Selviah, D. R, & Midwinter, J. E. (1989). Memory Capacity of a novel optical neural net architecture. Proceedings of *"Optics in Computing International Symposium*, Toulouse,17-18 October 1989:ONERA-CERT, pp. 195-201

MEMORY CAPACITY OF A NOVEL OPTICAL NEURAL NET ARCHITECTURE

D. R. Selviah and J. E. Midwinter
Dept. of Electronic and Electrical Engineering, University College London, UCL, WC1E 7JE England, United Kingdom

Abstract
A new associative memory neural network which can be constructed using optical matched filters is described. It has three layers, the centre one being iterative with its weights set prior to training. The other two layers are feedforward nets and the weights are set during training. The best choice of central layer weights, or in optical terms, of pairs of images associated in a hologram is considered. The stored images or codes are selected carefully form an orthogonal set using a novel algorithm. This enables the net to have a high memory capacity equal to half the umber of neurons with a low probability of error.

Key words: Optical neural network

1. Introduction

Associative memory neural nets are useful for pattern recognition and for associating the recognized patterns with corresponding output patterns. In many applications, such as robotic vision, very fast recognition of complicated images is required. In particular, highly distorted images must be distinguished accurately. In these respects, optical systems offer two main advantages over their electronic counterparts. Firstly, the speed of the signal (the speed of light) increases the system operating speed provided fast, non-linear optical thresholding devices, such as MQW based opto-electronic logic devices [1], are used. Secondly, the high degree of connectivity or fan out possible using volume holograms, in photorefractive crystals means that totally connected networks with high number of neurons, 10^4-10^6, are within the rang of optics but not electronics (10^3 at present). However, one of the most serious challenges facing optics is the limited dynamic range offered by current components, typically 30 dB. This becomes most troublesome in the most interesting case when the input images are almost obscured by large amounts of noise. We, therefore, designed a novel three layer version [3, 4] of the well known Hopfield [5] net storing orthogonal codes in the intermediate layer to minimise the number of errors incurred by input noise. In this paper we show how the memory capacity, M, can be maximized (i.e. the error rate minimized) by a careful choice of the codes actually stored in the intermediate layer. We derive an expression for the dependence the memory capacity on the number of neurons. Section 2 reviews the original three layer Hopfield net. Section 3 looks in detail at the threshold behaviour. In section 4 the algorithm for selecting the codes to be stored is given and the memory capacity is derived. Conclusions are given in section 5.

2. Three Layer Hopfield Net

The three layer version of the Hopfield net is shown in Fig. 1 and is fully described in detail in the literature [3, 4]. The net is drawn using the matched filter formalism [3] in the time domain to aid clarity. This formalism has been found to be most useful for analysing [3] and designing [4] neural networks. Direct conversion to the space domain can be obtained by replacing time with space as the dependent variable. The small boxes in figure 1 are matched filters with their impulse responses indicated by their labels. The overall net consists of three subnets or layers. The first is a pattern association feedforward net and consists of an array of correlating matched filters. These give correlation speaks having magnitudes dependent on the agreement of the input unknown code and the impulse response of the filter. The noise gates simply gate out correlation time sidelobes (or space sidelobes in the case of an optical implementation) to give an ideal Dirac δ-function, scaled by the correlation peak. The last bank of filters in the first subnet generate, therefore, their impulse responses scaled by the correlation peak magnitude. These are summed bit by bit before being passed to the second subnet or layer. In this way the first layer associates or labels a set of arbitrary known binary, bipolar codes, $\boldsymbol{s_i}$, i=1,…,M (whose presence is to be recognised in the input) by a set of orthogonal codes, $\boldsymbol{o_i}$. The second layer has weights pre-programmed before training an simply associates the orthogonal codes with themselves. This layer has feedback via a non-linear threshold. In fact, this layer is simply a Hopfield net [5] operated in a synchronous manner, storing orthogonal codes and having diagonal elements equal to M. So it, ideally, converges onto the strongest orthogonal code in the sum which corresponds wit the strongest of the stored codes, $\boldsymbol{s_i}$. This constitutes recognition. The final layer associates an arbitrary output code, $\boldsymbol{p_i}$, with the orthogonal code label, $\boldsymbol{o_i}$, that has been selected by the second layer. In this way a code, pattern or image can be recognise and a corresponding associated code, pattern or image can be output.

