Contents

Chapter 1

Week 1

1.1 Food Balance Estimation by Using Personal Dietary Tendencies in a Multimedia Food Log

In this paper [1], the authors use a Bayesian Framework to improve a deterministic method (use the same feature extraction with a SVM classifier) on the FoodLog dataset. The BF takes into account the likelihood, the prior distribution and the mealtime category (breakfast, lunch and dinner). The prior distribution is updated each time the user adds a new photo.

As stated above, the dataset used is FoodLog.

For the methods, they use:

1. Feature extraction:
   (a) Color features (RDB mean and variance / HSV mean and variance ...)
   (b) Circle features (Hough transform to detect it)
   (c) Bag-Of-Feature using SIFT (Scale-invariant feature transform)
2. Bayesian framework: based on the Gaussian Naive Bayesian (suppose independence between every pair of features + the likelihood of the features is assumed to be normally distributed)

Comment:

It supposes to have some extra information (mealtime) and to have incremental data. It can't be used on static dataset such as UEC food 100.

1.2 A virtualization mechanism for real-time multimedia-assisted mobile food recognition application in cloud computing

In this article [2], the authors present their mobile application for the Android platform: run on the smartphone with communication with an emulated Android phone on the Cloud. Most of the work is done on the cloud, synchronization between the device used the VNC (virtual network computing) protocol.

They use their own dataset composed of:

1. Positive image (labelled image of food): the data set comprises of 40 different categories of food and fruits
2. Negative image (do not contain the relevant object)

The used methods are just cited, not even presented:

1. Graph cut segmentation
2. Deep convolutional neural networks

Comment: Mainly technical aspect of the mobile application. No presentation of the segmentation / machine learning.

For another technical presentation: see this.

1.3 Image Recognition of 85 Food Categories by Feature Fusion

In [3], authors developed a system for image recognition of 85 kinds of food (such as omelet, hamburger ...). They use several image feature extractors (described in methods) with the MKL algorithm as a feature fusion. They use SVM to learn. Then, they compare the different features.

Dataset: 85 food categories with 100 "relevant" images per category. I could not find it.

Used methods:

1. Image features:
   (a) Bag of feature: three methods of point-sampling (DoG (DIfference of gaussian), random and grid) using SIFT descriptor. Codebook of 1000 or 2000 words.
   (b) Color histogram
   (c) Gabor texture feature
   (d) Gradient histogram
2. Multiple kernel learning (using the Shogun toolbox written in C++) to fusion the features
3. Classification : SVM with the χ^2 kernel (justification: "is commonly used in object recognition tasks") and one-vs-all classification strategy.
4. Evaluation:
   (a) 5-fold cross validation
   (b) used the classification rate (= average value of diagonal elements of the confusion matrix)
   (c) used the recall rate [= (the number of correctly classified images)/(the number of all the image in the category)]

Overall accuracy: 62 %

Chapter 2

Week 2

2.1 Food-101 - Mining discriminative components with random forests

In [4], the author use the Random forest clustering algorithm to create superpixels (selecting only the discriminative one). On these superpixels, a dense SURF and L*a*b* color value is computed and encoded with improved fisher vectors (IFV) with Gaussian mixture model (GMM) of 64 Gaussians. Then, they use PCA to reduce the size of the vector and the machine learning method is structured-output multi-class SVM. They use their method on Food-101 (50.76% accuracy) and MIT-indoor (58% of accuracy on the full dataset) and compare it against several previous implementations.

Method:

1. Feature: SURF and L*a*b* color
2. Coding: fisher vector with GMM of 64 modes (reduced size with PCA)
3. Machine learning: multi-class SVM

2.2 Auto-Recognition of Food Images Using SPIN Feature for Food-Log System

In this paper [5], the authors mainly describe its new feature. They use the Hough transformation to detect circle (they constitutes the food region). Then, they use their new rotation invariant feature named SPIN: Segment the circle in multiple rings (16 rings for their experiment) Extract the HSV color for each ring (experiment: 40 bins for channel H (hue), 10 bins for channel S (saturation) and V (value)) Concatenate the results Then, use a multi-class SVM for food recognition on their internal dataset. They obtain an average accuracy of 62% per category.

Dataset:

Internal. 10 categories, 30 images per category

Method:

For learning, they use a multi-class SVM.

Comment:

It makes the assumption that the food is inside a round container (plate, glasses ...). It uses a limited dataset that I don't have access to.

2.3 Food Image Recognition with Deep Convolutional Features

In [6], the authors state first that using DCNN alone (letting the algorithm find what to feature is optimized) is not applicable as it is a too small dataset (less than 100 pictures per category for the experimental dataset). That's why they

are using a pre-trained DCNN (that can be found here) and reuse its two last layers as features. Moreover, they add conventional image features: HOG and color patches encoding in Fisher vector. Coupling the 3 features, they obtain 72% of accuracy for UEC-FOOD 100.

Dataset: UEC-FOOD 100

Methods:

1. Feature:
   (a) HOG (more exactly, RootHog that is "an element-wise square root of the L1 normalized HO") (8 orientations per block of 2 * 2)
   (b) color patches (mean and variance values of RGB value of pixels from each of 2*2 block)
   (c) Deep convolutional neural network last two layers
2. Encoding: Fisher vector
3. Machine learning: either linear SVM or Adaptive Regularization of Weights (implementation)

Chapter 3

Week 3

3.1 Combining global and local features for food identification in dietary assessment

In [7], the authors use local and global features to identify and quantify the food consumed. For the global features, they use:

- color properties: color, entropy and predominant color statistics (selecting the 4 dominants colors) of the whole picture.
- texture information: Gabor filter

For the local features, they use the **Bag of features** approach:

- point of interest detection with Difference of Gaussian method
- local descriptors around a $M \times M$ neighborhood of a point of interest (*local color, local entropy color, Tamura perceptual features, Gabor filters, SIFT descriptor, Haar wavelets based on the SURF descriptor, Steerable filters, and DAISY descriptor* in this paper).
- vocabulary construction: using the K-means clustering to obtain a codebook of N words
- supervised selection

To classify, they use SVM (using the Radial Basis function kernel).
They build 3 datasets composed of hand segmented images (one picture can contain several categories)

1. 19 food items and 63 images in total. 98 % average accuracy for all the features combined (half of the data for training and half for testing)
2. 28 different foods and 116 images. 97 % accuracy
3. combined 1 and 2. 86 % accuracy

Comments:
I find interesting to combine local and global feature. Many features are used without any description.

