Effective Video Copy Detection Using Statistics of Quantized Zernike Moments

Jiehao Chen, Chenglong Chen, and Jiangqun Ni(✉)

School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510006, People's Republic of China
{chenjiehao818,c.chenglong}@gmail.com,
issjqni@mail.sysu.edu.cn

Abstract. Video copy detection has found wide applications in digital multimedia forensics and copyright protection. With video copy detection, one can not only determine the presence of a query video in the massive video database, but also locate it precisely. This paper presents an effective video copy detection scheme based on the statistics of quantized Zernike moments. In our approach, each video frame is partitioned into non-overlapping blocks. The Zernike moments of first few orders are then calculated for each block. Finally, the frame-level feature is generated by aggregating statistics of the quantized Zernike moments of all the blocks in the video frame. Through extensive experiments on a public video database, this frame-level feature is demonstrated to be robust against geometric transformation, color adjustment, noise contamination and many other commonly used content-preserving operations. Compared with existing schemes in the literatures, the proposed method yields better or at least comparable performance in a series of experiments.

Keywords: Digital multimedia forensics · Video copy detection · Zernike moment

1 Introduction

In recent years, with the advance of network technology, digital videos are easily published and shared on the Internet and become increasingly popular in our daily life. Due to the availability of powerful video editing tools, the number of video copies also grows explosively, which results in not only a waste of network bandwidth and storage space, but also the infringement of intellectual property rights. Therefore, it is vital to develop the effective method for video copy detection to ensure the protection of intellectual property rights.

In general, the existing schemes for video copy detection can be classified into two categories, i.e., the active watermarking based [1] and the passive content based video copy detection. The watermarking based scheme requires additional information to be embedded into the video contents prior to distribution, which

Y.Q. Shi et al. (Eds.): IWDW 2013, LNCS 8389, pp. 232–243, 2014.
DOI: 10.1007/978-3-662-43886-2_17,

can then be extracted to establish ownership upon request. For a long time of the past, the watermarking based approach was the mainstream for video copy detection. However, it is almost prohibitively expensive and time-consuming to embed watermarks into the emerging massive video contents. As a result, the applications of watermarking based approaches are greatly restricted. On the other hand, content based copy detection schemes do not require any extra information but the video itself. The motivation behind content based copy detection is that each video content has its own unique fingerprint, which can be extracted for video copy detection [2]. In this paper, we focus on the design of content based copy detection method.

The objective of video copy detection is not only finding to which video in the database a query video belongs, but also locating the query video precisely, which is a quite challenging task. There are two primary concerns in the design of video copy detection scheme, i.e., robustness and discriminability. Robustness refers to the scheme's capability to tolerate content-preserving operations that give rise to distortions, including changes in color, illumination, display format, as well as different geometric transformations. Discriminability enables the scheme to distinguish between videos with different contents such that false detections can be minimized. It is noted that the higher the robustness or discriminability is, the more accurate the result of video copy detection becomes.

Several effective schemes for video copy detection have been proposed. In [3], the ordinal measure was first exploited as a fingerprint for video sequence matching. Later, Kim et al. in [4] improved the ordinal measure by further employing the spatial ordinal measure in video copy detection. This method works quite well in detecting copies of display format transformations, especially letter-box and pillar-box operations. In [5], the temporal ordinal measure was incorporated which yields better performance than the spatial ordinal measure in [4]. The ordinal measures above, however, fail to detect copies of some geometric transformations, such as rotation and flipping. As a result, color-based methods are developed for video copy detection, such as color histogram intersection [6], color correlation histogram [7], amongst others. These color-based approaches can resist most geometric transformations, because they do not rely on any spatial information within the video frames. The color-based methods, however, become less reliable when some color adjustments are applied, which can significantly change the color distributions in the frame. In addition, some local features [8–10] are also adopted in video copy detection. While they are more robust to various distortions, local features are more memory and computation demanding.

