

A bitrate clustering recognition algorithm for wireless video transmission

Wen-Juan Shi^{†,1,2}

¹*School of New Energy and Electronical Engineering,
Yancheng Teachers University, Yancheng 224002, China*

²*School of Information and Electrical Engineering,
China University of Mining and Technology, Xuzhou 221113, China*

[†]*E-mail: winterswj@126.com*

www.yctc.edu.cn

The wireless video transmission process, due to the complexity and variability of wireless communication channels, requires the bitrates to be adjusted to match the dynamic wireless channel. An analysis of the video frame quality could be used as an important basis to adjust the wireless video bitrates. In order to recognize different bitrates for wireless video streaming according to the distribution characteristics of the video frame quality, this paper proposes a video bitrate clustering recognition algorithm called BCRA. BCRA can recognize different bitrates by partitioning the frames of the same bitrate into clusters. The proposed algorithm uses the structural similarity (SSIM) between every original and terminal decoding frame to quantify the degradation of structural information with regard to the video frame quality. Then, the algorithm computes the measure of similarity between two neighboring SSIM values, which indicates the similarity of neighboring frames and can be used to partition the video frames into frame intervals. Finally, a clustering algorithm composed of the hierarchical interval division algorithm and the standard deviation ratio based clustering algorithm is proposed to recognize the different bitrates. Experimental results on the LIVE mobile video quality assessment (VQA) database show that the proposed algorithm can recognize the change of video bitrates by analyzing the video frame quality and is consistent with the real bitrates for wireless video transmission.

Keywords: bitrate recognition; video frame quality; bitrate clustering; wireless video transmission Template.

1. Introduction

In recent years, mobile video applications have generated a rapidly growing proportion of overall mobile traffic, and mobile video services have become one of the most popular streaming media applications. According to the Cisco Visual Networking Index Global Mobile Data Traffic Forecast Update, mobile video traffic accounted for 55% of total mobile data traffic in 2015 and will generate 75% of total mobile data traffic by the end of 2020 [1]. Due to the tremendous increase in mobile video traffic, more and better user-centric quality concepts

and metrics are required to characterize and analyze network traffic; this analysis can contribute to better QoE for real world operational networks [2].

A wireless channel is subject to radio interference, multipath fading and shadowing, and sudden and severe fluctuations in the wireless bandwidth; all of these factors can cause the traffic patterns of the compressed video streaming to change dynamically and can significantly degrade the received video quality [3]. Atzori et al. [4] proposed a source rate control scheme for streaming video sequences over wireless channels; this scheme uses a reduced-reference quality estimation approach at the user-side to avoid abrupt video quality changes that can be caused by sudden variations in the channel throughput. In [5,6], the subjective experiment results showed that wireless user-end videos at high bitrates have statistically better visual quality than those at low bitrates. Results also showed that viewers preferred to switch to a higher bitrate when there were fluctuations in video quality; therefore, this indicates that the distribution of the video frame quality can reflect the changes of wireless video bitrates. As this research demonstrates, bitrate recognition, achieved by extracting the features of the terminal video frames and analyzing the characteristics of these features, can complement the process of dynamically adjusting the bitrates of wireless networks. This paper focuses on automatically recognizing the bitrates based on video frame quality. Considering the consecutive feature of video frames and the similarity measure of neighboring frames, we propose a bitrate clustering recognition algorithm called BCRA to partition the frames at the same bitrates into clusters.

The rest of this paper is organized as follows: Section 2 provides the framework of BCRA and illustrates the detail of the proposed algorithm. Section 3 shows the experiment results on LIVE mobile VQA database. Section 4 comprises the conclusion.

2. Bitrate Clustering Recognition Algorithm

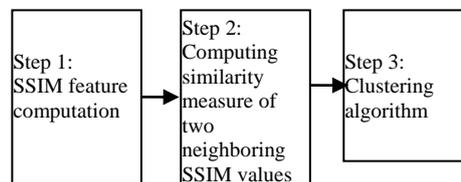


Fig. 1. The framework of bitrate clustering recognition algorithm

In this paper, bitrate recognition is identified as a continuous frame clustering problem, since a video consists of consecutive frames. It needs to be established that a frame cluster will correspond to a continuous frame interval.

The frames within a cluster are approximately at the same bitrate, and are distributed in a frame interval. A bitrate clustering recognition algorithm called BCRA, based on the principle of interval arithmetic and recursive subdivision, is proposed in this paper. The framework of the proposed algorithm is shown in Figure 1. First, we use structural similarity [7,8] (SSIM) between each original and decoded frame as the video frame feature to quantify the degradation of the structural information of the video frame quality. Second, the algorithm computes the similarity measure between two neighboring SSIM values, which indicates the similarity of neighboring frames and is used to partition the video frames into frame intervals. Finally, the clustering algorithm, which comprised of hierarchical interval division algorithm and the standard deviation ratio based clustering algorithm is proposed to recognize the different video bitrates.

