IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 23, 2020, accepted February 9, 2020, date of publication February 13, 2020, date of current version February 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973704

State Recognition of Food Images Using Deep Features

GIANLUIGI CIOCCA, GIOVANNI MICALI, AND PAOLO NAPOLETANO
Department of Computer Science, Systems and Communications, University of Milano-Bicocca, 20126 Milan, Italy
Corresponding author: Gianluigi Ciocca (gianluigi.ciocca@unimib.it)

ABSTRACT State recognition of food images is a recent topic that is gaining a huge interest in the Computer Vision community. Recently, researchers presented a dataset of food images at different states where unfortunately no information regarding the food category was included. In practical food monitoring applications it is important to be able to recognize a peeled tomato instead of a generic peeled item. To this end, in this paper, we introduce a new dataset containing 20 different food categories taken from fruits and vegetables at 11 different states ranging from solid, sliced to creamy paste. We experiment with most common Convolutional Neural Network (CNN) architectures on three different recognition tasks: food categories, food states, and both food categories and states. Since lack of labeled data is a common situation in practical applications, here we exploits deep features extracted from CNNs combined with Support Vector Machines (SVMs) as an alternative to the End-to-End classification. We also compare deep features with several hand-crafted features. These experiments confirm that deep features outperform hand-crafted features on all the three classification tasks and whatever is the food category or food state considered. Finally, we test the generalization capability of the most performing deep features by using another, publicly available, dataset of food states. This last experiment shows that the features extracted from a CNN trained on our proposed dataset achieve performance quite close to the one achieved by the state of the art method. This confirms that our deep features are robust with respect to data never seen by the CNN.

INDEX TERMS Image state recognition, food state recognition, object state recognition, image understanding, image classification, CNN-based features.

I. INTRODUCTION

In the last few years, one of the most active topics in the Computer Vision community is the image understanding for object recognition [1], [2]. Within this context, automatic food analysis [3]–[7] is one application scenario that received great attention recently.

Accurate tracking of daily nutrition intake is not only conducive for people to maintain a healthy weight, but also important to treat and control food-related health problems like obesity and diabetes. Conventionally, this has been accomplished by exploring daily recorded manual logs. Nowadays, technology can support the users to keep track of their food consumption in a more user-friendly way allowing for a more comprehensive daily dietary monitoring. Computer vision techniques can help to build systems to automatically locate and recognize diverse foods as well as to estimate the food quantity. For example, one may simply take a picture of a plate of food using a smartphone and the whole process towards measuring the total calorie in the plate can be achieved by a visual understanding framework [8]–[11]. Cooking videos can be processed in order to extract food items, utensils and cooking procedures to construct an interactive, computer-aided system for learning how to cook healthy recipes [12]–[14].

Automatic food recognition is thus an important task for different applications. However, food recognition is a challenging task due to the intrinsic properties of the food items. For instance, food is a non-rigid object. It is characterized by intrinsic high intra-class variability where the same food can have a very diverse visual appearance in different images due to different preparations, placements in the plate, or acquisition point of view. This can be seen in Figure 1 that shows different images of ''Panette e crocchette''.

Moreover, if we consider a video recipe, during the preparation of the dish, a food item (e.g. a zucchini) assumes different shapes and appearances. For example, if we look at the video recipe of the ''Zucchini cream'' in Figure 2,

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano.

FIGURE 1. Visual differences of the same food: "Panette e crocchette".

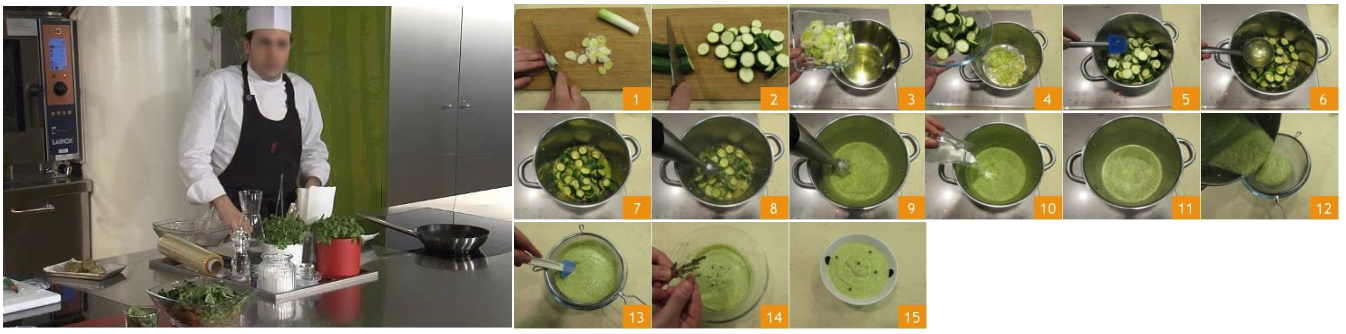

FIGURE 2. Video recipe example: "Zucchini cream". We can see the different food states induced by the cooking procedures.

we can see that the zucchini and the leek are initially whole and raw, then they are chopped in different styles (i.e. oblong and round for the leek and zucchini respectively), mixed, stir-fried, and finally whisked. During all these processing steps the appearance of the initial food significantly varies. The food is transformed in different *states* induced by the preparation steps themselves, and these states must be dealt with if we want to correctly identify the food throughout the whole video recipe.

Food state recognition is a topic that has not been extensively studied. The only previous works that tackled this problem are those by Jelodar *et al.* [15] that first introduced a new food state challenge dataset, and Salekin et al. [16].

According to [15], object states are characteristics into which an object can be transformed by some activity, and it can be described as a form of changes in form, color, or *texture*. As it can be seen in Figure 2, the texture of the zucchini is greatly changed as it undergoes the processing transformation required by the recipe. Its visual texture changes as it is being sliced and whisked as well as its color and shape are heavily affected by the cooking process. The object is always a zucchini, but the states are very different and each of them should be dealt with.

An ideal food recognition system needs to recognize food independently by its state, but also it needs to identify a food state within the recipe. This is very important for automatic video recipe transcript as well as fine-grained human activity understanding. The recognition of the different states of food is essential if we want also to determine the nutritional values of the food. While the food transition from one state to another, its nutrients may change due to seasoning, cooking or other cooking procedures. Being able to fully describe a food and its states will enable the implementation of intelligent dietary monitoring systems supporting users in controlling their food intake.

In this paper, inspired by the work of Jelodar *et al.* [15], we want to investigate the following issues:

- Can we recognize foods across different states?
- Can we recognize a food state independently by its identity?
- How robust are end-to-end Convolutional Neural Networks (CNNs)?
- How robust are CNN-based features with respect to hand-crafted features?

To answer these questions we created our own dataset of food states that differs from the one in [15] in that it allows us to perform three classification tasks: recognizing a food item across different states, identifying a food state across different foods and recognizing a food item at a given state. This is not possible with the existing dataset which allows only to perform the first task. Our dataset has been carefully curated and is composed of 11,943 images. It contains 20 categories of fruits and vegetables acquired in 11 states: batons, creamy paste, floured, grated, juiced, julienne, peeled, sliced, wedges/quarters, and whole. We used the dataset for the classification experiments evaluating different state-of-the-art hand-crafted features and learned features from recent

Convolutional Neural Networks architectures. We evaluated the generalization capability of our best CNN-based features on the food state recognition task using the dataset in [15].

The rest of the paper is organized as follows. In section II we present the most common used visual description for food classification. In section III we present the dataset, the hand-crafted features, and CNNs used in the experiments. Section IV presents results achieved with the end-to-end classification strategy, deep features, hand-crafted features and finally results achieved on the dataset presented in [15]. In section V we comment on the results achieved and present future works.

II. VISUAL DESCRIPTIONS FOR FOOD CLASSIFICATION

A huge variety of features have been proposed in the literature for describing the visual content. They are often divided into Hand-Crafted (HC) features and Learned Features (LF). Hand-crafted descriptors are features extracted using a manually predefined algorithm based on expert knowledge. Learned descriptors are features extracted using Convolutional Neural Networks (CNNs). In the following subsections, we will provide an overview of works related to food classification approaches exploiting hand-crafted features and learned features [17], [18].

A. HAND-CRAFTED FEATURES

Many works in the literature exploit hand-crafted visual features for food recognition and quantity estimation both for desktop and for mobile applications. Since using only a single feature is not enough to describe image contents, most of the approaches in the literature exploit several image descriptors at once in an early fusion or late fusion framework.

