Available online at www.sciencedirect.com
ScienceDirect
Signal Processing 87 (2007) 1493–1503
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

No-reference JPEG-image quality assessment using GAP-RBF

R. Venkatesh Babu[a,*], S. Suresh[b], Andrew Perkis[c]

[a]*Department of Electrical Engineering, Indian Institute of Science, Bangalore, India*
[b]*School of Electrical and Electronics Engineering, NTU, Singapore*
[c]*Q2S, NTNU, Trondheim, Norway*

Received 8 February 2006; received in revised form 18 December 2006; accepted 26 December 2006
Available online 13 January 2007

Abstract

In this paper, we present a novel no-reference (NR) method to assess the quality of JPEG-coded images using a sequential learning algorithm for growing and pruning radial basis function (GAP-RBF) network. The features for predicting the perceived image quality are extracted by considering key human visual sensitivity factors such as edge amplitude, edge length, background activity and background luminance. Image quality estimation involves computation of functional relationship between HVS features and subjective test scores. Here, the functional relationship is approximated using GAP-RBF network. The advantage of using sequential learning algorithm is its capability to learn new samples without affecting the past learning. Further, the sequential learning algorithm requires minimal memory and computational effort. Experimental results prove that the prediction of the trained GAP-RBF network does emulate the mean opinion score (MOS). The subjective test results of the proposed metric are compared with JPEG no-reference image quality index as well as full-reference structural similarity image quality index and it is observed to outperform both.

Keywords: Image quality; No-reference metric; Blockiness measurement; GAP-RBF; Neural network; JPEG

1. Introduction

The main objective of image/video quality assessment metrics is to provide an automatic and efficient system to evaluate visual quality. It is imperative that these measures exhibit good correlation with perception by the human visual system (HVS). The most widely used objective image quality metrics, namely mean square error (MSE) and peak signal to noise ratio (PSNR), as widely observed do not correlate well with human perception [1] besides requiring the original reference image to compute distortion. There have been attempts to estimate PSNR without reference image [2]. Most images on the Internet and in multimedia databases are only available in compressed form, and hence inaccessibility of the original reference image, makes it difficult to measure the image quality. Therefore, there is an unquestionable need to develop metrics that closely correlate with human perception without needing the reference image.

Considerable volume of research has gone into developing objective image/video quality metrics that incorporate perceived quality measurement with due consideration for HVS characteristics.

*Corresponding author.
E-mail addresses: venkatesh.babu@gmail.com (R. Venkatesh Babu), suresh99@gmail.com (S. Suresh), andrew@q2s.ntnu.no (A. Perkis).

0165-1684/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2006.12.014

However, most of the proposed metrics based on HVS characteristics require the original image as Refs. [3,4]. Though it is easy to assess the image quality without any reference by manual observations, developing a no-reference (NR) quality metric is a difficult task. To develop NR metrics, it is essential to have *a priori* knowledge about the nature of artifacts. Currently, NR quality metrics are the subject of considerable attention by the research community, visibly so, with the emergence of video quality experts group (VQEG) [5], which is in the process of standardizing NR and reduced-reference (RR) video quality assessment methods.

In recent years, neural networks have emerged as powerful mathematical tools for solving problems as diverse as pattern classification/recognition [6], medical imaging [7], speech recognition [8], etc. The increasing popularity of neural networks is due to their ability to construct good approximation of functional relationship between the known set of input and output data. In neural networks, the choice of learning algorithm, number of hidden neurons and weight initialization are important factors in the learning performance. In particular, the choice of learning algorithm determines the rate of convergence, computational cost, and the optimality of the solution. The choice of number of hidden neurons determines the learning and generalization ability of the network. Another important problem is that of the re-training process involved in the architectures, whenever new set of observations (images) arrives. Sometimes, the new set of observations may change complexity of the input–output relationship (complexity of the model) and in-turn affect the approximation ability of the neural network model. The process of developing the new neural model with the current training set may leads to increase in computational time. Sequential learning algorithms, which do not require retraining whenever new observation is received, helps to overcome the aforementioned problems faced by neural network.

In this work, problem of image quality estimation without requiring the reference image is reduced to a function approximation problem using growing and pruning radial basis function (GAP-RBF) networks. The unknown functional relationship between the HVS features and mean opinion score (MOS) is captured by the leaning phase of GAP-RBF network. The input features extracted in our work use the following critical HVS characteristics such as edge amplitude, edge length, background activity, background luminance. The experiment is carried out using two disjoint sets of original images with its compressed versions from JPEG LIVE image quality database [9]. The performance of the proposed GAP-image quality metric is compared with the NR [1] and FR [10] on same database.

The paper is organized as follows: Section 2 presents the reported work on blockiness metric. Section 3 presents the concepts underlying feature extraction based on various HVS criteria. The basics of GAP-RBFN image quality model are dealt with, in Section 4. Subjective test results and discussions are presented in Section 5. Finally Section 6 concludes the paper.

