Near-duplicate Video Detection Algorithm Based on Global GSP Feature and Local ScSIFT Feature Fusion

Xidao Luan, Yuxiang Xie, Jingmeng He, Lili Zhang, Chen Li and Xin Zhang

1College of Computer Engineering and Applied Mathematics, Changsha University, Changsha 410003, China
2College of Information Systems and Management, National University of Defense Technology, Changsha 410073, China

*Email: yxxie@nudt.edu.cn

Abstract. The main problem with near-duplicate video detection is the high computational complexity and the low efficiency. Near-duplicate video detection methods based on global feature is running fast but with low accuracy, on the contrary, methods based on local feature is accurate, but the calculation is large and time-consuming. Therefore, a near-duplicate video detection algorithm combining global GSP feature and local ScSIFT feature is proposed. Firstly, the video clips and the query ones are discretized into a set of key frame sequences, and the temporal information is recorded at the same time. Secondly, by filtering the video clips on global Gaussian-Scale pyramid feature, similar video clips are selected as the candidates to assure the high recall. Then, combined with the temporal features of the keyframes, the candidate videos are further detected by the local ScSIFT feature to obtain a higher precision. Experimental results show that the proposed algorithm can improve the accuracy of near-duplicate video detection on the basis of guaranteeing the timeliness of the algorithm.

1. Introduction
Near-duplicate video is generally the same from the original video but is compressed to different quality, redefined for different sizes and frame rates, or in space and time to make different edits to modify the videos. In literature [1], by statistics of YouTube's near-duplicate video in the amateur production and professional production, it finds that the most popular video are mostly produced by the professional. Professional production of video is more easily to be copied and modified; obviously in these shared video can find a considerable number of near-duplicate video clips. Therefore, many research scholars have begun to pay attention to how to quickly and effectively remove near-duplicate videos in the video search results. The core technology is near-duplicate video detection, which has already had a lot of mature algorithm. Near-duplicate video detection mainly focuses on two factors like the detection accuracy and efficiency of the algorithm. These two factors usually have inversely proportional relationship; higher detection accuracy will usually bring a higher computational burden, but if the algorithm is fast, it will obviously reduce the accuracy of algorithm detection.

Judging the similarity of video content based on keyframes is a commonly used method. This type of methods is usually measured by the number of matching similar keyframes. At present, most of the near-duplicate keyframe detection research is based on visual features, or the fusion of other modal features. Early researches only consider the matching of video data attributes. These methods are
simple, but the video similarity is calculated only by the proportion of the number of similar keyframes in the videos. Since the temporal information of the video keyframe is ignored, the similarity of near-duplicate video clips is not consistent with human cognitive. In [2], a large amount of redundant video is eliminated by clustering the similar video clips, thus the efficiency of near-duplicate video detection is improved. In literature [3], videos which are generated by different camera shooting angle are also considered in the detection, and a near-duplicate video detection system called UQLIPS is established according to the temporal information of the keyframe. The system can detect the similar video clips quickly and effectively. [4] proposes a method of matching video for internal changes (such as different shooting angles, different camera motions, etc.).

In the related work of near-duplicate shot detection, the researchers are more concerned with the performance of video sequence matching [5]. For example, literature [6] uses the editing distance to match the video. However, this method is only applicable to small video data sets. The algorithm proposed in [7] and [8] detects the feature points of keyframes by Harris corner detection algorithm, establishes a 20-dimension feature vector with temporal information for each feature point, and searches for similar feature points by Hilbert curves with sublinear search complexity. The algorithm can be applied to both near-duplicate image and video detection, but the algorithm is less real-time.

From the above analysis, it can be seen that the main disadvantage of near-duplicate video detection based on keyframes is that the amount of calculation is so large that the algorithm is inefficient. Therefore, it is possible to improve the efficiency of the algorithm by reducing the amount of test keyframe data, reducing the number of feature points and the dimension of the feature vector in each keyframe. Near-duplicate video detection based on the global feature of the keyframe is fast but inaccurate; however, near-duplicate video detection based on local feature of the keyframe is accurate but takes too long. This makes it possible to fuse two features in near-duplicate video detection. Combined global gray-scale pyramid feature with SIFT feature based on sparse coding acceleration, a near-duplicate video detection algorithm which is based on fusion of global GSP and local ScSIFT feature called GLC-NDVD is proposed.

