Discriminative Training of Hyper-feature Models for Object Identification*

Vidit Jain[1], Andras Ferencz[2] and Erik Learned-Miller[1]
[1] University of Massachusetts Amherst, Amherst MA USA
[2] MobilEye Vision Technologies, Hartford CT USA
[1] {vidit, elm}@cs.umass.edu, [2] ferencz@cs.berekeley.edu
http://vis-www.cs.umass.edu/projects/hyperfeatures/

Abstract

Object identification is the task of identifying specific objects belonging to the same class such as cars. We often need to recognize an object that we have only seen a few times. In fact, we often observe only one example of a particular object before we need to recognize it again. Thus we are interested in building a system which can learn to extract distinctive markers from a single example and which can then be used to identify the object in another image as "same" or "different".

Previous work by Ferencz et al. introduced the notion of hyper-features, which are properties of an image patch that can be used to estimate the utility of the patch in subsequent matching tasks. In this work, we show that hyper-feature based models can be more efficiently estimated using discriminative training techniques. In particular, we describe a new hyper-feature model based upon logistic regression that shows improved performance over previously published techniques. Our approach significantly outperforms Bayesian face recognition that is considered as a standard benchmark for face recognition.

1 Introduction

Distinguishing among similar objects within a class is more effective if we use expertise about the class. To build the best possible classifiers, we should use features that are repeatable and salient. In object identification, the features should be object specific and be able to discriminate between a particular object and similar objects of the same class. For example, door handles, headlights and roof tops might be distinctive markers for identifying cars. The complexity of determining these detailed features is increased by the general variability of different images of the same car. This "within-instance" variability is due to viewing angles, lighting and other factors.

An additional constraint for the object identification task is that we often need to recognize an object that we have seen only a few times. For humans, a single example is usually sufficient for finding distinctive features of an object given its class. For example,

*This work was partially supported by NSF CAREER award IIS-0546666.

BMVC 2006 doi:10.5244/C.20.37

if we are looking at a human face, we often notice the shape of the nose and lips, the color of the eyes, the hairdo, etc. We expect some of these features to provide interesting patches which might be useful for distingushing a particular face. The set of useful patches can be different for different faces, e.g., a cleft chin for John Travolta and a mole near the lips for Cindy Crawford. Also, we expect to see these features at certain approximate locations within the face. We might have accumulated this knowledge from various human faces that we have seen before. This knowledge can be encoded as a function of features (position, appearance, etc.) of image patches that determines whether that patch would be useful or not for identifying a particular object. It is these features (representation of knowledge) that tell us about the likely utility of an image patch that Ferencz et al. call *hyper-features* [5].

Ferencz et al. [5] demonstrated the efficacy of the hyper-feature models for object identification. Their system was shown to outperform all other existing algorithms that they compared their results with, on this class of problems. However, they optimize the decision criterion indirectly by modeling the conditional distributions independently and not optimizing the log-likelihood ratio that is used for making a decision about match or mismatch. We propose a discriminative approach that optimizes the ratio of the posterior probabilities directly. Our experiments show marked improvements in accuracy over the existing generative models, for both the case in which entire images are used for classification and also for the case when only a subset of the most informative image patches are used for classification.

Most of the patch based identification methods [15, 9] model the distributions of appearances of different patches. This provides a generative framework for the image patches. Our approach is different from these techniques as we are modeling the patch differences conditioned on the patch appearances. Thus our approach is directly optimizing the criterion for identification. Moghaddam et al. [12] modeled the interpersonal and intrapersonal variations as fixed multivariate normal distributions. Our system improves on this approach by adapting these distributions according to individual faces. Cox et al. [3] addressed this by using a different parameter values for individual clusters of faces. For a new face image, the parameter values of the nearest cluster are chosen. This corresponds to piecewise constant parameter values as a function of the features, which is generalized by our system by providing a smooth interpolation over the entire feature space.

Huang and Russell [6] did a Bayesian analysis of object identification in the context of traffic surveillance. Their system required multiple images of a vehicle to build an appearance probability model for subsequent observations. As mentioned above, in a more general setting, we observe only a single image to build a model for future inferences. Learning from one example has also been explored in different contexts [11, 9]. In most of these approaches, off-line training involves parameter estimation for a fixed model. Our system, however, learns how to identify an arbitrary number of good features for the given category and thus use different set of patches for each object in the category. For face identification, the best performing PCA and LDA algorithms with face specific preprocessing match a face as a single object [2]. To obtain the required level of accuracies, a large number of principal components are usually required to approximate the underlying distribution of the face appearances. The hyper-features based approach was shown to outperform these systems in [5]. Our model shows a further improvement in performance.

