Improved Quality Keyframe Selection Method for HD Video

Hyeon Seok Yang[1], Jong Min Lee[2], Woojin Jeong[1], Seung-Hee Kim[3], Sun-Joong Kim[3], and Young Shik Moon[1*]
[1] Department of Computer Science and Engineering, Hanyang University
Ansan, 15588, South Korea
[2] Artillery Systems Team, R&D Center, Hanwha Land Systems
Seongnam, 13488, South Korea
[3] Broadcasting & Telecommunication Media Research Lab.,
Electronics and Telecommunications Research Institute (ETRI)
Daejeon, 34129, South Korea
[e-mail: hsyang@visionlab.or.kr, jmlee@visionlab.or.kr, wjjeong@visionlab.or.kr, seung@etri.re.kr, kimsj@etri.re.kr, ysmoon@hanyang.ac.kr]
*Corresponding author: Young Shik Moon

Received July 26, 2018; revised November 16, 2018; accepted January 2, 2019; published June 30, 2019

Abstract

With the widespread use of the Internet, services for providing large-capacity multimedia data such as video-on-demand (VOD) services and video uploading sites have greatly increased. VOD service providers want to be able to provide users with high-quality keyframes of high quality videos within a few minutes after the broadcast ends. However, existing keyframe extraction tends to select keyframes whose quality as a keyframe is insufficiently considered, and it takes a long computation time because it does not consider an HD class image. In this paper, we propose a keyframe selection method that flexibly applies multiple keyframe quality metrics and improves the computation time. The main procedure is as follows. After shot boundary detection is performed, the first frames are extracted as initial keyframes. The user sets evaluation metrics and priorities by considering the genre and attributes of the video. According to the evaluation metrics and the priority, the low-quality keyframe is selected as a replacement target. The replacement target keyframe is replaced with a high-quality frame in the shot. The proposed method was subjectively evaluated by 23 votes. Approximately 45% of the replaced keyframes were improved and about 18% of the replaced keyframes were adversely affected. Also, it took about 10 minutes to complete the summary of one hour video, which resulted in a reduction of more than 44.5% of the execution time.

Keywords: Keyframe extraction, shot-based keyframe selection, video analysis, video summarization, VOD.

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (2015-0-00219, Development of smart broadcast service platform based on semantic cluster to build an open-media ecosystem).

http://doi.org/10.3837/tiis.2019.06.017 **ISSN : 1976-7277**

1. Introduction

With the widespread use of the Internet, services for providing large-capacity multimedia data such as video-on-demand (VOD) services and video upload sites are rapidly increasing. In order for such a large-capacity multimedia to be efficiently provided to the users, video analysis method, summarization method, search method, and browsing method suitable for the characteristics of the multimedia are required. In particular, VOD service providers want to provide a large amount of high-definition (HD) and full high-definition (FHD)-quality videos having a length of about one hour within a few minutes after the broadcast ends. Video summary can provide meaningful and interesting summaries through keyframe extraction in high-quality, high-volume videos. These summaries improve the VOD service and help users quickly and intuitively understand the content of each video and quickly pick the videos of interest.

Recent researches in the field of video summarization include the extraction of keyframes considering differences in pixel values between adjacent frames [1],[2], the keyframe selction using local features [3], keyframe generation considering probability of occurrence [4],[5], video summarization using deep learning [6],[7], a conversion of video image to cartoon based expression using optimization technique [8], and a summarization of videos taken from multiple cameras [9],[10].

Kim [1],[2] proposed a method of automatically extracting keyframes using an activity function. The activity function can measure the change in the content of the frame by summing the difference between the pixels of adjacent frames. Shots are segmented based on this value, and more keyframes are allocated to shots with large amount of changes. In each shot, the number of keyframes is distributed considering the content change.

Guan et al. [3] proposed a keypoint-based keyframe selection method that uses local features instead of representing each frame as global features. They aimed to increase the representation of keyframes and reduce the redundancy of keyframes. To do this, they defined the coverage and redundancy criteria for keypoints. And they select keyframes with a set of keypoints to improve the satisfaction of the two criteria through the optimization method.

Kumar et al. [4] proposed a visual semantic-based 3D video retrieval system using the keyframe representation proposed in [5]. The method proposed in [5] generates a keyframe considering the probability of occurrence of intensity for each pixel of frames in a shot. The generated keyframe represents the frequency of appearance of a frame in a shot in a pixel, but the visual quality of the keyframe tends to deteriorate in a shot with a large change.

Mahasseni et al. [6] proposed a method for unsupervised video summarization with Adversarial long short-term memory network (LSTM) networks. Deep summarizer network performs unsupervised learning to minimize the distance between video and keyframes. They proposed a new generative adversarial framework consisting of summarizer and discriminator. The summarizer decodes the summarization after selecting the video frame with the autoencoder LSTM, and the discriminator performs the distinction between the original video and the generated summary with the other LSTM.

Panda et al. [7] proposed a method of extracting video set information based on video data grouped by topic keywords and summarizing video of the set. To do this, they proposed a collaborative video summarization (CVS) approach. This method aims to find rare shots with representative and diversity and to capture the important characteristics of the video and the

generality of the set. To do this, they set representative, scarcity, and diversity as attributes of a good summary, extract features using CNN in video, perform a collaborative sparse representative selection, and finally generate a summary.

Chu et al. [8] proposed a system to convert video into comics-based presentation, as an effective and interesting storytelling method. This system has page allocation, layout selection, and speech balloon placement as main components. Each component was formulated as an optimization problem and sought a solution in a systematic way. In the preprocessing step, they found the shot boundary, extracted keyframes by shot, and used it as input for the system. Color histogram distance and motion type of the shot were considered for keyframe extraction. Also, to avoid the case of near-duplicate frames due to dynamic motion, keypoint-based keyframe selection [3] was applied to eliminate redundant keyframes.

