Food Object Recognition Using a Mobile Device: State of the Art

Simon, Knez[1] vbknez@hotmail.com
and Luka Šajn[1] luka.sajn@fri.uni-lj.si

Faculty of Computer and Information Science, University of Ljubljana, Ljubljana, Slovenia

Abstract. In this paper nine mobile food recognition systems are described based on their system architecture and their core properties (the core properties and experimental results are shown on the last page). While the mobile hardware increased its power through the years (2009 - 2013) and the food detection algorithms got optimized, still there was no uniform approach to the question of food detection. Also, some system used additional information for better detection, like voice data, OCR and bounding boxes. Three systems included a volume estimation feature. First five systems were implemented on a client-server architecture, while the last three took advantage of the available hardware in later years and proposed a client only based architecture.

1 Introduction

Because of the raise of diet related diseases, a lot of easy to use applications were made for the purpose of dieting guidance. At first, dieting guidance was mediated by manual monitoring of eating activity (keeping records of foods and beverages consumed), followed by storing them on a web site application with instant analysis (web sites like Calorie King, CRON-O-Meter, Selfnutritiondata, etc.). All this applications allow the user to get information about the nutrients of the selected food, but they need to know which food and how much of it has been eaten. For the purpose of easier tracking of food activity, a merge of object detection and previous diet monitoring systems was bound to happen. The first food object detection algorithm, that was implemented for practical use on a mobile device, was described in the paper of Joutou in the year 2009 [1]. New approaches to lower the amount of knowledge and hardware needed for the users analysis of eating followed. The outline of the paper is as follows: section 2 describes the methods of analysis of the applications, section 3 reviews these applications. In section 4 the paper concludes.

2 Methods

2.1 Comparisons of applications

In this paper nine mobile food recognition applications are described. Each application is described by its core features and its system architecture. At the end

of the paper, a table of all used algorithms for food recognition are listed and grouped into 5 stages of object recognition.

2.2 The applications

Nine applications were described in this paper, aging from the year 2009 to the year 2013: Joutou and Yanai, 2009 [1], Puri et al., 2009 [2],Hoashi, Joutou and Yanai, 2010 [3], Kong and Tan, 2012 [4], Rahman et al., 2012 [5], Kawano and Yanai, 2013 [6], Kawano and Yanai, 2013 [7], Anthimopoulos et al., 2013 [8] and Pouladzadeh, Shirmohammadi and Arici, 2013 [9].

3 Review of applications

All of the reviewed applications in this paper use the mobile phone to record eating activity via images. To automatically detect what food objects were eaten a food object recognition algorithm is needed. Object recognition algorithms are mostly comprised of the following steps or stages: image acquisition, image processing, image segmentation, image feature extraction and image classification. Image acquisition deals with different techniques of data capturing (some applications use one image, some use multiple images, some use OCR or voice data), while the processing stage deals with different pixel normalization techniques for the reduction of the noise present. These two stages are needed for better image segmentation, where ROIs - regions of interest are extracted. Regions of interest (usually background, foreground subtraction) allow to shorten the processing time of the later stage of feature extraction, since the area of analysis is smaller and object specific. The next stage is to extract different features from the segmented images. Many different image features techniques can be used. This features are later used for the classification. After the classification is done, the algorithm outputs the class to which the inputed set of features belongs to. After the output, it is in the hands of the front-end developer to display the results in a desired shape or form (caloric analysis, recommendation for further eating activities, recipes suggestions...). Because of the difference of availability of technology and the object detection algorithms, mobile food detection applications differ through the years in most of the stages. While some aspects of a particular stage co-exist, newer and better one replaced the old ones or are added to the mix.

