Towards Personalised Training of Machine Learning Algorithms for Food Image Classification Using a Smartphone Camera

Patrick McAllister[1], Huiru Zheng[1(✉)], Raymond Bond[1], and Anne Moorhead[2]

[1] School of Computing and Mathematics, Ulster University,
Newtownabbey, Northern Ireland
`mcallister-p2@email.ulster.ac.uk`, {`h.zheng,rb.bond`}`@ulster.ac.uk`
[2] School of Communication, Ulster University, Newtownabbey, Northern Ireland
`a.moorhead@ulster.ac.uk`

Abstract. This work is related to the development of a personalised machine learning algorithm that is able to classify food images for food logging. The algorithm would be personalised as it would allow users to decided what food items the model will be able to classify. This novel concept introduces the idea of promoting dietary monitoring through classifying food images for food logging by personalising a machine learning algorithm. The food image classification algorithm will be trained based on specific types of foods decided by the user (most popular foods, food types e.g. vegetarian). This would mean that the classification algorithm would not have to be trained using a wide variety of foods which may lead to low accuracy rate but only a small number of foods chosen by the user. To test the concept, a range of experiments were completed using 30 different food types. Each food category contained 100 images. To train a classification algorithm, features were extracted from each food type, features such as SURF, LAB colour features, SFTA, and Local Binary Patterns were used. A number of classification algorithms were used in these experiments; Nave Bayes, SMO, Neural Networks, and Random Forest. The highest accuracy achieved in this work was 69.43 % accuracy using Bag-of-Features (BoF) Colour, BoF-SURF, SFTA, and LBP using a Neural Network.

Keywords: Obesity · Machine learning · Classification · Food logging · Photographs

1 Introduction

Obesity is increasing globally [1] and is the cause of many chronic conditions such as diabetes, heart disease, hypertension, and some cancers [2]. Within the UK, 61 % of adults are classified as either obese or overweight. Within the period from 1993 to 2013 obesity levels in men increased from 13.2 % to 26 % and 16.4 % to 23.8 % for women [3]. The main cause of obesity is a result of a high fat/calorie

diet and when the energy is not burned off through physical activitiese, then the excess energy will be stored as fat around the body [2].

The high prevalence of obesity also puts an economic burden on governments and health institutions around the world [3–5]. Information Communication Technologies (ICT) have been developed to allow individuals to self-manage their diet. There is a plethora of Smartphone applications available that allows users to document their energy intake. These applications allow the user to search for a food item and to determine the energy intake, however this can be cumbersome and time consuming for the user since they are required to navigate through numerous drop down menus to identify the correct food item [6]. A convenient approach would be to take photograph of a meal using a Smartphone camera and use the touch screen to draw around the food portion which can be automatically classified using computer vision methods. The amount of calories in a food item can also be estimated by taking in account the geometric area of the food portion, which can correlated with the amount of calories. The food item's area can be calculated using reference such as a coin in the photograph or a shape (area of reference shape is known to the user). Work has been completed in this area by using a reference point to determine portion size [7]). The food portion classification would then be used to search for the calorie content and portion size.

This paper focuses on using computer vision methods to extract features from a food image dataset and then used to train machine learning models to classify food items within the images. The rationale for the work presented in this paper is to inform the development of a personalised classification model that is tailored to the user's food selection. Much of the research completed in this area [10–12] use a multitude of classification models to classify a large range of food categories. Moreover, these models result in inaccuracies due to the large number of food classes the model can classify, this can be seen as wasteful since many of these food classes are not needed within an individuals diet. For example, a user may be vegetarian and would only be interested in using a system that is able to classify a selection of vegetables or meals that would be in their diet. We envisage that a user would select these foods and submit their selections to a cloud service where a classification model would be trained using features extracted from the chosen food categories. The personalised classification model would be downloaded onto the users device. The work presented in this paper uses a smaller number of food categories that are hypothetically selected by the user to classify food items. This system is described in Fig. 2. This work will consist of a range of experiments using a food image dataset (representing foods from typical food groups) and using feature combination and classification techniques to predict the food type. The remainder of this paper will discuss related work in this area; the methods used in the experiment which relate to feature detection and descriptors; the machine learning classification techniques used and the statistical methods used for evaluation.

