On the Performance of Visual Semantics for Improving Texture-based Blind Image Quality Assessment

Pedro Garcia Freitas* and Mylène C.Q. Farias‡
*Department of Computer Science, ‡Department of Electrical Engineering,
University of Brasília, Brazil
Email: *sawp@sawp.com.br, ‡mylene@ieee.org

***Abstract*—Blind image quality assessment (BIQA) methods aim to estimate the quality of a given test image without referring to the corresponding reference (original) image. Most BIQA methods use visual sensitivity models, which take into consideration intrinsic image characteristics (e.g. contrast, luminance, and texture) to identify degradations and estimate quality. For example, texture-based BIQA methods are based on the assumption that visual impairments (degradations) alter the characteristics of the image textures and, therefore, their statistics. Although these methods have been are known to provide an acceptable performance, they do not take into account the semantic information of the image. In this paper, we propose a BIQA method that estimates quality using texture characteristics and semantic information. The texture characteristics are obtained using the Opponent Color Local Binary Pattern (OCL) operator. The semantic information is obtained by estimating the probability distribution of the scene characteristics. A random forest regression algorithm is used to map semantic and texture-based features into a quality score. Results obtained testing the proposed BIQA method on several public databases show the method has a good accuracy on quality prediction.**

I. Introduction

Image quality assessment (IQA) is a research area that has achieved a great importance in the last years, mostly due to the exponential growth of the popularity of digital visual information (images). Given this high volume of visual information, the task of accurately assessing the quality of an image has become crucial for several multimedia applications. More specifically, IQA methods are used to estimate the performance of compression algorithms [1], multimedia transmission [2], [3], display technologies, image enhancement and restoration algorithms [4].

Over the past decades, a lot of progress has been made in the area of image quality, with a large number of IQA methods being proposed. IQA methods can be classified into three types, according to the amount of information required to perform the assessment task. Full-reference (FR) methods [5] require the original image and are, usually, more precise. Reduced-reference (RR) methods require only part information (e.g. features) about the original image [6], [7]. Because needing even partial information of the reference image can be a hindrance for several multimedia applications, frequently, the most adequate solution is to use blind image quality assessment (BIQA) methods. BIQA methods [8], [9] blindly estimate the quality of a test image without requiring any information about its reference.

Many BIQA methods have been proposed [8]–[11]. Among the available approaches, methods based on texture analysis in combination with machine learning techniques have been very successful. As an example, we can cite the work of Peng Ye and Doermann [12], which uses local Gabor-filter features to build a visual codebook that is used to estimate quality. Recently, several BIQA methods based on a texture descriptor known as the Local Binary Pattern (LBP) operator [13] have been proposed. State-of-the-art LBP-based BIQA methods include the efforts of Freitas *et al.* [9], [14], Rezaie *et al.* [15], Li *et al.* [11], Zhang *et al.* [10], and Wu *et al.* [16].

Although the aforementioned methods achieve an acceptable prediction accuracy, some issues remain open. As stated by Chandler [17], so far, IQA developments focus on improving the prediction accuracy for popular distortions, such as JPEG, blurring, or noise. There are few methods that perform efficiently for multiple distortions. Therefore, there are very few general purpose BIQA methods. In this paper, we investigate if semantics can improve the lack of generality of BIQA methods.

Most IQA methods assume that the perceived quality depends exclusively on the sensitivity to impairments. In this paper, we study how image semantics can affect quality. Our work is inspired by the subjective study performed by Siahaan *et al.* [18], which demonstrated that visual quality is influenced by the semantic content. Moreover, Farias & Akamine [19] studied how to incorporate visual saliency into IQA methods, obtaining interesting results. Since saliency is an aspect of image semantics [20], we believe that image semantics can indeed be used to improve the accuracy performance of IQA methods.

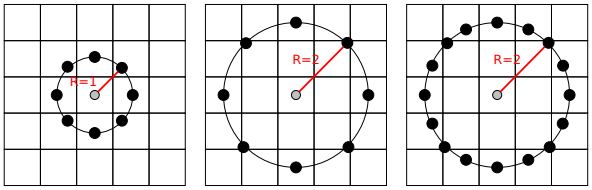
Differently from Siahaan *et al.* [18], who performed an investigation using subjective experiments, we aim to incorporate semantic features into the design of a BIQA method. More specifically, we use a pre-trained deep convolutional neural network to generate semantic categories of an image. These

categories are, then, combined with texture features to blindly estimate the image quality.

The rest of this paper is organized as follows. In Section II, we review the opponent color local binary pattern (OCL) operator and in Section III we discuss the semantic analysis. In Section IV, we describe how to apply the OCL and the semantic features into a BIQA method. Sections V and VI present the experimental setup and the results, respectively. Finally, Section VII presents the conclusions.

II. The Opponent Color Local Binary Pattern

Local Binary Pattern (LBP) is arguably one of the most powerful texture descriptors currently available. It was first proposed by Ojala *et al.* [21] as a particular case of the Texture Spectrum Model [22]. Ojala *et al.* [23], [24] formalized the LBP descriptor and it has since been proven to be an effective feature extractor for texture based problems. Because of its effectiveness, several extended versions of the LBP operator have been proposed [25], making it possible to adapt this operator to specific applications.



(a) $R = 1, P = 8$ (b) $R = 2, P = 8$ (c) $R = 2, P = 16$

Fig. 1. Circularly symmetric P neighbors extracted from a distance R.

The traditional LBP operator [24] takes the form:

$$LBP_{R,P}(I_c) = \sum_{p=0}^{P-1} S(I_p - I_c)2^p, \tag{1}$$

where

$$S(t) = \begin{cases} 1 & t \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

In Eq. 1, $I_c = I(x, y)$ is an arbitrary central pixel at the position (x, y) (gray dots in Fig. 2) and $I_p = I(x_p, y_p)$ is a neighboring pixel surrounding I_c (back dots in Fig. 2), where:

$$x_p = x + R\cos\left(2\pi\frac{p}{P}\right) \text{ and } \quad y_p = y - R\sin\left(2\pi\frac{p}{P}\right).$$

In this case, $p = \{1, 2, \cdots, P\}$ is the number of neighboring pixels sampled from a distance R (radius) from I_c to I_p. Fig. 1 illustrates examples of symmetric samplings with different numbers of neighboring points (P) and radius (R) values.

