Proceedings of the IEEE 7th Int. Conference on Tools with
Artificial Intelligence, Nov.5-8, 1995, Herndon, Virginia, pp. 138-141

A Spectral multi-resolution Image Encoding Network

Rüdiger W. Brause, Jürgen Glitsch
J. W. Goethe-University, Computer Science Dep.,
Frankfurt a.M., Germany
brause@informatik.uni-frankfurt.de

1 Abstract

After a short introduction into traditional image transform coding, multirate systems and multiscale signal coding the paper focuses on the subject of image encoding by a neural network. Taking also noise into account a network model is proposed which not only learns the optimal localized basis functions for the transform but also learns to implement a whitening filter by multi-resolution encoding. A simulation showing the multi-resolution capabilitys concludes the contribution.

2 Introduction

In the sensor encoding area, all approaches try to minimize the necessary information for a given reproduction error. One actual, important approach is the *transform coding* concept that is the base for the JPEG and MPEG image encoding standards [3], [5]. This concept sees the pixels of an image as parallel signals which have to be encoded. For this purpose, the picture is subdivided into subimages (e.g. 8x8 pixels) and transformed by a linear transform into coefficients. Afterwards, the code coefficients are quantized according to a quantization table. For the reconstruction process, these procedures are inversely done. In figure 1 the encoding and decoding situation is visualized.

The transforms are based on the decomposition

$$\mathbf{x} = \sum_{i=1}^{n} y_i \varphi_i(\mathbf{x}) \qquad (2.1)$$

for a vectored signal x and the basis functions ji. It is well known that transform coding minimizing the least mean squared error (LMSE) for the reproduced images can be obtained by the Karhunen-Loève expansion (KLT) or principal component analysis (PCA), which can be implemented by lateral inhibited neural networks, e.g [1]. Nevertheless, the JPEG and MPEG standard uses the cosine transform, (DCT) a special kind of Fourier transform (DFT) as basis functions φ_i which become basis vectors in the descrete case.

They all share the property that the part of the sampled data, the block, is of equal length for all basis functions. Since the different basis functions are sinus or cosines of different frequencies, this means that we sample the sensor data with the same inter-sample distance for different frequencies. According to the sample theorem of Shannon [10] , a signal containing a certain highest frequency f can only be recontructed if the sample frequeny f_s (i.e. the sampling distance), is doubled $f_s \geq 2f$ (i.e. the sampling interval is smaller than half of the period of f). So, by choosing a small sampling interval for a reliable reconstruction we oversample all the lower frequency components which results in highly correlated and thus redundant

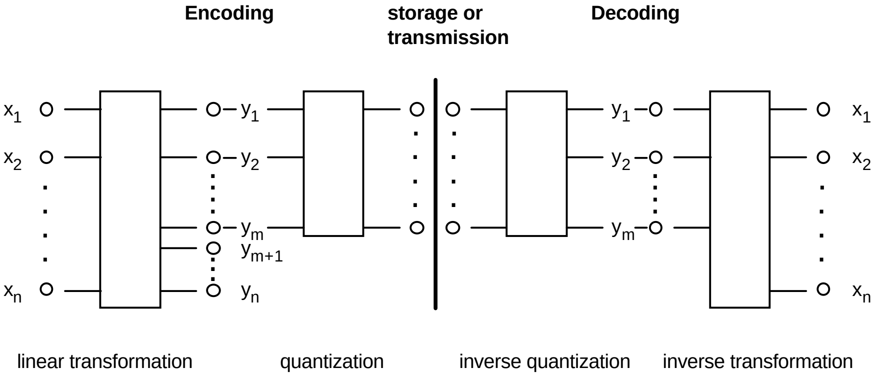

Fig. 1 *The transform coding approach*

coefficients even for the PCA case. This can be changed by a spectral multi-resolution approach where each frequency component has ist own sampling block size. This spectral multiresolution idea is also reflected by the well known subject of *multirate filter systems*, see e.g. the book of Vaidyanathan[11]. Here, we start with the approach of deviding the power spectrum |y(f)| of a signal x(t) into several overlapping intervals or *subbands* by the linear decomposition of basis functions with different frequency characteristics (*filter banks*). In figure 2a a filter bank system and in figure 2b the frequency responses of the different encoded signals y_i are shown.

