Recognition and Localization of Food in Cooking Videos

Nachwa Aboubakr, Rémi Ronfard, James L. Crowley

Recognition and Localization of Food in Cooking Videos

Nachwa Aboubakr[1] Remi Ronfard[2] James Crowley[1]
firstname.lastname@inria.fr

[1] Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP*, Laboratoire LIG
[2] Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP*, Laboratoire LJK

1 Abstract

In this paper, we describe experiments with techniques for locating foods and recognizing food states in cooking videos. We describe production of a new data set that provides annotated images for food types and food states. We compare results with two techniques for detecting food types and food states, and then show that recognizing type and state with separate classifiers improves recognition results. We then use this to provide detection of composite activation maps for food types. The results provide a promising first step towards construction of narratives for cooking actions.
Keywords— Object localization, Weakly supervised learning

2 Introduction

This paper reports on experiments with detection and localization of food types and food states. This is a first step toward automatic construction of cooking narratives (causal sequences of events) for food preparation that can be used to explain how food was prepared. This is a challenging problem for computer vision given the large variety of appearance for food, and the semi-structured nature of manipulation actions.

Most work on recognition of cooking activities has concentrated on recognizing actions and activities from the spatio-temporal patterns of hands motion [1]. While some cooking activities may be directly recognized from motion, the resulting description is incomplete, as it does not describe the state of the ingredients, and how these have been transformed by cooking actions. We believe that a fuller description requires a description of how food ingredients have been transformed during the cooking process.

We propose to address the construction of cooking narratives by first detecting and locating ingredients and tools, then recognizing actions that involve transformations of ingredients, such as "dicing tomatoes", and use these transformations to segment the video stream into visual events. We can then interpret detected events as a causal sequence of voluntary actions, providing a narrative for the implementation of the recipe.

We use the term "food type" to refer to specific foods such as tomatoes or cucumbers. We use the term "food state" to refer to the shape and/or physical appearance of a food type as it undergoes preparation. For example, sliced, diced and peeled are food states.

Many common food types may have a variety of shapes. Changes in food state can also entail dramatic change in visual appearances, as well as changes to geometry and even topology. This introduces an inter and intra variability to ingredients which in turn adds complexity to the recognition task.

Recent progress in machine learning [2] has provided techniques that can be used to detect and locate tools and food. However, such techniques require a large number of annotated images. Unfortunately, none of the commonly available data sets for food provides images or annotations for different food states. To remedy this situation, we have created a new annotated dataset from Google Images, using food types and food states as keywords for queries. We use this dataset to fine-tune a pre-trained model with the weakly-supervised learning technique of Zhou et al. [3]. We use the resulting activation maps to train a new layer which recognizes food states and food types simultaneously.

Section 2 discusses the problem of recipe recognition and reviews previous work on recognition of cooking activities, as well as available data sets, and discusses the problem of weakly supervised learning of techniques for food localization. Section 3 describes the problem of learning food concepts. We discuss two methods for combining concepts for localization and derive two competing techniques. In section 4 we present the results of experiments showing that learning separate layers for food types and food states results in improved detection and localization of composite food classes.

3 Related work

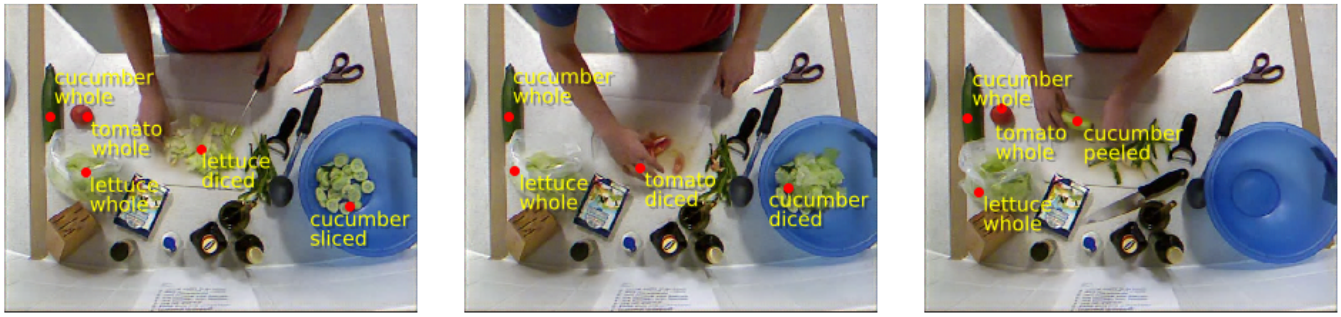
In this section, we review existing methods for cooking action recognition, existing datasets of cooking videos and recipes, and existing methods for weakly-localizing objects in images.

