Predicting Personal Preferences in Subjective Video Quality Assessment

Jari Korhonen
Department of Photonics Engineering (DTU Fotonik)
Technical University of Denmark (DTU)
Kgs. Lyngby, Denmark

***Abstract*—In this paper, we study the problem of predicting the visual quality of a specific test sample (e.g. a video clip) experienced by a specific user, based on the ratings by other users for the same sample and the same user for other samples. A simple linear model and algorithm is presented, where the characteristics of each test sample are represented by a set of parameters, and the individual preferences are represented by weights for the parameters. According to the validation experiment performed on public visual quality databases annotated with raw individual scores, the proposed model can predict the scores by individuals more accurately than the average score for the respective sample computed from the scores given by other individuals. In many cases, the proposed algorithm also outperforms more generic Parametric Probabilistic Matrix Factorization (PPMF) technique developed for collaborative filtering in recommendation systems.**

Keywords—subjective quality assessment; individual characteristics, collaborative filtering

I. INTRODUCTION

One of the fundamental challenges in visual quality assessment is to develop algorithms and mathematical models that can predict the subjectively perceived quality of the content by analyzing the digital representation of the content directly. Such models are usually referred as objective quality metrics. The use of objective quality metrics is important for saving the time and resources required for subjective quality assessment, i.e. test campaigns where real people (test subjects) assess the quality. However, subjective quality assessment is still of great importance, because the results of subjective studies are needed as a ground truth when objective metrics are developed and verified. The goodness of an objective metric is typically evaluated by computing the correlation coefficients between the objective quality ratings and the subjective quality ratings (ground truth).

Traditionally, Mean Opinion Score (MOS) has been used as a measure of subjective quality. In its simplest form, MOS is computed as an average of the individual quality scores given by different test subjects to the same test image or video sequence. The fundamental problem of MOS is that different people may use the rating scale differently, depending on personal factors, such as tendency to emotional involvement, preferences regarding the content, and level of expertise for quality rating tasks [1]. In addition, several external factors may influence rating, e.g. viewing order, lighting conditions etc. These problems can be alleviated by careful test design, diverse selection of test material, randomized viewing order, removal of outliers, and using sufficiently large number of test subjects [2]. With these precautions, MOS can be considered as a relatively reliable measure of the overall subjective quality.

Unfortunately, MOS as a quality measure does not consider differences in individual preferences. Test subjects who give constantly ratings deviating significantly from the average, will usually be considered as outliers and removed from the data. This may be a reasonable approach, if the intention is to find one MOS rating that describes the subjective quality of the test sample as it is perceived by an average person. On the other hand, there are several applications where it may be desirable to predict the quality score separately for each individual user. For example, for choosing a tone mapping operator, there is typically a trade-off between high contrast and naturalness of the image, and different persons may have different preferences between those features. A personal quality score estimate would help to choose the tone mapping operator that fits to the taste of each individual. Another possible application is to use predicted values to fill in missing scores; this would reduce the workload in subjective quality assessment campaigns, because total MOS scores could be predicted more accurately from an incomplete set of ratings.

In this paper, we propose a simple linear quality model, where the sample is represented by a set of parameters, and the user preferences are represented by weights for each parameter. The quality estimate can be computed as a sum of parameters for the sample, each multiplied by the respective weight of the individual person. We propose a simple method for computing the weights and the parameters that represent latent factors, i.e. they are agnostic to any qualitative features of the content. The weights and parameters can be computed in a continuous manner: each time a new subjective rating is given, the values for the rated sample and the weights for the test subject concerned are updated. We have tested the proposed method using several different annotated video quality databases, and the results show that the proposed method can predict the quality scores more accurately than the MOS computed from the ratings by other individuals on the same sample, and the performance is on par with prior art in matrix factorization.

QoMEX2017 – Erfurt, Germany; 978-1-5386-4024-1/17/$31.00 © 2017 IEEE

II. Background and Related Research

Vast majority of the prior art in Quality of Experience (QoE) research is focused on development of objective metrics that would maximize the correlation coefficients between the objective quality scores and the respective subjective MOS scores. Until relatively recently, there were only few studies concerning the individual characteristics in visual quality assessment. Most of those studies focus on measuring some personal characteristics, for example by using a questionnaire, and then analyzing the correlations between quality ratings, features of the test samples, and the personal characteristics [1]. Some objective metrics including personal features have been proposed: for example, Rodríguez et al. [3] have proposed a metric where explicitly given content preference is taken into account for improving the accuracy of the predicted MOS.

