Neural Implementation of Bayesian Vision Theories by Unsupervised Learning

Stelios M. Smirnakis and Alan L. Yuille *
Division of Applied Sciences,
Harvard University, Cambridge, MA 02138, USA.

* Please address correspondence to: Prof. A.L. Yuille, G12e Pierce Hall, Division of Applied Sciences, 29 Oxford Street, Cambridge, MA 02138. (617)495-9526. yuille@hrl.harvard.edu FAX (617) 495-9837.

Preferences: Categories : Theory and Analysis/Modeling and Simulation. Theme: Sensory Systems (Visual)/Learning and Memory/Development.

Abstract

It is often claimed that Bayesian theories of vision require time consuming, and biologically implausible, relaxation algorithms. We show that, on the contrary, Bayesian theories can be implemented by feedforward networks, multilayer perceptrons, where the weights of the network are trained by unsupervised learning using a novel variant of backpropagation. Both multilayer perceptrons and backpropagation are of questionable realism but our approach can be generalized to more biologically plausible models. We illustrate our theory on an example of image segmentation. Such unsupervised learning might have a role in the development of the visual system.

1 Introduction.

Biological organisms have to estimate properties of the world from visual signals. Many authors, see [1] and references therein, have suggested that Bayesian estimation theory gives a natural framework for visual perception. Bayesian models exist for such visual abilities as depth perception, object recognition, image segmentation and self-organization. Indeed all of cognition can, in principle, be formulated in Bayesian terms [1].

It is unclear, however, whether such theories can be implemented in a biologically plausible way. It is often claimed that such theories require relaxation algorithms which

[1] Regularization theory [2] can be obtained as a special case of Bayesian theories.

are time consuming and may require feedback loops, see [7] for a recent example of such a system.

We show that, on the contrary, Bayesian theories can be implemented by feedforward networks, multilayer perceptrons, where the weights of the network are trained by unsupervised learning using backpropagation. Multilayer perceptrons and the backpropagation algorithm were chosen for convenience in order to illustrate our point. Multilayer perceptrons are, at best, a weak approximation to real neural systems but they remain, at present, the best studied paradigm for such systems. Similarly, the original backpropagation algorithm seems biogically unrealistic but more recent implementations using feedback loops are more plausible [6]. Our theory can be implemented by more realistic models of this type. Note that our model requires the use of feedback loops during training only, but that it is strictly feedforward after learning.

We emphasize that our approach involves unsupervised learning where a teacher is not required, but instead the system self-organizes by conforming to a set of principles selected during its evolution [2]. Though a "teacher" is plausible for acquiring some visual abilities, it seems unlikely that it is available for all of them. Hence unsupervised learning is desirable.

2 Theory.

For a specific visual task we let S represent the properties of the world that we wish to extract and let I be the visual input. The goal of a Bayesian theory is to find an estimate, $S^*(I)$, of the world properties as a function of the input I. The criteria commonly used is to pick the S that maximizes the *a posteriori* probability of the stimulus S, $P_{post}(S|I)$, given by $P_{post}(S|I) = P_l(I|S)P_{prior}(S)/P(I)$, where $P_l(I|S)$ and $P_{prior}(S)$ are the likelikood function and the prior probability respectively [3] (see, for example [1]). Formally, this means $S^*(I) = \arg\max_S P_{post}(S|I)$ and is called the MAP estimator.

We will demonstrate that it is possible to determine a close approximation to $S^*(I)$ by a feedforward network after *unsupervised* training. This shows: (i) that is possible to approximate $S^*(I)$ very quickly, and (ii) that time consuming relaxation algorithms are not required to implement Bayesian, or regularization theories.

We assume that the function $S^*(I)$ can be approximated by a feedforward network with one layer of hidden units, provided the weights can be chosen appropriately . Theoretical results will guarantee this if we have enough hidden units [3]. We express the output of the network as $S = f(I;\omega)$, where ω represents the weights. The learning task is to determine the set of weights ω^* so that the network closely approximates $S^*(I)$.

