Regularized Parameter Estimation in an Inhomogeneous Cellular Network

Lars Nonboe Andersen and Lars Kai Hansen
CONNECT, Electronics Institute, build. 349
Technical University of Denmark,
DK-2800 Lyngby, Denmark
email: nonboe,lkhansen@ei.dtu.dk

Extended summary v3.0 – NNSP'95

1 Introduction

The *Maximum A Posteriori* (MAP) approach has found ample use in signal processing [12, 4]. When applied to image data it leads to algorithms that map well onto networks of locally connected, simple processing elements ie. cellular neural networks [2]. Furthermore, we find that the MAP approach in a very convenient way allows for specifiction of qualitative and flexible priors. Using adaptation schemes the parametrized priors may be trained to optimal performance in a particular environment, see e.g., [3, 6]. In most applications, sofar, the MAP approach has been used to derive homogeneous Markov Field models. In many image processing applications the visual field is inhomogeneous, e.g., in sonar or radar imagery where the two axes implement deflection and range respectively. In this note we formulate simple inhomogeneous MAP models hence, involving space-variant parametrizations of the prior. We use the Boltzmann Machine learning rule [8] for parameter estimation and augment the learning rule, using the Bayesian formulation, to incorporate prior information on the hyper-parameters [10] .

Geman and Geman [4] introduced Metropolis sampling from Gibbs distributions of Markov Random Fields as a simulation tool for visual reconstruction and argued that sampling by *Simulated Annealing* [9] improves the speed of the sampling process. Such sampling implements a stochastic neural network with symmetric connections that can be taught by the *Boltzmann Machine* learning rule[8, 6]. Peterson and Anderson [11] introduced the *Mean Field* approximation, and obtained substancial improvements in speed and performance. This is in line with the earlier observations in the physics community where the mean field approximation is used to produce quick and often surprisingly accurate estimates of ordered phases in statistical mechanics systems [7].

In the next section we design the cellular networks for segmentation using the Mean Field approach. In section three we show how the Boltzmann Machine learning rule may be augmented by priors on hyperparameters and the scheme is applied for parameter adaptation in the inhomogeneous case. Section four contains experiments on image segmentation and concluding remarks.

2 Image Segmentation

Segmentation is important to most computer vision systems, however, even in its simplest form: binarization of a grey-scale image there exist no established standard practise. Here we use the Bayes scheme to derive a cost function which in turn is minimized by a cellular neural network. The resulting cost function is a generalization of the one used by Carnevali *et al.* [1], to allow for spatial dependence in the cost function parameters (coupling constants). The target signal is a smooth binarization of a grey-scale image d_j, in terms of two-valued pixels $S_j \in \{-1, +1\}$. The prior distribution is designed to emphasize smoothness:

$$P[S] = Z_1^{-1} \exp\left(-\frac{1}{2} \sum_{j=1}^{N} \sum_{j'=1}^{N} M_{j,j'} (S_j - S_{j'})^2 \right) \tag{1}$$

$M_{j,j'}$ defines the connectivity, hence the unit cell of the cellular network. Here we just use coupling (with strenght w_n) to the nearest neighbors, but we might allow for the nearest neighbor coupling to vary as function of the spatial location. More complex connectivity structures have been designed for modeling textural features [3].

We assume the signal degradation to consist in addition of white, space variant, Gaussian noise. This degradation process leads to the following conditional distribution:

$$P[d|S] = Z_2^{-1} \exp\left(-\frac{1}{2} \sum_{j=1}^{N} \alpha_j (S_j - d_j)^2 \right) \tag{2}$$

We want to approach real data for which the noise statistics, here represented by the space variant parameter α_j[1]. are unknown, hence has to estimated as part of the learning process. We use Bayes to combine and obtain the posterior distribution of the Markov Random Field,

$$P[S|d] = Z^{-1} P[d|S] P[S]. \tag{3}$$

Clearly this distribution is of the Gibbs form[2], with a cost function given by the negative logarithm of the posterior distribution: $-\log P[S|d]$. We further note that the state dependent part of the cost function is linear in the parameters $M_{j,j'}$ and α_j.