2. Related work

Existing algorithms to measure blockiness have used a variety of methods to do so. Wang and Bovik proposed an algorithm based on computing the FFT along the rows and columns to estimate the strength of block-edges [11]. Further, they proposed a nonlinear-model for NR quality assessment of JPEG images, where the parameters of the model were determined with subjective test data [1]. The method proposed by Lin et al. [12] uses gradient across block boundary for measuring local blockiness. However, the above method does not perform better than Wang's NR metric on LIVE database [12]. Vlachos used cross-correlation of subsampled images to compute a blockiness metric [13]. Wu and Yuen proposed a metric based on computing gradients along block boundaries while tempering the result with a weighting function based on HVS [14]. Here, the block edge strength for each frame was computed. Similar ideas about the HVS were utilized by Suthaharan [4], Gao et al. [15] and Wu et al. [14]. The general idea behind these metrics was to temper the block-edge gradient with the masking activity measured around it. These approaches utilize the fact that the gradient at a block-edge can be masked by more spatially active areas around it, or in regions of extremities in illumination (very dark or bright regions) [16]. The metric proposed by Gao et al. [15] is meant for evaluating the post-processing/de-blocking algorithms. The provided results show very high correlation (>0.97) between the metric and the quantization parameter. Hence, using this metric for quality evaluation is equivalent to estimating the quantization parameter which is independent of subjective quality scores. Jung et al. [17] proposed a NR metric

for emulating the full-reference metric proposed by Karunasekera and Kingsbury [3] using neural network. Here they have used general image features for training the neural network and the results are not compared against subjective test scores. On the other hand, recently, Gastaldo et al. [18,19] proposed a circular back propagation (CBP)-based image quality evaluation method using general pixel-based image features such as higher order moments without considering HVS. In all the abovementioned approaches, extracting large number of general image features is computationally quite complex for real-time implementation. Yang et al., recently proposed a video quality metric considering HVS and temporal dependency between images [20]. This metric uses the previously reconstructed frame as reference for measuring quality of the current frame. This method cannot be used for measuring quality of a stand-alone image. Recently, there has been a proposal to measure NR image quality of JPEG 2000 images (whose compression artifacts are different from JPEG) using natural scene statistics [21]. The following papers provide useful evaluations and survey of existing image quality metrics [22–24].

3. HVS-based feature extraction

The proposed NR metric is designed to take into consideration the various human visual criteria while quantifying the blocking artifact. These blocking artifacts would appear as horizontal and vertical edge distortions at the boundaries of 8×8 blocks. The visual sensitivity to these edges is affected by the following parameters [3]:

- *Edge amplitude*: Edge amplitude quantifies the strength of edge along the borders of 8×8 blocks. It is meant to distinguish between true image edges and those that apparently arise due to compression. Since JPEG uses DCT-based transform coding for every 8×8 image block, the effect of quantization appears as the blocking artifacts at the edges of each 8×8 block. Edge amplitude at the 8×8 block boundaries is proportional to the amount of compression the image is subjected to. In other words, the edge-amplitude increases with decreasing bit-rates.
- *Edge length*: Edge length quantifies the length of continuous block edges. Similar to edge amplitude, edge length, in general, also increases with compression.
- *Background activity*: Background activity measures the amount of high frequency texture content around the block edges. The blocking artifacts will be masked by the activity in the background. For instance, an edge that happens to lie in a textured region is less visible to an observer compared to the edge against a plain background.
- *Background luminance*: This attribute measures the amount of brightness around the block edges. The visibility of the edge is often affected by the surrounding regions. For example, an edge lying in a darker region is less visible compared to the edge in the brighter region.

The objective of the proposed metric is to integrate the aforementioned human visual factors to measure the quality of the JPEG-compressed images. The algorithmic steps for computing the feature vector are shown in Fig. 1. The edges along horizontal and vertical directions are obtained using the corresponding 'prewitt' edge operators. Activity along, as well as, on either sides of the horizontal and vertical edges, is captured by high-pass filtering. The final binary activity mask is obtained by hard thresholding the activity measure using the methodology outlined in next subsection. This mask only permits regions with lower activity to be considered for blockiness measurements. The background luminance weights are obtained based on the model proposed by Karunasekera et al. [3,25]. Here the darker regions (0–127) are given less weight and the brighter regions (128–255) are given higher weights. Each pixel of the edges that belongs to the activity mask is multiplied by the corresponding luminance weight, in order to obtain the final horizontal and vertical edge maps. The horizontal and vertical edge profiles are computed from these weighted edge maps. These profiles indicate the edge strength along each row and column of the weighted edge map. Since the effect of blockiness is available only at the block boundaries, every eighth location of the horizontal and vertical profiles is considered for measuring blockiness. The measure of deviation at every eighth location from the average value of the neighborhood of both (horizontal and vertical) profiles is used for extracting the features. The implementation details of feature extraction phase are described in the following subsection.

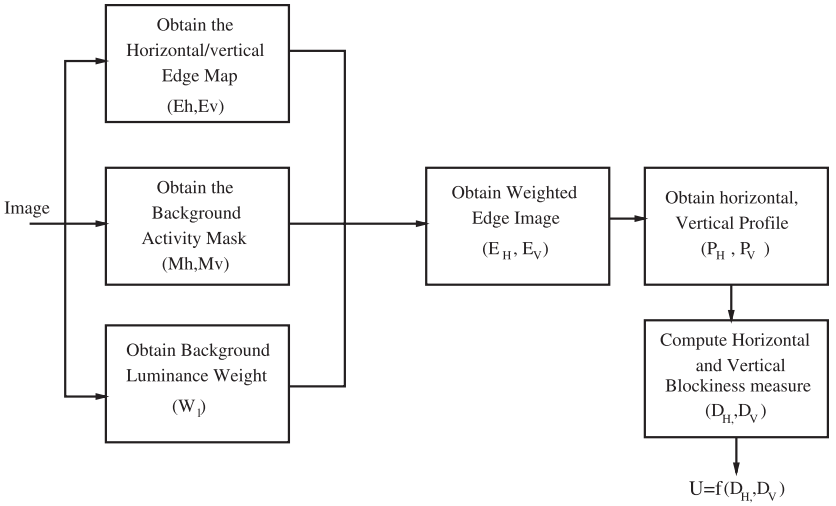

Fig. 1. Overview of feature extraction phase.

