Learning without Human Scores for Blind Image Quality Assessment

Wufeng Xue
Inst. of Im. Pro. & Pat. Rec
Xi'an Jiaotong Univ.
xwolfs@hotmail.com

Lei Zhang
Dept. of Computing
The Hong Kong Polytechnic Univ.
cslzhang@comp.polyu.edu.hk

Xuanqin Mou
Inst. of Im. Pro. & Pat. Rec
Xi'an Jiaotong Univ.
xqmou@mail.xjtu.edu.cn

Abstract

General purpose blind image quality assessment (BIQA) has been recently attracting significant attention in the fields of image processing, vision and machine learning. State-of-the-art BIQA methods usually learn to evaluate the image quality by regression from human subjective scores of the training samples. However, these methods need a large number of human scored images for training, and lack an explicit explanation of how the image quality is affected by image local features. An interesting question is then: can we learn for effective BIQA without using human scored images? This paper makes a good effort to answer this question. We partition the distorted images into overlapped patches, and use a percentile pooling strategy to estimate the local quality of each patch. Then a quality-aware clustering (QAC) method is proposed to learn a set of centroids on each quality level. These centroids are then used as a codebook to infer the quality of each patch in a given image, and subsequently a perceptual quality score of the whole image can be obtained. The proposed QAC based BIQA method is simple yet effective. It not only has comparable accuracy to those methods using human scored images in learning, but also has merits such as high linearity to human perception of image quality, real-time implementation and availability of image local quality map.

1. Introduction

With the ubiquitous use of digital imaging devices (e.g., digital cameras and camera phones) and the rapid development of internet service, digital images have been becoming one of the most popular types of media in our daily life. For example, one can easily find a huge amount of images in Google, Facebook and Flicker, etc. The quality of those images can be deteriorated due to noise corruption, blur, JPEG or JPEG 2000 compression, etc. However, in most scenarios we do not have the source of the distorted image, and consequently how to evaluate blindly the quality of an image has been becoming increasingly important [22]. The current blind image quality assessment (BIQA) methods can be classified into two categories: distortion specific methods [1, 8, 9, 18, 25] and distortion independent methods [4, 10, 13, 14, 16, 17, 21, 27]. The former category estimates the quality of an image by quantifying the particular artifacts induced by the distortion process, and usually works well for one specific type of distortion. The latter category often refers to the general purpose BIQA, which is clearly a much more challenging task than the former category due to the lack of distortion information. In this paper we focus on the general purpose BIQA methods.

Most of the state-of-the-art BIQA methods [4, 10, 13, 14, 16, 17, 21, 27] learn to estimate the image quality from training samples whose human subjective quality scores are available, e.g., the images in the TID2008 [15], LIVE [19] and CSIQ [6] databases. Generally speaking, all these methods follow a two-step framework: feature extraction and model regression by human scores. The method proposed by Moorthy et al. [13] first uses a support vector machine (SVM) to detect the distortion type and then uses a support vector regression (SVR) [20] model specified to that distortion for BIQA. Saad et al. trained a probabilistic model for BIQA based on the contrast and structural features such as kurtosis and anisotropy in the DCT domain [16]. The BIQA metric in [21] extracts three sets of features based on the statistics of natural images, distortion textures and blur/noise. Three regression models are then trained for each feature set and finally a weighted combination of them is used to estimate the image quality. A summarization of the used features and the regression algorithms in recently developed BIQA methods can be found in [27]. The mostly widely used algorithm for regression is the SVR with a radial basis function as kernel. In [4], the sparse representation based classifier firstly developed in face recognition literature [26] was used to regress the image quality score.

Though the above methods represent the state-of-the-arts of BIQA research, there are several important issues to be further addressed. First of all, all these methods need a large amount of human scored images for training. This makes the developed algorithm training dataset dependent,

(a) (b) (c) (d)

Figure 1. (a) The ten training images used by us. They are randomly selected from the Berkeley Segmentation database [7]. Reference images in the (b) LIVE database [19]; (c) TID2008 database [15]; and (d) CSIQ database [6].

and the results are heavily dependent on the size of training samples. Second, these methods usually learn a mapping function (e.g., using SVR) to map the extracted image features (e.g., global statistics) to a single perceptual score. This makes the BIQA process a black box and the relationship between features and quality score implicit. None of these methods can provide a local quality map of the distorted image, which is much desirable to understand the good and bad quality regions of the input image. Third, some of these methods can achieve relatively high BIQA accuracy, but their complexity is too high to be implemented in real-time, limiting their practical use.

Intuitively, one interesting question is can we develop an effective and efficient BIQA algorithm but without using human scored images for training? In [11], Mittal et al. ever proposed such an algorithm by conducting probabilistic latent semantic analysis (pLSA) on the statistical features of a large collection of pristine and distorted image patches. The uncovered latent quality factors are then applied to the image patches of the test image to infer a quality score. However, this method does not perform well compared with those methods learning with human scoring information.

