Dish Detection and Segmentation for Dietary Assessment on Smartphones

Joachim Dehais[1,2(✉)], Marios Anthimopoulos[1,3], and Stavroula Mougiakakou[1,4]

[1] ARTORG Center for Biomedical Engineering Research, University of Bern, Bern, Switzerland
{joachim.dehais,marios.anthimopoulos, stavroula.mougiakakou}@artorg.unibe.ch
[2] Graduate School of Cellular and Biomedical Sciences, University of Bern, Bern, Switzerland
[3] Department of Emergency Medicine, Bern University Hospital, Bern, Switzerland
[4] Department of Endocrinology, Diabetes and Clinical Nutrition, Bern University Hospital, Bern, Switzerland

Abstract. Diet-related chronic diseases severely affect personal and global health. However, managing or treating these diseases currently requires long training and high personal involvement to succeed. Computer vision systems could assist with the assessment of diet by detecting and recognizing different foods and their portions in images. We propose novel methods for detecting a dish in an image and segmenting its contents with and without user interaction. All methods were evaluated on a database of over 1600 manually annotated images. The dish detection scored an average of 99% accuracy with a .2s/image run time, while the automatic and semi-automatic dish segmentation methods reached average accuracies of 88% and 91% respectively, with an average run time of .5s/image, outperforming competing solutions.

Keywords: Diet assessment · Diabetes · Obesity · Image segmentation · Computer vision · Smartphone

1 Introduction

The epidemic spread of diet-related chronic diseases such as obesity and diabetes has severely affected public health on a global scale over the last decades. Diet management is key to prevent and treat such diseases; yet, traditional methods often fail because patients lack the motivation and skills to assess their food intake. This situation demands novel tools and services to provide automatic, personalized, and accurate diet assessment, which are now feasible thanks to powerful smartphones and the recent advances of computer vision. Recently, a plethora of solutions has been proposed for the automatic recognition of food images and the assessment of their nutritional content. Some methods classify food images directly into one meal type and then retrieve the nutritional content from databases. These methods address meals with specific composition and size (e.g. fast food restaurants) but are insufficient for meals with arbitrary content and portions. For such cases, another category of systems exists,

V. Murino et al. (Eds.): ICIAP 2015 Workshops, LNCS 9281, pp. 433–440, 2015.
DOI: 10.1007/978-3-319-23222-5_53

which first detect and segment food items, then classify them separately, and finally estimate food portions from their 3D shape. Here, we focus on the first two stages of such a system.

To deal with the automatic food detection/segmentation problem, the existing solutions make assumptions on the number, color, and shape of dishes in the image, as well as the possible number of food items in each dish and the visual properties of the background. Shroff et al. [1] assume disconnected food items and a background of uniform light color, making a simple adaptive thresholding method usable. He et al. [2] experimented with active contours [3], normalized cuts [4] and local variation [5], and concluded that the latter is the most suitable choice. In their experiments, multiple dishes were considered, and the background was defined as the region with the most frequent color. In [6], Zhu et al. used feedback from recognition to choose the number of segments in normalized cuts. Matsuda et al. [7] also considered multiple candidate food regions by using: (i) the whole image, (ii) a deformable part model [8], (iii) a circle detector based on Hough transform [9] and (iv) the JSEG segmentation method [10]. Then, all of the candidate regions are classified and a predefined number of them are kept according to their classification confidence, without providing their locations in the image. Bettadapura et al. [11] use the hierarchical image segmentation of [12] together with unspecified location heuristics and assumptions for segmentation. Puri et al. [13] consider a single circular dish, detected with a fixed time RANSAC circle detection method [14], and classify dense image patches to generate a segmentation map. Classification makes this approach inherently limited by the performance of the food patch recognizer. Anthimopoulos et al. [15] make similar assumptions and use a RANSAC-based ellipse detector for the dish, followed by mean-shift filtering in the CIELab color space for the segmentation of the contained food.

Semi-automatic methods have also been proposed to improve the accuracy of food detection and segmentation. Kawano et al. [16] assume multiple dishes containing single foods and ask the user to draw a box around each dish, then adjust the boxes using grab-cut [17] and recognize the content. Morikawa et al. [18] proposed a system where the user taps on each food item, providing seeds which then grow on square patches using RGB histogram similarity and empirically found thresholds. Oliveira et al. [19] also proposed a region-growing method in which the regions grow using spatial variation and average color similarity. Different color spaces are used along with experimentally found thresholds. From the multiple results the one with the best classification confidence is kept.