3.1. Implementation details

Consider a gray scale image of size $M \times N$ (*rows* $\times$ *columns*). The intensity of the image at any pixel location (i,j) is given by $I(i,j)$ which lies in the range of 0–255.

The algorithm is explained for computing the difference profile along horizontal direction. A similar approach is used for the vertical direction as well.

(1) Obtain the horizontal edge map using horizontal prewitt filter:

$$\hat{E}_{\mathrm{h}} = |I * P_{\mathrm{h}}|, \tag{1}$$

$$E_{\mathrm{h}}(i,j) = \begin{cases} \hat{E}_{\mathrm{h}}(i,j) & \text{if } \hat{E}_{\mathrm{h}}(i,j) < \tau_{\mathrm{e}}, \\ 0 & \text{otherwise,} \end{cases}$$

where

$$P_{\mathrm{h}} = \tfrac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

is the prewitt horizontal filter and '$*$' indicates convolution operation. Since the edges caused by blocking artifact have relatively lesser magnitude compared to a true image edge, a predesigned threshold is used to capture the edges due to blockiness and to avoid the true (strong) image edges. It has been observed experimentally that the maximum edge magnitude due to blockiness is below 40 for wide range of compressions. Choosing a lower threshold might result in missing the real blocky edges resulting from compression, so the typical value of the edge threshold (τ_{e}) used in all our experiments is set at 35.

(2) Measure the background activity along horizontal direction (A_{h}).

$$A_{\mathrm{h}} = |I * F_{\mathrm{ah}}|, \tag{2}$$

where

$$F_{\mathrm{ah}} = \tfrac{1}{8}\begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}.$$

The filter F_{ah} captures the background activity along the horizontal edges. The activity values of the entire image are normalized to the range [0–1].

(3) Mask the edges in the active background region by a pre-defined threshold τ_{a}. The mask M_{h} is given by

$$M_{\mathrm{h}}(i,j) = \begin{cases} 1 & \text{if } A_{\mathrm{h}}(i,j) < \tau_{\mathrm{a}}, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The typical value of the activity threshold (τ_{a}) used in our experiments is 0.15. It has been observed experimentally that the effect of blockiness is masked by the activity when it is more than this threshold (τ_{a}).

(4) The background luminance (W_l) of every pixel is obtained using a low-pass filter and the weight of each pixel is assigned as follows:

$$W_l(i,j) = \begin{cases} \sqrt{\dfrac{I_l(i,j)}{128}} & \text{if } 0 \leqslant I(i,j) \leqslant 128, \\ 1 & \text{otherwise,} \end{cases}$$

where

$$I_l = I * f_{lp}, \tag{4}$$

where

$$f_{lp} = \frac{1}{4}\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

is the low-pass filter used in order to avoid horizontal and vertical artifacts.

(5) The final weighted edge image (E_H) for the horizontal direction is obtained as:

$$E_H(i,j) = E_h(i,j) \times M_h(i,j) \times W_l(i,j). \tag{5}$$

(6) Obtain the vertical profile of E_H (projection of the rows of E_H on a vertical axis).

$$P_v(i) = \sum_{j=1}^{N} E_H(i,j). \tag{6}$$

(7) Obtain the difference profile along horizontal direction (D_H)

$$D_H = \frac{1}{M}\left(P'_v - \bar{P}'_v\right), \tag{7}$$

where

$$P'_v = P_v(8n) \quad \text{and} \quad \bar{P}'_v(n) = \bar{P}_v(8n), \tag{8}$$

where

$$\bar{P}_v = \text{median}(P_v).$$

To illustrate this process clearly, the vertical profile P_v and the median filtered vertical profile, using a five tap median filter, P'_v of an image are shown in Figs. 2(a) and (b). It can be observed from Fig. 2(a), that every eighth location shows a sharp rise from its neighboring locations due to the block-based compression. This rise in value at every eighth location is directly proportional to the amount of compression. In order to capture this phenomena, every eighth location of difference profile is subtracted