3.2 Food Detection and Recognition Using Convolutional Neural Network

In [8], the authors detail the use of convolutional neural network for food recognition. CNN is a multilayer neural network composed of convolution and pooling layers. It has some hyper parameters (number of middle layers, size of the convolution kernel and the active function) that are tunned for their experiments.

They build their own dataset for food recognition. It is based on data from FoodLog. They choose the 10 most frequent food items and select all the available images 2 months after the release of the website.

They compare their results with three existing techniques (using 6-fold cross validation to measure the average accuracy):

- spatial pyramid matching (SPM) using a color histogram and SVM. Average accuracy: 54%
- GIST feature and SVM. Average accuracy: 52%
- Bag of Words (SIFT detector) and SVM. Average accuracy: 60%

The best parametrized CNN reaches a 73% average accuracy.

3.3 Classifying food images represented as Bag of Textons

In [9], they use texture feature for food recognition.

They rely on the bag of visual words model (BoW) that is composed of 4 main steps:

- feature detection
- feature description (SIFT, Textons): for each image, we create a small list of Textons that are collected to form a global dictionary
- codebook generation (using K-means clustering to obtain a visual vocabulary of size K)
- image representation: study the distribution of Textons among the images. Each image is represented as a visual word distribution.

Then, the image representation is used to feed a SVM classifier.

The test image are represented by the pre-learned Textons vocabulary and classified by the pre-trained SVM methods.

Images are pre-processed with the Maximum Response Filter Banks (MR) (composed by Gaussian, first and second derivative of Gaussian and Laplacian of Gaussian filters) to obtain an orientation and scale invariance. The intensity of each image is normalized (zero mean and unit standard deviation on each color channel) to achieve global invariance of the illumination intensity.

The described technique is used on Pittsburgh Fast-Food Image Dataset (61 categories, 1098 images). They also regroup the 61 categories in 7 super-categories.

The bag of Textons has 31.3% average accuracy (using a 3-fold cross-validation with $2/3$ for training and $1/3$ for testing) for the 61 categories and 79% for the 7 major classes.

Chapter 4

Week 4

4.1 A food image recognition system with Multiple Kernel Learning

In [10], a food image recognition system with Multiple Kernel Learning (MKL) is built. The MKL method is used to integrate several image features (color, texture SIFT) and estimate the optimal weights of each features. The learning tool is a SVM classifier. They obtain an accuracy of 61.34 % on their dataset.

They use phones to take photos to recognize food. They took 166 food photos (users were not instructed how to "properly" take a photo) and obtained a 37.55% accuracy.

- Image features:
  - Bag of words:
    1. A set of local image points is sampled by an interest point detector (example: Difference of Gaussian), randomly or grid-based and visual descriptors are extracted on each point (example of descriptors: SIFT)
    2. Based on the distribution of SIFT descriptors on all the images, a codeword is generated, using the K-means clustering methods. Only a certain number of visual words is kept (example: codebook size of 1000 or 2000 visual words)
    3. The resulting distribution of visual descriptors is used to characterize the image.
  - Color histogram: divide the image into 2×2 blocks and extract a 64-bin RGB color histogram from each block with dividing the space into $4 \times 4 \times 4$ bins. Thus, it extracts a 256-dim color feature vector from each image.
  - Gabor texture feature: divide the image into 3×3 or 4×4 blocks. On each block, 24 Gabor filters with 4 kinds of scale and 6 kinds of orientation is applied (the filter response is average within the block) to obtain a 24-dim vectors per block. Thus, it extracts a 216-dim or 384-dim vector from each image.
- Classification: using MKL (each image feature is assigned to one kernel) - SVM (one vs all strategy) with the χ^2 kernel.
- Evaluation: 5-fold cross validation, classification rate (average value of diagonal elements of the confusion matrix) + recall rate ((the number of correctly classified images) / (the number of all the image in the category)).

Dataset: internal, composed of 50 categories, 100 images per category (picked from Internet)

4.2 Food Recognition Using Statistics of Pairwise Local Features

In [11], the authors use a novel feature, named PFD (pairwise local feature distribution) for food recognition and SVM. Then, they apply it on the Pittsburgh fast-food image dataset (PFID) dataset.

The different steps of this method are:

1. classify each pixel in one of the categories (common categories for fast food):

   - beef
   - chicken
   - pork
   - bread
   - vegetable
   - tomato/tomato sauce
   - cheese/butter
   - egg/other
   - background

   It's a soft labeling as the likelihood that a pixel belongs to each categories is kept. For classification, they use the Semantic Texton Forest, method based on local characteristics. It was previously trained on 16 manually-labeled pictures.

2. Global ingredient representation (GIR): for the 8 food categories, it sum up the soft label of all the ingredient pixel and normalize by the number Problem: no spatial relationship between ingredients

3. PFD: geometric pairwise feature on N ingredient pixels (picked randomly, thus N / 2 pairs):

   - log of the distance
   - orientation
   - soft label of the midpoint
   - soft label of each pixel along the line connecting the pair of pixels
   - joint feature (a mixed of the above characteristics)

   Accumulate the pairwise values into a distribution (using a multi-dimensional histogram of either 8 or 12 bins), weighted by the soft labels of the two pixels. Each pixel is mapped to its closest bin in the histogram. Then, normalization of the histogram.

