Morphological Neural Network Vision Processing for Mobile Robots

Donna Haun, Kristi Hummel, and Marjorie Skubic
Dept. of Computer Engineering and Computer Science,
University of Missouri-Columbia,
email: donnashaun@aol.com, khummel@ieee.org, skubic@cecs.missouri.edu

Abstract

Robust computer vision is thought to be essential for creating intelligent robots that can operate in unstructured and unknown environments. In this paper we investigate a vision processing algorithm for robot object recognition, using an ordinary shared weight morphological neural network. In particular, we test for robustness under variations in orientation, background contrast, and occlusion, while varying the neural network architecture to optimize object recognition versus processing time.

1. Introduction

Computer vision is thought to be essential for creating truly intelligent robots that can operate in unstructured and unknown environments. Indeed, robust vision is considered by many to be a significant bottleneck in the development of autonomous robots [5]. Building a robot that can interact in unstructured, unmodeled environments has been the ambition of AI and robotics researchers for decades. In this challenging environment, the robot must be able to successfully recognize objects of unknown orientation in cluttered and sometimes occluded situations.

One current approach to vision processing, a morphological shared-weight neural network developed by Won [4] for gray scale images, is investigated here. Criteria for evaluation include the ability to generalize results to accommodate changes in orientation, distance, lighting, and occlusion; and the speed of the algorithm.

Shared-weight neural networks, which may be used for pattern recognition with good generalization capability and the ability to learn feature extraction and classification simultaneously, have been successfully used by Gader et al [1] for target detection with such input images as an infrared tank and gray scale automobile. In contrast, they found that with normal linear correlation (matched) filters, such as Synthetic Discriminat Function Filters (SDF) and Minimum Average Correlation Energy Filters (MACE), correlation output is more dependent on the energy of the images than on the spatial structures, thus making necessary a large number of training images to account for differences in test images.

The algorithm we used, ordinary morphological neural network (OMNN) which is a shared-weight network, performs feature extraction using a "Hit/Miss Transform" [4]. When Won compared this network to other pattern recognition systems such as nonlinear correlation filter (NCFNN), generalized mean morphology (MMNN), and linear shared weight (LSNN) neural networks and the minimum average correlation energy filter technique (MACE), the OMNN was the best over-all performer as it trained quickly, results were independent of gray scale shifts (changes in lighting), and it was better at detecting occluded targets and reducing false alarms.

Imposing predefined constraints on weights can be used to improve the generalization capability of a neural network. Such methods include regularization, which uses *a priori* information to contribute to the solution; weight pruning, to eliminate weights that seem redundant; and weight sharing. On two types of shared-weight networks, a standard shared-weight neural network (SSNN) and a morphological shared-weight neural network (MSNN), Khabou [3] went on to investigate entropy optimized shared-weight neural networks as an additional constraint. His research has found that as the SSNN entropy value increases, the performance is enhanced but not to the level of the MSNN.

In this paper we test the concept of using an ordinary morphological neural network as mobile robot sensory input. In Section 2 we review the network model, and in Section 3 we describe network training and new image processing. Actual experimental process, results and discussion are included in Section 4 and conclusions follow in Section 5.

2. Network Model

The ordinary morphological neural network (OMNN) used for this paper is a cascaded network of one feature extraction (FE) layer and a feedforward (FF) network (see Figure 1).

Raw images, undersampled to decrease computation intensity, are input into the feature extraction layer. These pixels are mapped to feature maps using convolutions called hit/miss transforms (Figure 2), with one hit/miss weight matrix pair per feature map produced. The feature maps are essentially a composite of the targets eroded (Hit) and the backgrounds dialated (Miss).