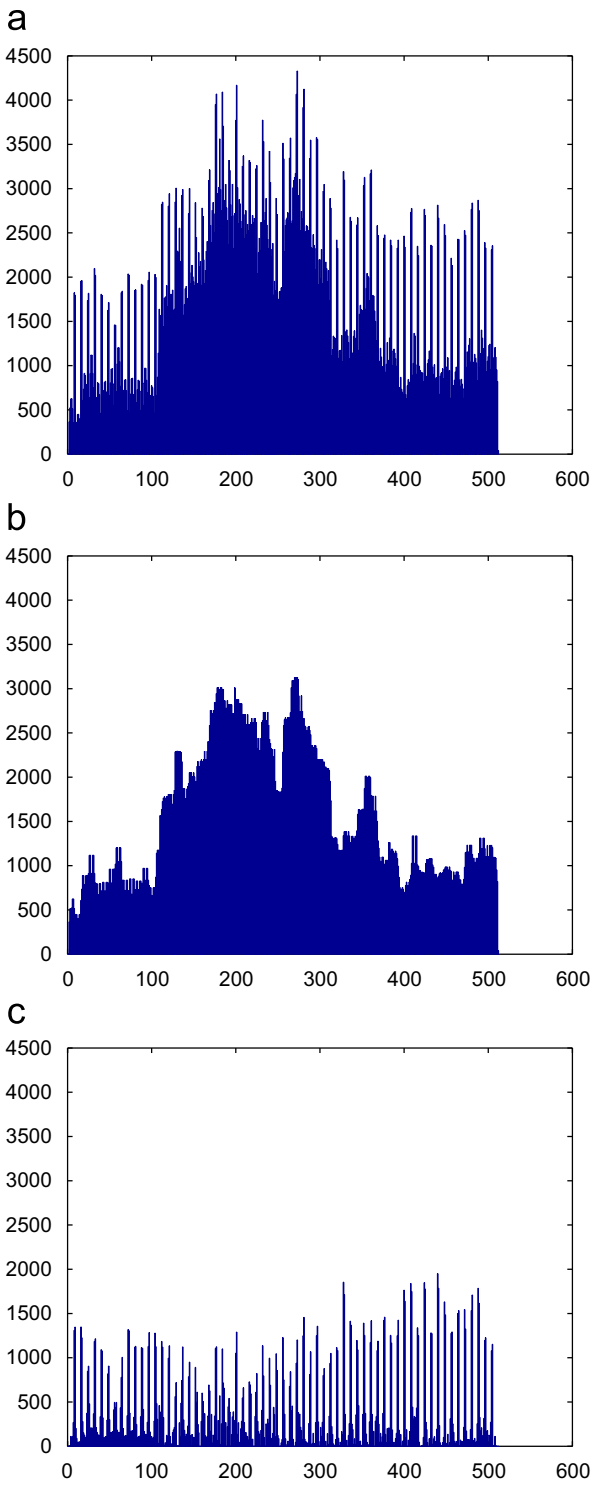

Fig. 2. Illustration of impact of median filtering: (a) the vertical profile P_v of an image, (b) the median filtered vertical profile $\bar{P}_v$, (c) the difference between P'_v and $\bar{P}'_v$.

by its corresponding median filtered value. Now every eighth location of P_v and $\bar{P}_v$ is represented by P'_v and $\bar{P}'_v$. Fig. 2(c) shows the difference between P'_v and $\bar{P}'_v$.

Similar steps are used to obtain the difference profile along vertical direction (D_V). The final difference profile (D_F) is obtained by combining the horizontal (D_H) and vertical (D_V) difference profiles (i.e., $D_F = \{D_H, D_V\}$). The final feature vector (U) is obtained by quantizing the combined difference profile D_F into 14 bins (14 features). The following bin-intervals are used to obtain the final feature vector: $[\text{Min}, -1)$, $[-1, -0.6)$, $[-0.6, -0.3)$, $[-0.3, -0.1)$, $[-0.1, 0.1]$, $(0.1, 0.3]$, $(0.3, 0.6]$, $(0.6, 1]$, $(1, 1.5]$, $(1.5, 2]$, $(2, 2.5]$, $(2.5, 3.5]$, $(3.5, 5]$, $(5, \text{Max}]$. It is observed that most of the difference profile values fall in the range of -1–2.5. Hence, the bins in this range (-1–2.5) are finely quantized with 10 bins and the values falling outside this range are captured by four coarsely quantized bins. The process of obtaining the feature vector from the difference profile is summarized below.

(1) Obtain the horizontal and vertical difference profiles (D_H, D_V) as explained above.
(2) Cascade these difference profiles to form final difference profile ($D_F = [D_H, D_V]$).
(3) Obtain the final feature vector (U) by constructing histogram of D_F with the specified bin-intervals. Let function $b : R \to 1, 2, \ldots, m$ associates the ith value of final differential profile (D_F^i) with the bin index $b(D_F^i) \in \{1, 2, \ldots, m\}$. Here, the value of m is 14 for the aforementioned bin-intervals

$$U_h = \sum_{i=1}^{n} \delta[b(D_F^i) - h], \tag{9}$$

where δ is the Kronecker delta function, n be the number elements in the final differential profile (D_F), and $h = 1, 2, \ldots, m$.

Since image quality is a subjective phenomenon, the human observer plays a major role in testing image quality metric. The subjective test designates the opinion of a viewer (opinion score) on a given image based on how it is perceived. The MOS is the average opinion score over all subjects. The aim of any quality metric is to predict the quality as close as possible to MOS. Hence, the objective here is to find the functional relationship between extracted HSV features and MOS to quantify image quality. To compute the unknown relationship, GAP-RBF network is used. In the next section we explain the network architecture and learning algorithm used for quantifying image quality.

4. GAP-RBFN image quality model

In recent years, many sequential learning algorithms have been developed to overcome the problems encountered in neural networks [26–28]. Here, radial basis function network (RBF) is used to approximate the functional relationship. These sequential learning algorithms perform better than the batch learning algorithms as they do not require retraining whenever new observations are received. In sequential learning algorithms, the training samples are presented only once to capture the functional relationship whereas in batch learning algorithms, the samples are presented many times. Hence, the sequential learning algorithms require less computational effort and memory requirement than the batch learning algorithms.

