

# A Comparison of Neural and Statistical Techniques in Object Recognition

Brian David Maciel and Richard Alan Peters II

Center for Intelligent Systems  
Vanderbilt University School of Engineering  
Station B, Box 131  
Nashville, TN 37235  
Telephone: (615) 343-0697  
Fax: (615) 322-7062  
E-mail: rap2@vuse.vanderbilt.edu

## *Abstract*

This paper reports on an experimental comparison of two visual object recognition methods: a radial basis function network (RBFN) which is an artificial neural network, and a synthetic discriminant function network (SDFN) which classifies objects statistically via analysis with optimal spatial filters. Both methods require training with a set of images representative of the objects to be recognized. A comparative performance analysis was performed after training both networks with the same image sets. The algorithms were implemented on a Pentium-class PC under MS Windows NT 4.0. Training images were captured from a color CCD cameras with standard NTSC resolution. Experiments were performed on both methods to determine the number of images per object necessary to train the networks, to estimate the two networks' accuracy of recognition, and to characterize their tolerance to image noise. It was found that when presented with a new image of one of the objects, RBFNs are more accurate at recognition than SDFNs. However, SDFNs are slightly more accurate in the presence of additive noise. Under the conditions of the experiments, RBFNs were found to provide an overall minimum classification accuracy of close to ninety percent.

## I. INTRODUCTION

**T**HE algorithms analyzed in this work are two standard methods for automatic visual object classification: radial basis function networks and synthetic discriminant function networks. RBFNs are a type of artificial neural network (ANN) [1] and SDFNs are statistical correlators implemented as optimal spatial filters [2]. Both algorithms are trained with a set of images of a set of objects. The training set contains several different images of each object with one and only one object in each image. Training forms classes within the parameter spaces of the networks, ideally one distinct class per object. Given another image of one of the objects, each algorithm returns the label of the class to which the object most likely belongs.

Both RBFNs and SDFNs incorporate basis set theory. From the training images each network, in effect, computes a set of orthonormal basis functions which span a parameter space for the given objects. The basis functions define axes that tend to maximize the linear separability of the object classes within the parameter space. Assuming that the classes are linearly separable, discriminant surfaces (in particular, hyperplanes formed by specific linear combinations of the basis functions) partition the parameter space into regions in which all points belong to the same class.

To determine the basis functions for a given training set, an RBFN minimizes a cost functional using Green's functions [3] in such a way as to derive the connection weights for an ANN through regularization [1]. During training, an SDFN minimizes a cross-correlation functional of the images to calculate an optimal linear filter from the sum of products of the image covariance matrix and training set images [2].

### *A. Radial Basis Functions*

Radial basis functions are defined in an  $n$ -dimensional metric space,  $\{\mathcal{V}, d\}$ , by a collection of points  $\{\mathbf{t}_j\}_{j=1}^K$ , called centers. Metric  $d(\mathbf{x}, \mathbf{y})$  is a generalization of the distance between points  $\mathbf{x}$  and  $\mathbf{y}$  in  $\mathcal{V}$ . Assume that  $\mathcal{V}$  is a linear vector space and that each center,  $\mathbf{t}_j$ , is a source point of a scalar field with amplitude  $\alpha_j$  and a Gaussian decay of variance  $\sigma_j^2$  with respect to distance from its center. Then at point  $\mathbf{x}$  the field intensity,  $f(\mathbf{x})$ , is given by

$$f(\mathbf{x}) = \sum_j^K \alpha_j G_j \left[ \frac{d^2(\mathbf{x}, \mathbf{t}_j)}{2\sigma_j^2} \right]. \quad (1)$$

The functions  $\alpha_j G_j \left[ \frac{d^2(\mathbf{x}, \mathbf{t}_j)}{2\sigma_j^2} \right]$  are the RBFs. Given a discrete set of data points, and their corresponding values, the parameters of (1) can be chosen so that it becomes a continuously differentiable interpolation function. Moreover, (1) can be implemented as a three-layer artificial neural network.

RBFNs were first used for multivariate function interpolation [4]. They found later use in the analysis of the XOR problem [5] and in the prediction of chaotic time series [6]. In [5] it was shown that the processing units in an RBFN can be adapted for individual data points to yield a designated response. More recent adaptation of RBFNs can be found in Haykin [7] and in Orr [8]. The RBFN used in the present work is described in [9] and in [1] which cast the general RBFN construction in terms of Catastrophe Theory and of Regularization Theory, respectively.

