Exploiting Unlabelled Data for Hybrid Object Classification

Alex D. Holub & Pietro Perona
Computation and Neural Systems
California Institute of Technology
holub,perona@caltech.edu

Max Welling
Department of Computer Science
University of California Irvine
welling@ics.uci.edu

Abstract

We propose a semi-supervised learning algorithm for visual object categorization which utilizes statistical information from unlabelled data to increase classification performance. We build on an earlier hybrid generative-discriminative approach by Holub et al. [6] which extracts Fisher scores from generative models. The hybrid model allows us to combine the modelling power and flexibility of generative models with discriminative classifiers. Here we illustrate how the generative framework can be used to add prior knowledge obtained from unlabelled images, which the discriminative classifier can subsequently exploit. We illustrate the effects of using different sets of images as prior knowledge and find that the greatest benefits are incurred when the prior exemplars have similar statistics to the classes being discriminated between. Our tests show that strong performance (85%) can be obtained in discriminating between the faces of two different people using prior knowledge and only 2-3 training examples. Furthermore, we extend our approach to multi-class discrimination and show state-of-the-art performance on the Caltech101.

1 Introduction

The visual world contains thousands of object categories which we would like computers to recognize and distinguish between. Classification algorithms typically require many exemplars from each category in order to accurately represent intra-class variability. Given the large numbers of categories and training images necessary to create accurate representations of these categories it will ultimately become necessary to generalize information across object categories. Consider two examples of sets of object classes where generalization might be useful: (1) Most animals which fall into the class 'Mammals' are furry and therefor have a highly textured appearance. In addition most have 4 legs. (2) Human faces contain eyes, a nose, ears, a chin, etc. However individual human faces can differ along other dimensions including hair color, skin tone, and adiposity. It seems natural to exploit and share the statistical similarities across similar object categories in order to both reduce the number of training exemplars necessary, as well as increase the performance of the recognition systems. We propose to exploit these similarities by employing a hybrid generative-discriminative hybrid recognition system.

In previous work [6] it was shown that generative constellation models [11] can be combined into a discriminative framework using Fisher scores. This approach allows us to take advantage of the flexibility and modelling power of generative modelling while maintaining the classification performance of discriminative classifiers. In particular we are able to solve missing data problems within the generative framework. One particular missing data problem which is important for object recognition is the correspondence problem: how to find the mapping from an unordered set of detected features to object parts. The generative setting provides an intuitive solution which involves marginalization over a hidden 'mapping' variable (see below). Note that, by comparison, it is unclear how to solve this correspondence problem in a discriminative setting.

The problem of leveraging unlabelled data to improve a classifier is known as *semi-supervised learning* in the machine learning community. Although many interesting methods have been proposed (see e.g. [9] for an approach relevant to Fisher kernels) we decided to implement a relatively simple idea that attempts to learn the kernel using the available unlabelled data. In the context of Fisher kernels this boils down to learning a probabilistic model on the unlabelled data-set (possibly including the labelled data but ignoring their labels). We subsequently extract Fisher-scores based on this model, but evaluate it at the *labelled* data. These Fisher scores are combined into a kernel matrix and provide input to the SVM.

The use of prior knowledge within a constellation model framework was suggested by Fei Fei et al. [3] who extended the work of Fergus et al. [4] into a full Bayesian setting, thereby allowing for the introduction of prior distributions. The authors of [3] used unrelated categories to construct priors for learning new categories (in particular they performed experiments with the categories Airplanes, Leopards, Motorcycles, and Faces). This work raises the question of why unrelated categories, e.g. Airplanes, Leopards, and Motorcycles, should provide any useful information for learning Faces given that the shape and appearance models for these categories vary drastically among one another? We approach the issue of learning with prior knowledge in a different setting, and attempt to provide more insight into the questions raised above.

The paper is organized as follows: We first provide a brief overview of the hybrid modelling approach, discussing both the generative constellation model as well as a method for extracting fisher scores from this model. We then describe experiments on both 2-class face discrimination tasks as well as multi-class object discrimination. We conclude with a discussion.

2 Methods

Our proposed method consists of two steps: 1) Train an ensemble of generative models separately for each class, 2) extract Fisher scores to construct Fisher-kernels for all the models separately.

