NEURAL NETWORK BASED IMAGE CODING QUALITY PREDICTION

Pascal Fleury *Olivier Egger*
Signal Processing Laboratory
Swiss Federal Institute of Technology
1015 Lausanne - Switzerland
fleury@lts.epfl.ch

ABSTRACT

Current developments in digital image coding tend to involve more and more complex algorithms, and require therefore an increasing amount of computation. To improve the overall system performance, some schemes apply a different coding algorithms to separate parts of an image according to the content of this subimage. Such schemes are referred to as *dynamic coding schemes*. Applying the best suited coding algorithm to a part of an image will lead to an improved coding quality, but implies an algorithm selection phase. Current selection methods require the computation of the reconstructed image after coding and decoding with all the selected algorithms in order to choose the best method. Some other schemes use ways of pruning the search in the algorithm space. Both approaches suffer from a heavy computational load. Furthermore, the computational complexity is increased even more if the parameters have to be adjusted for a given algorithm during the search. This paper describes a way to *predict* the coding quality of a region of the input image for any given coding method. The system will then be able to select the best suited coding algorithm for each region according to the predicted quality. This prediction scheme has low complexity, and also enables the adjustment of algorithm specific parameters during the search.

1. INTRODUCTION

Digital image compression tends to give better results when the coding scheme is adapted to the content of the image. Most coding schemes apply a single algorithm on a whole image. The so called second generation coding schemes decompose the scene into visual primitives such as regions and code each one independently [1]. Furthermore, in a *dynamic coding* scheme the algorithm used to encode each region may vary from region to region [2]. The growing number and the increasing complexity of those coding schemes [3] make it possible to substantially increase the global coding quality of a digital image coder.

The drawback of having such a panoply of algorithms is the intensive computation required to find the optimal combination for each image. Current coding schemes rely on an exhaustive search in the coding-algorithm space [4], use some intelligent pruning to guide the search [5, 6] or arbitrarily decimate the number of schemes, as in the emerging *MPEG-4* standard [7]. An exhaustive search also makes it impractical to explore a coding algorithm parameter space during the coding process. It is for example computationally intensive to select dynamically the optimal block size in a DCT based coding algorithm.

In order to overcome this drawback we propose an external system which is capable of predicting the coding quality for each region several times for a given coding algorithm. This external prediction scheme is of very low computational complexity. In this way we can avoid the heavy computational load of effectively coding and decoding each region for the selection of the best coding algorithm.

The proposed prediction scheme is based on artificial neural networks (ANN). These have proved to be capable of approximating complex nonlinear functions depending on several parameters [8]. The design of the proposed prediction scheme involves two steps. The first step consists in creating a database of small images and computing their coding quality for all the selected coding algorithms. This has to be done by encoding and decoding each image with the coding algorithm and by measuring the resulting coding quality for a panoply of different compression ratios. The second step consists in designing the neural net and train it on this database, for it to approximate the input feature – coding quality function.

This paper is organized as follows. In Section 2. we briefly describe artificial neural networks. In Section 3. the selected coding algorithms are introduced. The features given at the input of the neural net are detailed in Section 4. Their different representations is described in Section 5. The performance of the proposed prediction scheme on simulation results is shown in Section 6. Finally, conclusions are drawn in the last section.

2. ARTIFICIAL NEURAL NETWORKS

An artificial neural network is a nonlinear system. A network consists of a number of layers. Each layer in turn consists of a number of synapses. Traditionally, the first layer is called input layer, the last one output layer and the ones in between are called the hidden layers. In our system we use neural networks having only one hidden layer. This architecture is shown in Figure 1. In our scheme we restricted the possible connections to feed–forward connections. Also we use synapses using the classical *sigmoid* function. These are described by

$$y = f(\sum_{i=0}^{N-1} w_i x_i) \tag{1}$$

where y is the output of the synapse, w_i are the weights of the synapse, x_i are the inputs to the synapse and N is the number of inputs. Choose $f(\cdot)$ to be the sigmoid function defined by:

$$y = f(a) = \frac{1}{1 + exp(-a)}. \tag{2}$$

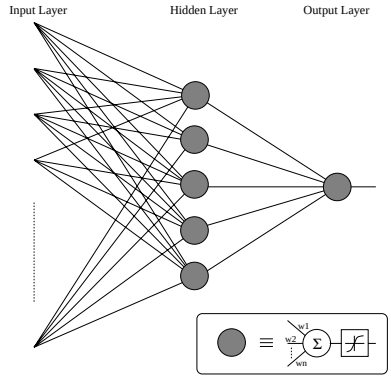

Figure 1. The topology of the used networks: feed-forward, fully-connected and sigmoid transfer function at each node.

In order to find the optimal weights w_i for each synapse of our problem, we have used the classical backpropagation algorithm [8].

3. CODING METHODS

The first coding algorithm considered for our simulations is the *DCT* mode. This block based coding algorithm performs a discrete cosine transform (DCT) on the block, followed by a linear quantization, a zigzag scanning followed by a run-length coding of the parameters and finally a Huffman coding of the resulting run-length codes. The DC coefficient is quantized separately. The scheme is very similar to *H.261* [9], except that the quantization step is modified so as to obtain the target bitrate on a macro-block basis, not on a frame basis.

The second algorithm considered is the *N-Level* mode. This block based coding algorithm performs a classification of the pixels into n clusters, according to their gray level value. The clustering is performed with a fuzzy c-means clustering algorithm [10]. The resulting pixel values are then run-length coded, and then Huffman entropy coded. The number of clusters is varied so as to reach the target bitrate. In this mode, the level vs. bitrate function is highly non-linear, therefore n is modified until the reached bitrate is lower than or equal to the target bitrate.

4. DECISION FEATURES

The prediction system approximates the real function that maps the input parameter space onto the output parameter space. The input parameter space contains data from the input image, e.g. pixel values, as well as data describing the coding environment, e.g. the bitrate of the block size. When considering different block sizes or different shapes, the number of considered pixels might vary. Thus, the number of data points at the input will not be a constant. As we want to model the predictor with an ANN it is impractical to handle varying number of input parameters. Furthermore, a deterministic preprocessing of the raw input data introduces a feature extraction step such that the prediction system becomes less complex. Therefore our prediction system is based on computed *decision features* which themselves depend on the input image content and the coding environment status.

Our system uses the following six features for each region:

size Represents the size of the considered region, in number of pixels.

Feature	Size	Variance	Type	Bitrate	Inputs
Range	16–256	0–16000	0–100	0.1–1.9	
Binary	4	8	7	8	41
SlotRange	4	8	7	8	47
Ensemble	3	3,5,7	3,5,7	3,5,7	78
Thermometer	4	8	7	8	41

Table 1. Considered range and number of bits used for each of the input features in the different representations. Values outside the range have not been used for training/testing the networks.

variance Represents the variance of the pixel value in the considered region.

pixel type The pixel of each region are classified into three types of pixels, as described in [11]. This results in three percentages of *edge*, *texture* and *uniform* pixel types.

bitrate This parameter models the environment of the coding scheme.

This set of features has produced the highest prediction accuracy among the different sets we have considered [12]. Also note that the decision feature values have to be computed only once, independently of the number of coding schemes which are considered.

5. FEATURE REPRESENTATION

The representation of the data at the input of the neural networks is a critical issue. It can be viewed as a further step of decorrelation of information [13]. Most of the different representations increase the number of inputs to the ANN, but each input then carries less information. The ANN can then better exploit the information. This tends to augment its capacity of generalization. In general, all inputs are normalized to the range $[0, 1]$ to facilitate convergence. Bounds have been fixed to the values shown in Table 1. All representations transformations listed below are thus performed on normalized values of decision features.

