CRANFIELD UNIVERSITY

NOGARET BAPTISTE

AUTOMATED FOOD LOG ANALYSIS

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Computational and Software Techniques in Engineering

Master of Science
Academic Year: 2015–2016

Supervisor: Dr RÜGER Stefan
July 24, 2016

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Computational and Software Techniques in Engineering

Master of Science

Academic Year: 2015–2016

NOGARET BAPTISTE

Automated food log analysis

Supervisor: Dr RÜGER Stefan
July 24, 2016

This thesis is submitted in partial fulfilment of the requirements for the degree of Master of Science.

Declaration of authorship

Abstract

Type your abstract here.

Keywords

Keyword 1; keyword 2; keyword 3.

Contents

List of Figures

List of Tables

List of Abbreviations

CNN	Convolutional Neural Network
LBP	Local Binary Pattern
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
SVM	Support Vector Machine

Acknowledgements

I am really grateful to Dr. Stefan Rüger, my supervisor for the project, to have proposed this subject. His guidance and valuable advice were particularly helpful to realise the thesis.

Moreover, I would like to thank the University of Technology of Compiègne for giving me the opportunity to study one year in Cranfield University. I would also like to thank Cranfield University for its facilities.

I would like to express my gratitude to M. Kazu Shimoda and Pr. Keiji Yanai of the University of Tokyo that provided enlightenments and further details on their work and datasets.

Chapter 1

Introduction

Risk of obesity [1] (uk, world) "Overweight and obesity were significantly associated with diabetes, high blood pressure, high cholesterol, asthma, arthritis, and poor health status". Obesity is strongly associated with several major health risk factors.

Diabetes: - fast growing (current ... in to ... in) with forecast for ... to be ... - lead to high mortality - treatment cost. [2]: in 2010: 12 % of the total whorlwide health expenditure is spent on diabetes and will continue to increase.

Combination of drugs and food intake control have shown great results

Main reason: junk food: easily found, cheap.

One of the best way to fight it: watch over what we eat. Associated lifestyle changes and lose weight. Use as a prevention tool for population at risk

studies such as [3] show the benefit of reporting its daily diet to lose weight and improve the quality of its foos intake

Also a way to ... eat disorders

Currently, manually ... self reporting, using paper diaries : hard to do + cost a lot + have some problem (people tend to underestimate) + need a trained patient tedious, prone to error as the useer tend to under-estimate its intake

At the same time, improvement of the classification methods: example on Image Net results [4]: 1000 classes, more than 1,2 million images Every year since 2010 Numerous institutions (university, tech companies) are participated As described in figure ..., the mean error for each class for classification and localization has been greatly reduced between 2010 and 2014

Recently: proposition automize it. With the widespread use of smartphone, people can easily take pictures of a good quality. People are already taking picture of their food and posting them on website such as Food Gawker, Instagaram, Flickr, Yelp or

That's why, over the past few years, people ... automated it. Assist patient and their medical personnel Extends the reach of care in a cost effective ways and counters some of the previous problem (still pb with the elder / people who don't have access to smartphone)

Part of the rise of e-healthcare / m-healtchare [5, 6]

food recognition: promising applications of image processing and machine learning. Estimate food intake and people's habit

Overall process: extract characteristic (possible features are invariant of the liminosity, orientation, scale, ...)segmentate, classify, get calori value or a simplified version (using for example the ... systems), keep log and beng able to visualize it over the year

Feature description: key to achieve good object detection and image categorization

In this thesis: focus on the first two phases

Already have numerous challenges: large number of food items variation in appearance and shape different way to server it environmental condition –> lead to a high interclass variability challenging task for the human

Chapter 2

Previous work

General process

- divide the dataset in train and test Learning the parameters of a prediction function and testing it on the same data is a methodological mistake: a model that would just repeat the labels of the samples that it has just seen would have a perfect score but would fail to predict anything useful on yet-unseen data. This situation is called overfitting. To avoid it, it is common practice when performing a (supervised) machine learning experiment to hold out part of the available data as a test set. - learn and evaluate - feature description - choose of the classifier

Feature description

presenting different channel representation RGB, gray, HSV

Local binary pattern

Local binary pattern is a visual descriptor for texture composiiton of an image, first presented in .