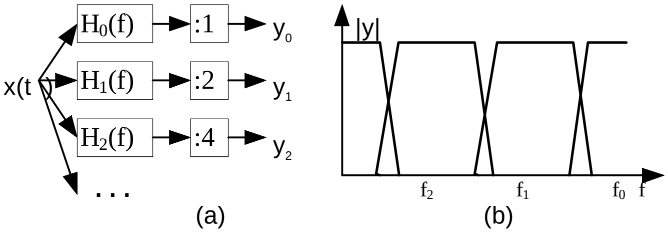

Fig.2 *Filter banks and subbands for multirate sampling*

According to the Shannon theorem, we can save sample points (and therefore encoding coefficients) by decreasing the sampling rates for the lower frequency bands, e.g. by a factor of two. If we arrange this parallel scheme in a sequential manner using only one type of filter bank which consists of two symmetrically aranged, overlapping filter banks (a high-pass filter and a low-pass filter: Quadrature Mirror Filter QMF-Filter[11]) we get the wavelet approach [4].

Since each filtered signal is subsampled, the corresponding part of the original signal is scaled (compressed) on the time scale. Thus, the corresponding basis function have to be rescaled (expanded) to represent the real basis function. The corresponding sampling interval is therefore also expanded, resulting in a different interval, i.e. in image encoding a different area surface, for each basis function.

Additionally, in many sensor encoding tasks the ability to deal with the noisy environment of the transmission or storage is of crucial importance. It can be shown [2] that for many parallel channels the information flow is maximized if the linear transformation produces decorrelated and normalized channel output which coincidents well with the classical result of a *whitening filter* by Shannon [9] for one channel. This means that we have no longer to implement a PCA, but to decorrelate and normalize the output data which can be done by an infinity of base vector systems. Among them, a PCA with scaled eigenvectors is just one sufficient solution, not a necessary one.

So, we can finally conclude that we are looking for an encoding scheme that uses non-uniform sampling and orthonormalizes the output data at the same time. For this task, neural networks are good candidates.

3 A multi-resolution network model

For our multi-resolution model, let us first specify the activity in the network.

3.1 The activity model

The linear activity due to eq.(1.1) can be easily implemented by a net of n linear neurons each implementing a scalar product

$$y = \sum_{j=1}^{n} w_j x_j = \mathbf{w}^T\mathbf{x}. \qquad (2.1)$$

As a network they implement a linear transform by their parallel action

$$\left.\begin{aligned} y_1 &= \mathbf{w}_1^T\mathbf{x}\\ &\cdots\\ y_n &= \mathbf{w}_n^T\mathbf{x}\end{aligned}\right\} = \mathbf{Wx} \qquad (2.2)$$

This is shown in figure 3 by the solid lines.

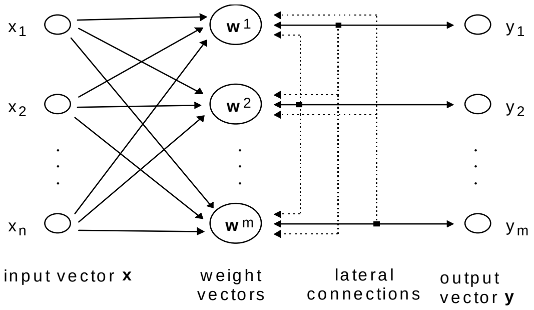

Fig. 3 *A symmetrically lateral inhibited network model*

Now, how can we get the vectors orthogonal ?

The ordinary PCA networks minimize the error of the reconstruction by assigning to the weight vectors the eigenvectors of the input covariance matrix $\mathbf{C}_{XX}$ which forms an orthogonal base. The main difference between the different models is given *how* the assignment takes place. This is done by learning rules which induce a special learning network, i.e. a network for the signal flow exclusive for the learning phase.

3.2 The learning network

The learning rules of the models can be devided into three categories: models using symmetric lateral inhibitions (as e.g. [1], see dotted lines in figure 3), models using asymmetric lateral inhibiton influence (e.g. [7]) and models without lateral inhibition, but using a kind of activity backpropagation through the weights influencing the signal source (e.g. [6],[8]). The first ones have the advantage that they are biologically plausible (lateral inhibition connections are found also in biological nervous systems) and can easily be implemented by analog circuits, because it uses only one line of feedback per neuron contrary to the signal influencing feedback of the latter ones.

