MSFE: BLIND IMAGE QUALITY ASSESSMENT BASED ON MULTI-STAGE FEATURE ENCODING

Qiuping Jiang, Feng Shao, and Gangyi Jiang

Ningbo University
Faculty of Information Science and Engineering
Ningbo, China, 315211

ABSTRACT

Blind image quality assessment (BIQA) methods based on visual codebooks have received much attention due to its prominent generalization capacity across different image domains. Existing codebook-based BIQA methods depend on large-size codebooks and high-dimensional features, which are memory-consuming and have the risk of over-fitting. Thus, it is necessary to design quality metrics with much smaller codebooks. This paper presents a novel multi-stage feature encoding (MSFE)-based BIQA method which requires much lower dimensional features while preserving comparable or even better performance. To specify, MSFE is performed over multiple cascaded and much smaller sub-codebooks to generate more compact and discriminative features for quality prediction. The latter stage takes the encoding residuals in the former stage as input. We use K-SVD and sparse coding for codebook training and feature encoding in the framework, respectively. Finally, the generated sparse feature codes in all stages are combined and aggregated over the entire image for quality prediction via support vector regression (SVR). We evaluate the proposed method on several natural and screen content image databases. The experimental results confirm its superiority in terms of both validity and universality.

Index Terms **—** Blind image quality assessment (BIQA), multi-stage feature encoding (MSFE), encoding residual, support vector regression (SVR)

1. INTRODUCTION

In the past decades, the pursuit of automatic prediction of image quality has received great interests from researchers in diverse areas. The investigations on image quality assessment (IQA) have led to considerable progress in the development of various perceptual quality metrics. According to the availability of reference image, state-of-the-art quality metrics can be classified into categories: i.e., full-reference (FR) [1-3], reduced-reference (RR) [4-6], and no-reference (NR) [7-20]. While in the case where original image is not available, NR/Blind IQA (BIQA) metrics would be the only possible solution. The initial research efforts on BIQA mainly focus on evaluating images corrupted by specific distortions [7-9]. However, their universality is limited by the given distortion type. Later, the research of BIQA is further evolved to handle diverse distortion types, known as the general-purpose BIQA.

Studies on general-purpose BIQA generally require a set of image samples and the associated opinion scores to learn a quality prediction model [10-20]. Thus, the determinant factor is to extract highly versatile quality-aware features that are sensitive to a broad range of image distortion types while robust to image content variations. Existing general-purpose BIQA methods can be further classified into two categories according to the types of used features: NSS-based [10-16] and codebook-based features [17-20]. The most widely used quality-aware features are handcrafted NSS features, which are derived based on the assumption that pristine natural images potentially possess regular statistical properties while the presence of distortions will inevitably change such statistics and make images unnatural. Although effective in assessing natural images, the performance of NSS-based methods deteriorates significantly when tested on images containing natural scenes, texts, and graphics simultaneously.

Compared to the handcrafted NSS features, codebook-based features are automatically encoded based on a codebook that is learned from raw image patches. In this regard, codebook-based features require less domain knowledge and are potentially generalizable across different image domains. The most widely used codebook-based feature representation method is the BoW model, which has gained popularity in image classification, has also been applied for BIQA [17-20]. Considering the essence of BIQA is a typical regression problem, the application of BoW to BIQA is intuitive. They generally share the following framework: local feature extraction, codebook construction, feature encoding, spatial pooling, and quality regression. Representative methods that use codebook-based features include CBIQ [17], CORNIA [18], and HOSA [19].

To achieve a satisfying performance, these codebook-based methods invariably require a large-size codebook. Thus, the feature encoding and quality prediction processes are somewhat memory-consuming. More importantly, the high dimensional features may have the risk of over-fitting. Therefore, the existing NSS-based and codebook-based BIQA methods have their own pros and cons, respectively. To fix these issues and also inherit the merits of codebook-based BIQA framework, we propose a novel BIQA method based on multi-stage feature encoding (MSFE) which can produce much lower-dimensional feature representations for quality prediction while achieving comparable or even better performance. To specify, the devised MSFE scheme investigates to utilize hierarchical feature codes which are obtained by feature encoding over multiple cascaded and much smaller sub-codebooks to build an efficient and universal BIQA model.