The advantages of RBFNs are twofold: in implementation and with respect to optimization of the training data. Implementation of the network is mathematically simple; it uses only basic linear algebra. Computations on the input data do not require iteration and are therefore relatively cheap. Optimization of training data occurs through via minimization of a cost functional that measures the deviation of the solution from its true value as given in the training data.

### B. Synthetic Discriminant Functions

Synthetic Discriminant Functions were introduced in [2] for military target recognition. Essentially, an SDF is a Matched Spatial Filter (MSF) derived from a training set of images of an object. It is an optimal linear filter that is created by minimizing the energy of the output signal when it is applied to the set of input images. The MSF is linearly related to the inverse of the autocorrelation of the set of input images. Since the autocorrelation of a signal is the inverse Fourier transform of the signal's power spectrum, the MSF is a function of the power spectra of the training images. Since a power spectrum has no phase information, SDFNs are relatively insensitive to lines and edges, especially the quasi periodic structures of surface texture.

A later variation, the Minimum Variance SDFN (MVSDF), proved to minimize the effects on the correlation output of additive noise in the input images [10]. The minimum average correlation energy (MACE) filter is a variant of the SDFN which enhances correlation peak intensities through minimization of the average energy in the correlation plane [11]. The SDFN used in the present work is a hybrid of the MVSDF and the MACE filter [12], [13].

SDFNs exhibit advantages over traditional ANNs in the classification of objects. An SDFN algorithm relies on simple statistical parameters and basic linear mathematics, which is less complex computationally than an ANN. Moreover, SDFNs tend to be insensitive to clutter and tolerant of geometrical distortion in comparison to ANN algorithms.

### C. Comparative Analysis

This paper compares RBFNs and SDFNs in terms of overall performance, computational expense, and noise tolerance. Experiments were performed using a set of 8 physical objects. Performance is binary; either the object is classified correctly or it is not. Computational expense was derived from the ratio of the recognition accuracy to the number of training images per set. Noise tolerance was tested by adding Gaussian distributed noise of five different variances.

## II. METHODS AND PROCEDURES

The RBFN and SDFN algorithms were implemented with MS Visual C++ 5.0 under MS Windows NT 4.0 on a 500 MHz dual-processor Pentium III PC. Since the algorithms were developed for the robots in the Intelligent Robotics Lab at Vanderbilt, they were written as software agents within the Intelligent Machine Architecture (IMA) [14].

All the images were taken with a color NTSC video camera. Only the luminance (L) component (quantized to 8 bits) of each LUV encoded image was retained. The original image size was  $640 \times 480$ . These were cropped to  $128 \times 128$  pixel windows centered on the original image. The cropped images were low-pass filtered by an  $3 \times 3$  nearest-neighbor average and decimated to  $64 \times 64$  pixels. Each image contained a single object from a set of eight. The objects were centered in each image, but varied in pose. The objects used were a spoon, a fork, a bowl, a coffee mug, a tennis ball, a Duplo block, and two aluminum cans. The test objects varied in size, spatial complexity, and pattern. They were selected to exhibit rotational or bilateral symmetry, self-occlusion, and different visual surface patterns.

The experimental methods for the comparison of the two algorithms were derived from [15] and from [16]. To compare the performances of RBFNs and SDFNs, 10 trials of training and detection were performed for each of the 8 objects in two different experiments. Over the course of the 10 trials for a given object, the number of images in the training set was increased from 5 to 90. Each training set contained images of the object to be recognized as well as *distractors* – images of the other objects. (See Table I.) After the training phase of a trial the algorithm is tested with the training set and with a previously unseen test set of 80 images that includes 10 images of the target object and 70 distractors (10 images of each other object). Thus, a set of 320 trials were performed, 160 on the RBFN and 160 on the SDFN. In each trial of one of the experiments, both networks were trained with the same image set and both were tested with the same set

The performance of an algorithm during a trial is rated by the percentage of correct identifications as given by the number of true-positives and true-negatives detected.

TABLE I  
TRAINING IMAGE DISTRIBUTION FOR THE 10 TRIALS

Trial	Images	Object	Distractors
1	5	4	1
2	10	8	2
3	20	15	5
4	30	20	10
5	40	25	15
6	50	30	20
7	60	35	25
8	70	40	30
9	80	40	40
10	90	40	50

True-positives indicate the correct identification of the object for which the network was trained. True-negatives are the correct identification of a distractor as such.

