Evaluating CNN-Based Semantic Food Segmentation Across Illuminants

Gianluigi Ciocca(✉), Davide Mazzini, and Raimondo Schettini

DISCo - Department of Informatics, Systems and Communication, University of Milano-Bicocca, Viale Sarca 336, 20126 Milan, Italy
{ciocca,mazzini,schettini}@disco.unimib.it

Abstract. In this paper we aim to explore the potential of Deep Convolutional Neural Networks (DCNNs) on food image segmentation where semantic segmentation paradigm is used to separate food regions from the non-food regions. Specifically, we are interested in evaluating the performance of an efficient DCNN with respect to variability in illumination conditions that can be found in food images taken in real scenarios. To this end we have designed an experimental setup where the network is trained on images rendered as if they were taken under nine different illuminants. We evaluate the food vs. non-food segmentation performance of the network in terms of standard Intersection over Union (IoU) measure. The results of this experimentation are reported and discussed.

Keywords: Semantic segmentation · Food analysis · Dietary monitoring · Convolutional Neural Network · Illuminants

1 Introduction

The problem of healthy and balanced meal is being seriously tackled by the different health agencies with the aim at reducing obesity and unbalanced nutrition. For example, the Department of Health of the Italian Government promoted an extensive campaign for food and nutrition education[1]. The Department of Health of the Australian Government, compiled a very detailed report with guidelines for healthy foods in school canteens[2]. In Japan, the Ministry of Education, Science and Culture, the Ministry of Health and Welfare, and the Ministry of Agriculture, Forestry and Fisheries developed dietary guidelines with the aim of promoting better dietary patterns[3]. Similar actions can be found across many other countries (e.g. UK[4], USA[5], etc...).

The significant increase rates of obesity and diabetes for both adults and children make it extremely important to devise ways for accurate tracking of

[1] http://www.salute.gov.it/imgs/c_17_pubblicazioni_1248_allegato.pdf.
[2] https://education.nt.gov.au/policies/canteen-nutrition-and-healthy-eating.
[3] http://www.maff.go.jp/j/syokuiku/pdf/yo-ryo-.pdf.
[4] http://www.schoolfoodplan.com/actions/school-food-standards/.
[5] http://www.fns.usda.gov/school-meals/child-nutrition-programs.

S. Tominaga et al. (Eds.): CCIW 2019, LNCS 11418, pp. 247–259, 2019.
https://doi.org/10.1007/978-3-030-13940-7_19

nutritional intakes. The conventional way to track food intake has been carried out by exploring manually recorded logs on a daily basis, which is error prone due to delayed reporting, inability to estimate the food type and quantity. More accurate and user-friendly solutions can be achieved by taking advantage of technology, e.g one can simply take a picture of a plate of food using a smartphone and corresponding calorie of the food can be calculated automatically by employing computer vision techniques. In the recent years many research works have demonstrated that machine learning and computer vision techniques can help to build systems to automatically recognize diverse foods and to estimate the food quantity and calories [2,10,15,19,20,24,29,30].

Fig. 1. A general workflow of automatic dietary monitoring.

As shown in Fig. 1, a general work-flow of automatic dietary monitoring a number of computational tasks need to be accomplished. Initially, a given image is segmented in order to locate the boundaries of the food regions (*food segmentation*). Then, each food region is processed to identify the depicted food (*food recognition*). Finally, the quantity of each recognized food is estimated (*quantity estimation*) which paves the way towards calorie measurement. It is undeniable that accurately accomplishing each of these tasks is very important to achieve a well-working dietary monitoring system. In addition, food segmentation has to be used as a pre-processing step for recognition images with multiple food items.

It is clear that any mistakes, e.g under-segmentation or over-segmentation may influence the performance of the other tasks which would inevitably yield to incorrect calorie calculation. Moreover, if the dietary monitoring system is aimed at supporting the user in logging its food consumption in real scenarios, the food segmentation task must be robust to different image acquisition conditions such as changes in illumination, different point of views or different cameras.

Food analysis is a particularly challenging task since food is characterized by an intrinsic high intra-class variability. For instance, the same food can have very diverse visual appearance in the images due to slightly different preparations, presentations, and background. At the same time, we can have a low inter-class variability in the food appearance: different food may have extremely similar visual characteristics that make them difficulty to disambiguate. Figure 2 shows some examples of these two problems.