For example, [19] proposes a fusion of color and texture features for fruit recognition. Color features are extracted as statistical measures from the H and S color channels of the HSV color space, while texture features are derived from co-occurrence matrices. A voting-based, late decision fusion classifier is considered in [20]. Color statistics, entropy statistics, predominant color statistics, and energy responses of Gabor filter banks [21] as used for global descriptors, while local color, local entropy color, Tamura perceptual features [22], Gabor filters, SIFT descriptor [23], SURF [24], Steerable filters [25], and DAISY descriptor [26] as considered as local descriptors. A late fusion approach enabled to achieve a 7% improvement in recognition with respect to the single classifiers.

In [27], different features are integrated into a Multiple Kernel Learning (MKL) classification approach for single food recognition. The features comprise color histogram, Gabor texture features, Histogram of Oriented Gradient (HOG) [28], SIFT bag-of-features [29]. MKL is also used in [3] for multiple food recognition. The images are processed with a candidate region detection aimed at locating food regions. Each region is described in terms of SIFT and CSIFT bag-of-features ([30]), HOG, Gabor texture, and color histograms [31]. [32] uses a k-NN classifier on local and global features. In [33] a vocabulary is constructed on textons and the food images are classified using SVM. The same classifier is used in [34] where local binary pattern and relationship between SIFT interest points are used to code the local and spatial information. SVM, Artificial Neural Networks and Random Forest classification methods are used in [35] where 14 different color and texture descriptors are evaluated. The one that provided the best result was the HSV-SIFT descriptor that describes local textures in the color channels.

Ten different features are considered in [36]: color histograms in different color spaces, shape with Pyramid of HOG, and GIST [37], texture with local Binary Patterns [38], Local Phase Quantization [39], Local Configuration Pattern [40], Binary Gabor Pattern [41], and MSR4-Gabor filter bank [42]), and data-driven features (CNN features). All these features are fed to a committee of classifiers built on Extreme Learning Machines whose outputs are combined in the final result.

Food recognition can leverage from the contextual information derived from the place where the food is consumed i.e. the restaurant. In [43] the food images are first geo-localized then several features are extracted and fed to an MKL classifier for recognition. The image descriptors are based on colors such as Color Moments Invariants, and Hue Histograms, and on variants of the SIFT descriptor. Local and global features are tested in [44]. The list of features used is: CEDD [45], Gabor Features, Opponent Gabor Features, LBP, Local Color Contrast, Chromaticity Moments, and Complex Wavelet Transform [46]. Among these features, CEDD achieved the best recognition results.

The arrangement of food ingredients is also a possible cue for food recognition. Given soft labeling of food pixels, in [47] spatial relationships between pixels of different food ingredients are described using pairwise local features. Results showed that, on the evaluated dataset, the approach outperforms other bag-of-features models.

Notwithstanding the large literature in hand-crafted features, these descriptors need to be carefully chosen for the task at hand, or a suitable feature selection procedure must be applied to limit the information redundancy and the *curse of dimensionality*.

B. LEARNED FEATURES

CNNs are a class of learnable architectures adopted in many domains such as image recognition, image annotation, image retrieval, etc... [48]. CNNs are usually composed of several layers, each involving linear as well as nonlinear operators. The layers' parameters are learned jointly in an end-to-end manner to solve a particular task. A CNN that has been trained for solving a given task can be also adapted to solve a different task. It is common to use a CNN that is pre-trained on a very large dataset and adapt it for new tasks [49].

Several studies have investigated deep neural networks for food recognition as end-to-end classifiers, or feature extractors. One of the first works that used features extracted from

CNNs within the context of food recognition was by [50]. The food images are described with the features extracted from the FC7 layer of an AlexNet-style architecture pre-trained on ImageNet and classified with SVM. Reference [51] evaluated different CNN-based techniques for food recognition. These techniques include network pre-trained with the large-scale ImageNet data, fine-tuned network for food classification, and the use of the activation features extracted from the CNN. In [5] the AlexNet network is used as a feature extraction module for the classification of food images acquired in a canteen environment. Experiments with traditional features using k-NN and SVM classifiers showed the superiority of the CNN-based features. Reference [52] used Google's image recognition architecture Inception V3. The network, composed of 54 layers, was designed to tackle the ImageNet's ILSVRC15 and it was fine-tuned for classifying food images. The network greatly surpasses the performances of previous approaches. Another approach based on the Inception architecture is DeepFood [53]. In this case, 1×1 convolutional layers are introduced to reduce the network complexity with some loss in performances. [54] devised the WIde-Slice Residual Network (WISeR) designed to specifically handle structures that can be found in food images. The network outperformed the Inception V3 architecture. CNNs can be also used to tackle different tasks simultaneously. Reference [55] used this ability to build a deep convolutional neural network architecture for simultaneous food ingredients recognition and food categorization. Reference [56] proposed NutriNet, a modified version of the AlexNet architecture which uses fewer parameters compared with the original design. The network was trained on a very large food database of more than 130,000 images.

The Residual Network ResNet-50 [57] is one of the most powerful and performing CNN architecture. The network is exploited in [58] for extracting features to be used for image retrieval in a dataset of 1,200 distinct dishes. The CNN-based features greatly outperform traditional bag-of-SIFT and textons features [33]. In [59] an extensive evaluation of different techniques for food recognition and retrieval is conducted on a dataset of more than 240,000 images of 475 different food dishes (Food-475). Seven different CNN architectures for end-to-end classification are evaluated: AlexNet, Caffe-Reference, GoogleNet, VGGNet-16, VGGNet-19, InceptionV3, and ResNet-50. Among these architectures, the ResNet-50 showed the best recognition accuracy. In the same work, CNN-based features are also evaluated. The features are extracted from a ResNet-50 trained with different food datasets and recognition is performed using a k-NN classifier. The same features are also tested in a retrieval task. Experiments showed that robust features can be obtained from very large and heterogeneous food datasets.

Recognizing a food identity during a dish preparation is quite challenging. Reference [15] was the first work to explicitly introduce and address the food state classification problem. By analyzing cooking procedures, eleven states of the most frequent foods are identified and a new dataset of food states is introduced. They proposed a ResNet based deep model solution to the state identification problem. Since state identification has a strong correlation with the type of food, individual models are fine-tuned for each food in the dataset. This strategy showed significant improvement with respect to a food-independent model. The Inception V3 architecture is used instead in [16]

CNNs are exploited also for other food-related tasks such as food localization, segmentation, ingredients recognition, quantity, and calories estimation. Readers interested in these tasks can refer to [60] and [61] for a comprehensive survey of recent techniques.

III. MATERIALS AND METHODS

The aim of this paper is twofold: the collection of a new dataset containing foods in different states, and the evaluation of features and classification methods. Specifically, we are interested in food recognition across different states, food state classification across different foods, and joint food and state recognition. In this Section, we first illustrate the procedure we adopted for the collection of the dataset and then we illustrate the classification pipeline as well as the procedure we adopted for the evaluation of the classification methods.

A. DATASET

The construction of our dataset is inspired by [15]. We identified 11 food states representative of the states that can be found in food recipes. The states are: batons, creamy paste, floured, grated, juiced, julienne, peeled, sliced, wedges/quarters, and whole. Most of these states apply to fruits and vegetables so we focus our attention on these food classes. Among the possible foods, we selected those with at least two states. The final list of fruits and vegetables is: apple, apricot, aubergine, banana, beet, carrot, garlic, lemon, melon, onion, orange, peach, pear, pepper, potato, pumpkin, strawberry, tomato, watermelon, and zucchini.

We searched and downloaded the images using the Google search engine using a Python script. Textual queries with combinations of food and state words (such as ''apple'' and ''diced'') were submitted in several languages (i.e. English, Chinese, French, German and Italian). The downloaded images were manually reviewed to ensure that they were pertinent to our food/state classification. We discarded images depicting food in cans since most of the time the food is covered by a large label. We also discarded images containing different foods or different states. Furthermore, we edited images having a very large background area in comparison to the food area to limit the influence of non-relevant regions during the classification. At the end of the analysis process, each image is filed in a double layer categorization: food identity, and state. In this way, we can perform food classification across states, state classification across different foods, or a paired food and state classification. Starting from an initial set of 180,000 downloaded images, we obtained 11,943 images manually inspected and categorized.

TABLE 1. Dataset classes with respect to the State categorization.