Show an image and its LBP representation

Moments : mean and variance Hu moment (+ raw and centralized and normalized moment and their property)

Cite + write formula

Bag of words

cite who use it first?

overall process

Sift

Surf

Feature detection - can use sift or surf - dense grid (cite why it is better)

Descriptor

clustering - k means

Classifier

k-nearest neighborhood

one of the simplest

Tree, random forest

Decision tree: can be used for classification or regression It can be represented as a graph and a simple representation is given 2.1.

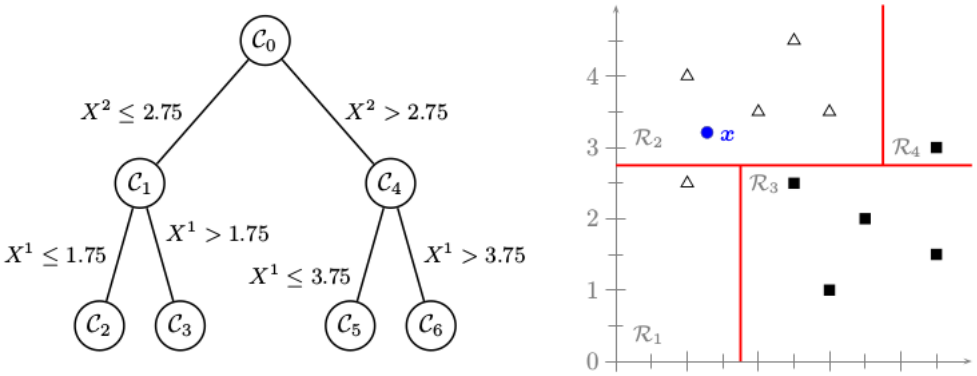

Figure 2.1: Decision tree of depth 2 for 10 elements belonging to 2 classes

Naive bayesian

SVM (binary case) + kernel trick + multi-class (one-versus-one or one-versus-all)

Kernel trick: to use the linear SVM for non-linear data: project the data in a new feature H space thanks to an application and then reserch for maximum margin hyperplan in H to make sure that the new problem has a unique solution, must satisfy the Mercer's condition or simply it must be a positiv-definit matrix give some "famous" kernel (chi2, rbf, polynomial)

cite the first to use kernel trick cite why i use χ^2

SGD classifier + loss function + regularization term

CNN

inspired by the neural system composed of different layers and communication shemes recent years: use of the adjectiv "deep" to qualify NN: many layers

Different types of layers: - convolutional (give the name of the type of NN) : The Convolution layer convolves the input image with a set of learnable filters, each producing one feature map in the output image. - max pooling - normalization layer - sigmoid - ReLu

Chapter 3

Dataset

Why do we use a dataset? - learning - some research make them freely available to test

Describe how it was build ?

3.1 Choice of the datatset

Numerous datasets are already existing and have been made freely available. I could create my own dataset but it would have been very time consuming and I wouldn't be able to compare my results with previous scientific papers. To choose, a couple of criteria were defined:

- Preferably, it should be a recent dataset
- It must have a decent number of pictures (a few thousand pictures)
- It must be composed of a general kind of food such as worldwide, Western or Asian
- It must contain pictures with multi-food items

As we can see in the table 3.1, UEC FOOD 256 is the dataset that best match our expectations.

Name	Release date	Number of pictures	Type of food	Number of classes	Multiple food items
PFID [7]	2009	4545	American fast-food	101	No
UEC FOOD 100 [8]	2012	14361	Japanese	100	Yes
FIDS 30 [9]	2013	971	Fruit	30	No
ETHZ Food-101 [10]	2014	101 000	European	100	No
FooDD [11]	2015	3000	Fruit	23	Yes
UEC FOOD 256 [12]	**2015**	**31395**	**World**	**256**	**Yes**

Table 3.1: Summary of some available food datasets according to the criteria

3.2 UEC FOOD-100 and UEC FOOD-256

UEC FOOD-100 and **UEC FOOD-256** are datasets used for food localization and recognition.

