International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE) Vol 1, Issue 1, Dec 2014

Video Sequence Matching Based On the Invariance of Colour Correlation With SVM

[1] Renjini.R,[2]Kavitha.N.Nair
[1]PG Scholar, Department of ECE, University College of Engineering, Muttom, Kerala – 685587
[2]Lecturer, Department of ECE, University College of Engineering, Muttom, Kerala - 685587
[1]renjiniremesan04@gmail.com

Abstract - **Video sequence matching aims to locate a query video clip in a video database. It plays an important role in reducing storage redundancy and detecting video copies for copyright protection. Here, we propose an effective method for video sequence matching based on the invariance of color correlation. The proposed method first splits each key frame into non overlapping blocks. For each block, we sort the red, green, and blue color components according to their average intensities, and use the percentage of the color correlation to generate a frame feature with a small size. Finally, the resulting video feature is made up of the consecutive frame features, which is demonstrated to be robust against most typical video content-preserving operations, including geometric distortion, blurring, noise contamination, contrast enhancement, and strong re-encoding. The experimental results show that the proposed method outperforms the existing methods in the literature, as well as the method based on the traditional color histogram. Use of Multi-SVM will increase the accuracy of video sequence matching. Furthermore, space complexity of our algorithm is satisfactory, which are very important for many real time applications.**

Keywords- **Color correlation; content-based technology;video sequence matching; Support Vector Machine(SVM) .**

I. INTRODUCTION

With the rapid advancement of multimedia processing and Internet techniques, the amount of digital content available to users has become very large and widespread. Today, it is common to observe many copies of digital media on the Internet, which inevitably leads to a waste of storage resources and problems related to copyright infringement. Therefore, it is desirable to have a method of detecting such copies effectively. Currently, there are two ways to deal with the problem that is the watermark-based technique and content-based technology.

The watermark-based methods [1] can be classified as an active technology since it embeds imperceptible information into the media so that it is protected prior to distribution. These methods include media fingerprints that facilitate copy detection at a later time. However, in many real applications, such side information is not available to the detectors for the following reasons. First, embedded watermarks can easily be destroyed by malicious users [2]. In addition, the embedded watermarks may be damaged significantly after certain types of media processing, such as geometric distortion and severe compression [3]. In such cases, the water mark based methods would fail at detecting copies. On the other hand, content-based technology can be regarded as a passive method since it does not need any extra information, such as a watermark, for detection. By measuring the similarity of some robust features between questionable media and those in the database, we can find the corresponding copies. Here, we focus on the content-based technology for video sequence matching.

Video sequence matching aims to locate the copies of a given query clip from a video database. Here the word "copy" does not mean the exactly same version of the original sequence, but its transformed ones [4] after content-preserving operations, such as geometric distortion, blurring, noise contamination, contrast enhancement, and re-encoding. Usually, the robustness of the video feature against the above-mentioned operations is one of the important issues for video sequence matching. The discriminability, as well as the computational and storage complexity, also needs to be considered in the sequence matching algorithms. Here, discriminability refers to the ability of feature to distinguish video clips with different content. The computational complexity denotes the time for generating the feature, which needs to be sufficiently low, while the storage complexity, which indicates the size of the extracted feature, should also be as small as possible. Usually, there would be a tradeoff among these characteristics when designing a proper algorithm to meet the real-world requirements.