For the PFID dataset, they obtain an accuracy between 19% and 28% for each of the 61 categories. When they pick the 7 major types of food, they get almost 80% of accuracy.

4.3 On the combination of local texture and global structure for food classification

In [12], the authors use a local texture feature and their spatial distribution to classify food images from the Pittsburgh fast-food image dataset (PFID).

The author use the Bag Of Words method. They use the SIFT detector to find points of interest in every image. For each point, they compute the Local Binary Pattern (LBP). It is a texture feature: for each pixel, thresholding the neighborhood of each pixel with the value of the center pixel and considers the result as a binary number (0 if the value is smaller, 1 otherwise). Only the most discriminative words are kept. The K-means algorithm is used to cluster to the nearest visual words.

The shape context algorithm is used to keep the spatial relationship between codewords (for each image, compute the histogram of one word compared to the others / then mean of the histograms).

For the classification, the authors pick the smallest cost between an image and a food category. For each interest points found with SIFT in the image, we associate a similarity between the point and each visual words of the codebook. The similarity function is based on the Bhattacharyya distance. Then, the shape context between the point of interests and the visual word is calculated and a cost is deduced for each food category. The category with the smallest cost is chosen.

Chapter 5

Datasets

I have found a couple of datasets:

- ETHZ Food-101 [1] : presented in [4], it is composed of 101 categories, 1000 images per category (250 pictures manually reviewed, used for the test set and 750 with noises for the training test). Pictures were extracted from the website foodspotting.com. The top 101 most popular dishes from this social sharing food images defined the categories.

  The authors obtain 56 % of accuracy with random forests.

- UMPC Food-101 [2]: presented in [13], it is composed of the same 101 categories as ETHZ food-101 dataset, with 1000 images per category. Yet, the pictures have been crawled from Google image, researching for recipes. Thus, images are associated with a text.

  The authors obtain 85.10 % of accuracy, using both textual (with Tf-Idf) and visual (using CNN pre-trained on huge datasets) features.

- UEC FOOD 100 [3] : created for [14], it conatins 100 types of food, mainly Japanese food. Each food picture has a bounding box indicating the location of food items. The authors get 55.8 % accuracy.

  In [6], the authors obtain 72.26% accuracy by using DCNN (Deep Convolutional Neural Network) features trained by the ILSVRC2010 dataset.

- UEC FOOD 256 [4] : presented in [15], it has 256 kinds of food. As for FOOD 100, every food photo has a bounding box indicating the location of the food item.

- Chinese food [5] : in [16], the authors build a dataset composed of 50 Chinese foods with 100 picture each category.

  In this paper, the authors obtain an average accuracy of 68.3 %.

- FIDS 30 [6] (Fruit Image Data set): this dataset [17] is composed of 30 fruit categories, each including at least 32 different images. A picture can contain one to dozens of the same fruit.

- FooDD [7] (Food detection dataset): in [18], the authors constructs a dataset of 3000 images of 23 food categories. They highlight the use of different cameras and conditions (lighting, shooting angle, white plate, thumb) to take the pictures. An image can include one or several food item.

[1] Dataset can be found at `https://www.vision.ee.ethz.ch/datasets_extra/food-101/`
[2] Dataset can be found at `http://visiir.lip6.fr/`
[3] Dataset can be found at `http://foodcam.mobi/dataset100.html`
[4] Dataset can be found at `http://foodcam.mobi/dataset256.html`
[5] Dataset can be found at `http://www.cmlab.csie.ntu.edu.tw/project/food/`
[6] Dataset can be found at `http://www.vicos.si/Downloads/FIDS30`
[7] Dataset can be found at `http://www.site.uottawa.ca/~shervin/food/`

- PFID (stands for Pittsburgh fast-food image dataset) [8]: presented in [19] in summer 2008 from the collaboration of Intel Labs Pitssburgh, Columbia and Carnegie Mellon universitie. It is one of the first mature datasets released for food recognition.

  It contains 101 meals (categories) from 11 popular fast food chains with images and videos captured in both restaurant conditions and controlled lab setting. It contains foods such as chickens, sandwiches, salads, burgers and drinks from :

  - Arby's
  - Bruggers Bagels
  - Dunkin Donuts
  - KFC
  - McDonalds
  - Panera
  - Pizza hut
  - Quiznos
  - Subway
  - Taco Bell
  - Wendy's

  The authors provide two food classification baseline methods:

  - Colour histogram + SVM classifier. They obtain a mean accuracy of around 12%.
  - Bag of SIFT features + SVM classifier. They obtain a mean accuracy of around 25 %.

  In [12], the author use a bag of LBP descriptor with the shape context method.

  Moreover, Fast foods, as they are standardized and have nutrition information available online, can easily be used to measure the calories. In [20], the authors are using the PFID's videos to estimate energy intake of a meal.

- Supermarket produce dataset[9]: in [21], the authors describe their new contribution named supermarket produce dataset. This dataset is composed of 2635 images divided into 11 fruit categories from a local shop: plum (264), agata potato (113), cashew (210), kiwi (171), fuji apple (212), granny-smith apple (155), watermelon (192), honeydew melon (145), nectarine (247), williams pear (159), and diamon peach (211). Pictures were taken with different illumination conditions and can contains several items for a same category.

[8] Dataset can be found at `http://pfid.rit.albany.edu/`

[9] **Dataset can be found at \url {http://www.ic.unicamp.br/~rocha/pub/communications.html}.**

Chapter 6

Week 5

6.1 A unified image segmentation approach with application to bread recognition

In [22] and [23], the authors create an application to automatically identify hand-made breads for a cash register system to display their price.

