2016

Photo Quality Assessment: Predicting Crowd Opinions

Uyen T. Mai
Portland State University

Follow this and additional works at: <https://pdxscholar.library.pdx.edu/honorstheses>

Let us know how access to this document benefits you.

Recommended Citation

Mai, Uyen T., "Photo Quality Assessment: Predicting Crowd Opinions" (2016). *University Honors Theses*. Paper 241.

<https://doi.org/10.15760/honors.283>

This Thesis is brought to you for free and open access. It has been accepted for inclusion in University Honors Theses by an authorized administrator of PDXScholar. Please contact us if we can make this document more accessible: pdxscholar@pdx.edu.

Photo Quality Assessment: Predicting Crowd Opinions

Anonymous ICCV submission

Paper ID 472

Abstract

Existing methods for photo quality assessment typically formulate photo quality assessment as a binary classification problem that labels a photo as low- or high-quality. Photo quality assessment, however, is subjective and people often rate a photo differently. Therefore, the quality of a photo sometimes cannot be fully described by a low- or high-quality label. In this paper, we present a subjective photo quality assessment method that predicts how a group of users rate a photo. Specifically, our method predicts a quality score distribution that is likely produced by a group of people rating the photo. Our method models the score distribution using the mean and standard deviation. Our method uses a regression approach and integrates a wide spectrum of image features, including manually crafted features, generic image features, and deep learning features, to predict the mean score and standard deviation. We experiment our method on the large scale AVA dataset where each photo on average is rated by xxx users with score ranges from 1-10. Our experiment shows that our regression approach can predict the mean score and standard deviation with RMSE errors 0.06 and xxx, respectively.

1. Introduction

Image quality assessment is important for a wide variety of applications and accordingly has attracted a significant amount of research effort. Early methods for quality assessment focus on the quality degradation due to compression and network transmission [4, 12, 24, 28, 22, 23, 19, 27, 30, 32, 31, 24, 34, 2]. These methods assess image quality by detecting and measuring various distortions, including blocking, ringing, mosaic patterns, blur, noise, ghosting, jerkiness, smearing, etc. While they are effective for measuring quality loss due to compression or data loss during transmission, these low-level distortion measurement-based metrics sometimes do not well reflect people's subjective perception of image quality.

Recently, subjective image quality assessment methods have been developed [1, 6, 5, 8, 9, 10, 14, 15, 16, 17, 20, 25,

(a) Photo with consistent rates (b) Photo with inconsistent rates
Figure 1. User rate distribution. Both photos in (a) and (b) have a similar average score around xxx; however, the user rates for (a) is mostly consistent while the user rates for (b) diverge significantly.

26, 33]. Most of these methods typically adopt a data-driven approach that represents images using carefully manually crafted features [10, 5, 15], generic image features [16], or features produced by deep learning methods [14], and then trains a classifier to label an input image as low or high quality. However, people's perception of image quality is very complex and often varies. For a single image, different people often have different opinions.

Figure 1 show two images from the AVA image quality assessment benchmark [18]. Each of these two images is rated by around xxx users. The score ranges from 1 to 10 with 10 being the best. The user rates for the image in (b) vary significantly. For this particular photo, some users consider it of high quality, saying "wonderful mood, tonality, and textures, just very painterly shot", and "has an awesome sense of motion, like any minute it's all going to pick itself up and waltz away" [7], while other users may give low scores for being dull. It is difficult for a binary label to fully convey the quality of this particular image. Sometimes, it is even difficult to use the average or median rate to indicate the image quality. Both photos in (a) and (b) have a

middle-range score, around xx out of 10, but the user rates for (a) are very consistent, implying this is a photo with an averaging quality while the user rates for (b) vary significantly as that photo is unorthodox: some users like it while the others do not.

In this paper, we propose to evaluate the quality of an image according to how a group of users are likely to rate this image. For example, for the image in Figure 1 (b), we will rate its quality with the mean and the standard deviation of the rates. The mean provides an overall quality assessment and the standard deviation indicates whether users' perception of the image quality is consistent or not. We expect that these two work together and can already provide a meaningful assessment of the users' subjective quality perception.

