# An Adaptive Quantization Algorithm for MPEG-2 Video Coding

Lijun Luo, Cairong Zou, Zhenya He
Department of Radio Engineering
Southeast University, Nanjing 210096, P. R. China

Isao SHIRAKAWA
Dept. of Information Systems Engineering
Osaka University, Osaka, Japan

## ABSTRACT

An adaptive quantization algorithm for MPEG-2 video coding using neural network is presented in this paper. The proposed algorithm uses BP neural network to divide the macroblock activity into one of four categories: flat, edge, texture, fine-texture, and thus the macroblock can be quantized adaptively according to the human vision system (HVS) sensitivity. Experiment results show that this method can reduce blocky artifacts of flat area and distortion at edge effectively. Meanwhile, the picture subjective quality and objective quality of each frame are improved.

## I. INTRODUCTION

The coding of video sequences has been the thrust of a great deal of research in recent years. The emergence of MPEG-2 [1], H.261 [2], and the proposed high definition television (HDTV) standards all have used adaptive quantization as a cornerstone of objective quality. Many researchers have assumed that if the minimum mean squared error (MSE) distortion of quantization could be matched to the just-noticeable distortion (JND) of the human vision, no loss of perceived quality is achieved [3]. However our human visual system (HVS) does not perceive quality in the MSE sense and thus the adaptive quantization algorithm of Test Model 5 (TM5) [4] for MPEG-2 is not sufficient, which can cause blocky artifacts in flat area and distortion at edges.

With the advances in visual psychophysics, understanding of human vision has progressed significantly. Various mathematical homomorphic models [5] of the HVS have been contrived and proven successful in image processing. The design of subjectively optimized quantizer can achieve high compression by adaptively analyzing the whole image in the spatial domain with HVS sensitivity function in a global sense. A perceptually classified transform encoder using the RM8 based H.261 encoder [6] can classify the local image content into one of four perceptual classes: flat, edge, texture, or fine-texture by using texture masking energy that is weighted by a HVS function in a local sense, and adapts the quantizer dynamically.

In this paper, an adaptive quantization algorithm for MPEG-2 video coding using neural network is given. The proposed perceptual code for MPEG-2 coding uses BP NN to divide MB activity into one of four categories: flat, edge, texture, fine-texture, and adapts the quantizer dynamically. In section II a sub-block texture masking energy function that is weighted by a HVS function is given. Since the weighting and threshold values of NN need be studied by using training samples, in section III the learning algorithm of BP NN is presented in detail. In section IV the simulation results of the quality improvement achieved by the proposed method is presented. Finally the features of the proposed algorithm and the conclusion are summarized in section V.

## II. SUB-BLOCK TEXTURE MASKING ENERGY FUNCTION

In the MPEG-2 syntax, images are divided into macroblocks (MB) of four adjacent $8 \times 8$ luminance (Y) blocks and two chrominance (U and V) blocks with the same size. MB can be quantized either by intra-mode or by inter-mode. The quantization factor of MB just means multiplication of adaptively visual quantization factor (AVQF) and state factor of buffer. Our aim is to adapt AVQF

to local image content and to exploit spatial masking effect. Several approaches exist, e. g. , based on MB variance in TM5 to derive AVQF, but unfortunately we find that the TM5 algorithm can cause blocky artifacts in flat area and distortion at edges because of coarse quantization.

The proposed adaptive quantization algorithm finely control AVQF according to the local image content, thus it can be used to finely quantize MB of flat or coarsely quantize MB of texture for the consistent scene quality. From Nill [6] and Ngan et al. [8], a HVS sensitivity function, H(f) is given by

$$H(f)=(0.31+0.69f)\exp(-0.29f)\{\frac{1}{4}+\frac{1}{\pi}\arctan[\ln(\frac{2\pi f}{d}+\sqrt{\frac{4\pi^2 f^2}{d^2}+1})]^2\}^{0.5} \quad (1)$$

where

$$f=\frac{\sqrt{i^2+j^2}}{2N}f_s, i,j=0,1,\ldots,N-1 \quad (2)$$

In (2). d=11.636, and $f_s$ is the sampling density dependent upon the viewing distance. For a CIF image with a height of 288 pixels and viewed at a distance of four times the image height, $f_s=20$;N is the DCT sub-block size and is chosen to be four.