Unfortunately, Hopfield nets suffer from two forms of incorrect convergence behaviour which increase the error rate (i.e. for a fixed error rate these reduce the memory capacity). One is convergence to an incorrect memorised code, which is most serious, since this cannot be detected. The other is convergence to a "spurious" code which is none of those memorised. The first type of error can be eliminated by storing only orthogonal codes since the cross correlations are zero. This increases the memory capacity for a given error rate [6] and enables codes to be distinguished using the limited dynamic range of optics in the greatest amount of .input noise. The second type of error occurs when the input is so distorted by noise that its correlation with several of the known codes are very similar in magnitude. This is considered below.

3. Threshold Behaviour

Consider the central subnet which has thresholding iterative feedback. The threshold is an odd function (figure 2) and so can be expanded in odd powers of the input.
For example, a soft hyperbolic tangent function with a low signal level gain of α is given by

$$\tanh(aA_{in}) = \alpha A_{in} - \frac{(\alpha A_{in})^3}{3} + \frac{2(\alpha A_{in})^5}{15} - \frac{17(\alpha A_{in})^7}{315} + \quad \Lambda \tag{1}$$

where A_{in} is the input amplitude and $|\alpha A_{in}| < \pi/2$ for equation (1) to be valid. The input to the threshold consists only of the sum of codes memorised in the Hopfield net, each having a different amplitude, a_i

$$A_{in} = a_1 o_1 + a_2 o_2 + a_3 o_3 + a_4 o_4 + \quad \Lambda \tag{2}$$

In our case these are orthogonal codes. From equation (1) the output from the threshold has only terms which are products of an odd number of input stored codes. Since we are restricting the stored codes to be bipolar

$$o_i o_i = E \tag{3}$$

where E is a vector whose bits are all +1. This can be used to reduce the number of terms. Let us assume that the amplitudes, αa_i, are much less than one, at least initially. Then we can neglect terms in $(\alpha a_i)^5$ and higher to give

$$
\begin{aligned}
A_{out} = \; & \alpha a_1 o_1 \left(1 - \frac{\alpha^2 a_1^2}{3} - \alpha^2 a_2^2 - \alpha^2 a_3^2 - \alpha^2 a_4^2 - \; \Lambda \right) \\
& + \alpha a_2 o_2 \left(1 - \alpha^2 a_1^2 - \frac{\alpha^2 a_2^2}{3} - \alpha^2 a_3^2 - \alpha^2 a_4^2 - \; \Lambda \right) \\
& + \textit{other similar first order terms in } o_i \\
& - 2\alpha^3 a_1 a_2 a_3 o_1 o_2 o_3 - 2\alpha^3 a_1 a_2 a_4 o_1 o_2 o_4 - 2\alpha^3 a_1 a_3 a_4 o_1 o_3 o_4 - 2\alpha^3 a_2 a_3 a_4 o_2 o_3 o_4 \\
& - \textit{other similar third order terms in } o_i \qquad (4)
\end{aligned}
$$

If we had included all terms explicitly there would be ones in products of 5 and higher odd multiples of different stored codes.

Equation (4) shows that the threshold passes the stored input codes with a change in their relative amplitudes and also generates new output codes which are products of odd numbers of stored codes. These can all add in such a way at the output to give a spurious binary, bipolar code. If $\boldsymbol{o_1}$ and $\boldsymbol{o_2}$ have very similar magnitudes (a_1 and a_2; $a_1 > a_2$) before the threshold they experience different gains, g_1 and g_2, where the differential gain is given approximately by

$$g_1 - g_2 = \frac{2}{3}\alpha^3 \left(a_1^2 - a_2^2 \right) \qquad (5)$$

Similar differential gains occur between all pairs of codes in the input sum. This is also the fractional difference in the code magnitudes after the threshold. So on repeated passes through the threshold the fractional difference grows until only one binary code remains. Note that this effect is dependent on the third order non-linearity. Fifth and higher odd order non-linear terms have been neglected. We note, in passing, that a similar differential gain could be achieved by placing square law non-linearities in the correlation plane at the same place as the gates. The threshold could then have linear gain, $2\alpha^3/3$.

Let us assume for the remainder of this section that all of the higher odd product codes also have been stored in the network. Products of orthogonal codes are also orthogonal codes so the output of the threshold passes unaltered through the network to return to the input of the threshold. If the threshold gain is too high the third and higher order code terms will be so strong that they will mix with the first order terms on this second pass through the threshold, so perturbing their magnitudes in an undesirable manner. By reducing the threshold gain, α, for fixed input code amplitudes, the operating region moves into the linear part of the curve and the higher order non-linearities decay so that the higher odd product codes are reduced with respect to the first order ones. However, the differential gain which depends at least on the third order non-linearity also drops. In fact, both the third order product term and the differential gain vary as α^3, so reducing α affects both at the same rate.