Fig. 2 illustrates the steps for applying the LBP operator on a single pixel ($I_c = 8$) located in the center of a 3×3 image block, as shown in the bottom-left of this figure. The numbers in the yellow squares of the block represent the order in which the operator is computed (counter-clockwise direction starting from 0). In this figure, we use an unitary neighborhood radius ($R = 1$) and eight neighboring pixels ($P = 8$). After

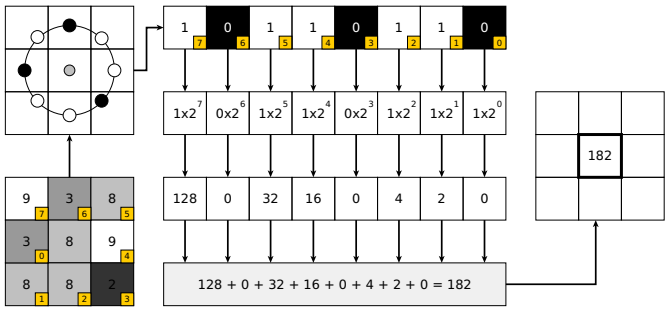


Fig. 2. Calculation of LBP labels.

calculating $S(t)$ (Eq. 2) for each neighboring pixel I_p, we obtain a binary output for each I_p ($0 \leq p \leq 7$), as illustrated in the block in the upper-left position of Fig. 2. In this block, black circles correspond to '0' and white circles to '1'. These binary outputs are stored in a binary format, according to their position (yellow squares). Then, the resulting binary number is converted to the decimal format. This decimal number is the output generated by the LBP operator for I_c.

Although the LBP descriptor is efficient for describing grayscale textures, it is not sensitive to some types of impairments, such as contrast distortions or chromatic aberrations. As discussed by Maenpaa *et al.* [26], color and texture have complementary roles. When texture descriptors on luminance domain (e.g. LBP) obtain good results, color descriptors can also obtain good results. However, when color descriptors fail, luminance texture descriptors can produce a good performance. Therefore, operators that combine both texture and color information are more effective in predicting a wider range of impairments.

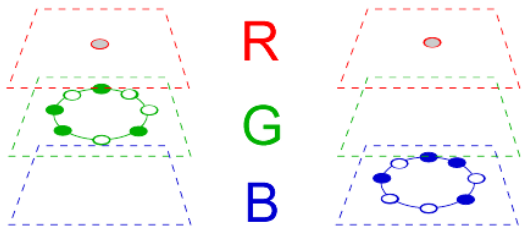
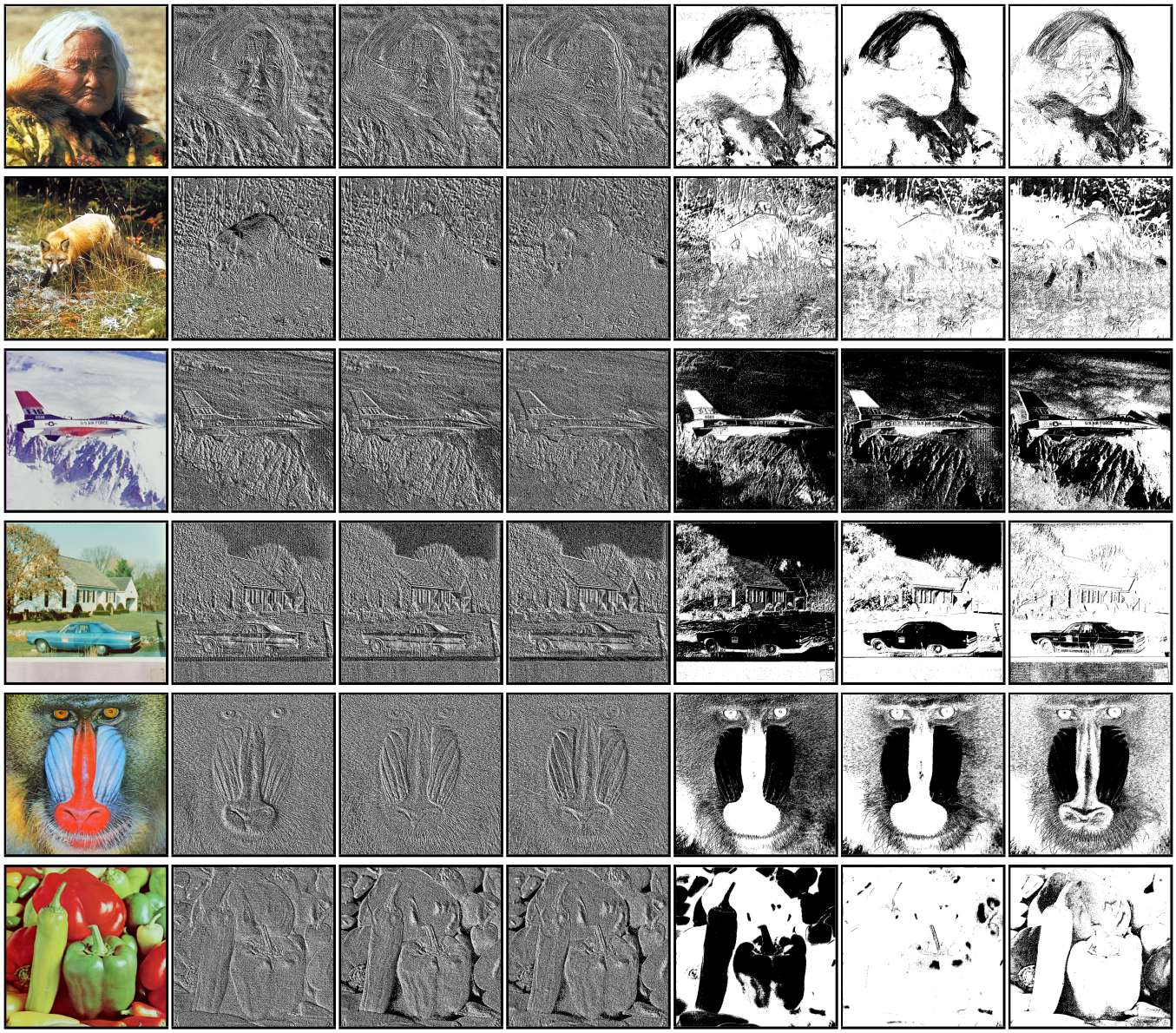


Fig. 3. Sampling scheme for the OCL_{RG} and OCL_{RB} descriptors.