In this paper, we present a novel solution to BIQA using no human scored images in learning. They key is that we propose a quality-aware clustering (QAC) method to learn a set of quality-aware centroids and use them as the codebook to infer the quality of an image patch so that the quality of the whole image can be determined. With some reference and distorted images (but without human score), we partition them into overlapped patches and use a percentile pooling strategy to estimate the quality of each patch. According to the estimated quality level, the patches are grouped into different groups, and QAC is applied to each group to learn the quality-aware centroids. In the testing stage, each patch of the distorted image is compared to the learned quality-aware centroids, and a simple weighted average operation is used to assign a score to it. The perceptual quality score of the whole image can then be figured out by summing over all patches.

The proposed QAC based BIQA method is simple yet effective. Our experimental results validate that it has comparable accuracy to those state-of-the-art methods learning from human scored images. The QAC method has the following feature points. First, it shows that even without using human scored images for training, we are still able to develop effective BIQA algorithms. Second, it builds an explicit relationship between the image feature and the quality score, and could provide a local quality map of the input image, which is not achievable by all the other BIQA methods. Third, the proposed QAC is very fast and can work in real-time, making it applicable to devices with limited computational resources (e.g., cell phones). At last, QAC has a very high linearity to human perception of image quality.

The rest of the paper is organized as follows. The learning of quality-aware centroids by QAC is described in detail in Section 2. Then how to use the learned centroids to perform blind quality estimation is described in Section 3. Experiments and discussions are detailed in Section 4. Finally, Section 5 concludes the paper.

2. Quality-aware clustering

2.1. Learning dataset generation

Our method works on image patches and aims to learn a set of quality-aware centroids for blind image quality assessment (BIQA). To this end, we need some reference and distorted images for training but do not need to know the human subjective scores of the distorted images. Considering that the existing IQA databases [6,15,19] will be used to evaluate and compare the different BIQA algorithms in the experiments, we do not use them in our method to better validate the generality and database-independency of our approach. Instead, we randomly selected from the Berkeley image database [7] ten source images (please refer to Fig. 1(a)), which have different scenes from the images in the databases [6,15,19] that will be used in our experiments (please refer to Fig. 1(b)$\sim$ Fig. 1(d) for these images).

We then simulated the distorted images of the ten images. The four most common types of distortions are simulated: Gaussian noise, Gaussian blur, JPEG compression and JPEG2000 compression. These four distortion types are also the ones TID2008, LIVE and CSIQ databases have in common. For each image, we generate its distorted versions of each type on three quality levels by controlling the noise standard deviation (for distortion of Gaussian noise), the support of blur kernel (for distortion of Gaussian blur), the resulted quality level (for distortion of JPEG compression) and the compression ratio (for distortion of JPEG2000

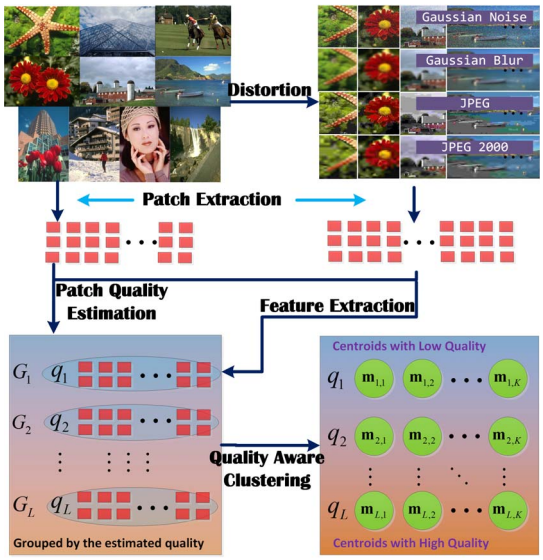

Figure 2. Flowchart of the proposed quality-aware clustering (QAC) scheme.

compression), respectively. Finally, we obtain a dataset of 120 distorted images and 10 reference images. A choice of the three quality levels should make sure that the quality distribution of the resulted samples in the next section is balanced.

2.2. Patch quality estimation and normalization

With the simulated dataset which has no human subjective quality score, we aim to learn a set of quality-aware centroids for BIQA. The flowchart of our learning scheme is illustrated in Fig. 2. We partition the reference and distorted images into many overlapped patches. Denote by $\mathbf{x}_i$ a patch of one reference image and by $\mathbf{d}_i$ the distorted version of it. One key problem in our method is how to assign a perceptual quality to $\mathbf{d}_i$. To this end, we can first use the similarity function in some state-of-the-art full-reference image quality assessment (FR-IQA) method, such as SSIM [23] and FSIM [29], to calculate the similarity between $\mathbf{x}_i$ and $\mathbf{d}_i$. By this way, the dependency on human score are removed. In this paper, we use FSIM:

$$s_i = S(\mathbf{x}_i, \mathbf{d}_i) = \frac{2PC(\mathbf{x}_i)PC(\mathbf{d}_i) + t_1}{PC(\mathbf{x}_i)^2 + PC(\mathbf{d}_i)^2 + t_1} \times \frac{2G(\mathbf{x}_i)G(\mathbf{d}_i) + t_2}{G(\mathbf{x}_i)^2 + G(\mathbf{d}_i)^2 + t_2} \quad (1)$$

where $PC(\mathbf{x}_i)$ and $G(\mathbf{x}_i)$ refer to the phase congruency [5] and gradient magnitude at the center of $\mathbf{x}_i$, respectively, and t_1 and t_2 are positive constants for numerical stability.

