Bulletin of Electrical Engineering and Informatics
Vol. 8, No. 1, March 2019, pp. 231~238
ISSN: 2302-9285, DOI: 10.11591/eei.v8i1.1418

VMQ: an algorithm for measuring the Video Motion Quality

Nawaf O. Alsrehin, Ahmad F. Klaib
Computer Information Systems Department, Faculty of Information Technology and Computer Science,
Yarmouk University, Irbid, Jordan

Article Info

Article history:

Received Oct 22, 2018
Revised Nov 25, 2018
Accepted Dec 28, 2018

Keywords:

Distorted video
Video Motion Quality
Video quality
Video transcoding

ABSTRACT

This paper proposes a new full-reference algorithm, called Video Motion Quality (VMQ) that evaluates the relative motion quality of the distorted video generated from the reference video based on all the frames from both videos. VMQ uses any frame-based metric to compare frames from the original and distorted videos. It uses the time stamp for each frame to measure the intersection values. VMQ combines the comparison values with the intersection values in an aggregation function to produce the final result. To explore the efficiency of the VMQ, we used a set of raw, uncompressed videos to generate a new set of encoded videos. These encoded videos are then used to generate a new set of distorted videos which have the same video bit rate and frame size but with reduced frame rate. To evaluate the VMQ, we applied the VMQ by comparing the encoded videos with the distorted videos and recorded the results. The initial evaluation results showed compatible trends with most of subjective evaluation results.

Corresponding Author:

Ahmad F. Klaib,
Computer Information Systems Department,
Faculty of Information Technology and Computer Science,
Yarmouk University, 21163, Irbid, Jordan
Email: ahmad.klaib@yu.edu.jo

1. INTRODUCTION

Video creating, recording, and playing have been facilitated due to the capability improvement of mobile devices, digital cameras, and the development of their applications. For example, the 12Mega Pixels digital camera in the iPhone 7 Plus® captures high-resolution video up to 4K and it offers SLOw-MOtion (SLO-MO) in 1080p at 120 frames per second (fps) and 720p at 240 fps. In addition, Cisco® predicts that the online videos will account for more than 80% of all consumer internet traffic by 2020 [1]. Furthermore, over 8 billion videos are watched on Facebook every day [2].

However, delivering videos to the end-user based on the required video format and desired Quality of Services (QoS) is still very expenses due to the limited or unpredictable network bandwidth, diversity of end-user devices, vast amount of data in digital videos, and variety of video formats. Video transcoding is required to allow critical end-users to watch any requested video at any time and from anywhere based on the required format and desired QoS [3]. Due to the limited bandwidth and congestion problems of some wireless network, reducing the video frame rate is highly applicable in video delivery systems. Dropping some frames from the reference video to generate a new distorted version reduces the video file size; and thus saves more bandwidth. However, dropping some frames affects the motion level of the distorted video and therefore reduces the motion quality of the perceived video.

Satisfying the desired QoS requires transcoding the reference video in a way that keeps the motion level in an acceptable level and therefore optimizes video delivery process. Measuring the video motion quality of the distorted video requires a metric that compares frames from both videos in a way that smartly considers the reduction in the frame rate from both videos.

Frame rate represents the number of complete still images shown every second. The Human Visual System (HVS) is capable of differentiating between 10 and 12 of still images per second, after this frame rate, the HVS starts just seeing it as motion [3]. The motion looks jagged if the frame rate is too slow and blurred if the frame rate is too high. Choosing the right frame rate is an interesting human-factors and network bandwidth problem, but it is outside the scope of this paper.

Naturally, motion representation in videos plays an important role in the perception of video quality [4]. Evaluating the video motion quality can be done either objectively or subjectively. Objective evaluation techniques are mathematical models that approximate expert judgments, these techniques are full-reference, reduced-reference, and no-reference [5]. Subjective evaluations, on the other hand, require expert judgments. Subjective studies are used to evaluate the performance of objective methods and algorithms. However, subjective studies are time consuming, difficult to implement, cumbersome, expensive, have to be undertaken manually, and impractical for most applications due to the human involvement in the evaluation process. Therefore, the objective methods are used to achieve the ultimate goal of matching the human perception [6]. Evaluating the performance of objective video quality models is usually done by calculating the correlation and error values between the model results and the results obtained with subject tests [7].

