Communicated by Richard Lippmann

Robust Classifiers without Robust Features

Alan J. Katz
Michael T. Gately
Dean R. Collins
Central Research Laboratories,
Texas Instruments Incorporated, Dallas, TX 75265 USA

We develop a two-stage, modular neural network classifier and apply it to an automatic target recognition problem. The data are features extracted from infrared and TV images. We discuss the problem of robust classification in terms of a family of decision surfaces, the members of which are functions of a set of global variables. The global variables characterize how the feature space changes from one image to the next. We obtain rapid training times and robust classification with this modular neural network approach.

1 Introduction

Neural networks are finding increasing application in the area of pattern recognition and classification (DARPA 1988). Most pattern recognition problems of practical interest are so complex that they do not easily reduce to simple rule sets; moreover, the computational loads of many pattern recognition problems severely tax present-day serial computers. Neural networks offer not only the potential of parallel computation but an approach to pattern recognition and classification rooted in learning: the networks train on a set of example patterns and discover relationships that distinguish the patterns. But neural network classifiers constructed through training may fail to be robust: success with the training data may mean little when the classifier is tested on data on which it did not train. The success of a learning-based approach depends on how representative the training data are of the complete data set. If the training data are not representative, then the likelihood that the mapping encoded by the network extends to test data is small. How to ensure the robustness of the classifier — whether a conventional or neural network classifier — is an open and fundamentally important issue in pattern recognition and classification. In this paper, we address the issue of robust neural network classification in the context of a particular, but representative, pattern recognition problem: the automatic target recognition (Roth 1990) problem of distinguishing targets from clutter.

Neural Computation **2, 472–479** (1990)

2 Robustness: A Framework

We define both targets and clutter in terms of a feature set $F = (f_1, f_2, \ldots, f_n)$. For any one image, I_a, targets and clutter are separable in the n-dimensional feature space. We can, therefore, build a neural network classifier to distinguish the two classes (we only consider multilayer, perceptron-type neural networks). As long as we train and test the neural network classifier with data points from I_a or images closely related to I_a, we find that the network classifies target and clutter reliably.

A problem usually arises when we test with data points from an unrelated image I_b. The classifier often fails to recognize the new target and clutter test points. Target and clutter points from I_b still separate in the n-dimensional feature space, but we find little correlation between the set of decision surfaces defined by the data from I_a and those defined by the data from I_b. As we go from image I_a to image I_b, the clusters of target and clutter points (and, hence, the decision surfaces) shift and rotate in complex ways in the n-dimensional feature space (see Fig. 1).

Though the features are nonrobust, we can still maintain some robustness if we can determine how the classification problem transforms as a function of some set of image parameters. The features then depend on the parameter set $P = (\lambda_1, \lambda_2, \ldots, \lambda_s)$, which varies from image to image. These parameters are any variables that affect the feature data (e.g., lighting, time-of-day, ambient temperature, and context of scene). For example, we can gain insight into P from knowledge of the sensor and how it operates. Since the features depend differently on elements of P, variations in the parameter set P lead to complicated transformations of

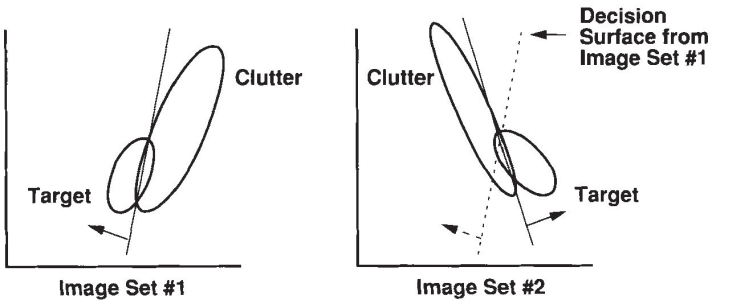

Figure 1: Target and clutter objects are separable within a given image. The statistical meanings of target and clutter, however, change from image to image and result in translations, rotations, and reflections of the decision surface.

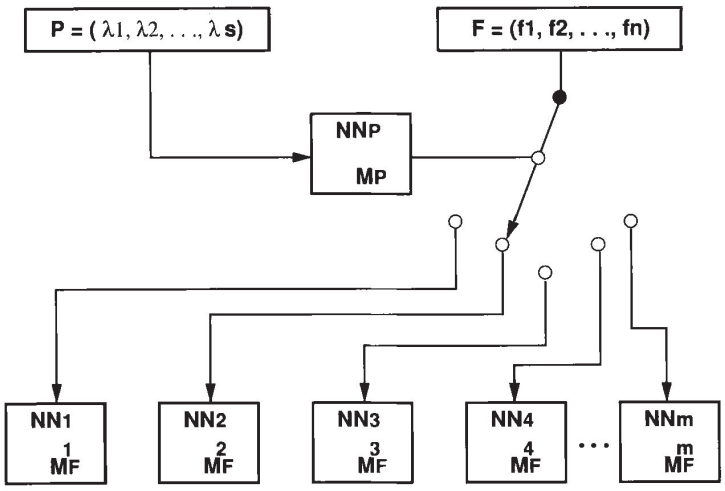

Figure 2: The parameter settings P drive the switch NN_p to route feature data F to a network NN_i, which maps the feature data to a binary output space (target or clutter).

points in the feature space. Hence, we replace the original classification problem, which was a mapping from the feature space F to some set of output classes (we consider only two output classes, targets and clutter), with a more complex classification problem, which involves a family of mappings $\mathcal{F}$, defined by the parameter set P. Usually, we will not have a deterministic model that relates mappings and the parameters of P, so we will have to derive the relationship between P and $\mathcal{F}$ through training with relevant examples.