#### A. Experiments: Descriptions

Two experiments were performed to compare the relative capabilities of the two algorithms. The first of the two experiments estimates the number of images required in a training set to obtain accurate results from the algorithms. The second measures the sensitivity of the algorithms to noise.

1) *Experiment 1:* Neither the RBFN nor the SDFN necessarily improve their performance as the number of training images increases. It depends on the object. If the number of images in the training set is too small, the feature space of the RBFN will not contain enough data to form accurate inter-class partitions.

If the number of images in the training set for the RBFN is too small, the feature space will not be adequately partitioned for accurate object identification. On the other hand, it is possible to overtrain an RBFN. This happens when the number of input images is sufficiently large to induce so many decision boundaries in the feature space that some regions are completely cutoff from the parameters of any real objects. Therefore, there is an optimum number of images for any specific object that maximizes the probability of correct detection.

Likewise, if the number of training images for the SDFN is too small, the power spectrum of the object's image scene, will not be adequately defined for accurate object identification. Overtraining for the SDFN, however, differs slightly from the RBFN. This occurs when a sufficiently large number of extraneous class images are added to the training set which masks the power spectra of the filter with an excessive amount of extraneous information, lowering correlation output values between the filter and desired/extraneous class images.

The first experiment tested to find the optimum number of training images that maximizes the performance of both the RBFN and SDFN. Identical training sets were used to train each method for each corresponding trial. After training, each method was presented with an 80 image test set and output performance was assessed. Performance for each of the ten trials is formulated as a percent correctness ratio expressed as the sum of true-positives and true-negatives to the total number of test images. Data for both methods are shown in Figures 1 and 2.

2) *Experiment 2:* The second experiment tested the degradation of algorithm performance as a function of the variance of zero-mean Gaussian noise added to the test images. Each method was presented with an 80 image test set for each of the five noise trials. Starting with a zero-variance trial, five percent variance was added to each subsequent trial arriving at a final trial of 20% additive noise. Data for both methods are shown in Figures 3 and 4, demonstrating the percent correctness of each method for each trial as a ratio of the sum of true-positives and true-negatives to the total number of test images.

#### B. Experiments: Results

RBFN results from Experiment 1 proved successful in consistently recognizing certain objects. The data presented in Figure 1 is compiled from the true/false-positives/negatives of each test set and presents an overall network correctness. Complete tables of all data can be found in [17]. The RBFN's detection accuracy of the ball and the two cans increased with the number of training images, but decreased when detecting the block, bowl, cup, fork and spoon. The block and the bowl achieved a peak accuracy level when trained with 10 to 40 training images; the cup, fork and spoon achieved a peak accuracy level when trained with 10 to 20 training images.

SDFN results from Experiment 1 were consistently poor in comparison to RBFNs. The data presented in Figure 2 is compiled from the true/false-positives/negatives of each test set and presents an overall network correctness. The SDFNs proved to attain a maximum detection accuracy of 75% with the ball when using a training set of 40 images. All remaining test objects fluctuated in detection accuracy within the ranges of 20% to 40%.

RBFN results from Experiment 2 showed a gradual decline in detection accuracy over the noise trials. Data for these experiments are shown in Figure 3. The noise tolerance of the ball and the block proved to be the most robust in the presence of additive noise and fluctuated slightly through all trials. The remaining test objects showed a diminished detection accuracy, dropping an average of five percent over the span of five trials.

SDFN results from Experiment 2 proved sufficiently robust to additive noise. Data for these experiments are shown in Figure 4. Detection accuracy for test objects

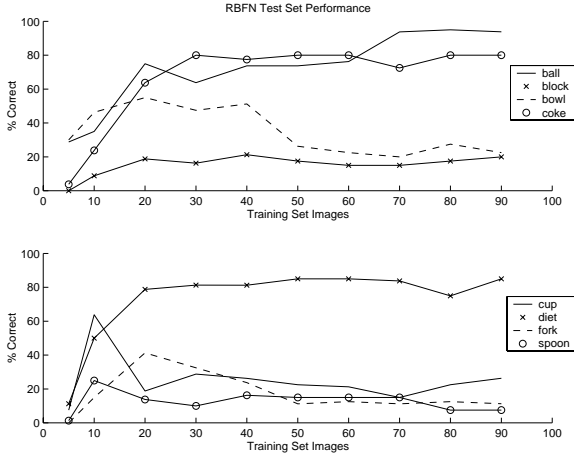


Fig. 1. RBFN Test Set Performance.