As we known, texture can be defined as any deviation from a flat field within a region of interest, which in our case, is the 4 × 4 luminance sub-block. From Tan et al. [7], the correspondent texture masking energy function can also be expressed as a function of an ac coefficient in spatial domain in a region of interest, in here, the k-th ( k=0,1,....,15) sub-block in a macroblock weighted by the HVS function shown in (1). Therefore the sub-block texture masking energy function of k-th sub-block if defined as

$$e_k=\frac{1}{\beta}\left[\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}H(i,j)^2X_k(i,j)^2\right]^{0.5}_{(i,j)\neq(0,0)},k=0,1,\ldots,15 \quad (3)$$

In (3), $\beta$ is set to five for intra-mode sub-block and four for inter-mode sub-block.

## III. THE BP NN ADAPTIVE QUANTIZATION ALGORITHM

The adaptive quantization algorithm of MPEG-2 video coding using BP NN can be divided into three parts of studying of the weighting and threshold values of NN, classification of MB activity using NN, and adaptive quantization.

For the learning part of the weighting and threshold values of NN, we use two-layer BP NN to evaluate MB activity. The input vector of NN is $\mathbf{X}=(x_0,x_1,\ldots,x_{n-1})^t=(e_0,e_1,\ldots,e_{n-1})^t$, where n is 16. The hidden layer of NN has p neutrons, $\mathbf{t}=(t_0,t_1,\ldots,t_{p-1})^t$, where p is 4. The output layer has m neutrons, $\mathbf{y}=(y_0,y_1,\ldots,y_{m-1})^t$, and m is 4, which correspond to one of four activities of MB: flat, edge, texture, fine-texture in turn. The weighting and threshold of hidden layer are $w'_{ij}$ and $\theta'_j$ respectively. The weighting vector and threshold of output layer are $w''_{jk}$ and $\theta''_k$ respectively. Generally speaking, two-layer network should be adequate for most applications. For convenience, we set

$$x_n=-1;t_p=-1;\theta'_j=w'_{nj};\theta''_k=w''_{pk} \quad (4)$$

Thus the neutrons output of each layer are as

$$\begin{cases} y_k=f(\sum_{j=0}^{p}w''_{jk}t_j) \\ t_j=f(\sum_{i=0}^{n}w'_{ij}x_i) \\ f(u)=\dfrac{1}{1+e^{-u}} \end{cases} \quad (5)$$

As we known, the weighting and threshold values of NN need to be trained. In here, we use L different kinds of learning samples, $\mathbf{x}^1,\mathbf{x}^2,\ldots,\mathbf{x}^l$, and the corresponding teacher signals are $\mathbf{d}^1,\mathbf{d}^2,\ldots,\mathbf{d}^l$, where L is 512. The selection of training samples comes from the statistical analysis of different images.

The objective of BP network is to train the weights of hidden layer and the output layer so as to minimize the least-square-error criterion E:

$$\text{minimize } E=\frac{1}{2}\sum_{l=1}^{L}\sum_{k=0}^{m-1}(d_k^l-y_k^l)^2 \quad (6)$$

Using the block-adaptive method, the basic gradient-type learning formulas of each layer of NN are given as (7) [10]

The learning process of BP NN is as follows:

(1) Select the weighting and threshold values of each layer in random;

(2) Input all the samples to NN one by one, and calculate the neutron output of each layer using equation (5) for each sample;

(3) Modify the weightings and thresholds of each layer using equation (7);

(4) Calculate the neutron output of each layer using equation (5) and the total error of all samples. If the total error is bigger than a given constant, go to step 2, else finish the training process.

$$\begin{cases} w''_{jk}(n+1) = w''_{jk}(n) + \eta \sum_{l=1}^{I} \delta'_{jk} y'_k \\ w'_{ij}(n+1) = w'_{ij}(n) + \eta \sum_{l=1}^{I} \lambda'_{ij} t'_j \\ \delta'_{jk} = (d'_k - y'_k) y'_k (1 - y'_k) \\ \lambda'_{ij} = \sum_{k=0}^{m} \delta'_{jk} w''_{jk} t'_j (1 - t'_j) \\ 0 \le \eta < 1 \end{cases} \tag{7}$$

In (7) n is the iteration time.