In this paper, we present a subjective image quality assessment method that predicts the crowd opinion about an image. Specifically, our method predicts and rates the quality of an image with the distribution of rates that a group of users are likely to cast. Our method uses the mean and the standard deviation to describe the rate distribution. Our method uses a regression approach that integrates a range of features, including carefully manually crafted features, generic image features, and deep learning features, to predict the mean and the standard deviation of the scores that users are likely to rate this image with.

The main contribution of this paper is the use of crowd user opinion to measure image quality instead of the binary low- or high-quality label. Accordingly, this paper also provides a regression approach to predict the crowd user opinion in terms of the mean and the standard deviation. Our experiments on the large-scale AVA benchmark show that our method can predict the mean score and standard deviation with RMSE errors 0.06 and xxx, respectively.

2. Quality Score Distribution Prediction

A wide variety of features have been designed and used for photo quality assessment. This paper leverages these existing feature designs to predict how a group of users are likely to rate an image. We include three categories of features, namely manually crafted image features for image quality assessment, generic image features, and features learned by a deep learning algorithm specifically for photo quality assessment. While the performance of features varies, our goal is to include as many features as possible and use a data-driven method to make proper use of them. Below we first briefly introduce the set of features used in our method for completeness and then describe how we use a regression approach to predict the crowd opinions of a photo's quality.

2.1. Features

2.1.1 Manually Crafted Photo Quality Features

Previous research on photo quality assessment provides a ranges of carefully crafted image features. These features are shown effectively on some early subjective image quality assessment benchmarks [?]. We therefore include and test them in our method. For completeness, we briefly describe these features below. Please refer to the original papers for more details.

Basic image statistics. Following [5, 10], our method includes the average hue, saturation, intensity, and brightness as features. Our method also include basic image properties, including the image size and aspect ratio [5].

Hue count. As described in [10], while most high-quality images look more colorful than low-quality ones, they surprisingly contain a smaller number of hues. Therefore, the hue count of a photo can be used to indicate the photo quality. Specifically, we first convert an image into HSV color space and then construct a 20-bin hue histogram. We only use pixels with the brightness value in the range of [0.15, 0.95] and the saturation value > 0.2 to construct the hue histogram. As detailed in [10], we then compute the hue number f_{hnum} from the hue histogram H as follows.

$$f_{hnum} = 20 - \|N\|, \text{ where } N = \{i, H(i) > \alpha m\} \quad (1)$$

where m is the histogram's largest bin value, and α is a parameter with the value 0.05 as suggested in [10].

Blur. Although professional photographers often use a narrow depth of field to blur the background so that the main subject stands out, an overall blurry image is typically of bad quality. Our method therefore includes the blur feature from [10]. Specifically, a blurring image is considered as a result of applying a Gaussian filter to a sharp image. Accordingly, the blurriness can be measured by computing the 2D Fourier transform of the image and calculating as the amount of high-frequency components removed from the input image.

$$f_b = \frac{\|I\|}{\|C\|}, \text{ where } C = \{(u, v) | |F(u, v)| > \theta\} \quad (2)$$

where I and F are the input image and its 2D Fourier transform, and $\|I\|$ is the number of pixels in I . $\|C\|$ is the maximum frequency present in I as the Gaussian filter removes only high frequencies. θ is a parameter with value 5, as suggested in [10].

Contrast. Professional photos often have high contrast. We therefore measure the contrast as a feature using the histogram method from [10]. Specifically, we first compute the gray scale histogram and then measure the width of the middle 98% mass of the histogram as the contrast feature.

Compositional Features. Professional high-quality photos are often well composed. We therefore include features to measure how a photo respects photo compositional rules.