States	N. of images	N. of foods
Batons	479	8
Creamy paste	984	13
Diced/Chopped	1492	13
Floured	79	4
Grated	596	7
Juiced	903	14
Julienne	451	6
Peeled	1310	15
Sliced	1521	13
Wedges/quarters	931	14
Whole	3197	20

TABLE 2. Dataset classes with respect to the Food categorization.

Food	N. of images	N. of states
Apple	772	9
Apricot	310	3
Aubergine	597	8
Banana	428	5
Beet	824	10
Carrot	1080	9
Garlic	311	3
Lemon	332	4
Melon	354	4
Onion	718	8
Orange	474	5
Peach	472	5
Pear	258	4
Pepper	543	4
Potato	1039	10
Pumpkin	546	8
Strawberry	512	6
Tomato	940	7
Watermelon	486	5
Zucchini	947	10

Table 1 shows the organization of the dataset according to the state categorization. We can see that the state ''floured'' is the one containing the fewer images (i.e. 79), while the ''whole'' state is the class containing the most images. We also report the number of foods that are present in each state. It can be seen that not all states contain all foods. Table 2 details the content of the dataset according to the food categorization. In this case, the number of images in each class has less variability than in the previous categorization. The number of images ranges from 300 to about 1,000. None of the 20 foods has all the 11 states. Beet, potato, and zucchini have 10 states, while garlic has only three states.

Figure 3 shows some examples of images in our dataset. These images are representative of the food/state categorization. We can notice that the food states have a very large visual diversity. Also within a state, the visual appearance of the foods is influenced by the distance of acquisitions, lights, colors, and textures.

The images in our original dataset have been split into three sets by allocating 70% (8,233 images) for the training, 15% (1,855 images) for validation, and the last 15% (1,855 images) for testing.

B. METHODS

We evaluated several hand-crafted and deep learning based feature extraction methods. The evaluation pipeline includes a feature extraction module and a classification module (one for each task) based on an SVM classifier with a radial basis function (RBF). The validation set of the dataset is used for the choice of the RBF parameters. Learned features are extracted from several CNNs trained using our dataset. For the sake of comparison, we also evaluate the trained CNN architectures for end-to-end classification. In the following subsections we describe the chosen hand-crafted features, and the CNN models.

1) HAND-CRAFTED FEATURES

We considered both color and grey-scale hand-crafted features. The grey-scale image L is defined as follows: $L = 0.299R + 0.587G + 0.114B$, where R = Red, G = Green and B = Blue. All feature vectors have been l^2 normalized (they have been divided by its l^2-norm):

- 256-dimensional grey-scale histogram (Hist L) [31];
- 768-dimensional RGB and normalized RGB space (rgb) marginal histograms (Hist RGB and Hist rgb) [38];
- 144-dimensional color and edge directivity descriptor (CEDD) features. This descriptor uses a fuzzy version of the five digital filters proposed by the MPEG-7 Edge Histogram Descriptor (EHD), forming 6 texture areas. CEDD uses 2 fuzzy systems that map the colors of the image in a 24-color custom palette [45];
- 8-dimensional Dual Tree Complex Wavelet Transform (DT-CWT) features obtained considering four scales, mean and standard deviation, and three color channels [46], [62];
- 580-dimensional Histogram of Oriented Gradients feature vector [28]. Nine histograms with nine bins are concatenated to achieve the final feature vector (HoG);
- 243-dimensional Local Binary Patterns (LBP) feature vector for each color channel. We selected the LBP with a circular neighborhood of radius 2 and 16 elements, and 18 uniform and both no-rotation invariant patterns (LBP) and rotation invariant (LBP-RI) [38];
- 729-dimensional LBP RGB combined with the LCC descriptor, as described in [63]–[65]
- 512-dimensional Gist features obtained considering eight orientations and four scales for each channel (Gist RGB) [37];

2) DEEP LEARNING-BASED FEATURES

For our experiments, we have decided to test four different CNNs [66] (see table 3): GoogLeNet [67] which

FIGURE 3. Example of foods and their relative states in our dataset.

won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) contest in 2014, Inception-v3 [68] which overcame the GoogLeNet in 2016, MobileNet-v2 [69] which is the most performing mobile network architecture,

TABLE 3. Overview of the CNN architecture considered.

CNN	Depth	Size (MB)	N. of parameters (million)	Size of input image	Feature vector size
GoogLeNet	22	27	7.0	224-by-224	1024
Inception-v3	48	89	23.9	299-by-299	2048
MobileNet-v2	53	13	3.5	224-by-224	1280
ResNet-50	50	96	25.6	224-by-224	2048

and ResNet-50 [57] which is part of the ensemble of CNNs that won the contest in 2015. The models were pre-trained on ImageNet so we fine-tuned them for food state recognition by modifying the last layers to match our classification tasks. Fine-tuning has been performed using the SGDM optimizer (Stochastic Gradient Descent with Momentum), a mini-batch of size 10, and a learning rate of 0.0003 for 12 epochs.

The CNN architectures are used as end-to-end classifiers as well as feature extractors. We extract the features from the fully connected layers before the actual classification layer in each model. Specifically, the extracted features have the dimension of 1024 values for the GoogLeNet, 2048 for both the Inception-v3 and ResNet, and of 1280 values for the MobileNet-v2. The features are used to train an SVM classifier (with RBF kernel) in the same way as we have done for hand-crafted features.

IV. RESULTS

We performed different classification experiments using the hand-crafted features and CNN-based features. More in details, we investigated the performance of the features in recognizing a food state regardless of the food identity, the recognition of the food across the different states, and jointly recognizing the food and state. All the experiments have been performed considering the three splits of our dataset as described in Section III-A and averaging the obtained results. We also evaluated the robustness of the best performing CNN-based features for the recognition of states of the dataset in [15]. For brevity, in the following tables, we have indicated the names of networks as G.Net, Inc-v3, M.Net and R.Net for the GoogLeNet, Inception-v3, MobileNet-v2, and ResNet50 respectively.

Detailed results are reported in terms of per-class Accuracy while the overall results are reported in terms of Average Accuracy and F_1-Score.

A. END-TO-END CNN CLASSIFICATION

In the first experiment, we evaluated the performance of the fine-tuned CNN models in an end-to-end classification scenario. Results for the food classification task are shown in Table 4, while the results for the state classification task are shown in Table 5. As can be seen, the four networks can achieve very good classification results on most of the classes. For the food classification task, the recognition of the "pumpkin" seems to be the most difficult with the best results obtained by the GoogLeNet with 80%. Also the "pear" exhibits a general recognition accuracy lower than the other classes. With respect to the state classification task, the "batons", and "flower" have the lowest accuracy.

If we examine the average accuracy of each network on all the classes in Table 6, we can see that all the models are able to achieve a food classification accuracy above 87%. The Inception-v3 and the ResNet-50 are the best models with an average accuracy of about 92%. It is worth noting that the MobileNet-v2 has only 2% drop in accuracy and

TABLE 4. Results of the fine-tuned networks for the food classification task.

Food	G.Net	Inc-v3	M.Net	R.Net
apple	84.44	86.67	80.00	83.89
apricot	78.91	82.31	85.03	89.12
aubergine	95.44	94.39	94.04	96.49
banana	90.55	93.53	91.04	92.04
beet	94.23	95.80	94.23	98.16
carrot	90.51	91.31	90.71	94.55
garlic	91.67	92.36	92.36	93.06
lemon	88.89	94.12	87.58	97.39
melon	87.88	91.52	93.94	96.97
onion	93.69	96.40	90.69	92.79
orange	88.74	93.24	94.59	87.84
peach	81.53	90.99	89.19	91.44
pear	69.92	88.62	85.37	86.99
pepper	91.16	94.78	91.57	96.39
potato	84.06	93.37	91.51	93.17
pumpkin	80.00	76.47	71.76	71.37
strawberry	90.00	94.58	93.75	92.92
tomato	76.09	97.93	94.25	95.17
watermelon	96.49	94.74	97.37	95.18
zucchini	94.82	96.85	96.85	95.72

TABLE 5. Results of the fine-tuned networks for the state classification task.

State	G.Net	Inc-v3	M.Net	R.Net
batons	89.91	90.79	74.12	86.40
creamy paste	95.70	96.34	94.19	96.13
diced chopped	97.68	98.70	97.10	99.42
floured	92.86	85.71	95.24	83.33
grated	96.70	96.70	95.60	94.87
juiced	95.17	99.77	99.31	98.85
julienne	95.24	94.76	92.86	90.48
peeled	90.26	96.04	90.26	93.89
sliced	92.77	98.01	94.33	96.74
wedges quarters	94.25	93.10	89.43	93.79
whole	98.85	99.32	99.19	98.98

TABLE 6. Overall classification results (Average Accuracy) of the fine-tuned networks on all the different tasks. The best results are in bold.