Section 2 summarizes the hyper-feature model and different components of our system. In Section 3, we describe the criteria for selecting a few patches from the image for comparison to make the system real-time. Section 4 provides a detailed discussion of advantages of discriminative learning of hyper-feature models.

2 The hyper-feature model

Here, we provide an outline of the hyper-feature model originally proposed in [5]. We begin by describing the basic components of the system, followed by the generative model used for the identification task. We then present a new discriminative model that addresses the problem in a more direct way. In our discussion, we will refer to the query image as the left (probe) image, I^L, and the reference image in the database as the right (gallery) image, I^R.

We are using patch based features to represent an image. We encode each candidate patch of the left (probe) image, I^L, as a vector, F_j^L, of the directional derivatives in eight fixed directions. The choice of representation is, however, not critical in the current approach. Note that we sample patches at different scales and positions.

The images are assumed to be roughly registered. For every candidate patch (F_j^L), we find the most similar patch (F_j^R) in a small neighborhood around the expected location in the right (gallery) image, I^R. We use $d_j (= 1 - xcorr(F_j^L, F_j^R))$ as the distance measure between two image patches, where *xcorr* gives the normalized cross-correlation between the two image patches. We will refer to such a matched left and right patch pair (F_j^L, F_j^R) together with the derived distance d_j as a *bi-patch* F_j.

Hyper-features represent the characteristic properties of image patches that determine if a patch will be useful for identifying a particular object. We choose a set of base hyper-features as simple properties of the patch such as its location in the image, mean intensity and edge energy. To increase the flexibility in the model, we introduce the monomials (of degree 1, 2 and 3) of these base hyper-features into the set of possible hyper-features. This gives a large number of hyper-features which might be correlated. Using least angle regression (LARS) [4], we select a few($\sim$ 20) of these hyper-features as useful hyper-features. This reduces the complexity of our model and avoids possible over-fitting.

We decide if I^L and I^R are same using the rule

$$\frac{P(C=1|I^L, I^R)}{P(C=0|I^L, I^R)} > 1, \tag{1}$$

which is the optimal maximum a posteriori (MAP) classification criterion. Since we are treating each image as a set of m patches, the likelihoods and posteriors will be approximated using the bi-patches $F_1, ..., F_m$ as $P(C|I^L, I^R) \approx P(C|F_1, ..., F_m)$ and $P(I^L, I^R|C) \approx P(F_1, ..., F_m|C)$, where C is the match-mismatch variable.

2.1 The generative model

In the generative approach to this problem described in previous work, separate distributions are estimated from training data for pairs of cars that match and for pairs that do not match. These distributions are optimized separately and only later combined to produce decisions. We now describe the details of the generative model.

Using Bayes' rule, equation 1 can also be written as

$$\frac{P(I^L,I^R|C=1)P(C=1)}{P(I^L,I^R|C=0)P(C=0)} > 1 \quad \Rightarrow \quad \frac{P(I^L,I^R|C=1)}{P(I^L,I^R|C=0)} > \lambda, \tag{2}$$

where $\lambda = \frac{P(C=0)}{P(C=1)}$. Thus, by varying the values of this parameter λ for making a decision, we are essentially changing the ratio of priors. This formulation is used as the decision criterion for the generative model. Furthermore, we will assume a naïve Bayes model in which the bi-patches are independent of each other when conditioned on C:

$$\frac{P(I^L,I^R|C=1)}{P(I^L,I^R|C=0)} \quad \approx \quad \frac{P(F_1,...,F_m|C=1)}{P(F_1,...,F_m|C=0)} = \prod_{j=1}^{m} \frac{P(F_j|C=1)}{P(F_j|C=0)}. \tag{3}$$

Let h_j be the random variable representing the hyper-features of the left patch in the bi-patch F_j. Then we have

$$P(F_j|C) = P(d_j,h_j|C) = P(d_j|C,h_j)P(h_j|C) \quad \propto \quad P(d_j|C,h_j) \tag{4}$$

where Equation 4 is obtained by assuming the independence between h and C, which holds almost exactly in practice.