- Rule of thirds. If we divide an image into 3×3 grids, rule of thirds states that important image content should be placed along one of the thirds lines or their intersections [11]. We use the method from [5] to compute the rule-of-thirds feature according to the spatial distribution of the image content with regard to the thirds lines. Please refer to [5] for more detail.
- Depth of field. Professional photographers often use a shallow depth of field to blur the background to draw a viewer's attention to the sharp subject of interest. We use the method from [5] to compute features that reflect the depth-of-field of an image. Specifically, we perform wavelet transform on each of the three image channels (H, S, and V) to get the high-frequency wavelet coefficients w_h . We uniformly divide the image into 4×4 grids and then compute the ratio between the amount of high-frequency components in the center-four blocks and the whole image as the depth-of-field feature. We compute such a feature for each of the three image channels (H, S, V) separately.

$$f_{dof}^H = \frac{\sum_{(x,y) \in M_6 \cup M_7 \cup M_{10} \cup M_{11}} w_h^H(x,y)}{\sum_{i=1}^{16} \sum_{(x,y) \in M_i} w_h^H(x,y)} \quad (3)$$

where M_i is one of the 16 image blocks, M_6 , M_7 , M_{10} , M_{11} are the four middle four blocks. The feature values for the S and V channels are computed in the same way.

- Texture analysis. Texture is often used for photo composition. We use the method from [5] to characterize the texture in an image. Specifically, we perform Daubechies wavelet transform to measure the spatial smoothness of the image. We apply a three-level wavelet transform to each of the three HSV color channels and obtain three levels of wavelet transforms. We finally calculate the average amount of high frequencies at each of the three levels.

$$f_{tex}^i = \frac{\sum_{x,y} w_i^{hl}(x,y) + \sum_{x,y} w_i^{lh}(x,y) + \sum_{x,y} w_i^{hh}(x,y)}{|w_i^{hl}| + |w_i^{lh}| + |w_i^{hh}|} \quad (4)$$

where $w_i^{hl}(x,y)$, $w_i^{lh}(x,y)$, and $w_i^{hh}(x,y)$ are the three high-frequency components of the wavelet transform at level i . We have three levels, therefore obtain 3 feature values for each color channel. For a color image, we obtain 9 feature values in total.

- Spatial edge distribution. In professional photos, the image background is often simplified to emphasize the important foreground object. Therefore, high frequency edges are often clustered around the image center. We follow [10] to measure the spatial distributions of image edges to reflect this photography rule. Specifically, we apply a Laplacian filter to an input

image and obtain a Laplacian image, which is then resized to 100×100 . We then measure the size of the image area that edges occupy by projecting the Laplacian image onto the x and y axes. We finally find the width w_x and w_y that contain 98% mass of each projection. Then the edge distribution feature is computed as follows

$$f_{ed1} = 1 - w_x w_y. \quad (5)$$

Ke *et al.* also provides another image edge distribution feature that is computed according to the edge distribution similarity between an image and an average professional photo (or an average snap photo) [10].

$$f_{ed2} = \sum_{x,y} |L(x,y) - L_s(x,y)| - |L(x,y) - L_p(x,y)| \quad (6)$$

where L , L_p , and L_s is the Laplacian of an input photo, the mean Laplacian of professional and snapshot photos, respectively.

Saliency-enhanced features. Luo *et al.* found that roughly separating the subject region from the background and then computing quality features can enhance the photo quality assessment results [15]. We follow their methods to extract the saliency map and then compute four saliency-enhanced features, including clarity/contrast, lighting, simplicity, and composition geometry. Please refer to [15] for details.

Attribute-based Features. We also include two popular attribute-based features from Dharētal [6]: depth of field and presence of face. attribute features are binary. Specifically, we use the method from [6] to determine whether an image can be tagged with depth of field or not. For the presence of face attribute, we use the Viola-Jones face detector [29].

