To appear in CAIP, 2003

Learning Statistical Structure for Object Detection

Henry Schneiderman

Robotics Institute, Carnegie Mellon University
Pittsburgh, PA 15213, USA
hws@cs.cmu.edu
http://www.cs.cmu.edu/~hws/index.html

Abstract. Many classes of images exhibit sparse structuring of statistical dependency. Each variable has strong statistical dependency with a small number of other variables and negligible dependency with the remaining ones. Such structuring makes it possible to construct a powerful classifier by only representing the stronger dependencies among the variables. In particular, a semi-naïve Bayes classifier compactly represents sparseness. A semi-naïve Bayes classifier decomposes the input variables into subsets and represents statistical dependency within each subset, while treating the subsets as statistically independent. However, learning the structure of a semi-naïve Bayes classifier is known to be NP complete. The high dimensionality of images makes statistical structure learning especially challenging. This paper describes an algorithm that searches for the structure of a semi-naïve Bayes classifier in this large space of possible structures. The algorithm seeks to optimize two cost functions: a localized error in the log-likelihood ratio function to restrict the structure and a global classification error to choose the final structure. We use this approach to train detectors for several objects including faces, eyes, ears, telephones, push-carts, and door-handles. These detectors perform robustly with a high detection rate and low false alarm rate in unconstrained settings over a wide range of variation in background scenery and lighting.

1 Introduction

Many classes of images have sparse structuring of statistical dependency. Each variable has strong statistical dependency with a small number of other variables and negligible dependency with the remaining ones. For example, geometrically aligned images of faces, cars, push-carts, and telephones exhibit this property. Figure 1 shows empirical mutual information among wavelet variables representing frontal human face images. (Mutual information measures the strength of the statistical dependence between two variables.) Each "image" is a visualization of the mutual information values between one chosen wavelet variable, indicated by an arrow, and all the other variables in the wavelet transform. The brightness at each location indicates the mutual information between the variable at this location and the chosen variable. These examples illustrate common behavior where a given variable is statistically related with a small number of other variables.

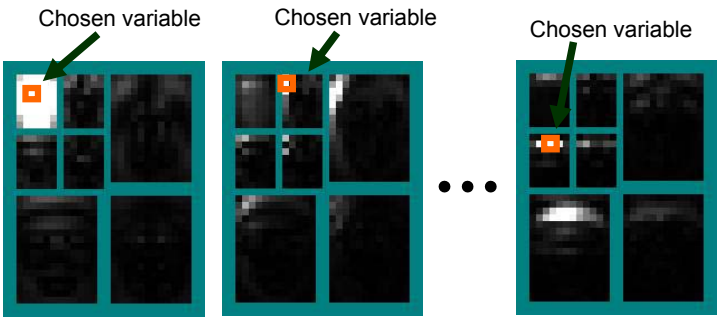

Fig. 1. Empirical mutual information among wavelet variables sampled from frontal faces images.

Sparse structuring of statistical dependency explains the empirical success of "parts-based" methods for face detection and face recognition [1][2][3][4][5]. Such parts-based methods concentrate modeling power on localized regions, in which dependencies tend to be strong and use weaker models over larger areas in which dependencies tend to be less significant.

In general, though, the problem of actually learning the statistical dependency structure for image classification has not received much attention in the computer vision community. Previous "parts-based" methods use hand-picked "parts." In this paper, we propose a method to learn the dependency structure from data and use the dependency structure to build a semi-naïve Bayes classifier.