Ferencz et al. [5] use gamma distributions to model these $P(d|C,h)$ i.e.,

$$P(d|C=0;h) \sim \Gamma(\alpha_0(h),\theta_0(h)) \quad \texttt{and} \quad P(d|C=1;h) \sim \Gamma(\alpha_1(h),\theta_1(h)). \tag{5}$$

Here, a gamma distribution is parametrized by (α, θ) and h are the hyper-features of the given patch. These parameters, $\alpha_0,\alpha_1,\theta_0,\theta_1$, are modeled using a generalized linear model [10] fit over the training values as a function of selected hyper-features, h.

2.2 A discriminative model

In the above-mentioned generative model, we are modeling $P(d|C=0,h)$ and $P(d|C=1,h)$ independent of each other. Thus we are using an indirect optimization for the decision criterion (Equation 2). In this section, we use the MAP-optimal criterion (Equation 1) as the decision rule. We describe a discriminative model which estimates $P(C|d,h)$ and thus directly optimizes the decision rule, $\frac{P(C=1|d,h)}{1-P(C=1|d,h)} > 1$.

Logistic regression is a special generalized linear model suitable for modeling binary responses. It allows one to predict a discrete outcome from a set of variables that may be continuous, discrete, dichotomous, or a mix of any of these. In our model, C is the binary response which depends on (d,h). Thus, we build the following parametric model (sigmoid function):

$$P(C|d,h) = \frac{1}{1+e^{-X\beta}}, \tag{6}$$

where X is the vector representation of (d,h), also called the *predictor matrix*, and β is a vector of coefficients that we learn through logistic regression

$$log\left(\frac{P(C|d,h)}{1-P(C|d,h)}\right) = X\beta + \varepsilon. \tag{7}$$

Here ε is the error term having a binomial distribution. Note that we append a constant term to X to include an offset in the linear fit.

However, the estimate of the posterior probability that we obtained by using the predictor matrix, $X = (d,h)$, does not give us much flexibility to model $P(C|d,h)$. We are interested in obtaining good estimates of $P(C|d,h_0)$ when we observe a left patch having the hyper-feature values h_0. We want this curve to have sufficient flexibility to model the underlying variability. Any logistic curve can be specified by exactly two parameters, viz. location where the function takes value = 0.5 (say α_1) and its slope at that point (say α_2). Ideally, we would like both of these parameters to be dependent on h_0. Let us split β into three parts corresponding to the offset and distance, d, and hyper-features, h, as β_0, β_d and β_h respectively. Thus, $X\beta = \beta_0 + d\beta_d + h_0\beta_h$. It can be easily shown that

$$\alpha_1 = -\frac{\beta_0 + h_0\beta_h}{\beta_d}\ ,\ \alpha_2 = \frac{\beta_d}{4}. \tag{8}$$

Clearly, α_2 does not depend on h_0 when $X = (d,h)$. Hence, our estimates were not very good with this model.

In the generative model discussed in the previous section, we were making the parameters of the gamma distributions as linear combinations of the hyper-features. We can obtain a similar flexibility by making both α_1 and α_2 as linear combinations of the hyper-features. This can be attained by constructing the predictor matrix as $X = (d,h,dh)$.

In Figure 1, we show the estimates for the posterior probability obtained from actual training samples (dots at the top and bottom) by logistic regression with the predictor matrix containing $[1\ y\ y^2\ y^3]$, where y is the y-position of the center of the patch in the image.

3 Patch selection

Since the patches can occur anywhere in the scale-space [7] of the image, the set of possible patches is very large. To make this algorithm feasible for real-time applications, we should be able to evaluate an image match quickly by using only a few patches that were rated as most informative in a given image without sacrificing much accuracy. In other words, we want to choose the patches which contain the most information about the match-mismatch variable C. Let us define saliency of a patch as the amount of information gained if the patch were to be matched.

It is important to note that our algorithm selects these patches before seeing a potential match. Thus it selects these patches based only on their appearance and position in a single image (the left image in this case). We do this by estimating the mutual information between C and d as a function of h.