2. GLC-NDVD Algorithm

In the GLC-NDVD algorithm, global feature selection of the keyframe is the color histogram feature, which is the most commonly used and fast to be calculated feature. In order to avoid the influence of illumination and clipping on the keyframe image, the gray-scale pyramid is established for the keyframe, and the local feature adopts the ScSIFT. The algorithm first discretizes the video clips and the query video into a set of keyframe sequences, and records the temporal information of the keyframe at the same time. Second, the global GSP feature is used to do coarse filter in the query library, with a higher recall to ensure that the keyframe similar to the video can be included in the candidate set. Then, combined with the temporal feature of the key sequence, the candidate set is further detected by the local ScSIFT feature to obtain a higher detection precision. The algorithm can quickly improve the efficiency of the algorithm by filtering the dissimilar video based on the GSP feature and reducing the amount of the candidates by local feature detection.

2.1. Process of the algorithm

Near-duplicate video detection algorithm based on the global GSP feature and the local ScSIFT feature is divided into two parts, one is the off-line index extraction and construction of query video library features, the other is the real-time video detection. The algorithm of off-line index extraction and construction of query video library features is a combination of gray-scale pyramid construction algorithm and SIFT feature extraction algorithm based on sparse coding. The basic process is as follows:

1. Discretize the query video into a keyframe sequence and record its temporal information;
2. Establish the gray-scale pyramid of the query video keyframes;
3. Extract the color histogram of the gray-scale pyramid image in each keyframe;
4. Extract the SIFT feature of each keyframe, sparse all SIFT features with the sparse dictionary...
D, save the ScSIFT feature $\alpha$, and build the index index;
(5) End.

The basic process of real-time video detection algorithm is as follows:
(1) The color histogram of the test keyframe is extracted, and the video candidate set with similar histogram of keyframe is selected by comparing with the gray-scale pyramid feature of the query library image;
(2) Read the ScSIFT features $\alpha_c$ of all keyframes in the candidate set, and the ScSIFT feature index $\text{index}_c$ of the candidate set;
(3) Extract SIFT feature of the test video keyframe, sparse represent the extracted SIFT feature by sparse dictionary $D$, and record it as $\beta$;
(4) Set the similarity distance threshold of the ScSIFT feature, keyframe similarity threshold, video similarity threshold, and read the first column of ScSIFT feature $x = \beta_0$ of the test video keyframe;
(5) According to the index $\text{index}_c$, query the nearest $k$ column coefficients $\alpha_{ck}$ of the ScSIFT feature $x$;
(6) Calculate the distance $d_{ck}$ between $x$ and each column $\alpha_{ck}$, if $d_{ck} \leq \sigma$, the feature matching amount $\Sigma_i$ of the column which belongs to keyframe $i$ will add 1. The distance can be calculated as follows:

$$
\|\alpha_x - \alpha_y\|_2 = \sqrt{\sum_i \alpha_{xi}^2 + \sum_j \alpha_{yi}^2 + \sum_k (\alpha_{xki} - \alpha_{yki})^2}
$$

(1)

Where $i$ is the element ordinal for those value non-zero in vector $\alpha_x$ and value zero in vector $\alpha_y$.
Similarly, $j$ is the element ordinal for those value zero in vector $\alpha_x$ and value non-zero in vector $\alpha_y$. $k$ is the element ordinal for those value non-zero in both vector $\alpha_x$ and $\alpha_y$.

(7) Repeat steps (4), (5) until all columns of ScSIFT feature $\beta$ are cycled;
(8) For each feature matching amount $\Sigma_i \neq 0$ of image $i$, the total number of sparse feature points $\text{Total}_i$ is counted and the similarity degree $\Sigma_i / \text{Total}_i$ is calculated. If $\Sigma_i / \text{Total}_i \geq \theta$, keyframe $i$ in the query library is a similar key frame for the test image;
(9) If the number of similar video frames of the two videos exceeds the proportion $\delta$ of the number of test keyframes, then the first $n$ video with the most similar temporal information is selected as the near-duplicate video detection result;
(10) End.