To combine both texture and color information into a joint descriptor, Maenpaa [27] proposed to use the Opponent Color Local Binary Pattern (OCL) operator. This operator improves the operator proposed by Jain & Healey [28] by substituting the Gabor filter with a variant of the LBP operator, what decreases the computational cost of the method.

The OCL operator has two approaches. In the first, the LBP operator is applied, individually, on each color channel, instead of being applyied only on a single luminance channel. This approach is called 'intra-channel' because the central pixel and the corresponding sampled neighboring points belong to the same color channels.



(a) Original (b) LBP_R (c) LBP_G (d) LBP_B (e) OCL_{RG} (f) OCL_{RB} (g) OCL_{GB}

Fig. 4. Original images and their output channels, computed using the OCL operator.

In the second approach, called 'inter-channel', the central pixel belongs to a color channel and its corresponding sampled neighboring points belong to another color channel. More specifically, for an OCL_{MN} operator, the central pixel is positioned in the channel M, while the neighborhood is sampled in the channel N. For a three-channel color space, such as RGB, there are six possible combinations of channels: OCL_{RG}, OCL_{RG}, OCL_{RB}, OCL_{RB}, OCL_{GB}, and OCL_{GB}.

Fig. 3 depicts the sampling approach of OCL when the central pixel is sampled in R channel. From this figure, we can notice that two combinations are possible: OCL_{RG} (left) and OCL_{RB} (right). In this OCL_{RG}, the gray circle in the red channel is the central point, while the green circles in the green channel correspond to '0' sampling points and the white circles correspond to '1' sampling points, respectively. Similarly, in the OCL_{RB} the blue circles correspond to '0' sampling points and the white circles correspond to '1' sampling points, respectively.

After computing the OCL operator for all pixels, a total of six texture channels are generated. As depicted in Fig. 4, three LBP intra-channels (LBP_R, LBP_G, and LBP_B) and three LBP inter-channels (OCL_{RG}, OCL_{RB}, and OCL_{GB}) are generated. Although all possible combinations of the opposite color channels allows six distinct channels, we observed that the symmetric opposing pairs are very redundant (e.g. OCL_{RG} is equivalent to OCL_{GR}). Due to this redundancy, only the three more descriptive inter-channels are used.

III. Visual Semantic Models

In the context of this paper, visual semantic information refers to a description of a scene. More specifically, we use an image as input to a pre-trained machine-learning algorithm that automatically interprets the image content. This algorithm associates the elements in the scene (e.g. objects, people, animals, etc) to semantic categories, which are then associated with a class (i.e., a unique numeric label associated with the textual description of the semantic content). At the end, the algorithm outputs a multi-dimensional vector, with the j-th vector element representing the probability of the content being described by the j-th class.

In the literature, there are several algorithms that perform the aforementioned task. Among these algorithms, those based on deep convolutional neural networks (CNN) represent a big progress for the area of image classification and visual semantic extraction [29]. This progress has been driven by the ImageNet challange [30]. The goal of this challenge is to train a model that can classify an input image into 1,000 separate object categories.

Since the ImageNet was started, many methods have been proposed and the leader-board for this challenge has been dominated by CNN and deep learning techniques since 2012. Among the CNN-based algorithms, it is worth mentioning AlexNet [31], VGG [32], GoogleNet [33], and ResNet [34] methods. All these methods use deep-learning approaches, but differ in terms of the architectures of the neural networks, such as the number of layers, pooling, and fine-tuning.

Table I depicts the visual semantic elements generated using ResNet. The output is a 1,000-dimensional vector, with each element representing a semantic category (class). The images in Table I are the inputs, while the barplots indicate the probability distribution of each class describing the input image. For each image, we list the three more likely class (descriptions), as predicted by the ResNet.

IV. Proposed Method

Fig. 5 depicts a block-diagram of the proposed BIQA method. First, we compute the OCL channels, as described in the Section II. Then, we extract the statistical information of each channel, by computing the histograms of the OCL channels. These histograms are computed as follows:

$$\mathcal{H}_\varphi = H(C_\varphi) = \{h_\varphi(l_1), h_\varphi(l_2), \cdots\}, \tag{3}$$

where:

$$h_\varphi(l_i) = \sum_{x,y} \delta(C_\varphi(x,y), l_i), \tag{4}$$

and

$$\delta(v,u) = \begin{cases} 1 & v = u, \\ 0 & v \neq u \end{cases}. \tag{5}$$

In the above equations, C_φ depicts a OCL channel, (x,y) is the position of a random pixel of C_φ, and l_i is the i-th label of C_φ. The concatenation of all histograms (i.e., $\mathcal{H}_{\text{LBP}_R}$, $\mathcal{H}_{\text{LBP}_G}$, $\mathcal{H}_{\text{LBP}_B}$, $\mathcal{H}_{\text{OCL}_{RG}}$, $\mathcal{H}_{\text{OCL}_{RB}}$, and $\mathcal{H}_{\text{OCL}_{GB}}$) generates a single vector of texture features.

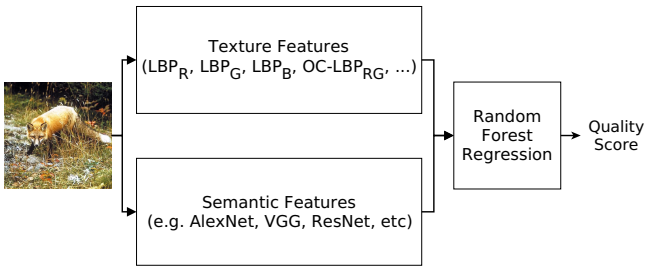


Fig. 5. Block diagram of the proposed BIQA method.

The semantic features are computed as described in Section III (i.e., the 1,000-dimensional vector containing the probability distribution of each class). After computing both the OCL histograms and the semantic features, the final feature vector is constructed by concatenating these two sets of features. Finally, the prediction stage uses the random forest regression (RFR) [35] technique to obtain a quality estimate. It is worth pointing out that, although the support vector regression (SVR) is widely used in the field of IQA, recent results indicate that RFR provides competitive performances [36]–[38].

