Active Labeling Application Applied to Food-Related Object Recognition

Marc Bolaños
Dept. MAIA, U. de Barcelona
Gran Via de les Corts
Catalanes 585
Barcelona 08007, Spain
mark.bs.1991@gmail.com

Maite Garolera
Neeuropychology Unit,
Consorci Sanitari de Terrassa
Ctra.Torrebonica s/n
08227, Terrassa, Spain
mgarolera@cst.cat

Petia Radeva
Dept. MAIA, U. de Barcelona
Gran Via 585, Barcelona
08007, Spain
CVC, Campus UAB, Bellaterra
(Barcelona), Spain
petia.ivanova@ub.edu

ABSTRACT

Every day, lifelogging devices, available for recording different aspects of our daily life, increase in number, quality and functions, just like the multiple applications that we give to them. Applying wearable devices to analyse the nutritional habits of people is a challenging application based on acquiring and analyzing life records in long periods of time. However, to extract the information of interest related to the eating patterns of people, we need automatic methods to process large amount of life-logging data (e.g. recognition of food-related objects). Creating a rich set of manually labeled samples to train the algorithms is slow, tedious and subjective. To address this problem, we propose a novel method in the framework of Active Labeling for constructing a training set of thousands of images. Inspired by the hierarchical sampling method for active learning [6], we propose an Active forest that organizes hierarchically the data for easy and fast labeling. Moreover, introducing a classifier into the hierarchical structures, as well as transforming the feature space for better data clustering, additionally improve the algorithm. Our method is successfully tested to label 89.700 food-related objects and achieves significant reduction in expert time labelling.

Categories and Subject Descriptors

Computing Methodologies [**Image Preprocessing and Computer Vision**]: Scene Analysis—*Time-varying imagery; Color; Object recognition; Shape*

Keywords

Active labelling, food-related object recognition.

1. INTRODUCTION

It is clear that every day, technology is a little bit more present in our daily life. There are unlimited aspects and

CEA'13, October 21 2013, Barcelona, Spain.
Copyright 2013 ACM 978-1-4503-2392-5/13/10 ...$15.00.
http://dx.doi.org/10.1145/2506023.2506032 .

Figure 1: Picture of the wearable camera SenseCam.

usual tasks in which Pervasive Computing (Ubiquitous Computing) can improve our quality of life. A way in which this emerging field can help us the most, is based on our feeding habits and all their related aspects: nutrition, physical activities, emotions and social interaction. And one of the most evident problems, for which we could be interested in logging every bit of the diet of a person, would be healthy weight management.

An adequate and rich nutrition is clearly an important issue to take into account for anyone who wants to be healthy. Nutrition problems are widely known in our society, although not everyone is concerned about it and does much to solve them. Obesity and anorexia are diseases called ”diseases of the XXI century”. Given the advantages of keeping a record of feeding habits, interventional psychologists treating obese people, ask to record their lifestyle by writing diaries with annotation of all feeding activities during the days. However, several studies have reported that people tend to underestimate their food intake, meanwhile overestimating their physical activities [18].

An application based on lifelogging could make a big leap to solve this problem. People who clearly need help with their nutrition-related habits, could get an incredible benefit by collecting more explicit and objective the information related to their day-to-day by wearable cameras. Moreover, taking into account the importance of nutrition to prevent diseases, every person could also take a great advantage from a device like that. Hence, life-logging by a wearable camera appears as a natural solution by being able to objectively acquire day-by-day the feeding habits of persons. Starting by recognising objects related to feeding, and ending by being able to analyse the components of every meal, as well as defining the eating patterns of people, are important steps towards the success [7]. In this paper, we focus on using lifelogging technology and dealing with the huge amount of data that it produces to recognise objects closely related to nutrition and eating habits, more precisely, automatic recognition of dish objects in lifelogging records. To our knowledge, this work for first time addresses the problem of Active Labeling applied to the field of food-related object recognition.