In the related literature, there are also some studies where the content characteristics, such as spatial and temporal activity, are used to improve accuracy of objective quality assessment [4, 5]. However, these studies still aim at predicting MOS, not the personal scores for different users. Unfortunately, assessment of the personal preferences and characteristics *a priori* require some kind of user involvement, such as taking a personality test, which may not be possible in real-life applications, since users are often reluctant to spend time on additional tasks they do not benefit directly. More attractive alternative is to use user's prior rating behavior to find biases towards certain features of the content.

At the time of writing, we are not aware of any prior studies where personal rating history is used to predict personal scores in visual quality assessment tasks. However, several such studies have been made concerning recommendation systems that aim predicting e.g. ratings of movies [6], referred as collaborative filtering. A typical model for such predictions is based on a joint latent factor base, where the user factors and the item factor interactions are modeled so that the predicted rating $\hat{r}_{u,i}$ as a dot product of the item factor vector x_i and the sample factor vector y_u:

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

The challenge of such a model is to find a good mapping of each user and item into factor vectors so that the actual ratings can be predicted as accurately as possible. With a several earlier ratings, the optimal values can be solved by factorizing the user-item rating matrix; however, the sparseness of the matrix can limit feasibility of this approach, and learning algorithms, such as stochastic gradient descent and alternating least squares can be used to approximate the vectors [6]. By adding side information, such as movie genre, more accurate predictions can be achieved [7]. For applications such as movie ratings, temporal dynamics should also be considered, since the popularity of evaluated items tends to change, and also users' inclinations evolve [6]. On the other hand, we can assume that the preferences for technical quality do not evolve in a similar fashion as preferences for consumer products, and therefore we have omitted temporal dynamics in this paper.

Our approach is based on the latent factor model described above; however, the basic approach has been modified to suit better to the peculiarities of visual quality assessment tasks. Our first attempts with gradient descent and alternating least squares did not work well on the test data, since they seemed very sensitive to the model parameters. Therefore, we have developed our own approach for initializing new factor vectors and learning their values along new ratings.

III. Proposed Model

In the proposed model, we have adopted the generic latent factor model for collaborative filtering [6]; however, we have designed our own initialization mechanism and learning algorithm that are suitable for visual quality assessment data. For testing, we have implemented the proposed model in Matlab. The predicted quality score $\hat{q}_{u,s}$ by user u for test sample s is computed from Eq. (2):

$$\hat{q}_{u,s} = \sum_{i=1}^{n} w_{u,i}\alpha_{s,i} \,, \tag{2}$$

where n is the number of parameters in the model, $w_{u,i}$ is the user's weight for each parameter i, and $\alpha_{s,i}$ is the parameter i defining characteristics of the sample (in vector notation, $W_u = \{ w_{u,1},\ldots, w_{u,n} \}$ and $A_s = \{ \alpha_{s,1},\ldots, \alpha_{s,n} \}$). Note that the model parameters $\alpha_{s,i}$ are agnostic to the actual qualitative characterization of the sample. It may be possible to observe correlations between measurable characteristics (such as contrast or noise) and the parameters; however, such analysis is left for the future work.

For the sake of simplicity, we assume that the quality scores are normalized to interval [-1, 1]. Due to the nature of visual quality assessment tasks, where the subjective quality is influenced primarily by relatively restricted technical attributes such as blurriness or contrast, we may assume that parameters characterizing the sample may have either positive or negative impact on the quality; only their relative weights will be different for different users. Therefore, parameters $\alpha_{s,i}$ can get both negative and positive values, but the user weights $w_{u,i}$ are always positive and their sum for each user is always 1:

$$\forall u : \sum_{i=1}^{n} w_{u,i} = 1 \tag{3}$$

The main challenge with the proposed model is to find a method to derive the appropriate values for the parameters and the weights with incomplete training data, which can be reduced to the classical sparse matrix factorization problem [6]. However, our aim is to develop a memoryless method, that updates the parameters and weights for the concerned sample and user each time a new rating is added. This approach is feasible even if the number of ratings for different samples is uneven.