2.1.2 Generic Features

Generic image descriptors trained in an unsupervised manner on a large set of images have been shown successful in many visual recognition tasks such as object recognition [?], scene classification [?], and image quality classification [?]. In this paper, we employ 3 types of generic features:

Bag-of-Visual-Word (BOV): In the BOV model, each image is divided into a set of local patches, Each represented as a descriptor extracted from the low-level features such as edges and colors. A clustering algorithm is used to quantized the space of patch descriptors collected from a large number of training images to form a dictionary of code words, each corresponds to a cluster component. Each image is then represented as a histogram of code words formed by its collection of local patch descriptors. In our implementation of BOV model, we use the average RGB color values and the SIFT feature [13] as patch descriptors and *Gaussian Mixture Model (GMM)* clustering algorithm with 1024 Gaussians for dictionary creation.

Fisher Vector is a powerful method that combine generative approach and discriminative approach in learning image descriptors [?]. Similar to the BOV model, each image is first represented by a set of local patch descriptors. The collection of local descriptors from a large number of training image is then used to train a GMM model. Given a new image, the the density function of the GMM is used to compute its derivatives of each local patch descriptors, which are then accumulated to form the global descriptor of the image. In this paper, we use the VLFeat¹ Fisher Vector implementation of [?] with both SIFT and color features as local patch descriptors and 256 Gaussians for GMM training.

GIST: The GIST features was first proposed by Oliva and Torralba [21] for scene categorization, then was used by Marchesotti *et al.* for aesthetic assessment [16]. To compute GIST features we use the code published online by the authors [21].

2.1.3 Deep Learning Features

2.2. Score Distribution Regression

In this paper, we represent the score distribution by its mean value and standard deviation. Given an image I and its feature vector f_I , our goal is to construct the prediction function g that maps f_I to the image's score distribution S_I .

$$S_I = [\mu_{S_I}, \sigma_{S_I}] = g(f_I) \quad (7)$$

where μ_S and σ_S denote the mean and standard deviation of the score distribution S , respectively.

In this paper, we develop a data-driven approach to learn the prediction function g from training data. In particular, we model our prediction function g using a *Random Forest* (RF) framework. A random forest is an collection of randomized decision trees [3] independently trained on randomly selected set of features. In this paper, we customize the decision tree model to jointly predict the mean and standard deviation of the score distribution for an image.

Decision Tree Model for Score Distribution Prediction

In each decision tree, an input data f_I enters at the root node and recursively traverse down the tree until it reaches a leaf node. Each internal node i in the tree stores a binary decision function which uses the input's feature vector to decide which branch to direct the traversal of the input.

$$h_i(f_I; n_i, \tau_i) = I\{\phi_i(f_I, n_i) > \tau_i\} \quad (8)$$

where $\phi_i(f, n)$ extracts the n^{th} dimension of the vector f . I denotes the indicator function which returns 1 if its argument is true and zero otherwise. An input data f_I reaching

node i will traverse to its left branch if $h_i(f_I; n_i, \tau_i)$ returns 1 and to its right branch otherwise. The node parameters $\Theta_i = \{n_i, \tau_i\}$ at each node i is learned from the training data.

After training, each leaf L of tree contains a subset of training data points D_L traversed to the leaf. Given an input f_I , the decision tree T generate its prediction $g_T(f_I)$ by aggregating the label values of the training data stored at leaf node.

$$g_T(f_I) = \frac{1}{|D_{L(f_I)}|} \sum_{i \in D_{L(f_I)}} S_i \quad (9)$$

where $L(f_I)$ denotes the the leaf node in the tree that the new data f_I reaches after traversing the tree, $D_{L(f_I)}$ denotes the subset of training data assigned the leaf $L(f_I)$ by the training process. The training process of the decision tree optimizes the tree structure and each node's decision function parameters such that the training data stored at leaf have similar prediction labels. In this paper, we customize the tree node objective function to our task of score distribution prediction.

