ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

An Alternative Approach for the Evaluation of the Neocognitron

Michal Steuer, Praminda Caleb-Solly, Jim Smith

Intelligent Computer Systems Centre, University of the West of England,
Coldharbour Lane, Bristol, United Kingdom. Michal.Steuer@uwe.ac.uk

Abstract. The neocognitron network is analysed from the point of view of the contribution of the different layers to the final classification. A variation to the neocognitron which gives improved performance is suggested. This variant combines the low level feature extraction capabilities of the initial layers with alternative classifiers such as LVQ and Class Based Means Clustering. This is shown to give performance which is superior to the either of those classifiers acting on their own, and to the neocognitron in its standard form on two different instances of the letter recognition problem.

1 Introduction

An artificial neural network called "neocognitron" was designed in the beginning of 80's [5]. Since then it has undergone several improvements both by the original developers [2], [3], and other researchers [6]. The network has a hierarchical structure and it was originally designed for recognition of handwritten digits and letters. The structure of the network is, however, suitable for a number of different applications.

We are interested in an application that involves the classification of medical signals [7], using the neocognitron as one of the blocks in the processing chain. This has led to an investigation of how the different parts of the network contribute to the final recognition performance. To do so, the original task of recognition of handwritten digits was used. By using the values of all the parameters of the network as published in the previous work [2], [6], a comparison to the original network could be made.

It should be noted, that this work is related to the version of the neocognitron as described in [2], as this version is the most suitable for the medical application. There are more variants of the network which may give different results. This paper concentrates on the analysis of the contribution of the different layers to the final performance of the original network.

2 The Necognitron

The structure of the neocognitron network can be seen in figure 1. The network consists of one *input layer*, four *processing layers* and one *output layer*. Each

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

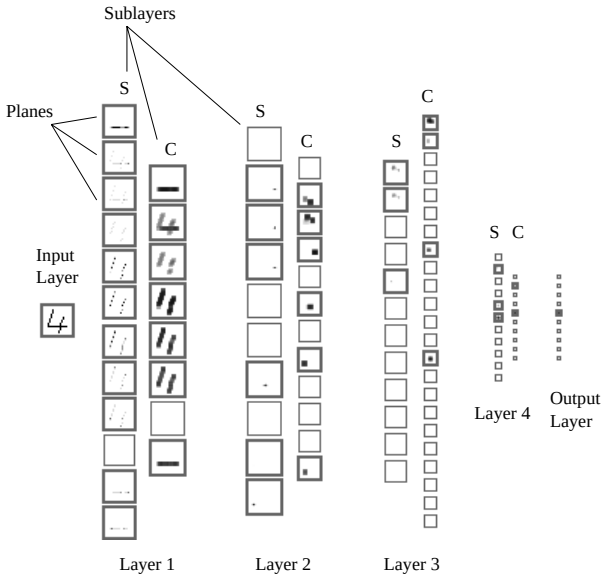

Figure 1: State of the network after propagation of the picture of the digit "4". The figure shows just a few selected planes from each layer.

processing layers can be further divided into two *sub-layers* named "S" and "C". Each *sub-layer* consists of several *planes*, square areas of neurons. In figure 1, the output of one neuron is represented by a single dot. The darker the point, the higher the output of the neuron. The number of planes and their sizes are different for each of the sub-layers.

Each neuron is connected to only a small square area of some of the selected planes of the previous layer. The size of this area is typically 3x3, 5x5 or 7x7 neurons. This square area shifts in parallel with the position of the target neuron, thus preserving topographical information between layers. All the neurons in a plane share the same set of weights. Sub-layer S contains the so called *feature extracting cells*. All the cells in a single plane are specially adapted to detect the presence of a particular feature in their input area. For example, neurons of the topmost plane detect small horizontal lines.

The connecting rule of sub-layers C is the same, but they play a different role. They "blur" the information received from the feature extracting cells to make the network more invariant to distortion of the input examples. Their weights are fixed and set accordingly.