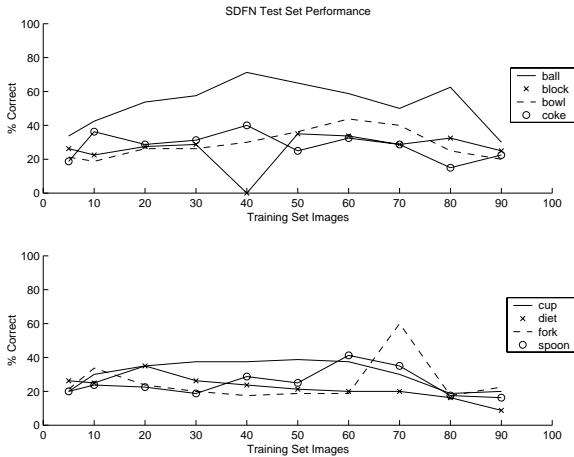


Fig. 2. SDFN Test Set Performance.

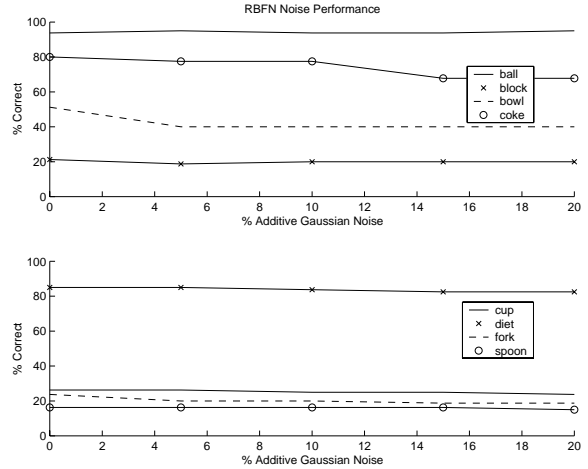


Fig. 3. RBFN Noise Performance.

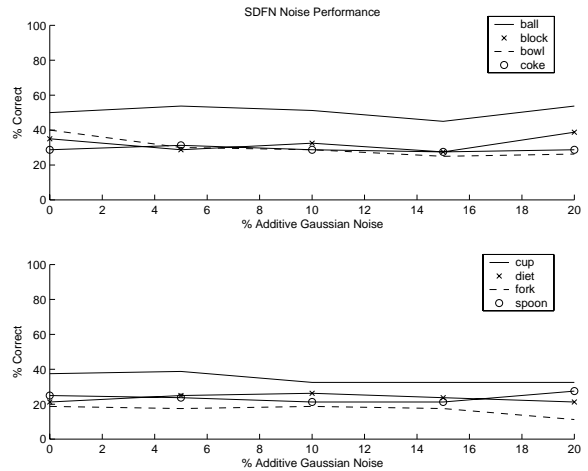


Fig. 4. SDFN Noise Performance.

fluctuated slightly from the initial zero-variance test set trial and quickly leveled off to a stable value. The fork was the only object which demonstrated a significant drop (occurring on the last trial) in detection accuracy over the five noise trials.

### III. DISCUSSION

RBFNs consistently performed better than the SDFNs under our test conditions. Over the span of ten performance trials all but three objects (block, fork and spoon) received a higher detection accuracy when tested against the 80 image test set using the RBFN. These objects received between a 15% to 25% higher accuracy with the SDFN. Further inspection of block, fork and spoon data in Figures 1 and 2 provides evidence that for objects that are not spatially symmetric, RBFN accuracy decreases as training set size increases. SDFN accuracy increases with the size of the training set.

For objects of complex surface texture or pattern, however, RBFN accuracy increases as training set size in-

creases, whereas the SDFN does not. This is demonstrated in Figure 3 where RBFNs showed a static or decrease in detection accuracy as noise variance increased. SDFNs remained insensitive to an increase in noise variance as they fluctuated slightly above and below their initial trial detection accuracy, shown in Figure 4.

SDFNs proved slightly more robust in the presence of additive Gaussian noise than the RBFNs. Data presented in Figure 3 demonstrated that RBFNs showed a static or decrease in detection accuracy as noise variance increased. SDFNs remained insensitive to an increase in noise variance as they fluctuated slightly above and below their initial trial detection accuracy, shown in Figure 4.

