No-reference Video Quality Assessment on Mobile Devices

Chen Chen, Li Song* *member IEEE*, Xiangwen Wang, Meng Guo

***Abstract*—The explosive growth of video applications and services on mobile devices has made it important to assess video quality. In this paper, we propose a no-reference video quality assessment method for mobile videos. Based on the analysis on common mobile video impairments, three features (blockiness, blurriness and noise) were extracted. The features are then trained to predict the DMOS (Differential Mean Opinion Score) through a support vector machine (SVM). To reduce complexity and increase adaptation, we capture a set of independent images from screen shot, and compute underlying features directly from the spatial domain. Dataset from a public database is used to train and test. Experimental results show that the proposed model provides satisfactory performance on characterizing the spatial domain impairments.**

***Index Terms*—No reference video quality assessment, spatial domain, feature extraction**

I. Introduction

Evaluating mobile video quality becomes a hot topic which is important for content providers and network services providers. Most existing methods depend on evaluating the quality of the underlying transport network such as video coding quality, packet loss, etc. However, Quality of Experience (QoE) is the overall experience a consumer has when accessing and using provided services. By utilizing objective measures of visual quality, perceptually optimized QoE could be delivered to the mobile terminal end-user. However, there are several issues needed to be addressed before practical objective measures can be done directly from client's mobile terminal.

First, mobile video bit stream is sometimes not available for the reason of unopened transmission protocols or digital right management issues. Even if the bit stream is available, problems still exist on mobile circumstance. For example, the original resolution of a video is 1280x720 but the mobile display resolution is 800x480. Then images will be scaled when playing. Thus assessing video quality only by analyzing the bit stream cannot give a true score as a person is now watching scale version of the original video.

Secondly, on mobile devices, it is impossible to access reference videos which are compressed, transformed and transmitted to the terminal. The approach we adopted in this paper is a No-reference (NR) quality assessment method, which needs no reference video when making prediction. In contrast, Both Full-reference (FR) and Reduced-reference (RR) methods, need information from the original source (full or partial).

Recent researches[1][2][3][4] show NR image quality assessment methods have similar or even better performance when compared to FR method such as well-known SSIM metric. All these methods were targeting at providing a generic image quality metric according to discrimination of image statistical features between an original image and its distorted version. For mobile application in our case, however, it is better to design specific NR system with prior information of concerned distortions.

Complexity is another major issue when designing a video quality assessment method running on mobile devices as computation resources and power are usually limited. Even offline training is generally implemented on powerful computer, fast feature extraction is still a very important issue for real-time mobile applications.

To address these challenges, in this paper we design a practical and efficient system to evaluate mobile video quality. It includes an offline training sub-system and an online testing executable program running on a smartphone. Specifically, we capture the screenshot of mobile device by reading the display frame buffer, which make us process the same images as human eyes see. Then we design efficient algorithms for three common but important video distortions: blockiness, blurriness and noise respectively. To get the relationship between these three factors and subjective scores, a support vector machine (SVM) regression model is used and parameters are trained offline by use of public dataset with subjective scores. With available model, online quality evaluation can be done by averaging results from each captured frame. Experiments show our system can provide an efficient solution for practical mobile video quality assessment.

The rest of this paper is organized as follows. In section II,

Manuscript received February 24, 2013. This work was supported by National 863 project (2012AA011703), National Key Technology R&D Program of China (2013BAH53F04), NSFC (61221001, 61271221), the 111 Project (B07022) and the Shanghai Key Laboratory of Digital Media Processing and Transmissions.

Chen Chen and Li Song is with the Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (Li Song's phone: 86-21-34204468; fax: 86-21-34204155; e-mail: 445452902@qq.com, song_li@sjtu.edu.cn).

Xiangwen Wang is with the Institute of Electronic and Information Engineering, Shanghai University of Electric Power, Shanghai 200240, China (e-mail: wxw21st@gmail.com).

Meng Guo is with the Research Lab of China Mobile, Beijing 100053, China(e-mail: guomeng@chinamobile.com).

we analyze various effects of spatial domain artifacts and introduce the corresponding quality assessment methods. In section III, the complete implementation of our prototype system is presented. In section IV, simulated results are presented. Section V concludes this paper.

II. Analysis and Measurement of Spatial Domain Impairment

Mobile videos are subject to various kinds of distortions like noise, blurriness and blockiness, so measuring the extent of these typical spatial domain artifacts can assess the video quality. It should be noted that temporal artifacts like frame freezing are easy to be detected, thus we will not discuss such issues in this paper.