2.2. GSP feature extraction
In order to solve the shortcomings of the color histogram on the change of illumination, clipping and poor distinction in images, the gray-scale pyramid (GSP) is built on the idea of establishing Gaussian scale pyramid in SIFT algorithm. Through the scale transform of each keyframe, the pyramid model is constructed by making the gray scale transformation of each key frame at different scales. Unlike Gaussian pyramid, the scale of the gray-scale pyramid achieves different scales by continuously cutting the image area of the fixed image in the current image to ensure that the color features of the image are invariant to the cropping. In this paper, the current image centre height is 70% of the original image area as the next scale image. The specific process of GSP feature extraction algorithm is described in [9].

2.3. ScSIFT feature extraction
SIFT feature acceleration detection algorithm based on sparse coding (ScSIFT) focuses on the sparsifying of SIFT feature vectors. After SIFT feature extraction, the algorithm does sparse coding for
the 128 dimension SIFT feature vector and establishes the sparse coding index to improve the matching speed. The main idea is using the SIFT feature extracted from the keyframe of the query library as a training sample to train the over-complete dictionary to obtain a set of over-complete bases. The SIFT feature vector of the keyframe in the query library is sparsely encoded with the over-complete dictionary, and the sparse feature vector is indexed. Furthermore, the SIFT feature of the test keyframe is matched with the feature index of the query frame by using the over-complete dictionary to obtain a set of similar candidate sets, compare the sparse coefficient between the test keyframe and the candidate, and finally get similar keyframe detection results. The sparse coefficient vector which is sparsely coded from SIFT feature is called ScSIFT feature of the keyframe.

3. Experimental results and analysis
In order to test the effectiveness of GLC-NDVD algorithm, a near-duplicate video detection algorithm based on global GSP feature and a near-duplicate video detection algorithm based on ScSIFT feature are selected as the control. The experimental query library contains 1000 clips of video, and a total of 10816 frames are extracted. The original video clips and 10 video clips which have been performed some editing styles like Gaussian blur, adding logo, gray scale transformation and scale cropping are selected as the test video. The near-duplicate detection is carried out with the query library video, and the experimental results are compared with each other. The results of three near-duplicate video detection algorithms based on different features are shown in Table 1.

|                | Recall (%) | Precision (%) | Run time (s) |
|----------------|------------|---------------|--------------|
| GLC GSP ScSIFT | GLC GSP ScSIFT | GLC GSP ScSIFT | GLC GSP ScSIFT |
| Original video | 98.9% 72.2% 98.9% | 92.7% 96.5% 49.7% | 29.41 4.29 192.37 |
| Gaussian blur  | 98.9% 72.2% 99.1% | 93.6% 96.5% 50.4% | 30.34 4.29 193.98 |
| Adding Logo    | 98.6% 55.5% 98.2% | 92% 96.9% 49.3% | 30.75 4.37 188.23 |
| Gray+1         | 97.1% 82.1% 98.7% | 78.3% 96.7% 48.5% | 22.92 4.40 183.55 |
| Gray+2         | 17.4% 43.7% 98.8% | 2.6% 13.8% 46.7% | 14.40 4.05 172.40 |
| Gray-1         | 92.8% 28.6% 99.2% | 20.2% 24.7% 49.6% | 30.74 4.26 184.91 |
| Gray-2         | 95.4% 40.0% 99.2% | 25.9% 27.1% 48.4% | 25.25 4.22 174.83 |
| Scale 0.7      | 98.6% 51.3% 98.9% | 76.2% 98.2% 49.2% | 40.18 4.36 141.55 |
| Scale 0.5      | 81.5% 44.3% 98.9% | 73.5% 99.3% 44.5% | 25.29 4.60 97.06 |
| Scale 0.35     | 82.2% 47.1% 99.0% | 69.8% 91.5% 28.9% | 15.54 4.40 61.94 |
| Average        | 86.1% 53.7% 98.9% | 62.5% 74.1% 46.5% | 26.48 4.32 159.08 |

Figure 1 shows the results corresponding to Table 1. In the figure, the abscissa indicates the precision or recall rate. "*", "o" and "□" refers to the test results of GLC algorithm, GSP algorithm and ScSIFT algorithm respectively.

