Assessing Personally Perceived Image Quality via Image Features and Collaborative Filtering

Jari Korhonen
School of Computer Science and Software Engineering, Shenzhen University, China
jari.t.korhonen@ieee.org

Abstract

During the past few years, different methods for optimizing the camera settings and post-processing techniques to improve the subjective quality of consumer photos have been studied extensively. However, most of the research in the prior art has focused on finding the optimal method for an average user. Since there is large deviation in personal opinions and aesthetic standards, the next challenge is to find the settings and post-processing techniques that fit to the individual users' personal taste. In this study, we aim to predict the personally perceived image quality by combining classical image feature analysis and collaboration filtering approach known from the recommendation systems. The experimental results for the proposed method show promising results. As a practical application, our work can be used for personalizing the camera settings or post-processing parameters for different users and images.

1. Introduction

As a result of the rapid development of mobile technology and integration of low-cost cameras in smartphones and tablets, the amount of visual content produced and shared in the internet by regular consumers is constantly growing. Due to the technical limitations of the compact mobile cameras, image quality enhancement via post-processing is typically more essential for mobile photos than in traditional photography. Since many users may lack the skills and interest for retouching their photos manually, automatic post-processing of consumer photos is an interesting and demanded application.

Several different algorithms for image enhancement are known from the prior art, ranging from denoising, sharpening and contrast enhancement to aesthetic and artistic filtering [1-8]. Since the ground truth for a visually pleasant image is based on users' subjective opinions and the sense of aesthetics, the post-processing studies are often accompanied by a subjective quality assessment study [4, 6, 8]. It is also possible to use automated (i.e. algorithm-based) image quality metrics for evaluating the quality of post-processed images. Unfortunately, the general purpose image quality metrics tend to perform poorly on post-processed images, and this is why several artifact specific objective image quality metrics designed for assessing particularly post-processed images have also been proposed in the literature [6-14].

Conventionally, the ground truth for the subjective quality obtained from a user study represents the average opinion by several test users. However, when aesthetic image enhancement is concerned, different users may have substantially different opinions about the quality of different images. In many applications, it could be desirable to select the post-processing method based on the personal opinion, rather than the average opinion. For example, if the images are intended for the personal use only, or if the user wants to apply the personally preferred post-processing method to the images shared in the social media, personalized camera settings or post-processing would be desired. Personal post-processing could be also useful in applications where several versions of the same image are available in the cloud, and the version that is displayed in the application is selected individually for each user, according to their personal taste.

In this paper, we study the problem of predicting the personal image quality preferences by combining the conventional approach for image quality assessment, based on a bag of features extracted from the image, with the collaborative filtering approach [15, 16], where the user's ratings are known for some images and this information, together with the image features, is exploited to predict the user's ratings for other images. The application could gather information about user preferences in different ways; for example, asking the user occasionally to select the preferred alternative among two differently processed images or two photos taken with different camera settings, or keeping a track of the user's preferred methods for manual retouching. In the experimental part of our study, we use annotated image quality databases with subjective ratings available, and the implementation details for collecting the subjective opinions in practical applications is beyond the scope of this paper.

The rest of this paper is organized as follows. In Section 2, we discuss the background and the relevant related work

concerning image quality enhancement and collaborative filtering for recommendation systems. In Section 3, we explain the proposed approach. In Section 4, we demonstrate the feasibility of the proposed method by presenting the experimental results, followed by brief discussion. Finally, the concluding remarks are given in Section 5.

2. Background and related work

Optimizing image post-processing techniques, such as denoising and contrast enhancement, has been studied extensively during the past years. Typically, those studies rely on subjective quality assessment studies to find the visually most preferred images among different alternatives. Then, different analytical or learning-based image quality metrics can be applied to find the most appropriate post-processing technique for each image. Collaborative filtering for recommendation systems has also been a topic of active research recently. In this study, we use image features together with collaborative filtering for predicting the personally experienced image quality of different images. To the best of our knowledge, this is the first attempt to combine image feature analysis and collaborative filtering for predicting image quality ratings by different users.

2.1. Subjective image quality

Several different post-processing methods have been proposed for improving the visually perceived image quality. Denoising [1, 3, 5] is particularly important for photos taken with a high ISO sensitivity value in low light conditions and therefore exposed to strong sensor noise. Image contrast and sharpness are strongly related to visual quality, and this is why several different contrast enhancement and sharpening techniques have been proposed to improve image quality, ranging from local sharpening at edges to different histogram adjustment methods for global contrast enhancement [6-8].

Since aesthetic judgement is subjective by nature, it is not possible to obtain ground truth for optimally enhanced images by analytical means. Several subjective studies concerning post-processed images have been published in the prior art [4, 6, 8, 10, 12, 13, 17, 18]. The visible differences between post-processed images are often rather small, and this is why conventional numerical rating scales (e.g. 1-5) may not discriminate different methods effectively. Therefore, pairwise comparison (PC) method is popular for subjective comparison of image enhancement techniques [4, 8, 10, 12, 17]. Rank ordering method has also been used for the same purpose [18].