Most of the sequential learning algorithms employ some strategy to obtain a compact network to represent input–output relationship. Recently, Huang et al. [28] proposed a new sequential learning algorithm called GAP-RBF network. In this algorithm, the criteria for growing/pruning of hidden neurons is based on the significance of the neurons to the network output. The algorithm updates only the parameters of the nearest neuron to minimize the approximation error. Hence this method is economical from computational as well as memory requirement point of view. Here, we use GAP-RBFN to approximate the functional relationship between HVS-based features and MOS. Also, we show that the GAP-RBF based image quality model adapts its features when new image sets are presented. Finally, we compare the performance of the proposed GAP-RBF model with the existing NR and FR image quality metrics [1,10].

The GAP-RBF image quality model is shown in Fig. 3. The basic building block for GAP-RBF is the RBFN. In general, a RBFN consists of three layers of processing elements. The first layer is linear and only distributes the input signal, while the next layer is nonlinear and uses Gaussian functions. The third layer linearly combines the Gaussian outputs. In a GAP-RBF quality model, the inputs are the extracted HVS features (U) of a given image while the output is the approximated image quality ($\hat{Q}$).

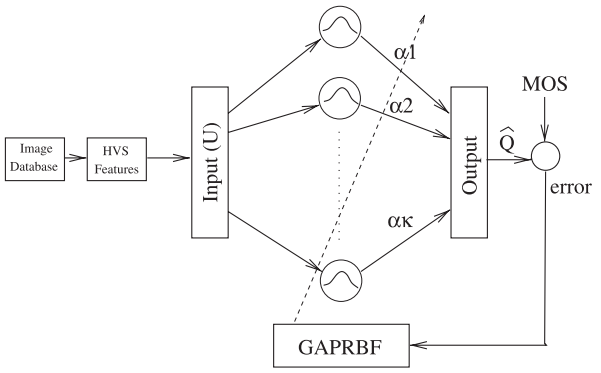

Fig. 3. Overview of the proposed image quality estimation algorithm.

The objective is to find a compact model to approximate the MOS using HVS-based features. The learning algorithm uses GAP strategy to decide on the significance of a neuron towards realizing a compact model. The network parameters such as center vectors, connection weights and widths of hidden neurons are tuned using extended Kalman filter (EKF) algorithm [28].

The output of an GAP-RBF quality model with K Gaussian neurons has the following form:

$$\hat{Q} = \sum_{i=1}^{K} \alpha_i \exp\left(-\frac{1}{\sigma_i^2}\|U-\mu_i\|\right), \tag{10}$$

where U is the HVS-based feature input vector, α_i is the weight connecting the ith Gaussian neuron to the output neuron, μ_i is the center vector of ith Gaussian neuron and σ_i is the width of the Gaussian neuron.

To begin with GAP-RBF is initialized with zero hidden neurons. As training images are received sequentially, the HVS-based features (inputs) are extracted and the network builds up based on a GAP criterion. The algorithm adds/prunes hidden neurons and also tunes the network parameters. Detailed description of the algorithms can be found in [28]. The following are the steps involved in obtaining a compact network.

4.1. GAP algorithm

Let $e_{\min}$ be the expected approximation error. Let us assume that K neurons are added to the GAP quality model after presenting $n-1$ observations. For each observation (U_n, MOS_n), where $U_n \in R^m$, $MOS_n \in R$, the following steps are carried out to learn the new sample.

(1) Estimate the image quality using the previously learned GAP-RBF model.

$$\hat{Q}_n = \sum_{i=1}^{K} \alpha_i \exp\left(-\frac{1}{\sigma_i^2}\|U_n-\mu_i\|\right). \tag{11}$$

Compute the learning error using the actual MOS value as

$$e_n = \hat{Q}_n - MOS_n.$$

(2) Compute the growth iteration:

$$\varepsilon_n = \max(\varepsilon_{\max}\gamma^n, \varepsilon_{\min}), \quad 0<\gamma<1. \tag{12}$$

The growth criterion ε_n is decayed exponentially in the entire input space, till it reaches the minimum value. Here, γ is the decay constant which controls the decay rate. Typical range of γ is between 0 and 1 for graceful neuron growth, while $\varepsilon_{\max}$ and $\varepsilon_{\min}$ are, respectively, the largest and smallest scales of interest in the input space with non-zero probability. Since the input features are normalized between 0 and 1, we can select $\varepsilon_{\max}$ in the range of [0.7–1] and $\varepsilon_{\min}$ in the range of [0–0.3].

(3) Adding neuron: Here a new Gaussian neuron is added based on growth criterion ε_n.
IF $\|U_n-\mu_{nr}\|>\varepsilon_n$ and $(1.8.\kappa\|U_n-\mu_{nr}\|)|e_n|/S(U)>e_{\min}$ then
Allocate $(K+1)$th hidden neuron.

Here nr is the nearest neighbor neuron of the new sample U_n, κ is the overlap parameter and $S(U)$ is the size of the input space ($S_u \approx 1$). The second condition $((1.8.\kappa\|U_n-\mu_{nr}\|)|e_n|/S(U)>e_{\min})$ in the growth criterion measures the significance of the nearest neighbor neuron. Significance of a neuron gives a measure of the information content in the neuron about the function to be learned. It is defined as the contribution made by that neuron to the network output averaged over all the input data received so far [28]. If the growing criteria are satisfied then allocate $(K+1)$th hidden neuron with

$$\begin{aligned}\alpha_{K+1} &= e_n,\\ \mu_{K+1} &= U_n,\\ \sigma_{K+1} &= \kappa\|U_n-\mu_{nr}\|.\end{aligned} \tag{13}$$

ELSE

Update the network parameters α_{nr}, μ_{nr} and σ_{nr}
for the nearest neuron only, using EKF algorithm [28].