1.1 Lifelogging and Wearable Devices

Lifelogging refers to the process of capturing large portions of peopleŠs lives by typically wearing computer or other digital devices. There are many different devices that help us logging any kind of information related to our daily routine, or any other sensor that can ”sense” what we are doing, where we are, what we are looking at, etc. Even our smart phones can become a powerful lifelogging machine that can feed us with a wide range of useful information. A. Sellen and S. Whittaker in [23] summarized the benefits of pervasive computing, in general, and lifelogging, in particular, as the ”Five Rs”: recollecting, reminiscing, retrieving, reflecting and remembering intentions.

For logging the nutritional habits of a person, we can use one of the multiple wearable cameras that are available in the market for general users or researchers. In this work, we use the Microsoft's SenseCam (see Figure 1). SenseCam is designed for any lifelogging issues, although, its main research goal, when it was created, was improving the memory retention of Alzheimer's patients. S. Hodges et al. [15] proved that people with memory problems using SenseCam used to delay significantly the progression of the disease by reviewing their records captured by the camera. Since then, SenseCam and other wearable cameras have been successfully applied for multiple purposes [15, 16, 7].

SenseCam is able of taking on average about 4200 images per day. Once switched on, it captures continuously 2 frames per minute, up to 12 hours per day. The camera has a privacy button to be switched-off, when necessary. Once connected to a computer, all pictures are automatically downloaded and removed from the camera.

1.2 Food-Related Object Recognition

Given the problem of monitoring the nutritional habits of an individual, we reformulate it as a problem of recognising objects related to nutrition in the surroundings of our subject. Analysing the photos taken by the SenseCam along the day, we want to know when and for how long people are in contact with food. Thus, similarly as it had been done in [20, 14], we need to construct a robust classifier to automatically detect food-related objects, taking into account the variation of their instances in all of their variants, shapes and positions.

Today, there is a large battery of supervised classification algorithms (KNN, AdaBoost, Decision Trees, Neural Networks, Support Vector Machines, etc.) [9] applied to many problems in computer vision. A common feature for most of them is that they need a large set of training data to achieve high performance. This amount of input data is necessary to learn the important patterns that describe the objects in order to be able to automatically determine if a new object tested by the classifier, is or not an instance of the substance that we are detecting in the images.

The main disadvantages of most supervised classifier techniques are that: 1) we need large amounts of labeled training samples, and 2) the variables (features) related to our objects are too complex in order to develop a highly reliable unsupervised classifier. Active learning is a field of machine learning whose main purpose is guiding the process of manually annotating large amounts of data optimizing it [24, 6]. In contrast to Active learning, where the main goal is to learn in the fastest possible way, the goal of Active labeling is to guide the labeling process of all samples the fastest way possible (minimizing time, effort, clicks, etc.). This problem can be of interest in severl applications such as annotating all samples of a set for training or validation purposes.

1.3 Active Labeling

The basic features of an active learning method are: environments on which we have large numbers of samples to label, and the necessity of an expert to label and validate the training samples [25, 11]. Roughly speaking, we can divide most of the Active Learning approaches into two types [5]:

1. *Classifier-Based Active Learning:* Having a distribution of samples and a subset already labeled by a master, the method determines the next region of data distribution (based on the classifier that we are using) to be inspected by the master, which is where the classifier's uncertainty of unlabeled samples is higher [17].
2. *Data Distribution-Based Active Learning:* In this approach, the data queried is based on information of data distribution instead of classifier performance.

Hierarchical Sampling (HS) method [6, 5] forms a part of the second group. This particular method starts by creating a Hierarchical Cluster binary Tree (HCT) of all the samples to label using their features as guidelines for the partition. The HCT is constructed using the Single Linkage approach [13, 12] and K-Means [19] clustering. Once the HCT has been created, the labels of samples of the different clusters are queried to the master according to a certain criterion. The algorithm uses an uncertainty and purity measure (called score or bound) in order to decide: 1) how pure the clusters (nodes in the HCT) are, according to their known labels, and 2) if their sub-clusters should be considered. Thus, once the master has answered the labels of an iteration, the next queried cluster (node of the HCT) depends on its ”purity” (the more similar the samples' labels are, the higher the probability of the cluster being chosen to label will be) [6]. We chose the HS algorithm, as a main skeleton of this work, due to its two main advantages:

1. It guides the labeling process through a previously calculated HCT depending on the impurity (or uncertainty) of the set of labels that reside in that particular level of the HCT, traveling downwards dividing the sets but never upwards.
2. It presents a method to explicitly obtain the error bounds or impurity of each of those clusters after each query step.