The similarity score s_i can reflect the quality of $\mathbf{d}_i$ to some extent, and it ranges from 0 to 1. In FR-IQA, we usually simply take s_i as the local quality score of $\mathbf{d}_i$, and average all s_i in one image as the final quality score of this image. Such a simple strategy works well for FR-IQA since the availability of reference image. However, our goal here is to learn for performing BIQA, and taking s_i as the quality score of $\mathbf{d}_i$ will have some problem. Suppose that the real human scored quality of a distorted image $\mathbf{d}$ is s, if we take s_i as the local quality score of its patch $\mathbf{d}_i$, then the average of all s_i can be very different from s, leading to much bias in the learning stage.

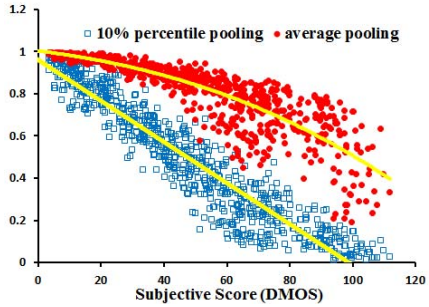

Figure 3. The effect of percentile pooling. Y-axis denotes the prediction score by IQA models. Note that the mean values of the lowest 10% predicted quality scores shows much better linearity to the human subjective scores.

To solve this problem, we must normalize s_i in order to make the average of all s_i in an image as close to its overall perceptual quality as possible. It is known that the similarity functions in FR-IQA methods can only give a nonlinear monotonic prediction of the human subjective score [23, 29]. Fig. 3 shows an example on the LIVE database. The red round point shows the FR-IQA results by FSIM with average pooling versus the subjective score with a two-order polynomial fitting. It is this nonlinearity that often makes the estimated quality score deviate from the human perception. On the other hand, it has been found that in an image, the predicted quality of the worst local areas has a good linearity to human perception [12, 24]. The blue squared points in Fig. 3 shows the worst 10% percentile pooling results of FSIM versus the subjective score, which has much better linearity. Based on this finding, we propose a percentile pooling procedure to normalize s_i. In particular, we divide s_i by a constant C such that the average quality of all patches in an image will equal to the percentile pooling result.

Denote by Ω the set of patch indices of an image, and by Ω_p the set of indices of the 10% lowest quality patches. The normalization factor C is calculated as:

$$C = \frac{\sum_{i\in\Omega} s_i}{10\sum_{i\in\Omega_p} s_i} \quad (2)$$

Then each s_i is normalized as: $c_i = {}^{s_i}/_{C}$.

2.3. Quality-aware clustering

With the patch quality normalization strategy in Section 2.2, finally we can have a set of patches $\{\mathbf{d}_i\}$ and their normalized quality scores $\{c_i\}$, based on which the quality-aware clustering can be conducted. The idea is that with

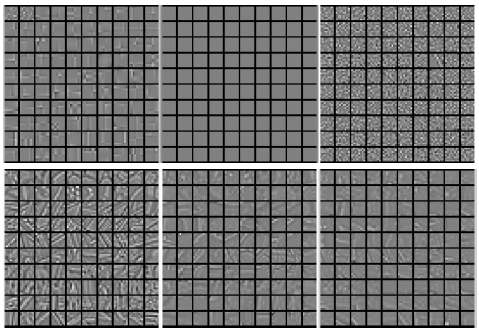

Figure 4. Examples of the quality-aware clustering outputs. Top row: 3 clusters on quality level $q_l = 0.1$; bottom row: 3 clusters on quality level $q_l = 1$.

$\{c_i\}$ in hand, we can group $\{\mathbf{d}_i\}$ into groups of similar quality, and then cluster those patches in the same quality group into different clusters based on their local structures.

Since c_i is a real-value number between 0 and 1, we first uniformly quantize c_i into L levels, denoted by $q_l = l/L$, $l = 1, 2, \ldots L$. Then the patches having the same quality level are grouped into the same group, denoted by $\mathbf{G}_l$. There is:

$$\mathbf{G}_l = \begin{cases} \{\mathbf{d}_i | q_{l-1} < c_i \leq q_l, & for\ l = 2 \ldots L\} \\ \{\mathbf{d}_i | c_i \leq q_l, & for\ l = 1\} \end{cases} \quad (3)$$

The clustering is then applied to each group $\mathbf{G}_l$. Since the quality of each group is aware, we call this clustering quality-aware clustering (QAC).

To enhance the clustering accuracy, the QAC within each $\mathbf{G}_l$ should be based on some structural feature of $\mathbf{d}_i$. In this paper, we use the following high pass filter to extract the feature of patch $\mathbf{d}_i$:

$$\mathbf{h}_\sigma(r) = 1_{r=0} - \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{r^2}{2\sigma^2}) \quad (4)$$

where σ is the scale parameter to control the shape of the filter. By convolving $\mathbf{h}_\sigma$ with the image, the image detailed structures will be enhanced. It has been shown that the profile of the receptive field of the ganglion in the early stage of human vision is analogous to the shape of the difference of Gaussian (DoG) filter [28]. The filter defined in Eq. 4 is a special case of DoG filter when the support size of the first Gaussian shrinks to 1.