We have tested the following different representations:

Linear Features are presented to the network as a normalized number. There is one input per feature.

Binary The normalized feature range is divided into $2^{n_b}-1$ slots where n_b is the number of representations bits. The feature is then represented by the binary index of the slot the value ends up in.

SlotRange The input feature range is divided into n_{sr} slots. A feature value will end-up in slot s_{sr}, and only that slot will have value 1. The other slots will have value 0. An additional input, called *range* will contain the (normalized) position of the feature value in the active slot s_{sr}.

Ensemble This representation is a multi-level slot representation. For each of the levels l_e we have a certain number of slots s_{l_e}. This kind of representation has interesting generalization and specialization features [13]. The number of inputs presented to the network will be $n_e = \sum_{k=1}^{l_e} s_k$ and they will all have a value in $\{0, 1\}$.

Thermometer This representation is like a single level *Ensemble* representation. The difference is that all slots from number 0 up to the active slot will present a value 1 to the network [13]. Hence the name.

The bits used for features in each representation are shown in Table 1. They have been selected to obtain similar computation complexities.

6. RESULTS

Neural Networks

The ANN have been trained on a dataset produced from 4 natural images. The set of all points has been split into a training and a test set of 10500 input-output couples each. The ANNs are feed-forward networks with an input layer, one hidden layer and an output layer. There is one ANN per coding scheme. This configuration leads to better classification than a single network having one output node per coding scheme [14]. Furthermore, the addition of a new coding algorithm is simpler, as it only implies the addition of a new ANN with the same input features. The used training algorithm is fixed learn-rate back-propagation.

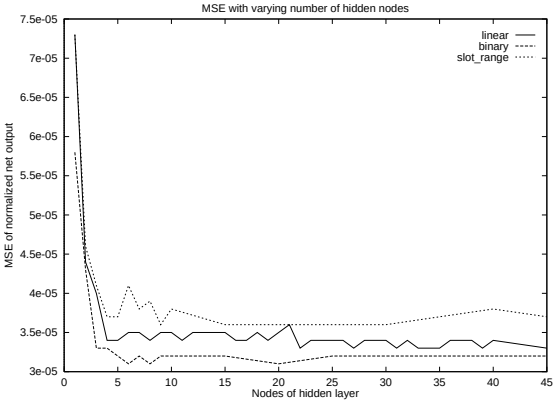

Figure 2. Variation of the Peak Signal to Noise Ration (PSNR) of the decoded image image in a fully-connected prediction neural network having 6 feature inputs, one output and a variable number of hidden nodes. The training and the test sets contain 10500 points each. The three curves show three input feature representations from the set we considered.

The number of hidden nodes in all our 3-layer networks has been set to 5. The mean square error (MSE) of the output, evaluated on the test set, with different number of hidden layer nodes, are shown in Figure 2. For more than 5 nodes improvements are no longer significant, but the computation time increases significantly in a fully-connected feed-forward network.

Quality Prediction

The prediction of the coding quality for the different representations of the input features are shown in Figure 3 for the *DCT* mode and in Figure 4 for the *N-level* mode. One can see that the networks converge quite rapidly. A number of about 10000 cycles over the dataset already leads to overlearning in the case of *Thermometer* in the *N-level* mode, and for *Binary* in the *DCT* mode.