V. Experimental Setup

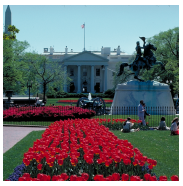
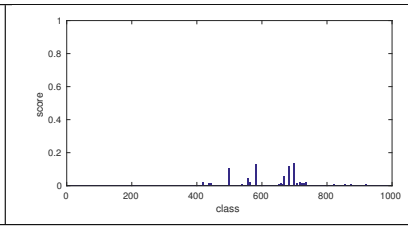
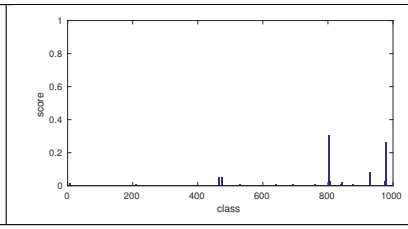
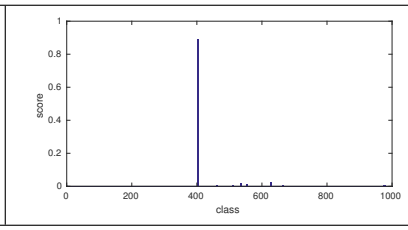
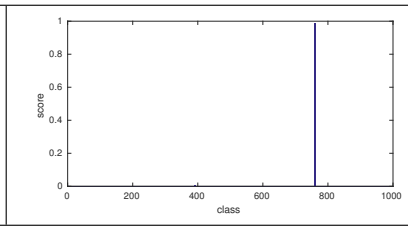
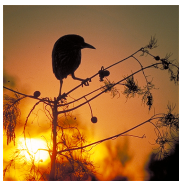
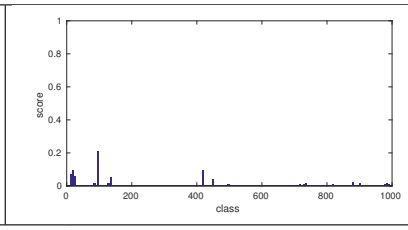
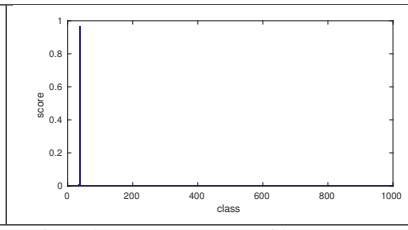
Results were obtained using an Intel® Core™ i7-4700MQ processor at 2.40GHz. To evaluate the prediction performance of the proposed method, the Spearman's Rank Ordered Correlation (SROCC) between the subjective scores and the predicted scores is used. The proposed method is compared with two FR-IQA methods (PSNR and SSIM [39]) and five publicly available state-of-the-art BIQA methods are considered for comparison (BRISQUE [40], CORNIA [41], CQA [42], SSEQ [43], and LTP [14]).

Since all tested BIQA methods are machine learning algorithms, we use the same procedure for training and testing. To avoid overfit, the databases are divided into content-independent training and testing subsets. In other words, image content in the training subset was not used in the testing subset, and vice-versa. From the complete database, 80% of the images are used for training and 20% are used for testing. This procedure is repeated 1,000 times, with the training and testing subsets being randomly selected at each time. For the SVR-based BIQA methods, we use the LibSVR implementation, which can be accessed using a Python interface provided by Sklearn library [44]. For each method, the optimal SVR metaparameters (C, γ, ν, etc) are automatically found using a grid search method. To generate the RFR prediction model of the proposed method, we also used Sklearn. No optimized search methods are used for the RFR version of the proposed method.

Four models are used to generate the semantic features, including AlexNet [31], VGG [32], GoogLeNet [33], and ResNet [34]. These models are provided with the MatConvNet library [45]. MatConvNet provides several pre-trained models.

TABLE I

EXAMPLE OF VISUAL SEMANTIC USING RESNET TO CHARACTERIZE FEATURES. ON EACH CELL, THE IMAGES ARE THE INPUTS USED ON CLASSIFICATION. THE BARPLOT INDICATE THE DISTRIBUTION OF PROBABILITIES. TEXTS DESCRIBES THE THREE MOST LIKELY PREDICTIONS.

1600	  (1) palace (class: 699, score: 0.136); (2) greenhouse, nursery, glasshouse (class: 581, score: 0.132); (3) obelisk (class: 683, score: 0.120);	Child   (1) snorkel (class: 802, score: 0.305); (2) sandbar, sand bar (class: 978, score: 0.262); (3) ice lolly, lolly, lollipop, popsicle (class: 930, score: 0.083)
City	  (1) aircraft carrier, attack aircraft carrier (class: 404, score: 0.888); (2) liner, ocean liner (class: 629, score: 0.026); (3) dock, dockage, docking facility (class: 537, score: 0.02);	Fisher   (1) reel (class: 759, score: 0.989); (2) coho, coho salmon, blue jack (class: 392, score: 0.007); (3) pole (class: 734, score: 0.001);
Sparrow	  (1) hummingbird (class: 95, score: 0.210); (2) balloon (class: 418, score 0.095); (3) magpie (class: 19, score 0.091);	Turtle   (1) box turtle, box tortoise (class: 38, score: 0.965); (2) terrapin (class: 37, score: 0.021); (3) mud turtle (class: 36, score: 0.012);

Among these methods, we used six VGG variants and four ResNet variants.

The three following databases are used to test the method:

- LIVE2 [46] has 982 test images, including 29 originals. This database includes 5 categories of distortions: JPEG, JPEG 2000 (JPEG2k), white noise (WN), Gaussian blur (GB), fast fading (FF).
- CSIQ [47] has a total fo 866 test images, consisting of 30 originals and 6 different categories of distortions. The distortions include JPEG, JPEG 2000 (JPEG2k), JPEG, white noise (WN), Gaussian blur (GB), fast fading (FF), global contrast decrements (CD), and additive Gaussian pink noise (PN).
- TID2013 [48] contains 25 reference images with the following distortions: Additive Gaussian noise (AGN), Additive noise in color components (AGC), Spatially correlated noise (SCN), Masked noise (MN), High frequency noise (HFN), Impulse noise (IN), Quantization noise (QN), Gaussian blur (GB), Image denoising (ID), JPEG, JPEG2k, JPEG transmission errors (JPEGTE), JPEG2k transmission errors (JPEG2kTE), Non eccentricity pattern noise (NEPN), Local block-wise distortions (LBD), Intensity shift (IS), Contrast change (CC), Change of color saturation (CCS), Multiplicative Gaussian noise (MGN), Comfort noise (CN), Lossy compression (LC), Image color quantization with dither (ICQ), Chromatic aberration (CA), and Sparse sampling and reconstruction (SSR).

