

Semi-Supervised Face Detection

Nicu Sebe¹, Ira Cohen², Thomas S. Huang³, Theo Gevers¹

¹ Faculty of Science, University of Amsterdam, The Netherlands

² HP Research Labs, USA

³ Beckman Institute, University of Illinois at Urbana-Champaign, USA

Abstract

This paper presents a discussion on semi-supervised learning of probabilistic mixture model classifiers for face detection. We present a theoretical analysis of semi-supervised learning and show that there is an overlooked fundamental difference between the purely supervised and the semi-supervised learning paradigms. While in the supervised case, increasing the amount of labeled training data is always seen as a way to improve the classifier's performance, the converse might also be true as the number of unlabeled data is increased in the semi-supervised case. We also study the impact of this theoretical finding on Bayesian network classifiers, with the goal of avoiding the performance degradation with unlabeled data. We apply the semi-supervised approach to face detection and we show that learning the structure of Bayesian network classifiers enables learning good classifiers for face detection with a small labeled set and a large unlabeled set.

1. Introduction

The rapidly expanding research in face processing is based on the premise that information about user's identity, state, and intent can be extracted from images and that computers can react accordingly, e.g., by observing a person's facial expression. Given an arbitrary image, the goal of face detection is to automatically locate a human face in an image or video, if it is present. Face detection in a general setting is a challenging problem for various reasons. The first set of reasons is inherent: there are many types of faces, with different colors, texture, sizes, etc. In addition, the face is a non-rigid object which can change its appearance. The second set of reasons is environmental: changing lighting, rotations, translations, and scales of the faces in natural images. To solve the problem of face detection, two main approaches can be taken. The first is a model based approach, where a description of what is a human face is used for detection. The second is an appearance based approach, where we learn what are faces directly from their appearance in images. In this work we focus on the latter.

There have been numerous appearance based approaches. We list a few from recent years and refer to the reviews [29] and [17] for further details. Rowley et al. [23] and Kouzani [18] used Neural networks to detect faces in images by training from a corpus of face and non-face images. Colmenarez and Huang [6] used maximum entropic discrimination between faces and non-faces to perform maximum likelihood classification, which was used for a real time face tracking system. Yang et al. [30] used SNoW based classifiers to learn the face and non-face discrimination boundary on natural face images. Others used support vector machines [16]. Wang et al. [28] learned a minimum spanning weighted tree for learning pairwise dependencies graphs of facial pixels, followed by a discriminant projection to reduce complexity. Viola and Jones [27] used boosting and a cascade of classifiers for face detection.

Face detection provides interesting challenges to the underlying pattern classification and learning techniques. When a raw or filtered image is considered as input to a pattern classifier, the dimension of the space is extremely large (i.e., the number of pixels in normalized training images). The classes of face and non-face images are decidedly characterized by multimodal distribution functions and effective decision boundaries are likely to be non-linear in the image space. To be effective, the classifiers must be able to extrapolate from a modest number of training samples. Another challenge is the relatively small amount of available labeled data. Constructing and labeling a good database requires time and effort. However, collecting unlabeled data is not as difficult. It is therefore desirable to use classifiers that can be learnt with a combination of labeled data and a large amount of unlabeled data. This learning paradigm is known as *semi-supervised learning*.

Is there value to unlabeled data in supervised learning of classifiers? This fundamental question has been increasingly discussed in recent years, with a general optimistic view that unlabeled data hold great value. Due to an increasing number of applications and algorithms that successfully use unlabeled data [2, 5, 14, 20, 26] and magnified by theoretical issues over the value of unlabeled data in certain cases [3, 8, 21, 25], semi-supervised learning is seen

optimistically as a learning paradigm that can relieve the practitioner from the need to collect many expensive labeled training data. However, several disparate empirical evidences in the literature suggest that there are situations in which the addition of unlabeled data to a pool of labeled data causes degradation of the performance [2, 20, 26], in contrast to improvement of performance when adding more labeled data.