2.1 Network Design

The Mean Field annealing method for estimation of averages over Gibbs distributions is well documented in the litterature see e.g. Hertz *et al.* [7]. The cellular neural network is designed to minimize the Mean Field *free energy* F,

$$\tau \frac{\partial \langle S_j \rangle}{\partial t} = -\frac{\partial F}{\partial \langle S_j \rangle} = -\langle S_j \rangle + \tanh\left(\beta^t \left(\sum_{j'=1}^{N} M_{j,j'} \langle S_{j'} \rangle + \alpha_j d_j \right) \right). \tag{4}$$

β^t quantifies the annealing schedule, in this work we use the simple schedule: $\beta^t = \beta^1 + (t/t_{max})(\beta^2 - \beta^1)$. In summary, the unit cell of the cellular network contains one unit approximating the local thermodynamic average: $\langle S_j^t \rangle$, and one input unit d_j. We assume that $M_{j,j'}$ connects nearest neighbors symmetrically with weight w_n, and the weights to the input units d_j are given by α_j.

[1] For Gaussian noise we have $\alpha_j = 1/\sigma_j^2$

[2] A distribution of the form $P(x) = Z^{-1} exp(-E(x)/T)$, where $E(x)$ is a cost function, bounded from below, and T is a parameter

3 Boltzmann Machine Learning

In order to apply the cellular network we need to estimate the parameters (denoted $w = (w_n, \alpha_j)$). Since our network is based on the Gibbs distribution we invoke the Boltzmann Machine learning rule [8, 11, 7]. The objective of the Boltzmann Machine learning rule is to minimize the Kullback information distance between a target distribution $P_{w^*}[S|d]$, (of which the training set is a finite sample), and the distribution $P_w[S|d]$ sampled by the current (stochastic) network with parameters w. *It is worth noting that if the empirical distribution is substituted for the target distribution, the Kullback information is in fact the normalized likelihood of the parameters.*

The learning rule is formulated for a general system specified in terms of inputs, hiddens, and outputs: (d, h, S). Since the states of the hidden units are unknown for the learning examples, we compare the marginal distributions, i.e. the distributions integrated over the hidden variables:

$$\Delta[P_{w^*}, P_w] = \int DdP(d) \int DS \int DhP_{w^*}[S,h|d] \log \frac{\int DhP_{w^*}[S,h|d]}{\int DhP_w[S,h|d]} \tag{5}$$

$\int DS$ is short for $\int \prod_j dS_j$. The learning algorithm is derived by gradient descent minimization of the information distance. The recursive learning algorithm reads:

$$w_\nu^{n+1} - w_\nu^n = \eta\beta \left[\langle \Omega_\nu (S,h,d) \rangle_{clamped} - \langle \Omega_\nu (S,h,d) \rangle_{free} \right] \tag{6}$$

for any Gibbs distribution with a cost function linear in the parameters w_ν:

$$E(S,h,d) = -\sum_\nu \Omega_\nu (S,h,d)\, w_\nu, \tag{7}$$

where $\Omega_\nu (S,h,d)$ is an expression in the stochastic variables. $\langle ... \rangle_{clamped}$ indicates that the average is performed with respect to the current Gibbs distribution, with *fixed* values for all the stochastic variables that are specified as *input or output* variables in an example of the database, i.e. (S, d). Similarly $\langle ... \rangle_{free}$ is the average with only *input* variables fixed. In brief we can characterize the learning process as follows: the parameters are adjusted to minimize the difference between the correlations in situations with and without the teacher specifying the correct output [8, 7].