3.1 Joutou and Yanai, 2009

According to the authors, this is the first food detection system, which demonstrated practical implications on a mobile phone. The main future of the implemented food detection algorithm is the multiple-kernel-learning (MKL) classification. In simple terms, the MKL algorithm classifies object based on the features and their corresponding weights. This means that the importance of

a particular image feature changes dynamically, depending on the food category. This goes very well in hand with food objects, especially when we move from simple fruit or vegetable detection, to detection of complex meals. In the following year (2010) this system was extended, to recognize 85 different food categories [3]. The goal of the authors was to retain the success rate of classification of the previous system. For this, eight additional HoG (histogram of oriented gradients) features were added.

3.2 Puri et al., 2009

In the paper of Puri et al. [2], the authors described and proposed a food item and volume estimation system. In the paper the authors recognize that dietary assessment of food is a very difficult procedure even for nutritional experts. The difficulty lies especially in dishes where some nutrients are occluded. Therefore an algorithm based system of food and volume detection needs as much information as possible. A method based on a Food Intake Visual and Voice Recognizer (FIVR) system was proposed. The system is then implemented using a mobile device and so uses image as well as voice data for food recognition and volume estimation[1]. The system is created to recognize multiple items on a plate. All the algorithms for image segmentation and classification are executed on a server. The mobile phone is used only for recording of voice and image data.

The system architecture. The main feature of the application is that it enables to recognize foods and estimate its volume. The system architecture is as follows: the user initially takes three pictures of a meal (the scene setup can be seen in Fig. 1[2]. Than the users accompanies this pictures with a speech data, to create food labels for the items on the plate. These data is then sent to the server[3]. The food is identified based on the speech data. For each identified food object, 3D reconstruction follows to estimate its volume. After the volume estimation, data of foods and their corresponding volumes are stored on the server and sent to the user via a text message. The message includes nutritional facts of each item.

3.3 Kong and Tan, 2012

In the paper of Kong and Tan [4] the DietCam mobile phone application was introduced. Similar to the system of Puri et al. [2] this system is used for food detection as well as volume estimation. The main difference in the case of the DietCam is, that it executes both functions only on the basis of images. Additionally, the mobile application uses the accelerometer, for calculating the angle of the device, which comes in handy for multiple view approach of picture recording. Besides regular food images, optical character recognition technique (OCR) is used for reading the food labels.

[1] For volume estimation 3D reconstruction method is used
[2] The image was derived from the paper [2]
[3] The system runs on an Intel Xeon workstation with a 3GHz CPU and 4GB of RAM.

The system architecture. The system architecture can be described as follows: first the users captures 3 pictures of different item poses or a video around the food item. Then the information is sent in a XML format to the server, where all the processes of image processing, feature extraction, classification and a 3D reconstruction for food item volume estimation are executed. First the image manager extracts the important features, which are used for the classification. The classification is first executed using a local database, where instances of users' images are stored. If none of the stored instances can be used for classification of the recorded food item, than the global database of all food images is used. This two step process allows for a quicker and more effective classification of food objects. After the classification, the volume of the food items is estimated via 3D reconstruction. The system has also the possibility to extract the residues of the meal, which is later used for total calories consumed. After this stages, the calories and foods consumed are send to the mobile phone and stored into the database.

3.4 Rahman et al., 2012

In the paper of Rahman, Pickering and Kerr [5] a new texture feature was proposed, which would increase the food recognition accuracy on a mobile platform. The proposed texture feature would be based on Gabor filter banks, which would produce scale and rotation invariant global texture features. The main difference with other dietary consulting applications is that it uses multiple scale and orientation images for food items. This allows for a better food item detection in different poses. For the implementation of this new texture feature, the Technology Assisted Dietary Assessment (TADA) was used. The system executes food detection as well as volume estimation.

The system architecture. The TADA system architecture is based on a interaction between a mobile device for image acquisition and server for image processing, segmentation, feature extraction, classification as well as volume estimation. The user starts with taking a photo of a food item, which is accompanied with a color calibration marker 1[4]. Besides the photo, the user also sends some additional data, like the date, time and geolocation. This information is then sent to the server where food identification and volume estimation is executed (based on the underlying processes). The results of food identification and volume estimation is later sent to the user, for a approval and possible adjustments. After the confirmation, the eating activity is stored on the server, based on the nutritional database (which is also located on the server). Finally, the results of nutrient calculations are sent to a researcher for further analysis, which would in future work incorporate user feedback for food group analysis and dietary recommendations.