Since subjective assessment studies are typically rather expensive in terms of time and resources (e.g. recruitment of test users can be time consuming), no-reference image quality metrics have been studied intensively. The most widely adopted general purpose blind image quality metrics, such as NIQE [19], have not shown very impressive accuracy in predicting the quality of post-processed images [6, 8], and this is why metrics designed specifically for assessing image post-processing algorithms have also been proposed [6-13]. We assume that the more recently developed learning-based image quality metrics, such as FRIQUEE [20], could achieve better accuracy in predicting the average subjective quality of post-processed images, if trained with an appropriate set of training images. Nevertheless, we did not find any published study reporting such results.

2.2. Collaborative filtering

Collaborative filtering is a traditional approach for recommendation systems to produce personal recommendations for music, movies, etc. The mechanism for producing recommendations can be formulated as a matrix factorization problem, where the elements in a sparse rating matrix represent ratings given by different users (represented by the columns of the matrix) to different items, such as movies or songs (represented by the rows of the matrix) [15]. A well-performing recommendation system can predict accurately the missing ratings from the existing ratings, and in this way to produce good recommendations of items for the users.

The conventional formulation of the problem defines a vector of latent item features x_i for item i, and a vector of latent user features y_u for user u, and then predicts the ratings $\hat{r}_{u,i}$ from the dot product of the vectors (1):

$$\hat{r}_{u,i} = x_i^T y_u \quad (1)$$

Different matrix factorization or learning methods, such as stochastic gradient descent or alternating least squares, can be used to approximate the user and item vectors from the sparse user-item rating matrix. If there is side information available, such as movie or music genre, it is possible to achieve higher prediction accuracy by including the side information in the latent feature vectors [16]. It is also possible to use e.g. neural networks to predict the latent features of the content from the textual descriptors or other kinds of high level features to predict the item latent feature vectors [21, 22].

Recently, it has been suggested that the simple inner product of the user and item features may not be sufficient to model the complex interactions between user and item characteristics, and this is why more complex models for collaborative filtering have been proposed. He et al. [23] proposed a combination of general matrix factorization and neural network. Fu et al. [24] proposed a method combining advanced learning model for user and item representations

based on their interactions combined with a deep neural network.

We are not aware of any earlier attempts of using the collaboration filtering technique to predict perceived quality of images by individual users in the prior art. However, some aspects of our work have been considered in the related literature. In [25], collaborative filtering was proposed to predict average ratings in subjective video quality assessment studies more accurately, when some of the ratings in the user-item rating matrix are missing. In [26], a collaborative filtering scheme combining latent features and image features has been proposed for selecting the individually preferred key frames from video. Nevertheless, our work is different from [25, 26] in many respects: the application scenario is different, and also many design choices, such as the use of decision tree based regression models instead of matrix factorization or neural networks, differentiate our work from the prior art.

3. The proposed method

In our work, we use a hybrid approach for predicting the preferred post-processing technique for different images: each image is represented by a feature vector composed by features extracted from the image, whereas each user is represented by a vector of latent features. The basic idea is illustrated in Figure 1: image feature vectors and the user latent feature vectors are combined into input features to the regression model, and the quality ratings for pairs of items and users are used as output of the regression model. When a complete user-item rating matrix is available, it can be divided into training and test sets in order to train and validate the regression model. In this Section, we outline the main characteristics of the proposed method.

3.1. Image features

Since we concatenate image and user features, the popular Convolutional Neural Network (CNN) is not directly applicable for joint feature extraction and regression. In principle, CNN could be used for image feature extraction, but embedding the features from the convolution layer into a dense feature vector that can be used jointly with the user features is not trivial and would require relatively large amount of training data. This is why we have chosen a more conventional approach with hand-crafted image features used as an input to a learning-based regression model. In theory, any kind of regression model could be applicable, but as we will discuss later, only certain regression models seem to perform satisfactorily for the proposed scheme.

To avoid the curses of dimensionality and to ensure that the model can also be used with small datasets, we have targeted at a relatively small number of image features. After experimenting with different alternative sets of

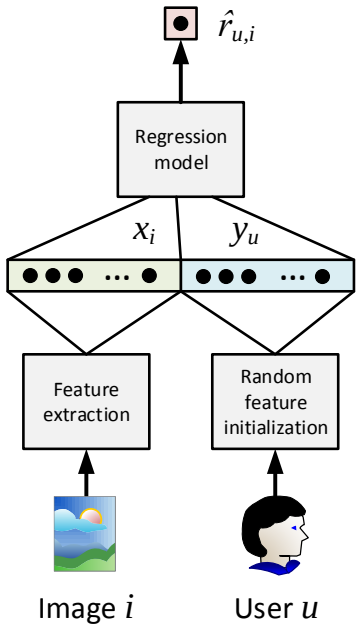

Figure 1: Predicting individual quality ratings from image and user features.

features, we selected twelve different hand-crafted features, representing spatial activity, under- and overexposure, noisiness, sharpness, and contrast and colorfulness. It should be noted that since the target application of our work is quality assessment of capture artifacts and the effects of aesthetic enhancement, rather than assessment of compression artifacts, we have intentionally omitted features that are mainly related to compression, such as blockiness. Due to space constraints, we will only give a brief summary of the features below. For more details, reader may refer to the Matlab script implementing the feature extraction, included in the additional material of the paper submission.