Due to their robustness with respect to data variability and thanks to their ability to extract and process relevant information within the images, Deep Convolutional Neural Networks (DCNNs) have been effectively used in different computer vision tasks. For these reason, in this paper we aim to explore the potential

Fig. 2. Top row: examples of high intra-class variability in visual appearance in food images ("Curry rice"). Bottom row: examples of low inter-class variability ("Fried noodles" vs. "Spaghetti", "Donuts" vs. "Bagel", and "French fries" vs. "Poutine").

of DCNNs on food and non-food image segmentation where semantic segmentation paradigm is used to separate food regions from the non-food regions. The results of this task could be leveraged both for a food recognition task and for a quantity and calories estimation task. Here we are interested in evaluating the performance of an efficient DCNN (GUNet [27,28]) with respect to variability in illumination conditions that can be found in food images taken in real scenarios. We want to investigate if and how we can cope with these variability at training time. To this end we have designed an experimental setup to evaluate a fast DCNN network trained on datasets of images rendered using different illuminants in a similar way as in [7]. We choose GUNet since, differently from other more complex and deep networks, it could be used also in mobile applications.

The paper is organized as follows. In Sect. 2 we briefly revise the state of the art in food segmentation. In Sect. 3, we describe our proposed approach for food segmentation. In Sect. 4 we present the segmentation results on a newly annotated food dataset. Finally, Sect. 5 concludes the paper.

2 Related Works

In this Section, we describe recent works dealing with the problem of food segmentation both using conventional approach as well as based on DCNNs.

Among the approaches using conventional image segmentation techniques adopted to food images we can find the work by He et al. [21] which evaluate two segmentation algorithms exploiting normalized cut (NCut) and local variation. Based on the experiments, local variation is found to be more stable among the two algorithms. In [15,16] JSEG segmentation algorithm [18] is used to locate food regions in images acquired in a canteen environment. Coupled with ad-hoc post-processing (given the image domain) of the resulting segmented images it is able to achieve very good performances. In [3], the JSEG-based food segmentation approach is further analyzed with extensive experiments evaluating different color spaces and algorithm's parameters used for the segmentation. Results show that, segmentation results can be improved with respect to the base algorithm.

In order to avoid the high computational cost of normalized cut-based segmentation algorithm in particular when applied per pixel on large images, Wang et al. [32] combine NCut with image superpixels. Superpixels are computed using the SLIC algorithm [1] and rely on color and texture cues. Results obtained on food and non food datasets, show that the proposed method exhibits a fast computation and an efficient use of memory.

Precise automatic food segmentation is a difficult task, in particular if it need to be performed on images acquired in the wild. For this reason, Inunganbi et al. [22] propose an interactive segmentation approach instead. They approach the segmentation task as a two class classification problem (i.e. food vs. non-food). Segmentation is performed using Random Forest ensemble learning, and boundary detection & filling and Gappy Principal Component Analysis methods are applied to refine the initial segmentation result.

The stunning success of DCNNs for image classification, encouraged the use of these techniques for other computer vision tasks as well. In particular they have been successfully employed for the semantic segmentation task where they are able to predict the category label of each image pixel among a set of given ones. However, properties that make DCNNs advantageous for classification tasks, i.e. robustness to spatial transformations and ability to learn increased abstraction of data, impede accuracy of the system for segmentation tasks where precise localization is desired rather than abstraction of spatial details [13]. For this reason, DCNNs that must perform image segmentation need to be designed with operations (i.e. layers) specifically tailored for the segmentation task as in FCN [25], DeepLab [13], SegNet [5] and GUNet [28] networks.

In the context of food segmentation, DCNNs have been used by Dehais et al. [17] to segment the food of already detected dishes in an image. The method combines region growing/merging techniques with a deep CNN-based food border detection to detect food regions outlines.

[30] is the first work accomplishing semantic segmentation specifically for food images by employing DCNNs. Specifically, the network architecture used is based on the earliest version of DeepLab [12]. This model uses a CNN to provide the unary potentials of a CRF, and a fully connected graph to perform edge-sensitive label smoothing (as in bilateral filtering). The network is initialized on ImageNet and fine-tuned on a newly annotated dataset of more than 12,000 images: Food201-segmented.