The feature extraction hit/miss transform is a type of sliding window. The window size is the size of two structuring element matrices used in the transform, and size selection is a key variable in optimizing the morphological neural network. During the transform process, the hit weight matrix is first subtracted from a matrix of pixels, with the pixel being mapped in the center, producing a difference matrix. The miss weight matrix is added to the same original pixel matrix,

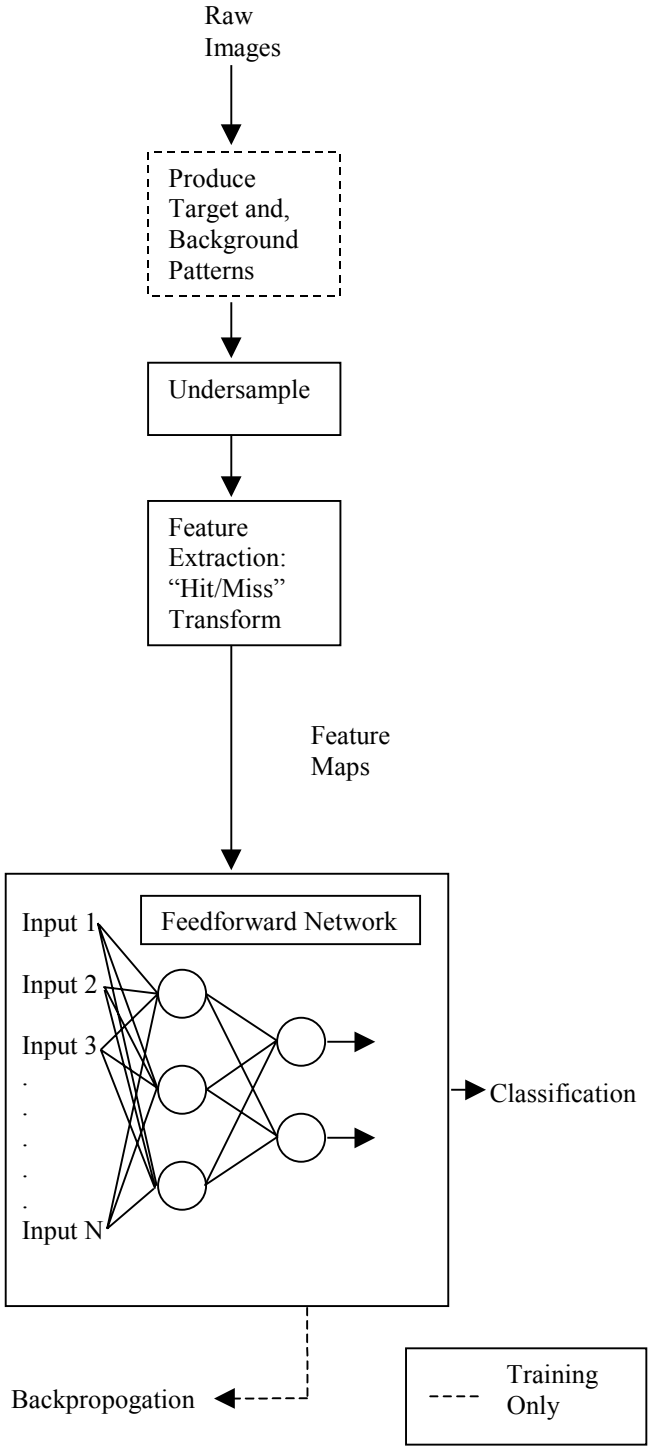

Figure 1. Ordinary Morphological Neural Network Block Diagram

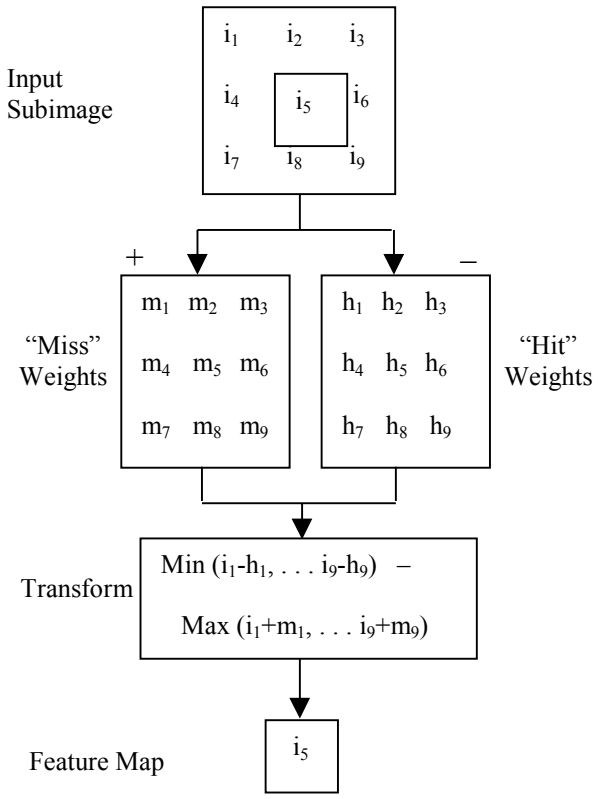

Figure 2. "Hit/Miss" Transform

producing a sum matrix. Then, the maximum of the sum matrix is subtracted from the minimum of the difference matrix, and this value is the mapping of the original pixel to the feature map.

Feature maps are then input into the feedforward (FF) network with windows large enough to contain the largest target expected [1]. In the feedforward network, the sums entering the neurons were increased by adding a bias. The activation function used to transform the input to output, is the sigmoid function. The output sets the value of that input into the next FF layer or that pixel in the output image. The network has one hidden layer, with these neurons enabling the network to learn complex object patterns by extracting progressively more meaningful features from the input patterns [2].

3. Network Training and Testing

During training, test sub-images provide the input to the first FE layer and the final output is a classification of "target" or "background". This method of training is called the "class-coded" mode of operation. While the network outputs values of 0 to 1 representing the confidence that an input represents a target or non-target, the returned result is an actual classification.