The Generative Model Any generative model can be used to construct a Fisher Kernel, as long as we can compute the gradients w.r.t. the parameters. We chose to experiment with a simplified probabilistic constellation model [4, 11, 7]. We do not explicitly model occlusion or relative scale as done in [4]. Although it is potentially advantageous to include these terms, excluding them offer us computational advantages and in particular it allows us to use more features than would be possible in a full model.

The constellation model is composed of a number of parts that correspond to detections of interest points in an image. Numerous methods exist for extracting and representing these interest points. The detectors used in our experiments are listed below. Both location, X_i and appearance information, A_i are extracted from images based on the detected locations.

To represent appearance, we used 11x11 pixel normalized image patches at the location and scale indicated by the detectors. We reduce the dimensionality of the patches to 10-20 by constructing a PCA basis using features from only the training images and projecting onto that basis.

Given a set of features that were detected in an image, we need to assign a unique detection to every model component (part). This is called the correspondence problem. Since we do not known a priori which interest point belongs to which model component, we introduce a hypothesis variable h which maps interest points to model parts. We order the interest points selected by h in ascending order of x-position and represent the positions of all interest points relative to the left-most interest point, thereby allowing for translational invariance. Given this correspondence, we model appearance and relative location of the model parts as joint Gaussian models $p(I_i|h_i,\theta) = p(X_i|h_i,\theta_X)p(A_i|h_i,\theta_A)$ where I_i denotes image i, $i = 1..N$ and $\theta = \{\theta_A, \theta_X\}$ are the parameters representing the means and diagonal variance components of both the appearance and location models. We marginalize over the hypothesis variable to obtain the following expression for the log likelihood for a particular class,

$$\sum_i \log\left(p(I_i)\right) = \sum_i \log\left(\sum_{h_i} p(A_i, h_i|\theta_A)p(X_i, h_i|\theta_X)\right) \tag{1}$$

We train our generative models using the EM algorithm [2]. The algorithm involves iteratively calculating the expected values of the parameters of the model and then maximizing the parameters. The algorithm was terminated after 50 iterations or earlier if the log likelihood stopped increasing. We found empirically that the discriminative performance benefitted from keeping the models loose and we imposed minimum values on the diagonal variance of the Appearance portion of our models.

Fisher Scores and Fisher Kernels for the Constellation Model The Fisher Score, $\boldsymbol{\phi}_{\mathcal{M}}(I_i) = [\boldsymbol{\phi}_A(I_i), \boldsymbol{\phi}_X(I_i)]$, is the derivative of log likelihood of the parameters for the model $\mathcal{M}$ at image I_i. It is not hard to show that one can compute these derivatives using the following expressions which are readily available from the EM algorithm at convergence,

$$\boldsymbol{\phi}_X(I_i) = \frac{\partial}{\partial\theta_X}\log\left(p(I_i|\theta)\right) = \sum_{h_i} p(h_i|I_i,\theta)\frac{\partial}{\partial\theta_X}\log p(X_i, h_i|\theta_X) \tag{2}$$

$$\boldsymbol{\phi}_A(I_i) = \frac{\partial}{\partial\theta_A}\log\left(p(I_i|\theta)\right) = \sum_{h_i} p(h_i|I_i,\theta)\frac{\partial}{\partial\theta_A}\log p(A_i, h_i|\theta_A) \tag{3}$$

Note that despite a potentially variable number of detections in each image its Fisher score has a fixed length. This is because the hypothesis h maps features to a pre-specified number of parts and hence there is a fixed number of parameters in the model.

The Fisher scores are subsequently centered and scaled by subtracting the sample mean and dividing by the square root of the sample variance (we compute this transformation using the training set and apply the same transformation to the test set). Based on these normalized scores we compute the following Gaussian kernel,

$$K_{\text{RBF}}(I_i, I_j) = \exp\left(-\frac{1}{2}\sum_{\mathcal{M}} ||\boldsymbol{\phi}_{\mathcal{M}}(I_i) - \boldsymbol{\phi}_{\mathcal{M}}(I_j)||^2\right) \tag{4}$$

We note that we can train and test our discriminative models using any subset of the Fisher scores we like. For instance, we can use only the Shape scores or only the Appearance scores to train our models. Training using only a subset gives us an idea of how important that particular set of coefficients is for a particular discrimination task (see Figure 5).