Kuanar et al. [9] proposed a method for solving multi-view video summarization problem by applying a graph-theoretic solution to efficiently display important information when multiple cameras were photographed from various perspectives. To model the shot representative frame after temporal segmentation, they used semantic feature consisting of visual bag-of-words and visual features. Gaussian entropy was applied to remove low activity frames. In addition, inter-view dependencies were identified through bipartite graph matching. In the preprocessing step, they performed the motion-based shot boundary detection method and used the middle frame of the shot as the keyframe.

Zhang et al. [10] proposed a system that performs video summarization on multiple georeferenced videos for tourists to preview a place of interest. When the user enters a query, the system searches for multi-video, generates a Gaussian-based capture intention distribution model for each video, and detects the geographic ROI (GROI) in keyframes of multiple videos. Then, the system performs GROI-based video segmentation, selects a geo-key, select a high-quality video, and generates a new summarized video. They used the middle frames as the representative frames for the generated summary videos.

Previous studies on video browsing focused on separating a video into shots. Keyframe extraction was not their main objective, and a simple strategy has been used—extracting the first frame or the middle frame of the shot as a keyframe. However, the previous method did not consider the quality of keyframes sufficiently and could not flexibly cope with the difference of video or genre. In addition, the previous methods take a long processing time for massive videos.

In this paper, we propose a fast and high quality keyframe selection method that can flexibly utilize multiple keyframe evaluation metrics. The proposed method can extract high-quality keyframes for about one hour of video in different genres in about 10 minutes.

The contents of the paper are as follows. In Section 2, we review shot-based keyframe extraction methods. In Section 3, the proposed method and the priority filter are introduced. Section 4 evaluates our method in terms of keyframe quality and the computation time. Section 5 concludes and discusses future research.

2. Related Work

Existing shot-based keyframe extraction schemes [11] include simple methods [12], motion-based methods [13], color-based methods [1],[2],[14], entropy-based methods [15],[16], keypoint-based methods [3], and clustering-based methods [17],[18].

The simple methods [12] are the simplest and easiest approach to extract keyframes at pre-defined positions in each shot such as the first frame or the middle frame. These methods need shot boundary detection.

The motion-based methods are methods of searching frames with locally minimum change in the video and extracting the frame as keyframes by calculating the brightness difference between pixels or using an optical flow [13]. They try to take advantage of the tendency of gestures and camera movements to stop in normally accented frames.

The color-based methods [14] are methods of using the color difference between frames, that increases when the shot changes or when the content change is large. The first frame of the video is selected as the first keyframe. If the difference between the color histograms of the succeeding frame and the color histogram of the previous keyframe is greater than the threshold value, the corresponding frame is added as a new keyframe.

The entropy-based methods [15],[16] calculate the difference between neighboring frames based on entropy and add the frame as a keyframe if the entropy difference is greater than the threshold value.

The keypoint-based keyframe selection methods [3] use local features instead of representing each frame as global features.

The clustering-based methods [17],[18] use a clustering algorithm to group similar frames by incorporating a new frame into an existing cluster based on the threshold of similarity, or by including it as a new cluster. A frame near the center of each cluster is selected as a representative keyframe.

The simple methods are fast because the algorithm is simple, but selected keyframes may not have good image quality. The motion-based methods, the color-based methods, the entropy-based methods, and the keypoint-based methods consider only the specific properties of the image and do not consider the quality and the contents of the frames. The clustering-based methods have a disadvantage that the computation time is very long.

In our previous study [19], we proposed a keyframe selection method using image contrast evaluation and motion evaluation. This method was limited to drama and did not take priority into consideration. In this paper, we extend our existing method and propose a method using a priority filter.

3. Proposed method

The objective of the proposed algorithm is to select high-quality keyframes after the shot boundary detection. **Fig. 1** shows the overall process of the proposed algorithm. First, the algorithm selects initial keyframes in each shot. Next, user selects evaluation metrics suitable for the given video genre and then we determine low-quality keyframes to be replaced, by utilizing the priority filter. Finally, the low-quality keyframes are replaced with the high-quality keyframes. The high-quality frames are searched within the corresponding shot.

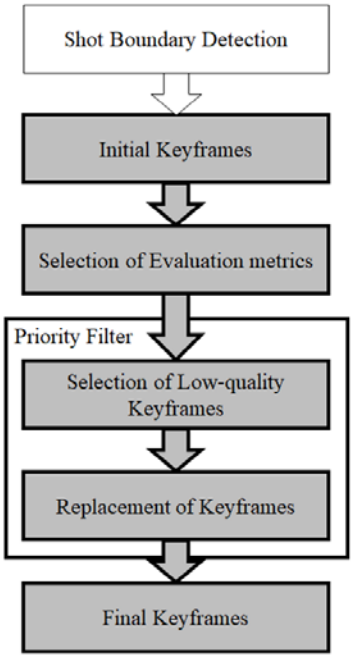

Fig. 1. Flowchart of proposed method.

3.1 Keyframe Selection Method

3.1.1 Initial Keyframe Extraction

At the initial keyframe extraction step, we detect the shot boundaries and extract initial keyframes by using the simple method. We use Park's method [20] as the shot boundary detection method, and then extract the first frames as the initial keyframes. Any shot boundary detection method can be used for our algorithm.

3.1.2 Keyframe Evaluation Metrics

As shown in **Fig. 2**, it is observed that initial keyframes may not be satisfactory to users because of the followings.

1) Image contrast is low.

2) Keyframes have motion blur.

3) Similar keyframes are repeatedly selected.

4) In drama, keyframes without faces are selected.

5) In news, keyframes without captions are selected.