A. Video Screenshot

Mobile videos (network videos especially) are always not full screen displayed. For example, people see videos through video website like *YouTube* and *Youku* (*http://www.youku.com/*). In this case, video bit stream is not reliable, because it is not the visual content actually received by people. Screenshot is more flexible, showing no difference with human perception.

In our application, we select the screenshot area and read corresponding RGB value from frame buffer on mobile device. Then, RGB format is converted to YUV (4:2:0) format. Only Y component is used to be analyzed.

How fast can screenshot be captured is a main factor which greatly influences the real-time performance. In our application, the time cost on screenshot is as much as that on feature extraction and predicting. The processing frame rate is dependent on the mobile processor capability.

B. Video Blockiness

One of the most common quality issues of mobile video is blockiness, an impairment in which the image contains artifacts that resemble small blocks of a single color.

Wang et al. [5] estimate JPEG images blockiness as the average differences across block boundaries. Quantization operation is applied to the DCT coefficients in each 8*8 coding block, so it is easy to locate the block boundaries. However, the case on mobile is quite different. The block is often scaled or shifted when the frame is captured. We need a method that is independent on the size and location of the block. Pan et al. [6] proposed an edge direction information based method which is invariant to the displacement, rotation and scaling of the images. We improve this algorithm and it behaves well in application.

Edge direction information based method constructs the edge direction histogram. Images with severe blockiness have very strong edge direction presences at 0◦ and 90◦, which is the result of the abrupt inter-pixel discontinuity of cross-block pixels in the horizontal direction and vertical direction. It is also noted that as the blocking artifacts become more severe more and more edge pixels will align in these two directions, and more pixels become inactive or "flat". Therefore the proportion of the orientation of edges along these two directions as well as becoming "flat" in an image is a very good indication how severe the blocking artifacts are.

Different from the procedure in [6], we do not compute the whole histogram of the direction. Only three angles were counted which greatly reduce the algorithm complexity.

The pixel gradient vectors can be approximated by using *Sobel* operator:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I \quad and \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I \quad (1)$$

Where "*" here denotes the 2-dimensional convolution operation, "I" denotes the image.

As the opposite gradient vectors indicate the same edge orientation, we double the angles of the gradient vectors.

$$(G_x + jG_y)^2 = G_x^2 - G_y^2 + \mathrm{j}2G_xG_y \quad (2)$$

To reduce the noise effect, average gradient can be calculated by averaging in a window W,

$$DF_x = \sum_W \left(G_x^2 - G_y^2\right), \; DF_y = \sum_W (2G_xG_y) \quad (3)$$

By introducing three counters: *CH* counts Horizontal pixels, *CV* counts Vertical pixels and *CF* counts Flat pixels, the gradient map can be obtained as:

If $DF_x = 0$ and $DF_y = 0$, *CF*++.

If $DF_x > 0$ and $\left|\frac{DF_x}{DF_y}\right| > 60$, *CH*++.

If $DF_x < 0$ and $\left|\frac{DF_x}{DF_y}\right| > 60$, *CV*++.

Then block feature is calculated by:

$$Blockiness = \frac{CH+CV}{Size_I}\left(1 + \beta \frac{CF}{Size_I}\right) \quad (4)$$

Where $Size_I$ denotes the total pixels in the image and β denotes a parameter with value 1.6 in our application.

C. Video Blurriness

Video blurriness typically occurs in two different ways: When capturing the video or performing the encoding of the video stream.

Referring to Min Goo Choi et al. [7], we developed an edged based blurriness detection algorithm. Blur estimation is divided into two steps: Detect the edge of an image first and then check whether the detected edge is sharp enough by comparing the neighboring pixels with the edge itself. When images contain a clear foreground and a background with blurriness, the MOS value keeps high. The edge detection can remove the impact of the background to make the assessment of blurriness more accurate.