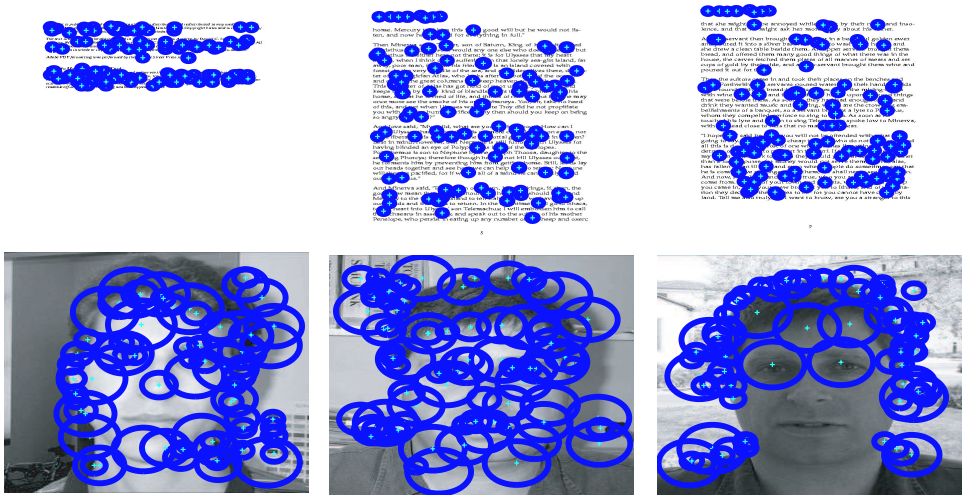

Figure 1: Features detected by the Kadir and Brady detector on image of (Top) the Odyssey text and (Bottom) the Faces-Easy Data-set. Note that the Faces-Easy data-set is equivalent to the Faces data-set of the Caltech 4 data-set except that the faces are cropped. Also note that the features found in the two sets of images have drastically different appearance statistics.

3 Experiments

We conducted experiments on two sets of data, the first involving 2-class face identity discrimination and the second involving multi-class object discrimination.

All generative models described below, unless otherwise noted, consist of 3 parts, a maximum of 30 detected interest points and 20 appearance coefficients.

Caltech Faces-Easy Data-Set The Caltech Faces-Easy (see Figure 1) consists of about 400 images of human faces. It is composed of 20 or more photographs of 19 individuals, while the remaining 40 or so images contain less exemplars. Thus we can divide the entire face category into smaller categories corresponding to individuals.

A linear kernel was used in the experiments below. Using an exponential kernel is difficult due to the inability to accurately tune the scale parameter σ: there are not enough exemplars to perform cross-validation.

Face Discrimination Results We compared performance using various different sets of images for learning the generative model (these correspond to the unlabelled images): (1) Faces-Easy data-set (using exemplars from all individuals except for the ones being trained/tested on), (2) the Caltech Background Images, (3) the Leopards data-set, (4) Images of printed pages from a copy of Homer's 'Odyssey'. Examples of the features found for classes (1) and (4) are shown in Figure 1 while (2) and (3) can be found at the website http://www.vision.caltech.edu/html-files/archive.html.

First a model was trained using images from the unlabelled data-set. Next, 2 individuals from the Faces-Easy set were selected at random to train the classifier. Normalized Fisher scores were extracted from the model. We did not include the training images in the unlabelled data-set. Median and 25th/75th quantiles were computed over 20 experiments by selecting 2 individuals at random.

Results are shown in Figure 2. There are several general lessons to learn: (1) The nature of the unlabelled data-set is critical: using a data-set unrelated to the classification problem at hand, e.g. the 'The Odyssey' or 'Leopards' data-sets in the context of face classification, results in the worst performance. Using a very general data-set, e.g. the Caltech 1 Background (BG1) data-set, results in better performance than using an unrelated data-set. Finally, using a data-set which describes the distribution of the images involved in the classification problem well, e.g. using the remaining face images from "Easy-Faces" to classify the faces of two held-out individuals, results in the best performance. (2) The impact of the unlabelled data-set tends to increase, as we add more exemplars. This is particularly true for the faces data-set, in which we notice an enormous increase in performance from 10 to 100 prior examples. The same qualitative effect is observed for the BG1 data-set. We observed that with few training examples the BG1 prior creates overfitted models with small variances, resulting in comparable performance to using the 'Odyssey' or 'Leopards' data-sets. As more training examples are added, the model becomes more general, and its utility as a prior improves.