2   Related Work

Much research has been conducted that use computer vision techniques for classifying objects in images. In [8] a classification technique using Random Forests was trained to mine images for discriminative parts (e.g. super-pixels). The discriminative components that were identified were then shared in order to improve accuracy. A challenging food image dataset was also constructed in [8]. This dataset is described as challenging as most of the images may contain multiple items and food types within each image. The dataset provided consisted of 101 food categories with 1000 images in each food category. The research presented in [8] also used Bag-of-Features (BoF), Histogram of Oriented Gradients (HOG), and LAB colour values were used. An accuracy of 50.76 % was achieved when using this dataset [8]. In [9] food texture and local gradient features were used to identify and classify food categories for dietary assessment. The texture and local gradient features used were entropy-based categorization and fractal dimension estimation (EFD), Gabor-based image decomposition and fractal dimension estimation (GFD), and the third descriptor is based on the spatial relationship of gradient orientations (GOSDM). GOSDM is obtained by finding out of the occurrence rate of pairs of gradient orientations of neighbourhood scales. The number of food categories used in this work is 46. The food items used in this experiment were segmented from the scene from feature extraction. Results from this work show that EFD with Neural Network achieved 79.2 %, GFD with Neural Network achieved 72.2 %, and also GOSDM with Neural Network achieved 65.3 % [9]. Also in [13] a real time food classification system was developed that utilises bounding boxes as an adjunct to segment the food area within an image in order to classify the region. In [13] two types of features were utilised; (1) BoF with a colour histogram and (2) a HOG patch and colour patch descriptor with Fisher-vector representation. As a result, the classification model [21] achieved 79 % accuracy for the top 5 categories. The work presented
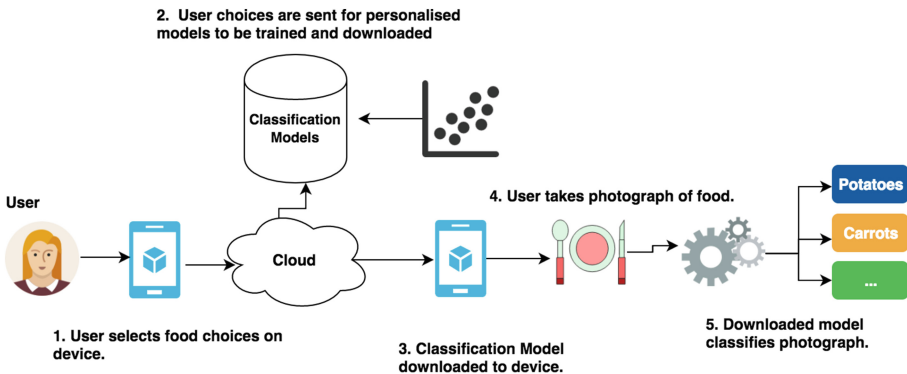

Fig. 1. Diagram describing proposed system that would allow users to download classification models.

in this paper seeks to explore the use of conventional machine learning classifiers for non-segmented food images to inform the development of a user tailored classification model.

3   Aim and Objectives

The aim of this work is to classify different images of foods using a combination of feature types with different machine learning algorithms. The number of food classes used in the image dataset for this work is 30. We also used different machine learning algorithms classifying the food items. This work will combine global and local features to classify images of food. To achieve this aim, a number of objectives need to be completed; (1) an image dataset would need to be collected consisting of different food categories, (2) different types of feature descriptors need to be extracted and represented as feature vectors for each image in the image dataset, (3) a number of supervised classifiers need to be used and evaluated using 10-fold cross validation. The extracted feature vectors will then be encoded into the classifiers. The final objective is (5) which involves evaluation metrics to assess the performance of each classifier and feature combination, and in turn to determine the best classifier and feature combination.

4   Methodology

4.1   Image Dataset

There has been much research dedicated to constructing food image datasets for the purpose of research computer vision classification methods [10–12]. The images used in this work was taken from [10]. Thirty food item types was used for this work.

4.2   Feature Selection

This section will discuss feature types used in this work. The feature types used in this work will consist of global and local features. LAB colour space statistics will be extracted from the image dataset and a standard bag-of-features (BoF) method will be applied to the extracted colour features to create a visual dictionary. Local features will also be extracted from the image dataset; Speed-Up-Robust-Features (SURF) will be used to extract features from the food image dataset. Again a BoF model will be applied to the SURF features to create a visual vocabulary to classify images. Segmented Fractal Textual Analysis (SFTA) and Local Binary Patterns (LBP) will also be used in this work. This section will give a brief overview of these methods.