To determine the correct weights we train the system over a representative set of inputs $\{I^\mu : \mu \epsilon \Lambda\}$. We pick ω^* to maximize the energy function $E[\omega, \Lambda] = \sum_{\mu\epsilon\Lambda} \log P_{post}(f(I^\mu;\omega)|I^\mu)$. In the limit as $|\Lambda|$ tends to infinity this energy function becomes:

$$E[\omega] = \sum_I P(I) \log P_{post}(f(I;\omega)|I). \tag{1}$$

[2] We are aware of work in progress (Hinton, personal communication) where a similar problem is tackled using a teacher.

[3] The form of these distributions is specified by evolution

Provided the class of input-output functions of our network includes $S^*(I)$, it is clear that $E[\omega]$ will be maximized by ω^* such that $f(I;\omega^*) = S^*(I)$ (recall that $S^*(I) = \arg\max_S P_{post}(S|I)$ and $P(I) \geq 0,\ \forall I$). If the class of functions is not representative enough we will still obtain the best approximation to the $S^*(I)$ within the class, under the assumption that the training process is capable of finding the optimal ω. We propose using stochastic training which, as recent results have shown [4], is resistant to local minima in the energy function. Observe that if P_{post} is specified by a Gibbs distribution, $P_{post} = (1/Z)e^{-E_{post}}$, then our criteria is equivalent to minimizing the expected value of the corresponding energy E_{post}.

In short, our approach involves using a regular backpropagation algorithm but with the standard error function being replaced by the function $\log P_{post}(f(I;\omega)|I)$.

3 Simulation Example: Image Segmentation

We now consider a specific example – the weak string/membrane model of image segmentation [5]. The goal of this model is to smooth noisy image while preserving, and detecting, intensity edges. We use a feedforward architecture with one input layer (I), feeding to a Hidden layer (H), feeding to the output layer (S).

The posterior distribution for the weak string model is specified by a Gibbs distribution with an associated family of energy functions $F^p(S)$:

$$F^p(S) = \sum_{i=0}^{N}(S_i - I_i)^2 + \sum_{i=1}^{N} g^p(S_i - S_{i-1}) \tag{2}$$

where :

$$\begin{aligned} g^p(t) &= \lambda^2(t)^2 \quad \forall|t| < q \\ &= \alpha - c(|t| - r)^2/2 \quad \forall q \leq |t| < r \\ &= \alpha \quad \forall|t| \geq r \end{aligned} \tag{3}$$

with $c = \frac{1}{2p}$, $r^2 = \alpha(\frac{2}{c} + \frac{1}{\lambda^2})$, $q = \frac{\alpha}{\lambda^2 r}$
Note that α and λ are fixed constants which correspond, respectively, to the cost incurred by a break and the length scale (in lattice cell width units) beyond which edge-edge interactions are insignificant. In our simulations we set $\alpha = 0.025$ and $\lambda = 5$.
The parameter p specifies the smoothness of the energy function ($p = 1$ is convex and $p = 0$ is nondifferentiable). Our variant of the GNC algorithm [5] consists of decreasing p from 1 to 0 in discrete steps, while minimizing F^p with respect to the weights for a certain number of iterations at each step by standard backpropagation, starting from the value of the weights obtained at the previous step.

3.1 Network Performance in 1D

We trained a network of three fully connected 32 unit layers on 1000 simulated intensity patterns of length 32. The network nodes were chosen to be sigmoidal with outputs lying in the range $[-0.5, 0.5]$. The input patterns were produced by a two stage process which