So, let us choose a lateral inhibition model, for instance the one of Brause [1]. Here, our objective function that implements a PCA for the learning model is

$$R\left(\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^m\right) = \underbrace{\frac{1}{4}\beta \sum_{i=1}^{m} \sum_{j=1,\, j\neq i}^{m} \left\langle y_i\, y_j \right\rangle^2}_{R_1} - \underbrace{\frac{1}{2}\sum_{i=1}^{m} \left\langle y_i^2 \right\rangle}_{R_2} \tag{2.3a}$$

accompanied by the restriction for obtaining normalized weights

$$|\mathbf{w}^k| = 1 \tag{2.3.b}$$

The objective function is composed by two terms R_1 and R_2. The first term becomes zero only when all crosscorrelation terms c_{ij} are zero while the second term become strongly negative for the variance getting maximal. Since the extrema of the objective function, even scaled by an arbitrary factor, remains the same the factor β denotes only the relative influence of the crosscorrelations with respect to the autocorrelation influence.

Now we let the weights of this feedforward network learn by the simple gradient descent learning rule

$$\mathbf{w}^k(t) = \mathbf{w}^k(t-1) - \gamma(t)\, \nabla_{w^k}\; R\left(\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^m\right)$$

$$k = 1, 2, \ldots, m$$

with the learning rate γ and the Nabla operator ∇ for the gradient.

With the gradient we can directly compute the deterministic learning rule for the k-th neuron. Introducing *lateral coupling weights* which are often observed in biological nervous circuitry

$$u_{ij} = -\, \langle y_i y_j \rangle \qquad \textit{lateral inhibition} \tag{2.4}$$

between the neurons for the learning process we finally get as the stochastic learning rules

$$\tilde{\mathbf{w}}^k(t) = \mathbf{w}^k(t-1) + \gamma(t)\, \mathbf{x} \left[\beta \sum_{i=1, i\neq k}^{m} u_{ki} y_i \; + y_k \right] \tag{2.5a}$$

$$\text{and} \quad \mathbf{w}^k(t) = \frac{\tilde{\mathbf{w}}^k(t)}{\left|\tilde{\mathbf{w}}^k(t)\right|} \qquad \textit{normalization} \tag{2.5b}$$

Please note that all weight vectors have a different number of dimensions; since they are only coupled by the scalar output y_i, the different dimensions do not imply any problems.

For noise suppression, the covariance coefficients have to become [2]

$$c_{ij} = \left\langle y_i\; y_j \right\rangle = \begin{cases} 0 & i \neq j \quad \text{(decorrelation)} \\ P & i = j \quad \text{(normalization)} \end{cases} \tag{2.6}$$

In contrast to the ordinary orthonormal networks, we do not change the length of the weight vectors to fit the variance P by a proper objective function (see [2] , but we introduce a second learning mechanism for the area (number of components) that the weight vector covers. By correspondingly increasing the image area, also the variance represented by the neuronal output is increased. Thus, we change from the equi-resolution encoding to the multi-resolution approach.

The mechanism can be realized by a simple gradient descent on the mean squared error

$$n(t+1) \;=\; n(t) \;-\; \gamma(t)\left(\left\langle y(n)_k^2 \right\rangle - P\right)\frac{\partial y}{\partial n} \tag{2.7}$$

For an unknown input statistic, we do not know explicitely y(n). Since we can assume that it is a monotoneous increasing function, we might as well replace it by a positive constant for the stochastic learning, including it in the (heuristic) learning rate $\gamma(t)$.

$$n(t+1) \;=\; n(t) \;-\; \gamma(t)\left(y_k^2 - P\right) \tag{2.8}$$

So, the system starts with all neurons having an equal input area. For a stable input statistic, by using a small γ each neuron should only slowly change the input area, much more slowly than the weights themselves.

The growing mechanism of (2.8) can also have a biological counterpart: the receptive field of the neuron might grow as long as it senses a difference between its own, small activity and the neighboured activity. This can also be interpretated as a kind of load balancing mechanism for a pool of processors.

The lateral inhibition weights should also be updated and reflect an average of the most recent patterns. Please note that the standard stochastic approximation approach yields some problems in this case because the distribution of the y_i is not stationery; they are subject for change of the weights.

It should be noted that the goal of the combined decorrelation and growing mechanism is no longer a PCA in the classical sense; the decorrelation of the output is produced by vectors which are no longer the eigenvectors of the input correlation matrix; they are something different. The exact analytical expression for the new multi-resolution goal is not easily to obtain; this is a subject for current research.