Two thresholds are applied in this algorithm. Generally, the “first threshold” and the “second threshold” are, respectively, the minimum mean difference and the minimum standard deviation ratio of SSIM values at different bitrates.

2.1. SSIM feature computation

In this first step of the algorithm, we use SSIM to describe the quality degradation of each video frame. SSIM is the most widely used full-reference quality metric, which is defined by combining luminance, contrast and structure comparisons [7,8]. It can quantify the differences between a distorted image and a reference image under the assumption that human visual perception is highly adapted to extract structural information from a scene. The SSIM value is in the range [0, 1]. As the value approaches 1, the quality improves.

We have conducted an experiment on the LIVE mobile VQA database [6] and found that the mean SSIM value increases and the standard deviation decreases as the bitrate increases, which indicates that the video frame quality improves as the bitrate increases.

For example, the SSIM values of four videos transmitted at different bitrates [R_1, R_2, R_3 and R_4 ($R_1 < R_2 < R_3 < R_4$)] are extracted; the results are illustrated in Figure 2. The horizontal axis is the frame number. The vertical axis is the SSIM values. The SSIM mean and standard deviation values are shown in Table 1. From Figure 2 and Table 1, it can be observed that the mean SSIM increases and the standard deviation decreases as the bitrate increases.

The SSIM values of a rate-changes video are shown in Figure 3. The horizontal axis is the frame number. The vertical axis is the SSIM values. Three types of bitrates can be clearly observed, which are distributed in three continuous frame intervals.

Table 1. The mean and standard deviation of SSIM values at four bitrates

| Bitrate | Mean | Standard Deviation |
|----------------|--------|--------------------|
| R ₁ | 0.8817 | 7.4155e-05 |
| R ₂ | 0.9322 | 3.1923e-05 |
| R ₃ | 0.9650 | 7.6577e-06 |
| R ₄ | 0.9833 | 1.0463e-06 |

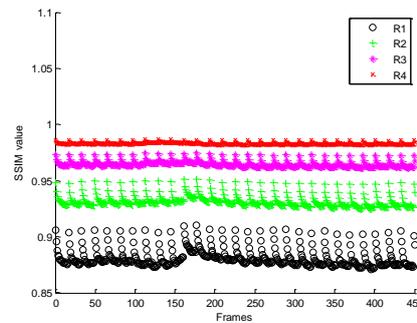


Fig. 2. The distribution of SSIM values at four bitrates

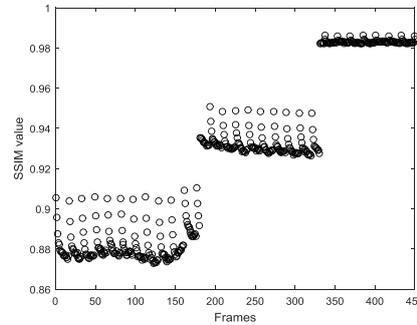


Figure 3. The distribution of SSIM values in a rate-changes video

2.2. Computing similarity measure

In the second step, the similarity measure of two neighboring SSIM feature values is performed, which is defined by:

$$s(x_i, x_{i+1}) = 1 - \frac{|x_i - x_{i+1}|}{|x_{\max} - x_{\min}|} \quad (1)$$

where x_i and x_{i+1} are the i th and $(i+1)$ th SSIM values of neighboring frames, and x_{\max} and x_{\min} are the maximum and minimum SSIM values in a frame interval. The value of the similarity measure is in the range [0, 1]. As the value approaches 1, the neighboring SSIM values are more similar. The smaller the

similarity value, the more likely it is that the corresponding point will be a discontinuity point that can divide a frame interval into two frame subintervals.

2.3. Clustering algorithm

In the third step, a clustering recognition algorithm is proposed, which is generated according to the mean and standard deviation of the SSIM values. In this section, according to the similarity measure between neighboring SSIM values, we perform a hierarchical interval division to produce a rough clustering; this can identify the distribution of the frames at high and low bitrates. Then, we divide the frame intervals into finer frame intervals using the standard deviation ratio. Finally, clusters with the same characteristics are merged into one another.