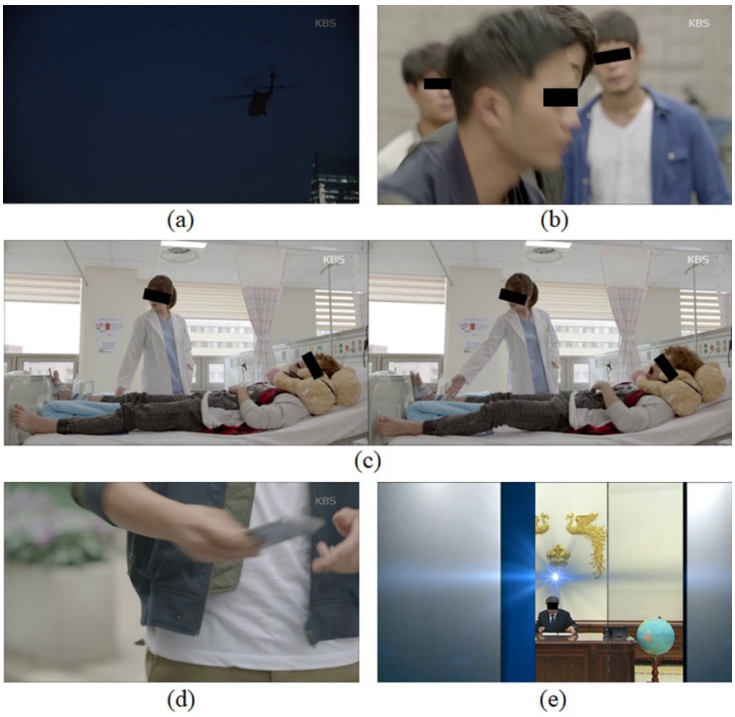

Fig. 2. Example of low-quality keyframe. (a) keyframe with low-contrast, (b) keyframe with motion-blur, (c) similar keyframes, (d) keyframe without face, (e) keyframe without caption.

We define the following "keyframe quality" metrics: image contrast evaluation, motion blur evaluation, similarity evaluation, face detection, and caption detection. We replace frames with low "keyframe quality" by frames with high "keyframe quality", based on this "keyframe quality" metrics. In the rest of this paper, we simply use the word "quality" meaning "keyframe quality".

1) Image Contrast Metric (ICM)

From the histogram of an image *I*, we first exclude the upper and lower 5%, and we select maximum brightness I_{max} and minimum brightness I_{min}. Then we compute the image contrast metric by (1).

$$ICM(I) = \frac{I_{max} - I_{min}}{256} \tag{1}$$

Using (1), we obtain the image contrast metric (*ICM*). The *ICM* represents the range from the minimum intensity to the maximum intensity of the frame *I*. If *ICM*(*I*) is lower than 0.25, frame *I* is determined as a low-contrast keyframe.

2) Motion Evaluation Metric (MEM)

The motion evaluation metric evaluates the degree of motion blur between i-th frame I_i and $(i + k)$-th frame I_{i+k}. We compute the motion evaluation metric (*MEM*) by (2).

$$MEM(I_i, I_{i+k}) = \frac{\frac{1}{L \times k} \sum_{l=1}^{L} \sqrt{{o_{l,x}}^2 + {o_{l,y}}^2}}{\sqrt{N^2 + M^2}} \tag{2}$$

Here, the numerator is the average of the optical flows between frame I_i and frame I_{i+k}. L is the number of optical flows between frame I_i and frame I_{i+k}. $o_{l,x}$ is the x component of l-th optical flow from frame I_i to frame I_{i+k}. $o_{l,y}$ is the y component of the l-th optical flow from frame I_i to frame I_{i+k}. The denominator of (2) is the length of the diagonal of the frame for normalization. N is the height of the frame and M is the width of the frame. We set k to 2.

The average motion between the before and the after frame of the keyframe is computed using the LK-optical flow [21]. If the initial keyframe is the first frame of the shot, we compute the average motion between the first and third frame of the shot. The threshold value is set to 0.015. If the motion is larger than the threshold value, we determine the keyframe to be a blurry frame to be replaced.

3) Similarity Metric (SM)

The similarity metric evaluates the similarity between two images by computing the histogram similarity. Equation (3) shows the histogram similarity between image I and image J.

$$SM(I,J) = \frac{\sum_{j=0}^{255} \min(H(I)_j, H(J)_j)}{N \times M} \tag{3}$$

In this equation, $H(x)_j$ is the value of j-th bin(j = 0, 1, …, 255) of the histogram of an image x. N is the height of the image and M is the width of the image. We evaluate the similarity of the (i-1)-th keyframe with i-th keyframe for replacement. If the similarity is greater than 0.9, then the i-th keyframe is determined as a similar keyframe. The similar keyframe is selected to be replaced by a dissimilar frame.

4) Face detection

To detect human faces, we use the Viola-Jones [22] method. This method detects the human face using AdaBoost with a cascade scheme. We assumed that the presence of a human face is highly related to the importance of a frame, especially in dramas. Thus we detect faces, and then keyframes without faces are considered to be a low-quality keyframe to be replaced.

When the threshold value is set to 1, keyframes without a person are replaced by a frame including a person.

5) Caption detection

We use the Neumann's method [23] to detect captions. Neumann uses an efficient sequential selection in the extremal region sets. In the case of news, captions contain important information. Therefore, we use the sum of the ratio of caption areas in the frame to select more-informative frames. The area of the main captions of the news is around 5%, and the threshold value is set to 0.02. If the ratio is lower than the threshold 0.02, the keyframe is considered to be a caption-free frame to be replaced.