Check the criterion for pruning the neurons:

IF $|(1.8\sigma_{nr})^l \alpha_{nr}/S(U)| < e_{\min}$, then,

Remove nrth hidden neuron and reduce the dimensionality of EKF algorithm.

Here, the pruning criterion is also derived based on the neuron significance. Hidden neurons which contribute very less to the network output over all learned samples are removed from the network [28]. Here, the parameter l is set to 1.

ENDIF

ENDIF

(4) Repeat the steps until all training samples are presented to the network.

5. Experiments and discussions

In our simulations, we have used the LIVE image quality assessment database [9]. Here, 29 JPEG images are used to generate a database of 204 JPEG images with different compression rates. including the original images, we have 233 JPEG images for image quality estimation. The study was conducted in two sessions (first session 116 images with 20 subjects (group-I) and the next session 117 images with 13 subjects (group-II)). Each observer was shown the images randomly and asked to mark the perception quality on a continuous linear scale that was divided into five equal regions marked with adjectives bad, poor, fair, good and excellent. The scale was then linearly transformed to 1–10 range. The resulting MOS was used to develop the GAP-RBF model to predict the image quality.

To develop the GAP-RBF image quality model, we have selected two disjoint sets of images for training and testing. The training set images and its compressed versions are not used in testing set. Out of 29 source images, 20 images were used for training and the remaining nine source images were used for testing. Totally 154 images were used for training (20 original and its 134 compressed versions) and 79 images for testing (nine original and its 70 compressed versions). Out of 154 training images, first 76 images are from group-I and the rest (77–154) are from group-II.

During training phase, we presented 154 training images sequentially to the GAP-RBF algorithm to develop the model. In our simulations, we set the following network parameters: $\varepsilon_{\min} = 0.001$, $\varepsilon_{\max} = 0.7$, $\gamma = 0.99$ and $\kappa = 0.7$. The expected minimum accuracy selected for our modeling is 0.01. The GAP-RBF network initialized with zero hidden neuron, builds the network based on the GAP strategy mentioned earlier. The neuron history (Fig. 4) shows that 25 neurons are required to approximate the functional relationship. From Fig. 4, we see that 19 neurons are required to learn from 79 group-I training images. Further six neurons are added while presenting the remaining training images from group-II. The neuron growth saturates at 25 after the 117th training image sample. There is no neuron growth for the last 37 training images which indicates the completion of learning phase.

After the training phase, the developed GAP-RBF network model is tested with the 79 test images. The correlation between MOS and GAP-RBF based image quality metric for training and test images are shown in Figs. 5 (a) and 6 (a). Similar study is carried out using Wang's NR quality metric (see Figs. 5(b) and 6(b)) [1] and full-reference SSIM index (Figs. 5(c) and 6(c)) (the SSIM index results are shown after fitting non-linear logistic function) [10].

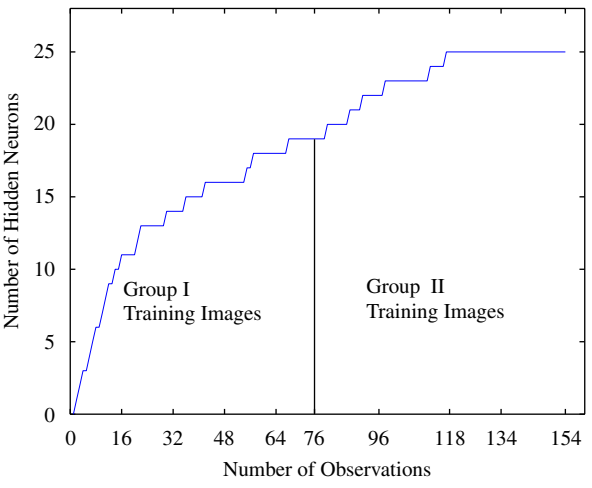

Fig. 4. Hidden neuron development.