The classification is based on the shape, size, texture and color ditribution of the images.

The authors build a dataset of 73 kinds of breads at three different orientations.

The steps are:

1. preprecessing: color, background, hightlight and γ correction, color balancing, transformation from RGB to HSI
2. thresholding (binarization of the image)
3. size and shape feature:
   - area
   - elongation
   - MBR (minimum bounding rectangle) ratio

   At this step, the authors are able to group breads in circular, square, rectangular and elliptical group.
4. texture analysis
5. color analysis: histogram of the hue component (with 14 bins) for the entire image and for bread parts. Matrices of the bin areas are computed and normalized (substract the mean and divide by the standard deviation).

6.2 Food log by analyzing food images

In [24], the authors create a food log system that can recognize food from image and analyze the food balance to provide useful advice / graphs of the log.

The food balance is based on the recommendation of [25] (it was replaced by MyPlate in 2011 [26]). This dietary model is composed of 5 kinds of food: grains, vegetable, meals and beans, milk and fruit. For each group, a recommended intake per day is associated. Quantity is categorized by "serving", making it simpler to compute and keep log.

For the food detection, a shape recognition of circles and rectangles is used to detect food area. The user can correct these detections. The features are:

- average and standard deviation values of RGB and HSV for the whole image
- each image is divided in 300 block to compute the color histogram and DCT coefficients for each block.

Every block is associated to one of the "MyPyramid" food group or "non-food" group using SVM.

Finaly, a histogram of the 5 ingredients is formed to determine the food balance (using SVM).

To check the validity of their model, the authors have two dataset:

- for the food detection: 600 images, half from FLickr websites and half from authors' own capture. Half of these images are food pictures. They obtain 88% accuracy.
- for the food balance: 100 food images where each food category is represented. The accuracy is 73%.

6.3 Recognition of multiple-food images by detecting candidate regions

In [14], the authors propose a food recognition system to identify food items of a picture. The first step is to detect potential region with multiple object detection algorithms. Then, for these regions, several feeatures are extracted and used to feed SVM with multiple kernel learning method.

To detect candidate regions, the authors use:

- Felzenszwalb's deformable part model (DPM): based on Histogram of Oriented Gradients (HOG). Multiple HOGs are
- circle detector: convert the image to a gray-scale image, extracts contour by the Canny Edge Detector and detect circle by the Hough Transform
- JSEG region segmentation: segmente region based on color. Only keep circular regions.
- whole image: for image with one large dish

Then, it agregates all the candidate region to get bounding box of each food item.

For each region, the image features used are:

- Bag of Feature of SIFT and CSIFT (sift with color invariant charateristics)
- Spatial pyramid representation: object regions are divided by hierarchical grids. In this paper, the three level pyramid is used: 1×1, 2×2, 3×3. For each grid, a BoF vector is extracted
- Histogram of Oriented Gradient (HOG)
- Gabor texture

After extraction of the feature vectors from each candidate region, Support Vector Machine trained by Multiple Kernel learning is used (χ^2 kernel).

For evaluation, the authors build a new dataset composed of 100 categories with their associated bouding boxes for a total of 9060 images. For multiple food item images, they obtain 55.8 % classification rate and 68.9 % for signle food item pictures.

Chapter 7

Week 6

7.1 Fast food recognition from videos of eating for calorie estimation

In [20], the authors are using the videos of people eating provided by PFID to estimate calory intake.

For the features, it is using SIFT detectors and descriptors with some geometric criterion to determine the foods (can have multiple items per video). Then, thanks to the available nutrition information provided by the Fasst Food restaurant, they compute the calorie intake.

It does not take into account the food portions, nor the remaining quantity.

7.2 Fruit Recognition using Color and Texture Features

In [27], the authors develop a method to recognize food items, using the supermarket produce dataset. Their novel approach is to use color and texture featues.

They use the HSV (hue, saturation and value) representation and compute a few key features to summarize the color and texture of the picture. The V component is used to extract texture features. It is decomposed using Discrete Wavelet Transform and a co-occurence matrix is constructed from the approximation sub-band by estimating the pair wise statistics of pixel intensity. From this matrix, contrast, energy, local homogeneity, cluster shade and cluster are computed. From the H and S components, they extract 4 statistical featues: mean, standard deviation, skewness (third standardized moment) and Kurtosis (fourth standardized moment).

Thus, they obtain 13 features (8 for the color, 5 for the texture).

The minimum distance classifier is used to recognize the food.

They are using the supermarket produce dataset. It contains 2635 images for 15 different fruits (Plum, Agata Potato, Asterix Potato, Cashew, Onion , Orange , Taiti Lime , Kiwi , Fuji Apple, Granny-Smith Apple , Watermelon, Honeydew Melon, Nectarine, Williams Pear and Diamond Peach).

7.3 Multilevel segmentation for food classification in dietary assessment

In [28], the authors are classifying multiple food items from an image. They are first segmented the image and then identifying non-background part.

The process is:

1. segmentation: predict the class of each pixel (background pixel are ignored in the next step)
2. feature extraction: see [7]

3. classification: they are using SVM with the RBF (radial basis function) kernel.

They are using an in-house database, build from nutritional studies. It has 200 images, each picture containing 6 - 7 food items. There are 32 food categories.

The average classification accuracy for the food classes is 44 %.

Chapter 8

Week 7

8.1 Recipe recognition with large multimodal food dataset

In [13], the authors created a new dataset named UMPC Food-101 ("twin dataset" of ETHZ Food 101) combining text and visual information for recipes. As a proof of concept, they develop a search application for recipe recognition. The user send a query (a food image) and as a result, the three best recipes (categories) are displayed.