Due to these properties, the algorithm optimizes significantly the master's labeling effort and achieves results very competitive to the state-of-the-art, as verified by labeling several public domain databases [6, 5].

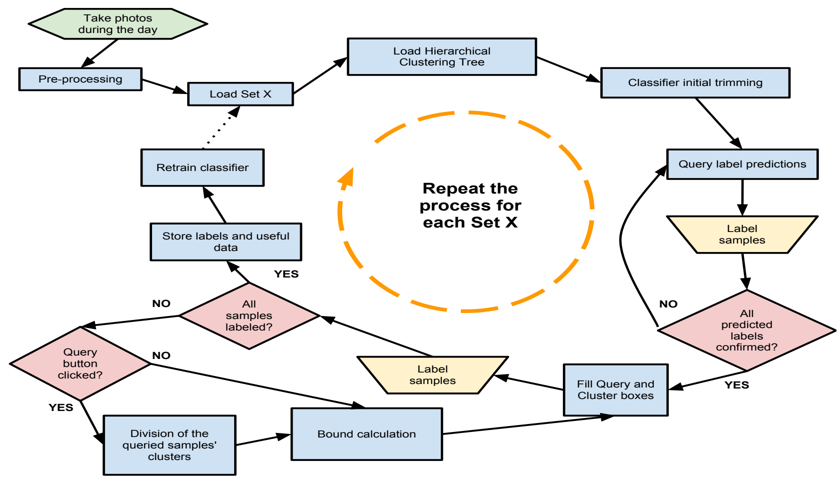

Figure 2: Scheme summarizing the basic functioning of the food-related object recognition application.

2. ACTIVE FOREST METHOD

The technique we propose is for creating a large set of training data for a food-related classifier. It follows the HS approach to treat the images of interest. We extend it by integrating the method used by [8, 21] which gives the master the authority of labeling some queries (Queries) sampled randomly [6] as well as validating a whole cluster (Cluster) with high value of purity[8]. In the latter case, homogeneous clusters are labeled with a single click. Thus, the algorithm achieves to optimize the number of clicks and decrease the time of unlabeled samples annotation.

Given our goal, which is labeling thousands of high dimensional samples in the minimum time possible, the set to label can be so huge that applying HS is getting computationally infeasible. Taking that into account, we made our data come in batches (e.g. corresponding to different days), we propose a novel method, called *Active Forest* which consists in splitting the data into different bags, for which different HCTs are constructed.

Our structure can be visualized as a forest of trees. A natural question arises: once the samples of a tree have been labeled, how do we transfer this knowledge to the next tree. Before constructing the next tree, we train a classifier on the objects of the already labeled data HCTs, and apply it to trim the samples set before constructing its HCT. All samples from the next set that are classified with high confidence, are labeled as a group by the master, and the rest is used to construct the HCT. Thus, in parallel to the labeling process, the Active forest also contains a supervised classifier that can predict the class labels of the data and reduce the samples to be labeled on the next step. Without loss of generality, we tested two of the most popular classifiers: K-NN [22] and AdaBoost [10], although any other supervised classifier can be applied.

Adding the supervised trimming data process as a step before each labeling session of the next tree, gives us the following benefits:

1. Smaller sets for an easier task given to the master.
2. No loss of information between the labeling sessions.
3. Decrease on the labeling time due to the smallest HCT constructed from the trimmed data.

Another issue is the optimal division of each partition set or clusters/nodes of the HCT in subsets using the clustering approach. Originally, in the HS approach it is done by applying unsupervised clustering. Note that moving downwards on the HCT, in each iteration, some of the samples of each partition set are already labeled. In the Active forest, we look for a transformation of the feature space of the current partition set, so as to minimize the distance between samples of the same class and maximize the distance between samples from different classes. In this way, we claim that sub-clusters will tend to be more "pure". To this purpose, we apply the Linear Discriminant Analysis [26, 2, 4], combined with a previously applied Principal Component Analysis [1]). By doing so, we optimize the purity of each cluster and reduce the dimensionality of the original high-dimensional feature space for faster clustering.