The UEC FOOD-100 dataset can be found in [1]. It was created in 2012 and presented in [8].

It contains 100 types of food, mainly Japanese food. Each kind is represented by at least 100 samples.

As presented in figure 3.1, a photo can contain more than one food items. The dataset contains files to indicate bounding boxes marking the location of a food items.

UEC FOOD-256 can be found in [2]. It was presented in [12] in 2015. It contains the 100 types of food from UEC FOOD-100 plus 156 new ones. The newly introduced food kinds are more international dishes with food from various countries such as France, Italy, the USA, China, Thailand, Vietnam, Japan and Indonesia. As for FOOD 100, every food photo has a bounding box indicating the location of the food item.

The most represented category is miso soup with 728 and rice with 620 pictures.

[1] Dataset can be found at `http://foodcam.mobi/dataset100.html`
[2] Dataset can be found at `http://foodcam.mobi/dataset256.html`

Figure 3.1: Example of pictures with multiple food items from UEC FOOD 256

Chapter 4

Implementation

4.1 Classification

4.1.1 Color histogram

For each picture:

1. extract the sub-image delimited by the bounding box
2. resize this sub-image to 224×224 pixels
3. extract the histogram of local binary pattern
4. extract the joint color histogram for the channel H and s of the HSV (hue, saturation and value) representation
5. extract the 7 hu-moment: invariant feature for translation, rotation and scale change (as stated in [13])

Normalized the data to have all features centered around zero (mean of 0) and have unit variance(variance equal to 1).

Then, apply multiple famous classifiers:

- decision tree
- random forest
- AdaBoost with decision tree
- k-nearest neighborhood
- SVM
- SGD Classifier

hyperparameter optimization: using a grid Try to optimize the accuracy for each classifier Separate the dataset in 3, 10 % for validation, 10 cross validation to select the best parameters Then 10 cross validations to train and test the classifier

Talk in result: show the best amelioration with hyperparemeter (but in general it only improve it by one or two percents)

4.1.2 Bag of words

For each picture:

1. extract the sub-image delimited by the bounding box
2. resize this sub-image to 224×224 pixels
3. detection of keypoints: use of a dense grid
4. descriptors: Root SIFT. Root SIFT is a simple variant of SIFT, presented in [14]. When the SIFT descriptors as been computed for each keypoints, we apply an element wise square root of the L1 normalized SIFT vectors

clustering: using the k-means algorithm to obtain a 2500-word codebook.

For each picture: compute the histogram of occurence counts of visual words

Kernel trick: use of a variant of the χ^2 kernel named additive χ-squared kernel presented in [15]

Then we apply the SVM classifier.

4.1.3 CNN

A pre-trained CNN used for image recognition on ImageNet Challenge 2014.

[16]

it is available [1].

The model is an improved version of the 19-layer model used by the VGG team in the ILSVRC-2014 competition.

4.2 Segmentation

A pre-trained CNN used for saliency detection.

[17]

it is available [2].

It is the same model as GoogleNet model. It is composed of 19 layers.

4.3 Code

The code is public [3].

[1] https://gist.github.com/ksimonyan/3785162f95cd2d5fee77/
[2] https://gist.github.com/jimmie33/339fd0a938ed026692267a60b44c0c58
[3] https://github.com/bnogaret/food_log

Using python 3.5.2 and its scientific stack (numpy, scipy, matplotlib) For the data structure: pandas [18] For the image processing: scikit-image [19] For most of the machine learning: package: sklearn [20] For the CNN framework: caffe framework [21] (using the pyhton layer) SIFT implementation: opencv 3.1 [22]

Documentation is generated from the python file using sphinx.