In this paper, an effective scheme based on the statistics of quantized Zernike moments is developed for video copy detection. In our approach, each video frame is firstly divided into non-overlapping blocks. Then, the Zernike moments of first few orders are extracted for each block in the frame. The frame-level feature is finally constructed by a set of histograms of the quantized Zernike moments. Extensive experiments are carried out which demonstrate the robustness of the proposed feature set against many common content-preserving operations,

especially scaling, flipping, Gaussian filtering and color adjustment. The superiority of our method for video copy detection is also verified in a series of experiments.

The rest of this paper is organized as follows. In Sect. 2, we will give a brief review of the basic properties of Zernike moments. Section 3 details the proposed method, which is followed by the experimental results and analyses, including a comparison with previous arts in Sect. 4. The last section summarizes the paper.

2 The Zernike Moments

The Zernike moments [11] are widely used in the areas of digital forensics, computer vision and multimedia processing. In this work, we propose to use Zernike moments for video copy detection. In the following, we will briefly review the basic properties of Zernike moments.

Let (ρ, θ) $(0 \leq \rho \leq 1, 0 \leq \theta \leq 2\pi)$ be the polar coordinate of point on the unit disc. For a continuous image function $f(\rho, \theta)$, the Zernike moment of order n with repetition m is defined as:

$$Z_{n,m} = \frac{n+1}{\pi} \int_0^{2\pi} \int_0^1 f(\rho, \theta) R_{n,m}(\rho) e^{-jm\theta} \rho \mathrm{d}\rho \mathrm{d}\theta \tag{1}$$

where n is a non-negative integer, and m is an integer such that $n - |m|$ is even and non-negative. The quantity $R_{n,m}(\rho)$ in the above formula is the radial Zernike polynomial which is defined as follow:

$$R_{n,m}(\rho) = \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s [(n-s)!] \rho^{n-2s}}{s!(\frac{n+|m|}{2} - s)!(\frac{n-|m|}{2} - s)!} \tag{2}$$

Note that low order Zernike moments represent the global shape of an image and are very stable. While on the other hand, high order moments correspond to the details and are quite sensitive to disturbance. In the experimental part (i.e., Sect. 4.2), we will discuss about the order of the Zernike moments to use for feature construction.

The Zernike moments exhibit some nice properties, such as rotation and shift invariance, which make it robust against attacks like geometric transformation. To demonstrate the Zernike moments' invariance to rotation for example, we herein consider an image I and its rotated counterpart $I^{'}$. Therefore, we obtain $I^{'}(\rho, \theta) = I(\rho, \theta - \alpha)$ with α being the angle of rotation. According to Eq. (1), the Zernike moments of $I^{'}$ is given as:

$$Z^{'}_{n,m} = \frac{n+1}{\pi} \int_0^{2\pi} \int_0^1 I(\rho, \theta - \alpha) R_{n,m}(\rho) e^{-jm\theta} \rho \mathrm{d}\rho \mathrm{d}\theta \tag{3}$$

Let $\theta^{'} = \theta - \alpha$, we obtain

$$\begin{aligned} Z^{'}_{n,m} &= \frac{n+1}{\pi} \int_0^{2\pi} \int_0^1 I(\rho, \theta^{'}) R_{n,m}(\rho) e^{-jm(\theta^{'}+\alpha)} \rho \mathrm{d}\rho \mathrm{d}\theta^{'} \\ &= \frac{n+1}{\pi} e^{-jm\alpha} \int_0^{2\pi} \int_0^1 I(\rho, \theta^{'}) R_{n,m}(\rho) e^{-jm\theta^{'}} \rho \mathrm{d}\rho \mathrm{d}\theta^{'} \\ &= Z_{n,m} e^{-jm\alpha} \end{aligned} \tag{4}$$

As shown, rotation only leads to a phase shift on the corresponding Zernike moments, and the magnitude of Zernike moments $|Z_{n,m}|$ remains unchanged after rotation. Therefore, $|Z_{n,m}|$ is invariant to rotation. In addition, Zernike moment is also robust against noise. With such nice properties, we adopt a set of Zernike moments as features to represent a video frame.

3 Proposed Method

In this section, we will describe the proposed method for video copy detection in detail. First, we show how to construct features for each video frame based on Zernike moments. Then, we use the features for the purpose of video copy detection.