We denote the image (M×N) as $I(x,y)$. The horizontal absolute difference value of a pixel is defined by

$$D_h(x,y) = |I(x,y+1) - I(x,y-1)| \quad (5)$$

If the pixel value of (5) is bigger than horizontally adjacent pixels, the pixel is considered as edge in horizontal direction

$$E_h(x,y) = \begin{cases} 1 & if\ D_h(x,y) > D_h(x,y+1)\ and \\ & \quad D_h(x,y) > D_h(x,y-1) \\ 0 & otherwise \end{cases} \quad (6)$$

Then, we examine the blurriness extent of the detected edge pixels.

$$BE_h(x,y) = \frac{|I(x,y) - D_h(x,y)/2|}{D_h(x,y)/2} \quad (7)$$

The larger the BE value is, the sharper the edge pixel is. In the same way, BE_v value can be estimated in the vertical direction. Blur Edge pixel is defined by

$$B(x,y) = \begin{cases} 1 & if\ \max\left(BE_h(x,y), BE_v(x,y)\right) < Threshold \\ 0 & otherwise \end{cases} \quad (8)$$

This equation means the pixel with blurriness extent under Threshold is considered as blurred. Finally, the mean is calculated by

$$Blur_{mean} = \frac{Sum_{BE}}{Sum_B} \quad (9)$$

Where Sum_B is the count of blurred pixels in and Sum_{BE} is the sum of BE.

D. Video Noise

Video noise is random variation of grey or color values in images, and is usually an aspect of electronic noise. It can be produced by the sensor and circuitry of a scanner or digital camera.

Noise artifact is easy to extract. If a pixel is considered as noise, it will be quite different from its neighbor. As a result, we use a 3*3 mask to perform convolution to the whole captured frame as follow

$$I' = I * \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} \quad (10)$$

Then get a value which describes the overall difference from each pixel to its neighbor.

$$Noise = \frac{Sum(|I'|)}{M \times N} \quad (11)$$

Every pixel in filtered image matrix I' is first converted to absolute value and summed up. After that, Noise feature is scaled by the size of the image.

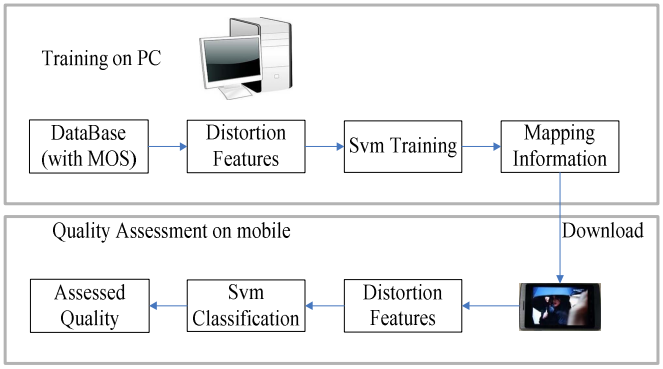

Fig. 1. System framework

III. System Implementation and Applications

The framework of our video quality assessment system is illustrated in figure 1 which consists of two key operations: SVM training on PC and quality assessment on mobile device.

A. Training procedure

Our dataset, LIVE Image Quality Assessment Database Release 2[8], consists of many distortion types. We choose 233 JPEG images, 174 Gaussian blur images and 174 white noise images from them, which include spatial domain impairments like: blockiness, blurriness and noise artifacts. The set is divided into training set (80%) and test set (20%). 186 JPEG images, 139 Gaussian blur images, 139 white noise images are training set while the others are used for testing performance. During extracting distortion features, algorithms mentioned in Section II are applied. Mapping information is learned from feature space to predict scores using a regression model. In our implementation, a support vector machine (SVM) regression model is used. LIBSVM package [9] is utilized in application. Training procedure is performed on PC. After training, mapping information can be downloaded on mobile devices.

B. Quality assessment on mobile devices

With the SVM mapping results, mobile devices can assessment video quality in real time. While playing any video on mobile, screenshot is captured as fast as possible. The speed is dependent on the mobile device. In our case, *Samsung i9100* cellphone can capture and analyze 5 frames per second (fps). Our algorithm only involves spatial domain information, so the difference in processing frame rate is endurable. We extract the distortion features of every frame captured and assess the quality. The averaging score is the final predict quality for the video.

C. Application

The operation of the software is very simple. Shaking the cell

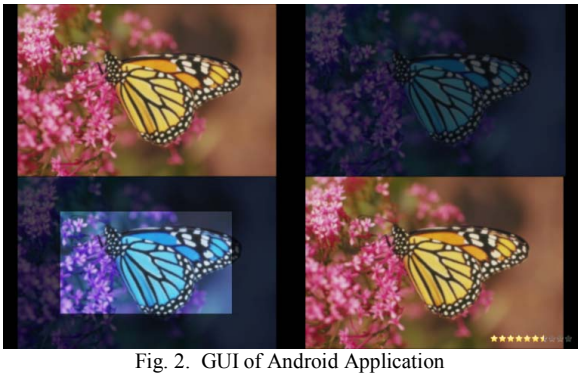

Fig. 2. GUI of Android Application

Fig. 3. Experiment results

phone to start our background quality assessment service, and shake again to end the process. Before the assessment actually begins, cell phone will get into a screencast mode. One can slide fingers on the screen to select the area we are interested in. A rating bar will be at the lower right corner while the service is on. Figure 2 shows the GUI of the developed application and figure 3 gives some results of our software.