In our implementation, we use three $\mathbf{h}_\sigma$ on different scales (σ = 0.5, 2.0, 4.0 in our experiments) to extract the feature of $\mathbf{d}_i$. The filtering outputs of $\mathbf{d}_i$ on the three scales are concatenated into a feature vector, denoted by $\mathbf{f}_i$. The QAC of $\mathbf{d}_i \in \mathbf{G}_l$ is then performed by applying the K-mean clustering algorithm to $\mathbf{f}_i$:

$$\mathbf{min}_{\mathbf{m}_{l,k}} \sum_{k=1}^{K} \sum_{\mathbf{d}_i \in \mathbf{G}_{l,k}} \|\mathbf{f}_i - \mathbf{m}_{l,k}\|^2 \quad (5)$$

where $\mathbf{G}_{l,k}$ is the k^{th} cluster in Group $\mathbf{G}_l$. Note that other similarity metric may be used for clustering. However, given the complexity cost, we just use the Euclidean distance. Besides, in the framework of quality aware clustering, this is not necessary. For the clustering, we use the spectrum clustering in [2], which is efficient to solve Eq. 5. As a result, for each group $\mathbf{G}_l$, we learn a set of centroids $\{\mathbf{m}_{l,k}\}$, $k = 1, 2, \ldots K$. Finally, we have L sets of centroids on L different quality levels, and we call them quality-aware centroids. Those centroids will then act as a structured codebook to encode the quality of each patch so that the overall quality of the image can be inferred.

In Fig. 4, we show three clusters of patches on the worst quality level ($q_l = 0.1$) and the best quality level ($q_l = 1$), respectively, by setting $L = 10$ and $K = 30$. One can see that the cluster of patches on the worst quality level exhibit obvious compression, blur and noise like distortions, while the clusters on the best quality level exhibit Gabor-like structures. These observations accord with the widely recognized conclusion that the visual receptive fields in the primary visual cortex (V1) are local orientated.

3. Blind quality pooling

With the learned quality-aware centroids $\{\mathbf{m}_{l,k}\}$ in Section 2, for each given distorted image, denoted by $\mathbf{y}$, we can easily estimate its perceptual quality by following the procedures: patch partition and feature extraction, cluster assignment on multiple quality levels, patch quality score estimation, and final pooling with all patches' quality.

Patch partition and feature extraction: For the test image $\mathbf{y}$, we partition it into N overlapped patches $\mathbf{y}_i$, and use the high pass filters $\mathbf{h}_\sigma$ to extract the feature vector, denoted by $\mathbf{f}_i^y$, of each $\mathbf{y}_i$, $i = 1, \ldots, N$.

Cluster assignment: By assuming that patches which have similar structural features will have similar visual quality, on each quality level l we find the nearest centroid to the feature vector $\mathbf{f}_i^y$ of patch $\mathbf{y}_i$. Denote by $\mathbf{m}_{l,k_i}$ this nearest centroid on level l. Then we will assign $\mathbf{y}_i$ to L clusters defined by $\mathbf{m}_{l,k_i}$, $l = 1, \ldots L$. The quality of patch $\mathbf{y}_i$ can be computed as the weighted average of the quality levels of these centroids.

Patch quality estimation: The distance between $\mathbf{f}_i^y$ and $\mathbf{m}_{l,k_i}$ is $\delta_{l,i} = \|\mathbf{f}_i^y - \mathbf{m}_{l,k_i}\|^2$. Clearly, the shorter the distance $\delta_{l,i}$ is, the more likely patch $\mathbf{y}_i$ should have the same quality level as that of centroid $\mathbf{m}_{l,k_i}$. Therefore, we can use the following weighted average rule to determine the final quality score of $\mathbf{y}_i$:

$$z_i = \frac{\sum_{l=1}^{L} q_l \exp(-\delta_{l,i}/\lambda)}{\sum_{l=1}^{L} \exp(-\delta_{l,i}/\lambda)} \quad (6)$$

where λ is a parameter to control the decay rate of weight $\exp(-\delta_{l,i}/\lambda)$ w.r.t. distance $\delta_{l,i}$. One can see that the distance based weighted average in Eq. 6 actually interpolates the real-valued quality score of patch $\mathbf{y}_i$ from the discrete

quality levels q_l. This makes the quality estimation more robust and more accurate.

Final pooling: With the estimated quality z_i of all patches $\mathbf{y}_i$ available, we can then infer the final single quality score, denoted by z, of test image $\mathbf{y}$. Various pooling strategies such as max pooling and percentile pooling have been proposed in literature [12, 27]. Here we use the simplest average pooling:

$$z = \frac{1}{N}\sum_{i=1}^{N} z_i \tag{7}$$

It can be seen that the testing stage of our method is very simple, while our experimental results in next section demonstrate its competitive performance. The complexity analysis and running time comparison can be found in Section 4.3, where we can see that the proposed method can run in real time, making it a very good choice for practical BIQA applications in various resource-limited devices.