Full-Reference (FR) metrics are metrics that compute the video quality of the distorted video by comparing the original video signal against the distorted video signal, in which every pixel from the source is compared against the corresponding pixel at the distorted video. In FR metrics, both the original and distorted videos should be available [5]. Video quality in full-reference models is a measure of how a distorted video looks compared to the original. Existing FR Video Quality Assessment (VQA) algorithms do not fully use motion information from both videos to estimate the video motion quality of the distorted one. In addition, these algorithms fail to do adequate job in evaluating the motion quality level based on the reduction of frame rate between the reference and distorted videos.

It is found that the Motion-based Video Integrity Evaluation (MOVIE) [4], Structural SIMilarity (SSIM) [8], Multi-Scale-Structural SIMilarity (MS-SSIM) [9] and Video Quality Model (VQM) [10] indexes tend to give the best performance [11]. The MOVIE requires high-computation and intensive operations that limit its effectiveness and applicability. In addition, to apply the MOVIE index, both videos should have the same frame rate. HVS is highly adapted for extracting structural information form scenes. SSIM [8] measures the quality of still images using a single-scale structural similarity paradigm, which provides a good approximation to perceived image quality. MS-SSIM [9] supplies more flexibility than SSIM method in incorporating the variations of image resolution and viewing conditions, and the experimental comparisons done in [9] demonstrate the effectiveness of MS-SSIM.

2. RELATED WORK

Yilin Wang et. al. [12] assumed that directly applying SSIM frame by frame is insufficient for VQA due to ignoring of the temporal information. Therefore the authors extended SSIM for Image Quality Assessment (IQA) by incorporating spatiotemporal information. However, their proposed method assumes that the reference and distorted videos have the same frame rate, which is not compatible with most of the real video delivery systems and applications.

Kai Kang et. al. [13] showed an effective and efficient objective video quality metric based on the video content which focuses on using motion information in VQA compared to the previous work which develops the VQA that takes advantages of the various characteristics of HVS. On the whole, the proposed method mainly combines the motion information in temporal and structure information in spatial domain of video sequences. However, their proposed method assumes that both videos have the same frame rate and it does not consider the distortion in video frame rate.

Phong V. Vu and Damon M. Chandler [14] proposed the Frame Distortion and Motion Dissimilarity (FDMD) algorithm as an approach to VQA which combines the frame-based distortion measurement with a spatiotemporal analysis of motion dissimilarity. The FDMD algorithm uses the Most Apparent Distortion (MAD) algorithm [15] to compute the frame-based distortion and develop a spatiotemporal model to capture motion dissimilarity through the STS images.

Jos´e Joskowicz et. al. [16] presented a review of a set of parametric models published by ten different groups of authors. Each model is briefly described, and the relevant parametric formulas are presented. The performance of each model is evaluated and contrasted to some other models, using a common video clips set, in different coding and transmission scenarios. It just uses the values of the frame rates from both videos as parameters in the general parametric model. Our proposed model uses the content of each frame from both videos.

Yen-Fu Ou et. al. [17] investigated the impact of temporal variation of the Frame Rate (FR) and the Quantization Step-Size (QS) on the perceptual video quality. Among all possible variation patterns, the study

focused on videos in which two FRs (or QS's) alternate over a fixed interval, and explored the human responses to such variation by conducting subjective evaluation of test videos with different variation magnitudes and frequencies. Zhongkang Lu et.al. [18] developed a numerical model that measures the effect of delectability and annoyance of periodic frame dropping on perceptual visual quality evaluation under different content and frame size conditions.

Yen-Fu Ou et. al. [19] attempted to understand how the perceived quality of a video varies as the frame rate changes and to explore the influence of video content and video resolution on the visual sensitivity to frame rate. Their proposed model does not use the frames content from both videos; it just uses the values of the frame rates from both videos as parameters. Our proposed model uses the content of each frame from both videos. Also, they assume that the reference video is artifact-free version, which is unreal assumption.