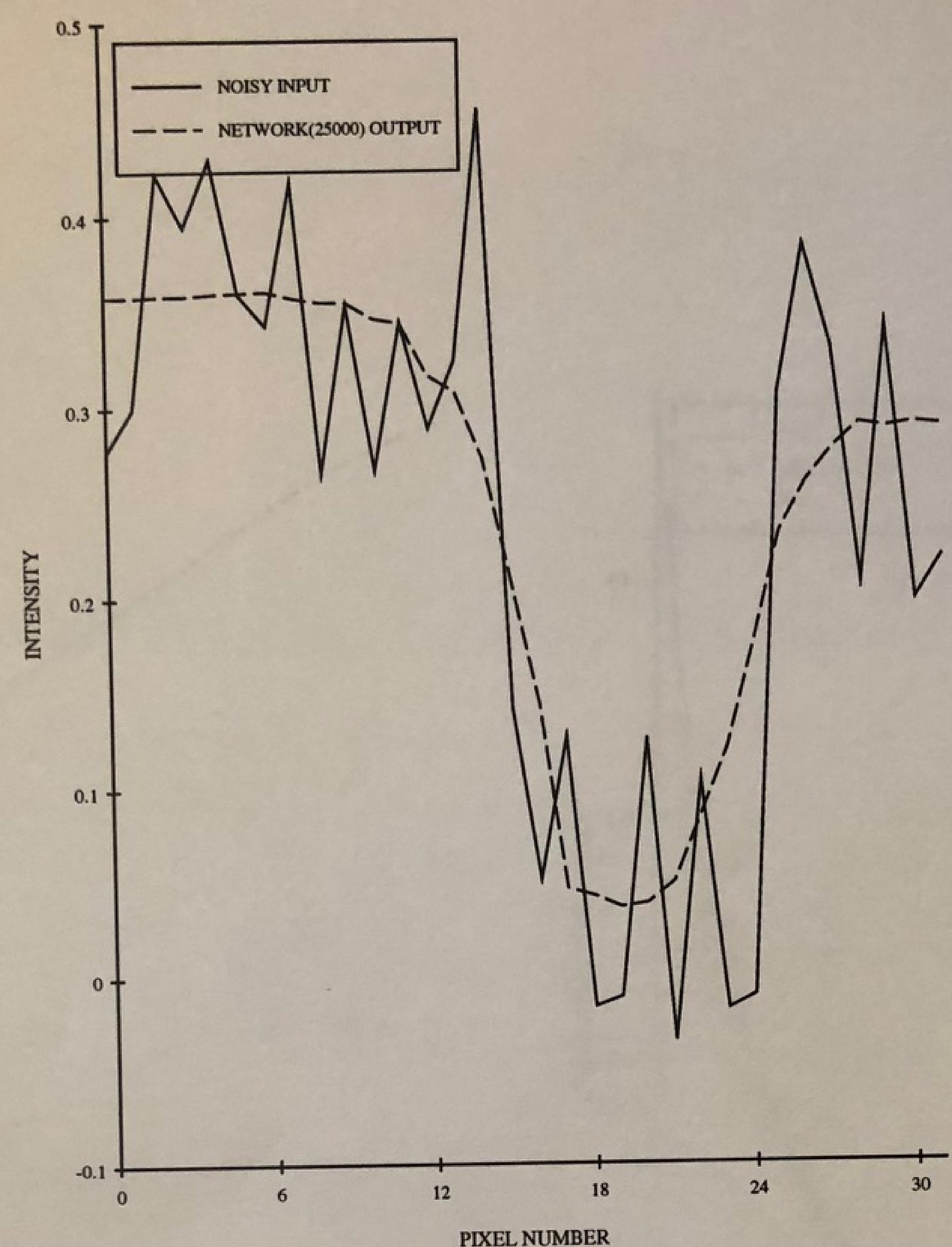

Figure 1: The dashed line is the output of the network after training when presented with noisy input stimulus (solid line). Note that the image is now smooth while the edges, and their locations, are preserved.

first generated piecewise constant patterns in the range $[-0.5, 0.5]$ and then added noise which was uniformly distributed in the range $[-0.2, 0.2]$. The noiseless piecewise constant patterns are the desired output of the system.

The resulting system was able: (i) to memorize the noiseless patterns in the dataset, (ii) to perform noise reduction on dataset patterns with additive noise, and (iii) to generalize to novel, noisy, stimuli by giving approximately piecewise constant output with edges correctly located. Figure 1 shows the performance of the network on a noisy stimulus and Figure 2 plots two measures of learning performance.

3.2 Results in 2D

We also implemented image segmentation by the weak membrane model in two dimensions. Our preliminary results on 12x12 pixel simulated 2-D data, see Figures 3 and 4, indicate that the network learns to memorize the training set, to perform noise reduction, and to generalize to edge detection on novel sets. The implementation is a straightforward generalization of the 1-D model. We again trained the network on 1000 randomly selected patterns.