Bolanos et al. [9] Bolanos et al. employed DCNNs to simultaneously perform food localization and recognition. First, the method produces a food activation map on the input image for generating bounding boxes proposals of food regions. Then, each proposal is processed to recognize food types or food-related objects present in each bounding box. A similar approach is employed by Wang et al. [33]. The segmentation method uses the class activation maps and an adapted VGG-16 architecture to perform weakly supervised learning. The network is trained on food datasets as a top-down saliency model, and a Global Average Max Pooling (GAMP) layer is introduced into the architecture. The activation

maps are used as constraints in biased normalized cut. The final segmentation masks are obtained by binarizing the biased normalized cut.

In order to be effective in the context of dietary monitoring in real scenarios (i.e. in the wild), food segmentation algorithms must be robust to changes in illumination conditions. Since the image acquisition environment is mostly uncontrolled we can have large variations in illuminants. A network trained on pristine image datasets could have low performances in real usage. For this reason, large, heterogeneous and representative, for the task at hand, food image datasets are crucial for the design of effective methods for food recognition, segmentation and quantity estimation. One of the largest dataset for food segmentation is Food201-segmented of [30]. Unfortunately the dataset is not publicly available. A smaller but available segmented dataset is, for example, the UNIMIB2015 and UNIMIB2016 datasets [15,16] created collecting tray images in a canteen environment. Finally, if the application require that the processing is to be performed on mobile devices (i.e. acquisition and analysis) in order to reduce the bandwidth consumption, it is essential that the DCNN used for segmentation must be light and thus requiring few operations (see [6] for a benchmark of different DCNN architectures).

3 Experimental Setup

For our experiments we adopted the Food50M described in Sect. 3.1. We performed our experiments using an efficient network named GUNet [28] which is presented more in details in Sect. 3.2. Every model is evaluated by means of the pixel-based IoU measure computed as:

$$IoU = \frac{groundtruth \cap prediction}{groundtruth \cup prediction} \tag{1}$$

this formula can be alternatively written as:

$$IoU = \frac{TP}{FP + TP + FN} \tag{2}$$

where TP, FP and FN represent True Positive, False Positive and False Negative pixels respectively. For every experiment we report the IoU measure computed independently for every class. We also report the mean of the food and non-food classes as a more synthetic indicator of the performance of the whole segmentation method.

3.1 Food Image Dataset

Available large food image datasets with segmentation information are scarce. The largest dataset in the literature is the one introduced in [30] but unfortunately is not publicly available. Other image datasets are either small of do not have a pixel-based segmentation annotation. For these reason, we decide to use

the Food-50 dataset [23] as a base to create our own segmentation datasets. With respect to the available UNIMIB2015 [15] and UNIMIB2016 [16] datasets, Food-50 is large enough to be used for training and validation and is more diverse. Food-50 contains 5,000 images divided into 50 food classes. We manually annotated the regions of all the food items present in the images. This mean that each food belonging to one of the 50 classes as well as any other food found, such as vegetable in side dishes or used for presentation, was annotated as "food". We call this food image dataset *Food50Seg*. Figure 3 shows some examples of images in the dataset with the corresponding pixel-based annotation.

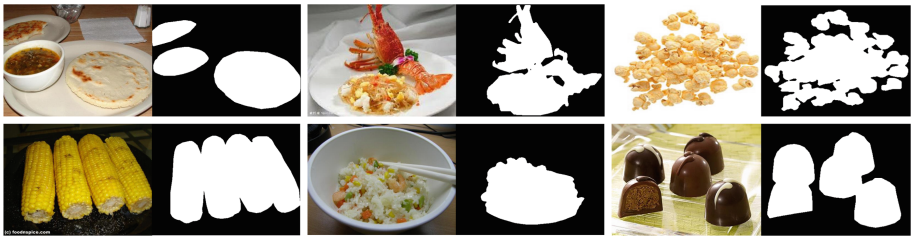

Fig. 3. Some segmentation examples of our Food50Seg food image dataset. Top row: "Arepas", "Lobster", and "Popcorn". Bottom row: "Corn", "Fried rice", and "Chocolate".