A semi-naïve Bayes classifier [6] decomposes the input variables into subsets, representing statistical dependency within each subset, while treating the subsets as statistically independent. This classifier, $f(X_1,\ldots,X_n)$, takes the following form when written as a log-likelihood ratio test:

$$f(X_1,\ldots,X_n) = \log\frac{P(S_1|\omega_1)}{P(S_1|\omega_2)} + \log\frac{P(S_2|\omega_1)}{P(S_2|\omega_2)} + \ldots + \log\frac{P(S_r|\omega_1)}{P(S_r|\omega_2)} > \lambda$$
$$S_1,\ldots,S_r \subset \{X_1,\ldots,X_n\} \qquad (1)$$

where $X_1 \ldots X_n$ are the input variables, $S_1 \ldots S_r$ are subsets of these variables, and ω_1 and ω_2 indicate the two classes. For the problem of object detection, the classes are "object" and "non-object" where the non-object class represents all possible visual scenery that does not contain the object. For example, ω_1 may correspond to face and ω_2 may correspond to "non-face." In this form, the classifier chooses class ω_1 if $f(X_1,\ldots,X_n) > \lambda$. Otherwise, it chooses class ω_2.

Learning the structure of a semi-naïve Bayes classifier is challenging. The search space is enormous. It is super-exponential in n input variables, where n typically is $\sim 10^3$ for image classification. Moreover, the solution is NP complete; that is, we must compare every possible structure in order to find the optimal solution. The Bayesian score [9][14][15] is an ideal metric for comparing these model structures. It naturally penalizes for overfitting. However, computing the score for one model involves summing the probabilities of the training data over all possible instantiations of the model's parameters. The computational cost of doing so can be quite large.

Solution using heuristic search and approximate metrics is unavoidable. On lower dimensional domains (e.g., under a hundred variables) proposed methods have focused on joining one variable at a time using estimates of pair-wise distributions [6] or accuracy using cross-validation [7]. Another method [8] induces products of decision tree like structures. Our strategy selects a structure by sequentially optimizing two cost functions using greedy search techniques. The first function models local error in the log likelihood ratio function over pairs of variables. This function assumes every pair of variables is independent from the remaining variables. We organize the variables into subsets such that this measure is minimized. In particular, we generate a large semi-redundant pool of candidate subsets. The second optimization chooses the final solution as a subset of these candidate subsets by minimizing a global measure of classification error.

We use this method in the context of object detection. Object detection is the task of finding instances of the given object anywhere in an image and at any size. To perform detection we use a classifier that discriminates between the object and scenery that does not contain the object. This classifier operates on fixed size input "window", e.g., 32x24 for frontal faces, and allows for a limited amount of variation in size and alignment of the object within this window. Therefore, to perform detection, we exhaustively scan the classifier over the input image (e.g., with a step size of 4 pixels) and resized versions of the input image (e.g. with a scale factor step size of 1.189).

We use this approach to construct classifiers for detecting several types of objects: faces, eyes, ears, telephones, push-carts, and door-handles. These detectors perform robustly with a high detection rate and low false alarm rate in unconstrained scenery over a wide range of variation in background scenery and lighting.

2 Construction of the Classifier

There are two aspects to constructing the classifier: learning the structure of the classifier (assignment of the variables to subsets in equation (1)) and estimating probability distributions over each such subset. In our approach, these two steps are coupled. Section 2.1 describes the initial selection of a pool of candidate subsets. Section 2.2 describes the estimation of probability distributions over these candidate subsets. Section 2.3 describes the selection of a subset of the candidate subsets to form the final classifier.

The training data consists of images of the object for class ω_1 and various non-object images for class ω_2. The input to the classifier, $X_1 \ldots X_n$, is a wavelet transform of input window. However, there is nothing about this method that is specific to the wavelet transform. Conceiavably, the raw pixel variables or any transform of the image could be used. For more details about our image pre-processing, training data, and overall system for detection, refer to [12].