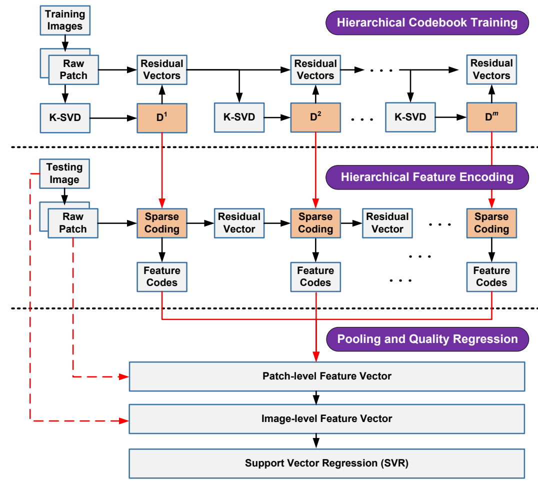

Fig. 1: Pipeline of the proposed MSFE-based BIQA method.

2. METHODOLOGY

Fig. 1 shows the pipeline of our proposed MSFE-based BIQA method, which contains three modules: hierarchical codebook training, hierarchical feature encoding, pooling and quality regression. Specifically, hierarchical codebook training aims at learning multiple cascaded sub-codebooks from a set of raw image patches. Then, in the hierarchical feature encoding module, MSFE takes normalized image patches of a testing image as input and encodes each patch over the pre-learned sub-codebooks stage by stage. This results in multiple-stage feature codes for each input normalized image patch. By concatenating the encoded feature codes in all stages, a final patch-level feature code is obtained. To fit the demand of BIQA, all the generated feature codes of local patches from a certain image are further aggregated by max pooling to get the image-level feature representation. Finally, SVR [21] is applied to learn a regression model to predict image quality by feeding the obtained image-level feature vector as input.

2.1. Local Feature Extraction

Instead of using elaborately designed handcrafted quality-aware features, only simple normalized raw image patches are extracted as local features in the method. This process is similar with CORNIA [18] and HOSA [19]. Specifically, given an image, we first convert it into grayscale from which a set of non-overlapping $B_s \times B_s$ image patches having rich structures and details are collected. The local feature vector is then extracted from each patch according to the following contrast normalized scheme:

$$\mathbf{y} = \frac{\mathbf{p} - \mu}{\sigma + 10}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^d$, μ, and σ respectively denote the contrast normalized feature vector, the mean value and standard deviation of patch $\mathbf{p} \in \mathbb{R}^d$ (d=$B_s \times B_s$). After the above contrast normalization process, we also perform zero-phase components analysis (ZCA) whitening [18, 19] to further remove the linear correlations between local features.

2.2. Hierarchical Codebook Training

Before encoding the extracted local features, a critical step is to construct a codebook or dictionary. Among various codebook learning algorithms, K-means clustering [22] is probably the most widely-used one for its simplicity. As a generalization of K-means, the K-SVD algorithm [23] aims to learn a codebook from a large set of samples adaptively to better capture the inherent structure of input data and achieve sparse signal representations. Consider a set of extracted local feature vectors $\mathbf{Y} = \{\mathbf{y}_k\}_{k=1}^{p} \in \mathbb{R}^{d \times p}$ from training images, the goal of K-SVD is to learn a codebook $\mathbf{D} = \{\mathbf{d}_i\}_{i=1}^{m} \in \mathbb{R}^{d \times m}$ (d<m) over which $\mathbf{Y}$ has a sparse representation $\mathbf{X} = \{\mathbf{x}_k\}_{k=1}^{p} \in \mathbb{R}^{m \times p}$, such that each $\mathbf{x}_k$ contains τ (τ<d) non-zero elements at most. Formally, this process can be written as the following optimization problem:

$$(\mathbf{D}, \mathbf{X}) = \arg\min_{\mathbf{D},\mathbf{X}} \left\{ \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \right\}, \; s.t. \; \forall k, \; \|\mathbf{x}_k\|_0 \le \tau, \tag{2}$$

where $\|.\|_F$ is the Frobenius norm and $\|.\|_0$ is the ℓ_0-norm that counts the number of non-zero elements in a vector. The K-SVD algorithm solves the above problem by performing two steps at each iteration: 1) sparse coding and 2) codebook update. For a detailed description of the K-SVD algorithm, please refer to [23].

With the use of K-SVD, an overcomplete codebook $\mathbf{D}$ and the associated sparse representation coefficient $\mathbf{X}$ of $\mathbf{Y}$ can be simultaneously obtained. That is, given a specific sparsity level τ, $\mathbf{Y}$ can be approximately reconstructed based on $\mathbf{X}$ and $\mathbf{D}$. The reconstruction residual $\mathbf{E}$ can be computed as follows:

$$\mathbf{E} = \mathbf{Y} - \mathbf{D}\mathbf{X}. \tag{3}$$

The proposed MSFE scheme implements K-SVD repeatedly to learn a hierarchical codebook set containing multiple sub-codebooks $\{\mathbf{D}^1, \mathbf{D}^2, \ldots, \mathbf{D}^n\}$. Meanwhile, the optimization of K-SVD in the (j+1)-th stage begins only when the j-th stage ends. It suggests that these multiple sub-codebooks are not learned independently; instead, the optimization of latter stage requires the residual data $\mathbf{E}$ in the former stage as its input. For better illustration, we summarize the pseudo-code of hierarchical codebook training in Algorithm 1. It should be emphasized that the hierarchical codebook set is created via an offline process and is afterwards used routinely, allowing for the adoption of this more sophisticated codebook construction technique with no additional cost on the computational speed during quality prediction.

Algorithm 1: Hierarchical codebook training.

Input: The training feature set $\mathbf{Y}$; the codebook size m; the desired sparsity level τ; the number of stage n.

Output: The learned hierarchical codebook set $\mathbf{D} = \{\mathbf{D}^1, \mathbf{D}^2, \ldots, \mathbf{D}^n\}$.

initialize $\mathbf{Y}^1 = \mathbf{Y}$;
for each $j \in [1, n]$ **do**
  do K-SVD on $\mathbf{Y}^j$ according to Eq. (2);
  output the sub-codebook $\mathbf{D}^j$ and sparse representation $\mathbf{X}^j$;
  compute the residual data $\mathbf{E}^j$ according to Eq. (3);
  assign the residual data to the next input: $\mathbf{Y}^{j+1} = \mathbf{E}^j$;
end for
concatenate all the sub-codebooks: $\mathbf{D} = \{\mathbf{D}^1, \mathbf{D}^2, \ldots, \mathbf{D}^n\}$.

2.3. Hierarchical Feature Encoding

For feature encoding, the classical BoW model adopts simple hard assignment (HA) scheme which activates only the nearest codeword to the descriptor, assigning zero weight to all others. However, methods like semi-soft assignment (SSA) [24] and sparse coding (SC) [25], which represent a descriptor by a few number of codewords with different weights, were shown to perform better than HA in many computer vision applications [26].