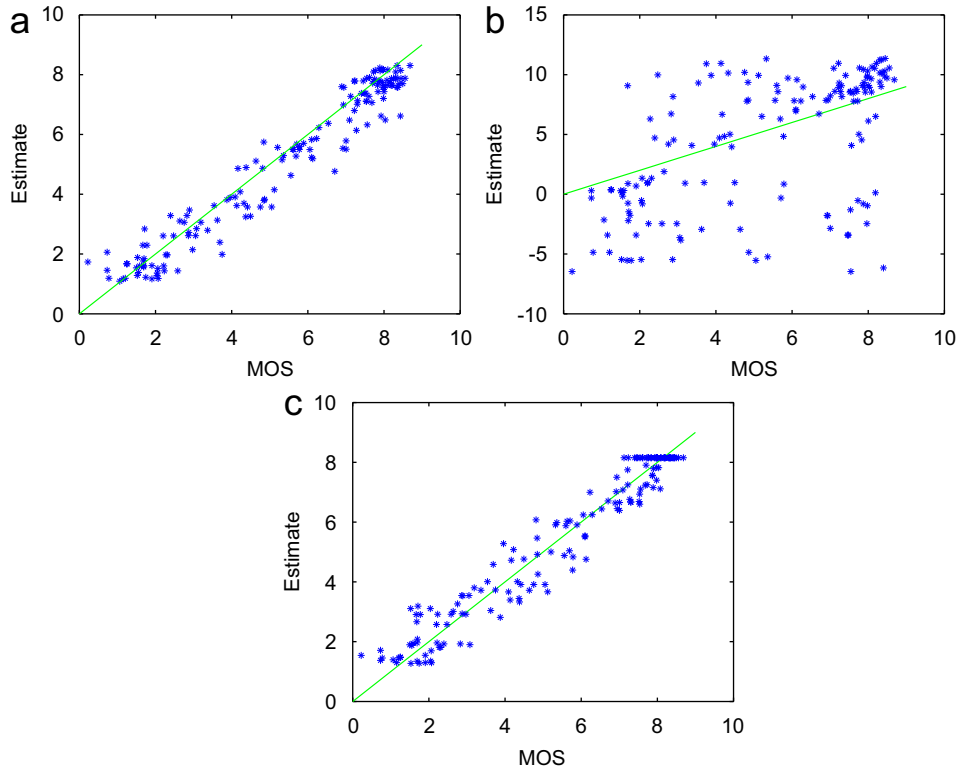

Fig. 5. Image quality prediction by: (a) proposed GAP-RBF method, (b) Wang–Bovik method and (c) SSIM score for 154 training images.

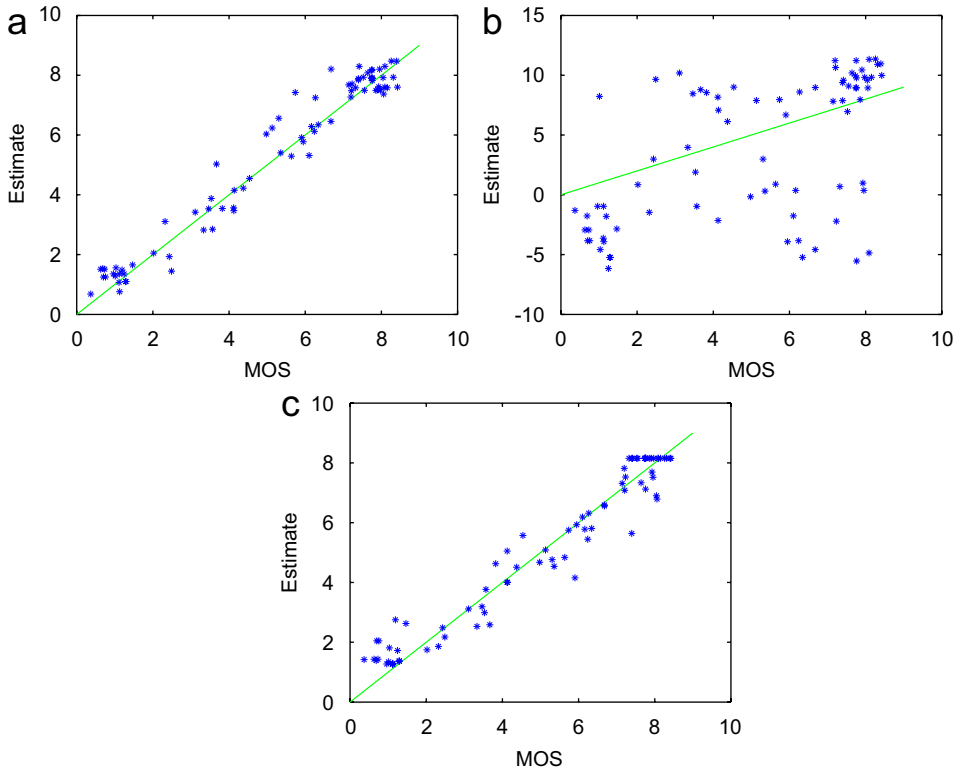

Fig. 6. Image quality prediction by: (a) proposed GAP-RBF method, (b) Wang–Bovik method and (c) SSIM score for 79 test images.

Table 1
RMSE between MOS and prediction

Metric	Testing	Training
GAP-RBF	0.57	0.66
Wang's	5.09	4.69
SSIM	0.65	0.63

Table 2
R-square estimate between MOS and prediction

Metric	Testing	Training
GAP-RBF	0.9564	0.9323
Wang's	−2.517	−2.393
SSIM	0.9431	0.9408

The results clearly show that the proposed GAP-RBF model predicts the image quality better than Wang's NR metric and comparable with the FR SSIM index. This can also be deduced from the quantitative performance analysis given in Tables 1 and 2. The root mean square error (RMSE) and deviation from MOS for image quality metric using different methods, given in Table 1. The commonly used R-square estimate for all three methods, given in Table 2.

6. Conclusions

In this paper, we have presented a system for predicting image quality using GAP-RBF network, considering various human visual characteristics. The functional relationship between the extracted HVS features and MOS is modeled by GAP-RBF network. Since sequential learning algorithm is used, GAP-RBF network does not require retraining when presented with a new data set. The developed GAP-RBF quality model can be updated easily in future, when the new set of samples arrive, with minimal computational and memory requirement. The performance of the proposed metric is found to be better than a previously reported NR image quality metric, and on par with an existing FR metric.