Spatial activity features are based on standard deviation of the pixel intensities after applying the standard Sobel filter. Under- and overexposure features are computed by searching for dark and bright areas with little intensity variation, and using their total coverage and the number of areas as features. Noise density and intensity is estimated by searching for pixels representing local minimum or maximum, i.e. pixels that are either darker or brighter than any other pixel in the surrounding window of 5x5 pixels. Sharpness is estimated by computing two-dimensional autocorrelation function on the pixels in Sobel domain, and then analyzing the slope of the autocorrelation function (sharp images produce a steep slope and blurry images produce a gentle slope). Finally, we have computed a contrast feature and a colorfulness feature by following simple heuristics based on histogram distribution and

saturation of the pixels with the most vivid colors.

3.2. User features

Ideally, the latent user features could be derived from some measurable characteristics of the users, such as age, sex, personality type etc. However, in a typical use case there is not much information available about the users, if any. Therefore, we cannot expect that the user features could be extracted in a similar fashion as the image features. In the latent feature models published in the prior art, feature vectors are typically initialized by using a sparse feature vector with one-hot coding that is then projected into a dense vector [23, 24]. In our method, we have simplified the process by using random initialization as a baseline: for each user, an individual feature vector containing random values between 0 and 1 is assigned. It is assumed that the appropriate length for the user feature vector depends on the used regression method, the length of the item vector, as well as the size of the dataset. This is why we validated our model with different lengths for the user feature vector in the preliminary testing phase.

3.3. Regression

To obtain the predicted quality ratings $\hat{r}_{u,i}$, we can basically use any regression model and train it with a set of user-item pairs with known quality ratings $r_{u,i}$ in the training set, and then validate the model by comparing the predicted ratings produced by the model against the known ratings in the validation set. The feature vectors are simply formed by concatenating the image feature vector and the user feature vector for each image-user pair.

The basic approach for predicting the quality ratings, as described above, is valid if all the quality ratings in the dataset are given in a common scale. In some scenarios, this assumption is however not valid. For example, when different post-processing techniques are compared using pairwise comparison or rank ordering methodology, the ratings reflect the relative quality differences between the different versions of the same image, rather than the absolute quality of the images. In this situation, we cannot expect that a regression model can predict the relative ratings for individual images accurately without any knowledge of the other images used as comparison point.

To predict the relative quality scores for a group of images produced from one source image by applying different post-processing methods, we can concatenate the image features from all the different versions of the image. Then, we can use multioutput regression model to jointly predict the relative quality ratings for the group of images. An example of this kind of scenario is illustrated in Figure 2, using three different post-processing methods, denoted as A, B and C. The relative ratings for the different versions of the image i by the user u are denoted as $r_{u,i,\mathrm{A}}$, $r_{u,i,\mathrm{B}}$, and $r_{u,i,\mathrm{C}}$.

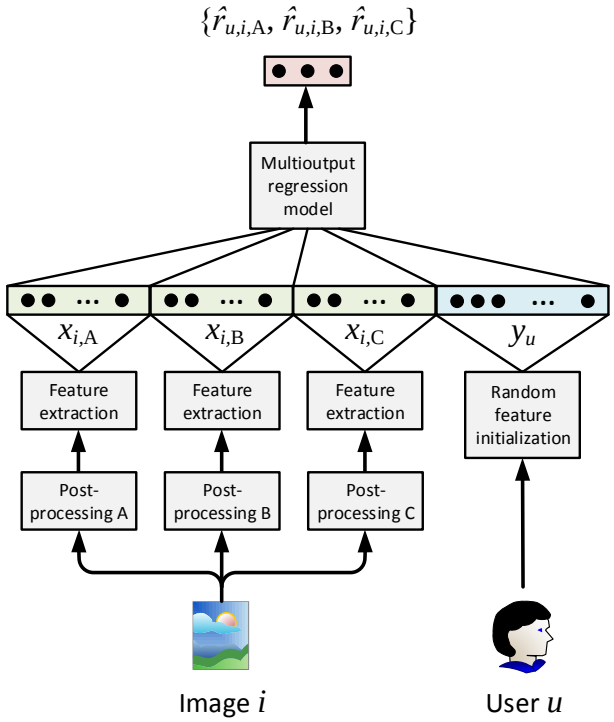

Figure 2: Predicting relative quality ratings for a group of post-processed images.

4. Validation study

To validate the proposed method, we have implemented the proposed image feature extraction in Matlab and the regression schemes in Python. Then, we have tested the method with publicly available image quality datasets. There is abundance of public image datasets annotated with quality scores. However, most of those datasets only report the average ratings, such as Mean Opinion Score (MOS), for each test image. Fortunately, we have been able to identify some public image quality datasets that also include all the ratings by individual users. In this Section, we describe the validation scenarios, methodology and the results in detail. The source code for reproducing the results is published in [27].

4.1. Image quality datasets

Among the few applicable public image quality datasets, we selected two for our experiments, representing two different application scenarios. The first dataset is Camera Image Database 2013 (CID2013) by Virtanen et al. [28]. CID2013 contains actually six different sets of images, evaluated in separate subjective quality assessment studies. Since each set of images has been assessed by a different

group of test subjects, we consider CID2013 database essentially as a collection of six different datasets. In total, CID2013 contains 480 images captured with 79 different devices, each image evaluated using Absolute Category Rating (ACR) with scale from 0 to 100. The images were rated by 26 to 35 subjects, depending on the dataset. The distortions are authentic capture artifacts, and therefore the dataset can be used for assessing different users' sensitivity for camera settings and device related artifacts.