[4] The image was derived from the paper [5]

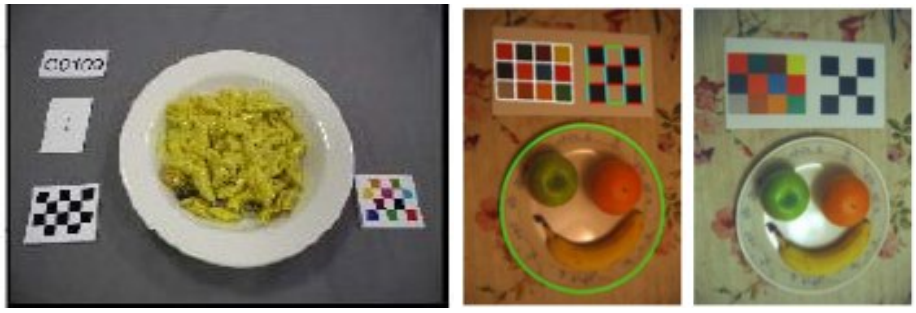

Fig. 1: The right image displays an instances of a scene used in [5], the left image displays the effects of color normalization in [2]

3.5 Kawano and Yanai, 2013

Kawano and Yanai have proposed a lightweight system for food object recognition, where all of the processing is executed on the mobile phone device[6]. This is the first application to do so. The purpose of executing all of the stages of food detection locally on the mobile phone was, to eliminate the un-desired delay and costs between the client and the server, as a result of data transmission. For additionally speed up, the user can draw bounding boxes around the food items, which allows for a quick image segmentation. Because food detection is executing in real-time fashion, the user can adjust the camera angles if the food detection algorithm doesn't detect the correct food item. This is done via an additional feature, which proposes the camera angle, where more information about the food should be visible. The system does not automatically estimate the volume of the items, but it allows the user to indicate the volume with a slider on the screen. In the same year (2010) the system was optimized by reducing the time for food detection inside the bounding boxes [7]. This has been successfully achieved using Fisher vectors [10] and different image features [5]. The optimized version had better processing time of food recognition (down from 0.26 seconds to 0.065 seconds) and slighty better success rate while doubling the number of categories (from 50 to 100). In 2015 these two systems were implemented on a android platform for public use [11].

The system architecture. The system architecture in this application is as follows. First the user points with his camera to the desired food item. Because food detection is executed in real time, the application is collecting frames continuously. Next, the user draws bounding boxes around the food items, which allows for a quick segmentation. After the user created ROIs, food detection is executed in each of them. The results of the food detection algorithm is a list of top 5 candidates and the direction of the region of food items. The direction

[5] To see the difference in the features used in both systems, see the Fig. 1

is displayed as an arrow on the screen, besides the food item in Fig. 2[6]. After the returned candidate list the user has two choices. If the correct food item is on the list, than the user can adjust the estimated volume of the food and select the food item (by touching the list element). If the top 5 candidate list does not include the correct food item, the user moves the mobile phone in the proposed direction indicated by the arrow. By doing so, the food detection algorithm is again executed. This adjustment continues until the desired food item is detected. After the selection of the desired food item, the food item and the nutritional information are displayed on the screen. Also, the user can save the meal records to a server. He can later access this information on the Web.

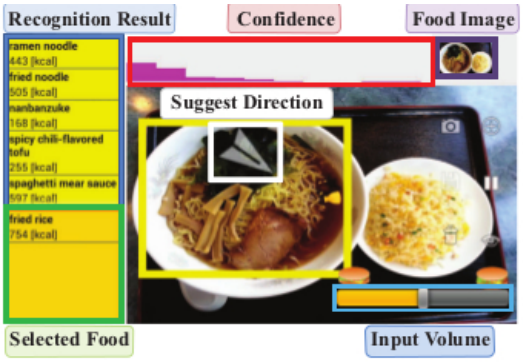

Fig. 2: The implementation of the system on a mobile device. Bounding boxes are user made, the arrows present the direction of the food items, and the sidebar includes top 5 candidates with their nutritional properties (kilo calories). Below the food items, there is a volume estimation slider.