2.1 Minimizing Local Error in the Log-Likelihood Ratio

This step creates a large collection of subsets of variables. Such a selection of subsets reduces representational power. Only dependencies within each subset are represented. We therefore must decide which variables we will not represent and which dependencies we will not represent. We evaluate the cost of a proposed reduction by its error in modeling the log-likelihood ratio function. Our error metric is the difference between the true log-likelihood ratio function and the log-likelihood ratio under the given reduction. However, we compute this error only over pairs of variables. In particular, we consider three possible cases given by the following costs:

$$C_1(X_i, X_j) = \sum_{x_i \in X_i} \sum_{x_j \in X_j} P(x_i, x_j) abs[\log \frac{P(x_i, x_j \mid \omega_1)}{P(x_i, x_j \mid \omega_2)} - \log \frac{P(x_i \mid \omega_1) P(x_j \mid \omega_1)}{P(x_i \mid \omega_2) P(x_j \mid \omega_2)}] \quad (2)$$

$$C_2(X_i, X_j) = \sum_{x_i \in X_i} \sum_{x_j \in X_j} P(x_i, x_j) abs[\log \frac{P(x_i, x_j \mid \omega_1)}{P(x_i, x_j \mid \omega_2)} - \log \frac{P(x_i \mid \omega_1)}{P(x_i \mid \omega_2)}] \quad (3)$$

$$C_3(X_i, X_j) = \sum_{x_i \in X_i} \sum_{x_j \in X_j} P(x_i, x_j) abs[\log \frac{P(x_i, x_j \mid \omega_1)}{P(x_i, x_j \mid \omega_2)}] \quad (4)$$

(Note, each of the random variables is assumed to be discrete-valued. We use upper case notation to denote the random variable and lower case notation to denote a particular instantiation of the variable; that is, each sum is over all possible values of the given random variable.) C_1 is the error in modeling the two variables, X_i and X_j, as independent; that is the cost of removing the dependency between the two variables. C_2 is the error of removing one variable, X_j, from the pair. C_3 is the error of removing both variables from the pair. Each of these assumes that the pair, (X_i, X_j), is independent from the remaining input variables. We obtain these measures by empirically estimating the probability distributions, $P(X_i, X_j \mid \omega_1)$ and $P(X, X_j \mid \omega_2)$, for every pairings of variables, X_i, X_j.

Under these approximations, the error associated with a given choice of subsets, $F = \{S_1, \ldots, S_r\}$, can be computed as:

$$E(F) = \sum_{\substack{(X_i, X_j), \forall i,j,i \neq j \\ X_i \bigcap X_j \notin S_k, \forall S_k \\ X_i \in S_k, \exists S_k \\ X_j \in S_k, \exists S_k}} C_1(X_i, X_j) + \sum_{\substack{(X_i, X_j), \forall i,j,i \neq j \\ X_j \notin S_k, \forall S_k \\ X_i \in S_k, \exists S_k}} C_2(X_i, X_j) + \sum_{\substack{(X_i, X_j), \forall i,j,i \neq j \\ X_j \notin S_k, \forall S_k \\ X_i \notin S_k, \forall S_k}} C_3(X_i, X_j) \quad (5)$$

We seek a set of candidate subsets, F, to minimize this localized error function. We search for such a solution using two steps. The first step assigns the variables to n subsets using n greedy searches, where each input variable is a seed for one search. This guarantees that every variable is represented in at least one subset and, therefore, there are no errors of the form C_2 or C_3 for this step. (This is a fairly reasonable way to optimize $E(F)$ since the errors due to removing a variable tend to be greater than those of removing a dependency.) Each of the greedy searches adds new variables by choosing the one that has the largest sum of C_1 values formed by its pairing with all

current members of the subset. Such a selection process will guarantee that the variables within any subset will have strong statistical dependency with each other.