Table 1. Algorithms and thresholds for evaluation metric

Evaluation metric	Algorithm	Threshold
Image contrast evaluation	Image Contrast Metric	0.25
Motion evaluation	LK-Optical flow [21]	0.015
Similarity evaluation	Histogram Similarity	0.9
Face detection	Viola-Jones [22]	1
Caption detection	Neumann [23]	0.02

Meaningful evaluation metrics are different according to the genre or attribute of the video, and the appropriate threshold values should be carefully determined. **Table 1** shows the algorithm for each evaluation metric and the threshold value set in this study. In the case of the image contrast evaluation, it is effective for selecting good keyframes for a video that includes night shots. The motion evaluation can be suitable for dynamic scenes, such as action scenes. The similarity evaluation can reduce the redundancy of keyframes and increase the content coverage by excluding similar keyframes. The face detection is suitable for drama. The caption detection is suitable for cases where captions contain important information, such as news.

The evaluation metrics are applied sequentially according to their priority. The priority was determined by considering the importance, frequency, and computation time of evaluation metrics. Because each evaluation metric has different characteristics, it should be used according to the characteristics of the videos. The evaluation metrics are used for low-quality keyframe selection and keyframe replacement using the priority filter.

3.1.3 Priority Filter

The priority filter is an algorithm that replaces an unsatisfactory keyframe with the most satisfactory frame within the corresponding shot, by applying the selected evaluation metrics with priority according to importance. Appropriate priorities may vary depending on the genre or the nature of the video, so you need to be able to select priorities. For example, the caption detection is not appropriate for dramas but it is of great importance for news. The priority filter algorithm consists of two stages: the first is to select low-quality keyframes to be replaced, and the second is to replace it with high-quality ones.

3.1.3.1 Procedure for priority filter

1) Selection of low-quality keyframes

In the first step, the evaluation metrics are applied for the selection of low-quality keyframes. If a keyframe does not satisfy one or more evaluation metrics, it is selected as a replacement target. In the evaluation procedure at the time of detection of the low-quality keyframe, we perform the simple evaluation metric first in order to save the computation time.

2) Keyframe replacement

At the keyframe replacement stage, we replace the low-quality keyframe with a high-quality frame in the corresponding shot. For each frame in the shot we evaluate the frame according to the priority, and then pass-score is recorded. If not passed, the following metrics are not used. For example, suppose we evaluate 3 metrics in the order of the image contrast evaluation, the motion evaluation, and the face detection. If a frame passes the image contrast evaluation, but fails to pass the motion evaluation, then the face detection is not evaluated. And the pass-score is 1. After all frames are evaluated, we select the best frame with the highest priority metric among the highest pass-score frames. **Algorithm 1** is the pseudocode of the priority filter.

Algorithm 1. Priority filter

```
Problem: Selecting high-quality keyframes
Input: A video consisting of shots, priority_evaluation_metrics
Output: keyframes
// Initialize the variables
INITIALIZE keyframes[total number of shots] to NULL, low_quality_keyframes[total number of shots] to FALSE

// Selection of initial keyframes
FOR  shot_index=1 to total number of shots
   STORE the first frame of shots[shot_index] to keyframes[shot_index]
END FOR

// Detection of low-quality keyframes
FOR shot_index=1 to total number of shots
   FOR each evaluation_metric in priority_evaluation_metrics
      IF low_quality_keyframes[shot_index] is FALSE THEN
         IF evaluate_keyframe( keyframes[shot_index], evaluation_metric) is low-quality THEN
            low_quality_keyframes[shot_index] = TRUE
         END IF
      END IF
   END FOR
END FOR

// Change low-quality keyframes to high-quality keyframes
FOR shot_index=1 to total number of shots
   IF low_quality_keyframes[shot_index] is TRUE
      shot = shots[shot_index]
      frame_length = total number of frames in shot
```

```
      INITIALIZE pass_scores[frame_length] to 0
      FOR frame_index=1 to frame_length
        FOR each evaluation_metric in priority_evaluation_metrics
          IF evaluate_keyframe(shot[frame_index], evaluation_metric) is high-quality THEN
            INCREMENT pass_scores[frame_index]
          ELSE
            BREAK
          END IF
        END FOR
      END FOR
      max_value = max(pass_scores) // Find the maximum value of pass_scores
      // In the shot, select the frame whose pass_scores is the max_value as the keyframe.
      keyframes[shot_index] = best_frame(pass_scores, max_value)
    END IF
END FOR
RETURN keyframes
```

3.1.3.2 How to set priority filter

The priority filter can easily and flexibly set various evaluation metrics, but requires user's judgment on priority setting. As a criterion for determining the priority of the evaluation metrics, it is possible to consider the genre and attribute of the video, the performance according to the metrics, and the computation time. Basically, it is appropriate to apply the metrics appropriate to the genre or the attribute of the video. It is also necessary to consider the performance and computation time of the metrics in practical terms. For example, in the case of drama, face detection is one of the significant metrics [24], but when the F-measure of the algorithm is relatively low and the computation time is long, it may be better not to use it or to set the priority lower. In addition, you can choose to use only light evaluation metrics depending on the available time.

When there are several evaluation metrics to be considered, there is a tendency that more keyframes that satisfy the high priority evaluation metrics are included. This is in a tradeoff relationship with the keyframe rate of the subordinate evaluation metrics. For example, if two evaluation metrics are set in a different order, approximately 60-70% of the images will obtain similar quality results, and the rest will have better keyframes in terms of the higher priority evaluation metric. On the other hand, the ratio of keyframes considering the subordinate evaluation metric is relatively small. This occurs because some shots do not satisfy multiple evaluation metrics at the same time. In addition, although it is advantageous in terms of execution time that the evaluation metric which takes a long time was less prioritized, the difference obtained from the experiment is not more than 2% on average.