In this study we assume that the taken food image contains a single elliptical dish with possibly multiple food items and propose:

- A fast and robust dish detection method using multi-layered RANSAC
- A fast automatic method for food segmentation using non-parametric region growing/merging and CIELab color similarity
- A semi-automatic version of the segmentation that enhances the results with minimum user input

2 Proposed Methods

The proposed system first detects the dish based on its elliptic projection, then segments it and localizes the different food items. We propose two segmentation methods based on a region growing paradigm: one automatic, and one semi-automatic.

2.1 Dish Detection

The dish is detected through the combination of edge detection, grouping, and robust model fitting (Fig.1). For edge detection we use the Canny filter [20] on images downscaled to a height of 240px with their intensity histogram equalized (Fig. 1(b)). The edge components are further filtered by eliminating junctions between edge curves, sharp corners, and small segments (Fig. 1(c)). The classical robust estimator RANSAC [21] is modified to an equivalent of groupSAC [22], where groups of edge curves are randomly sampled and tested on the complete edge set. The size of the sampled groups starts from one curve and increases up to five as long as increasing the size results in finding an ellipse with more support. Whenever the algorithm finds an ellipse with more support, iterative local optimization (LO) [23] takes the inliers to the new best ellipse, and randomly samples them to generate new ellipses, potentially with larger support. (Fig. 1(d)).

(a) (b) (c) (d)

Fig. 1. Dish detection: (a) original image, (b) Canny edges, (c) filtered edge segments, (d) detected dish border

2.2 Segmentation

For the segmentation of the dish we adopt the Seeded Region Growing (SRG) method [24]: a fast nonparametric partitioning method for image segmentation. SRG iteratively expands seed regions until the image is fully covered. At each iteration, only the neighboring pixels of the seed regions are considered for expansion, and the one pixel with the lowest distance from its region is absorbed in it. Its neighbors are then added to the border list, to be considered in future iterations. Once all pixels have been added to a region, the process stops, and the image is segmented. As distance between a pixel and a region, we modify the CIE94 distance [25] on the CIELab color space, putting emphasis on the color component, hence reducing the effect of intensity changes often caused by shadows:

$$dist\left(L,a,b,L',a',b'\right)=\sqrt{|\Delta L_N|+\Delta C_N{}^2+\Delta H_N{}^2}, \text{ with} \tag{1}$$

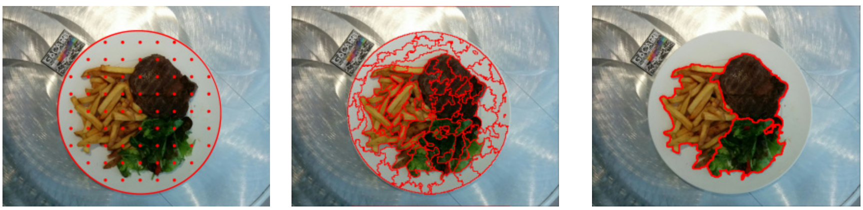
$$\Delta L_N=\left(L-L'\right), \tag{2}$$

$$\Delta C_N=(\sqrt{a^2+b^2}-\sqrt{a'^2+b'^2})/(1+0.045\sqrt{a^2+b^2}), \tag{3}$$

$$\Delta H_N=\sqrt{\left(a-a'\right)^2+\left(b-b'\right)^2-\Delta C_N{}^2}/(1+0.015\sqrt{a^2+b^2}) \tag{4}$$

Eq.1 is an asymmetric distance; to make it symmetric we apply it twice, with either the pixel or the region color as the origin, and the maximum of the two is used. The color of a region is defined as the median *L* and the average *a* and *b* values over all the region pixels. Pixels outside the dish are not considered while a small band of pixels on the inside of the dish border is automatically labeled as the dish seed in both the automatic and the semi-automatic version.

Automatic Segmentation. For the automatic segmentation, a fixed number of seeds are generated on a regular grid (Fig. 2(a)), and SRG grows these into small, consistent regions (Fig. 2(b)). The regions are then merged together iteratively using the Statistical Region Merging paradigm (SRM) [26] with a modified merging cost (Fig. 2(c)). SRM, like SRG, is a nonparametric aggregation process: in each iteration, the two regions with the smallest merging cost are merged, until a stopping criterion is met. The merging cost we use is the ratio of color distance (eq. 1) divided by square root of the edge length between two regions. The merging process continues until all regions are larger than a certain ratio of the dish area and all inter-region color distances are larger than a fixed threshold.



(a) (b) (c)

Fig. 2. Automatic segmentation: (a) grid seeds (b) grown regions (c) merged regions after removing the dish region

Semi-automatic Segmentation. In the semi-automatic version of the method the seed regions are given by the user. A dedicated smartphone interface displays the image and allows the user to draw paths over each food item by touching the screen. Fig. 3(a) shows an example input, where colored curves represents user-given seeds. The SRG algorithm grows these seeds as previously described, directly producing the final segmentation (Fig. 3(b)). Knowing the number and the approximate locations of the food items through the user seeds allows this method to outperform the automatic one, especially in difficult cases. The semi-automatic method is thus designed to be used whenever the automatic segmentation fails.