Very relevant to our work is the research of Baluja [2]. The author uses labeled and unlabeled data in a probabilistic classifier framework to detect the orientation of a face. He obtained excellent classification results, but there were cases where unlabeled data degraded performance. As a consequence, he decided to switch from a Naive Bayes approach to more complex models. Following this intuitive direction, we explain Baluja’s observations and provide a solution to the problem: structure learning. Another relevant work is the research of Schneiderman [24] who learns a sparse structure of statistical dependencies for several object classes including faces. While analyzing such dependencies can reveal useful information, we go beyond the scope of Schneiderman’s work and present a framework that not only learns the structure of a face but also allows the use of unlabeled data in classification.

This work is inspired by our previous research [5] which focussed on the machine learning aspects of semi-supervised learning with extensive discussion and experiments on the value of unlabeled data and on the conditions when the semi-supervised learning should be applied in computer vision. The main contribution of this paper is demonstrating the ability of Bayesian network classifiers to learn appearance based face models for face detection using both labeled and unlabeled data. As far as we are aware, this is the first complete analysis of using Bayesian Networks for learning the structure of a face using the semi-supervised approach. Note that the main goal of this paper is not to present a complete face detection system. We limit ourselves to show a face detection methodology that can use both labeled and unlabeled data and which can easily be applied to other face detection methods.

2. Semi-Supervised Learning for Mixture Models

The goal is to classify an incoming vector of observables \mathbf{X} . Each instantiation of \mathbf{X} is a *sample*. There exists a *class variable* C ; the values of C are the *classes*. Note that in our face detection approach, \mathbf{X} stands for the image pixels used as features for the classifier and we use two classes in C : face and non-face.

We want to build *classifiers* that receive a sample \mathbf{x} and output either one of the values of C . We assume 0-1 loss, and consequently our objective is to minimize the proba-

bility of classification error. If we knew exactly the joint distribution $p(C, \mathbf{X})$, the optimal rule would be to choose the class value with the maximum a-posteriori probability, $p(C|\mathbf{x})$ [9]. This classification rule attains the minimum possible classification error, called the *Bayes error*.

We take that the probabilities of (C, \mathbf{X}) , or functions of these probabilities, are estimated from data and then “plugged” into the optimal classification rule. We assume that a parametric model $p(C, \mathbf{X}|\theta)$ is adopted. An estimate of θ is denoted by $\hat{\theta}$. If the distribution $p(C, \mathbf{X})$ belongs to the family $p(C, \mathbf{X}|\theta)$, we say the “model is correct”, otherwise the “model is incorrect.” When the model is correct, the difference between the expected value $E_{\theta}[\hat{\theta}]$ and θ , is called *estimation bias*. When the model is incorrect, it is generally not meaningful to refer to the “true” θ and we use “bias” loosely to mean the difference between $p(C, \mathbf{X})$ and the estimated $p(C, \mathbf{X}|\hat{\theta})$.

We consider the following scenario. A sample (c, \mathbf{x}) is generated from $p(C, \mathbf{X})$. The value c is then either revealed, and the sample is a *labeled* one; or the value c is hidden, and the sample is an *unlabeled* one. The probability that any sample is labeled, denoted by λ , is fixed, known, and independent of the samples¹. Thus, the same underlying distribution $p(C, \mathbf{X})$ models both labeled and unlabeled data. Given a set of N_l labeled samples and N_u unlabeled samples, we use maximum likelihood for estimating $\hat{\theta}$. The assumed distribution $p(C, \mathbf{X}|\theta)$ can be decomposed either as $p(C|\mathbf{X}, \theta) p(\mathbf{X}|\theta)$ or as $p(\mathbf{X}|C, \theta) p(C|\theta)$. A parametric model where both $p(\mathbf{X}|C, \theta)$ and $p(C|\theta)$ depend explicitly on θ is referred to as a *generative model*. The log-likelihood function of a generative model for a dataset with labeled and unlabeled data is:

$$\begin{aligned} L(\theta) &= L_l(\theta) + L_u(\theta) + \log \left(\lambda^{N_l} (1 - \lambda)^{N_u} \right) \quad (1) \\ L_l(\theta) &= \sum_{i=1}^{N_l} \log \left[\prod_C (p(C = c'|\theta) p(\mathbf{x}_i|c', \theta))^{I_{\{C=c'\}}(c_i)} \right] \\ L_u(\theta) &= \sum_{j=(N_l+1)}^{N_l+N_u} \log \left[\sum_C p(C = c'|\theta) p(\mathbf{x}_j|c', \theta) \right] \\ &= \sum_{j=(N_l+1)}^{N_l+N_u} \log [p(\mathbf{x}_j|\theta)], \end{aligned}$$