Globally, the network is hierarchical. The first layer extracts very simple features like small lines of different orientations. The next layers extract features of increasing complexity. Features for the different layers are pre-defined, and they are not created or modified by training, so training examples are not required for the neocognitron. A few examples of features for the different lay-

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

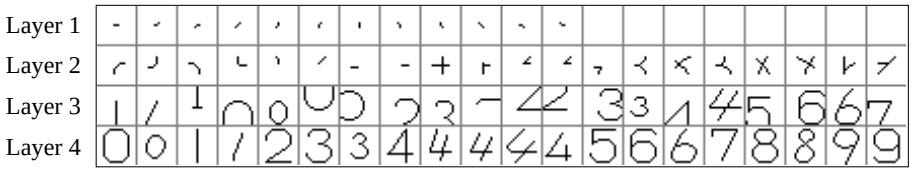

Figure 2: Examples of training features for different layers of the neocognitron

ers are shown in figure 2. The features are shown as they would appear in the input layer.

The size of planes in the last C sub-layer is only 1x1. There are 10 planes in this sub-layer, one plane per class. Here the classification output of the neocognitron is represented by the plane with the highest output.

After the last layer, an output layer has been added. This layer is not present in the original network [2]. This layer has the same number of planes as the last C sub-layer and its only role is to propagate only the plane with the highest output and to suppress the outputs of the other planes.

3 The Alternative Approach for Analysis

The classical approach is to evaluate the results obtained from the final layer of the neocognitron. As of yet there has been no attempt to investigate the performance of the intermediate layers.

Every sub-layer of the neocognitron can be seen as an independent neural network transforming its input vector space into an output vector space, which is later an input for the following layer. The network comprises of 8 *sub-layers*, one *input layer* (e.i. the vector space of the original input examples before any processing), and an *output layer*, a total of 10 vector spaces. It was decided to independently investigate each vector space by the application of standard classification algorithms.

As the S sub-layers primarily extract pre-defined features, these can also treated as *feature spaces*. By analysing the extraction of features in different layers we can explore the components that contribute to the formation of a "better" vector (or feature) space which can be more efficiently used for classifying the input examples.

To test these spaces two different classification methods were used. The first was Class Based Means Clustering (MC), in which for every class, a representative vector was calculated as an average of all the training examples. The testing examples were then classified according to the nearest center using Euclidian distance.

The second method used was Learning Vector Quantisation (LVQ) [4]. This is a well known neural network with more then one representative vector per class. The network is trained using supervised training. For details see [4].

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

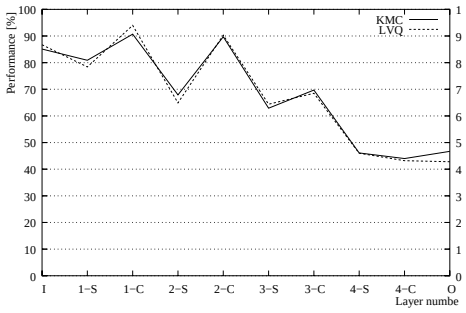

Figure 3: Performance on a layer by layer basis for set_11k

4 Training and testing sets

For testing the neocognitron and for training our evaluating algorithms, a set of examples of handwritten digits was needed. The author of [6] has kindly provided us with his set of 400 digits (40 digits per class). This set is later referred to as set_400. While 400 examples are adequate for evaluating the network as a whole, where no training examples are needed, the set is too small for our experiments where training examples are needed for the auxiliary classifiers. For this reason a second set of examples, obtained from [1], was used. This set contains almost 11000 digits from 44 different writers. This example set is referred to as set_11k. Unlike the set_400, this set had to be digitised from the original vector format, and the numbers were normalised in size during this digitisation. Another reason for having two sets was to have examples created by two independent groups.

For each run in each of the experiments presented in this paper, the sets were randomly partitioned into several training and testing sets. For the set_400 we used 40% of data for training (160 examples) and the rest for testing. For the set_11k we used 10% of data for training (1100 examples) and another 10% for testing. The pairs of training and testing sets were mutually exclusive.