VI. EXPERIMENTAL RESULTS

To investigate the relation between semantic features and visual quality, we train the BIQA metric using only the semantic features (see Fig. 5) as input to the RFR. Table II shows the results obtained, following the procedures described in Section V, but using only the semantic features to predict quality. The bold numbers in this table correspond to the best average SROCC scores (1,000 simulations). Notice that, depending on the different distortions and models, the visual semantic (VS) have a different effect on the accuracy performance. Also, VS-based BIQA perform better for the CSIQ database and worse for the TID2013 database. Moreover,

TABLE II
MEAN SROCC FROM 1,000 RUNS OF SIMULATIONS ON LIVE2, CSIQ, AND TID2013 DATABASES USING DIFFERENT PRE-TRAINED SEMANTICS.

Database	Distortion	Alexnet	GoogLeNet	VGG						ResNet		
				VGG-f	VGG-m	VGG-s	VGG-VD-16	VGG-VD-19	VGG-face	ResNet-50	ResNet-101	ResNet-152
LIVE	JPEG	0.4313	0.3714	0.2078	0.3085	0.3936	0.3603	0.5077	0.1583	0.5066	**0.5622**	0.4906
	JPEG2k	0.4127	0.3016	0.2998	0.4342	0.4061	0.3388	0.4245	0.1141	0.4296	**0.5164**	0.5041
	WN	0.5843	0.5456	0.5894	0.6141	**0.6604**	0.5789	0.5887	0.4384	0.5153	0.5266	0.5035
	GB	**0.6759**	0.5172	0.5721	0.5434	0.5801	0.4884	0.5137	0.2698	0.4994	0.5151	0.5992
	FF	**0.5736**	0.4566	0.5339	0.4609	0.5313	0.4709	0.4871	0.2601	0.4562	0.4703	0.5345
	ALL	0.5181	0.4267	0.4266	0.4612	0.4958	0.4383	0.4965	0.2401	0.4823	0.5273	**0.5292**
CSIQ	JPEG	0.2687	0.5946	0.3595	0.4001	0.2723	0.4839	0.5162	0.1546	**0.7093**	0.5731	0.6619
	JPEG2k	0.6578	0.6867	0.6062	0.6534	0.6737	0.5963	0.6341	0.3869	**0.7541**	0.6222	0.6657
	WN	0.1074	0.0933	**0.2105**	0.1616	0.1436	0.1161	0.1181	0.0874	0.1091	0.0806	0.1217
	GB	0.6831	0.6989	0.6094	0.6492	0.6918	0.6471	0.7118	0.4641	0.6890	0.6481	**0.7298**
	PN	0.3691	**0.5011**	0.4846	0.4861	0.3176	0.3391	0.4476	0.2327	0.2641	0.3049	0.3476
	CD	0.2436	0.2991	0.1905	0.2058	0.2941	0.3256	**0.3578**	0.2884	0.1356	0.1167	0.1719
	ALL	0.4262	**0.5268**	0.4459	0.4678	0.4378	0.4445	0.4875	0.1712	0.5135	0.4541	0.5201
TID2013	AGC	0.1542	0.1525	0.1499	0.2043	0.2011	0.1909	0.1725	0.1038	**0.2112**	0.1138	0.1364
	AGN	0.1615	0.1324	0.1277	0.1854	0.2256	0.2167	0.1619	0.1262	**0.2417**	0.1302	0.1321
	CA	0.4021	0.2929	**0.4331**	0.3689	0.3611	0.3575	0.3152	0.3939	0.3291	0.3613	0.3771
	CC	0.1441	0.1028	0.1661	0.1333	0.1761	**0.3195**	0.2959	0.2181	0.1377	0.1538	0.1103
	CCS	0.2196	0.2359	0.3171	**0.3289**	0.3214	0.2456	0.2253	0.2537	0.1794	0.2026	0.2521
	CN	0.2133	0.1991	0.2288	0.2304	0.2034	0.2943	0.1906	0.2264	**0.3522**	0.2927	0.2396
	GB	0.5903	0.4171	0.6619	0.6377	0.5979	0.5461	0.5411	0.1984	**0.6914**	0.4854	0.4322
	HFN	0.1976	0.1571	0.1573	**0.2591**	0.2403	0.2342	0.2003	0.1703	0.2535	0.1532	0.1144
	ICQ	0.2997	0.2293	0.2381	0.2942	0.3851	0.3797	**0.3953**	0.1562	0.3492	0.1923	0.2966
	ID	0.5891	0.5415	0.7045	**0.7431**	0.6797	0.7244	0.7149	0.2988	0.5705	0.4951	0.4443
	IN	0.2055	0.1754	0.2029	0.2544	0.2322	0.2271	0.2192	0.1365	**0.2711**	0.1492	0.1701
	IS	0.0816	0.1181	0.1124	0.0983	0.1048	0.0933	0.1073	0.1128	0.1091	**0.1234**	0.0928
	JPEG	0.3737	0.3399	0.3227	0.3421	0.4023	**0.5025**	0.4273	0.0969	0.4948	0.3789	0.3545
	JPEGTE	0.1731	0.1364	0.1152	0.1627	0.1763	0.2169	0.1066	0.2731	**0.2819**	0.1799	0.2251
	JPEG2k	0.6469	0.5295	0.6811	0.6376	0.6565	0.6122	0.6001	0.4044	**0.6837**	0.6237	0.6281
	JPEG2kTE	0.4345	0.2543	0.4061	0.4008	0.4529	0.3831	0.3285	0.3002	**0.4618**	0.3543	0.2806
	LBD	0.1415	**0.2331**	0.1761	0.1701	0.1651	0.1689	0.1811	0.1919	0.2034	0.1952	0.1631
	LC	0.3691	0.2317	0.3334	0.3197	0.3041	**0.4568**	0.3881	0.2547	0.4451	0.2969	0.3366
	MGN	0.1605	0.1616	0.1611	0.2369	0.2224	0.1731	0.2062	0.1405	**0.2805**	0.1297	0.1256
	MN	**0.2532**	0.2341	0.1384	0.1847	0.1988	0.1853	0.1318	0.1794	0.1663	0.1515	0.1422
	NEPN	0.1688	0.1677	**0.2549**	0.1763	0.1941	0.1371	0.1651	0.1287	0.1917	0.1171	0.1341
	QN	0.2672	0.2523	0.1768	0.2859	0.2946	0.3582	0.2816	0.2891	**0.3698**	0.1612	0.1881
	SCN	0.4097	0.2101	0.4141	0.3775	**0.4815**	0.3506	0.2846	0.1556	0.3366	0.2277	0.2177
	SSR	0.6191	0.5718	0.6399	**0.6754**	0.6390	0.6686	0.5835	0.4703	0.6608	0.5599	0.5496
	ALL	0.3055	0.2311	0.2897	0.3227	0.3311	0.3411	0.2918	0.1751	**0.3721**	0.2651	0.2543
Average		0.3561	0.3244	0.3459	0.3652	0.3723	0.3687	0.3633	0.2295	**0.3871**	0.3266	0.3362