Although many content-based methods have been proposed for image copy detection [5], these image-based methods cannot be extended to detect video copies and sequence matching directly since they usually lead to high storage complexity and demanding computation for feature extraction. The temporal ordinal measurements of video frames were reported. Compared with spatial ordinal measurements [6],[7] the temporal measurement can achieve better performance regarding the shifting and insertion of pattern. Subsequently, Gao et al [8] reported a principal component analysis (PCA) - based approach in the scaled luminance field to support video sequence retrieval. Moving Picture Experts Group (MPEG) also developed some video signature descriptors for the fast and robust detection of near-duplicate videos [9]. Based on our extensive experiments, however, the robustness of the above-mentioned global features against rotation and flipping operations is still poor. Local features like SIFT [10] and CS-LBP [11] are also used to detect sequence copies. Although the local features can handle many challenging distortions, their computational and storage complexity is unacceptably high. In addition, some trajectory-based techniques [12],[13] have also been proposed for video sequence matching. However, these methods mainly focus on temporal distortions, such as frame deletion and insertion. Furthermore, these methods are usually expensive for the matching phase, since the trajectories must be aligned first. In recent years, video content identification has seen great advancement, which can be applied to video sequence matching. For example, we can also employ effective methods from video hashing [14] for the purpose of video sequence matching.

A new classification system based on statistical learning theory [15], called the support vector machine [16] has recently been applied to the problem of remote sensing data classification [17]. This technique is said to be independent of the dimensionality of feature space as the main idea behind this classification technique is to separate the classes with a surface that maximise the margin between them, using boundary pixels to create the decision surface. The data points that are closest to the hyperplane are termed "support vectors". The number of support vectors is thus small as they are points close to the class boundaries [15]. One major advantage of support vector classifiers is the use of quadratic programming, which provides global minima only. The absence of local minima is a significant difference from the neural network classifiers. Like neural classifiers, applications of SVMs to any classification problem require the determination of several user-defined parameters. Some of these parameters are the choice of a suitable multiclass approach, choice of an appropriate kernel and related parameters, determination of a suitable value of regularisation parameter and a suitable optimisation technique. SVMs were initially developed to perform binary classification; though, applications of binary classification are very limited. Most of the practical applications involve multiclass classification, especially in remote sensing land cover classification. A number of methods are present to implement SVMs to produce multiclass classification. Most of the research in generating multiclass support vector classifiers can be divided in two categories. One approach involves in constructing several binary classifiers and combing their results while other approach considers all data in one optimization formulation.

SVM are based on statistical learning theory and have the aim of determining the location of decision boundaries that produce the optimal separation of classes [15]. In the case of a two-class pattern recognition problem in which the classes are linearly separable the SVM selects from among the infinite number of linear decision boundaries the one that minimises the generalisation error. Thus, the selected decision boundary will be one that leaves the greatest margin between the two classes [15], where margin is defined as the sum of the distances to the hyperplane from the closest points of the two classes. This problem of maximising the margin can be solved using standard Quadratic Programming (QP) optimisation techniques. The data points that are closest to the hyperplane are used to measure the margin; hence these data points are termed ‘support vectors’. Consequently, the number of support vectors is small. If the two classes are not linearly separable, the SVM tries to find the hyperplane that maximises the margin while, at the same time, minimising a quantity proportional to the number of misclassification errors. The trade-off between margin and misclassification error is controlled by a user-defined constant. SVM can also be extended to handle non-linear decision surfaces [15].

SVM were initially designed for binary (two-class) problems. When dealing with multiple classes, an appropriate multi-class method is needed. Vapnik suggested comparing one class with the others taken together. This strategy generates *n* classifiers, where *n* is the number of classes. The final output is the class that corresponds to the SVM with the largest margin, as defined above. For multi-class problems one has to determine *n* hyperplanes. Thus, this method requires the solution of *n* QP optimization problems, each of which separates one class from the remaining classes. This strategy can be described as ‘one against the rest’.

Originally, SVMs were developed to perform binary classification. However, applications of binary classification are very limited especially in remote sensing land cover classification where most of the classification problems involve more than two classes. A number of methods to generate multiclass SVMs from binary SVMs have been proposed by researchers and is still a continuing research topic. In this paper we use multi SVM for accurate video sequence matching.

The remainder of this paper is organized as follows. Section 2 explains about the proposed algorithm for video

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE) Vol 1, Issue 1, Dec 2014

sequence matching. Section 3 discusses the experimental results. Finally, conclusion and suggestion for future works are given in Section 4.