4.3   Bag of Features

Bag-of-features (BoF) or bag-of-visual words (BoVW) is a technique that is used to describe an image through a series of visual word occurrences using a visual dictionary. A vector is then produced after using a feature extraction method which represents features in an image. The vector is created through using an interest point extractor and then applying a descriptor such as SURF or SIFT to represent or describe the area around these points. BoF technique uses a code book or a visual dictionary that is created using features extracted from the training image set. Each visual word in the visual dictionary represents patches in a visual dictionary. An image can be classified by counting the amount of visual word occurrences that are present in the visual dictionary. The results feature vector can then be quantified using a histogram to represent the number of visual word occurrences in an image.

4.4   Speeded-Up-Robust-Features (SURF)

In this work, SURF feature descriptor was used. SURF is based on using a Hessian matrix to determining interest points. BLOB (large binary object) elements, used in SURF algorithm, are detected at a location where the determinant of Hessian is at maximum. The determinant of Hessian can also be used to for scale selection [14]. The following Eq. (1) defines the Hessian matrix at point X in an image.

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) \ L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) \ L_{yy}(x, \sigma) \end{bmatrix} \tag{1}$$

$L_{xx}(x, \sigma)$, $L_{xy}(x, \sigma)$ and $L_{yy}(x, \sigma)$ represent the convolution of the Gaussian second order derivative in image I at point x [4]. The SURF algorithm uses Hessian matrix and integral images to allow for quicker calculations. Convolutions are accelerated using the integral images method. In an integral image, a location represents the sum of all pixels in an image within a certain region. This is described in (2) [14].

$$I(x, y) = i(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1) \tag{2}$$

(x,y) represents a point in a rectangular region. The remaining points I(x−1,y), I(x,y−1) and (1,1) represent the remaining points in the rectangular region. This process only requires three operations to compute the value of the region [14]. In this work, a grid feature detection method was used. An 15×15 pixel grid was placed across each image and the SURF features were extracted from the locations where each horizontal and vertical grid connected. Features at these points were extracted at different scales to promote scale invariance.

4.5   Segmentation Based Fractal Textual Analysis (SFTA)

SFTA is a feature extraction method that is able to extract texture information from an image [15]. The algorithm accepts an input image and the images

are then decomposd into multiple binary images using a Two-Threshold Binary Decomposition (TTBD) method.

$$I_b(x,y) = \begin{cases} 1 & if \ t_l < I(x,y) \le t_u \\ 0, & otherwise \end{cases} \tag{3}$$

where $I(x,y)$ is a set of binary images. Binary images are computed by using thresholds from T and using the Two-threshold segmentation as described in (1) [15]. t_l and t_u represent a pair of upper and lower thresholds. Pairs of thresholds are applied to the input image to obtain a set of binary images. The reason for applying pairs of thresholds to obtain binary images is to ensure that objects in the input images are segmented. The binary images that are outputted from the TTBD method can be described as a sub set of binary images that would have been outputted using a single threshold algorithm. SFTA feature vector is constructed using the binary images by extracting the pixel count (size), gray level and boundaries fractal dimension [15]. These measurements are used to describe object boundaries in each input image. The SFTA feature vector size is directly related to the number of binary images generated using the TTBD algorithm, for example if eight images were computed after using the TTBD algorithm on an input image, the SFTA feature vector would be 8×3 (3 being the number of measurements extracted from the binary images: size, fractal dimension, and mean gray level).

4.6   LAB Colour Space

Global colour features were extracted from the food image dataset and used within a BoF model to create a visual dictionary to classify test images. Lab colour space is described as a 3 axis colour system; L representing lightness and A and B representing colour dimensions [16]. There are several advantages to using LAB colour space as a method to represent colour in images; it provides a precise means of representing colour and LAB is device independent and also LAB colour space can easily be quantified to compare images. [16]. In this work, RGB images are converted to LAB colour space. The image is divided in 16×16 pixel blocks and the average value of each block is computed. The image is then scaled down in order to compute the average LAB colour value over the entire image. The average LAB values are then stored in a matrix and normalised. The location from where the colour feature was extracted and appended to the feature.