The images in the Food50 dataset were collected on the Internet from different sources. However, all the images seem to have been acquired using different cameras and mobile devices. They exhibit no evident illuminant cast so we can assume that a form of white balancing has been applied by the imaging devices. Moreover, we can assume the images use the sRGB color space, and thus are rendered under CIE D65 illuminant. To understand the effect of lighting variations in the context of semantic food segmentation, we modified the Food-50 dataset by artificially changing the illumination conditions of the images. In particular we follow a procedure similar to [7] by using nine blackbody radiators with color temperature from 3000 K to 8000 K with step of 1000 K, and with color temperature of 10000 K, 12000 K, and 14000 K. We call this dataset Food50M. Figure 4 shows an image rendered using the nine illuminants considered.

3.2 Network Architecture

For our experiments we adopted the GUNet architecture presented for the first time by Mazzini in [27]. In particular we rely on the improved version with the Guided Upsampling Module [28]. This network is designed for real-time applications and thus its main characteristic is the low inference time. The reasons that lead us to employ this architecture are twofold: first, this model can be employed on mobile devices for dietary monitoring applications in real conditions. Second, the architecture is also very fast at train time allowing us to train a relatively high quantity of models for our experiments on a single Titan Xp GPU.

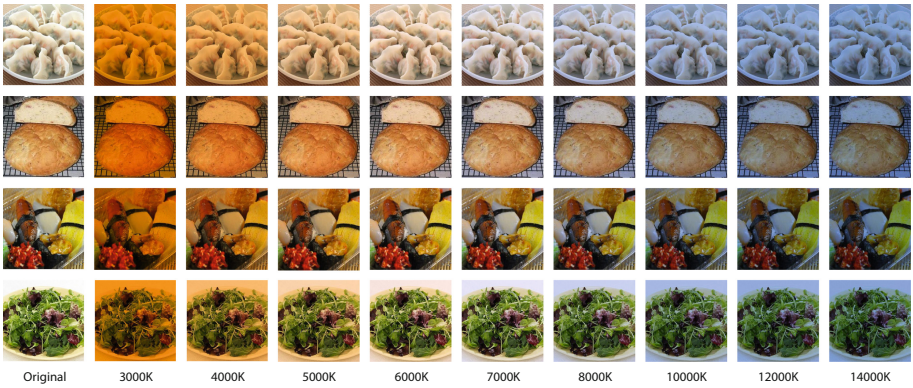

Fig. 4. An example of a food image in the Food50M dataset rendered using the nine illuminants considered. From top to bottom: "Dumplings", "Bread", "Sushi", and "Salad".

The network follows an encoder-decoder design. The encoder has a multi-branch structure to encode features at multiple resolutions: the first branch is the deepest one, encoding the most abstract features whereas the second branch is shallower by design in order to encode fine details without being too computational heavy. The first part of the decoder is a Fusion Module. It is composed by a first part where signals are pre-processed independently followed by a second part where signals are jointly processed and information coming from multiple resolution branches is fused together. The Decoder ends with a layer named Guided Upsampling Module that efficiently upsample the feature map. In this way the network is able to produce a semantic segmentation map of the same resolution of the network input.

The network exhibits a total of 19 Millions of parameters and requires 58,7 GFLOP to perform single inference on a 512×1024 image. Most parameters are in the Encoder part.

3.3 Training Recipe

All the network configurations in this paper have been trained with Stochastic Gradient Descent (SGD) plus momentum. Following [28], we set the base learning rate to 0.001 and trained for 150 epochs. We adopted a *fixed step* learning rate policy. The initial value is decreased two times by a order of magnitude (at 100 and 200 epochs).

4 Experimental Results

4.1 Assessing Robustness to Illuminants

With a first cluster of experiments, we want to assess if our DCNN model trained on Food50 training-set is robust to changes of the illuminant in the scene. The

same network, i.e. GUNet [28] described in Sect. 3.2, is trained only once on the original dataset and tested over nine modified versions of the test set with illuminants. Numerical results for this experiment are reported in Table 1 in terms of IoU measure vs illuminant color. The first column reports the network performance on the original test set. It is clearly noticeable a performance degradation on all the modified test sets with respect to the baseline. In particular the IoU levels decay with higher intensity for extreme cast values. The highest IoU values are obtained between 7000 K and 8000 K suggesting that the distribution of illuminants in the original image set is centered on such values (consider that the daylight temperature is 6500 K).