II. PROPOSED ALGORITHM

In this section, we introduce the concept of color correlation in the red, green, and blue (RGB) space for an image. We then propose a novel feature from a video frame based on the invariance of the color correlation [18]. We will also analyze the robustness of the proposed feature against some typical content-preserving operations. Finally, we use the consecutive frame features for the purpose of video sequence matching.

A. Colour Correlation in an Image

A digital color image (or video frame) is usually represented as a tuple of numbers, which are typically three or four values of color components [e.g., RGB, cyan, magenta, and yellow (CMY), cyan, magenta, yellow, and black (CMYK), and YCbCr color models]. These color models are employed in different applications. For instance, CMY and CMYK are mainly used for color printing. YCbCr (Y is the brightness component, while Cb and Cr such as MPEG-2 and H.264/AVC). In this paper, we choose the RGB color model, and consider the color correlation for it. The reasons for choosing the RGB model are as follows

1. The RGB color model can easily be transformed into and from other color models.
2. The color correlation in the RGB color model is more robust against most content preserving operations.

Color correlation is defined as the arrangement of red, green, and blue color components in order of intensity. For a color image (or a video frame) of size $w\times h$, we use a tuple of numbers $(R_{xy}, G_{xy}, B_{xy})$to represent the intensities of red, green and blue components of the pixel green, and blue components of the pixel at the coordinates (x, y) within the input image.

The color correlation of the three color components would satisfy one of the following six cases:

case ≠1: $R_{xy} \geq G_{xy} \geq B_{xy}$, case ≠2 : $R_{xy} \geq B_{xy} \geq G_{xy}$

case ≠3 : $G_{xy} \geq R_{xy} \geq B_{xy}$, case ≠4 : $G_{xy} \geq B_{xy} \geq R_{xy}$

case ≠5 : $B_{xy} \geq R_{xy} \geq G_{xy}$, case #6 : $B_{xy} \geq G_{xy} \geq R_{xy}$

where $1\leq x \leq w, 1 \leq y \leq h$.

The above color correlation divides the RGB color cube into six subspaces, thus being regarded as a special kind of color histogram (called color correlation histogram). It is well known that the color histogram is one of the commonly used features for representing a color image or video. Compared with other shape-based or texture-based features, the color histogram is very fast to compute and is flexible in terms of storage. Up until now, many features derived from the color histogram have been successfully applied to video retrieval, segmentation, and identification. However, few methods use a single feature from the color histogram for video sequence matching. Here, we must note that since the color correlation histogram is a global feature without any spatial information about the image, the robustness against most common content-preserving operations is expected to be much better. The discriminability of the color correlation histogram may be weak for a single image or video frame. However, it is shown that our feature is very promising for video sequence matching when the length of the video is increased. In order to further show the effectiveness of the proposed feature, we also compare it to the traditional color histogram-based feature for video sequence matching. The comparative results and discussions are shown below.

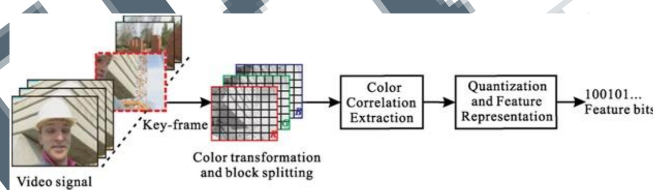


Fig.1. Basic steps of feature extraction for a color frame.

B. Feature Extraction For A Color Frame

Fig.1 shows the process of feature extraction for a color frame, which includes the following three steps.