3. PROPOSED ALGORITHM FOR VIDEO MOTION QUALITY

In this paper, we propose a new full-reference algorithm, called Video Motion Quality (VMQ) that evaluates the relative motion quality of the distorted video generated from the reference video based on the frame rate information from both videos. VMQ uses any frame-based objective quality metric; here we used the MS-SSIM [9], for comparing frames from both videos. It finds the intersection value between each two frames from the reference and distorted videos. It calculates the intersection value based on the timestamp for each frame, after that it multiplies this intersection value with the result of the objective quality metric that is generated by comparing these frames together. Finally, it calculates the weighted average for all these comparisons. Algorithm 1 describes the VMQ metric in more details. Figure 1 shows an example that depicts how VMQ calculates the motion quality of the distorted based on two videos that have different frame rates.

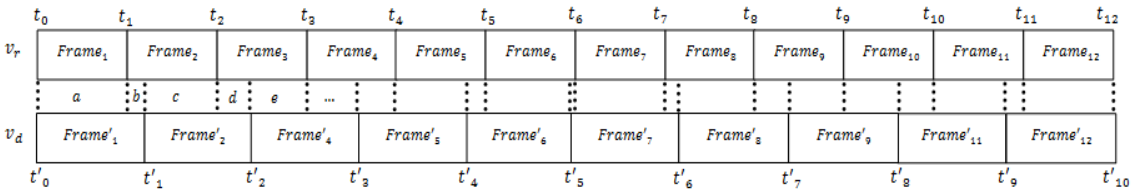


Figure 1. An example that shows how the VMQ works

The VMQ compares two videos that have different frame rates. Algorithm 1 generates a value that represents the video motion quality of the distorted video. This value ranges from 0 to 1, the higher the value, the better the motion quality. VMQ uses frames from both videos to perform the comparison and generate the timestamp values. It then divides the summation of all the comparison values by the summation of all the intersection values as a weighted aggregation function to produce the final result as shown in (1).

Each frame has a timestamp, which represents the time at which this frame will be displayed in the video. The intersection value represents the overlapping between two frames, from the original and the distorted videos, at a given time interval. Figure 1 shows an example of how VMQ extracts the timestamp differences; it shows two videos that are at different frame rates, v_r is at 12 fps and v_d is at 10 fps. v_d is generated from v_r by reducing the frame rate from 12 fps to 10 fps. Each video is represented as a sequence of boxes; each box represents a frame. Also, we numbered the boxes from $frame_1$ to $frame_{12}$. During the distortion process $frame_3$ and $frame_{10}$ were dropped from v_r to generate v_d. The dropping mechanism is an interesting issue related to the design of the video codec and how the transcoder works, but it is outside the scope of this paper.

For example and based on Figure 1, given $frame_1$ from v_r with t_0 timestamp, $frame_2$ from v_r with t_1 timestamp, $frame_3$ from v_r with t_2 timestamp, and so on. Also given $frame'_1$ from v_d with t'_0 timestamp, $frame'_2$ from v_d with t'_1 timestamp, $frame'_4$ from v_d with t'_2 timestamp. The time stamp values from t_0 to t_{12} represent the actual viewing time for v_r from the begging to the end. In addition, the time stamp values from t'_0 to t'_{10} represent the actual viewing time for v_d from the begging to the end. We represent the time stamp differences for some of these frames from both videos using a, b, c, d, and e variables where $a = t_1 - t'_0$, $b = t'_1 - t_1$, $c = t_2 - t'_1$, $d = t'_2 - t_2$, and $e = t_3 - t'_2$, and so on.

Algorithm 1 VideoMotionQuality(video v_r, video v_d) // the proposed algorithm

Input: two videos, a reference video, v_r, and a distorted video, v_d.