Poor performance of SDFNs can be attributed to the excessive number of extraneous training images used in each trial which weakens correlation output values for both desired and extraneous class images. As training set size increased the ratio of desired to extraneous class images decreased. Figure 2 demonstrates that as this ratio decreases, from an initial value of 4:1 to 5:3, the detection

accuracy gradually increases. As the ratio decreases to a final ratio of 4:5, the accuracy levels decrease, in most instances below the initial accuracy.

From performance data, an improvement in detection accuracy should occur with a decrease in extraneous class images and an increase in desired class images. Increasing the number of desired class images in a training set strengthens class characteristics of the desired class within the power spectra of the filter and aids in desired class identification. Decreasing the number of extraneous class images in a training set minimizes the masking of the filter's power spectra with extraneous class characteristics and improves correlation output between filter and desired/extraneous images.

Advantages in using the SDFN primarily lie in their ability to provide rotational invariance within an images scene and their sensitivity to additive Gaussian noise. Object orientation within an image scene is captured by the phase component of the Fourier Transform. Since the SDFN discards the phase component, only the magnitude component remains for training. The magnitude retains the characteristics of the object in the image scene but contains no information concerning its orientation and, thus, provides rotational invariance.

The SDFNs resilience to additive Gaussian noise is related to the phase component as well. The introduction of noise, which physically degrades the object in the image scene in time domain, is grouped primarily into the phase component in frequency domain. Since this component is not used to build the filter for filter/image correlation, an increase in additive noise has little effect on SDFN performance.

#### IV. CONCLUDING REMARKS

Having chosen the RBFN as the method which returns a higher recognition accuracy, the data for each object can then be tailored, yielding enhanced results. Threshold values for the RBFN experiments were constrained to 0.2 (extraneous classes) and 0.8 (desired classes). If threshold(s) could be redrawn to better discriminate desired from extraneous classes, percent correctness for each test object could increase dramatically. Figures 5 and 6 provide RBFN output for bowl and cup test objects as well as Coke can and fork test objects, respectively. Both figures have vertical partitions labeled a through h, which denote the grouping of each test object for their location in each test.

By inspection of both Figures 5 and 6, not all of the desired and extraneous class objects fall within the predefined threshold values. However, by redefining the thresholds to a value of 0.6 in Figures 5A and 5B, an overall accuracy ratio of 100% is obtained. Such accuracy is ideal, but even an accuracy ratio nearing 100% may be beneficial. By thresholding Figure 6A at 0.3 and Figure 6B at

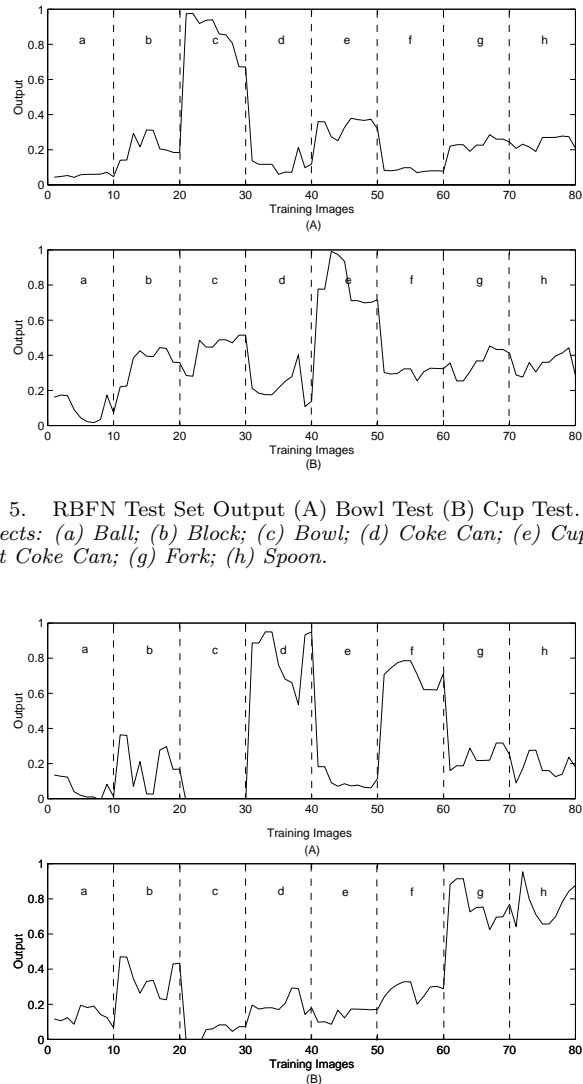


Fig. 5. RBFN Test Set Output (A) Bowl Test (B) Cup Test. Test Objects: (a) Ball; (b) Block; (c) Bowl; (d) Coke Can; (e) Cup; (f) Diet Coke Can; (g) Fork; (h) Spoon.