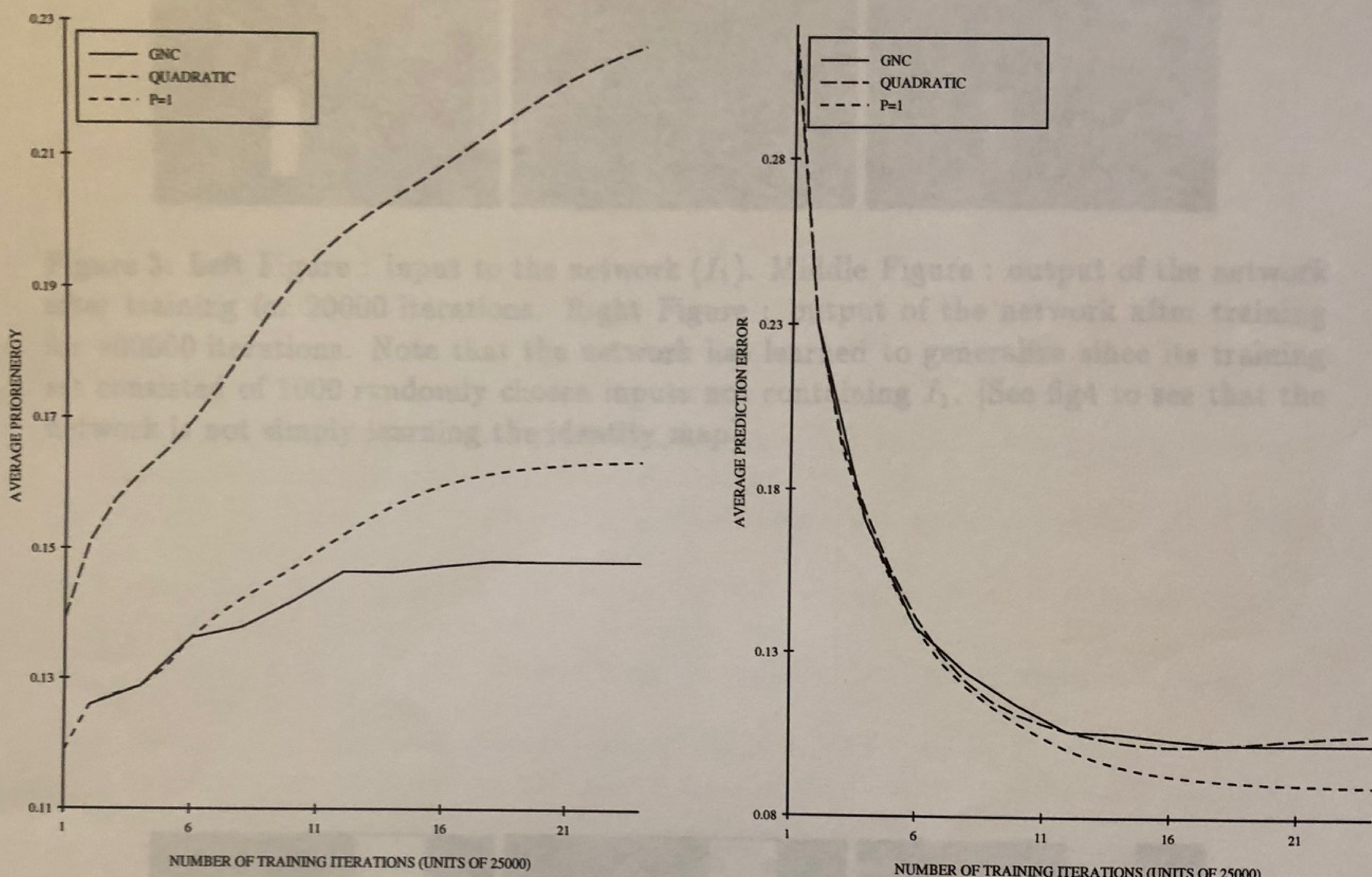

Figure 2: Plotting 1-D energy terms as a function of number of training iterations. We contrast networks trained by : [1] GNC (from $p = 1$ to $p = 1/8$), [2] $p = 1$ smoothness and [3] no smoothness [labeled Quadratic]. The left figure plots the average of the weak smoothness term : $\sum g^{p=1}$ over the outputs of 1000 noisy data not in the training set. Observe that the energy initially increases since the initial randomization of the weights gives smooth outputs, but the GNC trained network plateaus at a low level. The right figure plots the average of $\sum (S - D)^2$, where S is the network output and D is the desired output.