Another dataset, representing the post-processing scenario, is Contrast Enhancement Evaluation Database 2016 (CEED2016) by Qureshi et al. [10, 12, 14, 29]. The dataset contains 180 post-processed images: for each of the 30 different source photos, six different versions were generated by using different post-processing techniques. Then, those different versions were ranked by 23 test subjects by pairwise comparisons of all the 15 pairs formed from each group of six images. The relative quality scores for the images can be computed from the number of preferences: since each image is compared against five other images, the maximum score is five. In contrast to ACR method, this method produces scores that indicates the relative quality in comparison to the other five images, rather than the absolute quality.

4.2. Scenario 1: absolute category ratings

For testing the scheme with ACR scores, we implemented the regression model as shown in Fig. 1, where one image feature vector and one user feature vector is concatenated into a feature vector used as an input to the regression model, and the respective user-item rating $r_{u,i}$ is used as output of the model. In the model, the ratings were normalized to interval 0..1 by dividing each score by 100, and the image features were normalized to the same interval by using the standard min-max normalization with the minimum and maximum values for each feature obtained from the training set. The user feature vectors were initialized with random values between 0 and 1. We tested different lengths for the user feature vectors in the preliminary experiments, and in general, longer vectors seem to give more accurate results; however, there was no essential improvement with vectors more than 20 features, and this is why we fixed the user feature length to 20 for the final experiments. However, longer user feature vectors may be beneficial when larger datasets are concerned.

To find the most appropriate regression models, we tried several different regressors implemented in the scikit-learn toolbox for Python. Surprisingly, Support Vector (SV) and Multilayer Perceptron (MLP) regression, both commonly used in image and video quality assessment [19, 30, 31], produced disappointing results. Better results were obtained with regressors based on decision trees, such as Random Forest (RF) and Gradient Boosting (GB) regression, and

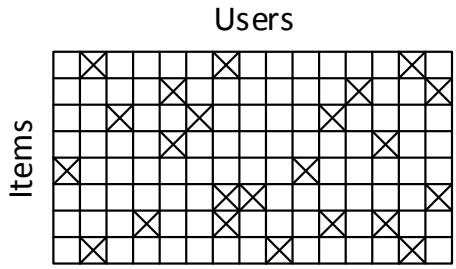

Figure 3: Random split of ratings into training and validation sets illustrated; crosses denote the ratings selected for validation.

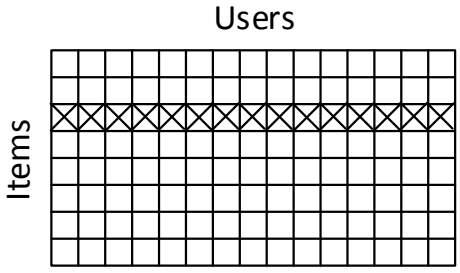

Figure 4: Validation by leaving one item out; crosses denote the ratings used for validation.

this is why we have used on those two methods in the rest of the validation study.

In the first actual validation experiment, we used 100 different random splits into training and validation sets, using different probabilities 10%, 20%, 50% and 80% for allocating each rating to the training set, respectively. Figure 3 shows an example of a possible outcome on a user-item rating matrix of size 8x15, when 80% of ratings are allocated to the training set. As a baseline, we used the average ratings for each image, computed from the ratings for each image available in the training dataset. The more similar preferences the users have, the more accurately the missing ratings can be predicted by using the baseline scheme. In addition, we have also tested a scheme that uses randomly initialized image feature vectors with the same length as the proposed handcrafted feature vector. This scheme can be considered as a generic collaboration filtering technique that is completely agnostic to the actual characteristics of the items and the users. The three schemes are denoted as 'Proposed,' 'Baseline,' and 'RandImFeats' (random image features), respectively. Same seed was used for the random number generator when comparing different schemes, to make sure that the training and validation sets were similar and the comparison was fair.

The performance of each model was measured by computing the Pearson linear Correlation Coefficient (PCC), Spearman rank order Correlation Coefficient (SCC), and Root Mean Squared Error (RMSE) between the