With a reduced threshold gain more iterations are required before convergence. On each iteration a new weak third order product code is generated from the input first order terms and this adds to the same code generated on previous threshold passes which also passes around the loop and through the threshold. In this way the accumulated effect of multiple iterations builds up strong third and higher order product codes which can cause convergence to a spurious code.

4. Novel Storage Algorithm and Memory Capacity

For N bit orthogonal codes it is possible to find a set of codes, none of which can be formed by multiplying any number of the other codes in the set. These are multiplicatively independent codes and by multiplying them together in various combinations one can form a complete set of orthogonal codes. If there are n independent codes, combining different numbers of them gives

$$N = 1 + {}^nC_1 + {}^nC_2 + {}^nC_3 + {}^nC_4 + \Lambda = 2^n \qquad (6)$$

$$o_1^2 \qquad o_1 \qquad o_1o_2 \qquad o_1o_2o_3 \quad o_1o_2o_3o_4$$

where typical code products are given under each term as examples, N is the number of bits in each code and the number of codes in a complete set of orthogonal codes. From this equation we can calculate n

$$n = \log_2 N \qquad (7)$$

If we store all of the independent codes and all of their products we run into the problem that the odd order product codes generated in the threshold pass straight through the interconnection net to the threshold where they are increased by the new odd order product codes generated on subsequent iterations. However, if the *odd multiples of the independent codes are not stored* the odd order product codes generated by the threshold are filtered out by the interconnection net on each iteration and so cannot build up to any appreciable value. Even multiples of the independent codes can be stored. The filtering action of the net on the odd product codes has no effect on the stored codes whose magnitudes separate more and more on each.iteration due to the differential gain. This gives a memory capacity of

$$M = 1 + {}^nC_2 + {}^nC_4 + {}^nC_6 + \Lambda = 2^{n-1} + n - 1 \qquad (8)$$

$$o_1^2 \quad o_1o_2 \quad o_1o_2o_3o_4 \quad o_1o_2o_3o_4o_5o_6$$

Using equation (7) this becomes

$$M = (N-2)/2 + \log_2 N \qquad (9)$$

Figure 3 plots equation (9) as memory capacity, M, versus number of neurons, N, in this novel net. The memory capacity is $0.75N$ for 4 neurons and asymptotes to $0.5N$ for large values of N. This is several times larger than the $0.15N$ memory capacity quoted by Hopfield [5]. Moreover, it is in agreement with simulations performed on a similar net [6].

Perfect recall will only be achieved, for highly distorted input codes to the full net (Fig. 1), when the threshold gain is sufficiently low that higher order terms above the third can be neglected at least for the first few iterations. Later, once the fractional separation of the magnitudes of the strongest codes is sufficiently large this will not be a problem.

5. Conclusions

A novel three layer Hopfield net was described which stores a special selection of orthogonal codes in the central layer. The selection of these codes (and, hence, the weights) is described. The memory capacity of the full net was found to be $0.5N$ for accurate recognition and association of arbitrary patterns.

Acknowledgement

The authors wish to thank José Capmany, visiting scholar from Universidad Politecnica de Madrid, Spain, for calculating the closed form expressions of the memory capacity series in equations (8) and (9). We would also like to acknowledge the financial support of the Joint Opto-Electronic Research Scheme of Great Britain: Digital Optics: Applications to Computing and Communications.

References

1. Selviah, D. R., & Midwinter, J. E. (1987). Pattern recognition using opto-electronic neural networks. *IEE colloquium digest, "Optical techniques for information processing"*: IEE, No. 1987/105, pp. 6/1-6/4

2. Midwinter, J. E., & Selviah, D. R. (1989). Digital neural networks, matched filters and optical implementations. Chapter 13 in Aleksander,I. (ed.) *Neural Computing Architectures: The Design of Brain-like Machines*. Kogan Page, North Oxford Academic Publishers Ltd., 1st edition, pp. 258-278. ISBN: 0946536473 and (2003) *(Paperback)*. The MIT Press Classics Series series. MIT Press, Reprint edition, 258-278. ISBN: 0-262-51150-9

