Leveraging robust signatures for mobile robot semantic localization

Javier Redolfi[1] and Jorge Sánchez[1,2]

[1] Centro de Investigación en Informática para la Ingeniería,
Universidad Tecnológica Nacional, Facultad Regional Córdoba,
X5016ZAA, Córdoba, Argentine, Tel: +54 351 5986044
[2] CIEM-CONICET, FaMAF, Universidad Nacional de Córdoba,
X5000HUA, Córdoba, Argentine, Tel: +54 351 4334051 int. 309
jsanchez@scdt.frc.utn.edu.ar

Abstract. This paper describes the participation of the CIII UTN FRC team in the ImageCLEF 2012 Robot Vision Challenge. The challenge was focused on the problem of visual place classification in indoor environments. During the competition, participants were asked to classify images according to the room in which they were acquired, using the information provided by RGB and depth images only. We based our approach on the Fisher Vector representation –a robust signature recently proposed in the literature– and the use of efficient linear classifiers. In order to exploit the information provided by different information channels, we adopted a simple fusion strategy and generated classification scores for each image in the sequence. Two tasks were proposed during the competition: in the first, images had to be classified independently of one another while, in the second, it was possible to exploit the temporal continuity of the stream. For the first task, we adopted a simple threshold based classification scheme. For the second, we considered the classification of groups of images instead of single frames. These groups, i.e. temporal segments, were automatically generated based on the visual similarity of the images in the sequence. Our team ranked first on both tasks, showing the effectiveness of the proposed schemes.

Keywords: Fisher vectors, place recognition, semantic localization, temporal segmentation.

1 Introduction

In the 2012 edition of the ImageCLEF Robot Vision Challenge, participant were asked to classify functional areas based on sequences of images acquired by a mobile robot within an office environment, either in a frame-by-frame basis (obligatory task) or by exploiting the temporal continuity of the image stream (optional task). For learning the classifiers, the organizers provided training sequences consisting on RGB and depth images acquired under different lighting conditions.

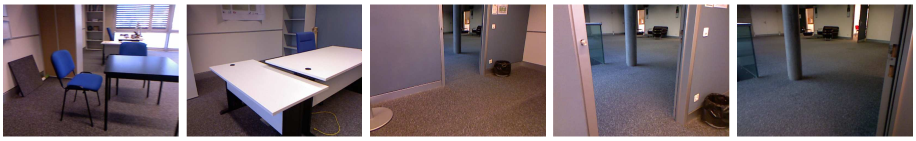

Fig. 1: Sample images from the "training2" sequence of the ImageCLEF Robot Vision Challenge 2012 dataset. All images belong to the "ProfessorOffice" class. Note the great amount of variability in the visual appearance within this group of images. Also, the last two seems more closely related to the "Corridor" class instead of that provided as ground truth.

This paper describes the participation of the CIII UTN FRC team in both tasks. Our methods leverage recent advances in the fields of image classification and retrieval, in which robust and efficient representations have been devised. Particularly, we consider the state-of-the-art Fisher Vector (FV) representation [6,8] which has been recently shown to give excellent results in a wide range of problems [8,3,2].

Before introducing the core components in our system, we first highlight some of the differences between the problem of visual place classification (VPC) in robotics and the more general problem of automatic image annotation (AIA)[3], i.e. the problem of assigning labels to images based on its content. First, the labeling of images in VPC is based on the physical location of the robot instead of a visually well defined concept. This makes the labeling of training images in some cases ambiguous, as images acquired at a particular location might reflect a different visual concept than the one assigned to them (e.g. last images in the sequence of Fig. 1). Second, images acquired for VPC exhibit a great degree of redundancy due to the temporal continuity of the image stream, i.e. labels associated to images acquired close in time are likely to belong to the same concept and share similar appearance. These peculiarities, originated in the very definition of the problem, make the visual classification of places a very challenging task.

The paper is organized as follows. In Sec. 2 we give a high level description of the different stages in our system. In Sec. 3, 4 and 5 we describe in detail the representation we use as well as the different classification schemes we applied. In Sec. 6 we present our experimental setup and in Sec. 7 we show results using the training set provided by the organizers of the challenge. Finally, in Sec. 8 we draw some conclusions.