Output: a number between 0 and 1 that represents the motion quality of the distorted video.

```
size_r ← num_of_frames(v_r);
size_d ← num_of_frames(v_d);
size = min(size_r, size_d);
for (i = 1 to size) do
    ot_c = v_r.getframe(i).getTimeStamp(); // the timestamp value of the current frame in v_r
    ot_n = v_r.getframe(i + 1).getTimeStamp();// the timestamp value of the next frame in v_r
    tt_c = v_d.getframe(i).getTimeStamp();v_o ();// the timestamp value of the current frame in v_d
    if (ot_c = tt_c)
        intersection = ot_n − ot_c;
        getValues(v_r.getframe(i), v_d.getframe(i), intersection);
    else if (tt_c > ot_c)
            tt_p = v_d.getframe(i − 1).getTimeStamp();// the timestamp value of the previous frame in v_d
            for (j = 1 to size_o) do
                ot_c = v_r.getframe(j).getTimeStamp();// the timestamp value of the current frame in v_r
                ot_n = v_r.getframe(j + 1).getTimeStamp();// the timestamp value of the next frame in v_r
                if (ot_c = tt_c)
                    intersection = ot_n − ot_c;
                    getValues(v_r.getframe(j), v_d.getframe(i), intersection);
                if ((ot_c > tt_p) AND (ot_c < tt_c) AND (ot_n = tt_c))
                    intersection = tt_c − ot_c;
                    getValues(v_r.getframe(j), v_d.getframe(i − 1), intersection);
                else if ((ot_c > tt_p) AND (ot_c < tt_c) AND (ot_n < tt_c))
                    intersection = ot_n − ot_c;
                    getValues(v_r.getframe(j), v_d.getframe(i − 1), intersection);
                    else if ((ot_c < tt_c) AND (ot_n > tt_c))
                        intersection = tt_c − ot_c;
                        getValues(v_r.getframe(j), v_d.getframe(i − 1), intersection);
                        end if
                    end if
                end if
                end if
                if ((ot_c < tt_c) AND (ot_n > tt_c))
                    intersection = ot_n − tt_c;
                    getValues(v_o.getframe(j), v_t.getframe(i), intersection);
                end if
            end for
        end if
    end if
end for
return indexAll /weightAll ;
```

To calculate the video motion quality of v_d generated from v_r using VQM, $frame_1$ is compared with $frame'_1$ from v_d using any frame-based metric, like MS-SSIM, and the result is multiplied by a, which represents the value of the time difference, then $frame_2$ from v_r is compared with $frame'_1$ from v_d and the result is multiplied by b, after that $frame_2$ from v_r is compared with $frame'_2$ from v_d and the result is multiplied by c and so on. Finally, the values of all these multiplications are added together and then divided by the summation of all the intersection values (i.e., the differences in timestamps). Here, we used the MS-SSIM [9] as a frame-based metric to compare frames.

Also, given a four-elements set, $S = \{frame_i, frame_j, inter_{i,j}, index_{i,j}\}$ such that $frame_i \in fr$, $frame_j \in fd$, $index_{i,j}$ represents the comparison value between $frame_i$ and $frame_j$ using frame based metric, and $inter_{i,j}$ represents the intersection value between $frame_i$ and $frame_j$ that is generated from timestamp differences described above. The video motion quality value, $v_d.quality_{VMQ,v_r}$, for the video v_d generated from video v_r and measured by VMQ is calculated based on all the comparisons and intersections values as follows:

$$v_d.quality_{VMQ,v_r} = \frac{\sum_{i\in fr, j\in fd}^{|S|}(index_{i,j} * inter_{i,j})}{\sum_{i\in fr, j\in fd}^{|S|} inter_{i,j}} \quad (1)$$

VMQ uses a set of variables to store intermediate values that will be used later. Her, we will describe just the functions. The getValues(...) function compares two frames (i.e., $frame_i$ and $frame_j$) using the getIndex(...) function. The getIndex(...) function represents the real use of the frame based index, such as MS-SSIM. After comparing two frames from both videos, the getValues(...) function multiplies the

intersection value with the index value returned from the getIndex(...) function and store the result in the weight variable. Then the getValues(...) function updates the values of the indexAll and weightAll static variables. The indexAll static variable represents the summation of all the multiplications of index and intersection values together and the weightAll static variable represents the summation of all the intersection values. The VMQ returns the final value as a video motion quality result (i.e., line 40).