Chapter 5

Evaluation

5.0.1 Environment

All the code has been run on the "Astral" high performance computer of Cranfield's university. The operating system is SUSE Linux Enterprise Server 11 (64 bits architecture), with a Linux 3 kernel.

The system is separated in login nodes and compute nodes. There are two "front-end" login nodes and they contain two Intel E5-2660 (Sandy Bridge - 8 cores) CPUs giving 16 CPU cores and have a total of 192 GB of shared memory. The login nodes enable the user to connect to the system and compile one's program. There are 80 compute nodes, each node having two Intel E5-2660 (Sandy Bridge - 8 cores) CPUs. This is giving a total of 1280 available cores. Each compute node have at least accessed to 64 GB shared memory. Nodes are connected with Infiniband^{TM} low-latency interconnect.

5.0.2 Segmentation metrics

[1]

[1] Information on the evaluation system can be found at `http://host.robots.ox.ac.uk/pascal/VOC/voc2012/devkit_doc.pdf`

To measure the precision of the localization / segmentation algorithm, we use the metrics as defined in [23].

Detections are considered true or false positives based on the area of overlap with ground truth bounding boxes. To be considered a correct detection, the **Intersection over Union** IoU between the predicted bounding box B_p and ground truth bounding box B_{gt} must exceed 50% by the formula:

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}$$

To simplify the calculation, this formula can be rewritten as:

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p) + area(B_{gt}) - area(B_p \cap B_{gt})}$$

Using this metric, we can compute the precision P, the recall R and the accuracy A given by:

$$P = \frac{T_p}{T_p + F_p}$$

$$R = \frac{T_p}{T_p + F_n}$$

$$A = \frac{T_p}{T_p + F_n + F_p}$$

with:

- T_p the number of true positives (the bounding boxes correctly localized)
- F_p the number of false positives (the predicted bounding boxes incorrectly localized)
- F_n the number of false negative (the ground truth bounding boxes not localized)

Note that given the convention from [23], if more than one predicted bounding box overlaps the same ground truth bounding box, only one will be considered as T_P, the rest will be F_Ps.

5.0.3 Classification metrics

cross validation accuracy confusion matrix

5.1 Results

5.1.1 Food segmentation

For the three metrics: Accuracy: 0.73 % Precision: 0.74 % Recall: 0.79 %

In [24], the authors use fine-tuned pre-trained Deep Neural Network and obtain around: Accuracy: 60 % Precision: 80 % Recall: 70 %

5.1.2 Classification

For using 10 fold cross validation without parameters optimization

using LBP (98 bins) + HS (30 * 30 bins) + mean and variance of each RGB channel + Hu-moments

- random forest: 21 % (250 trees, gini)
- decision tree: 6 % (gini)
- k-nearest neighborhood: (k=10, distance metric: minkowski, weights of each neighborhood point: uniform): 10 % and 16 % with hyperparameter optimization
- SGD classifier: 12 %

- Gaussian Naive Bayesian: 4 %
- Linear SVM: 9 % (no kernel trick)
- AdaBoost with decision tree: 4 % (SAMME.R algorithm)

using a 2500-word codebook, root-sift, k-mean, RF (500 trees): 10 %

using the CNN + Random forest (500 trees): 49 %

In [24], the authors use fine-tuned pre-trained Deep Neural Network and obtain 63 % accuracyon UEC FOOD-256.

In [25], the authors use fine-tuned pre-trained Deep Neural Network and obtain 67 % accuracyon UEC FOOD-256.