2 System Overview

In this section we describe the core components of our system. The methods we applied in solving both the obligatory and optional tasks comprise the following processing steps:

[3] The problem is also known in the literature as image classification, categorization or tagging.

- *Encoding*: images must be robustly represented in order to capture high level properties of the scene. We rely on the state-of-the-art Fisher Vector image signature. As far as we know, this is the first time such a representation is applied in robotics.
- *Scoring*: we generate, for each image and concept, a score that provide us with a measure on how likely is for an image to have been acquired at a particular location. We use simple linear classifiers, which are efficient both to train and to evaluate.
- *Classification*: based on the scores obtained in the previous step, we generate a prediction of the robot actual location. We consider two cases:
  1. Individual frames are classified without taking into account the temporal consistency of the stream. We treat this problem as a simple (baseline) image classification task using FV and efficient linear classifiers.
  2. The image stream is automatically segmented into visually similar groups of images and the classification is performed in a segment-by-segment basis. We propose an efficient temporal segmentation algorithm based on representation properties of the FV signature.

3 Encoding

We provide a brief overview of the representation in which our method is based, namely, the Fisher vector image signature. Further details can be found in [6,8].

3.1 Fisher Vectors and the Similarity Between Images

Let $u_\lambda : \mathbb{R}^D \rightarrow \mathbb{R}_+$ be a pdf of parameter vector λ modelling the generation process of low-level descriptors in *any* image. Let $X = \{x_n, n = 1, \cdots, N\}$ be a iid sample of such D-dimensional descriptors extracted from a given image. The Fisher vector is defined as

$$\mathcal{G}_\lambda^X = L_\lambda G_\lambda^X. \tag{1}$$

Here, G_λ^X denotes the gradient of the (average) log-likelihood of X under u_λ:

$$G_\lambda^X = \frac{1}{N} \sum_{n=1}^{N} \nabla_\lambda \log u_\lambda(x_n) \ , \tag{2}$$

and L_λ is a diagonal normalizer. We model u_λ as a mixture of M Gaussians with diagonal covariances, i.e. $u_\lambda(x) = \sum_{i=1}^{M} w_i u_i(x)$, parametrized in terms of $\lambda = \{w_i, \mu_i, \sigma_i, i = 1, \cdots, M\}$. Here, w_i, μ_i and σ_i^2 denote, respectively, the mixing weight, mean and variance vector corresponding to the ith component of the mixture. It can be shown that [6][4]:

$$\mathcal{G}_{\mu_i}^X = \frac{1}{N\sqrt{w_i}} \sum_{n=1}^{N} \gamma_n(i) \left(\frac{x_n - \mu_i}{\sigma_i} \right) \tag{3}$$

$$\mathcal{G}_{\sigma_i}^X = \frac{1}{N\sqrt{2w_i}} \sum_{n=1}^{N} \gamma_n(i) \left[\left(\frac{x_n - \mu_i}{\sigma_i} \right)^2 - 1 \right] \tag{4}$$

with $\gamma_n(i)$ representing the soft assignment of low-level descriptors to components of the mixture, i.e. $\gamma_n(i) = w_i u_i(x_n) / \sum_{j=1}^{M} w_j u_j(x_n)$. The image signature is the concatenation of partial terms[5], i.e.

$$\mathcal{G}^X = \left(\mathcal{G}_{\mu_1}^{X\,T}, \cdots, \mathcal{G}_{\mu_M}^{X\ \ T}, \mathcal{G}_{\sigma_1}^{X\,T}, \cdots, \mathcal{G}_{\sigma_M}^{X\ \ T} \right)^T \ , \tag{5}$$

resulting in a vector of dimensionality $E = 2MD$. Following [8], we apply the transformation $f(z) = sign(z)\sqrt{|z|}$ independently on each dimension and L_2-normalize the transformed vector. These transformations have been shown to be highly beneficial in classification [8,2] as well as in image retrieval problems [7,3]. An important property of the transformed representation is that it allows the similarity between images to be measured efficiently by a simple dot-product between their FVs [8,3]. Moreover, as the transformed vectors are L_2-normalized, the similarity between FVs –as measured by the dot-product– is upper-bounded by 1. In what follows, we use $\mathcal{G}_{norm}^X$ to denote the transformed signature.