![Figure 1. Precision and recall of three algorithms with different transformations.](image-url)
The average precision and recall rate of three algorithms are shown in Figure 2.

![Figure 2. Average accuracy and recall of three algorithms.](image)

In Figure 2, the abscissa "1", "2" and "3" refer to the average precision of GLC algorithm, GSP algorithm and ScSIFT algorithm, and are drawn by solid line. "4", "5" and "6" refers to the average recall rate of the three algorithms, drawn in dashed lines.

In contrast to Table 1, Figure 1 and Figure 2, it shows that the similarity of near-duplicate video detection based on the GLC feature is similar to the ScSIFT feature for most editing transformations, and is only obviously lower than ScSIFT feature for the transform like gray scale increase of 64. The average precision rate of near-duplicate video detection based on GLC feature is more than 85%. At the same time, the precision rate based on GLC feature is slightly lower than GSP, with an average of 62.5%, which is significantly higher than that of ScSIFT, which is 46.5%. As to the efficiency, the average run time of GLC algorithm is 26.48 seconds, 4.32 seconds slower than the GSP algorithm, but significantly better than ScSIFT feature which is 159.08 seconds. Therefore, GLC algorithm can achieve a higher level of accuracy with an average of more than 85%, recall rate is slightly less, but still around 62%. The efficiency of GLC algorithm is not the best, but about 25 seconds of running time is still in acceptable range. In summary, the GLC algorithm can ensure recall and precision at the same time, while ensures running fast. Therefore, near-duplicate detection algorithm based on the global GSP feature and local ScSIFT feature fusion is more effective than the other two detection algorithms based on single feature.

4. Conclusions
Aiming at the problem that near-duplicate video detection method based on global feature is fast but inaccurate and the detection method based on local feature is accurate but time-consuming, a near-duplicate video detection algorithm combining global GSP feature and local ScSIFT feature is proposed. The algorithm combines the high efficiency of global GSP features with the high accuracy of local ScSIFT features. Experiments show that the proposed algorithm has improved the detection algorithm of using only single feature. In the future, we will further study near-duplicate video detection based on video clips to improve the detection efficiency.

Acknowledgments
This work has been supported by the National Natural Science Foundation of China (61571453), Hunan Provincial Department of Education Science and Technology Program (15A020) and Changsha Science and Technology Project (ZD1601014).
References

[1] Kruitbosch G and Nack F 2008 *Proc. 3rd ACM Int. Workshop on Human-Centered Computing (Vancouver)* (ACM) pp 7-10

[2] Wu X, Ngo C W, Hauptmann A G and Tan H K 2009 *IEEE Trans. Multimedia* 11 196

[3] Shen H T, Zhou X, Huang Z, Shao J and Zhou X 2007 *Proc. Int. Conf. on Very Large Data Bases (Vienna)* (the United States: VLDB Endowment) pp 1374-1377

[4] Wu P H, Thaipanich T and Kuo C J 2009 *Proc. Int. Conf. on Image Processing (Cairo)* (New York: IEEE) pp 237-240

[5] Huang Z, Shen H T, Shao J, Cui B and Zhou X 2010 *IEEE Trans. Multimedia* 12 386

[6] Bertini M, Bimbo A D and Nunziati W 2006 *Proc. Int. Conf. on Image and Video Retrieval (Tempe)* vol 4071 (Berlin: SPRINGER-VERLAG BERLIN) pp 133-142

[7] Buisson J O and Frélicot C 2007 *IEEE Trans. Multimedia* 9 293

[8] Sarkar, Singh V, Ghosh P, Manjunath B S and Singh A 2010 *IEEE Trans. Circuits and Syst. Video Technol.* 20 870

[9] Xiao Niu, Yuxiang Xie, Chen Li and Xidao Luan 2016 *Int. Conf. on Signal and Image Processing (Beijing)* (New York: IEEE) pp 103-107