The flow diagram is given in Fig.2. Each of the labeling sessions is started by: a) trimming off samples classified with high confidence and confirmed by the master, and b) constructing the HCT to be labeled. Afterwards, the HS+LDA approach is applied in addition of using the Queries and the Cluster boxes for the master to choose. Finally, when all samples are labeled, the classifier is retrained to be used in the next set of labeling. The pseudo-algorithm definition is as follows:

Input: *One of the sets of unlabeled images with their corresponding features.*

Step 1: Initialize an empty tree structure T for keeping track of the pruning followed, the labeled and unlabeled samples that are in each cluster and their purity measure.

Step 2: If we have labeled more sets previously, apply initial trimming by the classifier trained on the previous HCTs.

Step 3: Choose randomly unlabeled samples and query the master.

Step 4: Save labels, set samples to "labeled" and increase the number of clicks.

Step 5: **While** *there is any unlabeled sample:*

Step 6: Get bounds (purity) of each node of the HCT and the most probable class assignment for each one (which will be temporary set as predicted until user's approval).

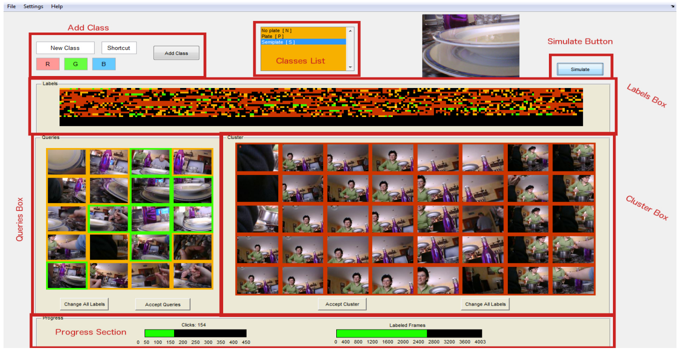

Figure 3: Main screen of our active labeling application with its different sections highlighted.

Step 7: Query N random samples and put them on the window ("Queries") with their predicted labels.

Step 8: Query the first M samples from the purest cluster and put them on the window ("Cluster") with their predicted labels.

Step 9: After user's approval save labels, set samples to "labeled" and increase the number of clicks (only from the chosen window samples).

Step 10: **If** *user selected "Queries": apply LDA and K-means in the new feature space to generate bi-partition of the clusters, where each of the queried samples belonged to.*

Step 11: **End While**

Step 12: Retrain the KNN classifier with the new labels.

Output: *Labeled images corresponding to the current HCT and the trained classifier.*

2.1 Food-Recognition Application for Active Labeling

After taking the photos with our lifelogging device along the day, images are pre-processed by: image crop by sliding window, set division for creating the trees for that day, and image features extraction. For each region, we extract the HOG descriptor and the mean (R,G,B) colors. We start the labeling process that is repeated until all the trees from that day are completely labeled. The Active forest is implemented in a user-friendly application to create the labeling set of the food-related objects. Figure 3 visualizes a scheme of its main window. On the top, the user has the possibility to add a new class and define its label. Bellow, a compacted view of all samples is presented where different colors code the labels of the samples, and black means "unlabeled". In the center, we have two groups of images: on the right, we have a "Cluster" that are the first samples of the cluster with higher purity. On the left, we have the "Queries" that are sampled randomly. Their frame shows the predicted label to be confirmed by the master. On the bottom, we have the statistics in terms of number of "clicks" (corrections) of the master and the percentage of the video that is labeled until the current step.

3. RESULTS

In order to validate the performance of the Active forest approach, we formed a validation set composed by: a public domain and a home-made database. We performed different tests to tune all the possible parameters and to compare the time needed for each labeling strategies. Given the aim of recognition of food-related objects, we illustrate the approach on plate recognition where any meal can be available. Without loss of generality, we assumed three different classes or labels for our data (Figure 4):

Figure 4: Samples from the three classes: NP (left), P (centre), SP (right).

1. No Plate (NP): The image does not show any plate or it is too far away from the camera.
2. Plate (P): There is a clear plate near the centre of the image and the subject is close to it, or a maximum 5% is out of the field of view.
3. SemiPlate (SP): A plate is only partially visible.