Network	Food	State	Food and State
GoogLeNet	87.45	94.49	84.65
Inception-v3	**92.00**	**95.39**	**90.48**
MobileNet-v2	90.29	92.88	86.76
ResNet-50	**92.03**	93.90	89.72

TABLE 7. Overall classification results (F_1-Score) of the fine-tuned networks on all the different tasks. The best results are in bold.

Network	Food	State	Food and State
GoogLeNet	87.55	93.79	85.19
Inception-v3	**92.26**	**95.86**	**90.85**
MobileNet-v2	90.48	92.87	87.14
ResNet-50	**92.13**	94.37	90.14

the network is by far the most lightweight of the four. With respect to the fine-tuned network for state recognition, all the networks are able to achieve an accuracy above 92%. In this scenario, the best model is the Inception-v3 with 95.39%, followed by GoogLeNet (94.49%), ResNet-50 (92.90%), and MobileNet-v2 (92.88%). Again it is worth noting the very good performances of the MobileNet-v2. We also evaluate the joint classification of food and state. Table 6 shows that the Inception-v3 network achieves the best result with an accuracy of 90.48%. All the networks are able to recognize both information with an accuracy of at least 85%.

Table 7 shows the overall results on the three tasks computed in terms of F_1-Score. The values do not exhibit a different behaviour than those in Table 6.

B. CNN-BASED FEATURE CLASSIFICATION

The previous experiments proved that the trained networks can effectively classify food and states with high accuracy. However, we were more interested in the features that have been learned by the networks. For this reason, we used the networks to extract the features embedded in the last layers of the networks. Specifically, we extracted the features from the following layers: "pool5-drop_7x7_s1", "avg_pool", "global_average_pooling2d_1", and "avg_pool" for the GoogLeNet, Inception-v3, MobileNet-v2, and ResNet-50 respectively. Classification has been performed by training an SVM classifier with a RBF kernel and using the same splits as before.

The detailed results for the food classification task are reported in Table 8, while the results for the state classification task are reported in Table 9. For the end-to-end classification results, we cannot see specific increments or decrements in the accuracy for all the classes. In some cases the CNN-based features exhibit worse performances than the end-to-end counterparts (e.g. for the G.Net the "apple" drops from 84.44% to 80.00%), while in other cases the accuracy increases (e.g. for the M.Net, the accuracy increases from 80.00% to 85.28%). If we consider the average results on all the classes reported in Table 10, we can see that the use of the embedded features coupled with the SVM classifier does not exhibits significant drops in classification accuracy. The drop is in the order of 1 percentage point on average.

TABLE 8. Results of the CNN-based features for the food classification task.

Food	G.Net	Inc-v3	M.Net	R.Net
apple	80.00	88.61	85.28	84.72
apricot	78.23	82.31	85.71	82.99
aubergine	89.82	94.39	95.09	97.89
banana	91.54	95.52	93.03	92.54
beet	93.44	97.38	96.06	96.85
carrot	90.71	91.72	91.11	93.13
garlic	86.11	88.89	88.89	88.89
lemon	90.85	94.77	92.81	94.12
melon	84.24	90.30	89.09	93.33
onion	92.49	96.40	95.20	93.09
orange	88.74	92.79	92.79	93.69
peach	82.43	85.59	87.84	89.19
pear	71.54	85.37	85.37	86.99
pepper	90.76	97.19	93.98	94.38
potato	89.03	93.79	94.00	94.20
pumpkin	76.08	74.90	76.08	76.47
strawberry	86.25	94.58	92.92	92.92
tomato	94.25	96.78	96.09	96.78
watermelon	93.42	96.05	97.81	96.49
zucchini	95.72	96.85	97.75	96.62

TABLE 9. Results of the CNN-based features for the state classification task.

State	G.Net	Inc-v3	M.Net	R.Net
batons	86.40	89.47	82.89	86.40
creamy paste	95.05	97.20	95.05	96.77
diced chopped	98.84	99.71	99.42	99.57
floured	80.95	80.95	76.19	83.33
grated	95.24	97.80	95.97	95.60
juiced	98.39	99.54	99.31	99.31
julienne	93.33	94.76	90.95	90.48
peeled	94.22	95.05	94.06	94.72
sliced	94.61	97.59	94.33	96.88
wedges quarters	91.26	93.10	92.64	94.02
whole	98.85	99.39	98.92	99.59

TABLE 10. Overall classification results (Average Accuracy) of the CNN-based features on the different tasks. The best results are in bold.

Network	Food	State	Food and State
GoogLeNet	87.28	93.38	85.64
Inception-v3	**91.71**	**94.96**	**90.53**
MobileNet-v2	91.34	92.70	88.88
ResNet-50	**91.76**	94.24	89.96

TABLE 11. Overall classification results (F_1-Score) of the CNN-based features on the different tasks. The best results are in bold.

Network	Food	State	Food and State
GoogLeNet	86.38	93.48	86.06
Inception-v3	**90.40**	**94.76**	**90.89**
MobileNet-v2	90.29	92.61	89.25
ResNet-50	**90.90**	94.26	90.28

TABLE 12. Classification results of the hand-crafted features on the food classification task across states.

Food	HIST-L	CEDD	HOG	GABOR	LBP	LBP-RI	LBP-LCC	GIST	DT-CWT	HIST-RGB
apple	13.61	9.72	20.56	23.89	25.28	31.39	30.00	30.56	24.72	23.06
apricot	15.65	8.16	13.61	14.97	31.97	34.69	23.13	25.85	18.37	22.45
aubergine	27.72	35.09	23.51	30.53	39.30	38.25	37.89	36.84	45.26	31.93
banana	12.94	25.37	25.87	20.40	16.42	20.40	25.37	31.84	12.44	26.87
beet	43.04	68.50	45.93	60.89	61.94	65.09	34.91	74.80	45.93	74.02
carrot	39.19	59.80	43.64	50.30	36.97	52.32	43.23	63.64	28.48	68.08
garlic	7.64	18.06	15.97	15.97	15.97	11.81	22.22	13.19	18.06	27.08
lemon	16.99	26.80	28.10	27.45	18.95	26.14	23.53	45.10	16.99	22.22
melon	13.33	10.30	15.15	13.33	16.97	24.24	27.27	27.27	9.70	16.97
onion	22.52	32.73	46.85	46.85	45.65	45.65	52.55	53.15	27.33	46.25
orange	18.47	25.23	36.04	31.98	33.33	47.75	41.44	52.25	28.83	32.43
peach	8.11	11.26	13.06	11.26	20.27	13.51	14.86	19.82	13.96	16.22
pear	7.32	6.50	14.63	8.94	8.94	15.45	17.07	23.58	10.57	9.76
pepper	18.47	25.70	15.26	18.07	20.08	38.15	32.93	26.91	15.26	38.55
potato	23.19	43.69	38.51	27.33	44.93	50.52	35.82	48.86	39.34	53.42
pumpkin	7.06	14.12	14.12	14.51	16.86	27.45	14.90	21.96	11.76	14.12
strawberry	12.50	14.58	42.92	34.58	39.58	40.83	31.25	40.42	30.42	26.25
tomato	36.09	51.49	41.15	39.54	37.47	48.28	33.10	65.29	31.26	67.36
watermelon	6.14	19.30	21.93	22.37	26.75	38.16	26.32	42.11	21.05	15.35
zucchini	18.24	74.55	46.17	57.21	59.46	56.76	38.06	81.31	55.41	81.08

TABLE 13. Classification results of the hand-crafted features on the state classification task across foods.