In our devised MSFE scheme, on the one hand we compute the sparse coefficients of an input local feature vector over the pre-learned hierarchical sub-codebooks using SC stage by stage. On the other hand, we also compute the reconstruction residual vector in the current stage and pass this derived residual vector to the next stage. Given the pre-learned n-stage codebook set $\{\mathbf{D}^1, \mathbf{D}^2, \ldots, \mathbf{D}^n\}$ with each component $\mathbf{D}^j$ ($1 \le j \le n$) representing the sub-codebook in the j-th stage with m codewords $\{\mathbf{d}_{j,1}, \mathbf{d}_{j,2}, \ldots, \mathbf{d}_{j,m}\}$, for an input $\mathbf{y}^j \in \mathbb{R}^d$ in the j-th stage, the corresponding sparse coefficient vector $\mathbf{x}^j \in \mathbb{R}^m$ and residual vector $\mathbf{r}^j \in \mathbb{R}^d$ are calculated as follows:

$$\mathbf{x}^j = \arg\min_{\mathbf{x}^j} \left\{ \left\| \mathbf{y}^j - \mathbf{D}^j \mathbf{x}^j \right\|_F^2 \right\}, \ s.t. \ \left\| \mathbf{x}^j \right\|_0 \le \tau, \tag{4}$$

$$\mathbf{r}^j = \mathbf{y}^j - \mathbf{D}^j \cdot \mathbf{x}^j, \tag{5}$$

Once the sparse coefficient vector $\mathbf{x}^j$ and the residual vector $\mathbf{r}^j$ are estimated at the current stage, $\mathbf{r}^j$ is passed to the next stage, i.e., $\mathbf{y}^{j+1} = \mathbf{r}^j$. Then, the above steps in Eq. (4) and (5) are iteratively performed until the final stage n is reached. Finally, the sparse coefficient vectors $\{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n\}$ from all stages are concatenated, yielding the final feature code of $\mathbf{y}$, i.e., $\mathbf{y} \in \mathbb{R}^d \rightarrow \mathbf{x} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n]^T \in \mathbb{R}^{(m \times n) \times 1}$, where n is the number of stages and m is the number of codewords in each sub-codebook. The pseudo-code of hierarchical feature encoding is summarized in Algorithm 2.

Algorithm 2: Hierarchical feature encoding.

Input: The extracted local feature $\mathbf{y}$ to be encoded; the hierarchical codebook set $\{\mathbf{D}^1, \mathbf{D}^2, \ldots, \mathbf{D}^n\}$.

Output: The produced sparse feature codes $\mathbf{x} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n]$ of $\mathbf{y}$.

initialize $\mathbf{y}^1 = \mathbf{y}$;
for each $j \in [1, n]$ **do**
  do SC on $\mathbf{y}^j$ over $\mathbf{D}^j$ according to Eq. (4) to obtain the coefficient vector $\mathbf{x}^j$;
  compute the residual vector $\mathbf{r}^j$ according to Eq. (5);
  assign the residual vector to the next stage as input: $\mathbf{y}^{j+1} = \mathbf{r}^j$;
end for
concatenate all the sparse coefficient vectors: $\mathbf{x} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n]$.

2.4. Spatial Pooling

To meet the demand of BIQA, the feature codes of all local patches from a certain image should be aggregated to get a final image-level feature vector convenient for quality regression. For a distorted image I to be evaluated, we first extract L salient patches $\{\mathbf{sp}_1, \mathbf{sp}_2, \ldots, \mathbf{sp}_L\}$ from it (the distortions appeared in salient regions actually play a dominant role in the overall image quality) and compute the corresponding local feature vectors $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_L\}$ according to Eq. (1). Then, following the hierarchical feature encoding process described in Algorithm 2, the associated patch-level feature code set $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L\}$ of this image is also obtained. For each $\mathbf{x}_l = [x_l^1, x_l^2, \ldots, x_l^{m \times n}]^T \in \mathbb{R}^{(m \times n) \times 1}$ in $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L\}$, performing max pooling on $\mathbf{x}_l$ can be expressed as:

$$x_l^t = \begin{cases} 1, & \text{if } x_l^t = \max\left(x_1^t, x_2^t, \ldots, x_L^t\right) \\ 0, & \text{otherwise} \end{cases}, \ 1 \le t \le (m \times n) \tag{6}$$