For the image recognition model, they use textual, visual or mix of both features:

- visual feature:
  - Bag-of-Words: spatial pyramid of 3 levels dense SIFT (window size: 4, step size: 8), 1024 words, soft coding (using probability to be this word). They obtain an average accuracy of 23.96 %.
  - Use an improved version of the Bag-ofWords: BossaNova only modify the pooling system. Instead of keeping the closest cluster of a SIFT descriptor, it represents it by keep distances between the descriptor and the words in the codebook. Average accuracy of 28.59 %.
  - Use deep CNN features: 7th layer of a pre-trained CNN ("OverFeat"). Average accuracy of 33.91 %.
  - Use vvery deep CNN features: 19th weight layers ("vgg-verydeep-19" from MatConvNet) Average accuracy of 40.21 %.
- text-feature: tf-idf and get 82.06% accuracy
- fusion of textual and visual feature (very deep CNN features): get 85.10 % accuracy

As a classifier, a linear SVM is used.

They develop their own dataset that is freely available (UPC Food-101, see section dataset for more information): 101 categories, 1000 images per categorie. Each picture has an associated text (the recipe). Each picture have been extracted from Google image search engine with the same 101 labels as ETHZ Food 101 dataset [4].

8.2 A novel method for measuring nutrition intake based on food image

[29] presents a novel food recognition system that is able to estimate of the nutrition intake. Moreover, they develop a mobile application to easily take pictures and keep track of the user's diet. To measure the food intake, authors compare before and after eating pictures and use the thumb as the calibration system (it supposes a one-time calibration to know the size of the thumb of the user).

The process to show the intake is:

1. user take food pictures
2. get the contour of each picture
3. recognition of the food using color, shape and size features with SVM.
4. volume calculation, that is computed in two steps:
   (a) user takes a picture from above. Then, the food shape is divided into known shape (rectangle, circle, triangle ...) to compute the area.
   (b) user takes a second picture from the side. This is used to compute the height of the food and calculate the overall volume.

   The system assumes that the plate is white and round.
5. using a nutrition databe to get the result

If the user hasn't eaten everything, the entire process is repeated.

For performance evaluation, they have their in-house dataset composed of 100 pictures that they describe as "simple". They get 89 % accuracy for the food recognition part. Yet, the authors don't state how many food categories have been used. On this dataset, the average error for the food intake estimation is equal to 4.22 %.

The drawbacks of this method is the user have to take several pictures, with one's thumb each time and it has been tested with a limited set of simple food types.

8.3 Automatic Chinese food identification and quantity estimation

[16] present a method to automatically identify food and estimate the quantity. They develop a mobile application for food classification. Measuring the quantity of food is only available for non-transparent food (not for cooked rice or water) and it assumes the use of a depth camera to get depth information.

Features use:

- Bag-Of-Word, using SIFT as descriptor
- Local binary pattern: LBP histograms of 59 bins on a 3-level pyramid and construct a codebook of 2048 words sparse coding: sift and local binary pattern for texture description
- color histograms: divide the image into 4×4 blocks and a 96-bin RGB color histogram is computed for each block

- gabor texture: divide the image into 4×4 blocks and each block is convolved with Gabor filters (keep the mean, variance of the Gabor magnitude).

They train a SVM classifier for each category, then fuse them with the multi-class AdaBoost algorithm.

For depth estimation, the area of the food container (bowl, plate) and the depth value of the contained food is computed to obtain the food volume. This technique is still limited as it can't detect some food item such as water or cooked rice and force the user to have a depth camera (such as Kinect).

A dataset has been created and release for experimentation. It only covers the food recognition part. It has 50 categories (mainly Chinese food), 100 images per category. Authors get an overall accuracy of 68.3 %. If we keep the top-3 results, the accuracy is even 90.9 %.

Chapter 9

Week 8

9.1 CNN-based food image segmentation without pixel-wise annotation

In [30], the authors presents their segmentation process for the UEC-FOOD100 dataset.

The proposed pipeline is composed of 6 main steps:

1. detect all the possible bounding box (maximum 2000 per image) using selective search
2. cluster the bounding box, using the ration of intersection over union (IOU, also call overlap ratio) to obtain 20 at most.
3. Deep CNN for all the selected bounding box to get a saliency map (using AlexNet CNN, pre-trained on the Salient Object Subitizing (SOS) dataset, fine-tuned with the UEC-FOOD100 one)
4. use the GrabCut algorithm to extract the foreground region from the food area.
5. In case of overlapped bounding box, the authors proposed to apply the non-maximum suppresion (NMS) algorithm.

The authors apply this process on the UEC-FOOD100 dataset and PASCAL VOC 2007 (use for detecting bounding box around 20 common classes (train, tv, cat, human ...)). A segmentation is correct if theoverlap ratio exceeds 50% between the detected bounding box and the ground truth bounding box.

For the first one, they obtain 49.9 % mean average precion. For the second one, they obtain 58.7 % precision.

9.2 FooDD: Food Detection Dataset for Calorie Measurement Using Food Images

In [18], the authors present their food recognition system on their new dataset called FooDD. They develop a smartphone application to take new photos and send it to the server. On the server-side, these pictures are pre-processed (mainly resized the picture to the standard size 970×720 pixels), segmented, classified and the calories are estimated.

FooDD is a food detection dataset composed of 3000 images of 23 categories. To respect the dataset condition, the user has to take picture with one's thumb (calibration to estimate the colume of food) and a white round plate. The plate can include several food items. The pictures are taken from a mix of conditions (3 different cameras, lighting, shooting angle).

They use several methods for the segmentation:

- color-texture

- graph-cut and color-texture segmentation
- deep neural network

They obtain an astonishing 100 % accuracy for the DNN, 95 % for the GC and color-texture and 92 % for the color-texture only (combined with SVM).