State	HIST-L	CEDD	HOG	GABOR	LBP	LBP-RI	LBP-LCC	GIST	DT-CWT	HIST-RGB
batons	12.72	10.09	6.14	22.81	31.58	23.68	29.39	17.54	20.61	10.96
creamy paste	26.45	19.14	53.76	24.73	44.30	52.90	43.66	73.33	34.19	26.45
diced chopped	17.68	19.71	43.91	36.81	52.32	52.75	51.88	72.61	44.64	23.48
floured	2.38	0.00	7.14	2.38	9.52	7.14	14.29	2.38	2.38	30.95
grated	9.52	6.96	24.18	21.25	35.90	46.52	40.66	57.14	32.23	15.75
juiced	17.93	31.95	73.56	56.78	49.20	73.33	67.36	80.92	19.77	20.46
julienne	32.86	23.81	12.38	42.38	53.81	48.10	40.95	42.86	41.90	40.00
peeled	18.81	20.63	27.72	25.25	24.59	25.08	24.75	28.22	20.63	23.60
sliced	23.55	26.10	35.89	28.23	29.79	45.25	37.73	40.99	25.82	30.50
wedges quarters	12.41	11.03	29.20	13.56	14.02	20.46	12.87	32.41	10.80	11.72
whole	75.61	77.98	79.67	73.71	80.08	86.31	83.13	86.31	74.73	79.20

TABLE 14. Comparison, in terms of Average Accuracy, between learned features and hand-crafted features on the Food, State, and Food-and-State classification tasks. The best results are in bold. The best results for the hand-crafted features are underlined.

Task	G.Net	Inc-v3	M.Net	R.Net	HIST-L	CEDD	HOG	GABOR	LBP	LBP-RI	LBP-LCC	GIST	DT-CWT	HIST-RGB
Food	87.28	**91.71**	91.35	91.76	18.41	29.05	28.15	28.52	30.85	36.34	30.29	41.24	25.26	35.67
State	93.38	**94.96**	92.70	94.24	22.72	22.49	35.78	31.63	38.65	43.77	40.61	48.61	29.79	28.46
Food-and-State	85.64	**90.53**	88.88	89.96	9.63	14.43	18.54	16.62	19.21	27.26	22.84	32.26	14.02	19.77

TABLE 15. Comparison, in terms of F_1-Score, between learned features and hand-crafted features on the Food, State, and Food-and-State classification tasks. The best results are in bold. The best results for the hand-crafted features are underlined.

Task	G.Net	Inc-v3	M.Net	R.Net	HIST-L	CEDD	HOG	GABOR	LBP	LBP-RI	LBP-LCC	GIST	DT-CWT	HIST-RGB
Food	86.38	**90.40**	90.29	90.90	18.27	27.71	28.67	28.06	29.22	37.44	32.61	42.96	24.04	37.41
State	93.48	**94.76**	92.61	94.26	23.45	24.79	36.29	34.40	38.86	47.38	47.15	52.11	32.55	29.57
Food-and-State	86.06	**90.89**	89.25	90.28	9.96	14.82	19.14	17.06	19.74	27.94	23.55	33.04	14.49	20.18

This means that the features are indeed robust enough to solve both classification problems. The use of a non-linear classifier allows the models to achieve, in some cases, even slightly better results than the end-to-end counterpart. This can be seen in the case of the joint food and state classification task. In this case, all the features have better results than the end-to-end counterparts with the MobileNet-v2 exhibiting an increase in accuracy of 2.2 percentage points.

Table 11 shows the overall results in terms of F_1-Score. As before, there are no significant differences with respect to the results in Table 10.

C. COMPARISON WITH HAND-CRAFTED FEATURES

Table 12 and Table 13, show the per-class accuracy of the ten hand-crafted features described in Section III-B.1. The hand-crafted features are not able to capture enough

TABLE 16. Comparison, in terms of Average Accuracy, between the best deep-based feature (Inc-v3) and its concatenation with each hand-crafted feature on the Food, State, and Food-and-State classification tasks. The best results are in bold.

Task	Inc-v3	Inc-v3 + HIST-L	Inc-v3 + CEDD	Inc-v3+ HOG	Inc-v3 + GABOR	Inc-v3 + LBP	Inc-v3 + LBP-RI	Inc-v3 + LBP-LCC	Inc-v3 + GIST	Inc-v3 + DT-CWT	Inc-v3 + HIST-RGB
Food	91.71	91.70	91.84	91.79	91.79	91.77	**91.89**	91.72	91.51	91.77	91.41
State	94.96	94.88	94.91	95.03	94.97	94.99	**95.29**	94.94	94.62	94.96	94.61
Food-and-State	90.53	90.55	90.65	90.60	90.58	90.62	**90.71**	90.48	90.23	90.57	90.14

TABLE 17. Comparison, in terms of F_1-Score, between the best deep-based feature (Inc-v3) and its concatenation with each hand-crafted feature on the Food, State, and Food-and-State classification tasks. The best results are in bold.

Task	Inc-v3	Inc-v3 + HIST-L	Inc-v3 + CEDD	Inc-v3+ HOG	Inc-v3 + GABOR	Inc-v3 + LBP	Inc-v3 + LBP-RI	Inc-v3 + LBP-LCC	Inc-v3 + GIST	Inc-v3 + DT-CWT	Inc-v3 + HIST-RGB
Food	90.40	90.38	90.64	90.52	90.48	90.45	**90.58**	90.46	90.32	90.40	90.16
State	94.76	94.81	94.85	94.89	94.91	94.95	**95.29**	94.92	94.67	94.87	94.54
Food-and-State	90.89	90.90	91.01	90.94	90.94	90.97	**91.06**	90.83	90.56	90.92	90.47

TABLE 18. Classification results of the learned features on the state classification task of the Jelodar dataset.

State	G.Net	Inc-v3	M.Net	R.Net
creamy paste	84.04	81.91	79.79	80.85
diced chopped	82.91	86.32	84.62	82.91
floured	88.71	88.71	87.90	92.74
grated	90.98	87.70	89.34	89.34
juiced	91.73	87.97	91.73	95.49
julienne	80.15	81.68	77.10	77.10
mixed	84.76	87.62	83.81	93.33
peeled	92.05	93.18	90.91	92.05
sliced	77.04	83.16	79.08	85.20
other	34.38	32.03	19.53	33.59
whole	79.43	83.43	78.29	81.14

TABLE 19. Learned features extracted from CNNs trained on the proposed dataset and used, coupled with SVM for classify states of the Jelodar dataset.

Method	Accuracy(%)
G.Net + SVM(RBF)	80.56
Inc-v3 + SVM(RBF)	81.25
M.Net + SVM(RBF)	78.37
R.Net + SVM(RBF)	82.16
Inc-v3+LBP-RI + SVM(RBF)	81.15
Jelodar [15] - Single network	80.40*
Jelodar [15] - Multiple networks	86.90*

* Result in [15] are computed on a different version of the dataset than the one used for the evaluation of the four CNN-based features in the table.

information about the image contents to discriminate between the different classes. Concerning the food classification task, we can see that some foods are more easily recognizable with the hand-crafted features than others. For example, the beet, carrot, tomato and zucchini are the classes that have higher recognition accuracy. This could be due to the characteristic color and shape of the food. On the other hand, the peach and pear are the fruits that are more difficult to recognize by the hand crafted features. If we look at the CNN-based features, the most difficult food to recognize is the pumpkin followed by the pear.

Concerning the state classification task, the best results are obtained for the ''whole'' state for both the CNN-based and hand-crafted features. This is not surprising since it corresponds to a traditional image recognition task. The most difficult state to be recognized is the ''floured'' one. This could be related to the fact that this state is not atomic but must be considered in conjunction with another state (e.g. sliced floured zucchini). Surprisingly, the HIST-RGB feature achieves an accuracy of about 31%, while other hand-crafted features do not reach 15% and some are completely unable to recognize this state.

Table 14 compares the results of the CNN-based features against the hand-crafted ones in terms of average accuracy. As expected, the CNN-based features achieve the best results among all the features. The best CNN-based features are those extracted from the Inc-v3 network. Among the hand-crafted features, the best overall result is obtained again by the GIST features (i.e 41.24%). This could be due to the fact that this descriptor summarizes texture information at different scales and orientations. Again, similar conclusions can be derived from the performances computed in terms of F_1-Score as shown in Table 15.

D. COMBINATION OF DEEP-BASED AND HAND-CRAFTED FEATURES

We concatenated the best performing features extracted from the Inception-v3 network (Inc-v3) with each of the hand-crafted feature. The aim was to investigate if the combination of different features can further improve the overall classification performance. Table 16 and Table 17 report the overall performance in terms of average accuracy and F_1-Score respectively.

Concerning the food classification task, the accuracy is above 91% and the F_1-Score is above 90% for all the combinations. The differences with respect to Inc-v3 are very small. The best combination achieves extra 0.18 points for the average accuracy, and 0.24 points for the F_1-Score. Concerning the state classification task, the results are more diverse. The best result is achieved by Inc-v3+LBP-RI with 95.29% of average accuracy against 94.96% of the Inc-v3 alone. This is also true for the F_1-Score (95.29% against 94.76%).