VMQ uses the MS-SSIM [9] for the comparison. However, plugging in any frame based metric that compares two frames, such as Peak Signal-to-Noise Ratio (PSNR) [20] or SSIM [8], is applicable and considered as an important feature in the VMQ.

4. MEASUREMENT METHODOLOGY

4.1. Video clips and encoding

In this paper, a set of raw, uncompressed HDTV videos from the VCDL video data set [21] were used. For evaluation, we selected 13 raw videos in total based on varieties of video content, ranges of scene source material, and varieties of color and brightness components. These videos range widely from slow motion to high motion. Each video is a 10-second length with no audio content, 1080p progressive scan with Full HD 1920x1080 as a frame size. We encoded each raw video to generate new sets of encoded videos at 30 frames per second (fps) as a frame rate, and at 2, 4, 6, 8, 10, 12, 14, 16, 18, and 20 Mbps as video bit rates using the H.264 video codec. From the raw videos, we generated 130 (13 original raw $*$ 10 different bit rates) different encoded videos.

4.2. Generating distorted videos

From each of encoded video, new sets of distorted videos were generated by reducing the frame rate without any temporal filtering. The reduction is from 30 fps to 27, 25, 23, 20, 17, and 15 fps. These sets includes 780 videos (130 encoded videos $*$ 6 different frame rates). To generate the distorted videos, we used Java® and Xuggler® for implementing the encoding and transcoding functionalities. Figure 2 shows an example of the general structure of the encoding and transcoding steps for each raw, uncompressed video to generate a distorted one, as we described above.

4.3. Quality estimation

In this step, we applied the VMQ algorithm by comparing each encoded video, at 30 fps, with its corresponding distorted video, at different frame rates specified above. Figure 3 shows the video quality evaluation for all selected videos, which is calculated using the VMQ by comparing the original and distorted videos. For example and to describe the comparison methodology based on Figure 2, we compared "src01_8Mbps" with "src01_8Mbps_17fps" using the VMQ algorithm.