3.1 Data Sets

During the lifelogging acquisition of images, different tasks of the subject's everyday life were recorded. We selected mealtime records of 6 different days (business days or holidays) that led to a total of 408 images. Apart from the SenseCam images, we decided to incorporate plate images from Image-Net.org. To sum up, we used 508 images (408 from SenseCam and 100 from Image-Net) from which, using different scales and crops as a compulsory preprocessing, we obtained 89.709 images (regions) divided in 24 different sets of approximately 4000 images each (basis of the Active Forest technique).

3.2 Sensitivity & Specificity

In order to get the optimal performance of the classifiers, we computed the sensitivity and the specificity of the KNN and the AdaBoost classifiers with different parameters. To perform these tests, we used a 10-fold cross-validation always with balanced classes, on which, as a first result, and due to the low performance using a NP vs P vs SP classifier, we decided to use cascade Combined Classifier [3]. It first discriminates NP vs (P + SP) as a first step; if the sample is classified to (P+SP), a second classifier will discriminate between P vs SP.

In order to compare the results of the AdaBoost and the KNN, we tested the sensitivity and specificity of the first classification step, NP vs (P + SP) (see Figure 5 (top)). In this figure, NT is the number of tests performed, and NR is the number of rounds. Given the opposite nature of sensitivity and specificity, we give the results according to the ratio between both measures. An alternative would be to use the F-measure. Figure 5 (bottom) gives the tests on the classification of P vs SP. We could see that both classifiers give approximately the same performance but to distinguish NP vs (P + SP), KNN is slightly better. Although, AdaBoost is slightly better in discriminating P vs SP, we decided to use the KNN for both classification problems due to its more stable results and less dependence on the parameters. We decided to take a conservative value of k equal to 15 due to its better generalization capability.

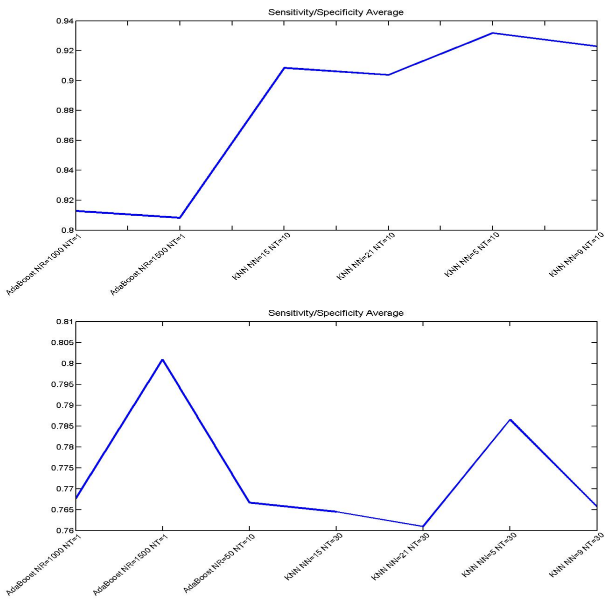

Figure 5: Sensitivity and Specificity of NP vs (P + SP) classifier (top), and Sensitivity and Specificity of P vs SP classifier (bottom)

3.3 Precision and Likelihood

The trimming step separates groups of objects with high likelihood (LH) assigning them a class, so that there is a high probability that the user will not have to correct their predicted labels. So we are interested in finding a likelihood threshold for the groups that optimizes the precision of the classifier (i.e. the groups should contain as much true positives as possible keeping only up to a small % of false positives). Figure 6 illustrates the precision of the KNN as a function of the LH threshold. Considering that a 95-97% of precision would be enough for the predicted labels to be trimmed with high probability to form a pure cluster, we fixed the corresponding LH values (NP: 0.7, P & SP: 0.7, P: 0.9, SP: 0.8) as thresholds to use in the supervised trimming procedure before constructing the next HCT. All samples that fall below these thresholds, will participate in the structure to be labeled by the hierarchical sampling method.

3.4 Active Forest Results

Finally, we present the performance results of our method. We used 89.709 labeled image regions and simulated the labeling on a MacBook Pro 2.66 GHz Intel Core i7 using Matlab running on Windows 7 on a VM. We ran 3 times each of the simulations shown in Figure 7.