5 Results

Comparing the performance after the first layer, i.e. using only one of the two classifiers, and the performance after the last layer, i.e. results from the neocognitron as a whole, we can see that the performance improves for the set_400 by almost 20% while it *decreases* for the set_11k by about 40%.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

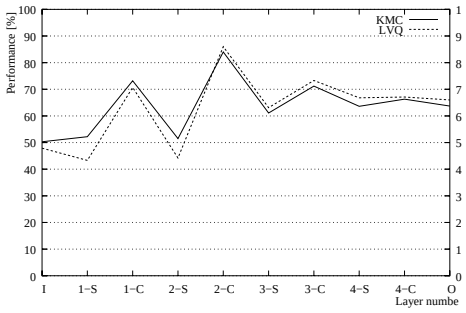

Figure 4: Performance on a layer by layer basis for set_400

Table 1: Results showing percentage average accuracy

	set_400				set_11k			
	MC		LVQ		MC		LVQ	
layer	avg	+/-	avg	+/-	avg	+/-	avg	+/-
I	**50.3**	1.8	**47.9**	4.2	**85.1**	0.5	**86.7**	1.1
1-S	52.2	6.1	43.3	3.4	80.9	0.7	78.4	6.3
1-C	73.2	0.6	70.7	1.0	90.7	1.2	94.0	0.4
2-S	51.5	2.3	44.2	2.9	67.9	2.3	64.9	5.7
2-C	**84.0**	2.2	**86.0**	0.7	**89.7**	0.8	**90.3**	1.4
3-S	61.1	1.8	63.1	0.7	62.9	1.9	64.4	3.6
3-C	71.2	0.1	73.3	0.9	69.7	1.3	68.5	1.5
4-S	63.6	0.6	66.8	0.7	46.1	2.2	46.0	3.5
4-C	66.3	3.3	67.1	3.3	44.0	0.2	43.2	2.3
O	**63.7**	0.9	**66.0**	3.2	**46.7**	0.7	**42.8**	2.2

Investigation of the performance on a layer by layer basis gives even more significant results. It shows that the maximum performance is usually reached at the fourth sub-layer (2-C). The point to note is that approximately the same maximum performance was reached for both sets at the fourth sub-layer regardless of different initial (layer I) and final (layer O) performances.

The graphs also show quite a significant improvement in performance between neighbouring S and C sub-layer. This indicates the important role played by the blurring C cells.

6 Conclusions

Experiments have shown that the performance of the neocognitron as a whole is not optimal for generic sets of numerals. The performance, as measured by other classifiers, improves only up to the 4-th sub-layer, and then starts to degrade. This suggests that an improved version of the neocognitron classifier could be constructed by using only the first 4 sub-layers followed by an MC or

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 125-130

LVQ classifier.

Our hypothesis is that as the set of features and their corresponding parameters are too specific in the latter layers (layer 3-S onwards) of the neocognitron, the resulting network can be regarded as "over-trained". This results in the sub-optimal performance on a general testing set when using the neocognitron as a whole. This hypothesis will be tested in future work.

Our work is continuing by investigations on how the feature sets could be made more general for character recognition, and on how to make the system applicable to a wider range of pattern recognition problems.

References

[1] F. Alimoglu and E. Alpaydin. Methods of combining multiple classifiers based on different representations for pen-based handwriting recognition. *Proceedings of the Fifth Turkish Artificial Intelligence and Artificial Neural Networks Symposium (TAINN 96), Istanbul, Turkey*, 1996. http://www.cmpe.boun.edu.tr/ alimoglu/tainn96.ps.gz.

[2] Kunihiko Fukushima and Nobuaki Wake. Handwritten alphanumeric character recognition by the neocognitron. *IEEE Transactions on Neural Networks*, 2(3):355–365, 1991.

[3] Fukushima K., Okada M., and Hiroshige K. Neocognitron with dual c-cell layers. *Neural Networks*, 7(1):41–47, 1994.

[4] T. Kohonen, J. Hynninen, J. Kangas, J. Laaksonen, and K. Torkkola. *LVQ_PAK: The Learning Vector Quantization Program Package*. Helsinki University of Technology, Laboratory of Computer and Information Science; Report A30, 1996. Available at http://www.cis.hut.fi/nnrc.

[5] Fukushima Kunihiko. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202, 1980.

[6] Lovell David R., Downs Tom, and Tsoi Ah Chung. An evaluation of the neocognitron. *IEEE Transactions on Neural Networks*, 8(5):1090–1105, 1997.

[7] M. Steuer, P. Caleb, G.B. Drummond, and A.M.S. Black. Visualisation and categorisation of respiratory mechanism using self organising maps. *IEE Proceedings Science, Measurement and Technology Journal*, 2000.