A second search may be desirable to reduce the number of subsets to smaller collection. We propose sequentially removing subsets until some desirable number, q, are remaining. At each step we remove the subset that will lead to the smallest increase in modeling error. In particular, it follows from equation (5) that the error in removing a given subset, S_k, is:

$$\sum_{\substack{X_i \in S_k, X_j \in S_k \\ X_i \bigcap X_j \notin S_i, \forall S_i, S_i \neq S_k}} C_1(X_i, X_j) + \sum_{\substack{X_i \in S_k, X_j \in S_k \\ X_j \notin Si, \forall S_i, S_i \neq S_k \\ X_i \in S_i, \forall S_i, S_i \neq S_k}} C_2(X_i, X_j) + \sum_{\substack{X_i \in S_k, X_j \in S_k \\ X_j \notin Si, \forall S_i, S_i \neq S_k \\ X_i \notin S_i, \forall S_i, S_i \neq S_k}} C_3(X_i, X_j)$$

In the experiments we describe later, the number of selected candidate subsets, q, ranged from 200 to 1,000. However, computational cost is linear in the number of candidate subsets and is not prohibitive for large numbers.

The sizes of the subsets are somewhat of an open question. Larger subsets have the potential to capture greater dependency. Subset size, however, increases the dimension of the probability distributions in equation (1) and, therefore, size must be balanced against practical limits in representational power and limited training data. One possible way of addressing this issue is terms of VC dimension as described by [4] or by Bayesian scoring techniques [9][14][15] For simplicity we describe the algorithm assuming the subsets will all have the same number of members, however, in our experiments we consider multiple sizes (leading to initially mn subsets above, where m is the number of sizes) to allow for greater variety in the representation.

2.2. Estimating the Probability Distributions

This step estimates log-likelihood ratio functions, $\log (P(S_k|\omega_1) / P(S_k|\omega_2))$, for each candidate subset, S_k. Any functional form (e.g. Gaussian, mixture model, Bayes net, non-parametric, etc.) is admissible as a choice for $P(S_k|\omega_1)$ and $P(S_k|\omega_2)$. In general, classification functions such as linear and quadratic discriminants, neural networks, or decision trees may also be admissible for $\log (P(S_k|\omega_1) / P(S_k|\omega_2))$ by proper normalization. In our current experiments, we represent each probability distribution by a table. This representation discretizes each subset of wavelet variables to a discrete feature value by vector quantization. (See [12] for more details on this representation.) The probability tables are then estimated by counting the frequency of occurrence of each feature value in the training data.

2.3. Minimizing Global Classification Error

We now form the overall structure of the semi-naïve Bayes classifier by choosing a group of the candidate subsets to form the final classifier. We choose the combination that minimizes an empirical classification error score. We measure performance by the area under the receiver operating characteristic (ROC) [13]. This measure of classification error accounts for the classifier's full operating range over values for the threshold, λ, in equation (1).

The difficulty in making this selection is that combinatorial space of candidate subsets is enormous. We use greedy search to incrementally combine subsets. In this search, the cost of evaluating each candidate combination on an example is small. In particular, the evaluation over any combination of subsets takes the form of equation (1) and is therefore simply the sum of the evaluations over the individual candidate subsets. Therefore, the individual candidate log-likelihood ratio functions only have to be evaluated once on each example. We then evaluate any combination as a sum of the appropriate pre-computed values.

In practice, it may be desirable to repeat this process several times where each time we prohibit identical choices to the previous searches. In particular, in our experiments we use a two part strategy that first finds *l* candidate combinations by comparing performance on training data (same images used to estimate probability distributions) then chooses the best of these by comparing performance on cross-validation data (images that are separate from other aspects of training).

3. Object Detection Experiments

We used this method to train detectors for frontal faces, eyes, ears, telephones, push-carts, and door-handles.

In frontal face detection, this method achieves relatively accurate detection rates at a fairly low computational cost. The table below show results on the MIT-CMU test set [10][2].

Recognition rate	86.5%	90.9%	94.0%	96.1%
False detections	3	7	22	65

These results are at least equal, and perhaps superior, to those of other state of the art detectors on this testing set including [1][2][4][5][10][11].

