A Comparison of Face Detection Algorithms

Ian R. Fasel[1] and Javier R. Movellan[2]

[1] Department of Cognitive Science, University of California, San Diego
La Jolla, CA, 92093-0515
[2] Institute for Neural Computation
University of California, San Diego
La Jolla, CA, 92093-0515 {ian,javier}@inc.ucsd.edu

Abstract. We present a systematic comparison of the techniques used in some of the most successful neurally inspired face detectors. We report three main findings: First, we present a new analysis of how the SNoW algorithm of Roth, Yang, and Ahuja (200) achieves its high performance. Second, we find that representations based on local receptive fields such as those in Rowley, Baluja, and Kanade consistently provide better performance than full connectivity approaches. Third, we find that ensemble techniques, especially those using active sampling such as AdaBoost and Bootstrap, consistently improve performance.

1 Introduction

Face detection is a crucial technology for applications such as face recognition, automatic lip-reading, and facial expression recognition (Pentland, Moghaddam, & Starner, 1994; Donato, Bartlett, Hager, Ekman, & Sejnowski, 1999). One aspect that has slowed down progress in this area is the lack of baselining studies whose goal is not just the development of complete systems but the analysis of how the different pieces of a system contribute to its success. The goal of this study then is to perform a systematic comparison of techniques used in three of the most successful neurally inspired face detection systems reported in the literature (Rowley et al., 1998; Roth et al., 2000; Viola & Jones, 2001). In particular, we focus on the different uses in these papers of high dimensional input representations, local versus global connectivity, and active sampling and ensemble techniques such as AdaBoost and Bagging.

2 Face Detection Framework and Image Database

The face detector used throughout this paper is based on the system described in Rowley et al. (1998). A small window is scanned across each image and a classifier is applied to each window, returning *face* or *nonface* at each location. This is repeated at multiple scales. Finally, nearby detections are suppressed using the clustering and overlap removal techniques described in Rowley et al. (1998). For training, we randomly selected 443 frontal faces from the FERET database

J.R. Dorronsoro (Ed.): ICANN 2002, LNCS 2415, pp. 1325–1330, 2002.

containing a variety of different individuals and facial expressions, and carefully aligned them within 20x20 pixel image patches. Following Rowley et al, small amounts of translation, scale, and rotation were randomly added to these images, resulting in a total set of 8232 training images. For negative examples we used 20,000 windows taken from scenery images known to contain no faces. Finally, to compensate for differences in lighting and camera gains, logistic normalization (Movellan, 1995)[1] is performed on each image subwindow before classification, except for an oval mask which blocks out background pixels. This normalization step was also performed for each window in the detection phase.

3 Factors for Comparison

We constructed sixteen experimental classifiers, each using a combination of the factors used in the Rowley et al. (1998), Roth et al. (2000) and Viola & Jones (2001) face detectors. The goal of these experiments was to clarify which particular techniques were responsible for the success of these algorithms.

Component Classifiers

There were two component classifiers. (1) **Ridge Regression** (Hoerl & Kennard, 1970) This method was used for training classifiers directly on real valued pixel inputs. Ridge regression is equivalent to linear backpropagation networks with weight decay. (2) **SNoW** This classifier first transforms the pixel inputs into a sparse binary representation and then uses the Winnow update rule of Littlestone (1998) for training. In effect, the resulting network performs an arbitrary function on each input pixel, then combines the function outputs linearly and applies a threshold. While this high-dimensional representation is counterintuitive to traditional neural network researchers (Alvira & Rifkin, 2001), Roth et al. have nevertheless reported the most accurate face detector in the literature. It is thus important to replicate Roth et al.'s results in order to form a better understanding of how SNoW produces such impressive results.

Each of these is optionally enhanced with **Bootstrap** . Rowley et al. (1998), Roth et al. (2000) and Viola and Jones (2001) all used a "Bootstrap" technique based on Sung and Poggio (1994). The Bootstrap technique is an active sampling technique for expanding the training set of a classifier during training. Bootstrap begins by training a classifier on the full set of face examples and a random set of 8000 nonface examples. This classifier is then used as a face detector on a set of unseen scenery images, and 2000 of the false alarms are randomly selected and added back into the training set. The existing classifier is then discarded, and a new classifier is trained on this expanded training set. The process repeats until the classifier has satisfactory performance.

Full vs. Local Connectivity

Rowley et al. (1998) used a standard multilayer perceptron, with receptive fields localized over 26 rectangular subregions inspired by Le Cun et al. (1989). The regions were 4 10x10 pixel patches, 16 5x5 pixel patches, and 6 overlapping

[1] $X = 1/(1 + e^{-\pi(.8)\mu/\sqrt[3]{\sigma}})$, where μ and σ are the mean and variance of the window.

20x5 horizontal stripes. In our experiments, component classifiers (trained with ridge regression or SNoW) received input from either the entire image or one of these subregions, which were then combined using an ensemble technique.