Node Training Objective Function: As our goal is to jointly predict the mean and standard deviation of the score distribution for an input image, we train the decision function at each node such that when a set of training data is passed to the node, it will be split into two subsets which are more compact in the joint space of μ_i and σ_i than the original set of training data. Specifically, we define our node training optimization as

$$\Theta_i = \underset{\Theta}{\operatorname{argmax}} V(D_i) - \sum_{j \in \{L, R\}} w_j V(D_i^j(\Theta)) \quad (10)$$

where D_i represents the set of training data reaches node i . $D_i^L(\Theta)$ and $D_i^R(\Theta)$ represents the subsets of D_i that are passed to the left and right branches of node i by evaluating the decision function in Equation 8 with parameters Θ . The weight $w_j = \frac{|D_i^j(\Theta)|}{|D_i|}$ weighs each subset by its relative size to avoid severely unbalanced split. The function $V(D_i)$ measures how much the set of data points are spread out in the joint space of aesthetics score mean and standard deviation.

$$V(D) = \frac{1}{|D|} \sum_{i \in D} \|S_i - \bar{S}\|^2 \quad (11)$$

where $S_i = [\mu_i, \sigma_i]$ is the score distribution labels, $\bar{S} = \frac{1}{|D|} \sum_{i \in D} S_i$ is the average score distribution. By training the decision tree to optimize the compactness of score distribution of training data in each node, while acting on the feature space, the resulted decision tree model can effectively capture the relation between the feature vectors f_I of the input image and its score distribution S_I .

¹<http://www.vlfeat.org/api/fisher.html>

Random Forest Prediction for Score Distribution

Given an image I and its feature vector f_I , the prediction of the overall forest model is obtained by accumulating the prediction from each tree.

$$g(f_I) = g_{RF}(f_I) = \frac{1}{T} \sum_{i=1}^T g_{T_i}(f_I) \quad (12)$$

where $\{T_i\}_{i=1..T}$ denotes the set of trees in the random forest. In this paper, we implement our Random Forest model using the Scikit-learn Python library².

3. Experiments

We experiment with our method on the *Aesthetic Visual Analysis (AVA)* benchmark provided by Murray *et al.* [18]. This dataset contains 255,529 images. As reported in [18], each image in the dataset is rated by 200 human raters on average. The scores range from 1 to 10, with 10 being the best. Please refer to [18] for a detailed discussion on this benchmark.

In our experiments, the score distribution for each image I in the dataset is represented by the mean μ_I and the standard deviation σ_I from all the scores given to image I by the raters. We use the same data partition as the recent work [14, 18], namely the training set with 235,599 images and the testing set with the remaining 19,930 images.

3.1. Score Distribution Prediction

We first evaluate the accuracy of our prediction on the mean and standard deviation for each image in the testing set using two standard regression accuracy metrics.

Root Mean Square Error (RMSE): Given a testing image set D where each image $I \in D$ has its ground-truth score distribution mean μ_I and standard deviation σ_I , and their prediction $\hat{\mu}_I$, $\hat{\sigma}_I$, the RMSE for the mean and standard deviation prediction is computed as

$$RMSE_{\mu}(D) = \sqrt{\frac{\sum_{I \in D} (\mu_I - \hat{\mu}_I)^2}{|D|}} \quad (13)$$

$$RMSE_{\sigma}(D) = \sqrt{\frac{\sum_{I \in D} (\sigma_I - \hat{\sigma}_I)^2}{|D|}} \quad (14)$$

Mean Absolute Error (MAE) is another popular accuracy metric to evaluate regression results. The MAE of the mean and standard prediction over the testing image set D is defined as

$$MAE_{\mu}(D) = \frac{1}{|D|} \sum_{I \in D} |\mu_I - \hat{\mu}_I| \quad (15)$$

²<http://scikit-learn.org/>

Table 1. RMSE Regression Accuracy

	$RMSE_{\mu}$	$RMSE_{\sigma}$
Hand Crafted Feature	0.69	0.18
Deep Learning Feature	0.7	0.20
BOV	0.72	0.19
Fisher Vector (FV)	0.76	0.20
GIST	0.75	0.19
FV + BOV + GIST	0.71	0.19
Combined	0.68	0.18

