CERN-TH.5758/90

NEURAL NETWORKS AND CALORIMETER CLUSTER RECOGNITION†

T. Altherr*
L.A.P.P., BP 110,F-74941 Annecy-le-Vieux CEDEX, France

and

J.C. Seixas*
TH Division, CERN, CH-1211 Geneva 23, Switzerland

ABSTRACT

We show how a very simple neural network algorithm could be used to analyse clusters obtained from a calorimeter. We further show that the algorithm is very fast and easy to implement in hardware.

---

† Talk presented at the International Workshop on Software Engeneering, Artificial Intelligence and Expert Systems for High Energy and Nuclear Physics, Lyon Villeurbanne (France), March 19-24, 1990.

* Also at Departamento de Física, IST, Portugal

CERN-TH.5758/89
May 1990

Neural networks have attracted much attention in the last years, as their potential to simulating the basic brain functions seems unmatched by any other approach to simulate intelligent systems. Despite all their successes in other fields,[1] neural networks seem to have aroused only recently the interest of particle physicists. High energy particle experiments are, however, an extremely powerful ground for testing the possibilities of these systems. In particular, data aquisition problems represent a fundamental challenge, since neural networks carry with them the promise of being fast and precise devices, thus representing an excellent solution for experimental online analysis.

In this paper we present a very simple network approach to first-level triggering of an electromagnetic calorimeter. In particular we show computer simulations of the performance of this network when applied to 2-dimensional plots obtained by computer simulation. The main aim is the recognition of the energy, width and position of clusters.

The network is composed of 7 layers, the architecture being strictly feed-forward (see Fig.1.a). Each unit acts essentially as a McCulloch-Pitts neuron,[2] the output and input values being structured such that a certain number of bytes correspond to the height of the pixel and additional bytes are used to measure the width of the cluster in a way that will be specified below. The algorithm works as follows: each unit j on a given layer looks at the 8 nearest neighbours in the previous layer, and compares the average input received $net_j = (\sum_{k \neq j} a_k)/8$ with the activation value a_j for the corresponding unit in the previous layer (see Fig.1.b). If $net_j > a_j$, unit j transfers its activation value to the unit in the next layer which corresponds to the position of the highest of its neighbours in the same layer. Meanwhile, the extra bytes are transferred exactly in the same way, when the activation value of a given unit reaches 0. The performance of the algorithm is different, however, for the different layers in the sense that it does not act with the same efficiency for all of them. In the first 4 hidden layers, activation for unit j passes from $a_j \rightarrow a_j - 1$ while for the winner unit a_k it passes from $a_k \rightarrow a_k + 1$. In the fifth hidden layer the efficiency is modified and it is the full activation value which is transfered. It is as if the first four layers acted as collimators for the fifth-layer analysis. The same is done for the extra byte. In Fig. 2 one can see the effect of the algorithm on an initial image composed of randomly generated Gaussian distributions.

The FORTRAN algorithm has been tested on a VAX 6000 and on an IBM 3090. Since the operations used are simply addition and comparison, the program is very efficient, and furthermore shows that a hardware implementation of this architecture might prove very easy and extremely fast. Copies of the program are available from ALTHERR@FRCPN11.

REFERENCES

[1] M. Caudill and C. Butler (eds.), Proceedings of the 1*st* IEEE International Conference on Neural Networks, San Diego, California USA, 21-24 June 1987.

[2] W. McCulloch and W. Pitts, Bull. of Math. Biophys. 5 (1965) 115.

FIGURE CAPTIONS

Fig. 1- Architecture of the neural network (a), and the connection with the nearest neighbours in a previous layer.

Fig. 2- Several steps in image processing within the network.