4.7   Local Binary Patterns (LBP)

Local Binary Patterns (LBP) is a visual descriptor that has been used for texture classification. To create an LBP vector the following method is used, firstly, the area in question is divided into a number of the cells. The cells in the area are measured 3×3 pixels usually. The center pixel in the cell is compared with its neighbours. If the center pixels value is greater than the neighbour, then the

neighbouring pixel is assigned as 0 or if the neighbouring pixel value is greater than the center pixel then it is assigned 1. After this process is completed, a binary sequence is then computed for each pixel within the cell. The binary sequence is computed to reveal an LBP code. A histogram is then generated to statistically measure the occurrence of LBP codes in an image. This histogram can then be used to classify an image [17].

4.8   Classifiers

In this work a range of classifiers were used to assess the performance using the extracted features types extracted. Table 1 is a list of the machine learning classifiers used in this work.

Table 1. Summary of classifiers and parameters used in this work.

Machine learning classifiers	Parameters used
Naive Bayes (NB)	Weka default parameters
SMO [19]	Polynomial Kernel
Neural Network (NN) [24]	1 layer, 100 neurons, 1000 iterations
Random Forest Tree (RF) [18]	300 trees

4.9   Evaluation and Statistical Analysis

Metrics were used to evaluate the performance of the machine learning algorithms. Ten-fold cross validation was used to accurately calculate the performance metrics. The output of the 10-fold cross validation included the kappa statistic for each experiment. This also included the mean percentage accuracy rate (number of correct classifications) as computed from each of the folds. Cohens Kappa was used to measure the agreement between the predicted class and the actual class for each food image. Initial experiments consisted of increasing the visual word count in the BoF model using 500 increments. This was done for BOF-SURF and BoF-colour. This was to find out other optimum visual word count for each classifier by using percentage accuracy as a measurement. The highest accuracy achieved for each classifier (using the 500 visual word increments) would be combined with the remained feature sets extracted from the image dataset. The labelled feature set combinations were extracted to a CSV file format using Matlab (R2016a) [22] and the Weka Analysis (v3.7.13) [23] platform was used to train machine learning algorithms using the features extracted.

5   Results

Experiments were completed using the image dataset described in Sect. 4. The image dataset consisted of images with other food items and other objects in

them i.e. noise and unrelated food items. Various combinations of BoF-SURF and BoF-Colour were fused together with SFTA and LBP features to achieve the highest result. Table 2 shows the percentage accuracy of increasing the visual words using BoF-SURF and BoF-colour features in a BoF model using for each machine learning classifier.

Results from the visual word experiments show the percentage accuracy achieved for each classifier. The results from these experiments were incorporated into future classification tests. Future experiments were carried out by combining feature types for each machine learning algorithms. SURF and colour visual words that achieved the highest accuracy were combined together for each

Table 2. Results from increasing the visual word count by 500 for SURF and colour features using BOF method. SMO classifier (SMO) and Naive Bayes (NB) was used in these experiments.([a] denotes highest accuracy achieved).

Visual Words	SMO SURF	SMO Colour	NB SURF	NB Colour
500	46.30	34.67	22.73	19.20
1000	48.17	33.43	23.87	21.37
1500	48.87	34.00	24.67	21.83
2000	50.17	34.73	25.13	22.90
2500	50.27[a]	35.67	25.63	24.00
3000	50.27	36.43	25.47	24.40
3500	49.90	35.60	26.40[a]	24.53
4000	49.70	35.87	25.70	24.67
4500	49.50	36.57[a]	25.17	24.73
5000	49.47	35.80	25.23	25.37[a]

Table 3. Initial results from increasing the visual word count by 500 for SURF and colour features using BOF method. Neural Network (NN) and Random Forest (RF) classifier were used in these experiments. ([a] denotes highest accuracy achieved).

Visual Words	NN SURF	NN Colour	RF SURF	RF Colour
500	49.80	38.67[a]	36.43[a]	40.87[a]
1000	52.37	37.93	35.47	40.27
1500	54.43	37.00	34.93	38.40
2000	54.33	36.33	35.07	38.23
2500	55.70	36.83	34.13	38.00
3000	54.47	35.53	33.90	38.03
3500	55.53	36.87	33.90	37.83
4000	56.33[a]	36.70	33.27	37.07
4500	55.97	36.37	32.80	37.33
5000	55.90	36.66	33.40	36.23

Table 4. Results from combining features together. (^{a} denotes highest accuracy achieved).