3.1 Recipe recognition

Recipe recognition can refer to at least two different problems. In image analysis, recipe recognition refers to the problem of recognizing recipes from images of the final dish. This is an object recognition problem that requires recognizing the ingredients used in the recipe from images of the final food state. Methods have been proposed to address this problem using multi-modal data (image and text in [4], context information [5]) and by multi-labeling visible ingredients for each dish [6]. In video analysis, recipe recognition is the problem of recognizing cooking activities involved in the preparation of a dish. This is commonly considered to be an action recognition problem where the task is to recognize the state-changing ingredients and the instructions in the recipe from spatio-temporal action recognition.

Recipes are sequence of instructions that can be performed in chronological order or simultaneously. Recipes are typically represented as semi-structured sequences of instructions, each of which consists of an operation that transforms an ingredient to a different state. This can be done using de-

[1] Institute of Engineering Univ. Grenoble Alpes



(A) after dicing lettuce (B) after dicing tomato (C) after peeling cucumber

Figure 1: Examples of food localization in key-frames of 50 salads dataset. Better viewed in colors.

Dataset	Recipes	Actions	Food types	Food states
MPII Cooking v2[10]	36 activities	Yes	None	None
50 Salads[11]	2 salads	Yes	None	None
Breakfast[12]	non-scripted	Yes	None	None
KUSK[13]	20 recipes	Yes	23	None
YouCook2[14]	89 recipes	Yes	33	None
EPIC-Kitchens[15]	non-scripted	Yes	163	None

Table 1: Available annotated cooking video datasets.

pendency trees [7], work-flow graphs [8], or action graphs [9]. Most work on this problem take ingredients into account by representing cooking actions as events that transform ingredients. We postulate that this problem can be simplified by including information about the location of food ingredients and the transformations of food.

3.2 Cooking datasets

Recent advances in object and action recognition have been facilitated by the increased availability of large-scale annotated datasets. In the cooking domain, this is more challenging due to the rich vocabulary of objects and activities, and their large inter and intra variability of food types and tools. Table 1 summarizes the size and content of several popular cooking video datasets, showing that while action annotations are widely available, food types are rarely annotated, and food states are never annotated. This observation motivates our decision to learn to localize foods and their states in a weakly supervised manner.

3.3 Locating food objects

Training deep neural network architecture on different image categories shows that bottleneck features can be used to describe images. Recently, some well-known neural networks pre-trained on large scale images datasets [16] have been shown to be useful for resolving problems like object detection and localization.

Convolutional Neural Networks (CNN) preserve the coarse grained spatial location of the network activations. Those network activations can be traced back in order to find a coarse estimation of the image region that triggered the network activation. Recent work has proposed to use network activations for modeling network attention to different images classes [17].

Adding a Global Average Pooling (GAP) on the last convolutional layer gives the network a limited ability to locate objects class. This has been proposed by Zhou et al. [3] as Class Activation Maps (CAM) that can be trained in a weakly-supervised manner using only image-level labels. CAMs have shown to perform well on the tasks of discriminative localization of classes [3] and visual questioning [18].

4 Learning food concepts

Describing food transformations requires combining recognition of food type and food state. We refer to these as "composite classes". For example, "diced tomato" or "sliced cucumber" are obtained by composing food type concepts (tomato, cucumber) with food state concepts (sliced, diced). Each composite class can be learned directly by using training examples from that class, or indirectly as a result of concept composition. The first method can be implemented using the previous work of Zhou et al. [3]. In this section, we explain the second method which uses food concept composition.