3.2 Optimization of computation time

In many VOD applications, keyframes of a video need to be provided right after the broadcast ends. So all the processing for keyframe selection should be done in, for example, 10 minutes for one-hour video contents. In this paper, we optimize the computation time by using downsampling and frame sampling for fast keyframe selection because the computation time of the priority filter is related to the processing time for each frame and the number of frames. For the face detection and the caption detection, it takes more than 1 hour to process at the original resolution. Therefore, downsampling and frame sampling are necessary for fast

processing. **Fig. 3** shows graphs of F-measures according to resolution change. **Fig. 4** shows graphs of computation time according to resolution change. In the case of the drama, the caption detection is omitted because there is no caption. In the case of the news, the face detection is not important, so it is omitted. Also the results of the caption detection of the original resolution are omitted because it takes too much computation time. Before we compute F-measure, the ground truth low-quality keyframes for each evaluation metric have been selected subjectively by 5 graduate students.

As shown in **Fig. 3 (a)** and **(b)**, the image contrast evaluation and the similarity evaluation do not substantially degrade the performance, even if downsampling is performed. However, the performance of the face detection and the caption detection decreases slightly with decreasing resolution. As shown in **Fig. 4 (a)** and **(b)**, the computation time is relatively short in the image contrast evaluation, the similarity evaluation, and the motion estimation, and the computation time gradually decreases as the resolution is reduced. Nevertheless, the face detection and the caption detection require a long computation time and the computation time can be significantly reduced by downsampling.

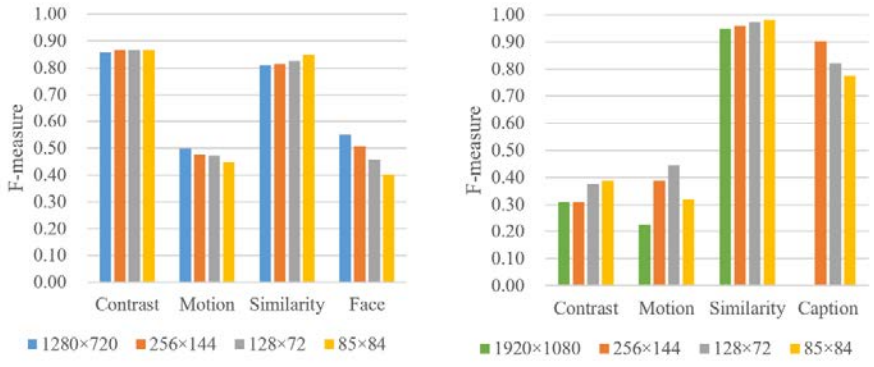

Fig. 3. F-measure changes by downsampling. (a) F-measure of the drama, (b) F-measure of the news.

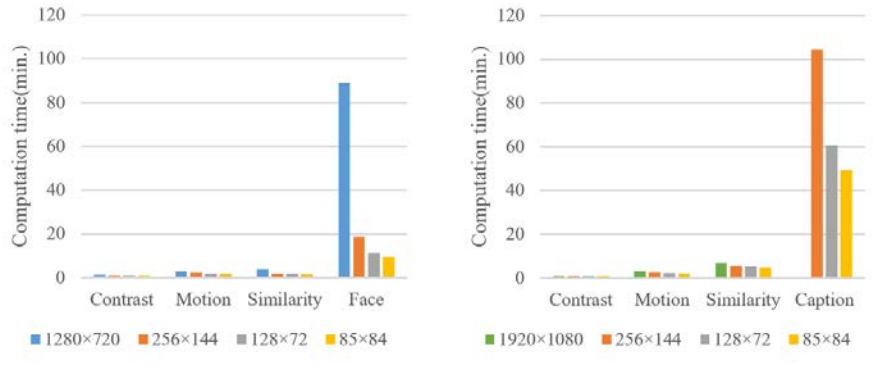

Fig. 4. Computation time changes by downsampling. (a) computation time of the drama, (b) computation time of the news.

When we replace a low-quality keyframe with a high-quality frame in the shot, we apply frame sampling to save computation time. **Fig. 5** compares frame sampling at every frame and at 15 frame intervals at a resolution of 128 × 72. The computation time for face detection was

reduced from 11.2 min to 5.5 min, and the computation time of the caption detection was reduced significantly from 60.7 min to 14.8 min. The evaluation at the keyframe selection stage is applied from the fast evaluation metric to the slow evaluation metric. If it does not pass the quick evaluation metric, it is judged to be the replacement target, so that it is possible to omit the inspection of the slow evaluation metric, thereby reducing the computation time. At the time of keyframe replacement, the test is limited to the frames satisfying the evaluation successively, thereby reducing the number of examinations of evaluation metrics with lower priority.

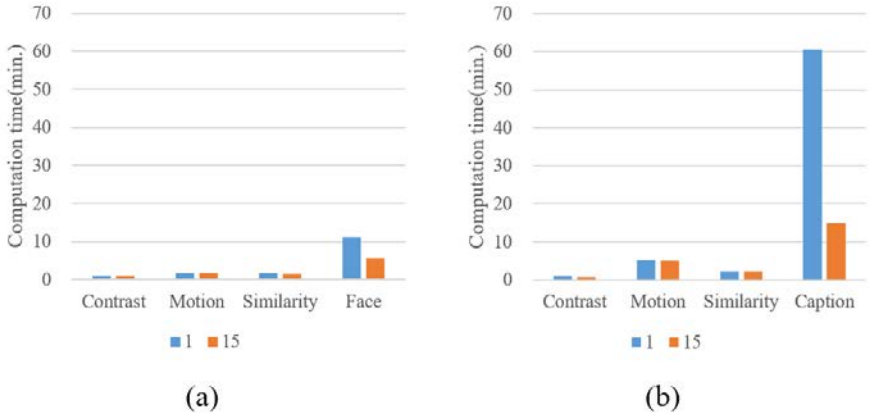

Fig. 5. Computation time according to frame sampling. 1 and 15 are the sampling intervals of the frame. (a) computation time of the drama, (b) computation time of the news.