2.3.1. Hierarchical interval division algorithm

The discontinuity points are listed in ascending order according to the degree of similarity, which can be detected according to the similarity measure discussed in section 2.2. The mean difference of the frame intervals on both two sides of the discontinuity point is compared against the first threshold. Since a frame interval corresponds to a cluster, if the difference of the frame intervals on both sides of the discontinuity point is greater than the first threshold value, the above frame intervals represent two clusters; otherwise, they would belong to one cluster and the frame interval would not be partitioned. Next, the algorithm recursively partitions the clusters into subclusters. The first threshold value must be set according to the mean differences of the SSIM values of the video. In general, the first threshold value is set to the minimum mean difference of the SSIM values at different bitrates. The clustering produced at each step results from the previous step in which a single cluster is split into two clusters. The hierarchical interval division algorithm is described in Algorithm 1.

Algorithm 1 The hierarchical interval division algorithm

Input: video frames

Output: frame intervals

1. Obtain the length N of video frames
 2. Calculate the SSIM values of each video frame
 3. Calculate the similarity measure between neighboring SSIM values
 4. Detect the discontinuity points $disc_i$ and define data set $disc$ of the discontinuity points $disc = \{disc_1, disc_2, \dots, disc_N\}$
 5. **for** $i=1: N$ **do**
 6. Calculate the difference d_i between the mean of SSIM values within the intervals on both two sides of the discontinuity point $disc_i$;
 7. **if** $d_i >$ the first threshold value **do**
 8. divide the frame intervals on both two sides of the discontinuity point $disc_i$ into two frame subintervals;
-

```

9.      renew the frame interval and the discontinuity point;
10.    else
11.      break;
12.    end if
13.  end for
14.  define the preliminary clusters {cluster1, cluster2, ...,
      clusters}

```

2.3.2 Standard deviation ratio based clustering algorithm

Because they are affected by the content of each frame, while the SSIM values cluster around the mean, the distribution of these values is random. The higher bitrates are much more difficult to distinguish than the lower bitrates in the hierarchical interval division algorithm. This is because the distribution of SSIM values at the lower bitrates is much more scattered than it is at the higher bitrates.

However, if a frame interval includes more than one bitrate, the standard deviation would be greater than in a case in which only one bitrate is included in the interval. The standard deviation of SSIM values at different bitrates with different orders of magnitude is an unsuitable measure for determining whether to partition the frame intervals. The standard deviation ratios of SSIM values in the frame intervals are applied to determine whether to partition the frame interval into frame subintervals, which can be done in order to solve the problem of the difference caused by orders of magnitude. The ratio is defined by dividing the larger standard deviation by a smaller one, which is compared with the second threshold. When the ratio is smaller than the second threshold, the frame interval is divided into two frame subintervals. Dichotomy is used to determine whether to divide the existing frame interval into two frame subintervals after application of the hierarchical interval division algorithm, which compares the standard deviation ratio with the second threshold. Generally, the second threshold value is set to the minimum standard deviation ratio. The clustering algorithm based on the standard deviation ratio is described in Algorithm 2.

Since sometimes the transmission process will occur at the same bitrate, it is essential to compute the mean of each frame interval; however, the neighboring frame intervals must be part of different clusters, according to the principles of the BCRA method. Therefore, in order to reduce the calculations required, one need only compute the mean difference between the frame interval and all other frame intervals except the neighboring interval.

Algorithm 2 The standard deviation ratio based clustering algorithm

Input: intervals produced by the hierarchical interval division algorithm

Output: finer frame intervals

1. Compute the number *Num* of frame intervals

```

2. for  $i=1: Num$  do
3.   Divide the  $i$ th frame interval into two halves:
      interval_half $_1$  and interval_half $_2$ 
4.   Calculate the ratio  $r_i$  of the standard deviation of SSIM
      values within the interval_half $_1$  and interval_half $_2$ 
5.   if  $r_i < \text{the second threshold value}$  do
6.     call the hierarchical interval division algorithm
7.   end if
8.   renew the frame interval
9. end for
10. define the finer clusters  $\{cluster_1, cluster_2, \dots, cluster_3\}$ 

```

3. Experiment and Discussion

In this paper, we use the LIVE mobile VQA database [4] to evaluate the performance of the BCRA method, which simulates video distortions in heavily trafficked wireless networks. The database consists of 10 source videos and 200 distorted videos at a resolution of 720p (1280×720). All videos in the database have a duration of 15 seconds and a frame-rate of 30 fps. The distortions include compression, wireless channel transmission losses, frame-freezes, rate adaptation and temporal dynamics.