4.1 Concepts activation maps

We use Class Activation Maps (CAMs) [3] as an indicator of the image region occupied by a class member. When different classes share the same concept, their CAMs are combined to learn a concept activation map. Concepts activation maps are extracted from activation maps of the combined classes (CAMs). Here, we study 2 separate concepts: food type and food state. The goal of this layer is to decouple localization of food type and food state from the combined examples. In practice, we used a depth-wise convolutional layer on top of CAMs layer that separates spatial and depthwise information of CAMs. The number of filters of this layer is equal to the number of values a concept can take.

We study this problem as a multi-label classification problem. In training phase, we compute a separate cross-entropy loss function for each concept as well as for CAMs. Concept Activation Maps are the resulting network activation of this layer. The training architecture is explained in Figure 2.

4.2 Concept composition

We consider a composite class to be composed of more than one concept. A region in the image is considered to belong to a specific composite class if it belongs to all of its concepts (Figure 3).

Here, we explain two ways of composing concepts:

Product Concept Composition where each pixel in the corresponding concept maps is element-wisely multiplied. This is defined as

$$P_{cc}(x,y) = \prod_{c=1}^{n} A_c(x,y)$$

where $A_c(x,y)$ is the network activation value for the c^{th} concept activation map at the pixel (x,y), and n is the number

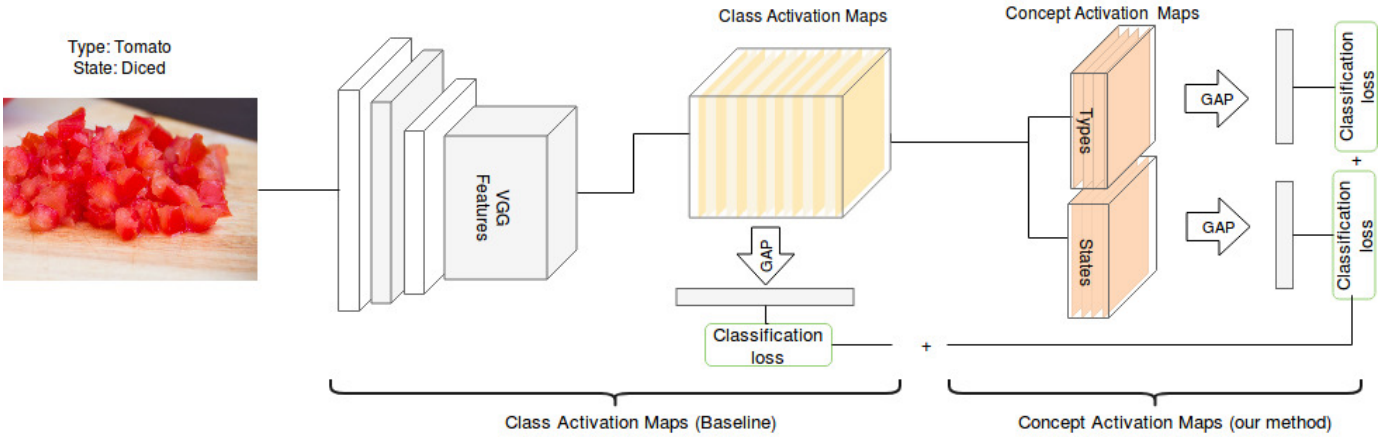


Figure 2: During training, we learn food concept maps for food types and food states from labeled examples.

of concepts.

Average Concept Composition where each composite concept map is the element-wise average prediction over the number of concepts. In practice, we define average composition of concepts as

$$P_{cc}(x,y) = \frac{\sum_{c=1}^{n} A_c(x,y)}{n}$$

The number of resulted composite concept maps $= C_1 \times C_2 \times \cdots \times C_n$ where C_n is the number of different classes of the n^{th} concept.