The selection of training samples comes from the statistical analysis of different images. Since the sub-block texture masking energy function $e_i, i = 0,1,\ldots,15$ in an MB is just among 0 and 56, so we use different kinds of training samples in this range.

As we known, an excessively large hidden-layer in the network may cause deterioration of the generalization performance. Moreover, it is very likely to cause serious numerical problems in terms of convergence and local minimum. But it is difficult to estimate the number of hidden units so that a satisfactory model can be obtained. Here we use network growing techniques to select the number of hidden-layer. The network growing techniques start with a small number of hidden units, and add new hidden units, one by one, to gradually refine the network. The results of selection is to use 4 hidden units.

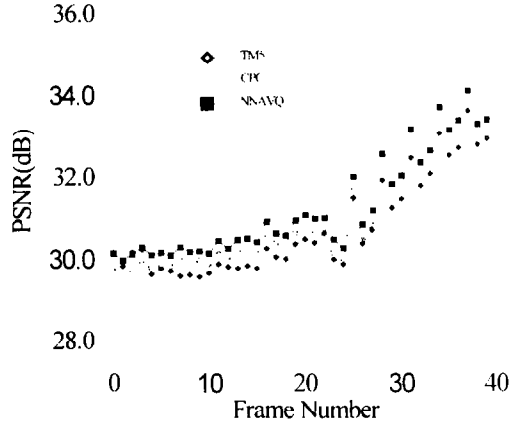
The output layer has 4 neutrons, which correspond to one of four activities of MB: flat, edge, texture, fine-texture in turn. In here, we get MB activity from the neutron of output layer that has the maximum value, and the corresponding adaptively visual quantization factors of the four MB activities are usually selected by verifying a lot of standard test sequences

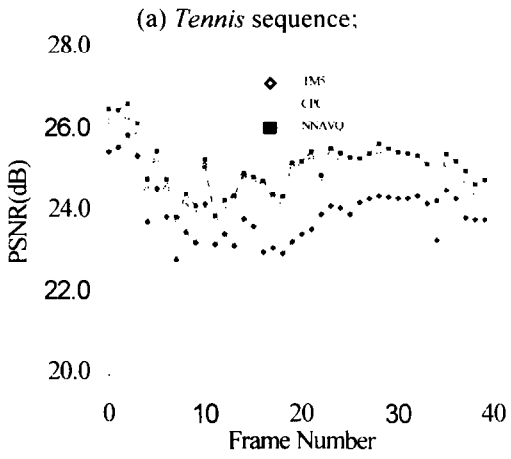
$$a_1 = 0.6, a_2 = 1.0, a_3 = 1.4, a_4 = 1.0 \tag{8}$$

## IV. SIMULATION RESULTS

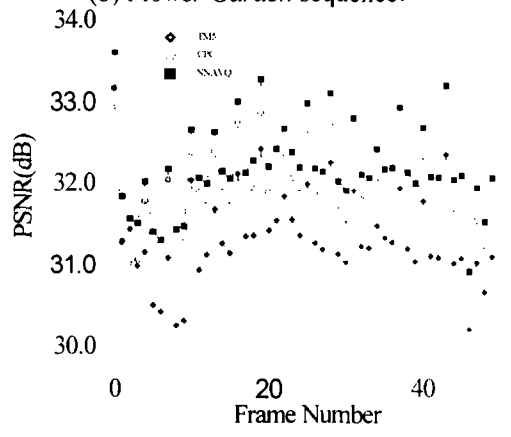
Using the MPEG-2 syntax, comparisons of the TM5 algorithm, the classified perceptual coding algorithm (CPC) [7], and the NN adaptively visual quantization algorithm (NNAVQ) are conducted using three CIF size test sequences of *Table Tennis, Flower Garden, Football*, coded at 1.152Mbps, and one CCIR601 size test sequence of *Calendar and Mobile*, coded at 5Mbps. The output peak-signal-to-noise (PSNR) curves of three algorithms are shown in Fig. 1 respectively, and the corresponding average PSNR are given in Table 1.