The rate-adaption videos and rate-switches videos in the database are tested in this paper. The rate-adaption videos are defined as follows: the videos start at bitrate WR_x , and then after n seconds, the bitrate switches to a higher bitrate WR_y , and then again after n seconds, the bitrate switches back to the original bitrate. Three different rate-switches are simulated as (1) WR_1 - WR_4 - WR_1 , (2) WR_2 - WR_4 - WR_2 , and (3) WR_3 - WR_4 - WR_3 , which are called in turn s_{14} , s_{24} and s_{34} .

The rate-switches video is defined as follows: the bitrate is varied between WR_1 to WR_4 multiple times. The five different rate-switches are simulated as (1) WR_1 - WR_4 - WR_1 - WR_4 - WR_1 - WR_4 , (2) WR_1 - WR_2 - WR_4 , (3) WR_1 - WR_3 - WR_4 , (4) WR_4 - WR_2 - WR_1 , (5) WR_4 - WR_3 - WR_1 , which are called in turn t_{14} , t_{124} , t_{134} , t_{421} and t_{431} . Unlike regular rate-adaption videos, the bitrates of rate-switches video are changed irregularly.

2.4. Evaluation accuracy

We tested the BCRA method on the LIVE mobile VQA database. Taking a video from the database called “dv” as an example. The accuracy of BCRA is evaluated by comparing the difference between the frame intervals and the corresponding clusters in the actual categories and also in the clusters of bitrates recognized by BCRA. The actual category and the recognized bitrate clusters and corresponding frame intervals of the video called dv in the LIVE Mobile VQA database are illustrated in Table 2. The different types of videos are separately divided into clusters in the experiments. A frame interval includes the frames at the same bitrate. For example, the frame interval [1, 150] indicates the

frame numbers from the first frame to the 150th frame at the same bitrate. According to the features of the SSIM values of videos at different bitrates, the first threshold and second threshold are set, respectively, to 0.017 and 0.14.

Table 2 displays the experimental results and shows that the recognized bitrate clusters are approximately the same as the actual bitrate category. The correlation coefficient between the recognized clusters and the actual frame intervals approximates 1. This means that BCRA is capable of recognition that is consistent with the actual bitrate.

Table 2. Comparison between the actual category of bitrates and the recognized clusters, based on BCRA of the video called dv.

| Video type | actual bitrate | | recognized clusters | |
|------------------|----------------|----------------|---------------------|----------|
| | interval | category | interval | category |
| s ₁₄ | [1,150] | R ₁ | [1,150] | Cluster1 |
| | [151,300] | R ₄ | [151,300] | Cluster2 |
| | [301,450] | R ₁ | [301,450] | Cluster1 |
| s ₂₄ | [1,150] | R ₂ | [1,150] | Cluster1 |
| | [151,300] | R ₄ | [151,300] | Cluster2 |
| | [301,450] | R ₂ | [301,450] | Cluster1 |
| s ₃₄ | [1,150] | R ₃ | [1,150] | Cluster1 |
| | [151,300] | R ₄ | [151,300] | Cluster2 |
| | [301,450] | R ₃ | [301,450] | Cluster1 |
| t ₁₄ | [1,90] | R ₁ | [1,90] | Cluster1 |
| | [91,150] | R ₄ | [91,150] | Cluster2 |
| | [151,240] | R ₁ | [151,240] | Cluster1 |
| | [241,300] | R ₄ | [241,300] | Cluster2 |
| | [301,390] | R ₁ | [301,390] | Cluster1 |
| | [391,450] | R ₄ | [391,450] | Cluster2 |
| t ₁₂₄ | [1,180] | R ₁ | [1,180] | Cluster1 |
| | [181,330] | R ₂ | [181,330] | Cluster2 |
| | [331,450] | R ₄ | [331,450] | Cluster3 |
| t ₁₃₄ | [1,210] | R ₁ | [1,210] | Cluster1 |
| | [211,360] | R ₃ | [211,360] | Cluster2 |
| | [361,450] | R ₄ | [361,450] | Cluster3 |
| t ₄₂₁ | [1,120] | R ₄ | [1,120] | Cluster1 |
| | [121,270] | R ₂ | [121,270] | Cluster2 |
| | [271,450] | R ₁ | [271,450] | Cluster3 |
| t ₄₃₁ | [1, 90] | R ₄ | [1, 90] | Cluster1 |
| | [91,240] | R ₃ | [91,240] | Cluster2 |
| | [241,450] | R ₁ | [241,450] | Cluster3 |

2.5. Performance comparison

The most well-known and commonly used partitioning methods are k-means and k-medoids [9,10]. A comparison of the performance of K-means, K-medoids and BCRA is evaluated by using the LIVE mobile VQA database; the results of this comparison are given in Table 3. Although the performance of the k-means and k-medoids algorithms exceeds that of the BCRA method, the common disadvantage of the first two algorithms is the necessity for users to specify the number of clusters. BCRA can automatically cluster with prior knowledge of the mean and standard deviation of SSIM values at different bitrates.