Training data consists of a set of subimages, developed from input images containing varying views of the "target". To create images for training of the "background" and to accomodate the inprecision of selecting the exact center of a training image, the model first selects as many background images as targets, then creates several subimages of each target and background subimage with slightly varied centers. An expected results vector is created for each of these "patterns", "1" for targets and "0" for backgrounds. Training data may be presented in the same sequence each epoch, or at random.

Several parameters specify and/or affect network training. The regularization parameter indicates the reliability of the training set, with a value of zero indicating that the set is completely reliable and a value approaching infinity indicating less reliability. The learning rate and momentum constant are used to adjust the speed of convergence and stability while reaching a desired error size.

Weights for the feature extraction operation are user-initialized, while the initial feedforward weight matrices are populated by a random number generator. All FE and FF weights are learned by back propagation. A signal completes its forward pass and then the correction its backward pass at the end of each training epoch, before the next input begins processing. A weight correction is the function of the learning and momentum parameters,

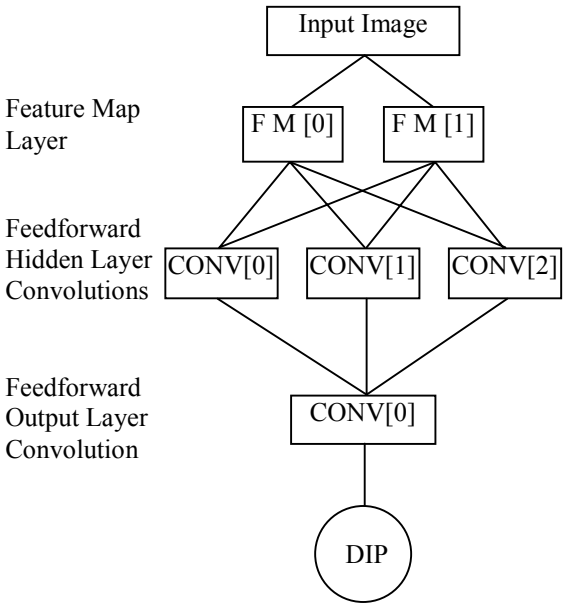

Figure 3. The processing of a new image to produce a detection image plane.

the local gradient of the activation function, and the input signal of the neuron.

When testing images for object recognition, the previously developed weights are used to perform the feature extraction and feedforward convolutions. Feature extraction is performed over the entire image rather than on a subimage, with the resulting feature maps being processed by subimage-sized windows. In this image-to-image transform, the output is a detection image plane (DIP) (See Figure 3).