among all tested pre-trained methods, we can observe that ResNet-50 presents the best overall (average) performance.

Given the previous results, we chose ResNet-50 as our semantic algorithm, that will be used on the combination with OCL features. We consider only the texture features (OCL) and texture features combined with visual semantic features (OCL+VS). Table III and IV depicts the results for these two methods. Numbers in italics represent the best correlation values among BIQA and FR-IQA methods, while numbers in bold correspond to the best SROCC scores considering only the BIQA methods.

From Table III, we can see that, for all databases, the proposed method achieves the best performance among the BIQA methods. For the LIVE2 database, the proposed method outperforms the BIQA methods for JPEG2, GB, and 'ALL' distortions. For CSIQ database, the proposed method has the best scores for all distortions, with the exception of CD. For TID2013, the proposed method presents the best performance for 14 out of 25 cases, followed by BRISQUE, CORNIA, and LTP. Table III also indicates that the incorporation of semantic features improves the prediction of subjective scores for LIVE2 and CSIQ. For TID2013, the incorporation of the semantics information decreases the accuracy performance, with OCL method obtaining the best performance.

To investigate the generalization capability of the proposed method, we perform a cross-database validation. This validation consists of training the ML algorithm using all images of one database and testing them on the other databases. Table IV depicts the SROCC values obtained using LIVE2 as the training database and TID2013 and CSIQ as the testing databases. To perform a straightforward cross-database comparison, only the shared subset of distortions are selected from each database. Notice that the proposed method outperforms the other methods for almost all types of distortions. For TID2013, the proposed method outperforms the other methods for 4 out of the 5 distortions, while for CSIQ it outperforms the other methods for all 5 distortions. Furthermore, the incorporation of visual semantics improves the performance in almost all cases, with the exception of JPEG2k artifacts in the CSIQ database. Therefore, the cross-database validation test indicates that the proposed method has a better generalization capability, when compared to the tested state-of-the-art

methods.

TABLE III
MEAN SROCC FROM 1000 RUNS OF SIMULATIONS ON LIVE2, CSIQ, AND TID2013 DATABASES USING THE STATE-OF-THE-ART METHODS.

DB	Distortion	PSNR	SSIM	BRISQUE	CORNIA	CQA	SSEQ	LTP	OCL	OCL+VS
LIVE	JPEG	0.8515	*0.9481*	0.8641	0.9002	0.8257	0.9122	**0.9395**	0.9312	0.9244
	JPEG2k	0.8822	*0.9438*	0.8838	0.9246	0.8366	0.9388	0.9372	0.9411	**0.9421**
	WN	*0.9851*	0.9793	0.9750	0.9500	**0.9764**	0.9544	0.9646	0.9731	0.9734
	GB	0.7818	0.8889	0.9304	0.9465	0.8377	0.9157	0.9530	0.9571	***0.9624***
	FF	0.8869	*0.9335*	0.8469	**0.9132**	0.8262	0.9038	0.8758	0.8936	0.8942
	ALL	0.8013	0.8902	0.9098	0.9386	0.8606	0.9356	0.9316	0.9418	***0.9422***
CSIQ	JPEG	0.9009	0.9309	0.8525	0.8319	0.6506	0.8066	0.9292	0.8943	***0.9437***
	JPEG2k	*0.9309*	0.9251	0.8458	0.8405	0.8214	0.7302	0.8877	0.8865	**0.9074**
	WN	*0.9345*	0.8761	0.6931	0.6187	0.7276	0.7876	0.6454	**0.8441**	0.8293
	GB	0.9358	0.9089	0.8337	0.8526	0.7486	0.7766	0.9244	0.9203	***0.9376***
	PN	*0.9315*	0.8871	0.7740	0.5340	0.5463	0.6661	0.7828	**0.8361**	0.8175
	CD	*0.8862*	0.8128	0.4255	0.4458	**0.5383**	0.4172	0.2082	0.4914	0.4169
	ALL	0.8088	0.8116	0.7597	0.6969	0.6369	0.7007	0.8280	0.8421	***0.8611***
TID2013	AGC	*0.8568*	0.7912	0.4166	0.2605	0.3964	0.3949	**0.5963**	0.5315	0.3182
	AGN	*0.9337*	0.6421	0.6416	0.5689	0.6051	0.6040	0.6631	**0.7253**	0.4539
	CA	*0.7759*	0.7158	**0.7310**	0.6844	0.4380	0.4366	0.6749	0.4254	0.4314
	CC	*0.4608*	0.3477	0.1849	0.1400	**0.2043**	0.2006	0.1886	0.0846	0.0973
	CCS	0.6892	*0.7641*	0.2715	0.2642	0.2461	0.2547	0.2384	**0.5704**	0.4911
	CN	*0.8838*	0.6465	0.2176	0.3553	0.1623	0.1642	0.3880	**0.5849**	0.3582
	GB	*0.8905*	0.8196	0.8063	0.8341	0.7019	0.7058	0.7465	**0.8607**	0.8398
	HFN	*0.9165*	0.7962	0.7103	0.7707	0.7104	0.7061	0.7626	**0.8118**	0.7101
	ICQ	*0.9087*	0.7271	0.7663	0.7044	0.6829	0.6834	0.7603	**0.7849**	0.7318
	ID	*0.9457*	0.8327	0.5243	0.7227	0.6711	0.6716	0.7063	0.7719	**0.7893**
	IN	*0.9263*	0.8055	**0.6848**	0.5874	0.4231	0.4272	0.6484	0.5069	0.3010
	IS	*0.7647*	0.7411	0.2224	0.2403	0.2011	0.2013	**0.3291**	0.1061	0.0916
	JPEG	*0.9252*	0.8275	0.7252	0.7815	0.6317	0.6284	0.6631	**0.8201**	0.7544
	JPEGTE	*0.7874*	0.6144	0.3581	**0.5679**	0.2221	0.2195	0.2314	0.5153	0.3429
	JPEG2k	*0.8934*	0.7531	0.7337	0.8089	0.7219	0.7205	0.7780	**0.8769**	0.8306
	JPEG2kTE	*0.8581*	0.7067	**0.7277**	0.6113	0.6529	0.6529	0.6594	0.5984	0.5181
	LBD	0.1301	*0.6213*	0.2833	0.2157	0.2382	0.2290	**0.3813**	0.1311	0.1746
	LC	*0.9386*	0.8311	0.5726	**0.6682**	0.4561	0.4460	0.6533	0.5692	0.3862
	MGN	*0.9085*	0.7863	0.5548	0.4393	0.4969	0.4897	0.6209	**0.6753**	0.4382
	MN	*0.8385*	0.7388	0.2650	0.2342	0.2506	0.2575	0.4243	**0.5146**	0.2874
	NEPN	*0.6931*	0.5326	0.1821	**0.2855**	0.1308	0.1275	0.1256	0.2198	0.1778
	QN	*0.8636*	0.7428	0.5383	0.4922	0.7242	0.7214	0.7361	**0.8207**	0.7769
	SCN	*0.9152*	0.7934	**0.7238**	0.7043	0.7121	0.7064	0.7015	0.7192	0.5804
	SSR	*0.9241*	0.7774	0.7101	0.8594	0.8115	0.8084	0.8457	**0.8892**	0.8867
	ALL	*0.6869*	0.5758	0.5416	0.6006	0.4925	0.4900	0.6078	**0.6417**	0.6345