A reasonable measure for how appropriate a kernel is for a particular classification task is the "kernel-alignment" proposed in [10],

$$A(\mathbf{y}\mathbf{y}^T, K) = \frac{\mathbf{y}^T K \mathbf{y}}{N\sqrt{\mathbf{tr}(K^T K)}} \tag{5}$$

where N is the number of training cases and $\mathbf{y}$ is a vector of $+1$ and -1 indicating the class of each data-case[1]. Figures 3 and 4 visualize the fact that certain unlabelled data-sets result in more appropriate kernels.

Caltech101 We also performed experiments within a multi-class framework, specifically we tried to discriminate between different classes within the Caltech101. These categories are varied in their visual appearances making an a priori decision about which feature detector to use problematic. For this reason, we chose to create multiple generative models using 3 different feature detectors: (1) Kadir and Brady [8], (2) multi-sclae Harris, and (3) multi-scale Hessian. Fisher scores were extracted for each of the generative models individually. These fisher scores can be combined by concatenating them into one large vector and subsequently used to train a kernel machine.

We chose the 1 vs. 1 discrimination learning paradigm for these multi-class experiments. Learning proceeded in the following manner: (1) exemplars from each class were separated into training and testing sets, (2) exemplars from the training set were used to create three generative models using features extracted from each of the 3 different detectors, (3) fisher scores were extracted for all training and testing examples from the generative models and combined into a single vector, (4) discriminative SVM classifiers were trained for each 1 vs. 1 discrimination task, (5) testing was performed on the test sets. We tended to get better performance by keeping the generative models loose. We note that data from all classes is used to create the generative model, such that the fisher scores extracted to train each classifier are implicitly using information from all the other classes, thereby following a similar semi-supervised learning paradigm as described above.

In figure 5 we illustrate the results on the Caltech101 which is also available at http://www.vision.caltech.edu/html-files/archive.html. We remove the "Faces-Easy" class since a "Faces" class already exists. In addition we remove the class "Background" since it contains no specific object category. There are several interesting issues to note in the plot.

[1]Why did we not choose the perfect kernel according to this metric, namely $K = \mathbf{y}\mathbf{y}^T$? The reason is that this kernel would overfit on the training data and exhibit poor generalization performance. The alignment measure is therefore only useful provided we do not overfit, which we would not expect given the fact that we learn the kernel on a separate, unlabelled data-set.

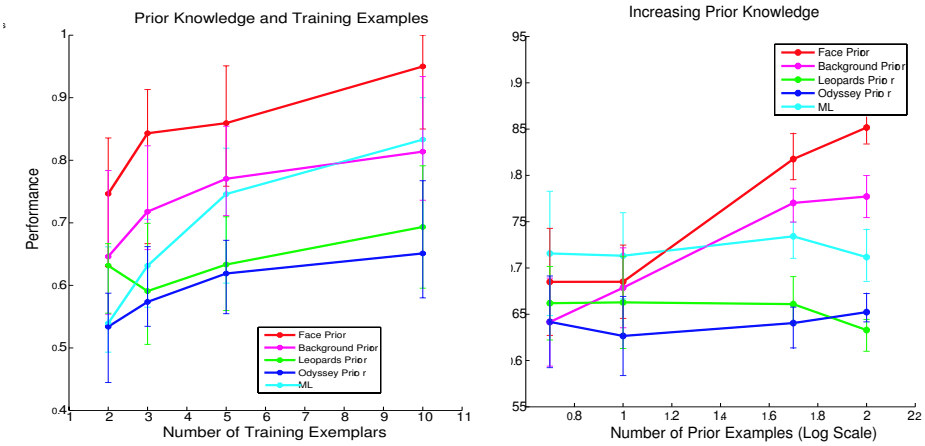

Figure 2: (Left) Performance on the 2-way faces classification problem as we vary the number of labelled training examples per class. Number of unlabelled examples was fixed at 100. (Right) Classification performance as a function of the number of unlabelled examples. 5 labelled training examples per class were used to train the SVM classifier. In both plots we show the median and 25th/75th quantiles computed over 20 experiments. Each line represents a different unlabelled data-set except for the cyan line which shows the maximum likelihood performance on the labelled training set only (i.e. it did not use any unlabelled data and classified by comparing the likelihood scores). Note that the nature of the unlabelled dataset has an important impact on the classification performance.