3.6 Anthimopoulos et al., 2013

In [8] a system for carbohydrate counting was proposed and was aimed especially for helping patients with type 1 diabetes. The classification of the images is based on six different categories: meat, breaded food, rice, pasta, potatoes and vegetables and is therefore not food specific. The goal is to separate and identify regions of the before mentioned food groups on a plate. The special property of this system is complexity of the food segmentation which is accomplished with different algorithms. [7]

The system architecture. The system architecture starts with a photo taken by the mobile phone camera. Later the recorded image gets segmented and recognized as on of the six food categories. Based on the segmentation and the classified food category, the system can return the estimated carbohydrate value and the corresponding insulin dose, based on the patients data.

[6] The image was derived from the paper [6]
[7] The algorithms used are listed in Tab. 1.

3.7 Pouladzadeh, Shirmohammadi and Arici, 2013

In the paper of Pouladzadeh [9] a system of automatic food measurement was proposed. Besides the food detection, the aim of the system is to estimate the eaten calories, by using before and after pictures of the meal (similar to the system proposed by Kong in his DietCam, described in section 3.3). The system is client based i.e. all of the the stages of food detection are executed on the mobile device.

The system architecture. The system architecture is as follows: the user records two images of the food - one from the top and another from the side. When taking the picture from the top, the user needs to accompany the image with its' thumb, which will later be used for volume estimation. One of the images is than processed and segmented. After this, different features are extracted for each segmented food item. The extracted features are then used for a correct classification. The application sends the user the information about the classification for the approval[8]. Next, the food area estimation follows. Area estimation is produced by using both pictures of the food item, with the help of the thumb. With the area estimation and nutritional labels, total caloric consumption is calculated. If the user does not finish the whole meal, he/she can later record an image of the leftovers. The volume of the leftovers is than subtracted from the volume of the initial meal.

4 Conclusions and discussion

Nine different mobile phone applications for tracking the eating activity were described in this paper. Through their review, we took a look at different approaches dealing with different system architectures. The main problem in correctly detecting food items is the vast inter-class variability of food items. Since food items are hand made, they differ in shape, texture and size, even if they present the same food instance. While these aspects can be solved with different algorithms which use local and global feature extraction (HoG, SIFT, SURF, deformable parts model etc.) another problem in food detection is occlusion, especially for food items in a complex meal. While detecting a whole food is mostly solved[9] (system described in section 3.7 demonstrated a 92% classification rate for 30 basic food items), the issue with complex food items, which includes chopping, mixing and covering food items with other food items is much more challenging. As mentioned in [9], even a human dietitian would not be able to recognize constituents of such meals by only looking at the plate. Solving this problem with images only will be therefore a hard if not impossible challenge. By using additional sensors for taste, the problem of occlusions might be solved or at least help to produce even better classification results.The last issue in food

[8] If the user does not approve, he/she can correct the information, that were sent as the result of classification

[9] Whole foods being un-processed foods

detection that is described in this paper is volume estimation. Some systems like [2,4,5,9] implemented the feature of automatic volume estimation, which is costly. But relatively good estimations are generated in all of those systems. This allows to calculate a much more accurate calorie consumption. Comparison of all the described systems, their properties and the results of their experiment[10] are shown in the Tab. 1.