3.1 Feature Extraction for Video Frame

(1) Preprocessing: Before extracting the frame-level based features, we need to preprocess the video frame.

a. Color transformation: Videos are made up of a sequence of video frames. Nowadays, almost all of video frames are color images, which consist of three or four color components. We transform color video frames into grayscale video frames and extract features from the pixel intensities of the images.

b. Border removing: In some videos of special format, there are two black bars placed on sides of video frame, e.g., letter-box and pillar-box videos. These black bars themselves don't contain any useful information for copy detection, so they are removed in advance.

c. Normalization: Each video frame is normalized to the same size beforehand. With lots of experiments, we choose to normalize the video frame to the fixed size of resolution 320×320.

d. Block splitting: Each video frame is partitioned into non-overlapping blocks of size $b \times b$ for further feature extraction.

(2) Zernike moments calculation: For the resulting video frame, we extract a L dimensional feature vector from each block of the frame. This feature vector consists of first n orders of Zernike moments (with n being a parameter of our method). The relationship between n and L is given in Table 1.

(3) Feature representation: To simplify the calculation of detection in the later period, Zernike moments are quantized and stored using histograms. For each dimension of the extracted Zernike moments, all the samples from all the

Table 1. The relationship between the feature dimension L and the order n of Zernike moments.

n	m	L
0	0	1
1	1	2
2	0, 2	4
3	1, 3	6
4	0, 2, 4	9
5	1, 3, 5	12
6	0, 2, 4, 6	16
7	1, 3, 5, 7	20

divided blocks of the video frame are aggregated into 8 bins to obtain a histogram. We therefore obtain L histograms in total. Finally, we obtain an $8 \times L$ dimensional descriptor to characterize a video frame.

3.2 Video Copy Detection

To perform video copy detection, we need some metric to measure how similar two videos are. To that end, we first define the distance between two video frames represented by features described above. Denote the normalized histograms of the quantized Zernike moments as $H_i(j)$, where $1 \leq i \leq L$, $1 \leq j \leq 8$. We should note that

$$\sum_{j=1}^{8} H_i(j) = 1 \tag{5}$$

Then, the distance between two video frames can be defined as

$$d(v_q, v_t) = \frac{1}{C} \sum_{i=1}^{L} \sum_{j=1}^{8} \left| H_{q,i}(j) - H_{t,i}(j) \right| \tag{6}$$

where $d(v_q, v_t)$ is normalized distance between two video frames and C is the normalized factor. In our case, we use $C = 2L$ to ensure the distance between any two video frames $d(v_q, v_t) \in [0, 1]$.

We now adopt the above definition of distance between two video frames for the purpose of video copy detection. Let $V_q = [v_q^1, v_q^2, \cdots, v_q^M]$ denote a query video with M frames and $V_t = [v_t^1, v_t^2, \cdots, v_t^N]$ denote an original video clip with N frames in the database. In this work, we perform video copy detection using the matching algorithm presented in [12,13]. Specifically, we first build a distance matrix through pair-wise distance of the frames based the histograms of Zernike moments. The distance matrix between query video V_q and original video clip V_t is given as:

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1N} \\ d_{21} & d_{22} & \cdots & d_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{M1} & d_{M2} & \cdots & d_{MN} \end{bmatrix} \tag{7}$$

where d_{ij} is the distance between the frame v_q^i and the frame v_t^j. Then Hough transform is applied on distance matrix to detect if the query video is a copy of the original video clip.

4 Experimental Results

In this section, experimental study is carried out to demonstrate the efficacy of the proposed method. In Sect. 4.1, we first describe the experimental setup and performance metric. Next, we discuss about the parameter settings for the proposed method. Finally, we compare the performance of the new approach with previous arts in Sects. 4.3 and 4.4.