Ensemble Classifiers

We used several ensemble methods. (1) **Bagging** In this ensemble method, multiple instances of a classifier are trained on random samples from the training set. The final hypothesis of each classifier is then combined with a unity vote. This procedure has been shown to improve performance in many types of classifiers (e.g., Breiman, 1996; Opitz & Maclin, (1999). (2) **AdaBoost** A modification of Bagging, AdaBoost (Freund & Schapire 1996) trains an ensemble of classifiers sequentially. For each round of boosting, a distribution over the training set is modified so that examples misclassified in previous rounds of boosting receive more emphasis in later rounds. This procedure guarantees an exponentially decreasing upper bound on training error, and in practice AdaBoost is reported to be resistant to overfitting (Opitz & Maclin, 1999; Schapire & Singer, 1998. (3) **AdaBoost and Bagging for Feature Selection.** Tiu and Viola (2000) and Viola and Jones (2001) used AdaBoost as a method for selecting a few key features from a large set of possible features by constraining the weak learners to make their decision using only one feature at a time. Using this technique, Viola and Jones (2001) were able to select about 200 features from their initial set of 45,396 to build a high performance face detector. We tested the flexibility of this technique by using the Rowley rectangular regions trained with SNoW or AdaBoost as the basic features. We also tried replacing AdaBoost with Bagging in this algorithm.

Ensemble Bootstrap [Bootstrap + Bagging]. We experimented with a novel condition in which the classifier at each round of Bootstrap is saved, and the resulting classifiers are combined with a unit vote. This is similar to Bagging, but with active sampling instead of random sampling, and makes for a fairer comparison with other ensemble techniques.

We trained different classifiers from different combinations of these methods in order to tease apart the role each method plays in the success of a face detector. Not all possible combinations of these methods could be practically tested and thus we focused on experiments that tested the following questions: (1) Is the patch-based representation proposed by Rowley helpful? (2) Does SNoW really work? How? (3) How helpful are ensemble methods in the face detection task? (4) How crucial is the Bootstrap method?

4 Results and Discussion

For each experimental classifier, our performance measure was the total error rate on a cross-validation set of 4434 unseen face and nonface examples withheld from the training set from FERET; this measure seems appropriate since the classifiers were trained to minimize overall error. Table 1 shows these results for all the conditions, sorted in order of decreasing error rate. The three main findings were (1) SNoW consistently performed among the best classifiers, confirming the

results of Roth et al. (2000). In addition, we found an intuitive explanation for how SNoW works, which we describe below. (2) The local receptive fields used by Rowley consistently improved performance over equivalent classifiers that used the full 20×20 input. (3) Active sampling consistently improved performance as well; AdaBoost was always superior to the equivalent network using Bagging, and Bootstrap was usually superior to the equivalent networks that didn't use Bootstrap.

Table 1. Performance on generalization set of withheld faces from FERET.

Condition	Total Error	Hit Rate	False Alarm Rate
13) Patches + Ridge + Bagging + Bootstrap	47.85 %	52.75%	48.21%
1) Global + Ridge	21.95 %	96.00%	32.46%
2) Global + Ridge + Bagging	21.86 %	96.00%	32.00%
11) Global + Ridge + Bootstrap	11.53 %	92.75%	14.03%
8) Patches + Ridge + Bagging	6.47 %	99.75%	10.09%
12) Global + Ridge + Bagging + Bootstrap	2.28 %	97.75%	2.30%
4) Global + SNoW	0.64 %	98.50%	0.15%
5) Global + SNoW + Bagging	0.46 %	99.00%	0.15%
14) Global + SNoW + Bootstrap	0.35 %	99.75%	0.40%
15) Global + SNoW + Bagging + Bootstrap	0.21 %	99.50%	0.04%
3) Global + Ridge + AdaBoost	0.18 %	99.50%	0.00%
6) Global + SNoW + AdaBoost	0.16 %	99.75%	0.15%
10) Patches + SNoW + Bagging	0.16 %	99.75%	0.11%
16) Patches + SNoW + Bagging + Bootstrap	0.12 %	99.75%	0.04%
9) Patches + SNoW + AdaBoost	0.12 %	99.75%	0.04%
7) Patches + Ridge + AdaBoost	0.09 %	99.75%	0.00%

SNOW: This architecture appeared in four of the best five experimental classifiers, demonstrating its strength in the face detection task (the differences between the classifiers in experiments 13-16 are not significant, $Z_{test} = -2.83$, $p = 0.93$). Figure 1 shows different visualizations of the representation learned by SNoW. The left image shows the intensity corresponding to the peak weight in each pixel of the SNoW network. This image represents the SNoW's "favorite" face, i.e., the pattern of pixel values that maximizes the output of the SNoW model. At the surface level, SNoW has learned a favorite face that looks very much like a template, and is very similar to the favorite face of the linear network. The center image shows the sum of the weights for each pixel, which are shown in greater detail in the callouts to the right. This image represents the importance, or attentional strength assigned by SNoW to each pixel region. Clearly, the areas where SNoW has developed large weights correspond closely to recognizable facial features while surrounding weights have been lowered close to zero. The callouts display the tuning curves learned by SNoW for several different pixel positions. We found that all the important pixels have unimodal tuning functions with a range of preferred intensities. The fact that the tuning curves developed by SNoW are unimodal is interesting, because SNoW could have developed arbitrary functions, such as linearly increasing or decreasing weights (which would be identical to the ridge regression solution). This also suggests a possible architecture for an improved face detector: since the SNoW weights resemble bandpass tuning functions, a classifier that explicitly uses such tuning functions in training may be able to perform even better.