4. Experimental results

4.1. Protocol

The performance of QAC is validated in terms of its ability to predict the subjective ratings of image quality. The three largest publicly available subject-rated databases are employed: LIVE [19], CSIQ [6] and TID2008 [15]. For each image in these database, a subjective quality/distortion score, i.e., the mean opinion score (MOS) or difference mean opinion score (DMOS), is assigned to validate the BIQA algorithms.

The LIVE database consists of 779 distorted images generated from 29 original images by processing them with 5 types of distortions on various levels: JPEG2000 compression (JP2K), JPEG compression, additive white noise (WN), Gaussian blurring (GB) and simulated fast fading Rayleigh channel (FF). These distortions reflect a broad range of image impairments, for example, edge smoothing, block artifacts and random noise. The CSIQ database is composed of 30 original images and their distorted counterparts by using six types of distortions on five different distortion levels. The TID2008 database is composed of 25 reference images and their distorted versions of 17 types on 4 levels. As in many previous works [4, 14, 16], in our experiments we only consider 4 types of distortions that are common to the three databases: JPEG2000, JPEG, WN and GB. Those four types of distortions are also the most commonly encountered distortions in practical applications.

To evaluate the performance of a BIQA metric, two correlation coefficients between the prediction results and the subjective scores are adopted: the Spearman rank order correlation coefficient (SROCC), which is related to the prediction monotonicity, and the Pearson correlation coefficient (PCC), which is related to the prediction linearity. A good BIQA method will demonstrate a big (close to 1) correlation coefficient with the subjective score MOS or a small (close to -1) correlation coefficient with DMOS.

Table 1. Blind image quality assessment results on LIVE.

SROCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8505	0.9401	0.9613	0.9094	0.8857
pLSA [11]	Blind	0.85	0.88	0.80	0.87	0.80
PSNR	FR	0.8954	0.8803	0.9853	0.7829	0.8749
SSIM	FR	0.9614	0.9764	0.9694	0.9517	0.9479
FSIM	FR	0.9717	0.9834	0.9652	0.9708	0.9685
PCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8381	0.9326	0.9236	0.9064	0.8608
pLSA [11]	Blind	0.87	0.90	0.87	0.88	0.79
PSNR	FR	0.8726	0.8654	0.979	0.7746	0.8578
SSIM	FR	0.8925	0.9279	0.9583	0.8881	0.829
FSIM	FR	0.9015	0.9071	0.9085	0.9084	0.8647

Table 2. Blind image quality assessment results on CSIQ.

SROCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8704	0.9126	0.8624	0.8483	0.8627
PSNR	FR	0.9361	0.8879	0.9363	0.9291	0.9218
SSIM	FR	0.9605	0.9543	0.8974	0.9608	0.9325
FSIM	FR	0.9685	0.9654	0.9262	0.9729	0.9616
PCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8822	0.9376	0.8735	0.8439	0.8768
PSNR	FR	0.927	0.7898	0.9438	0.9081	0.8463
SSIM	FR	0.8966	0.9165	0.8044	0.8692	0.8622
FSIM	FR	0.9073	0.9026	0.7642	0.8838	0.8795

Table 3. Blind image quality assessment results on TID2008.

SROCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8885	0.8981	0.7070	0.8504	0.8697
PSNR	FR	0.8249	0.8762	0.9183	0.9336	0.8703
SSIM	FR	0.9603	0.9354	0.8168	0.9598	0.9016
FSIM	FR	0.9763	0.9263	0.8571	0.9526	0.9526
PCC	Blind/FR	JP2K	JPEG	WN	GB	ALL
QAC	Blind	0.8778	0.9235	0.7200	0.8500	0.8377
PSNR	FR	0.8814	0.8681	0.9416	0.9271	0.8361
SSIM	FR	0.9472	0.9469	0.7576	0.8906	0.8927
FSIM	FR	0.9555	0.9312	0.7827	0.9073	0.9288

4.2. Implementation details and results of QAC

In the implementation, we partition the 130 training images into overlapped patches of size 8×8. In total, 161,181 patches are extracted for training. In feature extraction, we set the three scales of high pass filters (refer to Eq. 4) as $\sigma = 0.5, 2.0, 4.0$. In clustering, we quantize the quality into $L = 10$ levels; that is, q_l is from 0.1 to 1 with step length 0.1. On each quality level, $K = 30$ clusters are clustered by using the clustering algorithm in [2]. These centroids together form a codebook to encode the quality of test images. In the test stage, we set the parameter λ in Eq. 6 as 32. The Matlab source code of the proposed QAC can be downloaded at http://www.comp.polyu.edu.hk/ cslzhang/code.htm.