The representations leading to the best results are shown in bold in Table 2. It is easy to determine the number of operations needed to compute the prediction evaluation. These are shown in Table 2. This does not take into account the computation of the features, which has to be done once, for all coding schemes. The best results are obtained

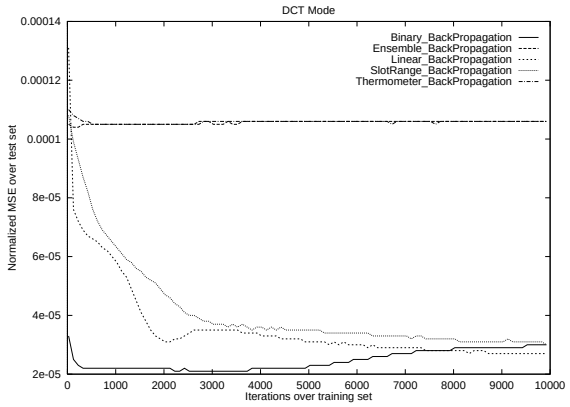

Figure 3. Convergence of the ANNs modeling the DCT coding algorithm, for different types of feature representation.

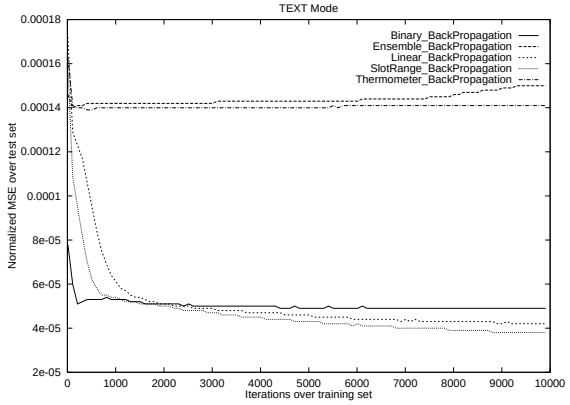

Figure 4. Convergence of the ANNs modeling the *N-level* coding algorithm, for the set of input representations

with the *Binary* representation for the *DCT* mode, and the *SlotRange* for the *N-level* mode.

Because of their quantizing property, the *Binary*, *Thermometer* and *Ensemble* representations are lossy in nature. Values which are quantized to the same slot all have the same value for the ANN. This might have several consequences,as shown below.

In the input feature space, the PSNR curve is represented by a grid of points when the feature values are quantized by the representation. Therefore, the ANN will not be able to predict a PSNR of a point which is not on that grid. For a point in the feature space we will get the predicted PSNR of the closest grid-point. If the PSNR surface is very rough, this approximation might lead to a bad prediction.

The *DCT* mode has an adjustable quantization step with 256 levels. It is possible to adapt the coding scheme to slight changes in the features. The *N-level* mode has only the number of levels (N) as adaptation parameter. A slight change in the feature space might make the coding scheme drop one level, and thus change the PSNR drastically.

As the grid for input feature quantization is solely defined by the number of slots (i.e. bits) for each feature, the prediction of the *N-level* mode is often badly predicted and leads

Mode	LIN	BIN	SLR	ENS	TRM
DCT	0.572	**0.504**	0.612	1.133	1.133
N-level	0.746	0.770	**0.678**	1.315	1.311
Complexity	35/0	5/210	10/215	5/395	5/210

Table 2. Numerical values for predicted PSNR accuracy (in dB), along with the number of multiplication/additions needed to perform the prediction task.

Algorithm	PSNR Range	Representation	Accuracy
DCT	0-110 dB	*Binary*	0.504 dB
N-level	0-110 dB	*SlotRange*	0.678 dB

Table 3. Rendered accuracy of the best systems for each mode.

therefore to slightly worse results than the *DCT* mode, as shown in Table 2.

This explains also the fact that the *Binary* representations leads to much better results than the two other quantizing representations. With n bits, the *Binary* representation is able to distinguish 2^n slots, whereas the two other representations distinguish at most n slots. The quantization of the input feature space is a rougher grid, and the prediction quality decreases. The best results in the non-normalized space are summarized in Table 3.

Bit Allocation

Using ANN to predict the coding quality saves computation time. Therefore, part of the savings can be invested in distributing the bits among the regions of the image in such a way that the resulting coding quality is increased. One possible algorithm is described below.