Table 2. MAE Regression Accuracy

	MAE_{μ}	MAE_{σ}
Hand Crafted Feature	0.56	0.14
Deep Learning Feature	0.56	0.16
BOV	0.58	0.15
Fisher Vector	0.62	0.16
GIST	0.62	0.15
FV + BOV + GIST	0.57	0.15
Combined	0.55	0.14

$$MAE_{\sigma}(D) = \frac{1}{|D|} \sum_{I \in D} |\sigma_I - \hat{\sigma}_I| \quad (16)$$

Table 1 and Table 2 show the regression accuracy for mean and standard deviation (std) prediction over the testing set using our Random Forest Score Distribution Regression (Section ??) with each type of features. Considered individually, the manually crafted features consistently gives the smallest regression error for both mean and std prediction. Combining all features together can further improve the prediction accuracy.

3.1.1 Comparing to the Baseline

To serve as our baseline, we consider the predictor that analyze the score distribution mean and standard deviation from all images in the training set to select a fixed value prediction $S^* = [\mu^*, \sigma^*]$ that minimize the error over the training set

$$S^* = \underset{I \in D_{train}}{\operatorname{argmin}}_S \sum ||S - S_I||^2 \quad (17)$$

where $||\cdot||$ denotes the L_2 vector norm. D_{train} is the set of training images. $S_I = [\mu_I, \sigma_I]$ denotes the ground-truth mean and std of the image I in the training set. The score S^* use the ground-truth score distribution of the training image set to predict the most likely score distribution an image obtains, regardless of its features. It is then used as a fixed-value prediction for all new test images.

To compare our regression results with the baseline predictor S^* , we compute the average error reduction of our

Figure 2. Error Reduction Rate Relative to the baseline method. learning the regression models on the useful features can provide better prediction accuracy than simply analyzing the score distribution of the training data without employing the image features.

regression results over S^* on all testing images. Specifically, the error reduction of the regression results $\hat{S} = \{\hat{S}_I\}_{I \in D_{test}}$ over the baseline S^* can be computed as

$$ER(\hat{S}) = 100(\%) \times \frac{1}{|D_{test}|} \sum_{I \in D_{test}} 1 - \frac{\|\hat{S}_I - S_I\|^2}{\|S^* - S_I\|^2} \quad (18)$$

Figure 4 shows the average error reduction of our regression results against the baseline over all testing images. The results show the usefulness of our features in score distribution prediction. Combining all the features, our score distribution regression model improve the reduction accuracy up to 14 percent compared to the baseline. Interestingly, while the BOV feature and the GIST feature do not improve over the baseline prediction when used independently, when they are combined together with the Fisher Vector feature, they can significantly improve the prediction performance.

This experiment shows that learning the regression models on the useful features can provide better prediction accuracy than simply analyzing the score distribution of the training data without employing the image features.

3.2. Joint Estimation Accuracy

The evaluation metrics RMSE and MAE used in the previous experiment evaluate the regression results on the mean and std of the score distribution independently. We further evaluate the performance of our score distribution prediction method by evaluating how good it is in the joint prediction of mean and std.

In this experiment, we develop a metric to comprehensively visualize the joint prediction accuracy. We consider the prediction $\hat{S}_I = [\hat{\mu}_I, \hat{\sigma}_I]$ for a test image I an ϵ -accurate prediction if the Euclidean distance between $\hat{S}_I$ and the ground-truth $S_I = [\mu_I, \sigma_I]$ falls below a tolerance threshold ϵ . We define the ϵ -Accuracy of the regression results over the testing set as the proportion of the test images that are ϵ -accurate. Varying the threshold value ϵ provides the ϵ -accuracy curves for the regression results.

Figure 3. ϵ -Accuracy curves. Each point in the curve indicates the percentage of test images our regression models accurately predict the score distribution mean and std with respect to the tolerance threshold value ϵ

Figure 4. Feature Effectiveness.

Figure 3 shows the ϵ -accuracy curves for the regression results from our method using different features. Our method produces encouraging results in predicting the user rate distribution. The regression models can accurately predict both the the mean and std of the high percentage of test images even at low error threshold values ϵ .