We can introduce prior information on the parameters w in the Boltzmann Learning rule by going through a Bayesian procedure similar to the one that the led to the posterior distribution of the Markov Field, see also [10]. We make the observation that the distance $\Delta[P_{w^*}, P_w]$, apart from a additive constant, is identical to the negative log likelihood of the Markov Field *parameters* conditioned on a given database D of input-output examples,

$$\Delta[P_{w^*}, P_w] = -\log P(D|w) + \text{const.} \tag{8}$$

Using Bayes formula we find the posterior distribution of the parameters,

$$-\log P(w|D) = \Delta[P_{w^*}, P_w] - \log P(w) + \text{const.} \tag{9}$$

With $P(w)$ being the prior on the Markov Random Field Parameters. This leads to a generalized Boltzmann Machine learning rule (n being the iteration index):

$$w_\nu^{n+1} - w_\nu^n = \eta\beta \left[\langle \Omega_\nu (S,h,d) \rangle_{clamped} - \langle \Omega_\nu (S,h,d) \rangle_{free} \right] + \eta \frac{\partial \log P(w)}{\partial w_\nu}. \tag{10}$$

4 Experimental evaluation and concluding remarks

We illustrate the performance of the procedure on the artificially generated multi-object scene shown in the left panel of figure 1. Space variant noise was added to produce the degraded image in the right panel

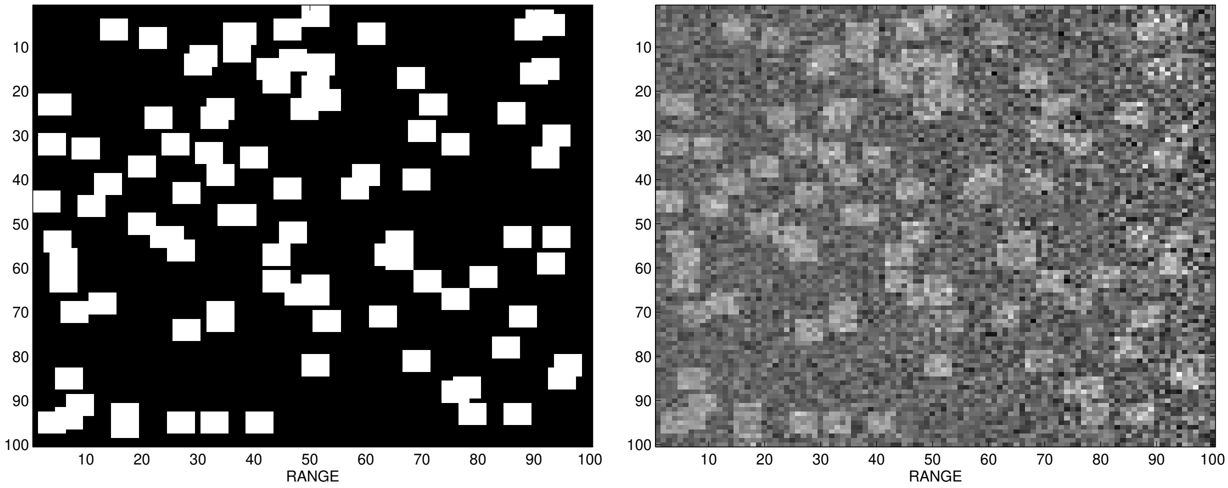

Figure 1: Left: Multiple object scenery. Right: Scene degraded by additive white noise of spatially varying variance (increasing in the "range" direction).