References

1. T. Joutou and K. Yanai, "A food image recognition system with multiple kernel learning," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 285–288, IEEE, 2009.
2. M. Puri, Z. Zhu, Q. Yu, A. Divakaran, and H. Sawhney, "Recognition and volume estimation of food intake using a mobile device," in *Applications of Computer Vision (WACV), 2009 Workshop on*, pp. 1–8, IEEE, 2009.
3. H. Hoashi, T. Joutou, and K. Yanai, "Image recognition of 85 food categories by feature fusion," in *Multimedia (ISM), 2010 IEEE International Symposium on*, pp. 296–301, IEEE, 2010.
4. F. Kong and J. Tan, "Dietcam: Automatic dietary assessment with mobile camera phones," *Pervasive and Mobile Computing*, vol. 8, no. 1, pp. 147–163, 2012.
5. M. H. Rahmana, M. R. Pickering, D. Kerr, C. J. Boushey, and E. J. Delp, "A new texture feature for improved food recognition accuracy in a mobile phone based dietary assessment system," in *Multimedia and Expo Workshops (ICMEW), 2012 IEEE International Conference on*, pp. 418–423, IEEE, 2012.
6. Y. Kawano and K. Yanai, "Real-time mobile food recognition system," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pp. 1–7, IEEE, 2013.
7. Y. Kawano and K. Yanai, "Rapid mobile object recognition using fisher vector," in *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*, pp. 476–480, IEEE, 2013.
8. M. Anthimopoulos, J. Dehais, P. Diem, and S. Mougiakakou, "Segmentation and recognition of multi-food meal images for carbohydrate counting," in *Bioinformatics and Bioengineering (BIBE), 2013 IEEE 13th International Conference on*, pp. 1–4, IEEE, 2013.
9. P. Pouladzadeh, S. Shirmohammadi, and T. Arici, "Intelligent svm based food intake measurement system," in *Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2013 IEEE International Conference on*, pp. 87–92, IEEE, 2013.
10. F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Computer Vision–ECCV 2010*, pp. 143–156, Springer, 2010.
11. Y. Kawano and K. Yanai, "Foodcam: A real-time food recognition system on a smartphone," *Multimedia Tools and Applications*, pp. 1–25, 2015.

[10] For a description of the experimental design, see the corresponding papers.

System		Food detection algorithms					Experiment	
Paper	Architecture	Acquisition	Processing	Segmentation	Features	Class.	Cat.	Success
[1]	Client-Server	1 picture	None	Manual clipping	BoF, Color, Gabor	SVM + MKL	50	61.34%
[2]	Client-Server	3 pictures, Voice, Checker box	None	None	Color, MR, Adaboost, Volume	SVM + χ^2	150	+90%
[3]	Client-Server	1 picture	None	Manual clipping	Bof, Color, Gabor, HoG	SVM + MKL	85	62.5%
[4]	Client-Server	3 pictures, checker board, credit card, OCR, user input	redundancy elimination - SIFT	fg-bg subtraction., template subtraction	SIFT, volume	local-global DB, K-means clustering	Not listed	92%
[5]	Client-Server	1 picture, checker board, user confirmation	Spatial and color calibration	fg-bg subtraction	Invariant Gabor	Not desc.	209	95%
[6]	Client	1 Picture, bounding box, food item selection	None	Bounding box	Bag-of-SURFS, RGB color, reliable direction	linear SVM	50	81.55%
[7]	Client	1 Picture, bounding box, food item selection	None	Bounding box	HoG, RGB color	linear SVM + Fisher vector	100	79.2%
[9]	Client	2 Pictures, users thumb	Cropping, padding	K-means clustering for color, texture segmentation	Color, texture, size, shape	SVM + RBF kernel	30	92.21%
[8]	Client	1 Picture, plate	CIELAB, pyramid mean-shift filtering, region growing	Region merging, bg-subtraction	Color, texture	Hierarchical k-means clustering	6	87%

Table 1: Comparison of main features of the system and their experimental results described in their papers. The first row presents the literature index, the architecture row display if the system is implemented solely on a mobile platform or with a help of a server, the next 5 rows show the basic properties of the food recognition stages. The last two rows display the results of the experiment in each paper. First the number of categories is listed and next the success rate the system achieved.