IV. Experimental Results

Only spatial information is utilized in our algorithm, so we compute video quality by averaging frames quality. We firstly evaluate the performance of our algorithm on the test set of LIVE Image Quality Assessment Database Release 2 as mentioned in Section III-A. Then we test our algorithm on LIVE Mobile Video Quality Database [10].

The Spearman's rank ordered correlation coefficient (SROCC) and Pearson linear correlation coefficient (PLCC) were used to check the performance of these features in this paper.

$$r_{Pearson} = \frac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\overline{Y})^2}} \quad (12)$$

$$r_{Spearman} = \frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\overline{y})^2}} \quad (13)$$

Where $X_i, Y_i, \overline{X}, \overline{Y}$ are raw scores and average values, and x_i, $y_i, \overline{x}, \overline{y}$ are corresponding ranks and average values.

A. Correlation of features with Human Opinions

In Figure 4-6, we plot the relation between each of these features and human DMOS from the LIVE IQA database for corresponding distortions in the database, to ascertain how well the features correlate with human perception of quality. No training is undertaken here, the plot is to illustrate that each spatial domain feature contains quality information.

The Features were passed through a logistic nonlinearity as described in [10] before computing correlation. The Spearman's rank ordered correlation coefficient (SROCC) and Pearson (linear) correlation coefficient (LCC) between each of these features and human DMOS are 0.92/0.84 (Blockiness), 0.82/0.86 (Blurriness) and 0.94/0.83 (Noise). A value close to one indicates good performance in terms of correlation with human opinion. Figure 4-7 shows the relation.