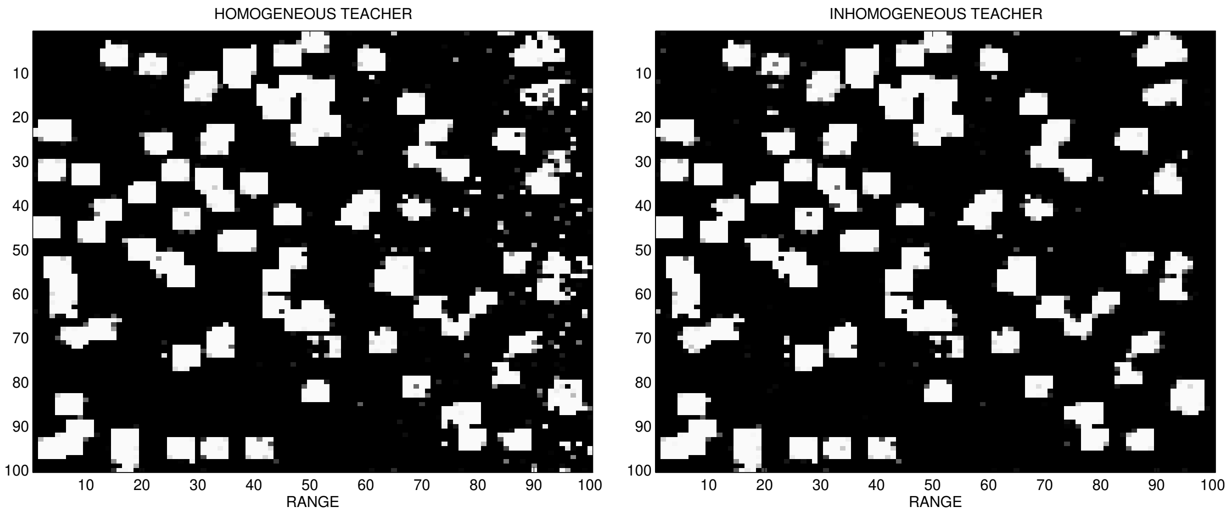

Figure 2: Left:Reconstructed scene by the "teacher net" with constant noise variance parameter α. Right: Reconstructed scene by a teacher net with spatially variant noise parameter (decreasing linearly in the range direction).

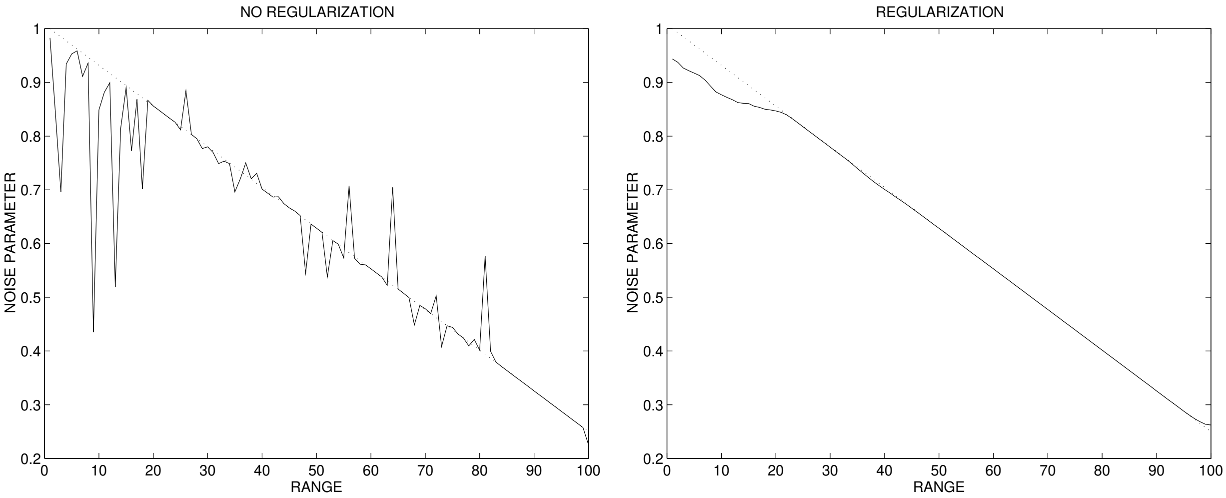

Figure 3: Left: Student learning from the teacher with spatially variant noise parameter but with no regularization (the teacher α is marked with the dotted line). Right: Student learning from the teacher with regularization enforcing smoothly variyng noise parameter (the teacher α is marked with the dotted line). .