		Metric	Dataset I	Dataset II	Dataset III	Dataset IV	Dataset V	Dataset VI	*Average*
Baseline		PCC	0.792	0.759	0.798	0.764	0.898	0.794	0.801
		SCC	0.786	0.738	0.792	0.741	0.892	0.754	0.784
		RMSE (±std)	18.8 (0.30)	20.2 (0.32)	19.0 (0.35)	17.7 (0.36)	14.0 (0.26)	17.2 (0.26)	17.8
RandImFeats	Gradient Boosting (GB)	PCC	0.802	0.760	0.788	0.823	0.903	0.810	0.814
		SCC	0.798	0.743	0.783	0.802	0.899	0.781	0.801
		RMSE (±std)	18.4 (0.33)	20.2 (0.35)	19.4 (0.38)	15.5 (0.37)	13.7 (0.25)	16.6 (0.31)	17.3
	Random Forest (RF)	PCC	0.812	0.773	0.803	0.825	0.906	0.820	0.823
		SCC	0.807	0.755	0.799	0.805	0.901	0.788	0.809
		RMSE (±std)	17.9 (0.33)	19.7 (0.32)	18.8 (0.36)	15.6 (0.42)	13.5 (0.31)	16.2 (0.28)	17.0
Proposed	Gradient Boosting (GB)	PCC	0.820	0.777	0.803	**0.850**	**0.910**	**0.829**	0.832
		SCC	0.815	0.761	0.799	**0.829**	**0.906**	**0.801**	0.819
		RMSE (±std)	17.6 (0.33)	19.5 (0.37)	18.8 (0.31)	**14.4 (0.32)**	**13.2 (0.25)**	**15.8 (0.25)**	16.5
	Random Forest (RF)	PCC	**0.824**	**0.785**	**0.814**	0.849	**0.910**	0.826	**0.835**
		SCC	**0.819**	**0.768**	**0.809**	**0.829**	**0.906**	0.796	**0.821**
		RMSE (±std)	**17.4 (0.32)**	**19.2 (0.35)**	**18.3 (0.33)**	14.5 (0.39)	**13.2 (0.28)**	15.9 (0.29)	**16.4**

Table 1. Prediction results for quality ratings by individual users in CID2013 database (randomly removed ratings, 50% for training and 50% for validation, average of 100 repetitions). The best results for each dataset are bolded.

		Metric	Dataset I	Dataset II	Dataset III	Dataset IV	Dataset V	Dataset VI	*Average*
MOS-based quality model	Gradient Boosting (GB)	PCC	0.548	0.406	0.360	0.659	0.698	0.634	0.551
		SCC	0.533	0.351	0.293	0.632	0.672	0.571	0.509
		RMSE	26.0	28.8	30.0	20.6	22.8	21.9	25.0
	Random Forest (RF)	PCC	0.546	0.441	0.410	0.667	**0.730**	0.566	0.560
		SCC	0.526	0.367	0.308	0.626	**0.727**	0.511	0.511
		RMSE	25.8	27.8	28.9	20.6	22.3	23.4	24.8
Proposed	Gradient Boosting (GB)	PCC	**0.584**	**0.486**	0.417	**0.772**	0.720	0.650	**0.605**
		SCC	**0.581**	**0.451**	0.378	**0.744**	0.715	0.614	**0.581**
		RMSE	**25.4**	**27.6**	29.2	**17.3**	**22.0**	21.5	**23.8**
	Random Forest (RF)	PCC	0.574	0.421	**0.453**	0.756	0.680	**0.661**	0.591
		SCC	0.569	0.406	**0.381**	0.738	0.684	**0.632**	0.568
		RMSE	25.5	29.0	**28.5**	18.0	23.6	**21.3**	24.3

Table 2. Prediction results for quality ratings by individual users in CID2013 database (leave-one-out validation excluding each image one by one). The best results for each dataset are bolded.

predicted and true ratings in the validation dataset. We have reported the average for each performance metric in Table 1 for all the six datasets in CID2013 database separately, using 50:50 split to training and validation sets. In Figure 5, we illustrate how the average performance changes along the proportion of ratings allocated in the training set. For clarity, we have only included the results for the proposed and RandImFeats schemes using RF regressor, but the results with GB regression were essentially similar. The plots show that the proposed scheme clearly outperforms both baseline and RandImFeats regardless of the training set size. RandImFeats works well when most of the ratings are available for training the model, but it performs even worse than the baseline when the user-item rating matrix used for training is sparse (i.e. less than 20% of ratings for training).

As the results in Table 1 indicate, the proposed method outperforms the baseline method on every dataset, according to any criterion. The RandImFeats method with randomly initialized image feature vectors tends to outperform baseline, and in some cases, its performance is on par with the proposed scheme using image features. There are some apparent differences between the datasets: for datasets III and V, collaborative filtering improves the performance only slightly when compared to the baseline, whereas the improvement is much larger for the datasets I, II, IV and VI. The result suggest that there are less systematic differences between the users' rating behavior for the datasets where the performance difference between

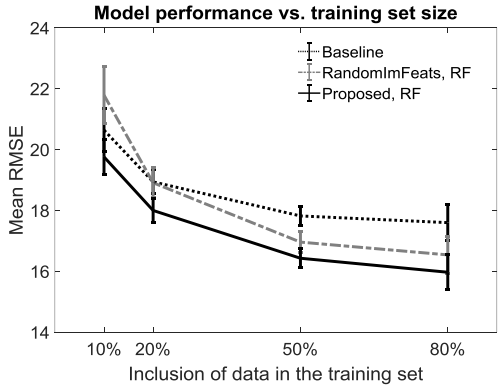

Figure 5. Model performance versus proportion of ratings included in the training set, in terms of average RMSE.

baseline and collaborative filtering is small (i.e. datasets III and V). The reasons for those differences probably lie in the differences in methodology as well as the content in the datasets: for example, users were instructed to use the extremes of the rating scale for the best and the worst image in test sets I-III, but not for the test sets IV-VI [28].