When a new sample is added to the pool of samples, its parameters are initialized to represent the range [-L, L] evenly:

$$\alpha_{s,i} := -L + \frac{2L(i-1)}{n-1}, \tag{4}$$

where i is an integer between 1 and n. To avoid scores computed from (2) to fall outside the target range [-1, 1], we should choose L=1. However, we have observed that more accurate model can be created, if we use larger L; in this case, the final predicted scores must be clipped so that values smaller than -1 are set to -1, and values larger than 1 are set to 1. By trial and error, we have chosen a compromise value L=1.5 in our test implementation.

When the first user u is added to the pool, the weights are initialized to be even: for every i=1..n, $w_{u,i}$=1/n. Then, when a new user is added, weights are initialized to the weighted average of the existing users that have rated at least one sample (N_x is the number of scores given by user x, and U is the set of indices for users who have given at least one score):

$$w_{u,i} := \left(\sum_{x \in U} w_{x,i} N_x \right) \Big/ \sum_{x \in U} N_x , i = 1..n \qquad (5)$$

If there is some *a priori* knowledge of the distribution of the scores, this information could be used for more accurate initialization of the parameters and weights. However, in this paper we assume that the distribution of the rates (both in training and test data) is relatively even, and further exploration of initialization strategies is left to the future work.

Even with the conditions listed above, there are several alternatives to search for parameters and weights that would predict the quality score as accurately as possible. In this work, we have chosen an iterative approach: parameters and weights are updated continuously every time a new rating is added to the pool of ratings. This would be a useful feature e.g. in applications where users rate test samples, such as video clips, over the Internet, according to their own schedule. As the number of ratings increase, the parameters and weights will converge towards the optimum.

To compute the new weights after a new score is given, the parameters are divided in two groups: those that are below and above the new score. Parameters equal to the new score are handled as a special case. New weights W_u' for the parameters below and above the target value are computed by changing the old weights W_u so that Equation (2) will give the new given score as a result. The algorithm (1) for computing the weights is written in pseudocode below.

algorithm (1): W_u' = compute_weights (input s,u,$q_{s,u}$)

$A := A_s;\ W := W_u;\ q := q_{s,u};\ w'_i := 0, i = 1..n$
$\forall A_{below} \subseteq A : \alpha_i < q$
$\forall A_{equal} \subseteq A : \alpha_i = q$
$\forall A_{above} \subseteq A : \alpha_i > q$

if ($A_{below} = \emptyset$ **OR** $A_{above} = \emptyset$)
  if $A_{equal} \neq \emptyset$
    $\forall \alpha_i \in A_{equal} : w'_i := 1/|A_{equal}|$
  else
    $w'_i := 1/n, i = 1..n$
  endif
else

$$p_{below} := \frac{q - \left(\sum_{\alpha_i \in A_{above}} \alpha_i \right) / |A_{above}|}{\sum_{\alpha_i \in A_{below}} \alpha_i - \left(|A_{below}| \sum_{\alpha_i \in A_{above}} \alpha_i \right) / |A_{above}|}$$

$$p_{above} := \left(1 - p_b |A_{below}|\right) / |A_{above}|$$

  if $A_{equal} \neq \emptyset$
    $\forall \alpha_i \in A_{below} : w'_i := 0.5 p_{below}$
    $\forall \alpha_i \in A_{equal} : w'_i = 0.5 / |A_{equal}|$
    $\forall \alpha_i \in A_{above} : w'_i := 0.5 p_{above}$
  else
    $\forall \alpha_i \in A_{below} : w'_i := p_{below}$
    $\forall \alpha_i \in A_{above} : w'_i := p_{above}$
  endif
endif

Respectively, to compute the new parameters A_s', the total difference D_{total} between the given score and the predicted score is first computed, using the old parameters A_s. Then, the weights are used to determine the proportional importance of each parameter; for this purpose, an auxiliary variable d_i is used so that the value of d_i is higher for those parameters that have higher weight; in our test implementation, d_i is derived straightforwardly from $w_{u,i}$. Then, the parameters are changed so that the total difference between the given score and the predicted score is covered. The algorithm (2) for computing the new parameters is written in pseudocode below.