4.1 Experimental Setup

For a quantitative evaluation of the proposed method, we conduct a series of experiments. To that end, we make use of MUSCLE VCD benchmark [14], which contains 101 videos of a total length equal to about 80 h. These videos' sources are various, such as web video clips, TV archives, movies, and et.al. Since the method in [7] operates with video represented in the RGB model, we further transform these videos into red, green, and blue channels. From 20 to 200 s with a step of 10 s, we randomly extract 95 query videos from the database with 5 query videos for each query length. To resemble the scenarios in practical applications, we apply various commonly used operations, such as geometric transformation, blurring, and color adjustment, to each query video. Specifically, there are twenty six modifications as listed below:

- Scaling: Scale each frame by factors of 0.5, 0.8, 1.2, 1.5 and 2.
- Flipping: Flip each frame horizontally and vertically.
- Cropping: Crop outer region of each frame by 5 %, 10 %, 15 % and 20 %.
- Rotation: Rotate each frame by angles of 90, 180 and 270 degrees.
- Letter-box: Transform aspect ratio of each frame to 4 : 3 by placing black bars above and below the frame.
- Pillar-box: Transform aspect ratio of each frame to 16 : 9 by placing black bars on the left and right sides of the frame.
- Gaussian filtering: Filter size: 5×5 and 9×9 with the standard deviation 3.
- Average filtering: Filter size: 5×5 and 9×9.
- Gaussian noise: Add white Gaussian noise with PSNR of 20dB and 25dB.
- Color adjustment: Select one of the three color channels randomly and then modify the values by multiplying ratios of 0.8 and 1.2.
- Color phase shift: Consider the following two kinds of color phase shift:

$$\begin{array}{lll} R' = G & & R' = B \\ G' = B & \text{and} & G' = R \\ B' = R & & B' = G \end{array}$$

where R, G, B are the three color channel components of each frame in the original video, and R', G', B' are those of the modified counterpart.

As described above, we obtain 26 near-duplications in total for each given sample. For evaluation of video copy detection schemes, we use the near-duplications of original samples as query clips. From the description above, we note that each query clip has exactly one positive matched sequence in the video database. The goal of video copy detection is therefore to determine the presence of a query video in the database and also locates the query video precisely.

To evaluate the performance of the proposed method, we use the true positive rate (TPR) and the false positive rate (FPR):

$$TPR = \frac{\text{number of correct detected copies}}{\text{total number of copies}} \tag{8}$$

$$FPB = \frac{\text{number of false detected copies}}{\text{total number of non-copies}} \tag{9}$$

Note that higher TPR means stronger robustness, and smaller FPR means better discriminability [7]. Furthermore, the receiver operating characteristics (ROC) curve is also adopted to illustrate the performances of different methods or the same method with different parameters.

4.2 Parameter Selection

We herein first discuss the parameters of the proposed method in the feature extraction phase, i.e., the block size b and the order n for Zernike moments. Through lots of experiments, we achieve a desirable performance with $b = 16$. Furthermore, this block size also conforms to some popular block-based video compression standards. Therefore, we use $b = 16$ for the proposed method in the following experiments.

We then proceed to determine the order n of the Zernike moments used in the feature extraction. Figure 1 shows the ROC curves of different order n, i.e., $n = 1,\ 2,\ 3,$ and 4. It is observed that the proposed method achieves similar performance with $n = 2,\ 3,$ and 4. While the performance of $n = 1$ degrades slightly, it is still quite satisfactory to some extent, e.g., $TPR > 99\,\%$ when $FPR = 0.5\,\%$. To trade-off the performance and the computation/storage complexity, we choose $n = 2$ (resulting in 32-dimensional feature for each video frame) in the following experiments.

4.3 Performance Comparison and Analysis I

To verify the effectiveness of the proposed method, we compare it with several popular methods for video copy detection. To that end, we take three

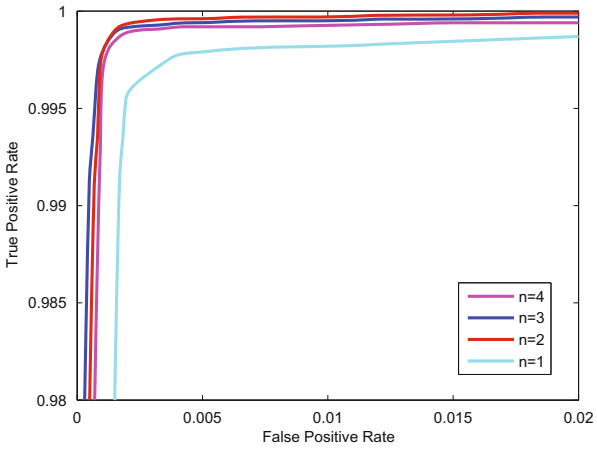


Fig. 1. *ROC* curves of the proposed method using different orders of Zernike moments.

state-of-the-art methods as benchmarks, i.e., spatial ordinal measure based method (SOM) [4], temporal ordinal measure based method (TOM) [5], and color correlation histogram based method (CCH) [7]. These methods have been reported performing quite well in video copy detection.