4. Experiment

4.1. Process

Gray scale raw images were captured using a Hitachi KP-D50 CCD Camera and a Matrox Meteor framegrabber, and processed by a Nomadic Technologies robot running Linux on a Pentium 1 processor. All training and processing occurred in the vision lab, with the camera approximately 7.3 meters from the far wall.

To develop an effective neural network, a proper training set was formulated to show the target in typical environments and configurations. For this experiment, the target image was a small robot, 20 cm. x 33 cm. x 18 cm., built by modifying a toy tank. A set of thirty-two images were created showing the robot at different angles from 0 to 360 degrees with varying amounts of clutter and partial occlusion (see Figure 4 for training image examples). The full image size was 320x240 pixels. The largest target in the training set was 30x24 pixels, and this became the subimage size.

Four different network configurations were trained using these thirty-two images, varying the size of the structuring elements and the number of feature maps (7x7 sub element with 2 feature maps, 7x7 sub element with 4 feature maps, 5x5 sub element with 4 feature maps, 3x3 sub element with 4 feature maps). An undersampling rate of 2 decreased our original subimage input size from 30x24 to 15x12 for the input to the FF network. Since each of the thirty-two training images had one target, one "background" was selected per image and used to produce four target "patterns" and four background "patterns" per training image.

The feedforward network was structured with one hidden layer of three nodes to provide some of the feature detection benefit but not the increased computation of several layers The sigmoid activation function was used for both layers. Initial weights for the structuring elements were set at 0.5, and a random number generator was used to initialize the feedforward weights. The parameter variables not being evaluated in this paper held the same values used by Won [4]. These were the regularization parameter (0.0 indicating that the training set is reliable), learning rate (0.015), momentum (9.3), and training order (sequential). For each network condition, trained occurred for 500 epochs.

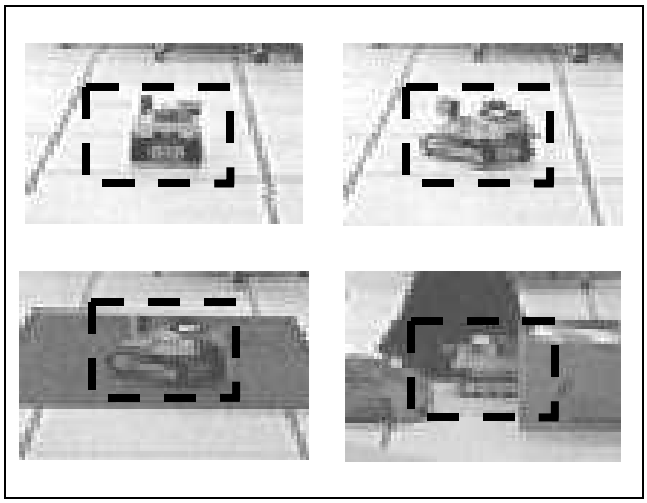

Figure 4. Four examples of images in the training set

When the training was complete, new images of the robot were tested to determine if the network could recognize the target residing in the new picture. In the detection image plane output, the shades of gray portray the confidence level of detecting the target at that particular location, with white meaning the target is likely to be there and black meaning the target is not likely to be there.

In addition to creating this detection image plane output, a supplementary output image was created by overlaying the original image with the DIP to allow visual evaluation of the target recognition results. Superimposed onto the original image are the pixels from the detection image plane that were found to lie above a given threshold value (indicating the detection of the target at that point), for which a value of 240 was used.

4.2. Results

Results are included here for training and testing with each of our four networks.

The training classification and RMSE training error results are shown in Table 1. At the completion of a network's training, all images in the training set, both target patterns and background patterns, were tested for correct classification. Targets were expected to be identified as targets, and backgrounds classed as backgrounds. The RMSE values given are the root mean square error at the end of training for the 500 epoch session.

Processing times on untrained images are shown in Table 2.