Considering the computation time, it is necessary to lower the resolution and to increase the frame sampling interval. However, the face detection and the caption detection are degraded as downsampling is performed. Therefore, it is appropriate to do frame sampling at longer intervals rather than do too much downsampling. Through such a method, it is possible to perform keyframe selection within a short time by reducing the computation time.

The time complexity of the proposed priority filter is $O(n)$, where n is the total number of frames in the video. Assume that the number of evaluation metrics is e, the probability of low-quality keyframe is p, the number of keyframes is K, and the sampling interval is s. In this case, the best-case time complexity is $B(n) = K * e$, when all initial keyframes are satisfactory. The average-case time complexity is $A(n) = (K + p * n/s) * e$, and the time complexity depends on the probability of low-quality keyframe. The worst-case time complexity is $W(n) = (K + n/s) * e$, where all initial keyframes are unsatisfactory but all the remaining frames meet the evaluation criterion.

4. Experiments

Our experiments use the dataset of **Table 2**. The dataset consists of a drama and a news video of about 1 hour each. First video is the drama “Descendants of the Sun,” Episode 1, and second video is the news “MBC Newsdesk.” We have extracted approximately 500 shots, depending on the requirements of service. We used Windows 10 64-bit, Intel® i7 3.40-GHz CPU, RAM 16 GB, and GTX1080 for the experiment.

To evaluate of the proposed algorithm, we compare the initial keyframes and the resulting keyframes of the proposed algorithm. 23 people voted on the improvement or the disimprovement of the results. The shot boundary detection stage is not included in the computation time.

Table 2. Datasets

Genre	Drama	News
Video name	KBS, Descendants of the Sun – episode 1	MBC, Newsdesk – episode 10343
Resolution	1280×702	1920×1080
Total duration	59 min. 15 s.	56 min. 44 s.
Numbers of shots	492	494
Numbers of frames	106559	102047

4.1 Drama

For the drama, we used the image contrast evaluation and the similarity evaluation. **Fig. 6** shows the performance evaluation results. Of the initial keyframes, 22% of the keyframes were replaced, and 49% of the replaced keyframes have shown improvements, 20% have shown disimprovements, and 32% have shown similar quality.

Fig. 7 shows examples of improvement in image contrast. Low contrast initial keyframes were selected for replacement and replaced with higher quality images than the initial keyframes. Although only the contrast evaluation and the similarity evaluation were used, the higher quality frames were selected as the keyframe.

Fig. 8 shows an example of reducing similarity. A similar frame was repeatedly selected as a keyframe, but the similar keyframe was replaced with a frame with a low similarity using the similarity evaluation.

Fig. 9 shows example of disimprovement. The proposed keyframe of shot 242 does not express the specific situation of the shot because only one person's close-up frame is selected.

In the drama, to reduce the computation time, the resolution of the frames was downscaled to 128 × 72 and the frame sampling was performed at intervals of 15 frames. As a result, the computation time has been reduced from 2.24 minutes to 1.24 minutes, which is 44.5% reduction.

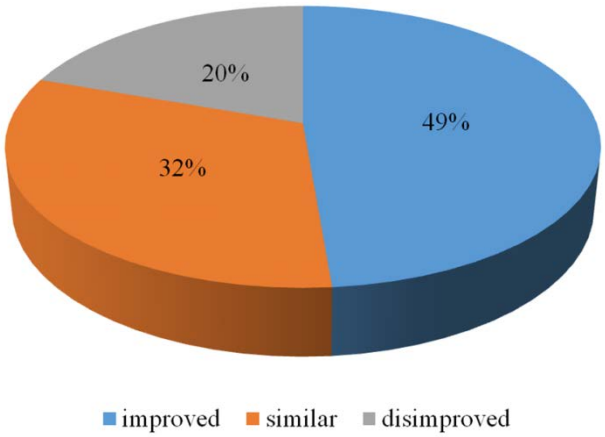

Fig. 6. Change of keyframe quality in drama

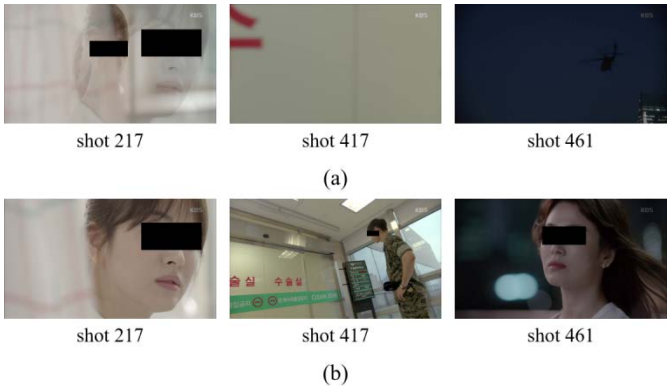

Fig. 7. Example of improvement by image contrast. (a) initial keyframe, (b) proposed keyframe.

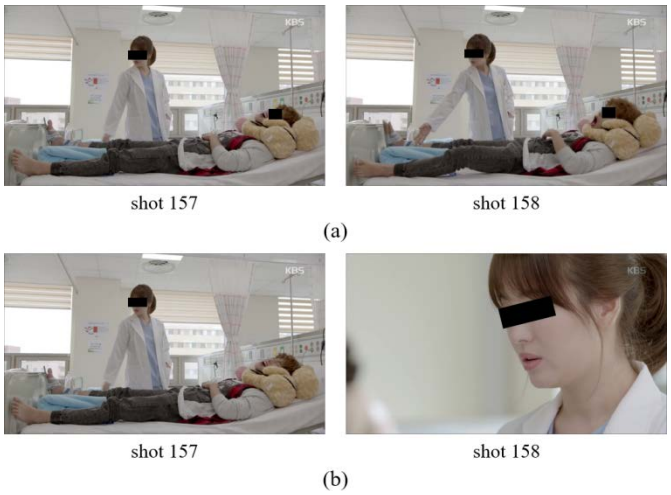

Fig. 8. Example of improvement by reducing similarity. (a) initial keyframe, (b) proposed keyframe.