where $I_A(Z)$ is the indicator function (1 if $Z \in A$; 0 otherwise) and $p(C = c')$ are the mixing coefficients. $L_l(\theta)$ and $L_u(\theta)$ are the likelihoods of the labeled and unlabeled data, respectively.

It would perhaps be reasonable to expect an average improvement in classification performance for any increase in the number of samples (labeled or unlabeled): the more

¹This is different from [7] where λ is a parameter that can be set.

data, the better. In fact, it would seem that any increase in the number of samples should contribute to a reduction in the variance of $\hat{\theta}$, and a smaller variance should be beneficial to classification - this intuitive reasoning suggests that unlabeled data must be used whenever available. Indeed, previous theoretical work [3] showed that unlabeled data are always asymptotically useful for classification. However, there is an assumption that the model is correct. In [5] we provide a full analysis which describes the general case and shows what happens when the model is incorrect. The main conclusions are summarized below:

- Labeled and unlabeled data contribute to a reduction in variance in semi-supervised learning under maximum likelihood estimation. *This is true regardless of whether the model is correct or not.*
- If the model is correct, the maximum likelihood estimator is unbiased and both labeled and unlabeled data contribute to a reduction in classification error by reducing variance.
- If the model is incorrect, there may be different asymptotic estimation bias for different values of λ (the ratio between the number of labeled and unlabeled data). Asymptotic classification error may also be different for different values of λ . An increase in the number of unlabeled samples may lead to a larger bias from the true distribution and a larger classification error.

This asymptotic analysis shows the importance of modeling assumption, but how do we then account for the success of other researchers in applications such as text classification [20], image understanding [2], and many others? There are two possibilities. First, it might be that the assumed model was truly the correct model. Alternatively, a more plausible explanation is that of bias vs. variance.

Consider the application of classifying face orientation [2]. The problem involves many observables (image pixels) with a small corpus of labeled data. From our theoretical analysis we know that regardless of modeling assumptions, the addition of unlabeled data decreases the variance of the estimator, while when the model is incorrect, the estimation bias can increase. Classification error with finite training data is a function of both the bias and the variance [11]. Therefore, when the amount of labeled data is small, the increase in bias caused by the unlabeled data is mitigated by the decrease in variance, hence causing an improvement in classification performance. This agrees with the conclusions of [26] who indicated that unlabeled data become more useful as the number of observables increases. Examples shown in [5] using artificially generated data further illustrate this explanation.

In closing, barring numerical instabilities, unlabeled data could appear to improve classification performance, even

with incorrect modeling assumptions, when the number of labeled data is small compared to the complexity of the classifier. An important question is, can we avoid the increase in bias while not greatly increasing the variance? We discuss this question in the following section.

3. Dealing with Performance Degradation for Bayesian Networks

The goal of the following discussion is to provide possible solutions for performance degradation in the framework of Bayesian network classifiers. It is our hope that such solutions form a guide to other types of classifiers. One of the main advantages of Bayesian networks is the ability to handle missing data which makes them the ideal tools for handling unlabeled data in learning.

Following the definitions of correct and incorrect models described in the previous section, we say that the assumed structure for a network, S' , is *correct* when it is possible to find a distribution, $p(C, \mathbf{X}|S')$, that matches the distribution that generates data, $p(C, \mathbf{X})$; otherwise, the structure is *incorrect*. Also, as a direct consequence of the analysis in Section 2, a Bayesian network that has the correct structure and the correct parameters is optimal for classification because the a-posteriori distribution of the class variable is accurately represented. Therefore, to solve the problem of performance degradation in BNs, we need to take a careful look at the assumed structure of the classifier (an extensive discussion is presented in [5]).