Finally, a ($m \times n$)-dim image-level feature vector $\mathbf{f}$ is obtained by summing up all columns, such that:

$$\mathbf{f} = \mathbf{x}_1 + \mathbf{x}_2 + \cdots + \mathbf{x}_L. \tag{7}$$

2.5. Quality Regression

After obtaining the image-level feature vector, we need to seek a suitable mapping function that is learnt from the feature space to subjective MOS values using a regression module, and then apply it to predict objective quality scores. Of course, any regression methods can be used here. To create a fair comparison with state-of-the-art BIQA methods, we use SVR [21] following the choices in CBIQ [17], CORNIA [18], and HOSA [19]. Here, the LIBSVM package [27] is adopted to implement the SVR.

Table I: Performance results of competing BIQA algorithms on LIVE, LIVEMD, and SIQAD databases.

Database	Criteria	Handcrafted feature-based methods					Codebook-based methods				
		DIIVINE	BLIINDS-II	BRISQUE	GM-LOG	NFREM	CBIQ	CORNIA-10K	CORNIA-1K	HOSA	Proposed
LIVE	SROCC	0.9162	0.9302	0.9409	**0.9503**	0.9376	0.9119	**0.9417**	0.9172	**0.9504**	0.9375
	PLCC	0.9172	0.9357	**0.9450**	**0.9539**	0.9421	0.9278	0.9434	0.9039	**0.9527**	0.9386
	RMSE	10.8103	9.6189	**8.9048**	**8.1723**	9.1012	10.2683	9.0204	11.9541	**8.2858**	9.6064
LIVEMD	SROCC	0.8738	0.8872	0.8972	0.8237	0.8989	0.8876	**0.9007**	0.8761	**0.9019**	**0.9105**
	PLCC	0.8936	0.9028	**0.9207**	0.8632	0.9190	0.8992	0.9150	0.8935	**0.9262**	**0.9266**
	RMSE	8.3843	8.1330	**7.3168**	9.4198	7.4132	8.1455	7.6737	8.3974	**6.9739**	**6.9682**
SIQAD	SROCC	0.7279	0.7561	0.7715	0.7989	0.7983	0.8314	**0.8352**	0.8028	**0.8484**	**0.8519**
	PLCC	0.7768	0.7982	0.8210	0.8330	0.8259	0.8492	**0.8533**	0.8307	**0.8636**	**0.8702**
	RMSE	8.6903	8.3688	7.9383	7.7005	7.8615	7.2571	**7.1989**	7.4723	**6.9594**	**6.7437**

3. EXPERIMENTAL RESULTS

Performance evaluation is conducted on three databases: LIVE [28], LIVEMD [29], and SIQAD [30]. For SIQAD database, we mainly consider four types of distortions which appear in the LIVE database, i.e., JP2K, JPEG, WN and GB. The proposed method is compared with three codebook-based methods, i.e., CBIQ [17], CORNIA-10K [18] and HOSA [19], and several NSS-based methods [12-16]. Three commonly used criteria, i.e., Spearman's rank order correlation coefficient (SROCC), Pearson's linear correlation coefficient (PLCC), and root mean squared error (RMSE), are employed to evaluate the performance.

There are several parameters that need to be decided in the proposed method: 1) B_s: patch size; 2) p: number of patches used for codebook construction; 3) τ: sparsity level; 4) m: sub-codebook size; 5) n: number of stages. In our experiments, the patch size B_s is set to 7, which is the same as used in CORNIA and HOSA. A number of 1K local patches having the highest standard deviation values are selected from each image in the training set, such that a total number of 600K image patches can be collected from the CSIQ database [31] for codebook construction (four shared distortion types are considered). The sparsity level τ is set to 5, the sub-codebook size is set to 500, and the number of stages is set to 4. We experimentally found that, by using these parameters, stable performance can be obtained across all databases. Thus, the final feature dimension is 2K in total which is much smaller than 10K used in CBIQ, 20K used in CORNIA and 14.7K used in HOSA.