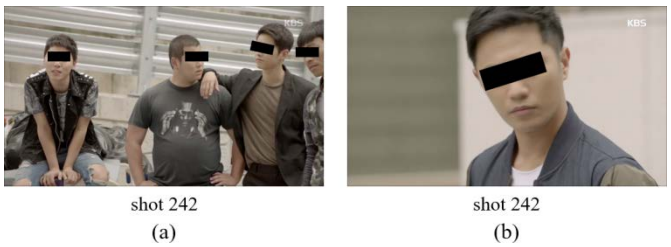

Fig. 9. Example of disimprovement. (a) initial keyframe, (b) proposed keyframe.

4.2 News

For the news, the caption detection and the image contrast evaluation were applied as evaluation metrics. **Fig. 10** shows the performance evaluation result. 54% of the initial keyframes were replaced, and 42% of the replaced keyframes have shown improvement, 15% have shown disimprovement, and 42% have shown similar quality.

Fig. 11 shows examples of improvement by caption evaluation. It can be confirmed that the keyframes without captions are replaced with the keyframes containing the captions, which provide more important information for news.

In the news, to reduce the computation time, the resolution of the frames was downscaled to 256 × 144 and the frame sampling was performed at intervals of 30 frames. As a result, the execution time of 10 hours was reduced to 11.12 minutes, which is about 50 times faster.

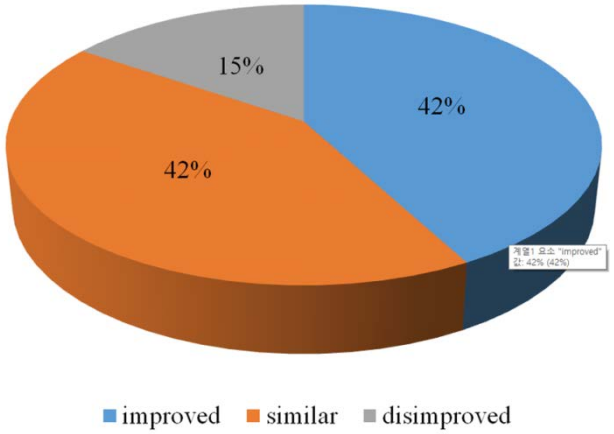

Fig. 10. Change of keyframe quality in news

Fig. 11. Example of improvement by caption evaluation. (a) initial keyframe, (b) proposed keyframe.

5. Conclusion

In this paper, we propose an improved quality keyframe selection method for HD video with less amount of computation time than existing methods. Five keyframe evaluation metrics are defined, the priority filter algorithm is proposed, and the performance in terms of keyframe quality and execution time is analyzed. In our experiment, the proposed algorithm replaces 22% of keyframes for about 1 hour of drama with 49% improvement and 20% disimprovement. The execution time is about 1.24 minutes, which is 44.5% reduction. The proposed algorithm replaces 54% of keyframes for 1 hour news with 42% improvement and 15% disimprovement. The execution time was about 11.12 minutes, which is about 50 times faster, compare to the one without the proposed scheme.

Future research needs to improve the algorithm's performance, such as improvement of detection rate of the face detection and improvement of the caption detection speed. It is also necessary to acquire various data and to carry out experiments through various genres to determine the optimum priority metrics for each video genre.

References

[1] K. W. Kim, "A new framework for automatic extraction of key frames using DC Image activity," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 12, pp. 4533-4551, Dec. 2014. Article (CrossRef Link).

[2] K. W. Kim, "An efficient implementation of key frame extraction and sharing in android for wireless video sensor network," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 9, pp. 3357-3376, Sep. 2015. Article (CrossRef Link).

[3] G. Guan, Z. Wang, S. Lu, J. D. Deng, and D. D. Feng, "Key-based keyframe selection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 4, pp. 729-734, Apr. 2013. Article (CrossRef Link).

[4] C. R. Kumar and S. Suguna, "Visual semantic based 3D video retrieval system using HDFS," *KSII Trans. on Internet and Information Systems*, vol. 10, no. 8, pp. 3806-3825, Aug. 2016. Article (CrossRef Link).

[5] K. W. Sze, K. M. Lam, and G. Qiu, "A new key frame representation for video segment retrieval," *IEEE Trans. Circuits Syst. Video Technol*., vol. 15, no. 9, pp. 1148-1155, Sep. 2005. Article (CrossRef Link).

[6] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial LSTM networks," in *Proc. of The IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 2982-2991, Nov. 2017. Article (CrossRef Link).

[7] R. Panda and A. K. Roy-Chowdhury, "Collaborative summarization of topic-related videos," in *Proc. of The IEEE Conf. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 4274-4283, Jul. 2017. Article (CrossRef Link).

[8] W. T. Chu, C. H. Yu, and H. H. Wang, "Optimized comics-based storytelling for temporal image sequences," *IEEE Trans. Multimedia*, vol. 17, no. 2, pp. 201-215, Feb. 2015. Article (CrossRef Link).

[9] S. K. Kuanar, K. B. Ranga, and A. S. Chowdhury, "Multi-view video summarization using bipartite matching constrained optimum-path forest clustering," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1166-1173, Aug. 2015. Article (CrossRef Link).

[10] Y. Zhang and R. Zimmermann, "Efficient summarization from multiple georeferenced user-generated videos," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 418-431, Mar. 2016. Article (CrossRef Link).