Table 3. The performance comparison of K-means, K-medoids and BCRA using the LIVE mobile VQA database.

| Algorithm | Accuracy(%) | Average running times (s) | Time complexity |
|-----------|-------------|---------------------------|-----------------|
| k-means | 97.8% | 0.6880 | $O(nkt)$ |
| k-medoids | 93.75% | 0.8629 | $O(k(n-k)^2)$ |
| BCRA | 92.74% | 0.6116 | $O(n)$ |

The time complexity of the k-means algorithm is $O(nkt)$, where n is the total number of objects, k is the number of clusters, and t is the number of iterations. The time complexity of every iteration in the k-medoids algorithm is $O(k(n-k)^2)$, where n and k are the same as in the k-means method. The time complexity of the proposed algorithm is $O(n)$, where n is the total number of objects.

In addition, the running time of the k-means, k-medoids and BCRA methods is evaluated, which is also shown in Table 3. The experiment is conducted on an Intel Core i5-4590 3.3GHz computer. The software platform is Matlab R2015. Table 3 shows that BCRA algorithm takes the least time of the three algorithms that were compared to recognize the bitrates in a video with 450 frames. When the number of video frames is large, the superiority of BCRA is demonstrated.

4. Conclusion

A bitrate recognition clustering algorithm for wireless video transmissions is presented; this algorithm can recognize the change in video bitrates by analyzing the video frame quality of wireless video transmissions. When compared with the k-means and k-medoids algorithms, this paper's results demonstrate that the proposed BCRA algorithm is effective and produces results that are consistent with the real bitrates.

In the present work, we have attempted to recognize bitrates using the features of the SSIM values of video frames. While limited by the disadvantages

of SSIM, BCRA is suitable for recognizing bitrates in a video with a clear structure found in the LIVE mobile VQA database. This paper's Authors believe there is a need to develop a feature other than SSIM that can better reflect the relationship between bitrates and the quality of video frames.

Acknowledgments

This work is supported by National Natural Science Foundation of China No.51274202, No. 51504214, No.51504255, the Fundamental Research and Development Foundation of Jiangsu Province (No.BE2015040 Natural Science Foundation of Jiangsu province of China No.BK20131124, No.BK2012068, No.BK20130199, the Perspective Research Foundation of Production Study and Research Alliance of Jiangsu Province No.BY2014028-01, the scientific research project of Yancheng Teacher University No.12YCKL002.

References

1. Cisco. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015-2020, Cisco, 2016, 01, <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>.
2. Ali El Essaili, Eckehard Steinbach, Daniele Munaretto, et al. QoE-driven resource optimization for user generated video content in next generation mobile networks. 18th IEEE International Conference on Image Processing, 2011, pp. 913-916.
3. Xiaoping Zhu, Bernd Girod. Distributed Media-Aware Rate Allocation for Wireless Video Streaming. IEEE Transactions on Circuits and Systems for Video Technology, 2010, 20(11), pp. 1462-1474.
4. Ya-Fan Su, Yi-Hsuan Yang, Meng-Ting Lu, et al. Smooth Control of Adaptive Media Payout for Video Streaming. IEEE Transactions on Multimedia, 2008, 11(7), pp. 1331-1339.
5. A.K. Moorthy, K. Seshadrinathan, R. Soundararajan, et al. Wireless Video Quality Assessment: A Study of Subjective Scores and Objective Algorithms. IEEE Transactions on Circuits and Systems for Video Technology, 2010, 20(4), pp. 587-599.
6. A. K. Moorthy, L. K. Choi, A. C. Bovik, et al. Video quality assessment on mobile devices: subjective, behavioral and objective studies. IEEE Journal of selected topics in signal processing, 2012, 6, pp. 652-671.
7. Tiesong Zhao, Jiheng Wang, Zhou Wang, et al. SSIM-Based Coarse-Grain Scalable Video Coding. IEEE Transactions on broadcasting, 2015, 61(2), pp. 210-221.

8. Z. Wang, A. C. Bovik, H. R. Sheikh, et al. Image quality assessment: from error measurement to structural similarity. *IEEE trans. on image processing*, 2004, 13, pp. 600-612.
9. Jiawei Han, Micheline Kamber. *Data Mining: concepts and techniques (Third Edition)* (Morgan Kaufmann Publishers, 2012).
10. S. Theodoridis, Konstantinos Koutroumbas. *Pattern Recognition (Fourth Edition)*(Academic Press, Burlington, 2009).