(a) (b)

Fig. 3. Semi-automatic segmentation: (a) user seeds (b) grown regions after removing the dish region

3 Experimental Results

3.1 Data and Experimental Setup

All proposed methods are evaluated on a dataset of 1620 meal images, with each image containing one fully visible dish. The meals were provided by the restaurants of the Bern university hospital, Inselspital, and photographed by us to obtain typical and complex cases. The dataset thus presents a large variety of food types, viewing angles, backgrounds and lighting conditions. Segmentation maps were manually created and provide the exact area of the dish and the different food items, where each food item is made of a single connected component. To test the semi-automatic segmentation, we used for seeds the pixels of each segment with a distance to the border in the top 3% (~.1% of the total region area).

Both the dish detection and segmentation are evaluated using region-based metrics similar to the Huang and Dom Index (HDI) [27]. Let $S = \{S_i\}_{i=1}^{m}$ and $T = \{T_i\}_{i=1}^{n}$ be two segmentations, where S_i (resp.T_i) is region i from segmentation S (resp. T) and m, n are the number of segments in S and T. We define two normalized directional indices based on worst and average segmentation performance:

$$NI_{min}(T{=>}S)=Min_i\left(\frac{Max_j(|S_i \cap T_j|)}{|S_i|}\right) \tag{5}$$

$$NI_{sum}(T{=>}S)=\frac{\Sigma_i Max_j(|S_i \cap T_j|)}{\Sigma_i |S_i|} \tag{6}$$

For each index, the indices for the two directions are combined in the harmonic mean to give the final two indices for the evaluation:

$$F_x=\frac{2*NI_x(T{=>}S)*NI_x(S{=>}T)}{NI_x(T{=>}S)+NI_x(S{=>}T)},\ x = \text{min or sum} \tag{7}$$

The background segment is excluded from the computation of the measures to make the results independent from the size of the dish. Average processing times are also provided; the experiments were conducted on an Intel i7-3770 CPU.

3.2 Results

Dish Detection. Because there is just one dish per image, indices (5) and (6) are equal when evaluating the dish detection. The average *F* score achieved by the proposed method is 99.1%, showing its robustness to different lighting and shooting conditions. The average processing time per image is 0.19 seconds, small enough to be used directly on mobile phones.

Table 1. Performance comparison for different color spaces and distances

Automatic	Average F_{min} (%)	Average F_{sum} (%)
RGB - Euclidian	45.7	61.1
CIE94 [25]	69.6	80.2
Proposed	80	88.2
Semi-Automatic	**Average F_{min} (%)**	**Average F_{sum} (%)**
RGB - Euclidian	57.9	72.3
CIE94[25]	66	78.2
Proposed distance	82.9	90.8

Segmentation. We first evaluate the influence of the color distance and the merging cost used for region growing and merging. As can be seen in Table 1, the proposed distance outperformed the RGB Euclidian distance by 20-30% and the CIE94 perceptual distance by nearly 10% for both automatic and semi-automatic methods and both evaluation metrics. Furthermore, Table 2 shows a considerable improvement for the automatic segmentation by including the length of the shared edge between segments in the merging cost.

Table 2. Comparison of merging cost

Merging Cost	Average F_{min} (%)	Average F_{sum} (%)
Color distance	74.7	85.8
Color distance/$\sqrt{\text{edge}}$	80	88.2

Table 3 compares the proposed food segmentation with methods from the related literature; for all methods, the true location of the dish was used to remove the background and dish segments. Flood fill corresponds to a version of region growing with a threshold on the maximal distance between a pixel and a region, similar to [18], but using the proposed distance and the dish seed (eq.1). For automatic segmentation the proposed system was closely followed by [15] in terms of accuracy, although the latter was more than four times slower. Local variation was less accurate and even slower, followed by ultrametric contours. For the semi-automatic case, the proposed method was about 1% better than flood fill, a non-negligible improvement considering the high accuracies of both methods.

Table 3. Comparison of segmentation methods

Automatic	Average F_{min} (%)	Average F_{sum} (%)	Time (s/image)
Proposed	80	88.2	0.45
Mean-shift [15]	78.2	87.5	2.1
Local Variation [5]	66.7	82.6	2.8
Ultrametric contours [12]	54.1	69.2	19
Semi-Automatic	**Average F_{min} (%)**	**Average F_{sum} (%)**	**Time (s/image)**
Proposed	82.9	90.8	0.49
Flood fill	81.2	89.9	0.52

4 Conclusion

We have presented methods to automatically and semi-automatically detect and segment food in images. The methods make limited assumptions: a single dish is present in the image, with circular or elliptic shape, and an arbitrary number of food items. First, the dish is detected and the different food items are segmented automatically, or with user interaction if needed. The average performance was 99% for the dish detection, 88% for the automatic segmentation and 91% for the semi-automatic, outperforming existing methods. The methods have been implemented in a client-server model, used remotely from a smartphone. Future work includes the extension of the methods to images with multiple dishes, the combination with 3D shape of the scene to improve the segmentation, and the porting of the methods to smartphones.