To preserve the balance between the bias from the true distribution and the variance we first consider the use of a small subset of simple models which can be learned efficiently. Two such examples are the Naive Bayes classifier, in which the features are assumed independent given the class and the Tree-Augmented Naive Bayes classifier (TAN) [13]. In the TAN's structure the class node has no parents and each feature has the class node and at most one other feature as parents, such that the result is a tree structure for the features (e.g., face pixels). Given a labeled and unlabeled dataset, we could start with a Naive Bayes classifier and, if performance degrades with unlabeled data, switch to the more complicated TAN classifier. If the correct structure can be represented using a TAN structure, this approach is assured to work. However, TAN structures are still a limited set of possible structures and switching to TAN might not always work.

A different approach to overcome performance degradation is to learn the structure of the Bayesian network without restrictions other than the generative one. There are a number of such algorithms in the literature [4, 12]. Nearly all structure learning algorithms use the 'likelihood based' approach [4]. The goal is to find structures that best fit the data (with perhaps a prior distribution over different structures).

Since more complicated structures have higher likelihood scores, penalizing terms are added to avoid overfitting to the data, e.g, the minimum description length (MDL) term.

Likelihood based structure learning approaches have been criticized when learning classifiers. With finite amounts of data, such approaches can lead to poor classifiers because the a-posteriori probability of the class variable could have a small effect on the score of a structure [13, 15]. Therefore, a network with a higher likelihood based score is not necessarily a better classifier. With unlabeled data, this problem could further be magnified since the likelihood of the unlabeled data is increased for structures that fit the marginal of the features and not the a-posteriori probability (see Eq.(1)).

To solve this problem, Friedman et al. [13] suggested maximizing the a-posterior probability of the class variable, but show that it is not computationally feasible. Greiner and Zhou [15] prove that even the problem of learning the parameters of a Bayesian network, without having to change the structure, and maximizing the likelihood of the class a-posteriori probability is NP-hard. They resort to gradient algorithms to attempt to learn the parameters so as to maximize the likelihood of the a-posteriori probability.

In our analysis [5], we took a different approach. Instead of trying to estimate the best a-posteriori probability, we try to find the structure that minimizes the probability of classification error directly. To do so we designed a classification driven stochastic search algorithm (SSS). We defined first a measure over the space of structures which we want to maximize:

Definition 1 *The inverse error measure for structure S' is*

$$inv_e(S') = \frac{\frac{1}{p_{S'}(\hat{c}(X) \neq C)}}{\sum_S \frac{1}{p_S(\hat{c}(X) \neq C)}}, \quad (2)$$

where the summation is over the space of possible structures and $p_S(\hat{c}(X) \neq C)$ is the probability of error of the best classifier learned with structure S .

We use Metropolis sampling to generate samples from the inverse error measure, without having to ever compute it for all possible structures. We estimate the classification error of a given structure using the labeled training data. Therefore, to avoid overfitting, we add a multiplicative penalty term derived from the Vapnik- Chervonenkis (VC) bound on the empirical classification error. This penalty term penalizes complex classifiers thus keeping the balance between bias and variance. In our experiments, we found that the multiplicative penalty outperformed the MDL and BIC complexity measures.

The advantages of the SSS algorithm are that it usually converges to better classifiers compared to the other methods, and asymptotically can be shown to converge to the classifier with minimum error. Its biggest disadvantage is

in the added complexity: for every structure being tested the parameters are estimated, followed by error estimation.

4. Semi-supervised Face Detection

In our face detection experiments we propose to use Bayesian network classifiers, with the image pixels of a pre-defined window size as the features in the Bayesian network. Among the different works, those of Colmenarez and Huang [6] and Wang et al. [28] are more related to the Bayesian network classification methods for face detection. Both learn some ‘structure’ between the facial pixels and combine them to a probabilistic classification rule. Both use the entropy between the different pixels to learn pairwise dependencies.