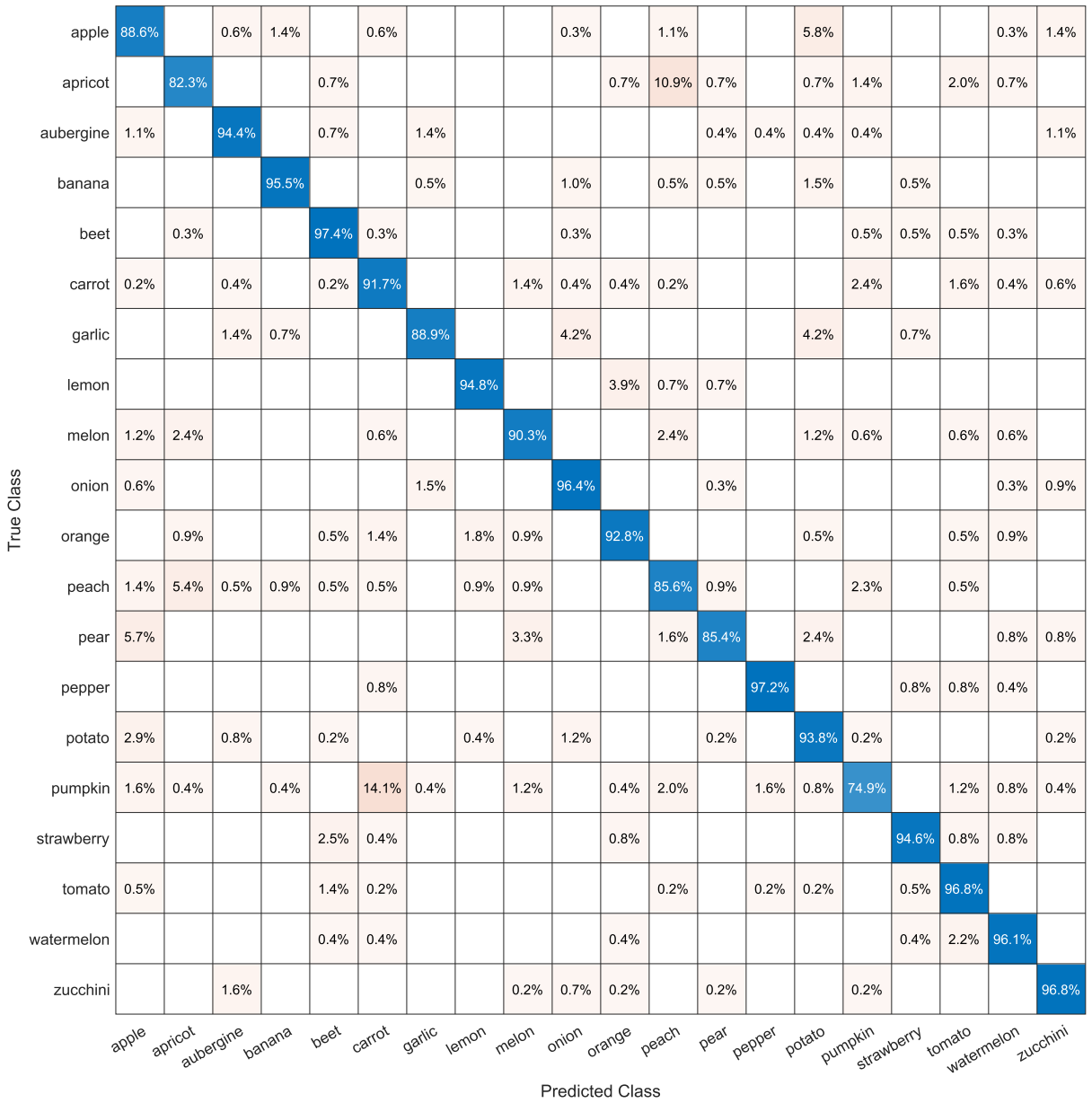

FIGURE 4. Confusion matrix of the Food classification task with the CNN-based features extracted from the best performing network (Inception-v3).

However, the overall gain is less than 1 percentage point. In the case of the food-and-state classification task, we notice very small differences. The Inc-v3+LBP-RI is again the best combination with 90.71% against 90.53%, and 90.71% against 90.89% for the average accuracy and F_1-Score respectively. These results show that there is no significant advantage in combining CNN-based and hand-crafted features for our problem.

E. COMPARISON WITH THE STATE-OF-THE-ART

We also tested how the CNN-based features can recognize the food states on a dataset in the state-of-the-art. Specifically, we evaluated the features extracted by the four networks on the Jelodar dataset [15]. This dataset is the only public dataset that is comparable to ours. The dataset contains some states common to our dataset but also new, unseen, states. For example, the ''mixed'' state is present in the dataset, and this state corresponds to different finely chopped ingredients that are blended. The ''other'' state is an heterogeneous class and comprises all the states that are not already considered. Results of our CNN-based features are reported in Table 18. As expected, the ''other'' state exhibits the worse results among the eleven states. On the overall the accuracy of our CNN-based features on the Jelodar dataset is lower than in the case of our dataset. This is to be expected in a transfer learning problem. If we compare the results obtained with the CNN-based features with those obtained in [15] (see Table 19), we can see that the

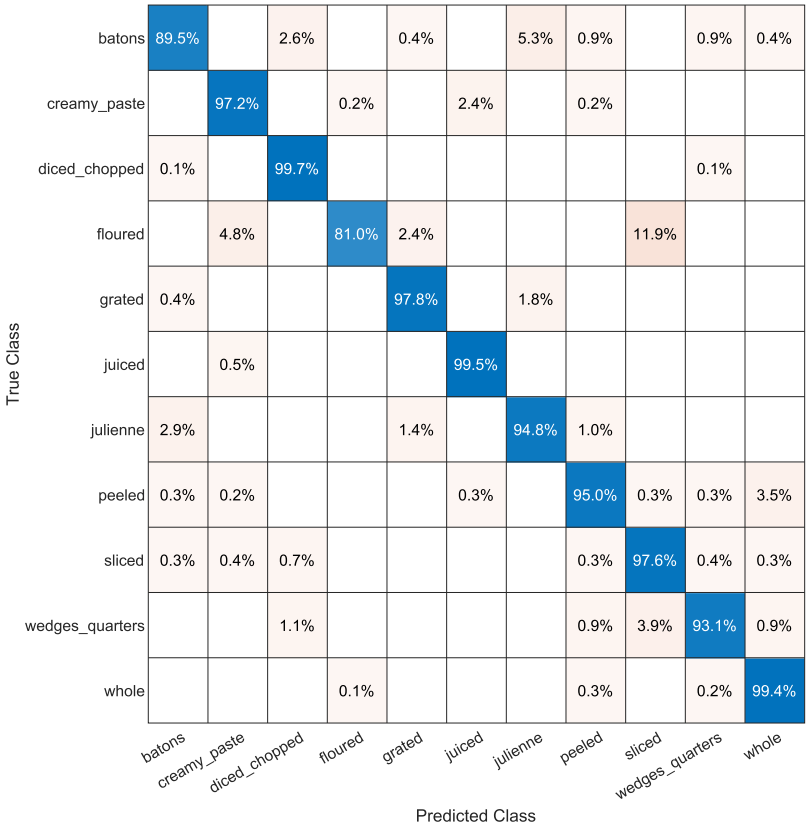

FIGURE 5. Confusion matrix of the State classification task with the CNN-based features extracted from the best performing network (Inception-v3).

best classification accuracy is obtained with the ResNet-50 (82.16%) followed by the Inception-v3 (81.25%). In this case, the MobileNet-v2 exhibits the lowest performances with an accuracy of 78.37%. The best combination of CNN-based and hand-crafted features achieves an accuracy of 81.15% which is one percentage point lower than the Inc-v3 only. Again there is no clear gain in combining the features.

As a comparison, we also report the results reported in [15]. In this case, a direct comparison is not possible because the dataset used in the original paper differs from the one used here. The results of the five methods in Table 19 have been obtained on a revised version of the original dataset that has been provided to us by the authors. However, we can see that our results are similar or better than those reported in [15] in the case of a single network, while, if multiple networks are trained on each state class, the results are about 5 percentage points lower.