The new image superposition outputs of detection image planes overlaying the original images are shown in Figures 5 and 6. White indicates that a possible target area has been found.

4.3. Discussion

In the network training phase, correct classification of the input image is the critical results measure. As expected, the more complex network performed better than the less complex.

When testing new images, both object recognition and processing times are critical measures. When comparing the object recognition abilities of the four networks (Figure 5), the three most complex networks seemed to produce results similar to each other, with strong target recognition and very few small indications of noise. However, it is clear (also from Figure 5) that the network using a 3x3 structuring element is not sufficient for distinguishing between the actual target and the other "false alarms" in the picture. The more complex networks also produced similar target recognition abilities across the various test images.

The new image processing times are significant because the object recognition process needs to occur in real-time for vision to be an effective sensory input for a mobile robot. Currently, our processing of new images is too slow to allow the robot to react in a timely manner

Network	**7x7 SE, 4 FM**	**5x5 SE, 4 FM**	**3x3 SE, 4 FM**	**7x7 SE, 2 FM**
Targets classified as Targets	128	128	126	127
Target s classified as Backgrounds	0	0	2	1
Backgrounds classified as Targets	2	2	11	9
Backgrounds classified as Backgrounds	126	126	117	119
Total Misses	2	2	13	10
Training RMSE Error	0.085	0.084	0.210	0.170

Table 1. Classification of the training images by trained network.

Network	**7x7 SE 4 FM**	**5x5 SE 4 FM**	**3x3 SE 4 FM**	**7x7 SE 2 FM**
Image 1	14 sec	13 sec	12 sec	7 sec
Image 2	15 sec	13 sec	12 sec	8 sec
Image 3	14 sec	13 sec	13 sec	7 sec
Image 4	14 sec	14 sec	12 sec	7 sec

Table 2. New image processing times.

7x7 SE with 4 FM

5x5 SE with 4 FM

3x3 SE with 4 FM

7x7 SE with 2 FM

Figure 5. (Left) Comparison of target detection abilities by network for Image 3.

Image 1

Image 2

Image 4

Figure 6. (Above) Capability of the 7x7 SE with 4 FM network over different input images.

from the processing results. We suggest two possible approaches to decrease this processing time. One is to test for the target in smaller images rather than a full 320x240. The logic behind this idea is to ensure that the target is always within a certain "sub-image" of the camera's view by constantly updating the robot's position as the target moves. Our data shows (Table 2), that when testing new images, the number of feature maps used seems to determine the real difference in scanning time. When comparing the networks with the 7x7 structuring elements, the processing time doubles when the number of feature maps doubles.

It has already been shown that using fewer feature maps in the network cuts the processing time dramatically. More experiments should be done using the smaller input image to determine whether this factor will yield any considerable effects on the processing time.

5. Conclusions

In this paper we have shown that morphological neural network vision processing can approach the robustness needed for sensory input for mobile robots. Four networks have been tested on several images not in the original training set and three of them provided clear recognition of the target.

We have not yet post-processed the detection image plane to eliminate the small detection areas that represent noise. We believe that future research could reduce processing time to approach “real time” by scanning only the portion of the image where the robot is expecting to find the target.

References

[1] P. Gader, J. Miramonti, Y. Won, and P. Coffield, Segmentation Free Shared Weight Networks for Automatic Vehicle Detection, *Neural Networks*, Vol. 8, No. 9, pp. 1457-1473, 1995.

[2] S. Haykin, Neural Networks, Prentice Hall, Upper Saddle River, NJ, 1994.

[3] M. Kahbou and P. Gader, Automatic Target Detection Using Entropy Optimized Shared-Weight Neural Networks, *IEEE Transactions on Neural Networks*, Vol. 11, No. 1, January 2000.

[4] Y. Won, *Nonlinear Correlation Filter and Morphology Neural Networks for Image Pattern and Automatic Target Recognition*, PhD Dissertation, University of Missouri – Columbia, 1995.

[5] M. Vincze and G. Hager, (ed.) *Robust Vision for Vision-Based Control of Motion,* IEEE Press, Piscataway, NJ, 2000.