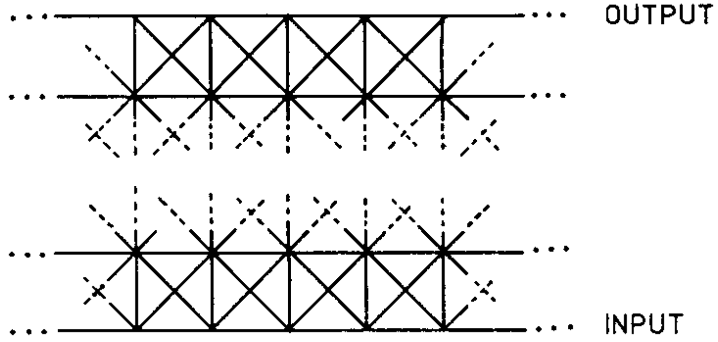


Fig. 1a

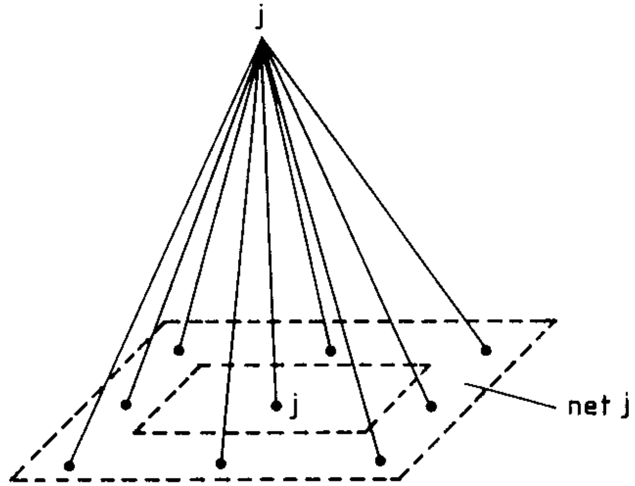
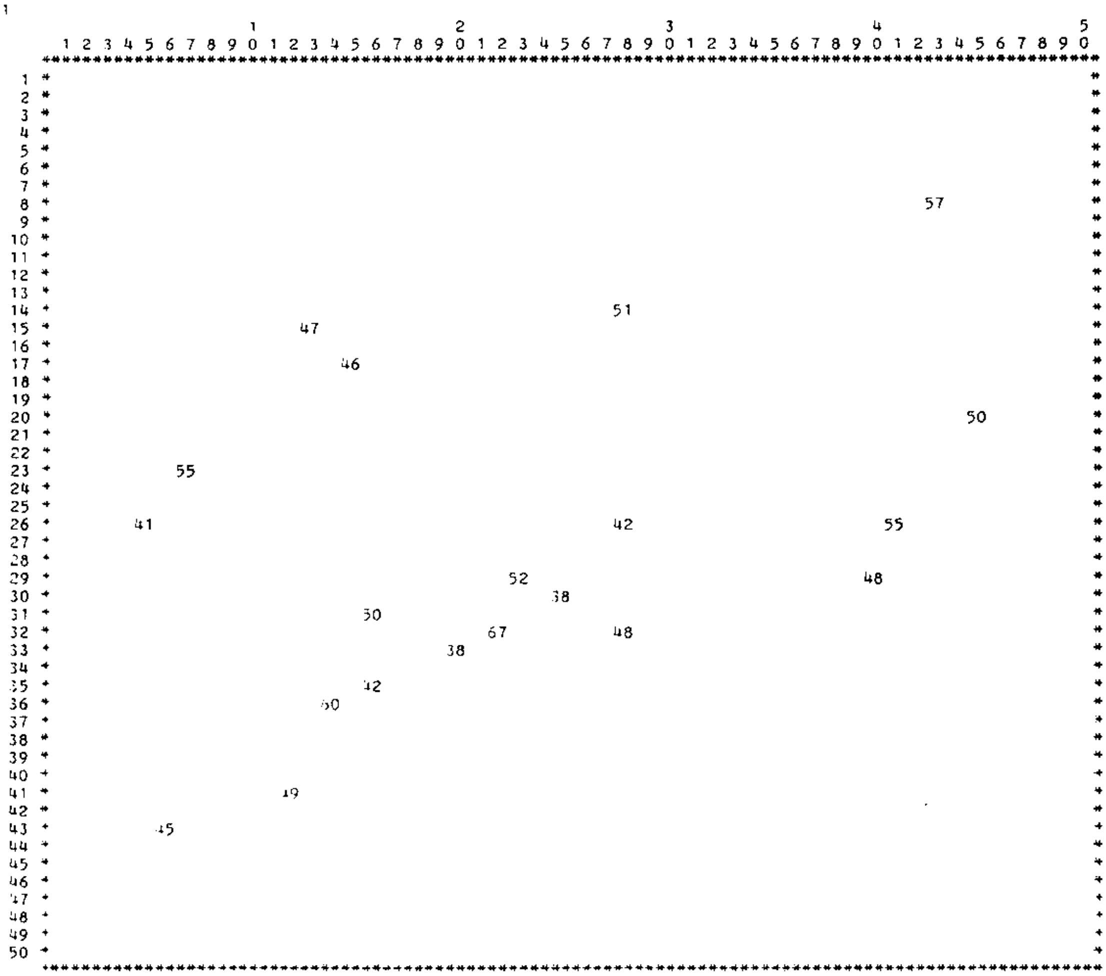