Classifier	Colour+ SURF	SURF+Colour SFTA	SURF+Colour+ SFTA+LBP	SURF+LBP
SMO	61.9	63.23	64.57	52.1
Naive Bayes	33.63	33.7	34.00	28.26
Neural Network	67.00^a	68.73^a	69.43^a	60.03^a
Random Forest	48.90	50.30	48.07	37.63

Table 5. Further results combining different feature types together. (^{a} denotes highest accuracy achieved).

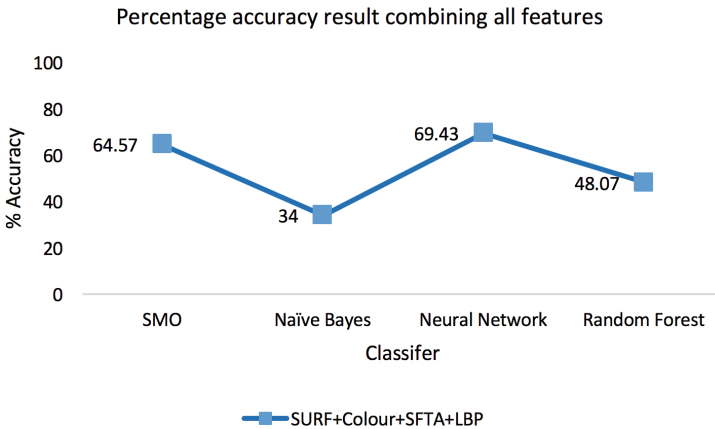| Classifier	SURF+SFTA	Colour+SFTA
SMO	53.8	40.53
Naive Bayes	27.1	25.56
Neural Network	59.26^a	44.43^a
Random Forest	40.33	43.53

Fig. 2. Diagram showing final results from combining features across different machine learning classifiers.

classifier e.g. SURF and colour features that achieved the highest accuracy for SMO were combined. Figure 2 shows the results of using feature combinations trained using the machine learning classification algorithms. The results from the 10-fold cross validation show that Neural Network trained with BoF-Colour, BoF-SURF, SFTA, and LBP feature combination achieved the highest accuracy with 69.43 %. From the combination feature results, Neural Network achieved the highest accuracy in all feature combination experiments (Table 4).
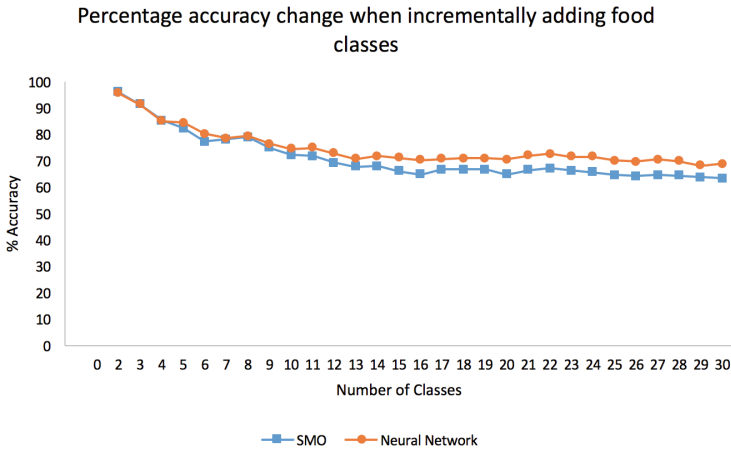
Fig. 3. Diagram describing the change in percentage accuracy when incrementally adding food classes to an image dataset. For this experiment SMO classifier was used with BoF-SURF, BoF-colour, and SFTA.
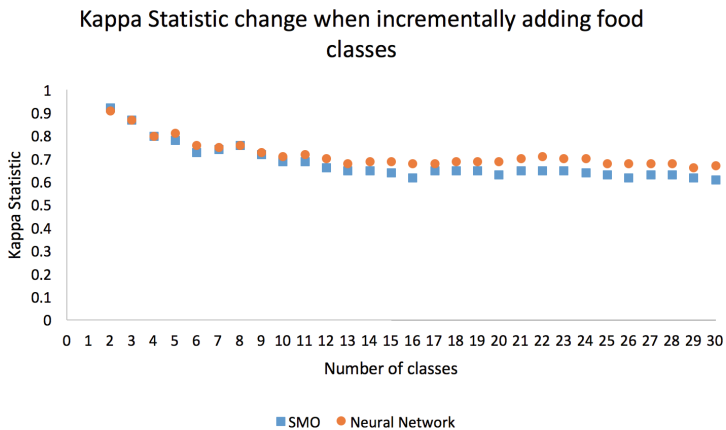