3.2 Low-level Descriptors

We extract two sets of low-level feature descriptors per image, computed independently from the luminance (Y) and depth (D) channels of each input frame. These sets of descriptors are used to compute two separate FVs that we denote by $\mathcal{G}_{norm}^{X_{lum}}$ and $\mathcal{G}_{norm}^{X_{depth}}$ respectively. Note that these two FVs originate from different probabilistic models, i.e. u_λ^{lum} and u_λ^{depth}. The parameters for these models can be estimated using the expectation maximization (EM) algorithm and a large set of descriptors.

4 Scoring

Let us denote by $\mathcal{C} = \{1, \ldots, C\}$ the set of concepts, i.e. locations defining the problem. For each channel (luminance and depth), we learn a set of C binary classifiers that provide us with a measure of how likely is for a given image to have been acquired on a particular location in the environment. Concretely, we generate a set of C linear predictors per channel, of the form:

$$s_c^\xi = {\theta_c}^T \mathcal{G}_{norm}^{X_\xi} + b_c^\xi, \tag{6}$$

[4] Vector divisions must be understood as term-by-term operations.

[5] We consider the gradient w.r.t. the mean and variance vectors only as the gradient w.r.t. the mixing weights has shown to provide little discriminative information [6].

where $\theta_c^\xi \in \mathbb{R}^E$, $b_c^\xi \in \mathbb{R}$ and $\xi \in \{lum, depth\}$. As we rely on simple linear models, learning the parameters for the $2C$ classifiers can be done very efficiently, e.g. by using Stochastic Gradient Descent (SGD) [1]. Given a test image, we generate a single score per class by computing the unweighted average of s_c^{lum} and s_c^{depth}. We denote this score by s_c.

5 Classification

In this section we describe our approach for robust place classification for both tasks of the challenge.

5.1 Obligatory Task (Task 1)

For the obligatory task, images have to be classified without considering the temporal continuity of the image stream, i.e. frame-by-frame. In cases of uncertainty, the system is allowed to refrain from making a decision (thus avoiding penalization points).

We treat this task as a simple (baseline) classification problem. Deciding to which class an image belongs was done according to the following rule:

$$\hat{c} = \begin{cases} c = \arg\max_i s_i, & \text{if } s_c > \alpha \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Here, 0 denotes “no classification”, i.e. the image is left unclassified. The parameter α is set empirically by cross-validation.

5.2 Optional Task (Task 2)

For this task, participants were allowed to exploit the temporal continuity of the image stream. This task also characterizes by the presence of *kidnappings*: situations in which the robot abruptly changes its location from one room to another.

Before introducing the methods we applied for this task, let us first introduce some notation. Let $I_{1:T} = \{I_t, t = 1 \dots T\}$, $I_t \in \mathcal{I}$, be a sequence of images acquired by a mobile robot up to time T. With a slight abuse of notation, we denote by $s_c(t)$ the classification score computed for image I_t and class $c \in \mathcal{C}$. We define the score vector $\mathbf{s}(t) := (s_1(t), \dots, s_C(t))^T$. Similarly, we use the notation $\mathcal{G}_{norm}^{X_\xi}(t)$ to represent the FV computed for image I_t using the set of descriptors extracted from channel $\xi \in \{lum, depth\}$.

Temporal Segmentation. Let us assume that for every t and t' there exists a function $m(t, t')$ that provide us with a measure of the *similarity* between images I_t and $I_{t'}$. Given a reference image I_a, we define a temporal segment as the sequence $I_{a:b} = \{I_t, t = a, \dots, b\}$, where b is the greatest integer such that

Algorithm 1 Temporal segmentation and classification

```
$t_0 \leftarrow 1$
$\mathcal{S} \leftarrow \{\mathbf{s}(1)\}$
for $t = 2, \ldots$ do
    if $m(t_0, t) < m_0$ then
        Classify images in $I_{t_0:t-1}$ according to MCT or MVT using the scores in $\mathcal{S}$
        $t_0 \leftarrow t$
        $\mathcal{S} \leftarrow \{\mathbf{s}(t)\}$
    else
        $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{s}(t)\}$
    end if
end for
```

$m(a, b) \geq m_0$ and m_0 a free parameter. Reference images are selected as the first image that follows a previously computed segment. The first of such reference frames is chosen as the first image in the sequence. The classification of images is performed segment-by-segment instead of frame-by-frame. Algorithm 1 provides an overview of our approach for temporal segmentation and classification.