algorithm (2): A_s' = compute_parameters (input s,u,$q_{s,u}$)

$$D_{total} := q_{s,u} - \sum_{i=1..n} w_{u,i} \alpha_{s,i}$$

if $D_{total} = 0$
  $A'_s := A_s$
  return
endif

for i:=1 **to** n
  $d_i := w_{u,i}^{1.5}$
endfor

for i:=1 **to** n
  if $w_{u,i} > 0$

$$\alpha'_{s,i} := \alpha_{s,i} + \frac{D_{total} d_i}{w_{u,i} \cdot \sum_{j=1..n} d_j}$$

  else
    $\alpha'_{s,i} := \alpha_{s,i}$
  endif
endfor

After using algorithms (1) and (2) to obtain W_u' and A_s', the actual updated weights and parameters are computed as weighted arithmetic mean of the old weights and parameters, and the new weights and parameters. For this purpose, the system keeps track on the number of scores user u has given, N_u, as well as the number of scores, M_s, that have been given to

the sample s (initialization counts as a first scoring). The updated weights and parameters are then computed as in Eqs. (6) and (7).

$$W_u := \left([N_u - 1] \cdot W_u + W_u^{'} \right) / N_u \quad (6)$$

$$A_s := \left([M_s - 1] \cdot A_s + A_s^{'} \right) / M_s \quad (7)$$

The process of updating the weights and the parameters is repeated every time a new score is given by a user. If all the scores are readily available, they can be sent to the algorithm sequentially.

IV. Validation Experiments

A. Test Data

Unfortunately, many of the public video quality assessment databases do not include ratings from individual users, but only MOS or Differential MOS (DMOS) values for each test sample (possibly with the standard deviations of the scores). However, we have identified a few databases with individual ratings that are appropriate to use for testing the proposed method. For our validation study, we have selected the classical EPFL-PoliMi database [8-10], VQEG HDTV Pool 2 database by University of Nantes [11-12], HDTV (25fps) database by Technical University of Munich (TUM) [13-14], and more recent Camera Video Database (CVD) from the University of Helsinki [15-16]. Characteristics of the used databases are listed in Table I.

TABLE I. Databases Used for Validation

	EPFL-P. [9]	**TUM [13]**	**VQEG [11]**	**CVD [15]**
Users	40	18	24	30+30+28+33+30 +32
Videos	78	48	168	27+30+39+42+48 +48
Scenes	6	4	9	5
Resolution	704x576	1920x 1080	1920x 1080	640x480 1280x720
Distortions	Packet loss	Coding	Coding, channel errors	Different cameras used
Score scale	Contin. 0-5	Discrete 0-10	Discrete 1-5	Continuous 0-100

It should be noted that CVD database contains data from six individual experiments with different video clips and test subjects. Therefore, each experiment has to be studied separately. In addition, for the first two tests in CVD database, each user has assessed each test video two times. For those two tests, we have used average of the two scores as a final score for each video. CVD database is also different from typical quality assessment database, since the artifacts are produced by capturing video clips with different cameras, and in the experiments, also additional data is collected [15]. In the EPFL-PoliMi, TUM and VQEG HDTV databases, there are source contents that are processed for different Hypothetical Reference Circuits (HRC) to produce compression and transmission noise. Therefore, EPFL-PoliMi, TUM, and VQEG HDTV datasets represent more typical quality assessment experiments.

B. Test Procedure

First, the database is split into training and test data, which can be done in several different ways. In order to avoid biased results, we have balanced the number of users rating each sample and number ratings for each sample in the training and test sets. The database can be defined as a user-sample matrix, where each column represents a user, each row represents a sample, and the score given by user u to sample s is given by the element $q_{u,s}$. Let us assume that train is a set that contains the indices of the training set, and test is a set that contains the indices of the test set. Fig. 1 shows the most straightforward way to split the ratings of four samples by six users into training and test sets, following a chessboard type of pattern. The test sets are generated respectively, using the remaining ratings.

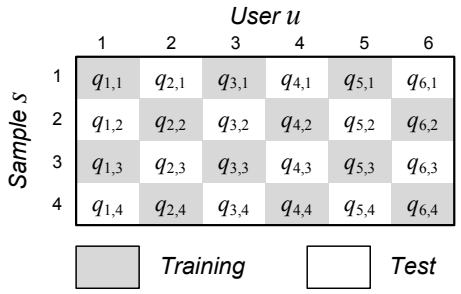

Fig. 1. Example of splitting data in training and test sets evenly.