We can incorporate the parameter set P into the classifier in several ways. First, we can use P to normalize the feature data prior to classification. Second, we can expand the original feature space F from n to $n+s$ dimensions and consider the mapping from an enlarged feature space F' to the output space: $M_{F'} : \Re^{n+s} \rightarrow \{0,1\}$. Third, since P defines a family of mappings, we can view the complex classification as a two step problem: (1) the mapping M_P from the s-dimensional parameter space P to the function space $\mathcal{F}= \{M_F^1, M_F^2, \ldots, M_F^p\}$, and (2) the mappings M_F^i from the n-dimensional features space F to the output space $\{0,1\}$. Alternatively, the order of the mappings M_P and M_F^i can be reversed (Hampshire and Waibel 1990).

Image set	Number of images	Location	Partially occluded targets
1	27	G	No
2	20	G	No
3	7	S	No
4	5	S	Yes
5	20	N	Yes

Table 1: Overview of Image Sets.

Our neural network approach is shown in Figure 2. The top network in the figure performs the mapping M_P, which is a switch for selecting one of the m classifiers in the second layer. The networks in the second layer are realizations of the mappings M_F^i. Modular neural network approaches (Waibel 1989), where the desired mappings are accomplished with several smaller neural networks, typically require less training time than approaches that utilize a single large network to carry out the mappings. Moreover, the modular approach is a means to train the system on specific subsets of the feature data and, therefore, to control the nature of the mapping learned by the networks.

3 Application to Automatic Target Recognition

3.1 Image Data. In this section, we apply a modular neural network approach to an automatic target recognition problem. The objective is to construct a classifier to distinguish various vehicles from background clutter. The data are five sets of bore-sighted infrared (IR) and daytime television (TV) images (256×256 pixels). Table 1 shows the number of images in each image set, the location codes, and which image sets have partially occluded targets. Each image set includes views of the targets from different viewing angles and at different distances. Locations S and G were mostly open fields with scattered trees, whereas location N was more heavily forested.

Extracting features from the images requires considerable image preprocessing. We perform the preprocessing using conventional algorithms. The first preprocessing step identifies regions in the image of high contrast and with high probability of containing a target. This screening step drastically reduces the overall processing time. High contrast areas are found separately for the IR and TV images. In the second preprocessing step, eight contrast-based and eight texture-based features are extracted from the screened regions (e.g., intensity contrast, intensity standard deviation, and signal-to-noise ratio). For each screened

area in an IR (TV) image, the corresponding set of pixels in the TV (IR) image is marked and features are extracted. We combine features from the same pixel locations in the IR and TV images to form a composite feature vector with 32 features. Fusing of information from the two sensors is crucial to distinguishing targets from clutter: using TV features or IR features alone makes the separation of targets and clutter more difficult. In addition to the 32 features, we extract 12 global features (e.g., total luminosity, maximum image contrast, and overall edge content) from each IR/TV image pair. These global features make up the parameter space P and provide information that relates similar images and differentiates divergent images.

We found that the feature space is nonrobust in the sense described in the preceding section. We trained five neural networks without hidden units, one for each of the five image sets, on the 32-component feature vectors. We demonstrated that the five decision boundaries defined by the neural networks are uncorrelated. The largest correlation between any two decision boundaries was only 0.61. We also cross-tested the data by training networks with images from one location and testing with images from the other locations. We found in one test, for example, that the false alarm rate increased by a factor of four (over a network trained on all locations) with no improvement in probability of detection.

3.2 Two-Stage, Modular Neural Network Classifier. To do the classification, we built a two-stage, modular neural network classifier. The first stage consists of an 8×5 feedforward neural network, with eight analog input neurons — we input eight of the 12 global features (the other four play a small role in the mapping) — and five output neurons, which designate the mappings M_F for the five image sets. This network performs the mapping from the parameter space P to the function space $\mathcal{F}$. The second stage of the classifier contains five independent neural networks, which effect the mappings M_F. The structure of the five neural networks are 16×1, 6×1, 4×1, 9×1, and $14 \times 4 \times 1$. The number of input neurons is smaller than 32 because we only include features that are relevant to the classification (we use the relative magnitudes of the weight vectors associated with each input neuron to determine the relevant features). All neural networks are trained with the backpropagation (Rumelhart et al. 1986) learning algorithm.