4.3 Food localization

We compute the location of a food type in a specific state from its corresponding composite map (P_{cc}). Figure 3 summarizes the process of food localization on test images. Firstly, input test images are transformed to multiple scales and passed to the network. Each of the output predicted concept maps is resized to the size of the original test image. Secondly, composite concept maps are computed for all different combination of concepts. Thirdly, composite concept maps are normalized and filtered as follows:

$$P_{cc}(x,y) = \begin{cases} P_{cc}(x,y), & \text{if } P_{cc}(x,y) >= Threshold \\ background, & \text{otherwise} \end{cases}$$

The threshold is set to 80% of overall activation maps. Therefore, the predicted label of pixel (x,y) is

$$P_l(x,y) = \underset{cc}{\operatorname{argmax}}(P_{cc}(x,y))$$

We compute the surface of connected components from $P_l(x,y)$ image for objects localization. We also compute the pixel-wise detection accuracy.

5 Experimental results

This section presents our experimental results for jointly learning food type and food state during cooking activities. Those results are compared to results from the work of Zhou et al. [3] as a baseline. We conduct the experiment on key-frames from 50 salads dataset.

5.1 Training and testing data

We collected a training set of images from Google Images using all possible composite concepts as query keywords. Those keywords are considered to be the image-level labels. We manually filtered irrelevant images to get a total of 468 images for training (in average 39 images per composite class).

For testing, we extracted key-frames from the 50 salads dataset [11]. This is a suitable dataset for our purpose because the same ingredients appear at different places and different states during the videos. The dataset has few number of ingredient types which facilitates the evaluation, and the recipes are scripted (the set of ingredients are fixed during image sequences).

We annotated by 251 key-frames for the following actions: *cut tomato*, *cut cucumber*, *cut cheese*, *peel cucumber*, *cut lettuce*. Key-frames have been chosen to be the mid frame of the *_post* part of annotated actions; we choose those actions as they are the moment where an ingredient was transformed into a new state. This state is expected to remain fixed until the next action. This annotation process results in average of 116 samples for testing per composite class. Each ingredient is segmented with a polygon using the *LabelMe* tool [1]. Ground-truth annotations will be made available [2].

5.2 Baseline

We used the CAM implementation [3] as the baseline for localizing foods for all the different composite classes. Activation maps of CAM are directly evaluated since composite classes are separately present in each activation map. We used similar network configurations for training the baseline and training our method.

5.3 Food localization results

In this section, we report the results of localizing food ingredients using the baseline method [3] and our proposed method.

We evaluated both methods using the Midpoint hit criteria as proposed by Rohrbach et al. [10]. The midpoint is computed as the center of gravity of the prediction values of composed concept map.

As in [10], a positive hit is considered if the midpoint falls inside the ground-truth mask, if the fired detection does not belong to the correct ground-truth label, it is counted as False Positive.

[1] https://github.com/wkentaro/labelme

[2] Please cite this work in case of using the annotation https://hal.archives-ouvertes.fr/hal-01815512/file/annotation_json.zip.

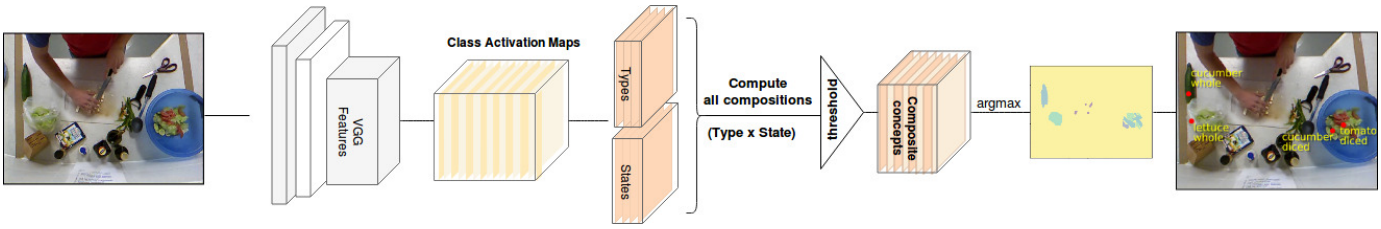


Figure 3: During testing, we compute the activation maps from unlabeled examples by composing the learned concept maps.