Table 1. Results of our DCNN model tested on different illuminants. The first column no-cast reports the results on the original test set.

Tested on											
			no-cast	3000 K	4000 K	5000 K	6000 K	7000 K	8000 K	10000 K	12000 K	14000 K
Trained on no-cast	IoU(%)	food	69.9	0.5	18.8	24.7	32.0	41.0	39.3	3.8	0.3	0.1
		non-food	87.0	87.0	74.3	71.0	73.2	74.3	75.5	77.2	77.2	71.4
		mean	78.5	35.7	46.0	49.5	53.7	59.0	58.2	37.6	35.6	35.6

By looking at detailed results (i.e. first two rows) in Table 1 we can observe an interesting detail: IoU values for the non-food class remains quite high for every color cast whereas the real performance decrease is remarkable only for the food class. This might be related to the fact that the non-food images in our test set have been collected "in the wild" and thus they exhibit a wider range of illuminants. The trained model is thus more robust to illuminant changes. In our next experiments we will augment the training set with images with illuminants to measure if the model can acquire a certain degree of robustness to this types of image transformations.

4.2 Augmented Training Set

With this set of experiments, we want to assess the behaviour of the DCNN if trained on an augmented training set and observe the difference with the same model trained on the natural dataset. In detail, we want to test if the model benefits from being trained on a dataset with a specific illuminant and compare it with a model which is trained on the union set of all the illuminants. In Table 2 are reported the results for three clusters of experiments: the first three lines represent our baseline from Table 1 (i.e. a single model trained on the original dataset).

The second cluster of experiments is exposed from line 4 to 6. Every column represents the result obtained by training and testing a model on *that* specific illuminant. Notice that the first and the last column are missing because they represent redundant data: the first column represents the model trained and tested on the no-cast dataset, which is shown in the first three rows of the table.

Table 2. Results of three different train setups tested against various illuminant casts. First 3 rows: the model is trained on the original test set. Line 4 to 6: the model is trained on the specific illuminant only. Last 3 rows: the training set is augmented with all the illuminants.

Tested on												
			no-cast	3000 K	4000 K	5000 K	6000 K	7000 K	8000 K	10000 K	12000 K	14000 K	all ill.
Trained on	original train set	food	69.9	0.5	18.8	24.7	32.0	41.0	39.3	3.8	0.3	0.1	23.5
		non-food	87.0	71.0	73.2	74.3	75.5	77.2	77.2	71.4	71.0	71.1	74.3
		mean	78.5	35.7	46.0	49.5	53.7	59.0	58.2	37.6	35.6	35.5	48.9
	specific illuminant	food	-	60.5	71.0	71.0	73.4	68.4	74.0	73.1	74.2	76.1	-
		non-food	-	74.2	84.0	84.2	89.2	82.7	89.2	87.8	88.0	89.0	-
		mean	-	67.3	77.5	77.6	81.3	75.5	81.6	80.5	81.1	82.6	-
	all illuminants	food	71.9	68.3	70.8	71.2	71.2	71.1	71.1	70.7	69.8	69.5	70.5
		non-food	86.2	83.6	85.5	85.9	85.8	85.7	85.9	85.6	85.2	85.1	85.5
		mean	79.0	76.0	78.2	78.5	78.5	78.4	78.5	78.2	77.5	77.3	78.0

Missing data in the last column represents the model trained and tested on all illuminants, which is presented in the last column. We augmented the original training set with images rendered with that illuminant. The test set, like in all the others experiments, consists of images with a single modified illuminant. The performance raises dramatically with respect to the baseline. Most benefit is relative to the food class.

The last 3 lines of Table 2 represent the results of a single model trained on a set composed of the union set of all the illuminants. Surprisingly, for some illuminants this is the best model outperforming even the models trained on specific illuminants. This suggest that the DCNN model benefits from training on a higher number of images with higher variability. We suppose that the model trained on different illuminants generalize better because it become able to discriminate food from non food areas, abstracting from the type of illuminant in the scene.

In Table 2, the first column indicates tests made on the original test set without any illuminant cast. The last column indicates a test set composed by the union of all the test sets with illuminant casts. The model trained on all the illuminants exhibits an interesting behaviour even in this two cases. With respect to the no-cast set it behaves slightly better than the baseline model, suggesting that this for of data augmentation do not hurt the performance on the original data, i.e. the transformations are balanced and they do not introduce a bias in the data. Concerning the all-illuminant test set it shows a similar performance with respect to the no-cast set. Notice that the performance of the model trained only on the original data are visibly degraded.