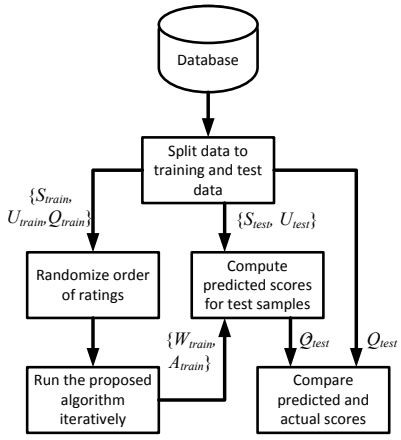

Fig. 2. Procedure for testing the method.

Now, the training data $\{S_{train}, U_{train}, Q_{train}\}$ can be used to obtain the parameters A_s and weights W_u by feeding the training data to the proposed algorithm, one by one. To make sure that the algorithm converges regardless of the order, the order of feeding the ratings to the iterative algorithm can first be randomized. Assuming that all the contents and users are represented in the training set, we can then use the obtained parameters and weights to compute the predicted quality scores $\hat{Q}_{test}$ for the test set, using the sample and user indices S_{test} and U_{test}. Since the database contains also the actual scores Q_{test}, the prediction accuracy can be computed. The procedure is outlined in the flowchart in Fig 2.

C. Results

To assess the performance of our method, we have used two benchmark methods: MOS for each sample computed directly from the training set, and PPMF proposed by Shan and Banerjee [17] (for PPMF, we have used the Matlab implementation provided in the Internet by the authors [18]). Even though we are aware that there are more recent advances in matrix factorization for recommendation systems, most of the recent contributions rely on side information (e.g. the movie genre), and therefore we consider PPMF as a well representative method for matrix factorization in recommendation systems.

In the following experiments, we used half-and-half split into training and test sets, following the balanced pattern as was shown in Fig. 1, to guarantee that all the users and samples are evenly represented in both sets. In the first experiment, we tried different number of parameters, n, to find the optimal n, with different random orderings. It was observed that apart from small fluctuation, the results do not improve when n gets higher than 4 (in fact, even n=3 gives closely similar results, but is less stable). Therefore, we have used value n=4.

The proposed method was designed as a single pass method. Therefore, the weights and parameters can be continuously updated when new ratings are made, without keeping track on all the former ratings. Unfortunately, this approach makes the method also less robust to the order of ratings. In contrast, PPMF is an iterative method that is computationally more complex, but relatively robust for different initializations. The robustness of the proposed method can be improved by feeding the ratings to the algorithm several times in different randomized orders, but in most cases the performance improvement is not very significant, and occasionally PPMF also fails to give good results.

To realistically estimate the typical performance of the methods, we have run each experiment with six different patterns to split data in training and test sets, and each of them ten times with different random order of input, and reported the average results, in terms of Linear Correlation Coefficient (LCC) and Root Mean Squared Error (RMSE), for each database in Table II. The results from different runs do not vary drastically, and therefore we considered direct averaging as a sufficiently accurate method for estimating the average performance. The best result among different methods is bolded. Since normalized scores are used, RMSE results between different databases are mostly comparable; however, the results for TUM and VQEG HDTV are influenced by the discrete rating scale, and they should be compared against the other databases with caution (the absolute prediction accuracy is slightly worse for discrete scale, due to quantization effect). Two examples of the predicted scores are plotted in Fig. 3. Figure 4. shows boxplots for the scores of two example video sequences in EPFL-PoliMi database. As this example shows, both PPMF and the proposed method can predict the average score of the test set more accurately than the baseline.

The results indicate that in average, PPMF performs slightly better in terms of LCC, but the proposed method predicts the individual scores slightly more accurately in terms of RMSE. However, there are large differences between databases. According to LCC results, PPMF predicts the relative differences between scores most accurately for all the databases, except VQEG HDTV. On the other hand, RMSE results show that the proposed algorithm predicts the absolute scores more accurately in four cases: TUM, VQEG HDTV, and CVD Tests 1 and 4. For TUM database, PPMF seems to misinterpret the range of the scores, and therefore RMSE is even worse than for the baseline. This explains largely the average RMSE difference between PPMF and the proposed method. The reason is unknown, but we assume that it might be due to rather unbalanced distribution of the scores (high scores are overrepresented in TUM dataset, in comparison to the other datasets).