The well-known disadvantage of collaborative filtering with latent features is that it cannot handle previously unknown items, commonly known as a *cold start* problem in recommender systems. In the second validation experiment, we aimed to demonstrate that the proposed scheme is also useful in scenarios where a new content is introduced without prior knowledge of its ratings. For this experiment, we used leave-one-out validation, where ratings for one image were used for validation and all the other ratings were used for training, as illustrated in Figure 4. The process was repeated to all the images, and the results were aggregated. In this scenario, we implemented also a baseline MOS-based quality model that uses the same regression model to predict the average rating (MOS) for each image, and assigns the same predicted rating to all the users. This is equivalent to the conventional learning-based quality modeling task, where only MOS is concerned. Due to the cold start, randomly initialized image feature vectors would not work in this scenario, and this is why we have omitted 'RandImFeats' method in this experiment.

The results for the second experiment are summarized in Table 2. Also in this experiment, the proposed approach shows an improvement in the overall results, when compared against the baseline quality model. There improvement in the average results is not very large, but the proposed approach seems to have a significant advantage with the dataset IV in particular. On the other hand, the baseline model works well with the dataset V, indicating a high agreement between the ratings by different users. Again, we assume that there are differences in methodology as well as content, which makes the users' rating behavior on those datasets less predictable than on the other datasets. In the average results, GB seems to work slightly better than RF, but the difference is not essential.

We assume that there is room for improvement with the used image features and the regression models. However, our results imply that the collaborative filtering approach combined with image features is indeed capable for predicting the quality ratings by individual users more accurately than just using the average ratings by the other users. This approach would be useful for maximizing the personal visual quality by applying individually chosen camera settings or post-processing algorithms.

4.3. Scenario 2: relative ratings

The main difference between the first scenario and the second scenario is that each user-item pair in CEED2016 database consists of a user and a group of six images and their relative ratings, rather than a user and only one image. Therefore, we have used an image feature vector formed by concatenating the six image feature vectors extracted from a group of six differently processed version of the same source image, and a user feature vector generated in a similar fashion as in the first scenario. Then, we have used a multioutput regression model to predict the relative quality ratings jointly for the group of images. The process is conceptually similar to the depiction in Figure 2. For the

		Metric	10% training	20% training	50% training	80% training	*Average*
Baseline		PCC (±std)	0.664 (0.019)	0.709 (0.014)	0.751 (0.009)	0.763 (0.017)	0.722
		SCC (±std)	0.646 (0.020)	0.689 (0.014)	0.729 (0.010)	0.741 (0.019)	0.701
		RMSE (±std)	0.245 (0.007)	0.224 (0.006)	0.206 (0.004)	0.202 (0.006)	0.219
		PAPI (±std)	0.460 (0.041)	0.465 (0.024)	0.479 (0.023)	0.486 (0.037)	0.473
RandImFeats	Gradient Boosting (GB)	PCC (±std)	0.667 (0.018)	0.715 (0.013)	0.767 (0.009)	0.786 (0.016)	0.734
		SCC (±std)	0.654 (0.018)	0.703 (0.013)	0.756 (0.010)	0.777 (0.017)	0.723
		RMSE (±std)	0.242 (0.007)	0.223 (0.005)	0.203 (0.004)	0.194 (0.007)	0.216
		PAPI (±std)	0.452 (0.026)	0.494 (0.022)	0.544 (0.027)	0.563 (0.036)	0.513
	Random Forest (RF)	PCC (±std)	0.720 (0.012)	0.753 (0.010)	0.792 (0.009)	0.807 (0.014)	0.768
		SCC (±std)	0.705 (0.013)	0.740 (0.010)	0.782 (0.010)	0.797 (0.016)	0.756
		RMSE (±std)	0.217 (0.004)	0.205 (0.003)	0.190 (0.004)	0.184 (0.006)	0.199
		PAPI (±std)	0.466 (0.025)	0.508 (0.022)	0.563 (0.023)	0.586 (0.040)	0.531
Proposed	Gradient Boosting (GB)	PCC (±std)	0.674 (0.016)	0.720 (0.012)	0.771 (0.009)	0.791 (0.014)	0.739
		SCC (±std)	0.662 (0.016)	0.710 (0.013)	0.761 (0.009)	0.781 (0.016)	0.729
		RMSE (±std)	0.240 (0.006)	0.221 (0.005)	0.201 (0.004)	0.192 (0.006)	0.214
		PAPI (±std)	0.460 (0.025)	0.502 (0.023)	0.544 (0.024)	0.565 (0.040)	0.518
	Random Forest (RF)	PCC (±std)	**0.729 (0.011)**	**0.760 (0.008)**	**0.796 (0.009)**	**0.810 (0.014)**	**0.774**
		SCC (±std)	**0.714 (0.011)**	**0.748 (0.009)**	**0.786 (0.009)**	**0.801 (0.015)**	**0.762**
		RMSE (±std)	**0.214 (0.004)**	**0.203 (0.003)**	**0.189 (0.004)**	**0.183 (0.006)**	**0.197**
		PAPI (±std)	**0.478 (0.023)**	**0.518 (0.019)**	**0.570 (0.023)**	**0.591 (0.040)**	**0.539**

Table 3. Prediction results for quality ratings by individual users in CEED2016 database (randomly removed ratings, 100 repetitions). The best results for each experiment with 10%, 20%, 50% and 80% of ratings in the training set are bolded.

validation study, we have followed a similar approach as in the first scenario, with certain exceptions.