We first compare the overall performances of the four methods for detecting the query clips described in Sect. 4.1. Figure 2 shows the corresponding *ROC* curves for this comparison. From the figure, we can observe that the *TPR* values of the proposed method are higher than the rest three methods for a given *FPR* in most cases. This indicates that our method is more robust to various common content-preserving operations. The reason for the difference in the *TPR* values is that the other methods are not robust against some specific video operations, as will be discussed below.

In Table 2, we further present the detection results of the involved methods for each single transformation. The performance is evaluated through the following procedure. First, we set $FPR = 0.1\,\%$ and acquire the corresponding threshold for each single transformation. Next, according to the threshold, we calculate the *TPR*, as shown in Table 2. It is observed that SOM and TOM perform quite similarly and yield almost 100 % correct detection for most cases. However, these two methods fail to handle geometric distortion, like flipping and rotation. Note that SOM and TOM mainly employ spatial structure as features. Therefore, they are not quite robust against some operations, e.g., rotation and flipping, that will significantly change the spatial information in frames.

On the other hand, CCH method can perfectly survive those geometric attacks and yields almost 100 % detection accuracy. This could be expected since it does not rely on any information of the spatial structure within the frames. Note that CCH exploits the color correlation among different color channels as features. These features are not robust against some operations, e.g., color adjustment and color phase shift, since they may significantly change the color

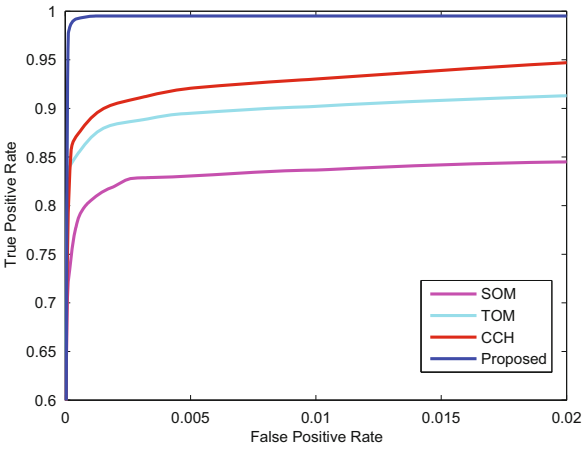


Fig. 2. *ROC* curves of different video copy detection methods.

information in frames. Therefore, CCH is very sensitive to such color transformations, as also indicated in Table 2. We want to note that these color modifications are also widely used, specifically for image/video enhancement. For example, it is quite often to adjust the color channel of an image in Photoshop. Furthermore, as will be discussed in Sect. 4.4, camcording may also significantly modify the color components of the original videos (see Fig. 4). Thus, the color attacks we have considered can be adopted to simulate the camcording operation to some extent.

As reported in Table 2, the proposed method performs quite well for all the content-preserving operations we have considered. Specifically, it can survive both geometric distortion and color modification. This makes the proposed method an alternative and promising tool for video copy detection.

4.4 Performance Comparison and Analysis II

In the experiments above, we have demonstrated the effectiveness of the proposed method for detecting video copies that have undergone various commonly used operations. In this part, we will further evaluate the performance of our method using the two tasks provided by the MUSCLE VCD benchmark. Task One (ST1) evaluates a system's capability to find the whole copies in the database, and contains 15 query videos with a total length over 2 h and 30 min. For Task Two (ST2), video clips from 21 different videos in the database are inserted into 3 query videos. This latter task aims to locate the inserted query clips within the database. Both tasks are challenging because various transformations have been operated on the query videos, e.g. change of color/brightness, blurring, recording with an angle, inserting logos/subtitles, etc. Fig. 3 shows some clips after and before transformations for these two different tasks.