Using our score distribution prediction results can help understand the crowd opinion of the image aesthetics quality. Ranking the images in the testing image set according to our predicted mean scores, we can retrieve the consistently high- and low-score images. The top row and bottom row of Figure ?? shows the images with highest and lowest predicted mean scores, respectively. For the image with mid-range mean score, we can also distinguish the images for which the user rates are consistent (the second row of Figure ??) from those for which the rates are inconsistent (the third row of Figure ??) by sorting those mid-range score images according to their predicted std.

3.3. Feature Analysis

In this paper, we make use if three groups of features to learn our score distribution regression models. To further analyze the effect of each types of features in our re-

Figure 5. A challenging example.

gression model, we compare the performance of the models trained with each group of features separately, the combination of them, and the combination of two out of three feature groups. We compare the score distribution prediction accuracy of each resulted regression model by assessing their error reduction rate relative to the baseline (Section 3.1.1). Figure ?? shows the comparison among our regression models trained using different group of features. From the results, we have the following observations.

First, the manually-crafted features and deep learning based features are more effective for score distribution prediction than generic features. One possible reason for the inferior performance of the generic features is that the dictionary used to form the generic features are learnt in an unsupervised manner. While the resulted features are generic to be useful for a wide range of visual recognition and classification tasks, they may not be discriminative enough for the specific fine-grained task of aesthetic score distribution regression.

Second, the results show that the manually-crated feature is more effective than the deep learning based features. This result can be explained by the fact that the aesthetics score prediction requires the recognition of not only the mid-level image features, but also the high-level, more abstract concept of aesthetics, such as composition, lighting and color harmonization. While deep learning based features are trained to capture useful mid-level image information, the manually-crafted features are designed to directly capture the recognition of the aesthetics concepts from the image. That renders the manually-crated features more informative to learn the score distribution prediction model.

Finally, while the three groups of features have different level of effectiveness, they seem complement one another. Removing each type of features from the combination decrease the regression performance.

3.4. Discussion

While the performance of our method in predicting the user rate distribution is encouraging, image quality assessment is inherently a challenging task. Figure 5 shows an example where the user rates diverges significantly. Some users consider this image interesting and find the photographer humorous while some others feel “disturbed” looking

at this photo. A method needs to have a semantic understanding of this photo to capture the user perception of its quality. For this particular example, incorporating optical character recognition and text analysis may help.

Photo quality assessment is subjective and a user’s photo quality perception at least partially depends on the user. Therefore, we expect that it will be useful to consider each individual user in photo quality assessment. We can model the user’s preference and incorporate the user model in image quality assessment possibly by borrowing research on recommendation systems.

4. Conclusion

References

- [1] S. Bhattacharya, R. Sukthankar, and M. Shah. A framework for photo-quality assessment and enhancement based on visual aesthetics. In *Proceedings of the International Conference on Multimedia*, pages 271–280, 2010. 1
- [2] T. Brandão and M. Queluz. No-reference quality assessment of h. 264/avc encoded video. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1437–1447, 2010. 1
- [3] L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, Oct. 2001. 4
- [4] N. Damera-Venkata, T. Kite, W. Geisler, B. Evans, and A. Bovik. Image quality assessment based on a degradation model. *IEEE Transactions on Image Processing*, 9(4):636–650, 2000. 1
- [5] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part III, ECCV’06*, pages 288–301, Berlin, Heidelberg, 2006. Springer-Verlag. 1, 2, 3
- [6] S. Dhar, V. Ordonez, and T. Berg. High level describable attributes for predicting aesthetics and interestingness. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1657–1664, June 2011. 1, 3
- [7] Dpchallenge Users. http://www.dpchallenge.com/image.php?image_id=932230. 1
- [8] W. Jiang, A. Loui, and C. Cerosaletti. Automatic aesthetic value assessment in photographic images. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pages 920–925, July 2010. 1
- [9] D. Joshi, R. Datta, E. Fedorovskaya, Q.-T. Luong, J. Wang, J. Li, and J. Luo. Aesthetics and emotions in images. *Signal Processing Magazine, IEEE*, 28(5):94–115, Sept 2011. 1
- [10] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 419–426, June 2006. 1, 2, 3
- [11] B. P. Krages. *The Art of Composition*. Allworth Communications, Inc., 2005. 3