5.1.3 Segmentation followed by classification

CNN Segmenter + CNN feature descriptor + RF classifier

Result: 0.27 % (0.73 % accuracy for segmentation, 0.37 % for classifier)

Accuracy: 0.73328912 Precision: 0.74412334 Recall: 0.7963661

accuracy: 0.37 precision: 0.54 recall: 0.45 f1-score: 0.41

Top 5 : french fries 0.93006986503 beef bowl 0.951754344221 hamburger 0.954545415101 rice 0.989278742795 miso soup 0.989988865517

Least 5: meatloaf 0.0 grilled eggplant 0.0 mozuku 0.0 chicken cutlet 0.0 tanmen 0.00943396137415

134 35 clear soup || miso soup 0.830188600926 124 35 zoni || miso soup 0.745613969683 156 35 oshiruko or red bean soup || miso soup 0.71717164473 88 35 Japanese tofu and vegetable chowder || miso soup 0.591549254116 135 35 yudofu || miso soup 0.572727220661 89 35 pork miso soup || miso soup 0.568345282853 82 5 cutlet curry || beef curry 0.54411760705

23 22 beef noodle || ramen noodle 0.503703666392 238 11 kaya toast || toast 0.453488319362 153 86 Caesar salad || green salad 0.444444389575

[24] : accuracy 36.84 %, 54.44 % precision, Recall 50.86 %

Chapter 6

Conclusions / Future work / Improvement / Comment

This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text.

Appendix A

Appendix

This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text. This is a sample of thesis text.

Bibliography

[1] Ali H Mokdad et al. “Prevalence of obesity, diabetes, and obesity-related health risk factors.” In: *JAMA : the journal of the American Medical Association* 289.1 (2003), pp. 76–9. ISSN: 0098-7484. DOI: 10.1001/jama.289.1.76..

[2] Ping Zhang et al. “Global healthcare expenditure on diabetes for 2010 and 2030”. In: *Diabetes Research and Clinical Practice* 87.3 (2010), pp. 293–301. ISSN: 01688227. DOI: 10.1016/j.diabres.2010.01.026. URL: http://dx.doi.org/10.1016/j.diabres.2010.01.026.

[3] Lora E. Burke, Jing Wang, and Mary Ann Sevick. “Self-Monitoring in Weight Loss: A Systematic Review of the Literature”. In: *Journal of the American Dietetic Association* 111.1 (2011), pp. 92–102. ISSN: 00028223. DOI: 10.1016/j.jada.2010.10.008. URL: http://dx.doi.org/10.1016/j.jada.2010.10.008.

[4] Olga Russakovsky et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252. ISSN: 15731405. DOI: 10.1007/s11263-015-0816-y. arXiv: 1409.0575.

[5] Richard Hillestad et al. “Can electronic medical record systems transform health care? Potential health benefits, savings, and costs.” In: *Health affairs (Project Hope)* 24.5 (2005), pp. 1103–17. ISSN: 0278-2715. DOI: 10.1377/hlthaff.24.5.1103. URL: http://www.ncbi.nlm.nih.gov/pubmed/16162551.

[6] Nir Menachemi and Taleah H. Collum. “Benefits and drawbacks of electronic health record systems”. In: *Risk Management and Healthcare Policy* 4 (2011), pp. 47–55. ISSN: 11791594. DOI: 10.2147/RMHP.S12985. arXiv: 0710.4428v1.

[7] Mei Chen et al. “PFID: Pittsburgh Fast-food Image Dataset”. In: *Proceedings - International Conference on Image Processing, ICIP* (2009), pp. 289–292. ISSN: 15224880. DOI: 10.1109/ICIP.2009.5413511.

[8] Yuji Matsuda, Hajime Hoashi, and Keiji Yanai. “Recognition of multiple-food images by detecting candidate regions”. In: *Proceedings - IEEE International Conference on Multimedia and Expo*. IEEE, July 2012, pp. 25–30. ISBN: 978-1-4673-1659-0. DOI: 10.1109/ICME.2012.157. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6298369.

[9] Škrjanec Marko. “Automatic fruit recognition using computer vision”. Mentor: Matej Kristan. Bsc thesis. Faculty of Computer and Information Science, University of Ljubljana, 2013.

[10] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. “Food-101 - Mining discriminative components with random forests”. In: *Lecture Notes in Computer Science*. Vol. 8694 LNCS. PART 6. 2014, pp. 446–461. ISBN: 9783319105987. DOI: 10.1007/978-3-319-10599-4_29. arXiv: 978-3-319-10599-4{_}29 [10.1007]. URL: http://link.springer.com/chapter/10.1007/978-3-319-10599-4%7B%5C_%7D29.

