. Comput. Sci. Technol. 14 (5-C) (2014) 1.
[14] H. Tao, L. Zhao, J. Xi, L. Yu, T. Wang, Fruits and vegetables recognition based on color and texture features, Trans. Chin. Soc. Agric. Eng. 30 (16) (2014) 305–311.
[15] E.P. Ijjina, C.K. Mohan, View and Illumination Invariant Object Classification Based on 3D Color Histogram Using Convolutional Neural Networks, Springer International Publishing, 2014.
[16] Y. Zhang, S. Wang, G. Ji, P. Phillips, Fruit classification using computer vision and feedforward neural network, J. Food Eng. 143 (6) (2014) 167–177.
[17] Y. Song, C.A. Glasbey, G.W. Horgan, G. Polder, J.A. Dieleman, G. van der Heijden, Automatic fruit recognition and counting from multiple images, Biosyst. Eng. 118 (1) (2014) 203–215.
[18] T. Kwon, E. Park, H. Chang, Smart refrigerator for healthcare using food image classification, ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (2016) 483–484.
[19] T. Sandholm, D. Lee, B. Tegelund, S. Han, B. Shin, B. Kim, Cloudfridge: a testbed for smart fridge interactions, Comput. Sci. 1 (2014) 133–135.
[20] W. Zhang, L. Xu, P. Duan, W. vGong, Q. Lu, S. Yang, A video cloud platform combing online and offline cloud computing technologies, Pers. Ubiquitous Comput. 19 (7) (2015) 1099–1110.