- [12] X. Li. Blind image quality assessment. In *Proceedings of International Conference on Image Processing*, pages 449–453, 2002. 1
- [13] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 3
- [14] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang. Rapid: Rating pictorial aesthetics using deep learning. In *Proceedings of the ACM International Conference on Multimedia*, pages 457–466, 2014. 1, 5
- [15] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *Proceedings of the 10th European Conference on Computer Vision*, pages 386–399, 2008. 1, 3
- [16] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csuska. Assessing the aesthetic quality of photographs using generic image descriptors. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1784–1791, Nov 2011. 1, 4
- [17] S.-M. C. Morgens and A. Jhala. Synthetic photographs for learning aesthetic preferences. In *AAAI (Late-Breaking Developments)*, volume WS-13-17 of *AAAI Workshops*. AAAI, 2013. 1
- [18] N. Murray, L. Marchesotti, and F. Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2408–2415, June 2012. 1, 5
- [19] A. Ninassi, O. Le Meur, P. Le Callet, and D. Barba. Considering temporal variations of spatial visual distortions in video quality assessment. *IEEE Journal of Selected Topics in Signal Processing*, 3(2):253–265, 2009. 1
- [20] M. Nishiyama, T. Okabe, I. Sato, and Y. Sato. Aesthetic quality classification of photographs based on color harmony. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 33–40, June 2011. 1
- [21] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vision*, 42(3):145–175, May 2001. 4
- [22] Y. Ou, Z. Ma, T. Liu, and Y. Wang. Perceptual quality assessment of video considering both frame rate and quantization artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, (99):1–1, 2010. 1
- [23] K. Seshadrinathan and A. Bovik. Motion tuned spatio-temporal quality assessment of natural videos. *IEEE Transactions on Image Processing*, 19(2):335–350, 2010. 1
- [24] H. Sheikh, A. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *IEEE Transactions on Image Processing*, 14(11):1918–1927, 2005. 1
- [25] H.-H. Su, T.-W. Chen, C.-C. Kao, W. H. Hsu, and S.-Y. Chien. Scenic photo quality assessment with bag of aesthetics-preserving features. In *Proceedings of the 19th ACM International Conference on Multimedia*, pages 1213–1216, 2011. 1
- [26] X. Tang, W. Luo, and X. Wang. Content-based photo quality assessment. *Multimedia, IEEE Transactions on*, 15(8):1930–1943, Dec 2013. 1
- [27] H. Tong, M. Li, H. Zhang, and C. Zhang. No-reference quality assessment for jpeg2000 compressed images. In *IEEE International Conference on Image Processing*, volume 5, pages 3539–3542, 2004. 1
- [28] H. Tong, M. Li, H. Zhang, C. Zhang, J. He, and W. Ma. Learning no-reference quality metric by examples. In *Proceedings of International Conference on Multimedia Modelling*, 2005. 1
- [29] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 511–518, 2001. 3
- [30] Z. Wang, A. Bovik, and B. Evan. Blind measurement of blocking artifacts in images. In *IEEE International Conference on Image Processing*, volume 3, pages 981–984, 2000. 1
- [31] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 1
- [32] Z. Wang, H. Sheikh, and A. Bovik. No-reference perceptual quality assessment of jpeg compressed images. In *IEEE International Conference on Image Processing*, volume 1, pages 477–480, 2002. 1
- [33] O. Wu, W. Hu, and J. Gao. Learning to predict the perceived visual quality of photos. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 225–232, Nov 2011. 1
- [34] F. Yang, S. Wan, Q. Xie, and H. Wu. No-reference quality assessment for networked video via primary analysis of bit stream. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1544–1554, 2010. 1