Table 2. The *TPR* of different methods for each distortion with $FPR = 0.1\,\%$.

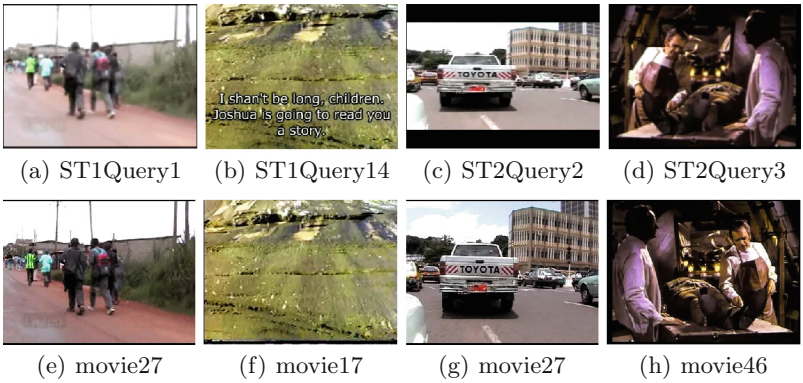
Distortion Types	SOM	TOM	CCH	Proposed
Scaling	100	100	100	99
Flipping	(0)	(26)	100	100
Cropping	89	99	99	95
Rotation	(0)	(27)	100	100
Letter-box	100	100	100	100
Pillar-box	100	100	100	100
Gaussian filtering	99	100	100	100
Average filtering	99	100	100	100
Gaussian noise	100	100	95	99
Color adjustment	100	100	(40)	100
Color phase shift	100	100	(0)	100

Table 3. The detection performances of different methods for ST1 and ST2.

Task	SOM	TOM	CCH	Proposed
ST1	$\frac{12}{15}$	$\frac{13}{15}$	$\frac{11}{15}$	$\frac{15}{15}$
ST2	NA	NA	NA	$\frac{15}{21}$

Table 3 shows the detection results of different video copy detection schemes for these two tasks. There, the performance metric represents the ratio of correctly detected copies. For ST1, it is observed that the proposed method achieves a perfect result (100 %) and outperforms the other three previous schemes. While on the other hand, the rest three methods fail to detect several query videos. This is because these methods are not robust against some specific transformations. For example, CCH cannot survive color modification (e.g., color adjustment and color phase shift) as indicated in Table 2. Therefore, it is not surprising that CCH fails to detect the four videos shown in Fig. 4, as these videos have undergone some kind of color modification. However, judged by human inspection, these four query videos should be considered as near-duplications of the original videos, as they all contain the same contents/objects.

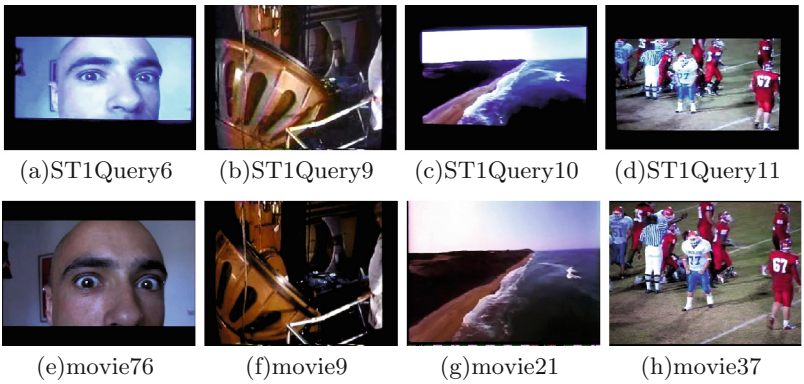
For ST2, the SCOV method [12] so far gets the best result and can correctly detect 19 query clips. As shown in Table 3, the proposed method also yields satisfying detection performance for ST2, with 15 query clips being correctly detected. This indicates that the Zernike moments based features could be refined further in future work. However, compared with SCOV method, the feature dimension of our proposed method is lower than SCOV method and our method requires less storage space. Note that methods like SOM, TOM, and CCH cannot handle the temporal operations in ST2.