In the last cluster of experiments we test the networks trained on specific illuminants against the original test set. With these experiments we want to verify if models trained on those augmented datasets maintain an acceptable segmentation performance even in the original test set not affected by illumination cast. Results are shown in Table 3. For the model trained on the 3000 K

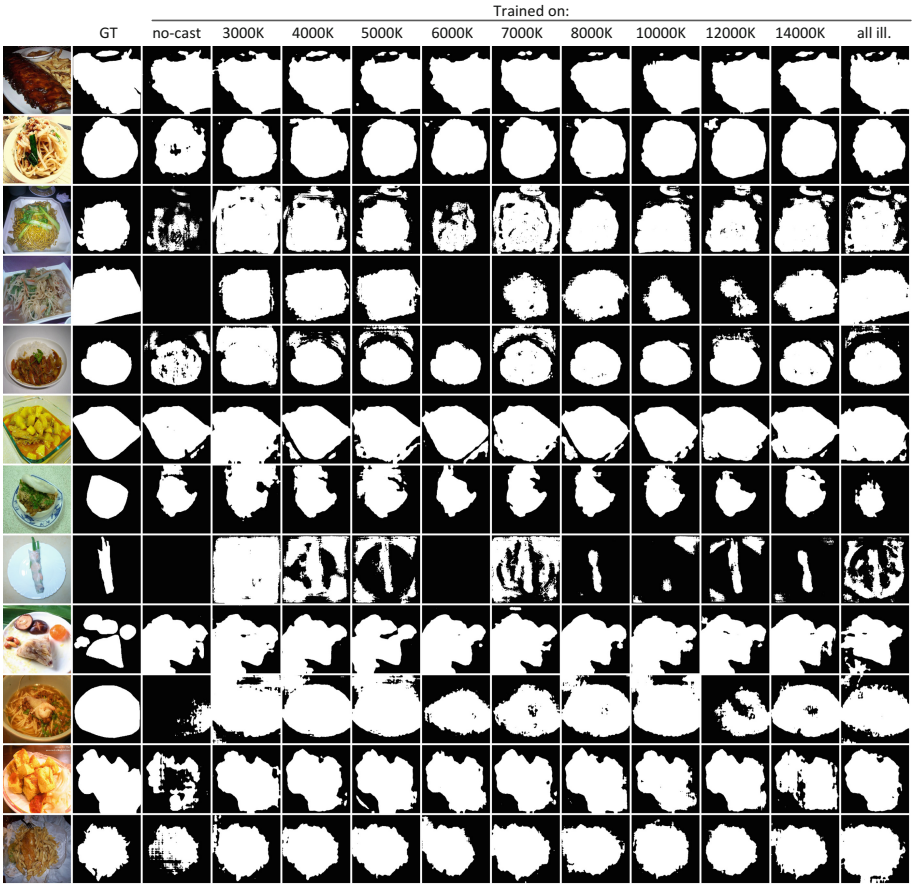

Fig. 5. Some segmentation results obtained using the different training strategies. no-cast denotes that the training is performed on the original data. "all ill." denotes that all the illuminants have been used during training. Images have been resized for display.

illuminant the mean IoU si quite lower than the baseline no-cast. This is probably due to the fact that 3000 K is an extreme illuminant color cast and such cast is not present in any of the original test set images. Surprisingly for some other illuminants like 14000 K or 8000 K the specific illuminant train setting is beneficial even when testing on the original set. However these numbers are not significantly higher and could be due the intrinsic randomness when initializing CNN models. As a general consideration we can state that the model do not exhibit evident performance degradation.

Figure 5 shows some visual results obtained using the different training strategies. The no-cast column represents the output of the model trained only on the original data. "all ill." denotes the output of the model trained on the union set

Table 3. Results of our DCNN model trained on different illuminants and tested on the original test set. The first column no-cast represents the results of the model trained on the original data.