Our first result is that the Active forest manages to label the set of 89.709 images, which turned out to be an impossible task for the HS method due to software and hardware limitations when creating the HCT. Moreover, each of its main contributions, results in improvement of the method: Figure 7 shows the results obtained by the Active Forest (blue), Active Forest with LDA (green) and Classifier (KNN) + Active Forest + LDA (red). We can see that, using Active forest allows us to label the images in up to 0.08 s/image, adding LDA improves it achieving 0.049s/image and adding the KNN classifier additionally reduces the time, achieving 0.03s/image, which represents an improvement of

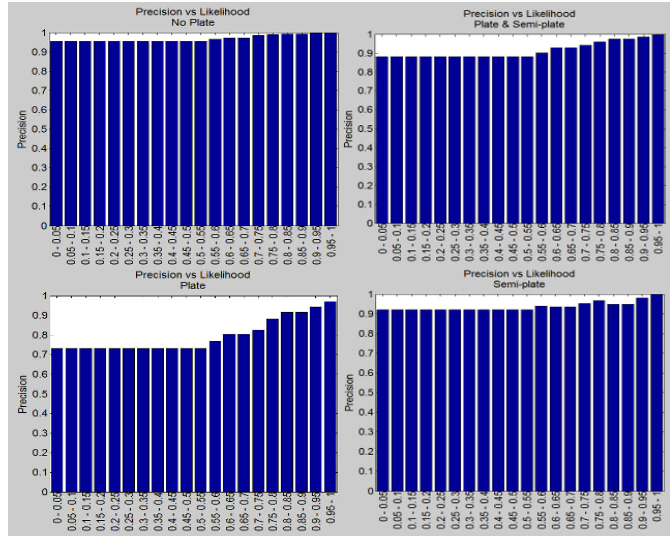

Figure 6: Precision vs Likelihood using KNN classifier for each of the labels of the Combined Classifier.

62% (all representing simulation times). We should note that in this simulation, the time needed by the masterŠs to "click" on an image is not considered, the time is spent in the algorithm execution. Interpreting the results, we can say that splitting the set and constructing several HCTs, the minimal time (that is the inferior limit), despising the master reaction time, will be about 1 hour and 42 minutes. Meanwhile, using the complete Active forest+LDA+KNN will label the set in 40 minutes (plus the time of masterŠs reaction). Since our tests shew that the number of master "clicks" was approximately the same in these cases, the times of the three algorithms would increase by a constant. When tests were done with a real master, the time increased to 0.55s per image.

Note that the theoretical bounds of HS are directly applicable to the Active forest, too. Since the purity estimation of each HCT node depends only on the number of labeled and unlabeled data, as well as the ratio of labels per class, the same estimation can be applied for the Active forest.

4. CONCLUSIONS

In this paper, we propose a novel technique (Active Forest), that allows to label large amounts of data. Our method contributes in three directions: 1) It extends hierarchical sampling method to a forest of hierarchical structures in order to make possible treating a huge volume of data; 2) It uses a trimming process applied by a supervised classifier before each labeling session has started that additionally optimizes the labeling time by 62%; and 3) Applying Linear Discriminant Analysis and a K-Means clusterisation on the nodes of the structure allows a dimensionality reduction of the feature space as well as a better clustering, minimizing the distances between samples of the same class and maximizing distances between different classes. Still, the Active forest maintains the same theoretical bounds as in the HS extracted from the purity of each node of the trees.

We integrated the Active forest method in a user-oriented application (Figure 3) that allows for an easy and fast labeling of huge amounts of data. Our application for Active

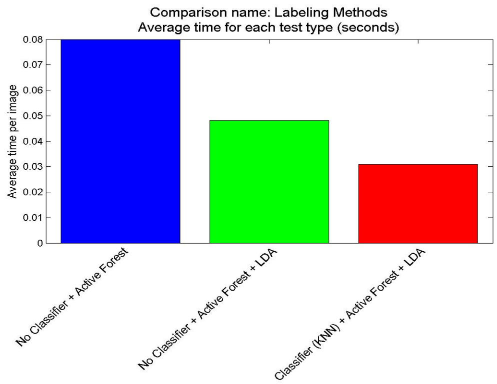

Figure 7: Labeling time of Active forest simulation.