A human eye detector trained by this method has been tested extensively. The eyes were successfully located within a radius of 15 pixels with an accuracy of 98.2% on over 29,000 images of faces in an experiment independently conducted at the National Institute of Standards and Technology (NIST) by NIST employees[1] and reported back to the author. This dataset is sequestered and is not available to the public. The dataset consists of mugshot still images where there is only one face per person in the image and the face is that most prominent object in the image. The algorithm assumed that one face was present per image for this experiment.

The telephone detector was tested on one model of telephone over a set of 43 images with 107 telephones. The telephones had small variations in design, coloring, and age, etc. Some examples are shown in Figure 5. The table below gives the performance over different values of the classification threshold in each column:

Recognition rate	61.7%	78.5%	85.5%	91.6%
False Detections	0	9	35	90

[1] The author would like to acknowledge Jonathon Phillips, Patrick Grother, and Sam Trahan for their assistance in running these experiments.

Accurate and efficient detectors were also trained for human ears, push-carts, and door-handles and are illustrated in Figures 2 – 5 where graphic overlays indicate the detected positions of these objects.

4. Conclusion

Sparse structuring of statistical dependency makes it possible to construct a powerful classifier by only representing the stronger dependencies among a group of variables. We have illustrated how such structure can be exploited by a semi-naïve Bayes classifier model. In particular, we have shown that the structure of the classifier can be learned by a search that optimizes a local error criterion given by equation (5) followed by another search that optimizes a global error criterion described in Section 2.3. We have shown that such a classifier can be effective for difficult object detection tasks. We believe these techniques of learning statistical structure will carry over to more complex models. In particular, a semi-naïve Bayes model is the most basic form of the larger graphical probability family of models including Bayes nets, Markov random fields, factor graphs, chain graphs, and mixtures of trees. Such models make it possible to represent more complex structural relationships such as conditional independence and hold further promise for improved image classification and object recognition.

References

1. Schneiderman, H., Kanade, T. "Object Detection using the Statistics of Parts." To appear in *International Journal of Computer Vision*. (2003)
2. Rowley, H.A., Baluja, S. and Kanade, T. Neural Network-Based Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23-38. (1998)
3. Moghaddam, B.; Pentland, A. *"*Probabilistic visual learning for object representation.*"* *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:7, 696 -710 (1997)
4. Heisele, B., T. Serre, M. Pontil and T. Poggio. "Component Based Face Detection." CVPR, 2001. (2001)
5. Viola P. and Jones M. Rapid Object Detection Using a Boosted Cascade of Simple Features. CVPR, 2001. (2001)
6. Kononenko, I. "Semi-Naïve Bayesian Classifier." Sixth European Working Session on Learning. pp. 206-219. (1991)
7. Domingos, P., Pazzani, M.. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. *Machine Learning*. 29:103-130 (1997)
8. Rokach, L. and Maimon, O. "Theory and Applications of Attribute Decomposition." *IEEE International Conference on Data Mining*. pp. 473-480. (2001)
9. Cooper, G. and Herskovits, E. "A Bayesian Method for the Induction of Probabilistic Networks from Data." *Machine Learning*. 9:303-347. (1992)
10. Sung, K-K., Poggio, T.. Example-Based Learning for View-Based Human Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39-51. (1998)
11. Roth, D., Yang, M-H., Ahuja, N. A SNoW-Based Face Detector. *NPPS-12*. (1999)
12. Schneiderman, H. CMU Robotics Institute Tech Report. In Preparation.

13. Duda, R. O., Hart, P. E., Stork, D. G. *Pattern Classification*. John Wiley & Sons. (2001)
14. Heckerman, D., Geiger, D., Chickering, D. H. (1995) "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data." *Machine Learning* 20(3): 197-243
15. Friedman, N. and Koller, D. (2002) "Being Bayesian about Network Structure: A Bayesian Approach to Structure Discovery in Bayesian Networks." *Machine Learning Journal*.

Fig. 2. Face, eye, and ear detection

Fig. 3. Door-handle detection

Fig. 4. Telephone detection

Fig. 5. Push-Cart detection