(a) ST1Query1 (b) ST1Query14 (c) ST2Query2 (d) ST2Query3

(e) movie27 (f) movie17 (g) movie27 (h) movie46

Fig. 3. Illustrations of some video transformations in ST1 and ST2. (a), (b), (c) and (d) have undergone the operation of blurring, inserting subtitles, letter-box and flipping, (e)–(h) are the corresponding ground truth of (a)–(d), respectively.



(a)ST1Query6 (b)ST1Query9 (c)ST1Query10 (d)ST1Query11

(e)movie76 (f)movie9 (g)movie21 (h)movie37

Fig. 4. Illustrations of the four failed detected clips in ST1 by CCH method. (a), (c) and (d) have undergone the similar operation of camcording. (b) has undergone the color phase modification and color adjustment, (e)–(h) are the corresponding ground truth of (a)–(d), respectively.

5 Conclusion

In this paper, we present a novel and effective approach for video copy detection based on the statistics of quantized Zernike moments. Extensive experiments are carried out, which demonstrate the strong robustness and good discriminability of the proposed method. Compared with traditional ordinal measure based methods and color based methods, our method exhibits better performance for detecting the copies with commonly used content-preserving operations, especially geometric distortion and color adjustment. The capability of the proposed method against temporal attacks is also satisfactory.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (61379156), the National Research Foundation for the Doctoral Program of Higher Education of China (20120171110037), and the key Program of Natural Science Foundation of Guangdong (S2012020011114).

References

1. Lee, S.J., Jung, S.H.: A survey of watermarking techniques applied to multimedia. In: Proceedings ISIE, Industrial Electronics, vol. 1, pp. 272–277 (2001)
2. Hampapur, A., Bolle, R.M.: Comparison of distance measures for video copy detection. In: Proceedings of the 2001 IEEE International Conference on Multimedia and Expo (ICME), pp. 737–740 (2001)
3. Mohan, R.: Video sequence matching. In: Proceedings of the 1998 IEEE International Conference on Speech and Signal Processing, vol. 6, pp. 3697–3700 (1998)
4. Kim, C., Vasudev, B.: Spatiotemporal sequence matching for efficient video copy detection. IEEE Trans. Circuits Syst. Video Technol. **15**(1), 127–132 (2005)
5. Chen, L., Stentiford, F.W.M.: Video sequence matching based on temporal ordinal measurement. Pattern Recogn. Lett. **29**(13), 1824–1831 (2008)
6. Swain, M.J., Ballard, D.H.: Color indexing. Int. J. Comput. Vis. **7**(1), 11–32 (1991)
7. Lei, Y.Q., Luo, W.Q., Wang, Y.G., Huang, J.W.: Video sequence matching based on the invariance of color correlation. IEEE Trans. Circuits Syst. Video Technol. **22**(9), 1332–1343 (2012)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**(2), 91–110 (2004)
9. Maani, E., Tsaftaris, S.A., Katsaggelos, A.K.: Local feature extraction for video copy detection in a database. In: Proceedings of 15th IEEE International Conference on Image Processing, pp. 1716–1719 (2008)
10. Liu, Z., Liu, T., Gibbon, D.C., Shahraray, B.: Effective and scalable video copy detection. In: Proceedings of the 2010 International Conference on Multimedia Information Retrieval, pp. 119–128 (2010)
11. Teague, M.R.: Image analysis via the general theory of moments. J. Opt. Soc. Am. **70**(8), 920–930 (1980)
12. Zheng, L.G., Qiu, G.P., Huang, J.W., Fu, H.: Salient covariance for near-duplicate image and video detection. In: Proceedings of 18th IEEE International Conference on Image Processing, pp. 2537–2540 (2011)
13. Zheng, L.G., Lei, Y.Q., Qiu, G.P., Huang, J.W.: Near-duplicate image detection in a visually salient riemannian space. IEEE Trans. Inf. Forensics Secur. **7**(5), 1578–1593 (2012)
14. Law-To, J., Joly, A., Boujemaa, N.: Muscle-VCD-2007: a live benchmark for video copy detection. http://www-rocq.inria.fr/imedia/civr-bench