The BIQA results (in terms of SROCC and PCC) of QAC

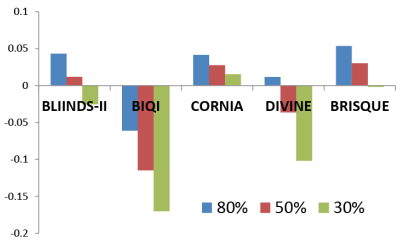

Figure 5. Average performance (SROCC) gain of the existing BIQA methods over the proposed QAC.

on the three databases are listed in Tables 1, 2, 3. The FR-IQA metrics PSNR, SSIM and FSIM are used for reference. Considering that pLSA [11] is the only one existing BIQA method which does not use the human subjective scores for training, we also list its result on LIVE here. (Results of pLSA on the other databases are not provided in [11].) Table 1 shows that the proposed QAC wins a large margin over pLSA in terms of both SROCC and LCC on LIVE. Compared with PSNR, QAC performs better on JPEG and GB distortions, and has better results in the overall database. Table 2 shows that on CSIQ QAC gives a good SROCC and better PCC than PSNR and SSIM. Table 3 shows that on TID2008, QAC outperforms PSNR on JPEG and JP2K, and has almost the same overall SROCC and PCC as PSNR.

In FR-IQA and BIQA, we always hope that quality prediction results of one method could be linearly proportional to the subjective score, i.e., the so-called linearity, so that one can avoid the additional optimisation procedures in the nonlinear mapping [3] and guarantee a consistent results between different distortion types. The linearity of QAC, PSNR, SSIM and FSIM are visualized as the scatter plots in Fig. 7. Clearly, QAC show much better linearity than PSNR, SSIM and FSIM. This makes QAC a very suitable blind quality estimator since there is no need of some nonlinear transformation to get the final prediction results.

4.3. Comparison with state-of-the-arts

We then compare QAC with state-of-the-art and representative BIQA methods, including BIQI [13], DIIVINE [14], BLIINDS-II [17], CORNIA [27] and BRISQUE [10]. Note that all these methods use the human scored images for learning. The codes of these methods are provided by the authors and we tune the parameters to achieve their best results.

Table 4 shows the results of the competing methods on the LIVE, CSIQ and TID2008 databases. Due the limit of space, we only present the SROCC results here since it is the most important index to evaluate BIQA metrics. (In fact, similar conclusions can be obtained by the PCC results.) Except for the proposed QAC, all the other methods need to partition the IQA database into a training set and a testing set. We present their results under three settings: 80%, 50% and 30% samples are used for training and the remaining

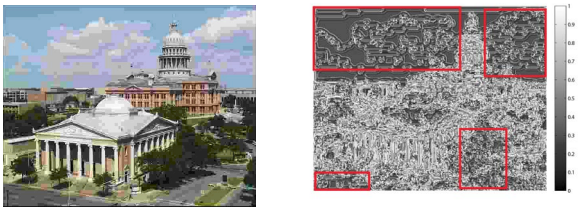

Figure 6. Left: JPEG distorted image from the LIVE database; Right: the local quality map predicetd by the proposed QAC method. The areas highlighted by red rectangles are of the worst quality, which are identical to human perception.

for testing. The partition is randomly conducted 1000 times and the average results are shown here. We also show the weighted average SROCC results over the three databases in Table 4, and the weights are based on the number of samples in the three databases.

From Table 4, we can see that QAC always performs significantly better than BIQI under different ratio of training samples. When 80% samples are used for training in the competing methods, QAC has a little lower SROCC than DIIVINE, and about 0.04∼0.05 lower than others. When 50% samples are used for training, QAC outperforms DIIVINE and lags behind BLINDS-II, CORNIA and BRISQUE. When 30% samples are used for training, QAC gives comparable results with CORNIA, and outperforms all the other methods. To make the above findings easier to observe, we draw in Fig. 5 the average SROCC gains of these competing methods over QAC under different ratio of training samples. We can see that the performance of most existing BIQA methods decreases rapidly with the decrease of the number of training samples. The CORNIA method has relatively good robustness to the number of training samples. However, it uses a 20,000-dimensional feature vector to represent each local descriptor, which may consume excess memory in implementation.

At last, since the proposed QAC evaluates each patch of an image to pool out the final score, it can naturally give a local quality map (LQM) of the distorted image, in which the value of each location indicates the quality of the surrounding patch. It enables us to tell the good regions from the heavily distorted regions in the image. Note that none of the existing BIQA methods [4,10,13,14,16,17,21,27] could give such an LQM. Fig. 6 shows an example. The left is a JPEG distorted image in the LIVE database, and the right is its LQM predicted by QAC. The red rectangles mark the most annoying regions introduced by the distortion. We can see that prediction is approximately identical to the human subjective perception of this image.

4.4. Computational complexity

Speed is another important factor to evaluate a BIQA method because in many practical applications we need to

Table 4. The SROCC comparison between QAC and other BIQA methods learning from human scored images.