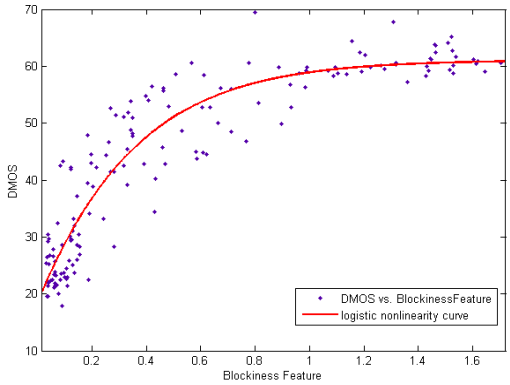

Fig. 4. Relation between DMOS and Blockiness feature

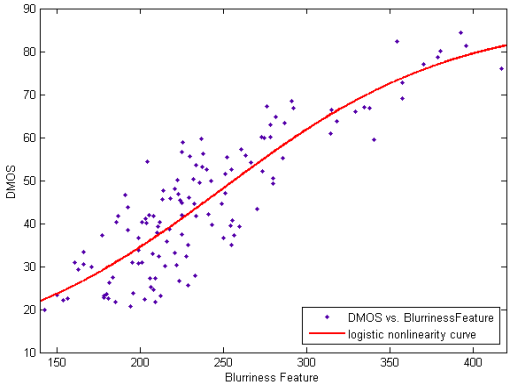

Fig. 5. Relation between DMOS and Blurriness feature

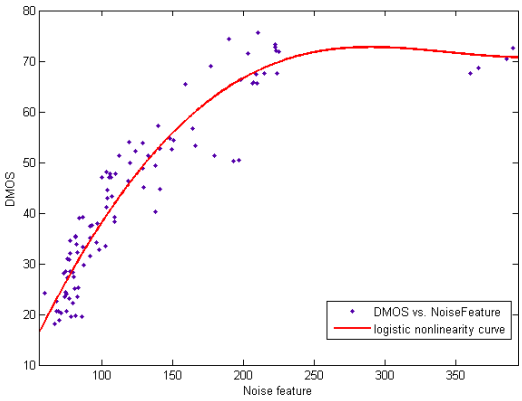

Fig. 6. Relation between DMOS and Noise feature

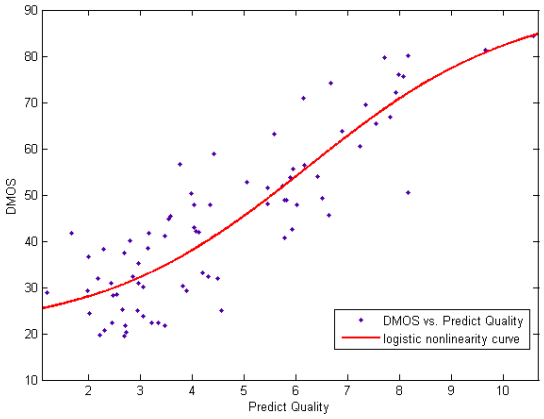

Fig. 7. Relation between DMOS and predict quality (LIVE IQA database)

B. Correlation of predict score with Human Opinions

Since our approach requires a training procedure to calibrate the SVM regression model, we divide the LIVE IQA Database (with 233 JPEG images, 174 Gaussian blur images and 174 white noise images) into two subsets – 80% training and 20% testing performance. This division ensures that results do not depend on features extracted from known spatial content, which can improve performance. Figure 7 shows the result.

The predict quality was passed through a logistic nonlinearity described in [11]. The SROCC and PLCC between the predict quality and DMOS are 0.83 and 0.86 respectively.

As mentioned in abstract, we treat video data as a set of independent images, thus reduce video quality issue to image quality assessment problem. Now, we test our algorithm on the LIVE Mobile Video Quality Database [10]. Four video sequences (*bf*, *dv*, *ss* and *tk*) were tested. Except for freezing distortion videos, each sequence has eleven kinds of distortion, including 4 layers of compression with fixed QP encoding, 4

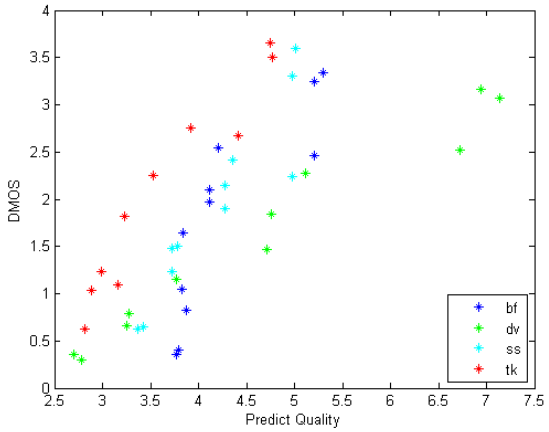

Fig. 8. Relation between DMOS and predict quality (LIVE Mobile Video Quality Database)

levels of simulated wireless channel packet-loss without error concealment, and 3 types of rate variation to simulate rate-changes as a function of time. All video files have planar YUV 4:2:0 format and 450 frames. Taking the limitation of screenshot rate on mobile into account, we analyze the data once every 10 frames. The average of every predict frame quality is considered as the video quality. Figure 8 shows the relation between DMOS and predict quality. Table 1 gives the correlation coefficient.

TABLE 1
CORRELATION BETWEEN DMOS AND PREDICT QUALITY OF EACH SEQUENCE

Sequences	LCC	SROCC
bf	0.86	0.95
cv	0.98	0.98
ss	0.93	0.98
tk	0.96	0.96
average	0.78	0.82

It can be seen a high correlation between human score and prediction quality exists within the same video sequence, while the performance drops cross the whole database. It is because absolute subjective quality varies a lot with different video contents.

In LIVE Mobile Video Quality Database, a temporal rate (and thus quality) dynamics condition was simulated to evaluate the effect of multiple rate-switches. We also use these videos to verify the validity of our metric. Two videos (*bf_t134*, *bf_t431*) were selected. "*bf*" is the source sequence name, "*t134*" means rate-switches: $R_1 - R_3 - R_4$. Each video has 450 frames and corresponding subjective temporal scores. The results are shown in figure 9 and 10. And the Pearson (linear) correlation and Spearman's rank ordered correlation of *bf_t134* and *bf_t431* are 0.86/0.86 and 0.79/0.82 respectively.