Labeling is not limited to the binary classification and can be easily adapted for labeling multiple types of objects.

5. ACKNOWLEDGMENTS

This work was partially founded by the projects: TIN2009-14404-C02-02, TIN2012-38187-C03-01 and 2009SGR696.

6. REFERENCES

[1] H. Abdi and L. J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[2] S. Balakrishnama and A. Ganapathiraju. Linear discriminant analysis-a brief tutorial. *Institute for Signal and information Processing*, 1998.

[3] X. Baró, S. Escalera, J. Vitrià, O. Pujol, and P. Radeva. Traffic sign recognition using evolutionary adaboost detection and forest-ecoc classification. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):113–126, 2009.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *PAMI, IEEE Trans. on*, 19(7):711–720, 1997.

[5] S. Dasgupta. Two faces of active learning. *Theoretical computer science*, 412(19):1767–1781, 2011.

[6] S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th ICML*, pages 208–215. ACM, 2008.

[7] A. R. Doherty and A. F. Smeaton. Automatically segmenting lifelog data into events. In *Image Analysis for Multimedia Interactive Services, WIAMIS'08*, pages 20–23. IEEE, 2008.

[8] M. Drozdzal, S. Seguí, C. Malagelada, F. Azpiroz, J. Vitrià, and P. Radeva. Interactive labeling of wce images. In *Pattern Recognition and Image Analysis*, pages 143–150. Springer, 2011.

[9] R. O. Duda, P. E. Hart, et al. *Pattern classification and scene analysis*, volume 3. Wiley New York, 1973.

[10] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[11] P. H. Gosselin and M. Cord. Active learning methods for interactive image retrieval. *Image Processing, IEEE Transactions on*, 17(7):1200–1211, 2008.

[12] J. C. Gower and G. Ross. Minimum spanning trees and single linkage cluster analysis. *Applied statistics*, pages 54–64, 1969.

[13] J. A. Hartigan. Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association*, 76(374):388–394, 1981.

[14] H. Hoashi, T. Joutou, and K. Yanai. Image recognition of 85 food categories by feature fusion. In *Multimedia (ISM), 2010 IEEE International Symposium on*, pages 296–301. IEEE, 2010.

[15] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood. Sensecam: A retrospective memory aid. In *UbiComp 2006: Ubiquitous Computing*, pages 177–193. Springer, 2006.

[16] Y. J. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *IEEE Conference on CVPR*, pages 1346–1353. IEEE, 2012.

[17] M. Li and I. K. Sethi. Confidence-based active learning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(8):1251–1261, 2006.

[18] S. W. Lichtman, K. Pisarska, E. R. Berman, M. Pestone, H. Dowling, E. Offenbacher, H. Weisel, S. Heshka, D. E. Matthews, and S. B. Heymsfield. Discrepancy between self-reported and actual caloric intake and exercise in obese subjects. *New England Journal of Medicine*, 327(27):1893–1898, 1992.

[19] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[20] C. Morikawa, H. Sugiyama, and K. Aizawa. Food region segmentation in meal images using touch points. In *Proceedings of the ACM multimedia 2012 workshop on Multimedia for cooking and eating activities*, pages 7–12. ACM, 2012.

[21] P. Radeva, M. Drozdzal, S. Segui, L. Igual, C. Malagelada, F. Azpiroz, and J. Vitria. Active labeling: Application to wireless endoscopy analysis. In *International Conference on HPCS'2012*, pages 174–181. IEEE, 2012.

[22] T. Seidl and H.-P. Kriegel. Optimal multi-step k-nearest neighbor search. In *ACM SIGMOD Record*, volume 27, pages 154–165. ACM, 1998.

[23] A. J. Sellen and S. Whittaker. Beyond total capture: a constructive critique of lifelogging. *Communications of the ACM*, 53(5):70–77, 2010.

[24] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 2010.

[25] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2:45–66, 2002.

[26] M. Welling. Fisher linear discriminant analysis. *Department of Computer Science, University of Toronto*, 2005.