9.3 Food Recognition for Dietary Assessment Using Deep Convolutional Neural Networks Stergios

In [31] is presented a Deep Convolutional neural network based approach to classify pictures containing one / several food items.

They apply their method on a dataset: composed of a 246 images with 573 food items divided into 7 food categories. For each picture, they extract patches from the inside of each food item (size 32×32 or 16×16). As there is a limited number of picture, the amount of patches is artificially increased by applying rotation, flip or both transformation on each patch.

The convolutional neural network is only used to classify food patches using the Caffe framework [**jia2014caffe**].

Several configuration were tested to select the one giving the best average F-score for the patches the best one is composed of 6 layers:

1. 32-kernel convolutional (5×5 kernel) followed by a 3×3 pooling region
2. same
3. same
4. 64-kernel convolutional-pooling
5. 128 fully-connected
6. 7 fully-connected

The activation function used for each convolutional layer is the rectified linear unit (ReLU). For the last two layers, the dropout method is applied to avoid the CNN to overfit (some node are randomly ignored (dropout probability of 0.5)).

Then, to classify the overall item, they compare multiple voting scheme. The most accurate one use a max-voting method with patches of 16×16 pixels.

Using a 5-fold cross validation, they obtain 84.9 % accuracy. A color histograms and multi-scale LBP features fed to an SVM with a Gaussian kernel pipeline obtains 82.2 %.

Chapter 10

Week 9

10.1 Multiple hypotheses image segmentation and classification with application to dietary assessment

In [32], the authors develop a mobile application to keep food records of a user that is taking pictures of one's meal. Their method can detect multiple food items in one picture. They use a color marker as an illumination and size indicator

They have a backend server to do the calculation. When the image has been classified, it is sent back to the user for confimation and review.

Their method is named "multiple hypotheses segmentation and classification (MHSC)". It is an iterativ method composed of a segmentation, description (extraction of features) and classification step. At the end of the classification step, a confidence score is assigned. If the total score is inferior to a certain threshold, the overral process is repeated. The previous step confidence score and classification label is used to improve the segmentation.

Segmentation:

1. salient region detection: reject background
2. multiscale segmentation using normalized cut
3. fast rejection: remove too small segmented regions

Description: using different features to describe the sub-image (on the whole segmented region or neighborhoods of pixelx)

- color (global descriptors):
  - first and second moment of each channel for RGB, YCbCr, L*a*b*, and HSV color spaces
  - first and second moment of the entropy in RGB
  - predominant color describtor
- texture (global descriptors):
  - statistics (entropy, moment) extracted from the Gradient Orientation Spatial-Dependence Matrix
  - entropy categorization and fractal dimension estimation
  - estimate the fractal dimension of the response of different gabor filter
- local feature (for each one, use of Bag-of-Words to form a visual vocabulary):
  - SIFT descriptors for RGB
  - SURF for RGB
  - SIFT descriptors for each channel of the RGB representation

- steerable filters

Classfication: classify each of the 12 descriptors independantly and use a late fusion function (either maximum confidence score or majority vote) to decide the final class. K-NN and SVM classifiers are used.

The dataset is composed of 83 classes (79 food classes and utensils, glasses, plates, and plastic cups), each class has at least 30 images.

The best descriptors is global color statistics. It obtains 68 % with KNN and 62 % with SVM.

Regrouping all the descriptors, the best accuracy is 75 %, using K-NN with the maximum confidence score (selecting for each kind of descriptors the top 8 classes)

10.2 Food image classification using local appearance and global structural information

In [33], the authors propos a food classification method using local appearance and global structural information of food objects.

Feature:

- texture features: encode the local texture with two methods:
  - local binary pattern (LBP)
  - non-redundant LBP (NRLBP)
- structural information: shape context descriptor between the interests points detected by SIFT

For the classification, they use the χ^2 distance between two histograms to pick the class. They have two methods to obtain the histogram:

- Create a codebook of LBP / NRLBP histograms, the points of interest being detected by SIFT. It is cluster using the K-means algorithm. AN image is represented by a bag-of-feature: describe the picture by the histogram of visual word frequency.
- each class of food has an individual codebook associated (filtering out the most frequent codewords)

Dataset: PFID: they obtain 0.68 % average accuracy for the method 1 and NRLBP, O.69 % for the method 2 and NRLBP. new dataset: 290 images, 5 categories (cakes, carrots, custards, pasta and pizza): 0.55 % for the method 1 and NRLBP, 0.63 % for the method 2 and NRLBP

Chapter 11

Week 10

11.1 A comparative analysis of edge and color based segmentation for orange fruit recognition

In [34], the authors describes and compare two segmentation method to localize an orange in a picture. The first method is based on edge segmentation, the second one on colour. These two algoriths are then applied on a small dataset of 20 pictures.

In more details, the methods are:

- edge based segmentation
    1. canny edge segmentation
    2. non-maximum suppression to suppress non-maxima pixels
    3. classfication of each pixels
- colour based segmentation
    1. gaussian low pass filter to normalize the lightning condition
    2. convert the image from RGB representation to $L * a * b$
    3. take the a channel 'a' to get a bunary image
    4. remove small object
    5. fill the binary image regions and holes

As already stated, the dataset is composed of 20 orange images (only one orange per image), with different lighting conditions and backgrounds (pictures are taken from the Internet).

Applying these processes, the authors get 85 % accuracy (17 out of 20) for the color segmentation, the edge detection method "was not successful" (impossible to detect only the orange edges among the background).

11.2 Leveraging context to support automated food recognition in restaurants

In [35], the authors develop an application to recognize food items from an image taken by the user in a restaurant. It uses some contextual data (the geo-localization) to improve the classification. Indeed, they use geo-localization to get the menu from internet and interrogate Google Search to get images (extract the top 50 pictures) of 15 dishes from the menu. These images are used as weakly-labeled training images.