Trained on												
			no-cast	3000 K	4000 K	5000 K	6000 K	7000 K	8000 K	10000 K	12000 K	14000 K	all ill.
Tested on no-cast	IoU (%)	food	69.9	60.5	71.0	71.0	73.4	68.4	74.0	73.1	74.2	76.1	71.9
		non-food	87.0	74.2	84.0	84.2	89.2	82.7	89.2	87.8	88.0	89.0	86.2
		mean	78.5	67.3	77.5	77.6	81.3	75.5	81.6	80.5	81.1	82.6	79.0

of every illuminant cast. The no-cast model clearly produces the worst visual results. In some images (i.e. rows 4,8) it doesn't even detect anything at all. Furthermore the three bottom images are clearly under-segmented. Every other method achieve overall better segmentation results. This is in line with what emerges from Table 3.

5 Conclusions

In this work we explored the potential of Deep Convolutional Neural Networks on food image segmentation to discriminate food regions from the background. In particular, we evaluated the performance of an efficient DCNN with respect to the variability in illumination conditions on real scene images. We built a new dataset named Food50M where images from the train and test set have been modified with nine different illuminants. Results show that the network trained on pristine data is not able to cope with strong illuminant shifts. By training and testing the model on a specific illuminant, performance improves with respect to the baseline as expected. Using all illuminants in training we are able to further improve the IoU measure. This demonstrates that we can effectively empower the model robustness exploiting an augmented training set composed by images with simulated illuminants. As future works we plan to evaluate color constancy algorithms or filter removal methods as pre-processing steps during the training phase, both traditional as well as CNN-based such as [8,11,14,31]. Moreover, our food/non-food segmentation approach can be coupled with a semantic food segmentation technique such as [4,13,26] in order to recognize food-specific regions. This will allow to implement a system able to recognize food and estimate their properties such as quantity, calories, etc...

Acknowledgements. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research. Published in the context of the project FooDesArt: Food Design Arte - L'Arte del Benessere , CUP (Codice Unico Progetto - Unique Project Code): E48I16000350009 - Call "Smart Fashion and Design", cofunded by POR FESR 2014–2020 (Programma Operativo Regionale, Fondo Europeo di Sviluppo Regionale - Regional Operational Programme, European Regional Development Fund).

References

1. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., et al.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
2. Anthimopoulos, M.M., Gianola, L., Scarnato, L., Diem, P., Mougiakakou, S.G.: A food recognition system for diabetic patients based on an optimized bag-of-features model. Biomed. Health Inform. **18**(4), 1261–1271 (2014)
3. Aslan, S., Ciocca, G., Schettini, R.: On comparing color spaces for food segmentation. In: Battiato, S., Farinella, G.M., Leo, M., Gallo, G. (eds.) ICIAP 2017. LNCS, vol. 10590, pp. 435–443. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70742-6_42
4. Aslan, S., Ciocca, G., Schettini, R.: Semantic food segmentation for automatic dietary monitoring. In: IEEE 8th International Conference on Consumer Electronics, Berlin (ICCE-Berlin), pp. 1–6 (2018)
5. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)
6. Bianco, S., Cadene, R., Celona, L., Napoletano, P.: Benchmark analysis of representative deep neural network architectures. IEEE Access **6**, 64270–64277 (2018)
7. Bianco, S., Cusano, C., Napoletano, P., Schettini, R.: On the robustness of color texture descriptors across illuminants. In: Petrosino, A. (ed.) ICIAP 2013, Part II. LNCS, vol. 8157, pp. 652–662. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41184-7_66
8. Bianco, S., Cusano, C., Piccoli, F., Schettini, R.: Artistic photo filter removal using convolutional neural networks. J. Electron. Imaging **27**(1), 011004 (2017)
9. Bolanos, M., Radeva, P.: Simultaneous food localization and recognition. In: 23rd IEEE International Conference on Pattern Recognition (ICPR), pp. 3140–3145 (2016)
10. Bosch, M., Zhu, F., Khanna, N., Boushey, C., Delp, E.: Combining global and local features for food identification in dietary assessment. In: 18th IEEE International Conference on Image Processing (ICIP), pp. 1789–1792 (2011)
11. Buzzelli, M., van de Weijer, J., Schettini, R.: Learning illuminant estimation from object recognition. In: 25th IEEE International Conference on Image Processing (ICIP), pp. 3234–3238 (2018)
12. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Semantic image segmentation with deep convolutional nets and fully connected crfs. In: International Conference on Learning Representations (2015)
13. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2017)
14. Ciocca, G., Marini, D., Rizzi, A., Schettini, R., Zuffi, S.: Retinex preprocessing of uncalibrated images for color-based image retrieval. J. Electron. Imaging **12**(1), 161–172 (2003)
15. Ciocca, G., Napoletano, P., Schettini, R.: Food recognition and leftover estimation for daily diet monitoring. In: Murino, V., Puppo, E., Sona, D., Cristani, M., Sansone, C. (eds.) ICIAP 2015. LNCS, vol. 9281, pp. 334–341. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23222-5_41
16. Ciocca, G., Napoletano, P., Schettini, R.: Food recognition: a new dataset, experiments and results. IEEE J. Biomed. Health Inform. **21**(3), 588–598 (2017)