[11] Parisa Pouladzadeh Abdulsalam Yassine and Shervin Shirmohammadi. “FooDD: Food Detection Dataset for Calorie Measurement Using Food Images”. In: *New Trends in Image Analysis and Processing – ICIAP 2015 Workshops* 9281 (2015), pp. 441–448. ISSN: 16113349. DOI: 10.1007/978-3-319-23222-5. URL:

http://link.springer.com/chapter/10.1007/978-3-319-23222-5%7B%5C_%7D54.

[12] Yoshiyuki Kawano and Keiji Yanai. "Automatic expansion of a food image dataset leveraging existing categories with domain adaptation". In: *Lecture Notes in Computer Science* 8927 (2015), pp. 3–17. ISSN: 16113349. DOI: 10.1007/978-3-319-16199-0_1.

[13] Ming-Kuei Hu. "Visual pattern recognition by moment invariants". In: *IRE Transactions on Information Theory* 8 (1962), pp. 179–187. ISSN: 0096-1000. DOI: 10.1109/TIT.1962.1057692.

[14] Relja Arandjelovic and Andrew Zisserman. "Three things everyone should know to improve object retrieval c". In: *IEEE Conference on computer vision and Pattern Recognition* April (2012), pp. 2911–2918. ISSN: 9781467312288. DOI: 10.1109/CVPR.2012.6248018.

[15] A Vedaldi and A Zisserman. "Efficient Additive Kernels via Explicit Feature Maps". In: *{IEEE} Int. Conf. on Computer Vision and Pattern Recognition* XX.Xx (2010), pp. 3539–3546.

[16] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *ImageNet Challenge* (2014), pp. 1–10. ISSN: 09505849. DOI: 10.1016/j.infsof.2008.09.005. arXiv: 1409.1556. URL: http://arxiv.org/abs/1409.1556.

[17] Jianming Zhang et al. "Unconstrained Salient Object Detection via Proposal Subset Optimization". In: *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)* (2016). URL: http://cs-people.bu.edu/jmzhang/sod.html.

[18] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference* (2010), pp. 51–56. URL: http://conference.scipy.org/proceedings/scipy2010/mckinney.html.

[19] Stéfan van der Walt et al. “Scikit-image: image processing in Python”. In: *PeerJ* 2 (2014), e453. ISSN: 2167-8359. DOI: 10.7717/peerj.453. arXiv: 1407.6245. URL: https://peerj.com/articles/453.

[20] Fabian Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: ... *of Machine Learning* ... 12 (2012), pp. 2825–2830. ISSN: 15324435. DOI: 10.1007/s13398-014-0173-7.2. arXiv: 1201.0490. URL: http://scikit-learn.org/stable/.

[21] Yangqing Jia et al. “Caffe: Convolutional Architecture for Fast Feature Embedding”. In: *Proceedings of the ACM International Conference on Multimedia* (2014), pp. 675–678. ISSN: 10636919. DOI: 10.1145/2647868.2654889. arXiv: 1408.5093. URL: http://arxiv.org/abs/1408.5093.

[22] G Bradski. “The OpenCV Library”. In: *Dr Dobbs Journal of Software Tools* 25 (2000), pp. 120–125. ISSN: 1044-789X. DOI: 10.1111/0023-8333.50.s1.10. URL: http://opencv.org/.

[23] M. Everingham et al. *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*. URL: http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[24] Marc Bolaños and Petia Radeva. “Simultaneous Food Localization and Recognition”. In: (2016), pp. 2–7. arXiv: 1604.07953. URL: http://arxiv.org/abs/1604.07953.

[25] Keiji Yanai and Yoshiyuki Kawano. “Food image recognition using deep convolutional network with pre-training and fine-tuning”. In: *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, June 2015, pp. 1–6. ISBN: 978-1-4799-7079-7. DOI: 10.1109/ICMEW.2015.7169816. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7169816.