As a proxy for $m(t, t')$ we use the dot product between FVs computed from either the luminance or depth channel features of I_t and $I_{t'}$ (Sec. 3.1). This provides us with a measure consistent with the classification model. It holds that $|m(t, t')| \leq 1$.

Using the above procedure, dealing with kidnapping situations becomes rather natural since, in such cases, the visual appearance of images is likely to change considerably from one frame to another. This abrupt change in appearance will trigger the generation of a new reference image and the classification of the segment extracted just before the kidnap point.

Segment classification. Based on the above definition, we classify all images in the segment $I_{a:b}$ according to one of the following rules: *a) Maximum confidence and threshold* (MCT), which takes into account the confidence of the classifiers w.r.t. the best alternative hypothesis within a given segment; or *b) Majority vote and threshold* (MVT), which tries to exploit the temporal and semantic consistency of the images. Details for these rules are given next.

Maximum confidence and threshold (MCT). Let $c_1, c_2 \in \{1, \cdots, C\}$ denote the indices to the best and second best scoring classifiers at time t and let $d(t) := s_{c_1}(t) - s_{c_2}(t) \geq 0$ denote the difference between the corresponding scores. All images in the segment $I_{a:b}$ are classified as belonging to class $\hat{c}$ according to the following rule:

$$\hat{c} = \begin{cases} c_1, & \text{if } s_{c_1}(u) > \beta,\ u = \arg\max_{t \in [a,b]} d(t) \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

Majority vote and threshold (MVT). Let $v_{a:b}(c) := \#\{s_c(t) > \beta, t = a, \ldots, b\}$ denote the number of times the classification score for the cth classifier is above β for images in the segment $I_{a:b}$. We consider the following voting strategy for the classification of the temporal segment $I_{a:b}$:

$$\hat{c} = \arg\max_c v_{a:b}(c) \tag{9}$$

As before, $\hat{c} = 0$ means that images in $I_{a:b}$ are left unclassified. In both cases, the parameter β is set empirically by cross-validation.

6 Experimental Setup

In this section we provide a detailed explanation of our experimental procedure.

6.1 Dataset

The training set for the Robot Vision Challenge 2012 consists of three sequences of 2667, 2532 and 1913 RGBD images respectively, acquired under different illumination conditions within the same floor of an office environment. They include motions in both clockwise and counter clockwise directions. Performance is measured based on the number of correctly and misclassified images in a given sequence and it varies from task to task.

Further details regarding the dataset and the evaluation methodology can be readily found in [5].

6.2 Low-level Features

Both RGB and depth images were reduced at half their original resolution before computations. We extracted 128-dimensional SIFT descriptors [4] from local patches of 32×32 pixels located at the nodes of a regular grid (step size of 4 pixels). We used the DSIFT implementation of [9]. We did not perform any normalization (rotation, intensity, etc.) on the image patches before computations. To account for variations in scale, we built a resolution pyramid of 5 levels using a scale factor of 0.707 between them. SIFT descriptors were extracted independently on each level using the procedure described above. In the case of depth images, we considered only descriptors whose magnitude was greater than a small value (set to 10^{-3} in our experiments). The dimensionality of SIFT descriptors was further reduced to 80 by Principal Components Analysis (PCA). PCA projection matrices were learned from a set of 10^6 randomly sampled descriptors from the training set.

6.3 Generative Model

For each channel, we trained a Gaussian Mixture Model (GMM) with M components under a Maximum Likelihood (ML) criterion using the Expectation-Maximization (EM) algorithm. We used 10^6 random samples from the training set. We initialized the EM iterations by running k-means and using the statistics of cluster assignments (relative count, mean and variance vectors) as initial estimates.