One starts with an initial bitrate budget for regions proportional to their size (in pixels). Then, for each region, one computes the *predicted* quality of the coded region. One can then take the region corresponding to the lowest quality, increase its bitrate budget and decrease the bitrate budget of regions which result in a quality higher than the average for all regions. This is iterated until the difference in quality among all the regions is below a given threshold. Although this heuristic approach does not guarantee an optimal rate–distortion bit allocation, it nevertheless allows a uniform distribution of the error. At the end of the iterations, we have for each region the allocated number of bits and the corresponding coding technique.

7. CONCLUSION

We have presented a new system for the selection of algorithms for each region in a dynamic coding scheme. Since the exhaustive search for the selection of the best algorithm for each region is computationally intensive, we propose a system based on a prediction of the coding quality for a number of algorithms.

Our prediction system is built around Artificial Neural Networks (ANN). As input, a set of 6 features is computed. The features represent the input image as well as the coding environment for each region. The prediction has been tested on two coding schemes: a DCT transform based scheme and a *N-level* pixel clustering scheme. To further enhance the quality of the prediction, several feature representations have been tested. The best results were given by the *Binary* representation for the *DCT* mode, and *SlotRange* for the *N-level* mode.

The proposed method reduces the computation time for the selection of the best coding scheme among a set of predefined ones. Part of this can then be used to determine the optimal distribution of the bits over different regions in the image. For this purpose, a bit allocation algorithm has been proposed.

REFERENCES

[1] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image-coding techniques", *Proc. of the IEEE*, vol. 73, pp. 549–574, April 1985.

[2] E. Reusens, O. Egger, and T. Ebrahimi, "Very low bitrate coding: which way ahead ?", *in IEEE Workshop on nonlinear signal and image processing*, pp. 1019–1022, Halkidiki, Greece, June 1995.

[3] R. J. Clarke, *Digital Compression of Still Images and Video*, Academic Press Ltd., 1995.

[4] E. Reusens, "Joint optimization of representation model and frame segmentation for generic video compression", *Signal Processing*, vol. 46, pp. 105–117, September 1995.

[5] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with application to mpeg video coders", *in Proc. of ICASSP*, vol. V, pp. 381–384, 1993.

[6] G. Poggi and R. A. Olshen, "Pruned tree-structured vector quantization of medical images with segmentation and improved prediction", *IEEE Trans. on Image Processing*, vol. 4, June 1995.

[7] T. Ebrahimi et al., "Dynamic coding of visual information, technical description ISO/IEC JTC1/SC2/WG11/M0320", MPEG-4, Swiss Federal Institute of Technology, October 1995.

[8] B. Widrow and Michael A. Lehr, "30 years of adaptive neural networks: Perceptron, madaline and backpropagation", *Proc. of the IEEE*, vol. 78, pp. 1415–1441, September 1990.

[9] CCITT SG XV, "Recommendation H.261 – Video Codec for Audiovisual Services at p×64 kbit/s", Technical Report COM XV–R37–E, International Telecommunication Union, August 1990.

[10] R. L. Cannon, J. V. Dave, and J. C. Bezdek, "Efficient implementation of the fuzzy c-means clustering algorithms", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, NO. 2, pp. 248–255, March 1986.

[11] O. Egger, W. Li, and M. Kunt, "High compression image coding using adaptive morphological subband decomposition", *Proc. of the IEEE*, vol. 83, pp. 272–287, February 1995.

[12] P. Fleury, J. Reichel, and T. Ebrahimi, "Image quality prediction for bitrate allocation", *in IEEE Proc. of ICIP*, vol. 3, pp. pp. 339–342, 1996.

[13] M. Smith, *Neural Networks for Statistical Modeling*, Van Nostrand Reinhold, 1993.

[14] A. Rangachari, M. Kishan, K. M. Chilukuri, and R. Sanjay, "Efficient classification for multiclass problems using modular neural networks", *IEEE Trans. on Neural Networks*, vol. 6, pp. 117–124, January 1995.