Local Connectivity is Better than Full Connectivity: Ensemble classifiers that split the input into the Rowley et al. patches typically performed better than classifiers that used full connectivity. The average performance of the four best classifiers using local patches (experiments 13, 14, 15, and 16) was significantly better than the performance of equivalent classifiers using global connectivity (experiments 8, 10, 11, and 12, $Z_{test} = -2.83$, $p < .005$). The conclusion we can draw from this is that the face detection task truly does benefit from the use of local receptive fields like the ones used in Rowley. Adding local connectivity to the ridge regression based ensemble classifiers even improved performance enough to produce the best overall classifier in the study.

Active Sampling: The active sampling done by AdaBoost and the Bootstrap method improved performance over their random sampling counterparts in all but one condition. Classifiers using AdaBoost (11, 12, 15, 16) performed significantly better than equivalent classifiers using Bagging (3, 5, 8, 13, $Z = -16.57$, $p < .005$). Interestingly, while AdaBoost helped in all cases, it was only slightly better than Bagging when used on SNoW. It seems that SNoW is able to account for most of the variation in the training set on the first round of Boosting, so that the impact of the active sampling done by AdaBoost is minimal. In contrast, AdaBoost provided huge benefits to the ridge regression classifiers.

This study provides clear evidence of the usefulness of some of the techniques used in face detection systems and suggests several areas for future improvements. First, we found that SNoW is indeed a promising classifier for face detection. The analysis of the way SNoW solved the problem suggests that a powerful face detectors may be built using explicit intensity tuning functions. Second, the superiority of sparse local representations, especially when used with the AdaBoost feature selection method, supports the exploration of other localist representations. Finally, the improvements provided by active sampling methods

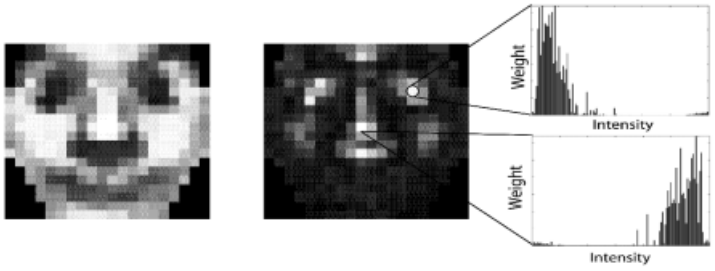

Fig. 1. SNoW generated weights from experiment (4). While at the pixel level the weights have learned the most likely intensities (left), most of the weights are close to zero (center). However for pixels in which the weights are not close to zero, the weights form a bandpass tuning curve. The callouts show the weights for two individual pixels which have this property. While eye pixels favor dark intensities, bridge of nose pixels favor high intensity.

like Bootstrap have exciting implications for the role of active sampling in other machine perception tasks.

References

Alvira, M., & Rifkin, R. (2001). *An empirical comparison of SNoW and SVMs for face detection* (Tech. Rep. No. CBCL Paper #193/AI Memo #2001–004). Massachussetts Institute of technology, Cambridge, MA.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24 (2), 123–140.

Donato, G., Bartlett, M., Hager, J., Ekman, P., & Sejnowski, T. (1999). Classifying facial actions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21 (10), 974–989.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Proc. 13th international conference on machine learning* (p. 148–146). Morgan Kaufmann.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, 12, 55–67.

Le Cun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L. (1989). Backpropogation applied to handwritten zip code recognition. *Neural Computation*, 1, 541–551.

Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: A new linearthreshold algorithm. *Machine Learning*, 2, 285–318.

Movellan, J. R. (1995). Visual speech recognition with stochastic networks. In . T. G. Tesauro, D. Toruetzky (Ed.), *Advances in neural information processing systems* (Vol. 7). MIT Press.

Opitz, D., & Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11, 169–198.

Pentland, A., Moghaddam, B., & Starner, T. (1994). View-based and modular eigenspaces for face recognition. In *IEEE conference on computer vision and pattern recognition.*

Roth, D., Yang, M., & Ahuja, N. (2000). A snow-based face detector. In *NIPS-12.* To Appear.

Rowley, H., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1 (20), 23–28.

Schapire, & Singer. (1998). Improved boosting algorithms using confidence-rated predictions. In *COLT: Proceedings of the workshop on computational learning theory.* Morgan Kaufmann.

Sung, K. K., & Poggio, T. (1994). *Example based learning for view-based human face detection* (Tech. Rep. No. AIM-1521).

Tieu, K., & Viola, P. (2000). Boosting image retrieval. In *Proceedings ieee conf. on computer vision and pattern recognition.*

Viola, P., & Jones, M. (2001). *Robust real-time object detection* (Tech. Rep. No. CRL 20001/01). Cambridge Research Laboratory.