For performance evaluation, following the evaluation methodology in related works [17-19], we first divide each database into two subsets: training and testing subsets. To ensure non-content overlapping between training and testing subsets, distorted images associated with 80% of the reference images in each database are selected for training, and the rest images are used for testing. Such random training-testing split is repeated 1000 times and the median performance is reported in Table I. We highlight the best three methods for each criterion in boldface.

From the Table, we have the following observations. First, our proposed method always has stable performance for all databases. This demonstrates that the proposed MSFE scheme has good ability to learn effective yet robust features in characterizing image quality with diverse visual contents and distortions. Second, the proposed method performs the best on LIVEMD and SIQAD databases while slightly worse on LIVE. It is encouraging because we only need to extract a 2K-dim feature vector while previous codebook-based methods extract 10K-dim feature vector at least. Although CORNIA-1K also produces 2K-dim feature, it performs much worse than our method across all databases. We believe that this phenomenon is mainly attributed to the encoding error with a single small-size codebook which could interfere with the quality evaluation. Instead, the proposed MSFE encodes each local feature descriptor with respect to multiple cascaded and small sub-codebooks (the encoding errors in the former stage are passed to the next stage), yielding more compact and discriminative quality-aware features. Third, compared to the competing NSS-based methods, the proposed method exhibits comparable performance on LIVE database. While for LIVEMD and SIQAD databases, especially the SIQAD, state-of-the-art NSS-based methods are inferior to the most codebook-based methods. This indicates that traditional NSS features cannot well reflect the quality information contained in distorted screen content images which contain not only natural scenes but also texts and graphics. However, the codebook-based methods are able to provide unique benefits in capturing the specific information in screen content images.

4. CONCLUSION

This paper has presented a novel blind image quality assessment (BIQA) method based on multi-stage feature encoding (MSFE) which requires much lower-dimensional features for quality prediction than existing codebook-based BIQA methods. Experiments on several databases confirm the superiority of the proposed MSFE scheme for BIQA. To specify, compared with NSS-based methods, our method is more suitable for use across different image domains; while compared with existing codebook-based methods, our method is much more memory-saving and can achieve comparable even better performance.

5. REFERENCES

[1] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," IEEE Transactions on Image Processing, 20(5): 1185-1198, 2011.

[2] L. Zhang, L. Zhang, X. Mou, et al., "FSIM: A feature similarity index for image quality assessment," IEEE Transactions on Image Processing, 20(8): 2378-2386, 2011.

[3] L. Zhang, Y. Shen, and H. Li, "VSI: A visual saliency induced index for perceptual image quality assessment," IEEE Transactions on Image Processing, 23(10): 4270-4281, 2014.

[4] A. Rehman and Z. Wang, "Reduced-reference image quality assessment by structural similarity estimation," IEEE Transactions on Image Processing, 21(8): 3378-3389, 2012.

[5] L. Ma, S. Li, F. Zhang, et al., "Reduced-reference image quality assessment using reorganized DCT-based image representation," IEEE Transactions on Multimedia, 13(4): 824-829, 2011.

[6] J. Wu, W. Lin, G. Shi, et al., "Orientation selectivity based visual pattern for reduced-reference image quality assessment," Information Sciences, 351: 18-29, 2016.

[7] R. Ferzli and L. Karam, "A no-reference objective image sharpness metric based on the notion of just noticeable blur (JNB)," IEEE Transactions on Image Processing, 18(4): 717-728, 2009.

[8] Z. Wang, A. Bovik, and B. Evans, "Blind measurement of blocking artifacts in images," in Proceedings of IEEE International Conference on Image Processing (ICIP), 2000, 981-984.