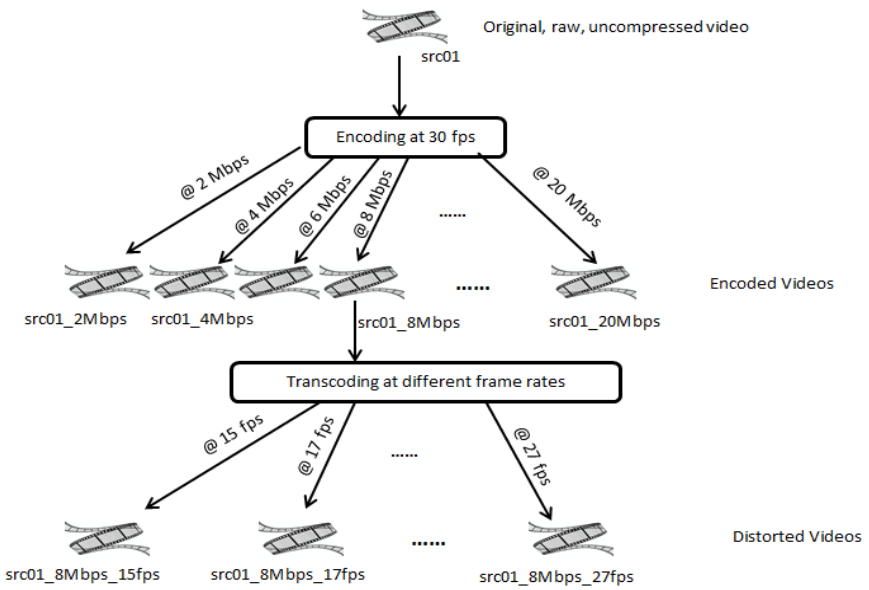
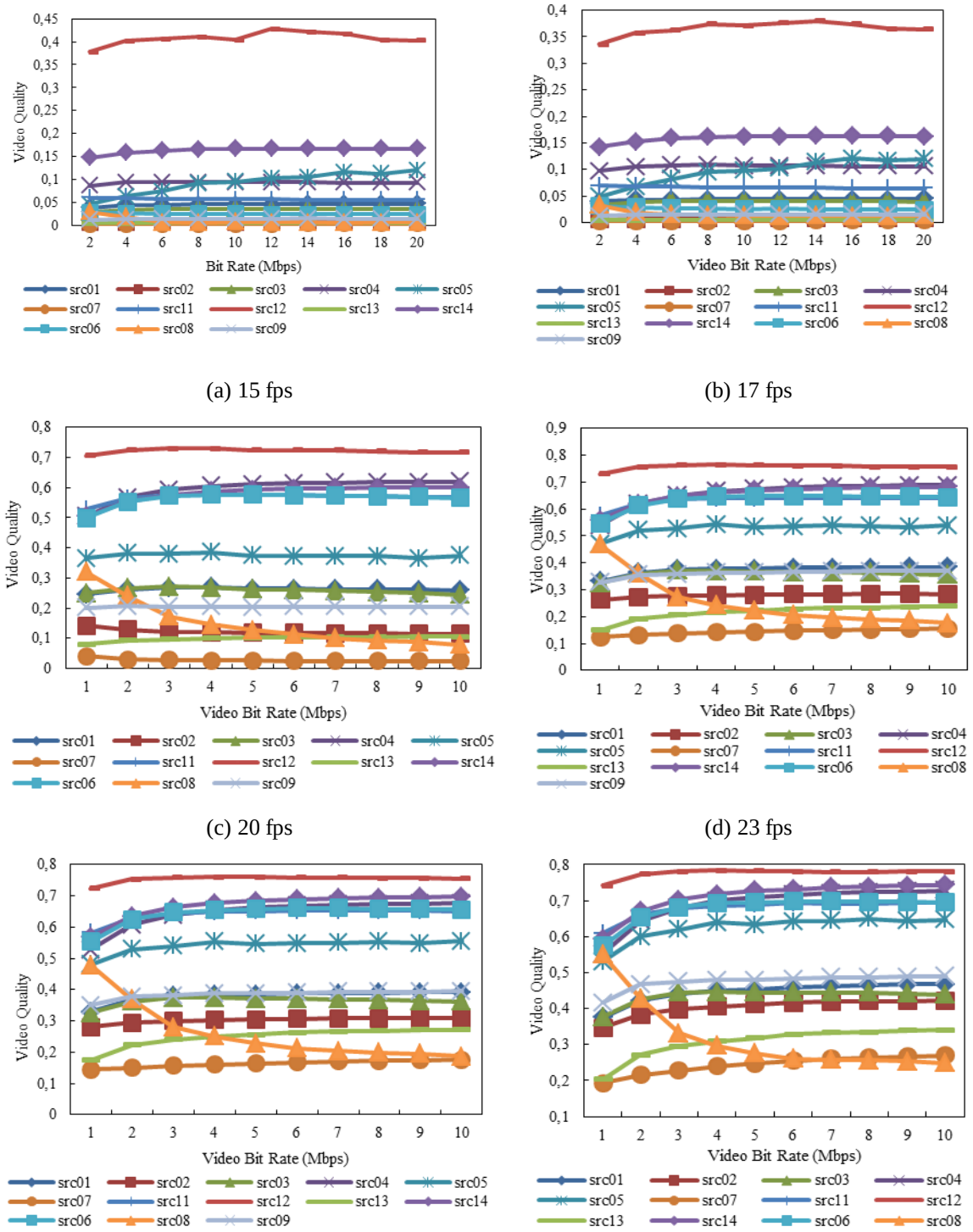


Figure 2. An example of the encoding and transcoding steps to generate the distorted videos for src01 video

Figure 3 shows that most of these videos have the same trend; there is a little improvement in the quality when the video bit rate is increased. However, for src08, there is a decreasing trend in video quality when the bit rate is increased at all the frame rate reductions. Also, for src02 and src07, there is a decreasing trend in video quality at the beginning when the reduction is from 30 to 20 fps as it shown in Figure 3(c). We believe that these videos have different trends because of the video content type and motion level. These two parameters must be considered in a future research.