	Top 1			Top 3		
Composite class	Baseline	Product	Average	Baseline	Product	Average
Cheese_diced	18.75	**80.00**	77.78	29.41	80.00	**87.50**
Cheese_whole	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Cucumber_diced	**61.54**	33.33	42.11	**81.40**	35.71	44.44
Cucumber_peeled	0.00	**40.00**	**40.00**	0.00	**50.00**	**50.00**
Cucumber_sliced	71.43	93.55	**93.94**	77.14	94.74	**95.00**
Cucumber_whole	**67.35**	64.38	65.56	**74.58**	65.79	68.37
Lettuce_diced	58.54	**91.30**	86.21	78.03	**92.59**	88.89
Lettuce_whole	42.55	**65.22**	51.52	67.61	**70.37**	64.29
Tomato_diced	**80.34**	79.78	74.26	**86.45**	82.86	79.51
Tomato_sliced	80.00	80.65	**83.33**	88.10	86.11	**89.19**
Tomato_whole	5.56	**89.80**	87.76	5.71	**92.06**	90.32
Mean (mAP)	53.28	**74.36**	72.95	62.58	77.29	**77.96**

Table 2: Food Localization results on key-frames from 50 salads dataset.

Table 2 reports the Average Precision (AP) of each class. The results show a significant improvement on localization precision (74% mAP) whereas the baseline achieves (53% mAP) on classifying composite classes. In our experiments, the two concept composition methods (product and average) achieve similar performance and further work is needed to more accurately measure their benefits.

We also computed pixel-wise accuracy of the resulting activation maps, both for our method (63% without background, 94% with background) and the baseline (23% without background, 32% with background), again showing a significant improvement.

6 Conclusion

Locating foods with changing states is a challenging task where state-of-the-art methods obtain moderate results (53% mAP). In this paper, we have shown that jointly learning "type" and "state" concepts from training examples can significantly improve those result (74% mAP). More work is needed to confirm those early results and demonstrate the possibility of recognizing state-changing cooking actions in video by detecting food state changes collocated with hand motion and tool use.

References

[1] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3218–3226, 2015.

[2] J. Redmon and A. Farhadi. Yolo9000: Better, faster, stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6525, 2017.

[3] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Computer Vision and Pattern Recognition*, 2016.

[4] Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frederic Precioso. Recipe recognition with large multimodal food dataset. In *Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on*, pages 1–6. IEEE, 2015.

[5] Luis Herranz, Weiqing Min, and Shuqiang Jiang. Food recognition and recipe analysis: integrating visual content, context and external knowledge. *CoRR*, abs/1801.07239, 2018.

[6] Jingjing Chen and Chong-Wah Ngo. Deep-based ingredient recognition for cooking recipe retrieval. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 32–41. ACM, 2016.

[7] Jermsak Jermsurawong and Nizar Habash. Predicting the structure of cooking recipes. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 781–786, 2015.

[8] Yoko Yamakata, Shinji Imahori, Hirokuni Maeta, and Shinsuke Mori. A method for extracting major workflow composed of ingredients, tools, and actions from cooking procedural text. In *Multimedia & Expo Workshops (ICMEW), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[9] De-An Huang, Joseph J. Lim, Li Fei-Fei, and Juan Carlos Niebles. Unsupervised visual-linguistic reference resolution in instructional videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[10] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, pages 1–28, 2015.

[11] Sebastian Stein and J. McKenna McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), Zurich, Switzerland*. ACM, September 2013.

[12] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of Computer Vision and Pattern Recognition Conference (CVPR)*, 2014.

[13] Atsushi Hashimoto, Tetsuro Sasada, Yoko Yamakata, Shinsuke Mori, and Michihiko Minoh. Kusk dataset: Toward a direct understanding of recipe text and human cooking activity. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, UbiComp '14 Adjunct, pages 583–588, 2014.

[14] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. *arXiv preprint arXiv:1703.09788*, 2017.

[15] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. *arXiv preprint arXiv:1804.02748*, 2018.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[17] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

[18] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *International Conference on Computer Vision*, 2017.