From the analysis of the results we can deduce that the CNN-based features extracted from the Inception-v3 network are those that are able to achieve the overall best results on all the three tasks (see Table14). The second best features are those extracted from the ResNet-50 network. Good results of the Inception-v3 features notwithstanding, we can see that for some classes, we still have some errors. Figure 4, shows the confusion matrix of these features on the food classification task, while Figure 6 shows some examples of food

FIGURE 6. Examples of food recognized incorrectly.

incorrectly classified. We can see that apples and potatoes are often confused. Apricots are often confused with peaches in

FIGURE 7. Examples of state recognized incorrectly.

some states. Peppers are confused with tomatoes, and this is true in many states, but especially whole, but not the reverse. Garlic and onions, if diced cannot be easily distinguished. Pumpkins are mistaken for carrots in many states and this is true also for the reverse, even if at a lower error rate. Strawberries are often confused with tomatoes especially when they are juiced, or finely diced.

Figure 5, shows the confusion matrix of the Inception-v3 features on the state classification task, while Figure 7 shows some examples of state classification errors. ''batons'', ''floured'', ''julienne'' and ''wedges'' are the states where more mistakes are made. For ''batons'' and ''julienne'' the reason is that they are mistaken for each other. This is mainly due to the caliber of the cut, being 6mm or more for batons and 2-3mm for julienne. ''floured'' and ''wedges'' foods are often mistaken for ''sliced'', but not the reverse. These problems could also be attributed to the fact that these classes have fewer images than others.

V. CONCLUSION

In this paper we presented a new dataset of images for food and state recognition. A similar dataset exists in the literature, but it tackles only the problem of state recognition. We started our investigation with several questions: Can we recognize foods across different states? Can we recognize a food state independently by its identity? How robust are end-to-end Convolutional Neural Networks (CNNs)? How robust are CNN-based features with respect to hand-crafted features? Our experiments effectively show that, with the proper network, we can obtain robust features to be used for different food-related classification tasks. These features outperform hand-crafted features by a large margin. Moreover, there is no significant advantages in combining hand-crafted features with learned ones. On the overall, it seems that the state recognition problem is more approachable by the CNN-based features than the food classification one. This could be associated with the visual appearance of the state classes where the texture is more important in the discrimination of the different states. Although the best features are those extracted from the Inception-v3 network, we must acknowledge the very good results of the features extracted from the MobileNet-v2 network. For applications where the computational cost is important, the MobileNet-v2 is a perfect candidate having very good results and efficient implementation. If food classification and state classification are important tasks in a general food recognition application, it is also important to classify food at a particular state. In our experiments, we have shown that, although with slightly minor success for the two base tasks, this can be also achieved with the use of the CNN-based features. Also, if applied to unseen food states, our features are able to achieve comparable or even better results than an ad-hoc network trained end-to-end to those specific states. This demonstrates the generalization capability of the features on new domains. This also demonstrates that CNN-based features are robust with respect to the intra-class visual variability of food images. For a general food recognition system, this is a very important feature.

Good results notwithstanding, we need to further investigate the robustness of machine learning methods to the variability of real world foods in images and videos in terms of illumination, scale, point of view, and cluttered scenes. For example, discerning some type of food across some states (i.e. creamy carrot vs creamy pumpkin, or juiced strawberry vs juiced tomato) is very difficult if we rely only on visual features. An idea could be to consider other type of related features such as nutrients, ingredients, or recipe procedures. Also some food and states can be confused if the images are acquired at different scales. From the acquisition point of view, illumination plays an important role. Different lighting conditions can make it problematic to distinguish different foods [70]. An integration with a carefully designed pre-processing procedure could alleviate this problem as demonstrated in [71]. For all these reasons, as future works, we intend to perform a more systematic investigation on the effect of these issues on the recognition of the foods and states, and design possible solutions. Finally, some of our classes are under-represented and this could be a problem for proper recognition. We are planning to increase the number of images for those under-represented classes. To let other research groups contribute to the food and state recognition problem, we intend to make our dataset publicly available.[1]

ACKNOWLEDGMENT

(Gianluigi Ciocca, Giovanni Micali, Paolo Napoletano are contributed equally to this work.)

[1] http://www.ivl.disco.unimib.it/activities/food-state-recognition/

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[2] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710.

[3] Y. Matsuda, H. Hoashi, and K. Yanai, "Recognition of multiple-food images by detecting candidate regions," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2012, pp. 25–30.

[4] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, M. Yunsheng, S. Chen, and P. Hou, "A new deep learning-based food recognition system for dietary assessment on an edge computing service infrastructure," *IEEE Trans. Serv. Comput.*, vol. 11, no. 2, pp. 249–261, Mar. 2018.

[5] G. Ciocca, P. Napoletano, and R. Schettini, "Food recognition: A new dataset, experiments, and results," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 3, pp. 588–598, May 2017.

[6] W. Min, B.-K. Bao, S. Mei, Y. Zhu, Y. Rui, and S. Jiang, "You are what you eat: Exploring rich recipe information for cross-region food analysis," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 950–964, Apr. 2018.

[7] G. Ciocca, P. Napoletano, and R. Schettini, "IVLFood-WS: Recognizing food in the wild using deep learning," in *Proc. IEEE 8th Int. Conf. Consum. Electron.-Berlin (ICCE-Berlin)*, Sep. 2018, pp. 1–6.

[8] A. Mariappan, M. Bosch, F. Zhu, C. J. Boushey, D. A. Kerr, D. S. Ebert, and E. J. Delp, "Personal dietary assessment using mobile devices," *Proc. SPIE*, vol. 7246, Feb. 2009, Art. no. 72460Z.

[9] Y. Kawano and K. Yanai, "FoodCam: A real-time food recognition system on a smartphone," *Multimedia Tools Appl.*, vol. 74, no. 14, pp. 5263–5287, Jul. 2015.

[10] A. Myers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. Murphy, "Im2Calories: Towards an automated mobile vision food diary," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1233–1241.

[11] D. Sahoo, W. Hao, S. Ke, W. Xiongwei, H. Le, P. Achananuparp, E.-P. Lim, and S. C. H. Hoi, "FoodAI: Food image recognition via deep learning for smart food logging," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2019, pp. 2260–2268.

[12] K. Doman, C. Y. Kuai, T. Takahashi, I. Ide, and H. Murase, "Video CooKing: Towards the synthesis of multimedia cooking recipes," in *Proc. Int. Conf. Multimedia Modeling*. Berlin, Germany: Springer, 2011, pp. 135–145.

[13] S. Bianco, G. Ciocca, P. Napoletano, R. Schettini, R. Margherita, G. Marini, and G. Pantaleo, "Cooking action recognition with iVAT: An interactive video annotation tool," in *Proc. Int. Conf. Image Anal. Process.* Berlin, Germany: Springer, 2013, pp. 631–641.

[14] S. Bianco, G. Ciocca, P. Napoletano, and R. Schettini, "An interactive tool for manual, semi-automatic and automatic video annotation," *Comput. Vis. Image Understand.*, vol. 131, pp. 88–99, Feb. 2015.

[15] A. Babaeian Jelodar, M. Sirajus Salekin, and Y. Sun, "Identifying object states in cooking-related images," 2018, *arXiv:1805.06956*. [Online]. Available: http://arxiv.org/abs/1805.06956

[16] M. S. Salekin, A. Ba. Jelodar, and R. Kushol, "Cooking state recognition from images using inception architecture," in *Proc. Int. Conf. Robot., Electr. Signal Process. Techn. (ICREST)*, Jan. 2019, pp. 163–168.

[17] P. Napoletano, "Visual descriptors for content-based retrieval of remote-sensing images," *Int. J. Remote Sens.*, vol. 39, no. 5, pp. 1343–1376, Mar. 2018.

[18] P. Napoletano, "Hand-crafted vs learned descriptors for color texture classification," in *Proc. Int. Workshop Comput. Color Imag.* Cham, Switzerland: Springer, 2017, pp. 259–271.

[19] S. Arivazhagan, R. N. Shebiah, S. S. Nidhyanandhan, and L. Ganesan, "Fruit recognition using color and texture features," *J. Emerg. Trends Comput. Inf. Sci.*, vol. 1, no. 2, pp. 90–94, 2010.

[20] M. Bosch, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, "Combining global and local features for food identification in dietary assessment," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 1789–1792.

[21] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

[22] H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *IEEE Trans. Syst., Man, Cybern.*, vol. 8, no. 6, pp. 460–473, Jun. 1978.

[23] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[24] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[25] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891–906, Sep. 1991.

[26] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.