Each of the five networks at the second stage train on feature data from three-quarters of the images (randomly selected) from the respective image set: all the image sets are, therefore, represented in the training set. The neural network at the first stage trains on global features extracted from all images in the training set. The remaining one-quarter of the images serve as the test set. Test results (which are averages over 48 runs) for the modular system are shown in Table 2, Row A. The error bars represent the standard deviation for results from the various runs. The two-stage classifier generalizes well even though the input

set is noninvariant over the data domain. The generalization rate of the first-stage neural network is 0.96 and indicates that the first-stage neural network performs well as a switch to the appropriate network in the second stage. When no errors are made at the first stage, the test results for the classifier did not improve.

To test how essential the global features are for proper generalization, we built a $32 \times 16 \times 1$ neural network that did not include the global features, and we trained the network on three-quarters of the image data (chosen the same way as above). The number of training cycles and the overall training time were an order of magnitude greater for the single network than for the modular system (1 hr vs. 10 hr on a Texas Instruments Explorer II LISP machine with a conjugate-gradient descent version of back propagation); the training and generalization results for the single network, which are shown in Table 2, Row B, were within the error bars of the results for the modular system. These results did not improve when we reduced either the number of input neurons or the number of hidden neurons. We conclude that the advantage of the two-stage, modular system over the single network for this data set is training efficiency.

We ran a second set of experiments to demonstrate the generalization advantages of the two-stage approach. The data were two single-sensor (IR) image sets, one with 30 images and the second with 90 images. Sixteen texture-based features were extracted from each high contrast region. Not only were the decision surfaces in the two sets uncorrelated (0.58), but there was considerable overlap between the target cluster in one image set and the clutter cluster in the second set. We expected that the absence of the global features would impair learning and generalization.

The training sets for the two-stage, modular and the single networks

Test case	Probability of detection	Probability of false alarm
A	0.85 ± 0.06	0.18 ± 0.06
B	0.82 ± 0.03	0.20 ± 0.03
C	0.99 ± 0.01	0.00 ± 0.01
D	0.96 ± 0.01	0.05 ± 0.01

Table 2: Network results are expressed as P_d, the probability of detection and P_{fa}, the probability of false alarms. P_d is the number of targets correctly identified as targets divided by the total number of targets. P_{fa} is the number of clutter objects incorrectly identified as targets divided by the total number of objects identified as targets. First set of experiments: (A) two-stage, modular system (numbers represent an average over 48 runs), (B) $32 \times 16 \times 1$ network (numbers represent an average over six runs). Second set of experiments: (C) two-stage, modular system, (D) $16 \times 32 \times 1$ network.

were identical. The results are shown in Table 2, Rows C and D. The generalization rates for the two-stage, modular system were nearly perfect ($P_d = 0.99 \pm 0.01$ and $P_{fa} = 0.00 \pm 0.01$). The structure of the two neural networks at the second stage were both 16×1. The single neural network had the structure $16 \times 32 \times 1$ (networks with smaller numbers of hidden units gave worse results while the results did not improve with a network with 64 hidden units). The generalization rates for the single network were $P_d = 0.96 \pm 0.01$ and $P_{fa} = 0.05 \pm 0.01$. These results are worse than the results for the two-stage, modular system.

3.3 Discussion. A key parameter in the two-stage, modular network approach is the number of independent networks at the second stage. What determines this parameter is the number of diverse environments. To quantify the differences between environments, we construct a neural network for each new data set and measure the degree of correlation between it and the other existing networks. For the data set described in Table 1, for example, we trained five neural networks without hidden units, one for each image set. The network weights define a single hyperplane discriminant boundary between targets and clutter. We parameterize the hyperplanes in terms of their direction cosines and define the correlation between these hyperplanes as a function of the dot-product of the direction-cosine vectors and their normal distances from the origin.[1] When the correlation between a given hyperplane and the others is small, we add the new network to the second stage. If the correlation is large, then we do not add a network unless the orientations of the surfaces are reversed (that is, the placement of targets and clutter is inverted).

4 Conclusions

We presented a two-stage, modular neural network for classifying objects in different environments. The first-stage neural network incorporates global environmental parameters and switches among different neural network classifiers at the second stage that are trained for specific environments. Inclusion of global environmental features provides for more robust classification. Moreover, the modular nature of our approach yields shorter training times than more monolithic network approaches.

Acknowledgments

We thank Bill Boyd and Jerry Herstein for providing the infrared and TV data. We also thank Virginia Fuller for creating the graphics.

[1] More generally, we consider the correlations between the directions of maximum separability of the target points and the clutter points.

References

DARPA. 1988. *Neural Network Study.* AFCEA International Press, Fairfax, VA.

Hampshire, J. B., and Waibel, A. 1990. Connectionist architectures for multi-speaker phoneme recognition. In *Advances in Neural Information Processing Systems* **2**, D. S. Touretzky, ed., pp. 203–210. Morgan Kaufmann, San Mateo, CA.

Roth, M. W. 1990. Survey of neural network technology for automatic target recognition. *IEEE Transact. Neural Networks* **1**, 28–43.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning internal representations by error propagation. In *Parallel Distributed Processing*, Vol. 1, pp. 318–362. MIT Press, Cambridge, MA.

Waibel, A. 1989. Modular construction of time-delay neural networks for speech recognition. *Neural Comp.* **1**, 39–46.

Received 8 June 90; accepted 17 August 90.