3. Selviah, D. R., Midwinter, J. E., Rivers, A. W., & Lung, K. W. (1989). Correlating matched-filter model for analysis and optimisation of neural networks *Radar and Signal Processing, IEE Proceedings F*, Vol.136, No. 3, pp.143-148. ISSN: 0956375X

4. Selviah, D. R, & Midwinter, J. E. (1989). Matched filter model for design of neural networks. *Institute of Physics conference proceedings, New developments in neural computing*, Taylor, J. G., Mannion, C. L. T. (ed.): IOP, pp. 141-148

5. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities, Proceedings of the National Academy of Sciences, USA, Vol. 79, pp. 2554-2558

6. Lam, T. Y. D, & Carroll, J. E. (1989). Double layer associative memories, *Optics Communications*, Vol. 70, No.4, pp. 293-298

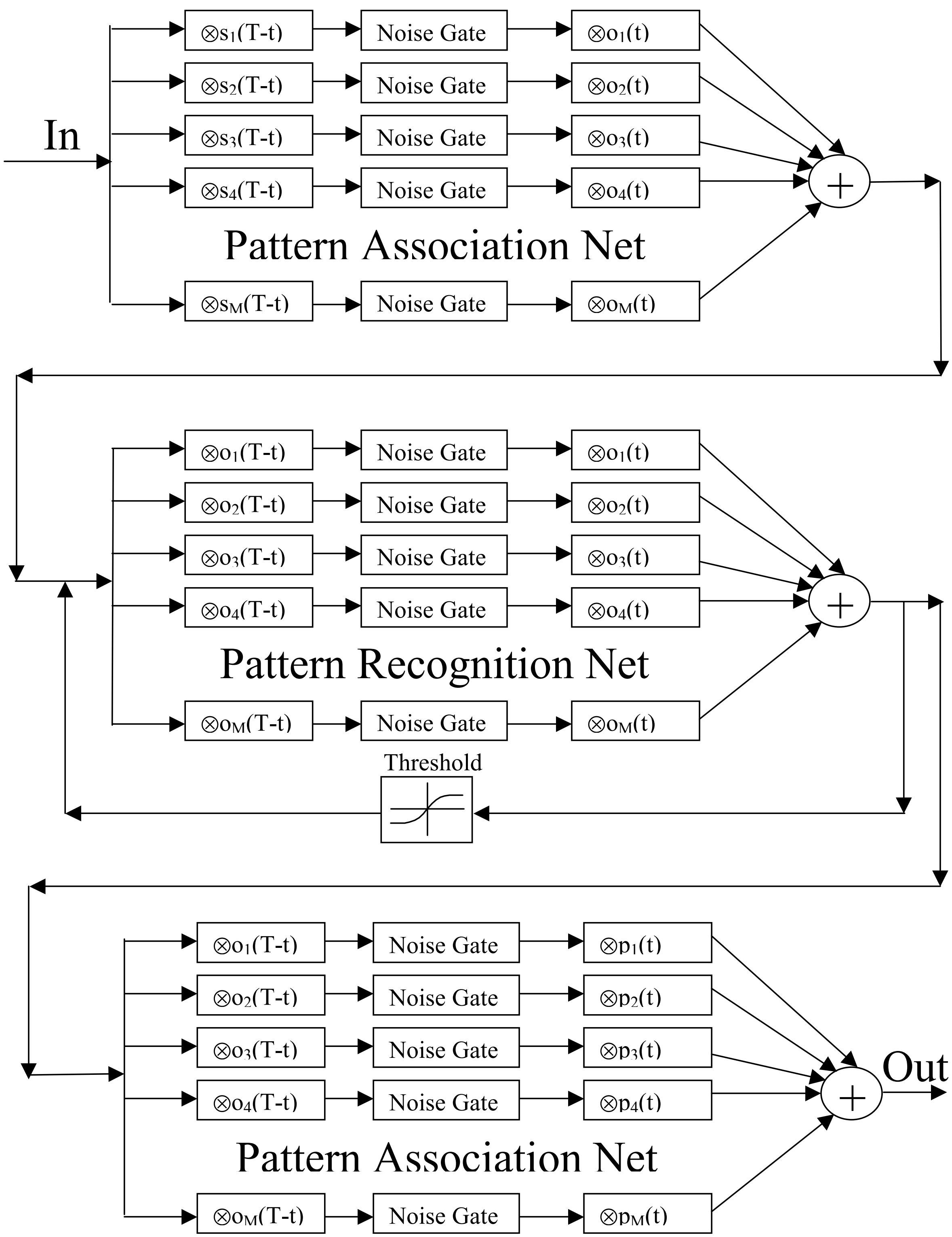

Figure 1 Three layer associative memory Hopfield net

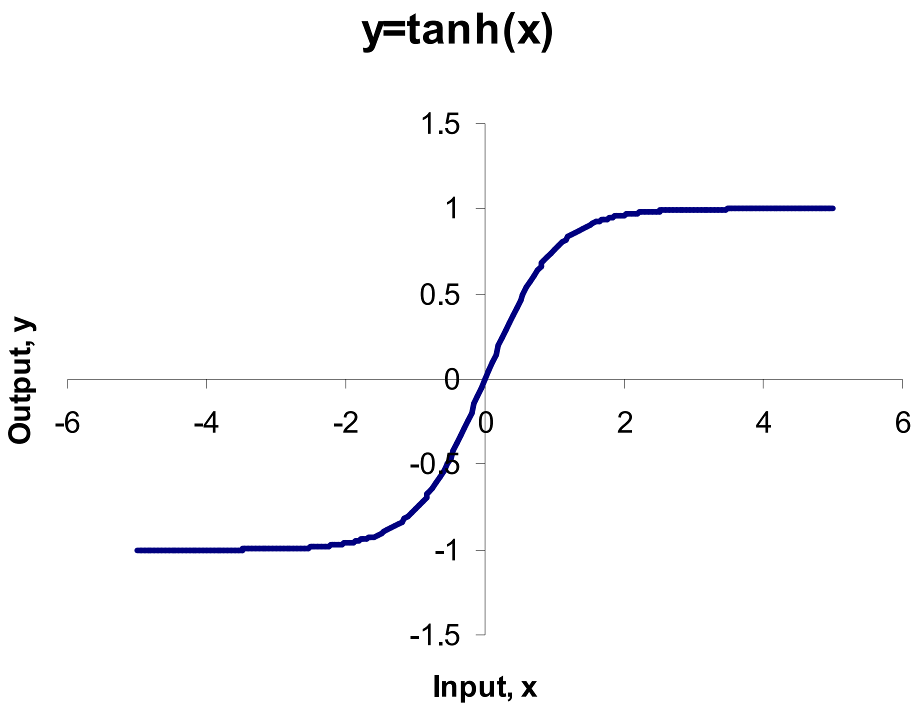

Figure 2 Non-linear threshold response

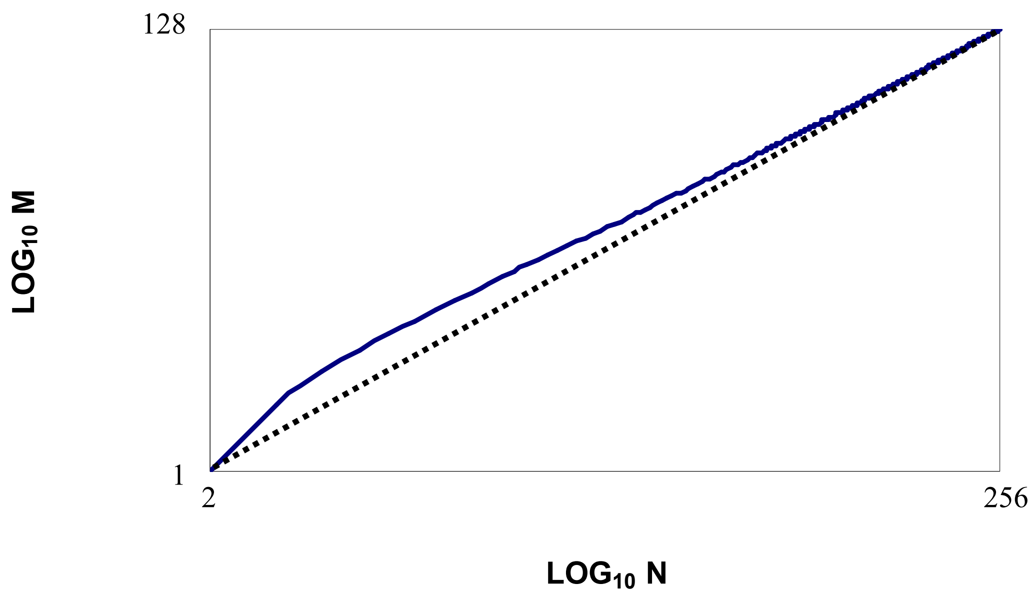

Figure 3 Memory capacity as a function of number of neurons in the novel net