BLIINDS-II [17]	Ratio of Samples for training			**DIIVINE** [14]	Ratio of Samples for training		
	80%	50%	30%		80%	50%	30%
LIVE	0.9425	0.9198	0.8973	LIVE	0.8946	0.8768	0.7954
CSIQ	0.9003	0.8832	0.8465	CSIQ	0.8697	0.8246	0.7838
TID2008	0.8982	0.8310	0.7690	TID2008	0.8930	0.7902	0.7132
Average	**0.9163**	**0.8851**	**0.8480**	**Average**	**0.8850**	**0.8369**	**0.7716**
CORNIA [27]	Proportion of Samples for training			**BRISQUE** [10]	Ratio of Samples for training		
	80%	50%	30%		80%	50%	30%
LIVE	0.9528	0.9414	0.9277	LIVE	0.9557	0.9410	0.9094
CSIQ	0.8845	0.8706	0.8605	CSIQ	0.9085	0.8857	0.8628
TID2008	0.8990	0.8814	0.8680	TID2008	0.9085	0.8696	0.8228
Average	**0.9147**	**0.9009**	**0.8886**	**Average**	**0.9270**	**0.9035**	**0.8716**
BIQI [13]	Ratio of Samples for training			**QAC**	N.A		
	80%	50%	30%				
LIVE	0.8429	0.7993	0.7484	LIVE	0.8857		
CSIQ	0.7598	0.7208	0.6721	CSIQ	0.8627		
TID2008	0.8438	0.7510	0.6778	TID2008	0.8697		
Average	**0.8123**	**0.7587**	**0.7034**	**Average**	**0.8733**		

Table 5. Computational complexity analysis.(*N* denotes the total number of pixels in the test image)

	Runtime (s)	Complexity	Notes
BLIINDS-II [17]	123.9	$O(^1/_{d^2} N \log(^N/_{d^2}))$	d: blocksize
DIIVINE [14]	28.20	$O(N(\log N + m^2 + N + 392b))$	m: neighbour size in DNT, b: # of bins in the 2-D histgram
CORNIA [27]	3.246	$O(Nd^2K)$	d: blocksize, K: codebook size
QAC	0.189	$O(N(^{h^2}/_{s^2}))$	s: block step; h: filter window size
BRISQUE [10]	0.176	$O(Nd^2)$	d: filter window size
BIQI [13]	0.076	$O(N)$	

judge the quality of an input image online. In Table 5, we summarize the computational complexity and the running time (the average processing time on the LIVE database) in the test stage of all competing methods[1]. We can see that BLIINDS-II and DIIVINE are the slowest, while BIQI is the fastest (only takes 0.0768s per image). However, the accuracy of BIQI is much worse than other methods (refer to Table 4 please). CORNIA has good robustness to the number of training samples with good accuracy; however, it takes over 3s to process an image. The proposed QAC has a similar speed to BRISQUE, and both of them need less than 0.2s to process an image. Overall, QAC provides a real-time solution to high performance BIQA.

5. Conclusions

We presented a novel general purpose blind image quality assessment (BIQA) approach, which is completely free of the human subjective scores in learning. The key of the proposed approach lies in the developed quality-aware clustering (QAC) scheme, which could learn a set of quality-aware centroids to act a codebook to estimate the quality levels of image patches. Via extensive experimental validations, we could have the following conclusions. First, as a database independent method, the proposed QAC achieves competitive SROCC results with those state-of-the-art BIQA methods which heavily exploit the human subjective scores in training. Second, QAC has very good linearity to human perception of image quality. Third, it can provide a local quality map of the distorted image, which is not available by other BIQA methods. At last, QAC provides a real-time solution to BIQA applications.

[1]The code of pLSA is not accessible, we didn't list its runtime. However, the EM algorithm used to estimate the frequencies of the 400 visual words makes it computational costly.

References

[1] J. Caviedes and S. Gurbuz. No-reference sharpness metric based on local edge kurtosis. In *ICIP*, 2002.

[2] X. Chen and D. Cai. Large scale spectral clustering with landmark-based representation. In *AAAI*, 2011.

[3] V. Q. E. Group et al. Final report from the video quality experts group on the validation of objective models of video quality assessment, phase II. *VQEG, Aug*, 2003.

[4] L. He, D. Tao, X. Li, and X. Gao. Sparse representation for blind image quality assessment. In *CVPR*, 2012.

[5] P. Kovesi. Image features from phase congruency. *Journal of Computer Vision Research*, 1(3):1–26, 1999.

[6] E. Larson and D. Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1):011006, 2010.

[7] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.

[8] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi. A no-reference perceptual blur metric. In *ICIP*, 2002.