The first step is the segmentation to localize the food and ignore the background through hierachical segmentation.

The feature descriptions are:

- color moment invariants
- hue histograms
- SIFT
- RGB SIFT: SIFT component for each RGB channel
- C-SIFT: a color invariant SIFT
- Opponent-SIFT: SIFT on colour-opponent channels

For the 4 SIFT representations: they build a codebooks of 100 000 visual words (using k-means clustering, k = 1000) to build Bag-of-Word histogram.

Then, for the image classification, they adopt the SMO-MKL (Sequential minimal optimization - Multiple kernel learning) multi-class SVM (Support-vector machine, χ^2 kernel) framework

It is applied on these two datasets:

- PFID to compare to other recognition system (baseline provided directly by the PFID in [19]). Their method obtain 48.5 % accuracy.
- image from 10 restaurants (divided in 5 different types of food: American, Indian, Italian, Mexican and Thai). It is composed of 600 pictures, 300 taken with a smartphone, 300 with Google glass.

  The overall average accuracy is 63.33%, only 15.67% without localization.

Chapter 12

Process

12.1 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 [36])

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
- k-nearest neighborhood
- SVM

12.2 Results

using 5 fold cross validation

using LBP (40 bins) + HS (10 * 10 bins) + Hu-moments

- decision tree: 16% (500 trees, gini)
- random forest: 6.6 % (gini)
- k-nearest neighborhood: (k=10, distance metric: minkowski, weights of each neighborhood point: uniform): 9%
- SGD classifier: 5.6 %

12.3 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 [37]. 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 1000-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 [38]

Then we apply the SVM classifier.

Bibliography

[1] Kiyoharu Aizawa et al. "Food balance estimation by using personal dietary tendencies in a multimedia food log". In: *IEEE Transactions on Multimedia* 15.8 (Dec. 2013), pp. 2176–2185. ISSN: 15209210. DOI: 10.1109/TMM.2013.2271474. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6548059.

[2] Parisa Pouladzadeh et al. "A virtualization mechanism for real-time multimedia-assisted mobile food recognition application in cloud computing". In: *Cluster Computing* 18.3 (Sept. 2015), pp. 1099–1110. ISSN: 15737543. DOI: 10.1007/s10586-015-0468-2. URL: http://link.springer.com/10.1007/s10586-015-0468-2.

[3] Hajime Hoashi, Taichi Joutou, and Keiji Yanai. "Image recognition of 85 food categories by feature fusion". In: *Proceedings - 2010 IEEE International Symposium on Multimedia, ISM 2010*. IEEE, Dec. 2010, pp. 296–301. ISBN: 9780769542171. DOI: 10.1109/ISM.2010.51. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5693856.

[4] 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.

[5] Minami Wazumi et al. "Auto-Recognition of Food Images Using SPIN Feature for Food-Log System". In: *Computer Sciences and Convergence Information Technology (ICCIT), 2011 6th International Conference on* (2011), pp. 874–877. URL: http://ieeexplore.ieee.org/xpls/abs%7B%5C_%7Dall.jsp?arnumber=6316741.

[6] Yoshiyuki Kawano and Keiji Yanai. "Food Image Recognition with Deep Convolutional Features". In: *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)* (2014), pp. 589–593. DOI: 10.1145/2638728.2641339. URL: http://dx.doi.org/10.1145/2638728.2641339.

[7] Marc Bosch et al. "Combining global and local features for food identification in dietary assessment". In: *Proceedings - International Conference on Image Processing, ICIP*. IEEE, Sept. 2011, pp. 1789–1792. ISBN: 9781457713033. DOI: 10.1109/ICIP.2011.6115809. arXiv: NIHMS150003. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6115809.

[8] Hokuto Kagaya, Kiyoharu Aizawa, and Makoto Ogawa. "Food Detection and Recognition Using Convolutional Neural Network". In: *ACM Multimedia*. 2. 2014, pp. 1085–1088. ISBN: 9781450330633. DOI: 10.1145/2647868.2654970. URL: http://dl.acm.org/citation.cfm?doid=2647868.2654970.

[9] Giovanni Maria Farinella, Marco Moltisanti, and Sebastiano Battiato. "Classifying food images represented as Bag of Textons". In: *2014 IEEE International Conference on Image Processing, ICIP 2014*. IEEE, Oct. 2014, pp. 5212–5216. ISBN: 9781479957514. DOI: 10.1109/ICIP.2014.7026055. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7026055.

[10] Taichi Joutou and Keiji Yanai. "A food image recognition system with Multiple Kernel Learning". In: *2009 16th IEEE International Conference on Image Processing (ICIP)* (Nov. 2009), pp. 285–288. ISSN: 9781424456543. DOI: 10.1109/ICIP.2009.5413400. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5413400.

[11] Shulin Yang et al. "Food recognition using statistics of pairwise local features". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, June 2010, pp. 2249–2256. ISBN: 9781424469840. DOI: 10.1109/CVPR.2010.5539907. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5539907.

[12] Zhimin Zong et al. "On the combination of local texture and global structure for food classification". In: *Proceedings - 2010 IEEE International Symposium on Multimedia, ISM 2010*. IEEE, Dec. 2010, pp. 204–211. ISBN: 9780769542171. DOI: 10.1109/ISM.2010.37. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5693842.

[13] Xin Wang et al. "Recipe recognition with large multimodal food dataset". In: *2015 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2015*. IEEE, June 2015, pp. 1–6. ISBN: 9781479970797. DOI: 10.1109/ICMEW.2015.7169757. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7169757.

[14] 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.

[15] 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.