of figure 1. To produce the output for supervised learning a "teacher network" with spatially varying noise parameter, α_k, where k index the range direction, was applied to the noisy scene producing the right image in figure 2. First, a "student network" was trained in supervised mode with output as produced by the teacher. The student network was initialised with uniform noise parameters, $\alpha_k = 0.5$. The resulting range dependence, after 100 iterations, is shown in figure 3, left panel. Secondly, we included a prior enforcing smoothness of the spatially varying noise parameter,

$$P(\alpha) \propto \exp\left(-\frac{1}{2}\sum_k \mu_x(\alpha_{k+1}-\alpha_k)^2\right). \tag{11}$$

This leads to the modified learning rule,

$$\alpha_k^{n+1} - \alpha_k^n = \eta\beta\left[\langle S_k \cdot d_k\rangle_{clamped,deflect} - \langle S_k \cdot d_k\rangle_{free,deflect}\right] + \eta\mu_x\left(\alpha_{k+1}+\alpha_{k-1}-2\alpha_k\right) \tag{12}$$

The average $\langle ...\rangle_{clamped,deflect}$ includes an average with respect to the, "deflection", direction orthogonal to the range direction. Also, we used that the Ω-operator conjugate to the noise parameter, α_k is the product of the input and output pixels $S_k \cdot d_k$ at the particular location k. 100 iterations of the regularized training procedure leads to smooth adaptation of the inhomogenous network as shown in the right panel of figure 3.

In conclusion we have shown that cellular neural networks may be generalized to allow for inhomogeneous parametrizations. With an inhomogenous parametrization it is not possible to hand-tune the parameters as is custom in most applications of the Weak Membrane. However, we demonstrated that the Boltzmann Machine learning rule can be used to adapt the inhomogeneous network and, furthermore, may be generalized to incorporate prior information on the parameters of the associated Gibbs distribution.

Acknowledgement

This research is supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center.

References

[1] P. Carnevali, L. Coletti and S. Paternello: *Image processing by simulated annealing.* IBM Journal of Research and Development **29**, 569-579, (1985).

[2] L.O. Chua and L. Yang: *Cellular Neural Networks: Theory.* IEEE Transactions on Circuits and Systems **35**, 1257-1272, (1988).

[3] G.R. Cross and A.K. Jain: *Markov Random Field Texture Models.* IEEE Transactions on Pattern Analysis and Machine Intelligence **5**, 25-39, (1983).

[4] D. Geman and S. Geman: *Stochastic Relaxation, Gibbs distributions and the Bayesian restoration of images.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6**, 721-741, (1984).

[5] L.K. Hansen: *Boltzmann Learning of Parameters in Bayes Visual Reconstruction.* Proceedings of the First Danish Conference on Pattern Recognition and Image Analysis. Ed.: S.I.Olsen. Department of Computer Science, University of Copenhagen, 92/8, (1992).

[6] L.K. Hansen: *Boltzmann Learning of Parameters in Cellular Neural Networks.* Proceedings of Second Int. Workshop on Cellular Neural Networks and Applications CNNA'92, Munich (1992). IEEE Service Center, Piscataway NJ, 62-67, (1992).

[7] J. Hertz, A. Krogh and R.G. Palmer: *Introduction to the Theory of Neural Computation.* Addison Wesley, New York (1991).

[8] G.E. Hinton and T.J. Sejnowski : *Learning and Relearning in Boltzmann Machines.* In Parallel Distributed Processing, vol. 1, chap 7. Eds. Rumelhart D.E. et al. Cambridge: MIT Press.

[9] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi: *Optimization by Simulated Annealing.* Science **220**, 671-680, (1983).

[10] D. MacKay: *Bayesian Interpolation.* Neural Computation **4**, 415-447, (1992).

[11] C. Peterson and J.R. Anderson: *A Mean Field Learning Algorithm for Neural Networks.* Complex Systems **1** 995-1019, (1987).

[12] M.W. Roth: *Survey of Neural Network Technology for Automatic Target Recognition.* IEEE Transactions on Neural Networks **1**, 28-43, (1990).