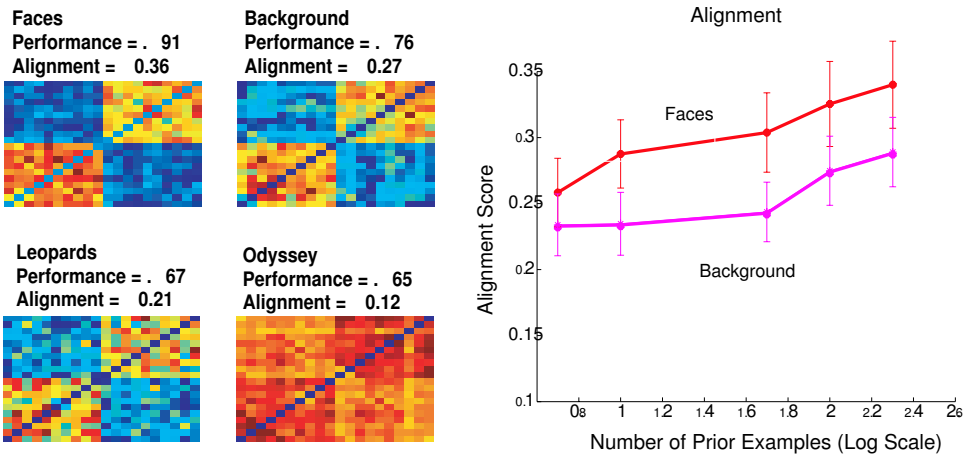

Figure 3: (Left) Kernel matrices for various models computed on different unlabelled datasets. Each model underlying a kernel was trained on 200 unlabelled data-cases. A kernel was computed as $\phi^T(x_i)\phi(x_j)$ with normalized Fisher scores and averaged over 20 experiments. Diagonal entries are zeroed out to improve resolution. For the Faces dataset one can easily discern a block-structure where the images in the same class are similar to each but dissimilar to the images of the other class (images in the same class correspond to first 10 entries and second 10 entries respectively and brighter colors indicate higher similarity between the data-points). Test performance and alignment values are also indicated. (Right) Alignment scores for increasing numbers of unlabelled examples when using the Faces and Background data-sets. The alignment increases as more prior examples are added indicating a more appropriate kernel is being used for this classification task.

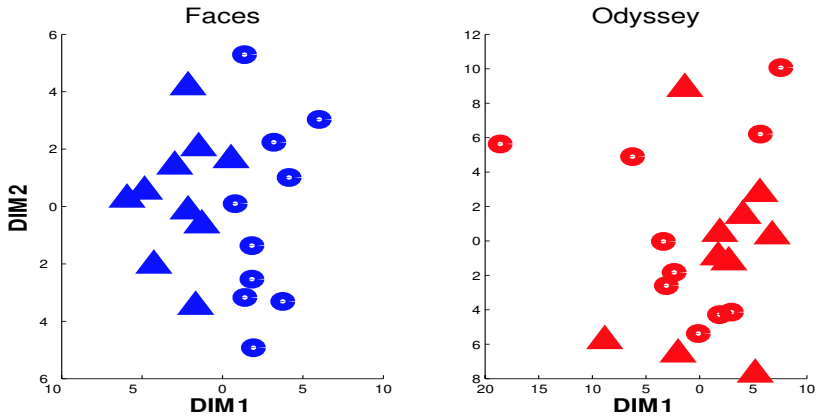

Figure 4: Visualization of different kernels learned from unlabelled data using "multi-dimensional scaling" (MDS). Distances in feature space were computed as $d_{ij} = ||\phi(x_i) - \phi(x_j)||$, where ϕ is normalized. These were input to the MDS procedure which embeds the data in a 2-D Euclidean space attempting to maintain the distances between data-points as best as it can. One can clearly see that the model learned on the Faces data (left) results in an embedding which is linearly separable (even in 2 dimensions), the embedding obtained from the Odyssey data-set (right) is not linearly separable (classes are represented by circles and triangles).