1) Color Transformation and Block Splitting: The input color frame of size $w\times h$ is first transformed into red, green, and blue channels, which are then divided into $b \times b$ nonover lapping blocks. For each image block, we calculate the average intensities for the red, green, and blue components in order to reduce the effects of noise-like operations, such as blurring and noise contamination. Finally, we obtain a lower resolution image of size $m \times n$, where $m = \left[\frac{w}{b}\right]$, $n = \left[\frac{h}{b}\right]$, and $[x]$ is the nearest integer to x. Based on our experiments, the block size b will significantly affect the performance of the algorithm. Please refer to Fig. 7 for more details. In this paper, we have chosen b = 16.

2) Color Correlation Extraction: For the resulting image with lower resolution $m \times n$, we extract its color correlation. In order to reduce the effect of special cases, we remove those pixels whose red, green, and blue components have the same value. Finally, we calculate the percentage of pixels belonging to their corresponding color correlations, and obtain six normalized real values for each image.

3) Quantization and Feature Representation: The resulting six real numbers are truncated, and the first five numbers are stored in a binary form as the feature for the input image. The above-mentioned process can be formulated as follows. Let P_i denote the set of those pixels whose three color components satisfy the *i*th case of the six color correlations, where $\widetilde{R_{xy}}$, $\widetilde{G_{xy}}$ and $\widetilde{B_{xy}}$ represent the three color components of the lower resolution image, where $1 \leq x \leq m$, $1 \leq y \leq n$ and $1 \leq i \leq 6$

$$P_i = \{(\widetilde{R_{xy}}, \widetilde{G_{xy}}, \widetilde{B_{xy}}) | \widetilde{R_{xy}}, \widetilde{G_{xy}}, \widetilde{B_{xy}} \ s.t.\ case \neq i\} \quad (1)$$

The normalized histogram of the color correlations can then be described as(2), where $|P_i|$ is the cardinality of set P_i

$$H(i) = \frac{|P_i|}{\sum_{i=1}^{6} |P_i|} \quad (2)$$

Please note that

$$\sum_{i=1}^{6} H(i) = 1 \qquad i = 1,2, \dots 6 \qquad (3)$$

Here we employ two significant digits to quantize the real number $H(i)$, and to keep the identity relationship in (3). Therefore, we only need to record the first five values of H in order to save on storage, and the last number can be recovered from the first five numbers. We use seven bits to represent a normalized number with two significant digits. Thus, the feature of each video frame can be denoted with 7 × (6 − 1) = 35 bits.

Fig. 2 shows four typical video frames, and Fig.3.3 shows the corresponding color correlation histograms. It is obvious that their distributions are different since visually distinct frames always contain different color correlations. Taking 'Foreman' for example, case #1 dominates (over 80%), whereas this case is less than 50% for the other three video frames.

Here we use the L1 norm to measure the similarity between two features H_q and H_t. As shown in (4), *C* is the normalization factor. Since the feature is normalized in the range of [0,1], we set *C* = 2 to ensure that the distance between any two features would be in the range of [0,1].

$$d(H_q, H_t) = \frac{1}{C} \sum_{i=1}^{6} |H_q(i) - H_t(i)| \qquad (4)$$

It is expected that the similarity distances for copy pairs are close to 0, whereas for video frames with different contents, the distances would be far from 0. In the examples in Figs.2 and 3, the distances between 'Foreman' and the other three frames are 0.390, 0.585, and 0.765, respectively. Instead of this similarity test we can use Multi-SVM [19].

SVM are based on statistical learning theory and have the aim of determining the location of decision boundaries that produce the optimal separation of classes. In the case of a two-class pattern recognition problem in which the classes are linearly separable the SVM selects from among the infinite number of linear decision boundaries the one that minimizes the generalization error. Originally, SVMs were developed to perform binary classification. However, applications of binary classification are very limited especially in remote sensing land cover classification where most of the classification problems involve more than two classes. A number of methods to generate multiclass SVMs from binary SVMs have been proposed by researchers. Here we use multi-SVM to improve accuracy.