17. Dehais, J., Anthimopoulos, M., Mougiakakou, S.: Food image segmentation for dietary assessment. In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management, pp. 23–28. ACM (2016)
18. Deng, Y., Manjunath, B.: Unsupervised segmentation of color-texture regions in images and video. IEEE Trans. Pattern Anal. Mach. Intell. **23**(8), 800–810 (2001)
19. Ege, T., Yanai, K.: Simultaneous estimation of food categories and calories with multi-task CNN. In: Fifteenth IAPR International Conference on Machine Vision Applications (MVA), pp. 198–201. IEEE (2017)
20. Ege, T., Yanai, K.: Multi-task learning of dish detection and calorie estimation. In: Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management, pp. 53–58. ACM (2018)
21. He, Y., Khanna, N., Boushey, C.J., Delp, E.J.: Image segmentation for image-based dietary assessment: a comparative study. In: IEEE International Symposium on Signals, Circuits and Systems (ISSCS), pp. 1–4 (2013)
22. Inunganbi, S., Seal, A., Khanna, P.: Classification of food images through interactive image segmentation. In: Nguyen, N.T., Hoang, D.H., Hong, T.-P., Pham, H., Trawiński, B. (eds.) ACIIDS 2018, Part II. LNCS (LNAI), vol. 10752, pp. 519–528. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75420-8_49
23. Joutou, T., Yanai, K.: A food image recognition system with multiple kernel learning. In: 16th IEEE International Conference on Image Processing (ICIP), pp. 285–288. IEEE (2009)
24. Kawano, Y., Yanai, K.: Foodcam: a real-time food recognition system on a smartphone. Multimed. Tools Appl. **74**(14), 5263–5287 (2015)
25. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
26. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
27. Mazzini, D., Buzzelli, M., Pauy, D.P., Schettini, R.: A CNN architecture for efficient semantic segmentation of street scenes. In: IEEE 8th International Conference on Consumer Electronics, Berlin (ICCE-Berlin), pp. 1–6 (2018)
28. Mazzini, D.: Guided upsampling network for real-time semantic segmentation. In: British Machine Vision Conference, BMVC 2018, Northumbria University, Newcastle, 3–6 September 2018, p. 117 (2018)
29. Mezgec, S., Koroušić Seljak, B.: Nutrinet: a deep learning food and drink image recognition system for dietary assessment. Nutrients **9**(7), 657 (2017)
30. Myers, A., et al.: Im2calories: towards an automated mobile vision food diary. In: IEEE International Conference on Computer Vision (ICCV), pp. 1233–1241 (2015)
31. Rizzi, A., Gatta, C., Slanzi, C., Ciocca, G., Schettini, R.: Unsupervised color film restoration using adaptive color equalization. In: Bres, S., Laurini, R. (eds.) VISUAL 2005. LNCS, vol. 3736, pp. 1–12. Springer, Heidelberg (2006). https://doi.org/10.1007/11590064_1
32. Wang, Y., Liu, C., Zhu, F., Boushey, C.J., Delp, E.J.: Efficient superpixel based segmentation for food image analysis. In: IEEE International Conference on Image Processing (ICIP), pp. 2544–2548. IEEE (2016)
33. Wang, Y., Zhu, F., Boushey, C.J., Delp, E.J.: Weakly supervised food image segmentation using class activation maps. In: IEEE International Conference on Image Processing (ICIP), pp. 1277–1281. IEEE (2017)