TABLE I. RESULTS FOR DIFFERENT DATABASES

Method	Baseline		PPMF		Proposed	
Database	***LCC***	***RMSE***	***LCC***	***RMSE***	***LCC***	***RMSE***
EPFL-PoliMi	0.90	0.254	**0.93**	**0.214**	0.91	0.230
TUM	0.65	0.424	**0.79**	0.545	0.75	**0.361**
VGEQHDTV	0.83	0.382	0.80	0.398	**0.87**	**0.340**
CVD Test 1	0.76	0.297	**0.86**	0.269	0.81	**0.265**
CVD Test 2	0.74	0.300	**0.79**	**0.284**	0.76	**0.284**
CVD Test 3	0.68	0.397	**0.69**	**0.389**	0.68	0.397
CVD Test 4	0.83	0.355	**0.86**	0.347	**0.86**	**0.330**
CVD Test 5	0.66	0.417	**0.73**	**0.377**	0.73	0.381
CVD Test 6	0.74	0.378	**0.78**	**0.361**	0.75	0.372
Average	*0.75*	*0.358*	***0.80***	*0.354*	*0.79*	***0.329***

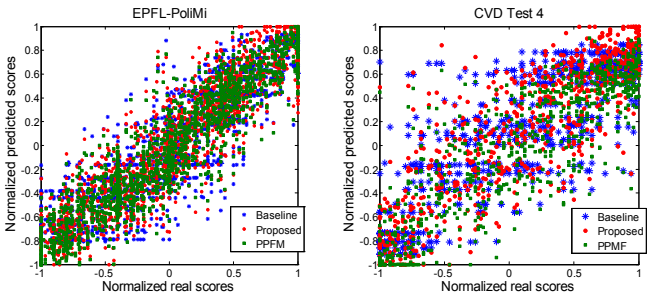

Fig. 3. Example of the results with EPFL-PoliMi and CVD Test 4 (normalized real scores vs. normalized predicted scores).

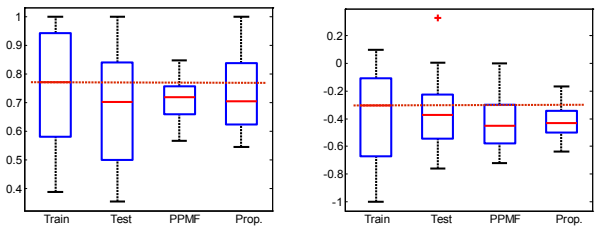

Fig. 4. Boxplots for the training set, test set, and test set predicted using PPMF and the proposed method for two example video sequences in EPFL-PoliMi database. Red dashed line is the baseline (mean score of training set).

D. Discussion of Results

The results comparing the performance of the benchmark (PPMF) and the proposed method seem inconclusive. In general, both PPMF and the proposed method give more

accurate estimates of scores than the baseline (MOS), which indicates that the personal preferences in quality assessment tasks indeed can be at least roughly estimated by analyzing the past ratings by the same person.

On the other hand, the type of the quality assessment study also influences to the results. For example, in EPFL-PoliMi database, the rates given by different people are relatively consistent and well in line with each other. In this case, even the baseline gives relatively accurate results, and they can be even further improved by collaborative filtering. EPFL-PoliMi database focuses on packet loss artifacts only, which probably makes the quality assessment task cognitively easy and the influence of random factors remains small. TUM and VQEG-HDTV databases represent typical quality assessment databases, where both compression and channel artifacts are represented. Videos in these databases are more challenging for test subjects than in EPFL-PoliMi database, but the variety of relevant artifacts is still relatively limited, and the prediction of individual scores is clearly improved by using either the proposed method or PPMF.

On the other hand, CVD database includes videos with qualitatively large range of different distortions and factors influencing the subjective quality. In addition to quality ratings, also other information is collected. Therefore, the subjective quality assessment task is significantly more challenging than for the three other datasets, and we assume that this is the reason why the scores in CVD tests tend to be less consistent and more difficult to predict, in particular for Test 3 and Test 5. We expect that the prediction accuracy could be significantly improved by using the additional evaluations (on sharpness, saturation etc.) collected about the characteristics of the videos in the CVD database, in a similar fashion as MOS values are predicted from those characteristics in [15].

V. Conclusions

In this paper, we have studied how the video quality scores given by individual users can be predicted from the user's earlier history of scores to different test videos, and different users' scores to the same test video. This problem is conceptually similar to the problem of predicting user preferences in recommendation systems, but to our knowledge, this problem has not been studied in the context of visual quality assessment before. We expect that the proposed approach will have interesting applications: first, it can be used to develop objective quality metrics that predict the individual quality scores based on earlier rating behavior; second, it can be used to create subjective quality assessment methods where each user rates only a subset of the test clips, and MOS can still be estimated accurately; and third, our approach can be used to study the validity and limitations of annotated quality databases.