TABLE IV
SROCC COMPARISON ON CROSS-DATABASE VALIDATION WHEN MODELS ARE TRAINED ON LIVE2 AND TESTED ON CSIQ AND LIVE DATABASES.

Database	Distortion	BRISQUE	CORNIA	CQA	SSEQ	LTP	OCL	OCL+VS
TID2013	JPEG	0.8058	0.7423	0.8071	0.7823	0.8472	0.8845	**0.9008**
	JPEG2k	0.8224	0.8837	0.7724	0.8258	0.9046	0.9024	**0.9301**
	WN	0.8621	0.7403	**0.8692**	0.6959	0.6881	0.8256	0.8243
	GB	0.8245	0.8133	0.8214	0.8624	0.8693	0.8641	**0.8782**
	ALL	0.7965	0.7599	0.8214	0.7955	0.8137	0.8491	**0.8586**
CSIQ	JPEG	0.8209	0.7062	0.7129	0.8141	0.8784	0.9254	**0.9308**
	JPEG2k	0.8279	0.8459	0.6957	0.7862	0.8914	**0.9151**	0.9074
	WN	0.6951	0.8627	0.6596	0.4613	0.7739	0.8799	**0.8837**
	GB	0.8311	0.8815	0.7648	0.7758	0.8712	0.9233	**0.9237**
	ALL	0.8022	0.7542	0.7114	0.7403	0.8628	0.8871	**0.8935**

VII. CONCLUSIONS

In this paper, we showed how semantic features can affect the prediction of visual quality. By combining these semantic features with the texture information, we generated a novel general-purpose BIQA method. The texture statistics are acquired using a texture descriptor: the OCL. This descriptor is used with the goal of incorporating both texture and color information into the quality measurement. This characterization is based on the fact that, when the image quality changes, it also affects the image texture information. Quality is predicted after a machine-learning training stage. Results show that, by combining the statistics of OCL with the semantic features, the accuracy performance is improved. This improvement is most visible for LIVE2 and CSIQ databases, while not present for the TID2013 database. On the other hand, in the important cross-database validation the improvement in performance is detected. This indicates that the use of semantic features are more suitable for some types of distortions. This work represents a contribution to the area of BIQA research, since it takes into account other factors besides the distortion sensitivity. The proposed approach is particularly attractive in circumstances where semantic information is readily available. Future works include the study of the impact of visual semantic on video quality assessment.

ACKNOWLEDGMENT

This work was supported by the University of Brasília (UnB), the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), and the Fundação de Empreendimentos Científicos e Tecnológicos (Finatec).

REFERENCES

[1] S. Wang, A. Rehman, Z. Wang, S. Ma, and W. Gao, "Perceptual video coding based on ssim-inspired divisive normalization," *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1418–1429, 2013.

[2] M. C. Farias and M. M. Carvalho, "Video quality assessment based on data hiding for ieee 802.11 wireless networks," in *Broadband Multimedia Systems and Broadcasting (BMSB), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 1–6.

[3] J. Joskowicz and R. Sotelo, "A model for video quality assessment considering packet loss for broadcast digital television coded in h. 264," *International Journal of Digital Multimedia Broadcasting*, vol. 2014, 2014.

[4] X. Liu, D. Zhai, D. Zhao, G. Zhai, and W. Gao, "Progressive image denoising through hybrid graph laplacian regularization: a unified framework," *IEEE Transactions on image processing*, vol. 23, no. 4, pp. 1491–1503, 2014.

[5] L. Zhang, Y. Shen, and H. Li, "Vsi: A visual saliency-induced index for perceptual image quality assessment," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4270–4281, 2014.