Acknowledgement. This work was funded in part by the Bern University Hospital "Inselspital" and the European Union Seventh Framework Programme (FP7-PEOPLE-2011-IAPP) under grant agreement n° 286408 [www.gocarb.eu].

References

1. Shroff, G, Smailagic, A., Siewiorek, D.P.: Wearable context-aware food recognition for calorie monitoring. In: 12th IEEE ISWC, pp. 119–120 (2008)
2. He, Y., Khanna, N., Boushey, C.J., Delp, E.J.: Image segmentation for image-based dietary assessment: a comparative study. In: IEEE ISSCS 2013
3. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. Int. J. Comput. Vis. **1**, 321–331 (1998)
4. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Tran. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Image segmentation using local variation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 98–104 (1998)

6. Zhu, F., Bosch, M., Khanna, N., Boushey, C.J., Delp, E.J.: Multiple Hypotheses Image Segmentation and Classification With Application to Dietary Assessment. IEEE J. Biomed. Health Inform. **19**(1), 377–388 (2015)
7. Matsuda, Y., Hoashi, H., Yanai, K.: Recognition of multiple-food images by detecting candidate regions. In: IEEE ICME, 2012, pp. 25–30 (2012)
8. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
9. Duda, R.O., Hart, P.E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures. Comm. ACM **15**, 11–15 (1972)
10. Deng, Y., Manjunath, B.S.: Unsupervised segmentation of color-texture regions in images and video. IEEE Trans. Pattern Anal. Mach. Intell. **23**(8), 800–810
11. Bettadapura, V., Thomaz, E., Parnami, A., Abowd, G.D., Essa, I.A.: Leveraging Context to Support Automated Food Recognition in Restaurants. In: WACV 2015, pp. 580–587
12. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2011)
13. Puri, M., Zhu, Z., Yu, Q., Divakaran, A., Sawhney, H.: Recognition and volume estimation of food intake using a mobile device. In: IEEE WACV, pp. 1–8 (2009)
14. Cai, W., Yu, Q., Wang, H., Zheng, J.: A fast contour-based approach to circle and ellipse detection. In: 5th IEEE WCICA (2004)
15. Anthimopoulos, M., Dehais, J., Diem, P., Mougiakakou, S.: Segmentation and recognition of multi-food meal images for carbohydrate counting. In: IEEE BIBE (2013)
16. Kawano, Y., Yanai, K.: FoodCam: A real-time food recognition system on a smartphone. Multimedia Tools and Applications, pp. 1–25 (2014)
17. Rother, C., Kolmogorov, V., Blake, A.: GrabCut: Interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. **23**, 309–314 (2004)
18. Morikawa, C., Sugiyama, H., Aizawa, K.: Food region segmentation in meal images using touch points. In: ACM Workshop on Multimedia for Cooking And Eating Activities (2012)
19. Oliveira, L., Costa, V., Neves, G., Oliveira, T., Jorge, E., Lizarraga, M.: A mobile, lightweight, poll-based food identification system. Pattern Recognition **47**, 1941–1952 (2014)
20. Canny, J.: A Computational Approach to Edge Detetion. IEEE Trans. Pattern Analysis and Machine Intelligence **8**(6), 679–698 (1986)
21. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM **24**(6), 381–395 (1981)
22. Ni, K., Jin, H., Dellaert, F.: Groupsac: efficient consensus in the presence of groupings. In: IEEE 12th International Conference on Computer Vision, pp. 2193–2200 (2009)
23. Chum, O., Matas, J., Kittler, J.: Locally optimized RANSAC. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 236–243. Springer, Heidelberg (2003)
24. Adams, R., Bischof, L.: Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence **16**(6), 641–647 (1994)
25. McDonald, R., Smith, K.J.: CIE94a new color difference formula. Journal of the Society of Dyers and Colourists **111**(12), 376–379 (1995)
26. Nock, R., Nielsen, F.: Statistical region merging. IEEE Trans. Pattern Analysis and Machine Intelligence **26**(11), 1452–1458 (2004)
27. Huang, Q., Dom, B.: Quantitative methods of evaluating image segmentation. International Conference on Image Processing **3**, 53–56 (1995)