As in the first scenario, we first attempted several different regressors and user vector lengths. Our observations were in line with the first scenario: regressors based on decision trees (RF and GB) tend to perform better than SV or MLP, and this is why we have run the full scale experiments using RF and GB only. In addition to PCC, SCC and RMSE, we defined one more performance criterion: the accuracy for predicting the most preferred version among the differently processed images, denoted as Prediction Accuracy for the Preferred Item (PAPI). This criterion is essential for applications trying to guess the post-processing technique preferred by a specific user.

The first validation experiment for the second scenario was similar as for the first scenario: we split the user-item pairs (one user and six images) randomly into training and validation sets using the same probabilities (0.8 for training and 0.2 for validation), and repeated the process 100 times with different splits. The user feature vectors of 20 latent features for each user were randomly initialized, just as in the first scenario. For the baseline scheme, we used the relative ratings of each source image in the training set to compute the expected relative ratings for the respective source images in the validation set.

The results for the first experiment are listed in Table 3. As the results show, both RandImFeats and the proposed scheme outperform the baseline scheme with all the tested proportions of ratings in the training set (10%, 20%, 50%, and 80%). According to the average results, the best performing method can improve the likelihood of predicting the most preferred image version from 47% of the baseline up to 54%. The proposed method with RF regression performs the best, but only with a slight margin to the RandImFeats scheme with RF method. Indeed, it seems that in this scenario, randomly initialized item feature vectors can discriminate different contents nearly as well as the features extracted from the content. RF regression outperforms GB regression clearly in the overall results.

Finally, we tested the method with the leave-one-out validation scheme in a similar fashion as in the first scenario. We assume that every user tends to prefer similar post-processing techniques regardless of the image content, and therefore we implemented a baseline scheme that computes the average ratings for different post-processing techniques given by each user in the training set. The results comparing the baseline and the proposed model with RF regression are summarized in Table 4.

As we can see from the results, the proposed scheme improves the accuracy of predicting the personally preferred post-processing techniques in comparison to the baseline, but the improvement is modest. Closer analysis reveals that there is a clear improvement for most of the images, but for some outlier images the baseline is better. Apparently, the dataset is too small (30 original images) to be trained to differentiate image content types accurately.

	Metric	Result
Baseline	PCC	0.685
	SCC	0.677
	RMSE	0.227
	PAPI	0.507
Proposed (RF regressor)	PCC	**0.707**
	SCC	**0.701**
	RMSE	**0.222**
	PAPI	**0.529**

Table 4. Prediction results for quality ratings by individual users in CEED2016 database (leave-one-out validation excluding each image one by one). The best results are bolded.

We are aware of image quality models employing more advanced features than in our work (e.g. [19]), and we are also aware of more sophisticated deep approaches for collaborative filtering (e.g. [23, 24]). Therefore, we believe that the method could be substantially improved by combining different techniques known in the prior art. Unfortunately, deep methods typically require large amount of training data, and this study is constrained by the limited sizes of the public image quality databases with individual user ratings available. We expect that larger user-centric image quality databases will be available in the future.

5. Conclusions

In this paper, we have proposed a method for predicting personal opinions or preferences on image quality. The proposed method is a hybrid scheme combining learning-based quality modeling based on a bag of image features, and collaborative filtering exploiting the available ratings in user-item rating matrix. Prediction of personal image quality ratings has a lot of interesting applications, such as automatic adjustment of camera settings or post-processing parameters for different images to match different users' personal taste, as well as displaying personalized versions of images on networked applications, such as content sharing platforms. The feasibility of the proposed method is demonstrated via experiments on public image quality databases. The results show that it is indeed possible to predict personally perceived image quality more accurately by exploiting the image features rather than using the plain content agnostic collaborative filtering. However, experiments with larger datasets would be needed to demonstrate the full potential of the proposed method.

Acknowledgements

This work was supported in part by the National Science Foundation of China under Grant 61772348.

References

[1] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering, in IEEE Trans. Image Process.: 16(8), 2080–95, Aug. 2007.

[2] G. Papari, N. Petkov and P. Campisi, Artistic Edge and Corner Enhancing Smoothing, in IEEE Trans. Image Process.: 16(10), 2449-62, Oct. 2007.

[3] H. C. Burger, C. J. Schuler, and S. Harmeling, Image Denoising: Can Plain Neural Networks Compete with BM3D? in *Proc. CVPR'12*, Providence, RI, USA, Jun. 2012.

[4] Z. Chen, T. Jiang, and Y. Tian, Quality Assessment for Comparing Image Enhancement Algorithms, in *Proc. CVPR'14*, 2014.

[5] Y. Chen, W. Yu, and T. Pock, On Learning Optimized Reaction Diffusion Processes for Effective Image Restoration, in *Proc. CVPR'15*, Boston, MA, USA, Jun. 2015.

[6] K. Gu, and G. Zhai, The Analysis of Image Contrast: From Quality Assessment to Automatic Enhancement, in IEEE Trans. Cybernetics: 46(1), 284-297, Jan. 2016.

[7] S. Wang, K. Ma, H. Yeganeh, Z. Wang, and W. Lin, Patch-structure Representation Method for Quality Assessment of Contrast Changed Images, in Signal Process. Letters: 22(12), Dec. 2015.