[6] Z. Wang and E. P. Simoncelli, "Reduced-reference image quality assessment using a wavelet-domain natural image statistic model," in *Electronic Imaging 2005*. International Society for Optics and Photonics, 2005, pp. 149–159.

[7] R. Soundararajan and A. C. Bovik, "Rred indices: Reduced reference entropic differencing for image quality assessment," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 517–526, 2012.

[8] L. Liu, Y. Hua, Q. Zhao, H. Huang, and A. C. Bovik, "Blind image quality assessment by relative gradient statistics and adaboosting neural network," *Signal Processing: Image Communication*, vol. 40, pp. 1–15, 2016.

[9] P. G. Freitas, W. Y. Akamine, and M. C. Farias, "Blind image quality assessment using multiscale local binary patterns," *Journal of Imaging Science and Technology*, vol. 60, no. 6, pp. 60 405–1, 2016.

[10] Y. Zhang, J. Wu, X. Xie, and G. Shi, "Blind image quality assessment based on local quantized pattern," in *Pacific Rim Conference on Multimedia*. Springer, 2016, pp. 241–251.

[11] Q. Li, W. Lin, and Y. Fang, "Bsd: Blind image quality assessment based on structural degradation," *Neurocomputing*, vol. 236, pp. 93 – 103, 2017, good Practices in Multimedia Modeling. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092523121631390X

[12] P. Ye and D. Doermann, "No-reference image quality assessment using visual codebooks," *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3129–3138, 2012.

[13] S. Brahnam, L. C. Jain, A. Lumini, and L. Nanni, "Introduction to local binary patterns: New variants and applications." in *Local Binary Patterns*, ser. Studies in Computational Intelligence, S. Brahnam, L. C. Jain, L. Nanni, and A. Lumini, Eds. Springer, 2013, vol. 506, pp. 1–13.

[14] P. G. Freitas, W. Y. Akamine, and M. C. Farias, "No-reference image quality assessment based on statistics of local ternary pattern," in *Quality of Multimedia Experience (QoMEX), 2016 Eighth International Conference on*. IEEE, 2016, pp. 1–6.

[15] F. Rezaie, M. S. Helfroush, and H. Danyali, "No-reference image quality assessment using local binary pattern in the wavelet domain," *Multimedia Tools and Applications*, pp. 1–13, 2017.

[16] Q. Wu, H. Li, and K. N. Ngan, "Gip: Generic image prior for no reference image quality assessment," in *Pacific Rim Conference on Multimedia*. Springer, 2016, pp. 600–608.

[17] D. M. Chandler, "Seven challenges in image quality assessment: past, present, and future research," *ISRN Signal Processing*, vol. 2013, 2013.

[18] E. Siahaan, A. Hanjalic, and J. A. Redi, "Does visual quality depend on semantics? a study on the relationship between impairment annoyance and image semantics at early attentive stages," *Electronic Imaging*, vol. 2016, no. 16, pp. 1–9, 2016.

[19] M. C. Farias and W. Y. Akamine, "On performance of image quality metrics enhanced with visual attention computational models," *Electronics letters*, vol. 48, no. 11, pp. 631–633, 2012.

[20] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

[21] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, vol. 1. IEEE, 1994, pp. 582–585.

[22] D.-C. He and L. Wang, "Texture unit, texture spectrum, and texture analysis," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 28, no. 4, pp. 509–512, 1990.

[23] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Gray scale and rotation invariant texture classification with local binary patterns," in *Computer Vision-ECCV 2000*. Springer, 2000, pp. 404–420.

[24] ——, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

[25] S. Brahnam, L. C. Jain, A. Lumini, and L. Nanni, "Introduction to local binary patterns: new variants and applications," in *Local Binary Patterns: New Variants and Applications*. Springer, 2014, pp. 1–13.

[26] T. Maenpaa, M. Pietikainen, and J. Viertola, "Separating color and pattern information for color texture discrimination," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 1. IEEE, 2002, pp. 668–671.

[27] T. Mäenpää, *The local binary pattern approach to texture analysis: extensions and applications*. Oulun yliopisto, 2003.

[28] A. Jain and G. Healey, "A multiscale representation including opponent color features for texture recognition," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 124–128, 1998.

[29] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[33] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[35] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[36] V. Rodriguez-Galiano, M. Sanchez-Castillo, M. Chica-Olmo, and M. Chica-Rivas, "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines," *Ore Geology Reviews*, vol. 71, pp. 804–818, 2015.

[37] E. Kremic and A. Subasi, "Performance of random forest and svm in face recognition." *Int. Arab J. Inf. Technol.*, vol. 13, no. 2, pp. 287–293, 2016.

[38] M. Liu, M. Wang, J. Wang, and D. Li, "Comparison of random forest, support vector machine and back propagation neural network for electronic tongue data classification: Application to the recognition of orange beverage and chinese vinegar," *Sensors and Actuators B: Chemical*, vol. 177, pp. 970–980, 2013.

[39] R. Dosselmann and X. D. Yang, "A comprehensive assessment of the structural similarity index," *Signal, Image and Video Processing*, vol. 5, no. 1, pp. 81–91, 2011.

[40] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *Image Processing, IEEE Transactions on*, vol. 21, no. 12, pp. 4695–4708, 2012.

[41] P. Ye, J. Kumar, L. Kang, and D. Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1098–1105.

[42] L. Liu, H. Dong, H. Huang, and A. C. Bovik, "No-reference image quality assessment in curvelet domain," *Signal Processing: Image Communication*, vol. 29, no. 4, pp. 494–505, 2014.

[43] L. Liu, B. Liu, H. Huang, and A. C. Bovik, "No-reference image quality assessment based on spatial and spectral entropies," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 856–863, 2014.

[44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[45] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 689–692.

[46] H. R. Sheikh, Z. Wang, L. Cormack, and A. C. Bovik, "LIVE image quality assessment database release 2," http://live.ece.utexas.edu/research/quality, 2005, accessed: 2016-09-30.

[47] E. C. Larson and D. Chandler, "Categorical image quality (CSIQ) database," Online, http://vision.okstate.edu/csiq, 2010, accessed: 2016-09-30.

[48] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti *et al.*, "Image database tid2013: Peculiarities, results and perspectives," *Signal Processing: Image Communication*, vol. 30, pp. 57–77, 2015.