Our approach in detecting faces is an appearance based approach, where the intensity of image pixels serve as the features for the classifier. In a natural image, faces can appear at different scales, rotations, and location. For learning and defining the Bayesian network classifiers, we must look at fixed size windows and learn how a face appears in such windows, where we assume that the face appears in most of the window’s pixels. The goal of the classifier is to determine if the pixels in a window are those of a face or non-face. While faces are a well defined concept, and have a relatively regular appearance, it is harder to characterize non-faces. We therefore model the pixel intensities as discrete random variables, as it would be impossible to define a parametric probability distribution function for non-face images. For 8-bit representation of pixel intensity, each pixel has 256 values. Clearly, if all these values are used for the classifier, the number of parameters of the joint distribution is too large for learning dependencies between the pixels (as is the case of TAN classifiers). Therefore, there is a need to reduce the number of values representing pixel intensity. Colmenarez and Huang [6] used 4 values per pixel using fixed and equal bin sizes. We use non-uniform discretization using the class conditional entropy as the mean to bin the 256 values to a smaller number. We use the MLC++ software for that purpose as described in [10].

Our methodology can be extended to other face detection methods which use different features. The complexity of our method is $O(n)$, where n is the number of features (pixels in our case) considered in each image window.

4.1 Experimental Analysis

We test the different approaches described in Section 3, with both labeled and unlabeled data. For training the classifier we used a dataset consisting of 2,429 faces and 10,000 non-faces obtained from the MIT CBCL Face database #1 [1] (examples of face images from the database are presented in Figure 1). Each face image is cropped and resampled

to a 19x19 window, thus we have a classifier with 361 features. We also randomly rotate and translate the face images to create a training set of 10,000 face images. In addition, we have available 10,000 non-face images. We leave out 1,000 images (faces and non-faces) for testing and train the Bayesian network classifiers on the remaining 19,000. In all the experiments we learn a Naive Bayes, TAN, and a general generative Bayesian network classifier, the latter using and the SSS algorithm.

To compare the results of the classifiers, we use the receiving operating characteristic (ROC) curves. The ROC curves show, under different classification thresholds, ranging from 0 to 1, the probability of detecting a face in a face image, $P_D = P(\hat{C} = face | C = face)$, against the probability of falsely detecting a face in a non-face image, $P_{FD} = P(\hat{C} = face | C \neq face)$.



Figure 1: Randomly selected face examples

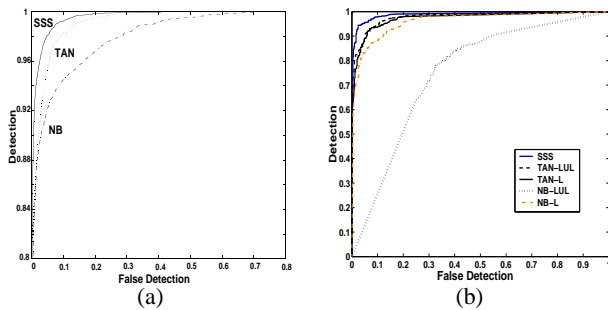


Figure 2: ROC curves showing detection rates of faces compared to false detection of faces for different classifiers when with (a) all the data are labeled and (b) 97.5% are unlabeled data.

We first learn using all the training data being labeled (19,000 labeled images). Fig. 2(a) shows the resultant ROC curve for this case. The classifier learned with the SSS algorithm outperforms both TAN and NB classifiers achieving about 98% detection rates with a low rate of false alarm.

Next we remove the labels of 97.5% of the training data (leaving only 475 labeled images) and train the classifiers (Figure 2(b)). We see that the NB classifier using both labeled and unlabeled data (NB-LUL) performs very poorly. The TAN based only on the 475 labeled images (TAN-L) and the TAN based on the labeled and unlabeled images (TAN-LUL) are close in performance, thus there was no significant degradation of performance when adding the unlabeled data. The classifier trained with SSS provides the best results.

In Table 1 we summarize the results obtained for different algorithms and in the presence of increasing number of unlabeled data. We fixed the false alarm to 1%, 5%, and 10% and we computed the detection rates. Note that the detection rates for NB are lower than the ones obtained for the other detectors. The SVM classifier (we used the implementation of Osuna et al. [22]) starts off very good, but does not improve performance. Overall, the results obtained with SSS are the best. We see that even in the most difficult cases, there was sufficient amount of unlabeled data to achieve almost the same performance as with a large sized labeled dataset.