[27] H. Hoashi, T. Joutou, and K. Yanai, "Image recognition of 85 food categories by feature fusion," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2010, pp. 296–301.

[28] O. L. Junior, D. Delgado, V. Gonçalves, and U. Nunes, "Trainable classifier-fusion schemes: An application to pedestrian detection," in *Proc. 12th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2009, pp. 1–6.

[29] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst. (GIS)*, 2010, pp. 270–279.

[30] A. E. Abdel-Hakim and A. A. Farag, "CSIFT: A SIFT descriptor with color invariant characteristics," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jul. 2006, pp. 1978–1983.

[31] C. L. Novak and S. A. Shafer, "Anatomy of a color histogram," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jan. 2003, pp. 599–605.

[32] Y. He, C. Xu, N. Khanna, C. J. Boushey, and E. J. Delp, "Analysis of food images: Features and classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 2744–2748.

[33] G. M. Farinella, M. Moltisanti, and S. Battiato, "Classifying food images represented as bag of Textons," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 5212–5216.

[34] D. T. Nguyen, Z. Zong, P. O. Ogunbona, Y. Probst, and W. Li, "Food image classification using local appearance and global structural information," *Neurocomputing*, vol. 140, pp. 242–251, Sep. 2014.

[35] M. M. Anthimopoulos, L. Gianola, L. Scarnato, P. Diem, and S. G. Mougiakakou, "A food recognition system for diabetic patients based on an optimized bag-of-features model," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 4, pp. 1261–1271, Jul. 2014.

[36] N. Martinel, C. Piciarelli, C. Micheloni, and G. L. Foresti, "A structured committee for food recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 92–100.

[37] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

[38] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[39] E. Rahtu, J. Heikkilä, V. Ojansivu, and T. Ahonen, "Local phase quantization for blur-insensitive image analysis," *Image Vis. Comput.*, vol. 30, no. 8, pp. 501–512, Aug. 2012.

[40] Y. Guo, G. Zhao, and M. Pietikäinen, "Texture classification using a linear configuration model based descriptor," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 1–10.

[41] L. Zhang, Z. Zhou, and H. Li, "Binary Gabor pattern: An efficient and robust descriptor for texture classification," in *Proc. 19th IEEE Int. Conf. Image Process.*, Sep. 2012, pp. 81–84.

[42] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *Int. J. Comput. Vis.*, vol. 62, nos. 1–2, pp. 61–81, Apr. 2005.

[43] V. Bettadapura, E. Thomaz, A. Parnami, G. D. Abowd, and I. Essa, "Leveraging context to support automated food recognition in restaurants," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 580–587.

[44] G. Ciocca, P. Napoletano, and R. Schettini, "Food recognition and leftover estimation for daily diet monitoring," in *Proc. Int. Conf. Image Anal. Process.* Cham, Switzerland: Springer, 2015, pp. 334–341.

[45] S. A. Chatzichristofis and Y. S. Boutalis, "CEDD: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval," in *Computer Vision Systems*. Berlin, Germany: Springer, 2008, pp. 312–322.

[46] M. E. Barilla and M. Spann, "Colour-based texture image classification using the complex wavelet transform," in *Proc. 5th Int. Conf. Electr. Eng., Comput. Sci. Autom. Control*, Nov. 2008, pp. 358–363.

[47] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2249–2256.

[48] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
[49] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 512–519.
[50] Y. Kawano and K. Yanai, "Food image recognition with deep convolutional features," in *Proc. ACM Int. Joint Conf. Pervas. Ubiquitous Comput. Adjunct Publication (UbiComp)*, 2014, pp. 589–593.
[51] K. Yanai and Y. Kawano, "Food image recognition using deep convolutional network with pre-training and fine-tuning," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jun. 2015, pp. 1–6.
[52] H. Hassannejad, G. Matrella, P. Ciampolini, I. De Munari, M. Mordonini, and S. Cagnoni, "Food image recognition using very deep convolutional networks," in *Proc. 2nd Int. Workshop Multimedia Assist. Dietary Manage. (MADiMa)*. New York, NY, USA: ACM, 2016, pp. 41–49.
[53] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma, "DeepFood: Deep learning-based food image recognition for computer-aided dietary assessment," in *Proc. 14th Int. Conf. Inclusive Smart Cities Digit. Health*, vol. 9677, 2016, pp. 37–48.
[54] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 567–576.
[55] J. Chen and C.-W. Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," in *Proc. ACM Multimedia Conf. (MM)*. New York, NY, USA: ACM, 2016, pp. 32–41.
[56] S. Mezgec and B. K. Seljak, "NutriNet: A deep learning food and drink image recognition system for dietary assessment," *Nutrients*, vol. 9, no. 7, p. 657, Jun. 2017.
[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[58] G. Ciocca, P. Napoletano, and R. Schettini, "Learning CNN-based features for retrieval of food images," in *New Trends in Image Analysis and Processing—ICIAP*. Cham, Switzerland: Springer, 2017, pp. 426–434.
[59] G. Ciocca, P. Napoletano, and R. Schettini, "CNN-based features for retrieval and classification of food images," *Comput. Vis. Image Understand.*, vols. 176–177, pp. 70–77, Nov. 2018.
[60] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain, "A survey on food computing," *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–36, Sep. 2019.
[61] M. A. Subhi, S. H. Ali, and M. A. Mohammed, "Vision-based approaches for automatic food recognition and dietary assessment: A survey," *IEEE Access*, vol. 7, pp. 35370–35381, 2019.
[62] F. Bianconi, "Theoretical and experimental comparison of different approaches for color texture classification," *J. Electron. Imag.*, vol. 20, no. 4, Oct. 2011, Art. no. 043006.
[63] C. Cusano, P. Napoletano, and R. Schettini, "Illuminant invariant descriptors for color texture classification," in *Proc. Int. Workshop Comput. Color Imag.* Berlin, Germany: Springer, 2013, pp. 239–249.
[64] C. Cusano, P. Napoletano, and R. Schettini, "Combining local binary patterns and local color contrast for texture classification under varying illumination," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 31, no. 7, p. 1453, Jul. 2014.
[65] C. Cusano, P. Napoletano, and R. Schettini, "Local angular patterns for color texture classification," in *Proc. Int. Conf. Image Anal. Process.* Cham, Switzerland: Springer, 2015, pp. 111–118.
[66] S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures," *IEEE Access*, vol. 6, pp. 64270–64277, 2018.
[67] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[68] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
[69] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.
[70] G. Ciocca, D. Mazzini, and R. Schettini, "Evaluating CNN-based semantic food segmentation across illuminants," in *Computational Color Imaging* (Lecture Notes in Computer Science), S. Tominaga, R. Schettini, A. Trémeau, and T. Horiuchi, Eds. vol. 11418. Cham, Switzerland: Springer, 2019, pp. 247–259.
[71] S. Bianco, C. Cusano, P. Napoletano, and R. Schettini, "Improving CNN-based texture classification by color balancing," *J. Imaging*, vol. 3, no. 3, p. 33, Jul. 2017.

GIANLUIGI CIOCCA received the M.Sc. degree in computer science from the University of Milano, in 1998, and the Doctor of Philosophy (Ph.D.) degree in computer science from the University of Milano-Bicocca (UNIMIB), in 2006. Since 2006, he has been a Research Scientist with UNIMIB. He is currently an Associate Professor in computer science with the Department of Informatics, Systems and Communications, UNIMIB. He has published more than 100 refereed articles in international journals and conferences. His research interests focus on image and video understanding, pattern recognition, classification, and machine learning. He is also a member of the Milan Center for Neuroscience (NEUROMI).

GIOVANNI MICALI received the degree in computer science from the University of Milano-Bicocca (UNIMIB), in 2019, discussing a thesis about recognition the state of food images, where he is currently pursuing the M.Sc. degree in computer science. His current research interests are in the field of food image analysis.

PAOLO NAPOLETANO received the master's degree in telecommunications engineering from the University of Naples Federico II, in 2003, and the Doctor of Philosophy degree (Ph.D.) in information engineering from the University of Salerno, Italy, in 2007. He is currently an Assistant Professor of computer science (tenure track–RTDB) with the Department of Informatics, Systems and Communication, University of Milano-Bicocca. His master's thesis focused on transmission of electromagnetic fields and the Ph.D. thesis focused on computational vision and pattern recognition. His current research interests focus on signal, image and video analysis and understanding, multimedia information processing and management, and machine learning for multimodal data classification and understanding.