Fig. 4. describing the change in Cohen's Kappa when incrementally adding food classes to an image dataset. SMO classifier was used with BoF-SURF, BoF-colour, and SFTA

Further experiments were conducted to depict the decline in accuracy when incrementally increasing the number of classes. Cohen's Kappa was noted from each experiment to measure the performance of each iteration. Figures 3 and 4 shows the results from these experiments. Figure 3 is a graph showing the percentage accuracy change when food classes were adding incrementally to the dataset. Figure 4 is a graph that depicts the Kappa Statistic change when classes were incrementally added.

6   Discussion

This work uses a feature combination approach to train several machine learning models. The motivation for this work was to inform the development of a personalised machine learning model approach to classify food images. Relating to Fig. 1, the user would be able to select their favourite foods for classification to predict the food meal in the image. The performance of this work was assessed using a 10-fold cross validation approach and results show that Neural Network trained with BoF SURF, BoF colour, SFTA, and LBP achieves the highest accuracy with 69.43 % accuracy. Neural Network consistently achieved higher accuracy across all feature combinations, and Nave Bayes achieved the lowest accuracy in each feature combination test. Table 6 is a comparison table from other works completed along with the accuracy and feature types, this shows that the results achieved in this work is comparable with other results achieved in this area. It is important to note that the images used in this work were not segmented but the entire image was used for feature selection. From the experiments, it is revealed that a reasonable degree of accuracy can be achieved through classifying non-segmented meal images. This could be increased by segmenting the meals to promote feature selection accuracy and ultimately classification accuracy. This work shows that there is potential to utilise conventional based feature extraction and machine learning classifiers to classify entire food meal images with reasonable accuracy however more comparative research is needed to compare further feature extraction methods (CNN feature extraction) (Table 6).

Table 6. Table showing highest accuracy achieved in other work in food classification.

Method	% Accuracy
SVM + mixture of global and local features [20]	0.861 (average across all tests, 39 classes)
BoF/Fisher-Vector [21]	79.2 % (top 5 classes)
Random Forest for component mining [8]	50.76 (101 classes)
Texture Features + Neural Networks [9]	79.2 % (segmented food images, 46 classes)
Neural Network (this paper)	69.43 % (30 classes, unsegmented images)

7   Future Work and Conclusion

Several limitations have been identified in this work. Some of the images used in this work for each category include other objects or food items are present in the image. Certain non-food features may be selected and used in the training of

the machine learning process, which can result in a number of misclassifications. Future work will be to address this issue by creating an image dataset using food images that focus in on the food item and texture directly and to ensure that no other non-food items or other food items are present in the scene. In future work, the food items would be segmented from the image and then feature types would be extracted from the segmented image. This would improve the algorithms accuracy by allowing relevant interest points to be selected. The number of images in this work was 100 per category, which can be considered to be a low number in comparison to other works. Future work would address this issue by increasing the amount of images in each category and ensure that these images do not contain any other food or non-food item. As a result, the increase in training data for each food category should also increase the accuracy of the algorithm. Other machine learning models would also be considered in future work; further analysis could be undertaken by changing different parameters for each model used, e.g. changing the number of layers in neural network structure along with the amount of neurons or changing kernels used in SMO classifier. Other machine learnings could be applied to the image dataset such as Self-Organizing Maps (SOM) or utilise multi-class classifier approaches and document the performance of these techniques (one vs one, one vs rest). For feature extraction, other feature types could be used such as Gabor Filters to extract textual information from the dataset. Research would also focus on developing an hierarchical classification approach to classify food type and then pinpoint exact food item.