Intuitively, if $P(d|C = 0,h)$ and $P(d|C = 1,h)$ are similar distributions, we do not expect much useful information from a value of d. Formally, this can be measured as the mutual information between the patch dissimilarity d and the match-mismatch variable C given the hyper-feature value, h, i.e., $I(d;C|h)$ as:

$$I(d;C|h) = H(d|h) - H(d|C,h), \tag{9}$$

where $H(\cdot)$ is Shannon entropy and $P(d|h)$ can be estimated by adding the estimates for $P(d|C = 0,h)$ and $P(d|C = 1,h)$.

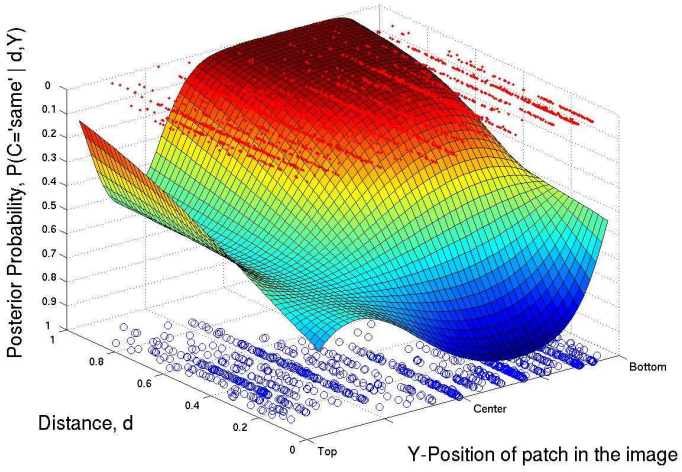

Figure 1: *Logistic regression based upon a single hyper-feature, the y-position*: The small points in the lower plane and the upper plane represent the pairs of training images for matched and mismatched cars respectively. Each point is plotted as a function of its match/mismatch label (C), the distance d between the patches, and a hyper-feature y, the y-position of the left patch of the patch pair. Notice that the points for matching cars (lower plane) which are in the bottom half of the original images have their d values clustered around zero. This is because d values tend to be low for patches near the bottom of the image when the cars match. On the other hand, for the same image position, the points representing mismatched cars have a more uniform distribution of d values. The goal of logistic regression is to approximate the original data points as well as possible while constraining each "slice" of the surface parallel to the d axis to be a logistic function. Furthermore, the parameters of the logistics at various y coordinates should be a smooth polynomial function of y. It is easy to see that the logistic surface "dips" to represent the low d values of the matching cars for patches in a particular y range.

Note that in a discriminative model, we do not have the estimates of $P(d|C,h)$ but have the estimates of $P(C|d,h)$. We can still estimate the mutual information, $I(d;C|h)$. [1] However, it is not clear which approach should be adopted for the patch selection as neither of them is actually optimizing the mutual information estimation. In our experiments, we use equation 9 for patch selection.

Using the estimates of mutual information, we can sort the image patches in non-increasing order and choose the top m patches. Here, we are assuming that the patches

[1]

$$I(d;C|h) = \sum_C \int_d P(d|h)P(C|d,h)\log\frac{P(C|d,h)}{P(C|h)}\,dd, \tag{10}$$

where $P(d|h)$ is estimated using histogram based approaches or kernel density estimation.

	40%	60%	80%
Bayesian ML	74.6± 7.83	60.5±8.38	54.8 ± 2.91
Bayesian MAP	74.8± 9.09	59.9± 8.59	54.3 ± 6.15
Generative	81.2 ± 6.35	63.4± 6.71	54.4 ± 6.37
Discriminative	**93.0 ± 6.29**	**78.9 ± 8.15**	**60.1 ± 6.97**

Table 1: Precision values at 40%, 60% and 80% recall for 10-fold cross-validation on the faces data set containing 500 pairs each of "same" and "different" faces.

are independent, which is a serious limitation. However, it has been shown by Ferencz et al. [5] that modeling pairwise relationships between patches does not improve the results drastically. Thus, for our comparisons, ignoring the pairwise dependencies between patches does not affect our conclusion.