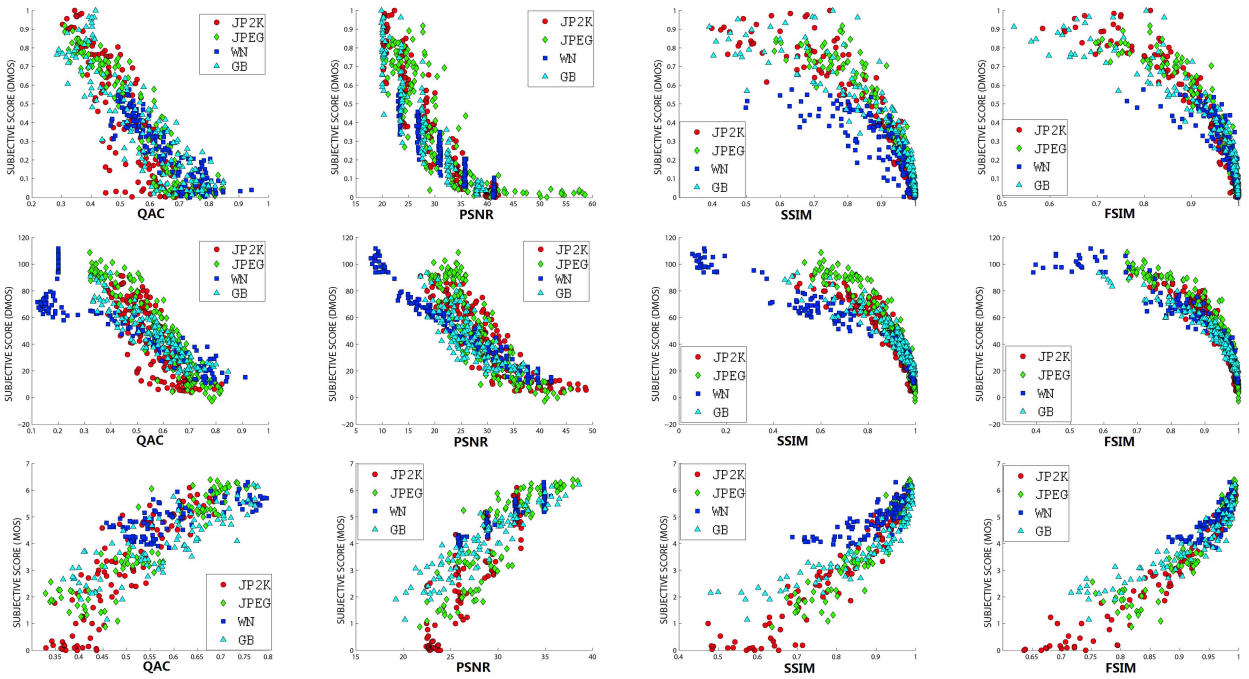

Figure 7. From left to right: the scatter plots of the prediction results of QAC, PSNR, SSIM and FSIM versus the subjective score in database of CSIQ (top row), LIVE (middle row) and TID2008 (bottom row). Each point stands for a distorted image in the database with distortion type indicated by color. A straight lined distribution of these points is better than a curved one.

[9] L. Meesters and J. Martens. A single-ended blockiness measure for JPEG-coded images. *Signal Processing*, 82(3):369–387, 2002.

[10] A. Mittal, A. Moorthy, and A. Bovik. Blind/referenceless image spatial quality evaluator. In *Asilomar Conference on Signals, Systems and Computers*, pages 723–727. IEEE, 2011.

[11] A. Mittal, G. Muralidhar, J. Ghosh, and A. Bovik. Blind image quality assessment without human training using latent quality factors. *IEEE SPL*, 19(2):75–78, 2012.

[12] A. Moorthy and A. Bovik. Visual importance pooling for image quality assessment. *IEEE Journal of Selected Topics in Signal Processing*, 3(2):193–201, 2009.

[13] A. Moorthy and A. Bovik. A two-step framework for constructing blind image quality indices. *IEEE SPL*, 17(5):513–516, 2010.

[14] A. Moorthy and A. Bovik. Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE TIP*, 20(12):3350–3364, 2011.

[15] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli, and F. Battisti. TID2008-a database for evaluation of full-reference visual quality assessment metrics. *Advances of Modern Radioelectronics*, 10(10):30–45, 2009.

[16] M. Saad, A. Bovik, and C. Charrier. A dct statistics-based blind image quality index. *IEEE SPL*, 17(6):583–586, 2010.

[17] M. Saad, A. Bovik, and C. Charrier. Model-based blind image quality assessment using natural DCT statistics. *IEEE TIP*, 21:3339–3352, 2011.

[18] H. Sheikh, A. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: JPEG2000. In *ICIP*, 2005.

[19] H. Sheikh, Z. Wang, L. Cormack, and A. Bovik. Live image quality assessment database release 2 (2005).

[20] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[21] H. Tang, N. Joshi, and A. Kapoor. Learning a blind measure of perceptual image quality. In *CVPR*, 2011.

[22] Z. Wang. Applications of objective image quality assessment methods. *IEEE Signal Processing Magazine*, 28(6):137 – 142, nov. 2011.

[23] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE TIP*, 13(4):600–612, 2004.

[24] Z. Wang and X. Shang. Spatial pooling strategies for perceptual image quality assessment. In *ICIP*, 2006.

[25] Z. Wang, H. Sheikh, and A. Bovik. No-reference perceptual quality assessment of JPEG compressed images. In *ICIP*, 2002.

[26] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE TPAMI*, 31(2):210–227, 2009.

[27] P. Ye, J. Kumar, L. Kang, and D. Doermann. Unsupervised feature learning framework for no-reference image quality assessment. In *CVPR*, 2012.

[28] R. Young. The gaussian derivative model for spatial vision: I. retinal mechanisms. *Spatial vision*, 2(4):273–293, 1987.

[29] L. Zhang, L. Zhang, X. Mou, and D. Zhang. Fsim: a feature similarity index for image quality assessment. *IEEE TIP*, 20(8):2378–2386, 2011.