(a) 15 fps (b) 17 fps

(c) 20 fps (d) 23 fps

(e) 25 fps (f) 27 fps

Figure 3. The video motion quality results for the VMQ using the MS-SSIM [9] metric

Figure 4 shows the video motion quality evaluation results for the six different frame rates (i.e., from 15 to 27 fps) combined together in one graph. It shows that the 27 fps curve achieves the highest quality while the 15 and 17 fps curves achieve the lowest video motion quality, which is compatible with general trends of most subjective evaluation results.

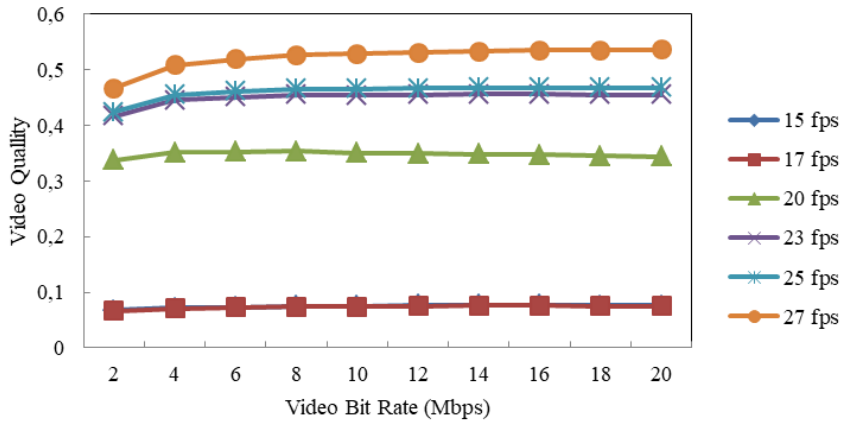


Figure 4. The video motion quality evaluation results based on the VMQ algorithm for all the frame rates

5. CONCLUSION

This paper presents a full-reference algorithm, called Video Motion Quality (VMQ), to evaluate the motion quality of the distorted video by incorporating the frame rate information from both videos. VMQ extracts the timestamp from each frame from both videos and calculates the time differences and then finds the weighted average of comparing two frames from both video based on the time difference. In addition, VMQ incorporates a well-known quality metric for still images, called MS-SSIM [9] to compare frames from both videos. Initial experiments were done to validate the proposed algorithm. The results suggest that the proposed algorithm is able to give a good approximation that is generally compatible with evaluations from subjective tests without manual tuning. In addition, the MS-SSIM [9] metric is able to distinguish between quality levels for different frame rates and used to quantify the behavior of different video content on the motion quality. VMQ represents an initial step toward developing a more robust metric that considers more factors such as motion level and video content type. Also, to illustrate the effectiveness and efficiency of the VMQ, we are planning to perform comprehensive subjective tests.

REFERENCES

[1] Cisco Systems, "Cisco VNI Forecast and Methodology, 2015-2020," San Jose, CA, USA, 2015.
[2] J. Constine, "Facebook Climbs To 1.59 Billion Users And Crushes Q4 Estimates With $5.8B Revenue," TechCrunch, Jan 27, 2016.
[3] A. Tarantola, "Why Frame Rate Matters," 14 January 2015. [Online]. Available: http://gizmodo.com/why-frame-rate-matters-1675153198. [Accessed 25 March 2017].
[4] K. Seshadrinathan and A. C. Bovik, "Motion Tuned Spatio-Temporal Quality Assessment of Natural Videos," in *IEEE Transactions on Image Processing*, vol. 19, no. 2, pp. 335-350, Feb. 2010.
[5] N. O. Alsrehin, "Toward A General Parametric Model for Assessing the Impact of Video Transcoding on Objective Video Quality," Utah State University, Logan, UT, USA, 2016.
[6] K. Seshadrinathan, R. Soundararajan, A. C. Bovik and L. K. Cormack, "Study of Subjective and Objective Quality Assessment of Video," in *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1427-1441, June 2010.
[7] J. Joskowicz, R. Sotelo and J. C. L. Ardao, "Towards a General Parametric Model for Perceptual Video Quality Estimation," in *IEEE Transactions on Broadcasting*, vol. 59, no. 4, pp. 569-579, Dec. 2013.
[8] Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, April 2004.
[9] Z. Wang, E. P. Simoncelli and A. C. Bovik, "Multiscale structural similarity for image quality assessment," *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, Pacific Grove, CA, USA, 2003, pp. 1398-1402 Vol.2.
[10] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," in *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312-322, Sept. 2004.