[9] H. Liu, N. Klomp, and I. Heynderickx, "A no-reference metric for perceived ringing artifacts in images," IEEE Transactions on Circuits and Systems for Video Technology, 20(4): 529-539, 2010.

[10] H. Sheikh, A. Bovik, and L. Cormack, "No-reference quality assessment using natural scene statistics: JPEG2000," in Proceedings of IEEE International Conference on Image Processing (ICIP), 2005.

[11] A. Moorthy and A. Bovic, "A two-step framework for constructing blind image quality indices," IEEE Signal Processing Letters, 17(5): 513-516, 2010.

[12] A. K. Moorthy and A. C. Bovik, "Blind image quality assessment: From natural scene statistics to perceptual quality," IEEE Transactions Image Processing, 20(12): 3350-3364, 2011.

[13] M. A. Saad, A. C. Bovik, and C. Charrier, "Blind image quality assessment: A natural scene statistics approach in the DCT domain," IEEE Transactions on Image Processing, 21(8): 3339-3352, 2012.

[14] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," IEEE Transactions on Image Processing, 21(12): 4695-4708, 2012.

[15] W. Xue, X. Mou, L. Zhang, A. C. Bovik, and X. Feng, "Blind image quality assessment using joint statistics of gradient magnitude and Laplacian features," IEEE Transactions on Image Processing, 23(11): 4850-4862, 2014.

[16] K. Gu, G. Zhai, X. Yang, and W. Zhang, "Using free energy principle for blind image quality assessment," IEEE Transactions on Multimedia, 17(1): 50-63, 2015.

[17] P. Ye and D. Doermann, "No-reference image quality assessment using visual codebooks," IEEE Transactions on Image Processing, 21(7): 3129-3138, 2011.

[18] P. Ye, J. Kumar, L. Kang, and D. Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," in the IEEE International Conference on Computer Vision and Pattern Recognition, 2012, pp.1089-1105.

[19] J. Xu, P. Ye, Q. Li, H. Du, Y. Liu, and D. Doermann, "Blind image quality assessment based on high order statistics aggregation," IEEE Transactions on Image Processing, 25(9): 4444-4457, 2016.

[20] L. Zhang, Z. Gu, X. Liu, H. Li, and J. Lu, "Training quality-aware filters for no-reference image quality assessment," IEEE Multimedia Magazine, 21(4):67-75, 2014.

[21] B. Schökopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," Neural Computation, 12(5): 1207-1245, 2000.

[22] S. Lloyd, "Least squares quantization in PCM," IEEE Transactions on Information Theory, 28(2): 129-137, 1982.

[23] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," IEEE Transactions on Image Processing, 54(1): 4311-4322, 2006.

[24] L. Liu, L. Wang, and X. Liu, "In defense of soft-assignment coding," in the IEEE International Conference on Computer Vision, 2011.

[25] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in Proceedings of Advances in Neural Information Processing Systems, 2007, pp. 801–808.

[26] S. Avila, N. Thome, M. Cord, E. Valle, and A. Araújo, "Pooling in image representation: The visual codeword point of view," Computer Vision and Image Understanding, 117(5): 453-465, 2013.

[27] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," ACM Transactions on Intelligent Systems and Technology, 2(3): 2011, [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[28] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," IEEE Transactions on Image Processing, 15(11): 3440-3451, 2006.

[29] D. Jayaraman, A. Mittal, A. K. Moorthy, and A. C. Bovik, "Objective quality assessment of multiply distorted images," in Proceedings of IEEE Conf. Rec. 46th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR), 2012, pp. 1693-1697.

[30] H. Yang, Y. Fang, and W. Lin, "Perceptual quality assessment of screen content images," IEEE Transactions on Image Processing, 24(11): 4408-4421, 2015.

[31] E. C. Larson and D. M. Chandler, "Most apparent distortion: Full-reference image quality assessment and the role of strategy," Journal of Electronic Imaging, 19(1), article no. 011006, 2010.