[8] L. Krasula, P. Le Callet, K. Fliegel, and M. Klima, Quality Assessment of Sharpened Images: Challenges, Methodology, and Objective Metrics, in IEEE Trans. Image Proc.: 26(3), 1496–1508, Mar. 2017.

[9] R. Hassen, Z. Wang, M. M. A. Salama, Image Sharpness Assessment based on Local Phase Coherence, in IEEE Trans. Image Process.: 22(7), 2798-2810, Jul. 2013.

[10] M. A. Qureshi, A. Beghdadi, B. Sdiri, M. Deriche, F. A. Cheikh, A Comprehensive Performance Evaluation of Objective Quality Metrics For Contrast Enhancement Techniques, in *Proc. EUVIP'16*, pp. 1-5, Marseille, France, Oct. 2016.

[11] L. Li, W. Xia, W. Lin, Y. Fang, and S. Wang, No-Reference and Robust Image Sharpness Evaluation based on Multiscale Spatial and Spectral Features, in IEEE Trans. Multimedia: 19(5), 1030-40, May 2017.

[12] M. A. Qureshi, A. Beghdadi, and M. Deriche, Towards the Design of a Consistent Image Contrast Enhancement Evaluation Measure, Signal Processing: Image Communication: 58(10), 212-227, Oct. 2017.

[13] K. Egiazarian, M. Ponomarenko, V. Lukin, and O. Ieremeiev, Statistical Evaluation of Visual Quality Metrics for Image Denoising, in *Proc. ICASSP'18*, Calgary, Canada, Apr. 2018.

[14] A. Beghdadi, M. A. Qureshi, and M. Deriche, A Critical Look to Some Contrast Enhancement Evaluation Measures, in *Proc. CVCS'15*, Gjoevik, Norway, Aug. 2015.

[15] Y. Koren, R. Bell, and C. Volinsky, Matrix Factorization Techniques for Recommender Systems, in IEEE Computer: 48(8), 30-37, Aug. 2009.

[16] Y. Shi, M. Larson, and A. Hanjalic, Collaborative Filtering beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges, in ACM Computing Surveys: 47(1), Jul. 2014.

[17] W.-T. Sun, T.-H. Chao, Y.-H. Kuo, and W. H. Hsu, Photo Filter Recommendation by Category-Aware Aesthetic Learning, in IEEE Trans. Multimedia: 19(8), 1870-1880, Aug. 2017.

[18] L. Bie, X. Wang, and J. Korhonen, Subjective Assessment of Post-processing Methods for Low Light Consumer Photos, in *Proc. QoMEX*, Sardinia, Italy, Jun. 2018.

[19] A. Mittal, R. Soundararajan, and A. C. Bovik, Making a Completely Blind Image Quality Analyzer, in IEEE Signal Process. Lett.: 22(3), 209–212, Mar. 2013.

[20] D. Ghadiyaram, and A. C. Bovik, Perceptual Quality Prediction on Authentically Distorted Images Using a Bag of Features Approach, in J. of Vision: 17(1), Jan. 2017.

[21] H. Wang, N. Wang, and D.-Y. Yeung, Collaborative Deep Learning for Recommender Systems, in *Proc. KDD'15*, Sydney, Australia, Aug. 2015.

[22] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, Collaborative Knowledge Base Embedding for Recommender Systems, in *Proc. KDD'16*, San Francisco, CA, USA, Aug. 2016.

[23] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, Neural Collaborative Filtering, in *Proc. WWW'17*, Perth, Australia, Apr. 2017.

[24] M. Fu, H. Qu, Z. Yi, L. Lu, and Y. Liu, A Novel Deep Learning-Based Collaborative Filtering Model for Recommendation System, in IEEE Trans. Cybernetics: early access version, Jan. 2018.

[25] J. Korhonen, Predicting Personal Preferences in Subjective Video Quality Assessment, in *Proc. QoMEX'17*, Erfurt, Germany, May 2017.

[26] X. Chen, Y. Zhang, Q. Ai, H. Xu, J. Yan, and Z. Qin, Personalized Key Frame Recommendation, in *Proc. SIGIR'17*, Tokyo, Japan, Aug. 2017.

[27] J. Korhonen, Assessment of Personally Preferred Image Quality, URL: https://github.com/jarikorhonen/personal_image_preferences, Mar. 2019.

[28] T. Virtanen, M. Nuutinen, M. Vaahteranoksa, P. Oittinen, and J. Häkkinen, CID2013: A Database for Evaluating No-reference Image Quality Assessment Algorithms, in IEEE Trans. Image Process.: 24(1), 390-402, Jan. 2015.

[29] M. A. Qureshi, B. Sdiri, M. Deriche, F. Alaya-Cheikh, A. Beghdadi, Contrast Enhancement Evaluation Database (CEED2016), Mendeley Data, v3. DOI: 10.17632/3hfzp6vwkm.3.

[30] M. A. Saad, A. C. Bovik, Blind Prediction of Natural Video Quality, in IEEE Trans. Image Proc.: 23(3), 1352-65, Mar. 2014.

[31] S. Bosse, D. Maniry, T. Wiegand, and W. Samek, A Deep Neural Network for Image Quality Assessment, in *Proc. ICIP'16*, Phoenix, AZ, USA, Sep. 2016.