Table 1: Detection rates (%) for various numbers of false positives

Detector		False positives		
		1%	5%	10%
NB	19,000 label	74.31	89.21	92.72
	475 label	68.37	86.55	89.45
	475 label + 18,525 unlabel	66.05	85.73	86.98
	250 label	65.59	84.13	87.67
	250 label + 18,750 unlabel	65.15	83.81	86.07
TAN	19,000 label	91.82	96.42	99.11
	475 label	86.59	90.84	94.67
	475 label + 18,525 unlabel	85.77	90.87	94.21
	250 label	75.37	87.97	92.56
	250 label + 18,750 unlabel	77.19	89.08	91.42
SSS	19,000 label	90.27	98.26	99.87
	475 label + 18,525 unlabel	88.66	96.89	98.77
	250 label + 18,750 unlabel	86.64	95.29	97.93
SVM	19,000 label	87.78	93.84	94.14
	475 label	82.61	89.66	91.12
	250 label	77.64	87.17	89.16

We also tested our system on the CMU test set [23] consisting of 130 images with a total of 507 frontal faces. The results are summarized in Table 2. Note that we obtained comparable results with the results obtained by Viola and Jones [27] and better than the results of Rowley et al. [23]. Two examples of the detection results on some of the images of the CMU test are presented in Figure 3. We noticed



Figure 3: Output of the system on some images of the CMU test set using the SSS classifier learned with 19,000 labeled data.

Table 2: Detection rates (%) for various numbers of false positives on the CMU test set.

Detector		False positives	
		10%	20%
SSS	19,000 label	91.7	92.84
	475 label + 18,525 unlabel	89.67	91.03
	250 label + 18,750 unlabel	86.64	89.17
Viola-Jones [27]		92.1	93.2
Rowley et al. [23]		-	89.2

similar failure modes as Viola and Jones [27]. Since, the face detector was trained only on frontal faces our system fails to detect faces if they have a significant rotation out of the plane (toward a profile view). The detector has also problems with the images in which the faces appear dark and the background is relatively light. Inevitably, we also detect false positive especially in some texture regions.

5. Summary and Discussion

In this paper, we suggested a methodology for learning to detect faces using both labeled and unlabeled data samples. We presented an analysis of semi-supervised learning which emphasizes the need to learn models faithful to the data generating distribution when learning with unlabeled data. In a nutshell, when faced with the option of learning with labeled and unlabeled data for face detection using Bayesian networks, our discussion suggests using the following path. Start with Naive Bayes and TAN classifiers, learn only with the available labeled data, and test whether the model is correct by learning with the unlabeled data. If the result is not satisfactory, then SSS can be used to attempt to further improve performance. If none of the methods using the unlabeled data improve performance over the supervised TAN (or Naive Bayes) the practitioner is faced with two options: discard the unlabeled data, or label some of the unlabeled data using the active learning methodology. Of course, active learning can be used as long as there are resources to

label some samples.

It is only fair to ask whether other semi-supervised learning methods, such as transductive SVM, co-training, and active learning will exhibit the phenomenon of performance degradation as mixture models learned with ML estimators. While extensive studies have not been performed, a few results from the literature suggest that it is a realistic conjecture. Zhang and Oles [31] demonstrated that transductive SVM can cause degradation of performance when unlabeled data are added. Ghani [14] described experiments where the same phenomenon occurred with co-training. In active learning [19], unlabeled data are added to a pool of labeled data by querying the user for the true label. Choosing the data to be labeled by queries is done so as to minimize the estimation variance of the classifier and there is an assumption that the bias is small (i.e., the model is almost correct). When this assumption is violated, there is no reason to believe that the performance degradation will not occur. However, we have seen that sometimes adding a small number of additional labeled data does improve classification accuracy dramatically, therefore methods that use the active learning strategy still hold great promise.

In closing, it is possible to view some of the components of this work independently of each other. The theoretical results of Section 2 do not depend on the choice of probabilistic classifier and can be used as a guide to other classifiers. Structure learning of Bayesian networks is not a topic motivated solely by the use of unlabeled data. Face detection could be solved using classifiers other than BNs. However, this work should be viewed as a combination of all three components; (1) the theory showing the limitations of unlabeled data is used to motivate (2) the design of algorithms to search for better performing structures of Bayesian networks and finally, (3) the successful application to face detection by learning with labeled and unlabeled data.