[11] Y. Li, T. Zhang, and D. Tretter, "An overview of video abstraction techniques," *HP Lab., Tech. Rep*. HPL-2001-191, vol. 6, pp. 1-23, 2001. Article (CrossRef Link).

[12] Y. Tonomura and S. Abe, "Content oriented visual interface using video icons for visual database systems," *J. Vis. Lang. Comput*., vol. 1, no. 2, pp. 183-198, 1990. Article (CrossRef Link).

[13] W. Wolf, "Key frame selection by motion analysis," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process*, vol. 2, pp. 1228-1231, May. 1996. Article (CrossRef Link).

[14] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognit*., vol. 30, no. 4, pp. 643-658, Apr. 1997. Article (CrossRef Link).

[15] M. Mentzelopoulos and A. Psarrou. "Key-frame extraction algorithm using entropy difference," in *Proc. of ACM SIGMM Int. Workshop on Multimedia Inf. Retr*., pp. 39-45, Oct. 2004. Article (CrossRef Link).

[16] Q. Xu, Y. Liu, X. Li, Z. Yang, J. Wang, M. Sbert, and R. Scopigno, "Browsing and exploration of video sequences: a new scheme for key frame extraction and 3D visualization using entropy based Jensen divergence," *Inf. Sci. (Ny)*, vol. 278, pp. 736-756, Sep. 2014. Article (CrossRef Link).

[17] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *Proc. of Int. Conf. Image Proc*., vol. 1, pp. 866-870, 1998. Article (CrossRef Link).

[18] S. K. Kuanar, R. Panda, and A. S. Chowdhury, "Video key frame extraction through dynamic Delaunay clustering with a structural constraint," *J. Vis. Commun. Image R*., vol. 24, no. 7, pp. 1212-1227, Oct. 2013. Article (CrossRef Link).

[19] J. M. Lee, H. S. Yang, Y. S. Moon, S. H. Kim, and S. J. Kim, "Effective shot-based keyframe selection by using image quality evaluation," in *Proc. of Int. Workshop Advanced Image Technology (IWAIT)*, pp. 176.1-176.3, Jan. 2018. Article (CrossRef Link).
[20] S. Park, J. Son, and S. Kim, "Effect of adaptive thresholding on shot boundary detection performance," in *Proc. of IEEE Int. Conf. Consumer Electronics-Asia (ICCE-Asia)*, Oct. 2016. Article (CrossRef Link).
[21] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of 7th Int. Joint Conf. Artificial Intelligence (IJCAI)*, vol 2, pp. 674-679, Aug. 1981. Article (CrossRef Link).
[22] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis*., vol. 57, no. 2, pp. 137-154, May 2004. Article (CrossRef Link).
[23] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 3538-3545, Jun. 2012. Article (CrossRef Link).
[24] Z. Rasheed and M. Shah, "Scene detection in Hollywood movies and TV shows," in *Proc. of The IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 1-6, Jun. 2003. Article (CrossRef Link).

Hyeon Seok Yang received his B.S. degree in the Department of Electronics and Information Engineering from Yeungnam University, Korea, in 2010. He received the M.S. degrees in the Department of Computer Science & Engineering from Hanyang University, Korea, in 2012. He is studying for his PhD. degree in the Department of Computer Science & Engineering from Hanyang University, Korea. His research interests include computer vision, pattern recognition, and deep learning.
Email : hsyang@visionlab.or.kr

Woojin Jeong received the B.S degree in the Department of Computer Science and Engineering from Hanyang University, Korea, in 2012. He is currently working towards PhD. Degree at the Department of Computer Science and Engineering from Hanyang University, Korea, From 2012. His research interests include computer vision and machine learning.
Email : wjjeong@visionlab.or.kr

Jong Min Lee received the B.S., M.S., and Ph.D. degrees in computer science and engineering from Hanyang University, Republic of Korea, in 2007, 2009, and 2015, respectively. From 2015 to 2016, he had been a Post-Doctoral Fellow in the Department of Computer Science & Engineering, Hanyang University, Korea. From 2016 to 2018, he had been a Research Professor in the Department of Computer Science & Engineering, Hanyang University, Korea. In 2018, he joined Hanwha Land Systems, Korea, as a senior research engineer. His research interests include computer vision, image processing, object detection/tracking, and pattern recognition.
Email : jm0608.lee@hanwha.com

Seung-Hee Kim received the B.S. and M.S. degrees in Electronic Engineering from Korea University, Korea, in 1982 and 1988 respectively. Since 1982, she has been a principal researcher at the Electronics and Telecommunication Research Institute(ETRI), Daejeon, Korea. Her research interests include open smart broadcast service platform, context-aware based personalization service, smart TV, and broadcast content metadata.
Email : seung@etri.re.kr

Sun-Joong Kim received her BS degree in computational statistics and her MS degree in computer science from Chungnam National University, Daejeon, Rep. of Korea, in 1989 and 2000 respectively. In February 1989, she joined ETRI, Daejeon, Rep. of Korea, where she is currently principal researcher and Project Leader. Her research interests include media service platform and content knowledge mining.
Email : kimsj@etri.re.kr

Young Shik Moon received the B.S. and M.S. degrees in Electronics Engineering from Seoul National University and Korea Advanced Institute of Science and Technology, Korea, in 1980 and 1982, respectively, and PhD. degree in Electrical and Computer Engineering from the University of California at Irvine, CA, in 1990. From 1982 to 1985, he had been a researcher at the Electronics and Telecommunication Research Institute, Daejon, Korea. In 1992, he joined the Department of Computer Science and Engineering at Hanyang University, Korea, as an Assistant Professor, and is currently a Professor. Dr. Moon served as General Chair of 2014 IEEE International Symposium on Consumer Electronics, and worked as the President of the Institute of Electronics and Information Engineer, Korea.
Email : ysmoon@hanyang.ac.kr