4 Simulating multi-resolution image encoding

For a small picture called „Nikita“, shown in figure 4, we implemented a multi-resolution encoding.

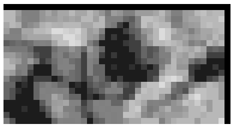

Fig. 4 *The image „Nikita“*

For this purpose we divided the 35x18 pixel image into blocks of different, but fixed size, input sequentially the blocks to one network with the input dimension *n* of the blocksize, and let the network converge by the learning rules. After convergence, we can compare the number of encoding coefficients and the measured reproduction error of an equal block size encoding, i.e. a classical KLT, with the ones of the multi-resolution approach which certainly does not result in a KLT but still decorrelates the output data. The following table compares the two encodings.

Encoding type	**No. of diff. comp.**	**comp. size**	**Total no. of comp.**	**Reprod. error**
KLT	2	4x2	162	0.1413
MR	2	4x2, 5x3	123	0.115
KLT	3	4x2	243	0.1215
MR	3	4x2, 5x3, 6x4	147	0.113

As we can see, the multi-resolution scheme MR has already advantages over the classical KLT encoding even for just two components. For three KLT components, the number of necessary encoding components is 198% compared to the MR encoding, even involving a higher reproduction error.

This simulation has shown the principal advantage of multi-resolution encoding over the classical encoding scheme of the KLT encoding. Nevertheless, for higher areas the model encounters heavy convergence problems due to the fact that the conditions (2.5c) are only valid for the actual, unknown variance (which is different for each area) and a KLT expansion, which is not given here.

For bigger pictures and more neurons our biologically-inspired approach of symmetrical networks poses too much problems. Thus, we have to devise a different algorithm for technical applications.

5 Conclusion

We presented a multi-resolution encoding scheme for image encoding and showed that the underlieing concept of non-uniform sampling yields higher compression at lower error compared with conventional methods.

Additionally, for the implementation of this concept we presented an adaptive algorithm for the generation of a noise-immune multi-resolution architecture of an neural network.

Nevertheless, for the multi-resolution encoding for technical applications other algorithms are needed which will be presented elsewhere.

Acknowledgment

We want to thank A. Schmid for providing us with the „Nikita“ simulation data.

References

[1] R. Brause: Transform Coding by Lateral Inhibited Neural Nets; Proc. IEEE Tools for Art. Intell. TAI-93

[2] R. Brause, M.Rippl: Image Encoding in the Presence of Noise; submitted to Int. Conf. on Art. Neural Networks ICANN95, Paris Oct. !995

[3] G.Wallace: The JPEG Still Picture Compression Standard; Comm. of the ACM, Vol.34, No.4, pp.31-44, April 1991

[4] S. Mallat: A Theory for multiresolution signal decomposition: The wavelet representation; IEEE Trans. Pattern Anal. and Mach. Intell., vol.31, pp.674-693, 1989

[5] D. LeGall: MPEG: A Video Compression Standard for Multimedia Applications; Comm. of the ACM, Vol.34, No.4, pp. 47-58, April 1991

[6] E.Oja: Neural Networks, Principal Components, and Subspaces; Int. Journ. Neural Syst. Vol1/1, pp. 61-68, World Scientific, London 1989

[7] J. Rubner, K. Schulten, P. Tavan: A Self-Organizing Network for Complete Feature Extraction; in R.Eckmiller, G.Hartmann, G.Hauske (eds.): Parallel Processing in Neural Systems and Computers; North Holland, Amsterdam 1990, pp. 365-368

[8] Terence D. Sanger: Optimal Unsupervised Learning in a Single-Layer Linear Feedforward Neural Network; Proc. of the Int. Conf. Neural Networks, Boston 1988, Pergamon press; und in Neural Networks, Vol. 2, pp.459-473 (1989)

[9] C.E. Shannon, W.Weaver: The Mathematical Theory of Information; University of Illinois Press, Urbana 1949

[10] C.E. Shannon: Communication in the presence of noise; Proc. of the IRE, Vol. 37, pp.10-21, Jan. 1949

[11] P.P. Vaidyanathan: Multirate Systems and Filter Banks; Prentice Hall PTR, Englewood Cliffs, New Jersey 1993