Fig. 6. RBFN Test Set Output (A) Coke Can Test (B) Fork Test. Test Objects: (a) Ball; (b) Block; (c) Bowl; (d) Coke Can; (e) Cup; (f) Diet Coke Can; (g) Fork; (h) Spoon.

0.5, the overall accuracy for each test object is close to 88%. Extraneous images which are not discriminated are similar to those of the desired class. Thus, a grouping of these test objects take place and a classification is made that the object is either an aluminum can, for Figure 6A, or an eating utensil, for Figure 6B.

Future work should be conducted in the the hybridization of methods to improve classification levels. Hybridization of both methods would take the best qualities of each in accomplishing the classification task. Although SDFNs proved relatively weak in discerning image classes with a limited number of training images, they provided stability in the presence of additive noise. Testing should be conducted using the RBFN with frequency domain training images for added noise tolerance.

## REFERENCES

- [1] T. Poggio and F. Girosi, "Networks for approximation and learning," in *Proceedings of IEEE*, 1990, vol. 78(9), pp. 1481–1497.
- [2] C.F. Hester and D. Casasent, "Multivariant techniques for multiclass pattern recognition," *Applied Optics*, vol. 19, pp. 1758–1761, 1980.
- [3] I. Stakgold, *Green's functions and boundary value problems*, John Wiley and Sons, New York, New York, 1979.
- [4] D.S. Broomhead and D. Lowe, "Multivariable function interpolation and adaptive networks," *Complex Systems*, vol. 2, pp. 321–355, 1988.
- [5] J.E. Moody and C.J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol. 1, pp. 281–294, 1989.
- [6] M.L. Minsky and S.A. Papert, *Perceptrons*, MIT Press, Cambridge, Massachusetts, expanded edition, 1988.
- [7] S. Haykin, *Neural Networks*, Macmillan Publishing Company, 1994.
- [8] M.J.L. Orr, "Regularisation in the selection of radial basis function centres," *Neural Computation*, vol. 7(3), pp. 606–623, 1995.
- [9] S.V. Chakravarthy and J. Ghosh, "Function emulation using radial basis function networks," *Neural Networks*, vol. 10(3), pp. 459–478, 1997.
- [10] B.V.K. Vijaya Kumar, "Minimum variance synthetic discriminant functions," *Journal of the Optical Society of America*, vol. A3, pp. 1579–1584, 1986.
- [11] A. Mahalanobis, B.V.K. Vijaya Kumar, and D. Casasent, "Minimum average correlation energy filter," *Applied Optics*, vol. 26, pp. 3633–3640, 1986.
- [12] B.V.K. Vijaya Kumar, A. Mahalanobis, S. Song, and S.R.F. Sims, "Distance-classifier correlation filters for multiclass target recognition," *Applied Optics*, vol. 35(17), pp. 3127–3133, 1996.
- [13] A. Mahalanobis, B.V.K. Vijaya Kumar, S. Song, S.R.F. Sims, and J.F. Epperson, "Unconstrained correlation filter," *Applied Optics*, vol. 33(17), pp. 3751–3759, 1994.
- [14] R.T. Pack, *IMA: The Intelligent Machine Architecture*, Ph. D. Dissertation, Vanderbilt University, Nashville, Tennessee, 1998.
- [15] P.R. Cohen, *Empirical Methods in Artificial Intelligence*, The MIT Press, Cambridge, Massachusetts, 1995.
- [16] M. Heath, S. Sarkar, T. Sanoeki, and K. Bowyer, "Comparison of edge detectors: a methodology and initial study," *Computer Vision and Image Understanding*, vol. 69(1), pp. 38–54, January 1998.
- [17] B. D. Maciel, "A comparison of neural and statistical techniques in object recognition," M.S. thesis, Vanderbilt University, Nashville, TN 37235, May 2000.