4 Results and discussion

For the face recognition task, Ferencz et al. [5] has outperformed the standard techniques like $PCA + MahCosine$ and $Filter + NormCor$. $PCA + MahCosine$ is the best curve produced by [2]. Through personal communications, Ferencz et al. asserted that their approach also beats local feature based techniques like SIFT [8], which is not designed for problems like object identification within a class, by a wide margin. A more sophisticated technique for face identification is Baysian face recognition [12], which was the top performer in the FERET face recognition competition, beating the above techniques described in [2]. Thus we chose to directly compare our technique with Ferencz et al. [5] and Bayesian face recognition [12]. Although we have not performed an exhaustive comparison with all the published face identification algorithms, the advantage of our method is clear from the wide margin with which we beat both of these leading techniques. Also note that due to the patch selection component, we are able to achieve acceptable performance using a small number of patches which makes it feasible for real-time applications.

As discussed in Section 3, there is no clear choice for a patch selection approach. In our experiments, we separated the two stages, patch difference modeling and patch selection, so that we can draw informative conclusions.

We compared the discriminative and generative approaches to modeling patch differences on a subset of the "Faces in the news" data set [1]. These faces are automatically extracted from news articles and aligned to a frontal pose. This is a difficult data set because of the large variations in lighting, background, facial expression and other factors. The generative model was shown by Ferencz et al. [5] to perform better than the PCA and LDA based algorithms with face specific preprocessing using CSU's evaluation system [2]. Figure 2 shows a big improvement of our own discriminative model over the previous model. In Figure 2, we show that our approach beats another state of the art approach, Bayesian face recognition [12], as well. Table 1 shows the comparison of precision values at different recall values for 10-fold cross validation on the faces data set. The gain is significant for a range of recall values (though not for all), and the boost in performance is clearly evident. Some pairs of face images that were correctly identified as "same" are shown in Figure 2.

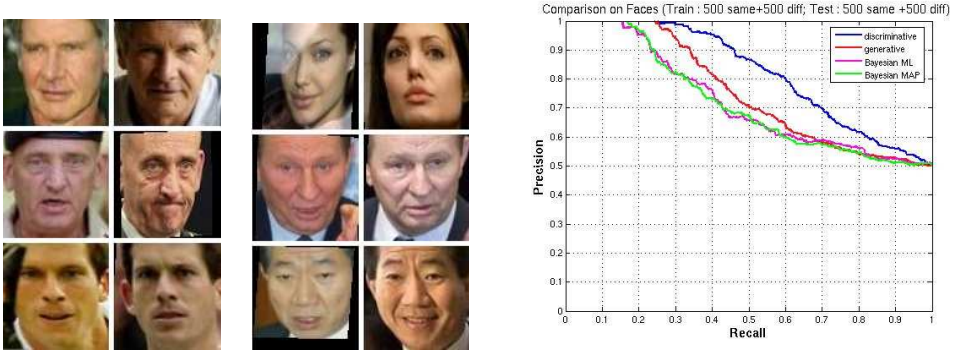

Figure 2: *Results on face data set:* **[Left]** These are some pairs of face images that are correctly marked as "same". There is a large variation in illumination, expression and background. The variation in pose has been countered by aligning the face images to make it approximately frontal. **[Right]** Both discriminative (blue) and generative (red) models are trained for 500 pairs each of "same" and "different" faces. The test set contains 500 pairs of "same" and "different" faces of people which are not in the train set. The patches are selected using the approach discussed in Section 3 in both the models. The boost in performance is large over a wide range of recall values. Note that our results outperform Bayesian face recognition [12] that was the best performer on FERET data set.

To demonstrate that our approach performs well on different object categories, we also ran some experiments on the car data set used by Ferencz et al. [5] in their experiments. In Figure 3, we show a comparison between the discriminative and the generative approach on the car data set.[2]

To directly compare the two patch difference modeling approaches, we compared the discriminative and generative models using the same patch selection criterion (Section 3). As shown in Figure 3, the discriminative method is uniformly better than the generative model.

Note that even with the selection of a few (20) patches, we do not observe a significant drop in performance because the top patches contain almost all the discriminative information. Another important observation is that even though the patches are selected through an approach that uses the estimations of quantities that are optimized in a generative fashion, the discriminative model beats the generative model in making the decision for match or mismatch. This is due to the fact that patch selection and match evaluation are decoupled from each other. Figure 4 show some identification results obtained by our system on the car data set.

As is evident in our experiments, the discriminative model outperforms the generative model for this task. This supports our hypothesis about the advantages of doing a direct optimization of posterior probabilities.

Recently in computer vision and machine learning, there has been a great deal of analysis and discussion about the relative strengths and weaknesses of generative and

[2]These results are not directly comparable to the published results in [5] as the training and testing set are different in the two cases.