Referring to Fig. 1 and Table 1, it is observed that the NN adaptively visual quantization algorithm can improve the picture subjective quality and objective quality efficiently. Simulation indicates that the perceptually coded images have far fewer blocky artifacts and distortion at edges than those of TM5 and CPC algorithm. These areas, such as the sky in *Flower Garden* sequence, the walls in *Tennis* sequence, are classified as flat and are coded finely, thus constant quality is achieved across the image. Examining the four test sequences, no sticky noise, which is found in newly uncovered background, is seen in the images using NNAVQ, and the blurring of moving objects, such as the ball in *Tennis* sequence, the heads and legs in *Football* sequence, are improved greatly.

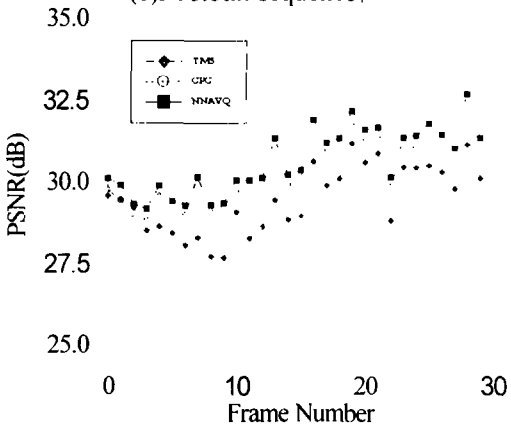





(a) *Tennis* sequence;




(b) *Flower Garden* sequence:




(c)*Football* sequence;

(d) *Calendar and Mobile* sequence.

Fig. 1 The output peak-signal-to-noise (PSNR) curves of three algorithms

## V. CONCLUSION

An adaptive quantization strategy of MPEG-2 video coding using NN is presented in this paper. Using BP neural network, the proposed algorithm can divide the macroblock activity into one of four categories: flat, edge, texture, fine-texture and adapt the quantizer according to the HVS sensitivity. Experiment results show that this method can reduce blocky artifacts of flat area and distortion at edge effectively. The picture subjective quality and objective quality of each frame are improved.

Table 1 The output average PSNR of three algorithms

| Image sequences | TM5 (dB) | CPC (dB) | NNAVQ (dB) |
|---|---|---|---|
| *Football* | 31.37 | 31.92 | 32.14 |
| *Tennis* | 30.85 | 31.03 | 31.25 |
| *Flower Garden* | 24.01 | 24.95 | 25.05 |
| *Calendar and Mobile* | 29.56 | 30.46 | 30.66 |

## REFERENCES

[1] ISO/IEC 13818-2 Coding of moving pictures and associated audio. 1995.

[2] CCITT Rec. H.261, "Video codec for audiovisual services at px64 Kbps," CDM XV-R37-E, 1990.

[3] Frei and B. Baxter, "Rate-distortion coding simulation for color images," *IEEE Trans. Commun.*, vol. COM-32, pp. 1385-1392, Nov. 1977.

[4] ISO/IEC/JTC1/SC29/WG11, "Test Model 5", Draft, Apr. 1993.

[5] Lee and B. W. Dickinson, "Temporally adaptive motion interpolation exploiting temporal masking in visual perception," *IEEE Trans. Image Processing*, vol. 3, No. 5, pp. 513-526, Sep. 1994.

[6] H. Tan, K. K. Pang, and K. N. Ngan, "Classified perceptual coding with adaptive quantization," *IEEE Trans. Circuits Sys. Video Technol.*, vol. 6, No. 4, pp.375-388, 1996.

[7] B. Nill, " A visual weighted transform for image compression and quality assessment," *IEEE Trans. Commun.*, vol. COM-33, pp.551-557, 1985.

[8] N. Ngan, K. S. Leong, and H. Singh," Adaptive cosine transform coding of images in perceptual domain," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1743-1750, Nov. 1989.

[9] Hechi-Nielsen, "Theory of the back propagation neural network," *Proc. of IJCNN*, vol. 1, pp. 593-603, 1989.