References

1. DHSSPSNI, Health Survey Northern Ireland, First Results 2013/2014 (2013–2014)
2. Nhs.uk, "Obesity, NHS Choices", Nhs.uk (2016). www.nhs.uk/Conditions/Obesity/Pages/Introduction.aspx. Accessed 15 Jun 2016
3. Noo.org.uk "About Obesity: Public Health England" (2016). www.noo.org.uk/NOO_about_obesity/adult_obesity/UK_prevalence_and_trends
4. Scarborough, P., Bhatnagar, P., Wickramasinghe, K., Allender, S., Foster, C., Rayner, M.: The economic burden of ill health due to diet, physical inactivity, smoking, alcohol and obesity in the UK: an update to 2006–07 NHS costs. J. Public Health **33**(4), 527–535 (2011)
5. Noo.org.uk, "Economics of obesity: Public Health England Obesity Knowledge and Intelligence team", Noo.org.uk (2016). www.noo.org.uk/NOO_about_obesity/economics. Accessed 15 Jun 2016
6. MyFitnessPal, C.: M. LLc, "Calorie Counter & Diet Tracker by MyFitnessPal on the App. Store", App. Store (2016). itunes.apple.com/gb/app/calorie-counter-diet-tracker/id341232718?mt=8. Accessed 15 Jun 2016
7. McAllister, P., et al.: Semi-automated system for predicting calories in photographs of meals. In: IEEE International Conference on Engineering, Technology and Innovation/International Technology Management Conference (ICE/ITMC). IEEE (2015)
8. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VI. LNCS, vol. 8694, pp. 446–461. Springer, Heidelberg (2014)

9. Bosch, M., et. al.: Food texture descriptors based on fractal and local gradient information. In: 19th European Signal Processing Conference, pp. 764–768. IEEE (2011)

10. Farinella, G.M., Allegra, D., Stanco, F.: A benchmark dataset to study the representation of food images. In: ECCV European Conference in Computer Vision. Zurich, Workshop Assistive Computer Vision and Robotics (2014)

11. 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)

12. Kawano, Y., Yanai, K.: Foodcam-256: a large-scalereal-time mobile food recognition system employing high-dimensional features and compression of classifier weights. In: Proceedings of the ACM International Conference on Multimedia, ser. MM 14, pp. 761–762 (2014)

13. Hartigan, J.A., Manchek, A.W.: Algorithm AS 136: a k-means clustering algorithm. J. Roy. Stat. Soc.: Ser. C (Appl. Stat.) **28**(1), 100–108 (1979)

14. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)

15. Costa, A.F., Humpire-Mamani, G., Traina, A.J.M.: An efficient algorithm for fractal analysis of textures. In: 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). IEEE (2012)

16. "Lab Color - MATLAB", Uk.mathworks.com (2016). http://uk.mathworks.com/discovery/lab-color.html. Accessed 15 Jun 2016

17. Ojala, T., Pietikinen, M., Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: Proceedings of the 12th IAPR International Conference on Pattern Recognition (ICPPR 1994), vol. 1, pp. 582–585 (1994)

18. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)

19. Platt, J.C.: Sequential Minimal Optimization: a fast algorithm for training support vector machines. In: Advances Kernel Methods, pp. 185–208 (1998)

20. Bosch, M., Zhu, F., Khanna, N., Boushey, C.J., Delp, E.J.: Combining global and local features for food identification in dietary assessment. In: Proceedings - International Conference on Image Processing, ICIP, pp. 1789–1792 (2011)

21. Kawano, Y., Yanai, K.: FoodCam: a real-time mobile food recognition system employing fisher vector. In: Gurrin, C., Hopfgartner, F., Hurst, W., Johansen, H., Lee, H., O'Connor, N. (eds.) MMM 2014, Part II. LNCS, vol. 8326, pp. 369–373. Springer, Heidelberg (2014)

22. MATLAB - MathWorks: Matlab.co.uk (2016). http://matlab.co.uk. Accessed 15 Jun 2016

23. Weka 3 - Data Mining with Open Source Machine Learning Software in Java. Cs.waikato.ac.nz (2016). http://www.cs.waikato.ac.nz/ml/weka/. Accessed 15 Jun 2016

24. (Convolutional) Neural Network, "amten/NeuralNetwork", GitHub (2016). https://github.com/amten/NeuralNetwork. Accessed 15 Sep 2016