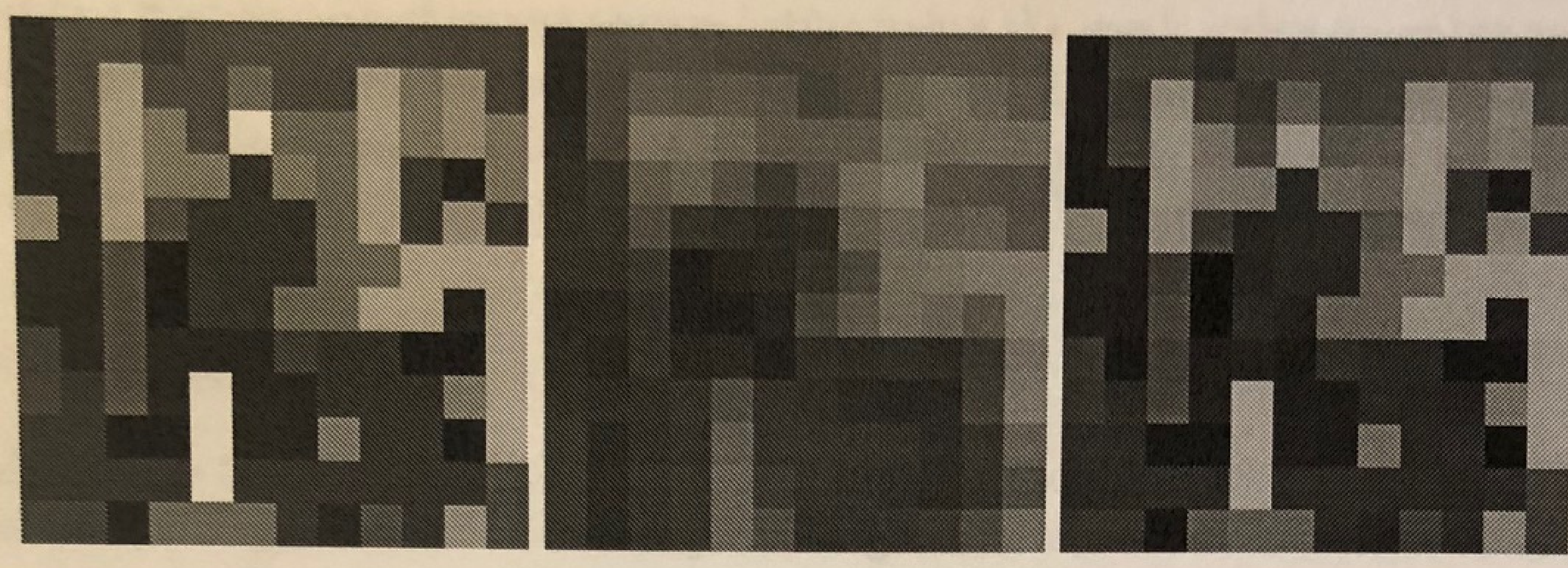

Figure 3: Left Figure : Input to the network (I_1). Middle Figure : output of the network after training for 20000 iterations. Right Figure : output of the network after training for 400000 iterations. Note that the network has learned to generalize since its training set consisted of 1000 randomly chosen inputs not containing I_1. [See fig4 to see that the network is not simply learning the identity map]