6.4 Base Classifiers

As base classifiers we used linear SVMs trained on the primal using Stochastic Gradient Descent (SGD) [1], i.e. minimizing the L_2 regularized hinge-loss in a sample-by-sample basis. The regularization parameter λ was chosen by cross-validation on the training set. We trained C classifiers per channel following a *one-vs-all* strategy. i.e. when training the models for class c we used the samples of that class as positives and the rest as negatives.

7 Results

In order to allow the system to cope with changes in illumination and the robot motion direction, we considered the following data augmentation strategies: *i)* adding new images by simulating uniform changes in illumination, i.e. generating darker/brighter versions of randomly sampled images; *ii)* generating mirrored (left-to-right) versions of the images provided as training material. In the first case, we did not observe any noticeable improvement while, in the second, we observed an increase of +30% (on average) w.r.t. a system trained using the original data only. This is to be expected, as our low-level features (i.e. SIFT vectors) are based on gradient information which makes them insensitive to uniform changes in the illumination. On the other hand, adding mirrored samples to the training set let the system learn up to some degree the symmetries originated from the changes in the robot motion.

7.1 Task 1

In this subsection we evaluate the performance of our system in classifying images independently (frame-by-frame). In particular, we consider the following aspects: *i)* the impact of using increasingly complex models (i.e. number of Gaussians, M); *ii)* the benefits of using different representation channels. For each configuration, we ran three experiments using different train/test splits of the data, using two of the sequences for training and the third for testing. Results are reported on Table 1. We show recognition performance[6] for models with $M = 256$, 512 and 1024 Gaussians and systems based on single and multiple descriptor channels. Results on Table 1 were obtained by setting $\alpha = -\infty$ in Eq. (7), i.e. *argmax* rule without thresholding. The classification performance obtained with $\alpha = -0.5$ is shown in parentheses.

If we consider the different train/test configurations, it can be observed a big drop in performance for the system trained on sequences 1 and 2. This drop can

[6] The score was computed using the scripts provided by the organizers of the challenge.

Table 1: Recognition performance for "Task 1" for models involving $M = 256$, 512 and 1024 Gaussian and different representation channels. See text for details.

		Luminance			Depth			Lum+Depth		
Train	Test	256	512	1024	256	512	1024	256	512	1024
1 & 2	3	275	411	491	511	489	609	835	897	965
2 & 3	1	1941	1945	1969	1403	1429	1461	1947	1961	1941
3 & 1	2	2150	2158	2154	1548	1564	1598	2050	2078	2094
Average		1455	1505	1538	1154	1161	1223	1611	1645	1667
($\alpha = -0.5$)		-	-	-	-	-	-	(1657)	(1718)	(1722)

be explained by noting that sequence 3 was acquired under very poor illumination. A system trained using only images acquired under "normal" illumination does not generalizes well to this previously unseen scenario. In contrast, systems to which this sequence was shown during training exhibit much better performance (second and third row in the table). As expected, the system based on the luminance channel alone performs worse than the system using depth information when testing on sequence 3. On the contrary, the luminance channel shows better performance on test sequences 1 and 2. The combination of both channels brings large improvements in all scenarios. Increasing the model complexity (number of components in the mixture) can bring additional improvements at the cost of a greater computational cost.

The system we submitted during the challenge included both luminance and depth features and models with $M = 1024$ Gaussians. The threshold parameter was set to $\alpha = -0.5$. Our system ranked first, achieving a score of 2071 points.

7.2 Task 2

For this task, we first evaluate the influence of the parameter m_0 in classification performance (Sec. 5.2, temporal segmentation). m_0 controls the degree to which an image is considered similar to another of reference, i.e if it belongs to the temporal segment defined by the second. Fig. 2 (left) shows the average score as a function of m_0 for different choices of the similarity measure (dot-product between luminance or depth FVs) and classification rules (Sec. 5.2), e.g. Lum+MVT corresponds to the system using the dot-product between luminance FVs for segmentation and the MVT rule for classification. Fig. 2 (right) show the average length of the temporal segments obtained by using FVs from either channel.