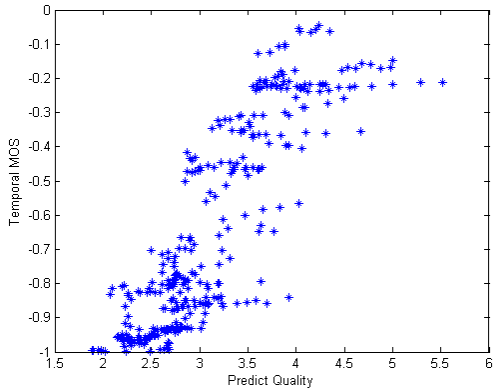

Fig. 9 Correlation between predict quality and temporal MOS (*bf_t134*)

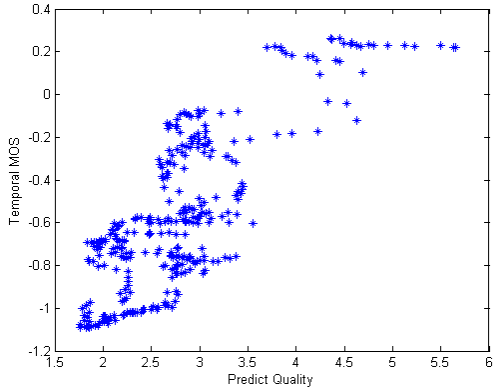

Fig. 10. Correlation between predict quality and temporal MOS (*bf_t431*)

We have also compared our algorithm to the BRISQUE [2] - a recent method based on statistical characteristics of natural images. Although the BRISQUE achieves better results (for example, SROCC is about 0.9), its complexity is much higher than ours. For example, even we extract only half of the original features; it still takes 3 seconds to process a 340x500 image. In contrast, our method achieves a good tradeoff between performance and complexity.

V. CONCLUSION

We proposed a no-reference video quality assessment metric on mobile devices based on spatial domain feature extraction. Three distortion features blockiness, blurriness and noise were modeled in the algorithm.

Considering the characteristic on mobile devices, we analyze the screenshot which is directly received by human instead of video bit stream. Then, we detailed the spatial domain features extracted, and demonstrated these features are highly correlated with human perception. A prediction score is available from the features after a SVM regression model. Finally, we evaluate the performance by computing the correlation between the predict quality and human subjective score. Compared to other state-of-the-art algorithms, the complexity of our algorithm is relatively low and makes it more attractive for mobile devices.

REFERENCES

[1] A. Mittal, R. Soundararajan and A. Bovik, "Making a Completely Blind Image Quality Analyzer ", *IEEE Signal processing Letters*, 2012 (online). http://live.ece.utexas.edu/research/quality/niqe_spl.pdf

[2] A. Mittal, A. K. Moorthy and A. C. Bovik, "No-Reference Image Quality Assessment in the Spatial Domain", *IEEE Transactions on Image Processing*, vol. 21(12), pp. 4695-708. Dec. 2012

[3] A. K. Moorthy and A. C. Bovik, "Blind Image Quality Assessment: From Scene Statistics to Perceptual Quality", *IEEE Transactions Image Processing*, vol. 20, no. 12, pp. 3350-3364, 2011.

[4] M. Saad, A. Bovik and C. Charrier, "Model-Based Blind Image Quality Assessment: A natural scene statistics approach in the DCT domain", *IEEE Transactions Image Processing*, vol. 21, no. 8, pp. 3339-3352, 2012.

[5] Z. Wang, H. Sheikh and A. Bovik. "No-Reference Perceptual Quality Assessment of JPEG Compressed Images". In *Proceeding of IEEE International Conference on Image Processing* (ICIP), 2002, vol.1, pp.477-480.

[6] F. Pan, X. Lin, S. Rahardja, E. Ong, W. Lin. "Using edge direction information for measuring blocking artifacts of images". *Multidim Syst Sign Process*, Vol.18:297–308, 2007.

[7] M. Choi, J. Jung and J. Jeon. "No-Reference Image Quality Assessment using Blur and Noise". *World Academy of Science, Engineering and Technology*, Vol.50, pp.163-167, 2009.

[8] H. Sheikh, Z. Wang, L. Cormack and A. Bovik. "LIVE Image Quality Assessment Database Release 2," http://live.ece.utexas.edu/research/quality.

[9] C. Chang and C. Lin. LIBSVM: A Library for Support Vector Machines [Online]. http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[10] A. Moorthy, L. Choi, G. deVeciana, and A. Bovik, "Mobile Video Quality Assessment Database," in *Proceedings of IEEE International Communication Conference Workshop on Realizing Advanced Video Optimized Wireless Networks*, Ottawa, Canada, June 10-15, 2012.

[11] H. Sheikh, M. Sabir, and A. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans Image Process*, vol.15, no 11, pp. 3440-3451, 2006.