Acknowledgments

The authors would like to thank Prof. Bovik and his lab members for providing the JPEG image quality assessment database to test our metric. The authors express their thanks to the anonymous reviewers for their constructive comments to improve the quality of this paper.

References

[1] Z. Wang, H.R. Sheikh, A.C. Bovik, No-reference perceptual quality assessment of JPEG compressed images, in: Proceedings of the ICIP'02, vol. 1, 2002, pp. 477–480.
[2] D.S. Turaga, Y. Chen, J. Caviedes, No reference PSNR estimation for compressed pictures, Signal Process.: Image Communication 19 (2) (2004) 173–184.
[3] S.A. Karunasekera, N.G. Kingsbury, A distortion measure for blocking artifacts in images based on human visual sensitivity, IEEE Trans. Image Process. 4 (6) (1995) 713–724.
[4] S. Suthaharan, A perceptually significant block-edge impairment metric for digital video coding, in: Proceedings of the ICASSP'2003, vol. 3, Hong Kong, 2003, pp. 681–684.
[5] Video Quality Experts Group (VQEG), website: ⟨http://www.vqeg.org⟩.
[6] Y.L. Murphey, Y. Luo, Feature extraction for a multiple pattern classification neural network system, Pattern Recognition Proc. 2 (2002) 220–223.
[7] M. Voultsidou, S. Dodel, J.M. Herrmann, Neural networks approach to clustering of activity in fMRI data, IEEE Trans. Med. Imaging 24 (8) (2005) 987–996.
[8] E. Trentin, M. Gori, Robust combination of neural networks and hidden markov models for speech recognition, IEEE Trans. Neural Networks 14 (6) (2003) 1519–1531.
[9] H.R. Sheikh, Z. Wang, L. Cormack, A.C. Bovik, Live image quality assessment database, ⟨http://live.ece.utexas.edu/research/quality⟩.
[10] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 3 (4) (2004) 600–612.
[11] Z. Wang, A.C. Bovik, B.L. Evans, Blind measurement of blocking artifacts in images, in: Proceedings of ICIP'00, vol. 3, 2000, pp. 981–984.
[12] F. Pan, X. Lin, S. Rahardja, W. Lin, E. Ong, S. Yao, Z. Lu, X. Yang, A locally adaptive algorithm for measuring blocking artifacts in images and videos, Signal Process.: Image Communication 19 (6) (2004) 499–506.
[13] T. Vlachos, Detection of blocking artifacts in compressed video, Electron. Lett. 36 (13) (2000) 1106–1108.
[14] H.R. Wu, M. Yuen, A generalized block-edge impairment metric for video coding, IEEE Signal Process. Lett. 70 (3) (1998) 247–278.
[15] W. Gao, C. Mermer, Y. Kim, A de-blocking algorithm and a blockiness metric for highly compressed images, IEEE Trans. Circuits Syst. Video Technol. 12 (12) (2002) 1150–1159.
[16] M. Yuen, H.R. Wu, A survey of hybrid MC/DPCM/DCT video coding distortions, Signal Process. 4 (11) (1997) 317–320.
[17] M. Jung, D. Léger, M. Gazalet, Univariant assessment of the quality of images, J. Electron. Imaging 11 (3) (2002) 354–364.
[18] P. Gastaldo, R. Zunino, I. Heynderickx, E. Vicario, Objective quality assessment of displayed images by using neural networks, Signal Process.: Image Communication 20 (2005) 643–661.

[19] P. Gastaldo, R. Zunino, Neural networks for the no-reference assessment of perceived quality, J. Electron. Imaging 14 (3) (2005) 33004:1–11.
[20] F. Yang, S. Wan, Y. Chang, H.R. Wu, A novel objective no-reference metric for digital video quality assessment, IEEE Signal Process. Lett. 12 (10) (2005) 685–688.
[21] H.R. Sheikh, A.C. Bovik, L. Cormack, No-reference quality assessment using natural scene statistics: JPEG2000, IEEE Trans. Image Process. 14 (11) (2005) 1918–1927.
[22] M.P. Eckert, A.P. Bradley, Perceptual quality metrics applied to still image compression, Signal Process. 70 (3) (1998) 177–200.
[23] Z. Wang, H.R. Sheikh, A.C. Bovik, Objective video quality assessment, Handbook of Video Databases: Design and Applications, 2003, pp. 1041–1078.
[24] J.-B. Martens, Image technology Design—A Perceptual Approach, Kluwer, Dordrecht, 2003.
[25] R.V. Babu, A. Bopardikar, A. Perkis, A perceptual no-reference blockiness metric for JPEG images, in: Indian Conference on Vision Graphics and Image Processing, 2004, pp. 455–460.
[26] N.B. Karayiannis, G.W. Mi, Growing radial basis neural networks: merging supervised and unsupervised learning with network growth techniques, IEEE Trans. Neural Networks 8 (6) (1997) 1492–1506.
[27] L. Yingwei, N. Sundararajan, P. Saratchandran, A sequential learning scheme for function approximation using minimal radial basis function (RBF) neural networks, Neural Comput. 9 (1997) 461–478.
[28] G.-B. Huang, P. Saratchandran, N. Sundararajan, An efficient sequential learning algorithm for growing and pruning RBF (GAP-RBF) networks, IEEE Trans. Syst. Man Cybern.: Part B Cybernetics 34 (6) (2004) 2284–2292.