It can be observed that for m_0 above 0.2, using luminance FVs for segmentation leads to better results than with depth FVs. Using luminance FVs, performance reaches a peak at $m_0 = 0.3$ (Lum+MCT: 1836, Lum+MVT: 1871). Within this range, MVT performs better than MCT. For values of $m_0 < 0.2$, the segmentation using depth FVs leads to better results. In this case, a peak is observed at $m_0 = 0.1$ (Depth+MCT: 1919, Lum+MVT: 1888) with MCT performing better. As a comparison, the average score obtained by setting $m_0 = 1$

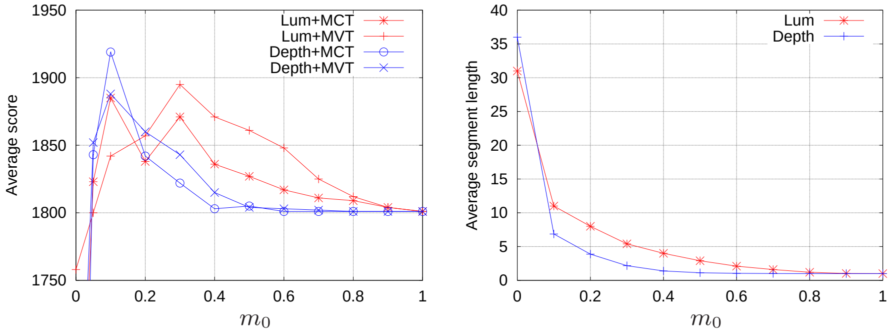

Fig. 2: Experiments for Task 2 using models with $M = 1024$ Gaussians and $\beta = -0.4$, given as a function of the segmentation threshold m_0. *Left*: average classification score for different choices of the similarity measure and classification rule; *Right:* average segment length.

(frame-by-frame classification) on this task is 1801. The average segment length at the above points is 5.4 and 6.86 for luminance ($m_0 = 0.3$) and depth ($m_0 = 0.1$) features, respectively. For the same value of m_0, luminance FVs lead to larger segments.

For the competition we submitted two systems: Depth+MCT and Depth+MVT using, as before, 1024 Gaussians. We set the segmentation and classification thresholds to $m_0 = 0.1$ and $\beta = -0.4$, respectively. Our systems ranked first, achieving 3930 points on this task.

7.3 Timings

Finally, we report computation times for the system based on models with $M = 1024$ mixture components. Reported times were measured on a AMD Opteron machine (8 cores @ 2GHz) with 8 GB of RAM. Table 2 show estimated times for both offline and online processes. Additionally, we report FV computation times for $M = 256$ and 512 Gaussians.

Table 2: Computation times for models with $M = 1024$ Gaussians. PCA and GMM parameters were estimated on 10^6 randomly sampled features.

Offline	PCA training	4min / channel
	GMM training ($M = 1024$)	1h 30min / channel
	Classifier training	5min / class / channel
Online	SIFT+PCA	170msec / image / channel
	FV ($M = 256$)	220msec / image / channel
	FV ($M = 512$)	310msec / image / channel
	FV ($M = 1024$)	490msec / image / channel
	Scoring	2.2msec / image / class

8 Conclusions

This paper describes the CIII participation at ImageCLEF Robot Vision Challenge 2012. Our approach leverages recent advances in the fields of image classification and retrieval. We proposed a temporal segmentation methodology based on the visual similarity of images that allowed us to classify groups of images in a robust manner. Our team ranked first on both the obligatory and optional tasks of the challenge, showing the potentiality and effectiveness of our approach.

Acknowledgments. Jorge Sánchez was partially supported by a grant from CONICET, Argentina.

References

1. Bottou, L.: SGD. `http://leon.bottou.org/projects/sgd` (2007)
2. Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: Proc. BMVC (2011)
3. Jegou, H., Perronnin, F., Douze, M., Sánchez, J., Pérez, P., Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Tr. on Pattern Analysis and Machine Intelligence 34(9), 1704 –1716 (2012)
4. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Intl. Jrnl. on Computer Vision 60(2) (2004)
5. Martinez-Gomez, J., Varea, I.G., Caputo, B.: Overview of the imageclef 2012 robot vision task. In: CLEF 2012 working notes (2012)
6. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: Proc. CVPR (2007)
7. Perronnin, F., Liu, Y., Sánchez, J., Poirier, H.: Large-scale image retrieval with compressed fisher vectors. In: Proc. CVPR (2010)
8. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: Proc. ECCV (2010)
9. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/` (2008)