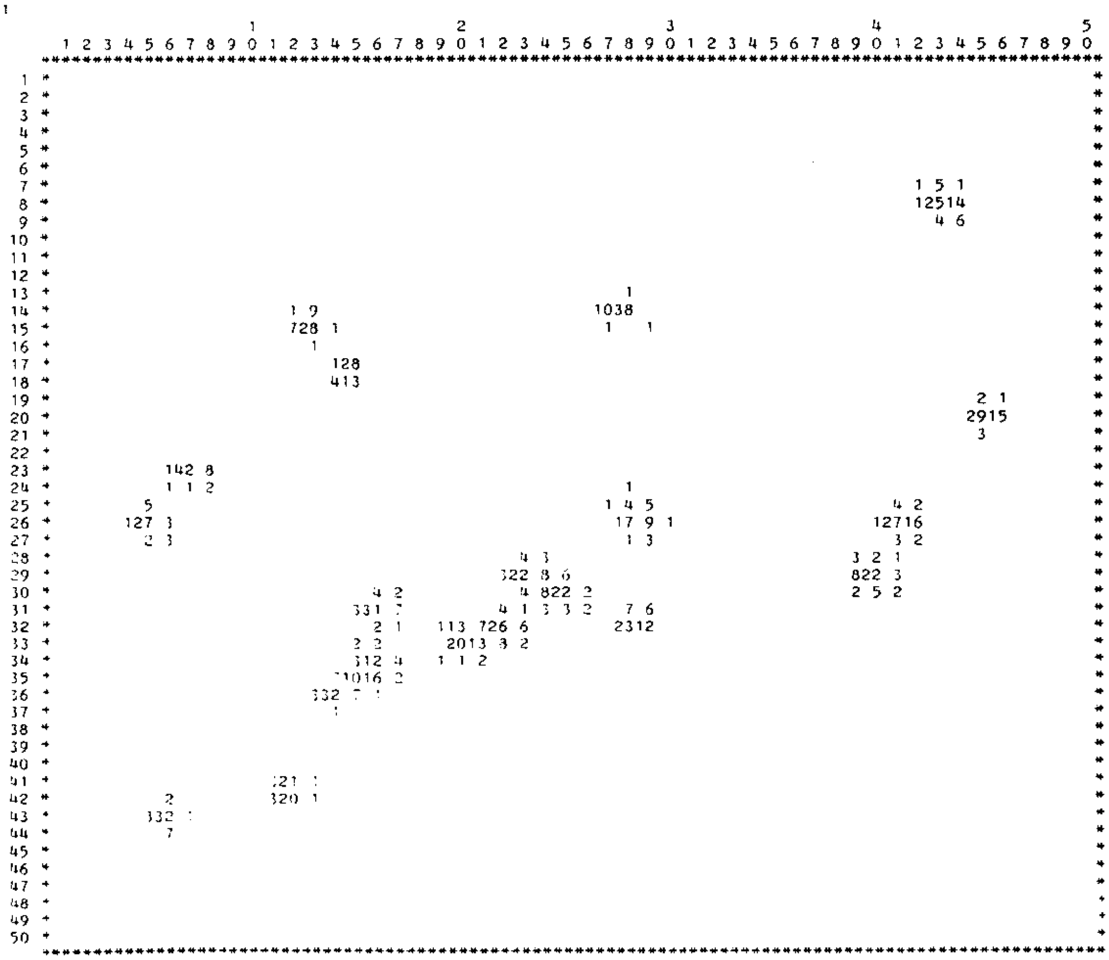


Fig. 1b



PATTERN AFTER WEIGTH TRANSFER

Fig. 2b



INITIAL PATTERN

Fig. 2a

Imprimé au CERN. Prix : 0.10 Fr.