We have proposed an algorithm that learns user's preferences and test videos' latent features continuously, as new scores are given. The proposed method was validated using public annotated video quality databases with raw subjective scores. The results show that the prediction accuracy for individual scores can be improved by using either the proposed method or PPMF matrix factorization method developed for generic collaborative filtering. However, the results depend highly on the content and the task. The proposed approach works best for well-defined, cognitively easy quality assessment tasks. For more challenging tasks and contents with a lot of different types of distortions, user preferences tend to be less consistent, and typically PPMF yields better results. More sophisticated hybrid approaches will be studied in the future work.

References

[1] I. Galloso, J. F. Palacios, C. Feijóo, A. Santamaría, "On the Influence of Individual Characteristics and Personality Traits on the User Experience with Multi-sensorial Data: an Experimental Insight," Multimedia Tools and Applications, vol. 75, 44 p., Feb. 2016.

[2] S. Winkler, "Analysis of Public Image and Video Databases for Quality Assessment," IEEE Selected Topics in Signal Processing, vol. 6, no. 6, pp. 616-625, Oct. 2012.

[3] D. Z. Rodríguez, R. L. Rosa, E. A. Costa, J. Abrahão, and G. Bressan, "Video Quality Assessment in Video Streaming Services Considering User Preference for Video Content," IEEE Trans. Consumer Electronics, vol. 60, no. 3, pp. 436-444, Mar. 2014.

[4] J. Korhonen, and J. You, "Improving Objective Video Quality Assessment with Content Analysis," VPQM'10, Scottsdale, AZ, USA, Jan. 2010.

[5] B. Ortiz-Jaramillo, J. Niño-Castañeda, L. Platiša, and W. Philips, "Content-Aware Objective Video Quality Assessment," J. Electronic Imaging, vol. 25, no. 1, 16 p., Jan. 2016.

[6] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," IEEE Computer, vol. 48, no. 8, pp. 30-37, Aug. 2009.

[7] 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," ACM Computing Surveys, vol. 47, no. 1, 45 p., Jul. 2014.

[8] F. De Simone, M. Naccari, M. Tagliasacchi, F. Dufaux, S. Tubaro, and T. Ebrahimi, "Subjective assessment of H.264/AVC video sequences transmitted over a noisy channel," Proc. QoMEX'09, San Diego, CA, USA, Jul. 2009.

[9] F. De Simone, M. Tagliasacchi, M. Naccari, S. Tubaro, and T. Ebrahimi, "H.264/AVC video database for the evaluation of quality metrics," Proc. ICASSP'10, Dallas, TX, USA, Mar. 2010.

[10] http://vqa.como.polimi.it/

[11] M. Barkowsky, M. Pinson, R. Pépion, and P. Le Callet, "Analysis of Freely Available Dataset for HDTV Including Coding and Transmission Distortions," Proc. VPQM'10, Scottsdale, AZ, USA, Jan. 2010.

[12] http://ivc.univ-nantes.fr/en/databases/VQEG_HDTV_Pool2/

[13] C. Keimel, J. Habigt, T. Habigt, M. Rothbucher, and K. Diepold, "Visual Quality of Current Coding Technologies at High Definition IPTV Bitrates," Proc. MMSP'10, Saint-Malo, France, Oct. 2010.

[14] ftp://ftp.ldv.ei.tum.de/videolab/public/TUM_1080p25_Data_Set/

[15] M. Nuutinen, T. Virtanen, M. Vaahteranoksa, T. Vuori, P. Oittinen, and Jukka Häkkinen, "CVD2014 - A Database for Evaluating No-Reference Video Quality Assessment Algorithms," IEEE Trans. Image Processing, vol. 25, no. 7. pp. 3073-3086, 2016.

[16] http://www.helsinki.fi/~tiovirta/Resources/CVD2014/

[17] H. Shan, and A. Banerjee, "Generalized Probabilistic Matrix Factorizations for Collaborative Filtering," Proc. of ICDM'11, Sydney, Australia, Dec. 2010.

[18] http://www-users.cs.umn.edu/~shan/ppmf_code.htm