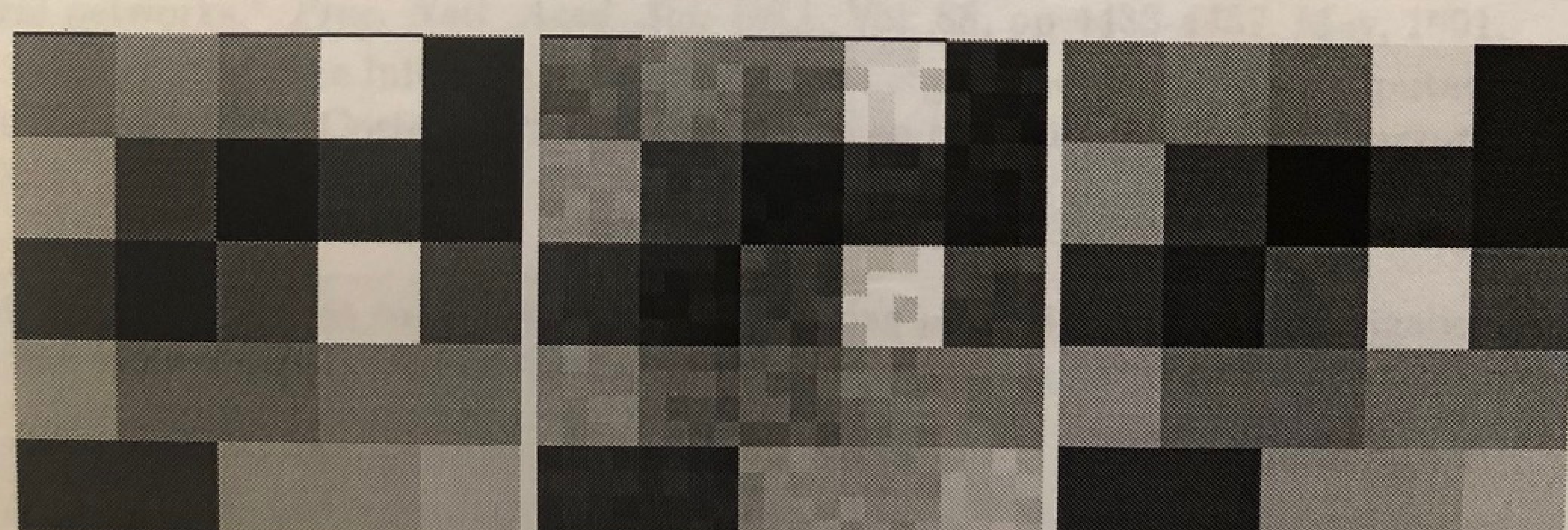

Figure 4: Left Figure : noisless input pattern (I). Middle Figure : I with uniform noise of amplitude 0.1 added independently on each pixel. Right Figure : output of the network after training for 20000 iterations on a small set of noisless data containing I.

4 Conclusion

Our theoretical results showed that a backpropagation network can be trained, by self-organization, to approximate the MAP estimator of a Bayesian theory. Simulation results confirmed this for the case of image segmentation using the weak string/membrane model. Such unsupervised learning might have a role in the development of the visual system.

These results show that it is possible to implement Bayesian theories in a feedforward manner which is both faster and more biologically plausible than relaxation algorithms [4].

Acknowledgements

We would like to thank DARPA for an Air Force contract F49620-92-J-0466. The neural network was implemented in the Aspirin/Migraines environment, 5th release, freely available through Mitre Corporation as a Neural Network Research tool.

References

[1] J.J. Clark and A.L. Yuille. **Data Fusion for Sensory Information Processing Systems.** Kluwer Academic Press. Boston/Dordrecht/London. 1990.

[2] T. Poggio, V. Torre and C. Koch. "Computational vision and regularization theory". *Nature*, 317, pp 314-319. 1985.

[3] K. Hornik, S. Stinchocombe and H. White. "Multilayer feed-forward networks are universal approximators". *Neural Networks* 4, pp 251-257. 1991.

[4] T.K. Leen and J.E. Moody. "Weight Space Probability Densities in Stochastic Learning". *NIPS 5*. pp 451-458. 1993.

[5] A. Blake and A. Zisserman. **Visual Reconstruction.** MIT Press. 1987.

[6] P. Mazzoni, R.A. Andersen and M.I. Jordan. "A more biologically plausible learning rule for neural networks." *Proc. Natl. Acad. Sci. USA*. Vol. 88, pp 4433-4437. May, 1991.

[7] Tai Sing Lee. "Surface Inference by Minimizing Energy Functionals: A Computational Framework for the Visual Cortex." PhD Thesis, Division of Applied Sciences, Harvard University, 1993.

---

[4] Note that feedback might be needed during the learning phase, to make backpropagation biologically plausible, but it is not needed after training.