[11] S. Chikkerur, V. Sundaram, M. Reisslein and L. J. Karam, "Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison," in *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 165-182, June 2011.
[12] Y. Wang, Q. Zhang and B. Li, "*Video Quality Assessment Via Spatiotemporal-Saliency-Weighted SSIM*," in Eighth International Workshop on Video Processing and Quality Metrics for Consumer Electronics , Chandler, Arizona, USA, 2014.
[13] K. Kang, X. Liu and C. Sun, "Video Quality Assessment Metric Based on Spatio-temporal Motion Information," *2013 IEEE 11th International Conference on Dependable, Autonomic and Secure Computing*, Chengdu, 2013, pp. 47-51.
[14] P. V. Vu and D. M. Chandler, "*Video Quality Assessment based on Motion Dissimilarity*," in Proceedings of Seventh International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, Arizona, 2013.
[15] E. C. L. a. D. M. Chandler, "Most Apparent Distortion: Full-Reference Image Quality Assessment and the Role of Strategy," *Journal of Electronic Imaging*, vol. 19, no. 1, 2010.
[16] J. Joskowicz, R. Sotelo and J. C. L. Ardao, "Towards a General Parametric Model for Perceptual Video Quality Estimation," in *IEEE Transactions on Broadcasting*, vol. 59, no. 4, pp. 569-579, Dec. 2013.
[17] Y. Ou, H. Zeng and Y. Wang, "Perceptual quality of video with quantization variation: A subjective study and analytical modeling," *2012 19th IEEE International Conference on Image Processing*, Orlando, FL, 2012, pp. 1505-1508.
[18] Z. Lu, W. Lin, B. C. Seng, S. Kato, E. Ong and S. Yao, "Perceptual Quality Evaluation on Periodic Frame-Dropping Video," *2007 IEEE International Conference on Image Processing*, San Antonio, TX, 2007, pp. III - 433-III - 436.
[19] Yen-Fu Ou, Tao Liu, Zhi Zhao, Zhan Ma and Yao Wang, "Modeling the impact of frame rate on perceptual quality of video," *2008 15th IEEE International Conference on Image Processing*, San Diego, CA, 2008, pp. 689-692.
[20] Wikipedia, "Peak Signal-to-Noise Ratio (PSNR)," 7 February 2017. [Online]. Available: https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio. [Accessed 25 March 2017].
[21] M. H. Pinson, "The Consumer Digital Video Library [Best of the Web]," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 172 - 174, July 2013.

BIOGRAPHIES OF AUTHORS

Nawaf O. Alsrehin received his B.Sc. degree in Computer Science and Master degree in Computer Information System from Yarmouk University, Irbid, Jordan in 2003 and 2006, respectively. Alsrehin received his PhD degree in Computer Science from Utah State University, USA in 2016. Currently, Alsrehin is an assistant professor and head of the Computer Information Systems department at the faculty of Information Technology and Computer Science, Yarmouk University, Jordan. He is a member in the American Society for Research (ASR), IEEE Advance Teaching for Humanity, and IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), 2019. He has several funded projects in the areas of smart transportation systems, smart homes, and multimedia security. His research interests include multimedia, video and image processing, video transcoding, multimedia services, video quality assessments, distributed multimedia systems, and multimedia applications in the cloud.

Ahmad F. Klaib is an assistant professor at the Computer Information Systems Department, Faculty of Information Technology and Computer Science, Yarmouk University, Jordan. He received his BSc degree in Computer Information Systems from Al Albayt University, Jordan in 2005, Master degree in Computer Science from University of Science, Malaysia in 2007 and PhD degree in Computer Science from the University of Huddersfield, UK in 2015. He has two funded projects in the area of smart homes, smart transportation systems. His research interests include string matching algorithms, text processing, video and image processing, optimization, health care and internet of things technology.