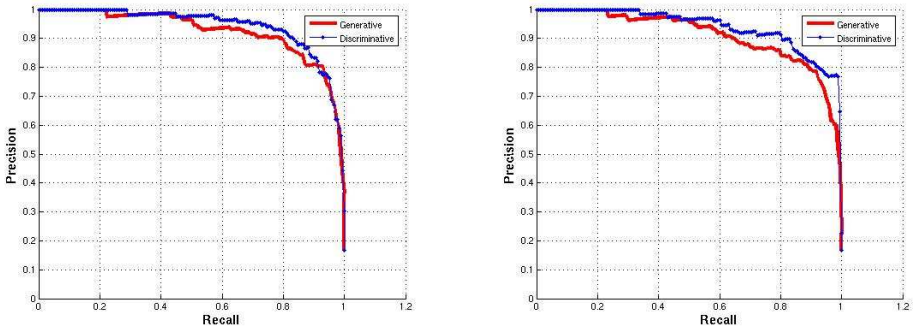

Figure 3: *Comparing performance of discriminative (blue) with generative (red) model on car data set.* Both models are trained for 178 different vehicles, each having one "same" and five "different" training instances. The trained models are then tested on 170 other vehicles. The test set has the same ratio of "same" to "different" pairs of car images. **[Left]** *Using all the patches*: The blue curve clearly shows a better performance than the red curve. The red curves overtakes the blue curve for a small interval, but the overall area under the P-R curve is more for the blue curve. **[Right]** *With patch selection*: We use the same patch selection method for the two models. The discriminative model is uniformly better than the generative model.

discriminative models (see, for example, [14, 13]). Ulusoy and Bishop [14] enumerate some of these strengths and weaknesses, and among other things conclude that "Other things being equal, it would be expected that discriminative methods would have better predictive performance since they are trained to predict the class label rather than the joint distribution of input vectors and targets."

It is interesting to note that Ng and Jordan [13] conclude that while discriminative models may converge to better solutions for large enough data sets, they suggest that generative models may perform better in some cases when data sets are small. This conclusion, however, is based upon an analysis of training discriminative classifiers with 0-1 loss, rather than with something like true logistic regression, in which a data point has a value that depends upon how far it is from the decision boundary. It is not clear what the conclusion should be for a discriminative model like our own which uses classical logistic regression, but it was our hypothesis that it would produce better results, which in fact it has.

References

[1] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y. Teh, E. Learned-Miller, and D. A. Forsyth. Names and faces in the news. In *CVPR(2)*, pages 848–854, 2004.

[2] D S. Bolme, J. Ross Beveridge, M. Teixeira, and B. A. Draper. The csu face identification evaluation system: Its purpose, features, and structure. In *ICVS*, 2003.

Figure 4: *Results on car data set:* The first two columns show three pairs of cars that are identified as "same" by our algorithm. The last two columns show three pairs of cars that are marked as "different" by our algorithm. The camera angle and illumination for the two images in each pair are clearly different. Note that there are distortions introduced in the images in the process of aligning the car images.

[3] I. J. Cox, J. Ghosn, and P. N. Yianilos. Feature-based face recognition using mixture-distance. In *CVPR*, pages 209–216. IEEE Press, 1996.

[4] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression, 2002.

[5] A. Ferencz, E. Learned-Miller, and J. Malik. Building a classification cascade for visual identification from one example. In *ICCV*, 2005.

[6] T. Huang and S. J. Russell. Object identification: A bayesian analysis with application to traffic surveillance. *Artificial Intelligence*, 103(1-2):77–93, 1998.

[7] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, Norwell, MA, USA, 1994.

[8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[9] M. Welling M. Weber and P. Perona. Unsupervised learning of models for recognition. In *ECCV (1)*, pages 18–32, 2000.

[10] P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman and Hall, 1989.

[11] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. In *CVPR*, pages 1464–1471, 2000.

[12] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian face recognition. *Pattern Recognition*, 33:1771–1782, November 2000.

[13] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *NIPS*, pages 841–848, 2001.

[14] I. Ulusoy and C. M. Bishop. Generative versus discriminative methods for object recognition. In *CVPR (2)*, pages 258–265, 2005.

[15] M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *ICCV*, pages 281–288, 2003.