References

- [1] CBCL Face Database #1. MIT Center For Biological and Computation Learning.
- [2] S. Baluja. Probabilistic modeling for face orientation discrimination: Learning from labeled and unlabeled data. In *NIPS*, pages 854–860, 1998.
- [3] V. Castelli. *The relative value of labeled and unlabeled samples in pattern recognition*. PhD thesis, Stanford, 1994.
- [4] J. Cheng, R. Greiner, J. Kelly, D. Bell, and W. Liu. Learning Bayesian networks from data: An information-theory based approach. *The Artificial Intell. Journal*, 137:43–90, 2002.

- [5] I. Cohen, F. Cozman, N. Sebe, M. Cirello, and T.S. Huang. Semi-supervised learning of classifiers: Theory, algorithms, and their applications to human-computer interaction. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(12):1553–1567, 2004.
- [6] A.J. Colmenarez and T.S. Huang. Face detection with information based maximum discrimination. In *CVPR*, pages 782–787, 1997.
- [7] A. Corduneanu and T. Jaakkola. Continuations methods for mixing heterogeneous sources. In *UAI*, pages 111–118, 2002.
- [8] F. Cozman, I. Cohen, and M. Cirelo. Semi-supervised learning of mixture models. In *International Conference on Machine Learning*, pages 99–106, 2003.
- [9] L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*. Springer Verlag, 1996.
- [10] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *ICML*, pages 194–202, 1995.
- [11] J.H. Friedman. On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77, 1997.
- [12] N. Friedman. The Bayesian structural EM algorithm. In *UAI*, pages 129–138, 1998.
- [13] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2):131–163, 1997.
- [14] R. Ghani. Combining labeled and unlabeled data for multiclass text categorization. In *ICML*, pages 187–194, 2002.
- [15] R. Greiner and W. Zhou. Structural extension to logistic regression: Discriminative parameter learning of belief net classifiers. In *UAI*, pages 167–173, 2002.
- [16] B. Heisele, P. Ho, J. Wu, and T. Poggio. Face recognition: Component-based versus global approaches. *Computer Vision and Image Understanding*, 91(1-2):6–21, 2003.
- [17] E. Hjelm and B. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.
- [18] A.Z. Kouzani. Locating human faces within images. *Computer Vision and Image Understanding*, 91(3):247–279, 2003.
- [19] A.K. McCallum and K. Nigam. Employing EM and pool-based active learning for text classification. In *ICML*, pages 359–367, 1998.
- [20] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2-3):103–134, 2000.
- [21] T.J. O’Neill. Normal discrimination with unclassified observations. *Journal of the American Statistical Association*, 73(364):821–826, 1978.
- [22] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. In *CVPR*, pages 130–136, 1997.
- [23] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Patt. Anal. Mach. Intell.*, 20(1):23–38, 1998.
- [24] H. Schneiderman. Learning a restricted Bayesian network for object detection. In *CVPR*, pages 639–646, 2004.
- [25] M. Seeger. Learning with labeled and unlabeled data. In *TR., Univ. of Edinburgh*, 2002.
- [26] B. Shahshahani and D. Landgrebe. Effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. *IEEE Trans. Geoscience and Remote Sensing*, 32(5):1087–1095, 1994.
- [27] P. Viola and M. Jones. Robust real-time object detection. *IJCV*, 57(2):137–154, 2004.
- [28] R.R. Wang, T.S. Huang, and J. Zhong. Generative and discriminative face modeling for detection. In *International Conference on Face and Gesture Recognition*, 2002.
- [29] M.-H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trans. Patt. Anal. Mach. Intell.*, 24(1):34–58, 2002.
- [30] M.-H. Yang, D. Roth, and N. Ahuja. A SNoW based face detector. In *NIPS*, pages 855–861, 2000.
- [31] T. Zhang and F. Oles. A probability analysis on the value of unlabeled data for classification problems. In *ICML*, pages 1191–1198, 2000.