[16] Mei-Yun Chen et al. "Automatic Chinese food identification and quantity estimation". In: *SIGGRAPH Asia* (2012), pp. 1–4. DOI: 10.1145/2407746.2407775. URL: http://dl.acm.org/citation.cfm?doid=2407746.2407775.

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

[18] 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.

[19] 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.

[20] Wu Wen and Yang Jie. "Fast food recognition from videos of eating for calorie estimation". In: *Proceedings - 2009 IEEE International Conference on Multimedia and Expo, ICME 2009* (2009), pp. 1210–1213. ISSN: 1945-7871. DOI: 10.1109/ICME.2009.5202718. URL: http://ieeexplore.ieee.org/xpls/abs%7B%5C_%7Dall.jsp?arnumber=5202718.

[21] Anderson Rocha et al. "Automatic produce classification from images using color, texture and appearance cues". In: *Proceedings - 21st Brazilian Symposium on Computer Graphics and Image Processing, SIBGRAPI 2008* (2008), pp. 3–10. ISSN: 1530-1834. DOI: 10.1109/SIBGRAPI.2008.9. URL: http://www.ic.unicamp.br/%7B~%7Dsiome/papers/rocha-sib08.pdf.

[22] D. Pishva et al. "A unified image segmentation approach with application to bread recognition". In: *WCC 2000 - ICSP 2000. 2000 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress 2000* 2 (2000), pp. 840–844. DOI: 10.1109/ICOSP.2000.891642. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=891642.

[23] D. Pishva et al. "Shape Based Segmentation and Color Distribution Analysis with Application to Bread Recognition". In: *WCC 2000 - ICSP 2000. 2000 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress 2000* 2 (2000), pp. 840–844. DOI: 10.1109/ICOSP.2000.891642. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=891642.

[24] Keigo Kitamura, Toshihiko Yamasaki, and Kiyoharu Aizawa. "Food log by analyzing food images". In: *ACM international conference on Multimedia* (2008), p. 999. DOI: 10.1145/1459359.1459548. URL: http://portal.acm.org/citation.cfm?doid=1459359.1459548.

[25] United States Department of Agriculture. *mypyramid.gov, steps to a healthier you.* 2005. URL: http://www.mypyramid.gov/.

[26] United States Department of Agriculture. *MyPlate.* 2005. URL: http://www.choosemyplate.gov/ (visited on 03/05/2016).

[27] S Arivazhagan et al. "Fruit Recognition using Color and Texture Features". In: *Information Sciences* 1.2 (2010), pp. 90–94. URL: http://cisjournal.org/archive/vol1no1/vol1no1%7B%5C_%7D12.pdf.

[28] Fengqing Zhu et al. "Multilevel segmentation for food classification in dietary assessment". In: *2011 7th International Symposium on Image and Signal Processing and Analysis (ISPA)* Ispa (2011), pp. 337–342. ISSN: 1845-5921. URL: http://ieeexplore.ieee.org/xpls/abs%7B%5C_%7Dall.jsp?arnumber=6046629.

[29] Rana Almaghrabi et al. "A novel method for measuring nutrition intake based on food image". In: *2012 Ieee I2Mtc* (2012), pp. 366–370. ISSN: 1091-5281. DOI: 10.1109/I2MTC.2012.6229581. URL: http://ieeexplore.ieee.org/xpls/abs%7B%5C_%7Dall.jsp?arnumber=6229581.

[30] Wataru Shimoda and Keiji Yanai. "CNN-based food image segmentation without pixel-wise annotation". In: *New Trends in Image Analysis and Processing – ICIAP 2015 Workshops.* Vol. 9281. 2015, pp. 449–457. ISBN: 9783319232218. DOI: 10.1007/978-3-319-23222-5_55. URL: http://link.springer.com/chapter/10.1007/978-3-319-23222-5%7B%5C_%7D55.

[31] Stergios Christodoulidis and Marios Anthimopoulos. "Food Recognition for Dietary Assessment Using Deep Convolutional Neural Networks Stergios". In: *New Trends in Image Analysis and Processing – ICIAP 2015 Workshops* 9281 (2015), pp. 458–465. DOI: 10.1007/978-3-319-23222-5. URL: http://link.springer.com/10.1007/978-3-319-23222-5%7B%5C_%7D56%20http://link.springer.com/10.1007/978-3-319-23222-5.

[32] Fengqing Zhu et al. "Multiple hypotheses image segmentation and classification with application to dietary assessment". In: *IEEE Journal of Biomedical and Health Informatics* 19.1 (Jan. 2015), pp. 377–388. ISSN: 21682194. DOI: 10.1109/JBHI.2014.2304925. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6733271.

[33] Duc Thanh Nguyen et al. "Food image classification using local appearance and global structural information". In: *Neurocomputing* 140 (2014), pp. 242–251. ISSN: 18728286. DOI: 10.1016/j.neucom.2014.03.017. URL: http://www.sciencedirect.com/science/article/pii/S0925231214004317.

[34] R. Thendral, A. Suhasini, and N. Senthil. "A comparative analysis of edge and color based segmentation for orange fruit recognition". In: *International Conference on Communication and Signal Processing, ICCSP 2014 - Proceedings* (2014), pp. 463–466. DOI: 10.1109/ICCSP.2014.6949884. URL: http://ieeexplore.ieee.org/xpls/abs%7B%5C_%7Dall.jsp?arnumber=6949884.

[35] Vinay Bettadapura et al. "Leveraging context to support automated food recognition in restaurants". In: *Proceedings - 2015 IEEE Winter Conference on Applications of Computer Vision, WACV 2015*. 2015, pp. 580–587. ISBN: 9781479966820. DOI: 10.1109/WACV.2015.83. arXiv: 1510.02078. URL: http://www.vbettadapura.com/egocentric/food/.

[36] 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.

[37] 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.

[38] 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.