First, our algorithm performs quite well compared with other approaches. Second, the testing using all the exemplars for each category results in a biased estimate of performance. Finally, using only very few training exemplars, 5 in this case, results in performance well above chance, 17%.

4 Conclusion

We have shown how the use of unlabelled data can be exploited in semi-supervised learning paradigm to increase classification performance. In particular, we made use of a hybrid generative-discriminative learning scheme and showed how the generative portion can be used to represent the unlabelled data. We show strong performance increases in both 2-class face discrimination tasks as well as multi-class object discrimination tasks.

References

[1] Alex Berg, Tamara Berg, and Jitendra Malik. Shape matching and object recognition using low distortion correspondence. *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[2] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. 39:1–38, 1977.

[3] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples. *Computer Vision and Pattern Recognition (CVPR) Workshop on GMBV*, 2004.

[4] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, page 264, 2003.

[5] Kristen Grauman and Trevor Darrell. The pyramid match kernel: Discriminative classification with sets of image features. *International Conference on Computer Vision (ICCV)*, 2005.

[6] A. Holub, M. Welling, and P. Perona. Combining generative models and fisher kernels for object class recognition. *International Conference on Computer Vision (ICCV)*, 2005.

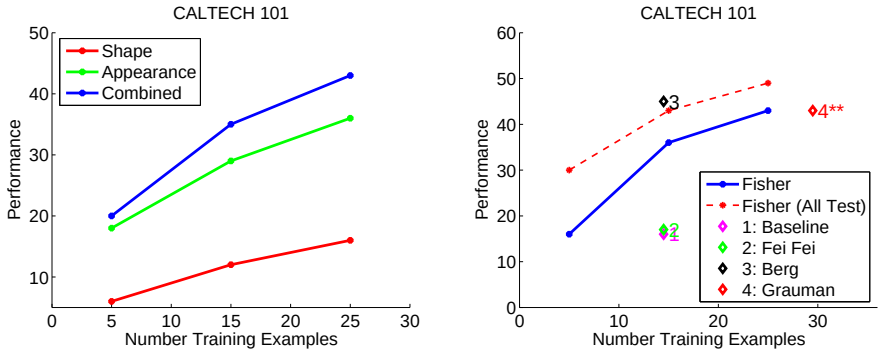

Figure 5: Performance on the Caltech 101. X-axis: the number of training examples. Y-axis: performance. (Left) Performance using different Fisher coefficients. The worst performance is obtained when only the Fisher shape coefficients are used and the best performance is obtained by using both shape and appearance coefficients. This is for a 3-part model using 25 detected features and a maximum of 50 testing samples. (Right) Comparison of our algorithm with other algorithms. The blue solid line is the performance of our algorithm using at most 50 testing examples in each category. The dotted red line is the same as the blue except that all available testing samples were used for each category. We notice that since some of the easier classes (airplanes-1000, motorcycles-1000, faces-600) contain far more exemplars than some of the harder categories (many of which have on the order of 30-70 exemplars). The diamonds indicate the performance of previous algorithms on this data-set. Baseline [1]: Histograms. Fei Fei [3]: Bayesian constellation model with priors. Berg [1]: Deformable shape matching. Implicit shape prior (thin plate spine) and segmentation of images. Grauman [5]: Kernel similarity metric. Note that these authors use all testing images (thereby directly comparable to the dotted red line). Note that the other authors eliminate this bias by either limiting the number of testing exemplars or by averaging performance across classes.

[7] Alex Holub and Pietro Perona. A discriminative framework for modeling object class. *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[8] Timor Kadir and Michael Brady. Saliency, scale and image description. *Int. J. Comput. Vision*, 45(2):83–105, 2001.

[9] M. Seeger. Covariance kernels from bayesian generative models. In *Advances in neural information processing systems*, volume 14.

[10] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[11] M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2000.