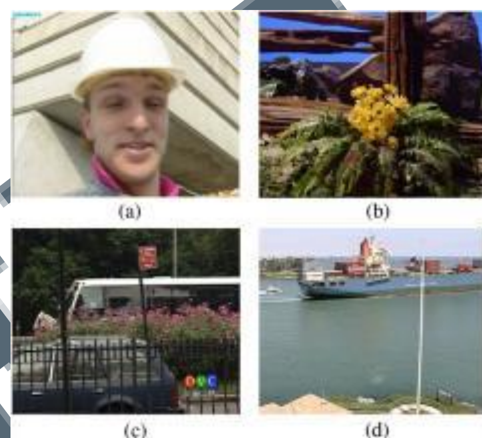


Fig. 2. Four typical videos. (a) Foreman. (b) Tempete. (c) Bus. (d) Container.

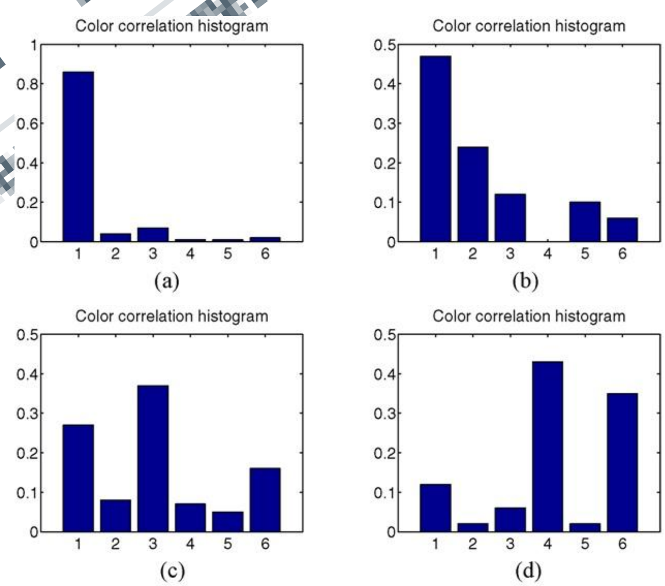


Fig. 3. Distributions of the color correlations of the first frame of the fourtypical videos. (a) Foreman. (b) Tempete. (c) Bus. (d) Container.

C. Robustness Analysis

Robustness refers to the amount that the color correlation histogram will change after some common content-preserving operations. Some examples of such operations are illustrated in Fig.4, which shows the operations will preserve the original content of the video frame. In the following, we will analyze the robustness of the proposed feature against several typical operations.

1) Noise-Like Contamination: Noise-like contamination includes blurring and adding noise, which can significantly change the color correlation of individual pixels. However, such effects can be effectively decreased through averaging (see the first step in the feature extraction).

2) Scaling, Rotation, and Flipping: Scaling operation will change the spatial resolution of the video, but will not significantly change the color correlation histogram for the following reasons. The pixels of the scaled frame can be divided into two parts. The first part consists of the original pixels, the color correlation of which will be well preserved. The second part consists of pixels that are interpolated by the neighboring pixels. Since the interpolation is usually performed independently in the red, green, and blue channels, the color correlation of the interpolated pixel will be approximately the same as the neighboring pixels from the first part. Therefore, the resulting color correlation histogram of the scaled frame will be similar to that of the original one. For pure rotation, redundant pixels with the same values for the three channels will be padded. These redundant pixels will be removed in the feature extraction. Thus, the color correlation histogram undergoes no change for such a rotation. The analysis of scaling and pure rotation can be combined for a similar result for rotation with scaling. Flipping (vertical or horizontal) operations only change the pixel positions, not the values of the pixels, thus the proposed histogram will also be preserved.

3) Letter-Box and Pillar-Box: As illustrated in Fig. 4(a) and (b), letter-box and pillar-box operations occur in the video when black bars are placed on the sides of the video. This is a commonly used operation for modifying the aspect ratio of the video. It should be noted that for both of these operations, only the black pixels are padded. Since such added pixels have the value of 0 for their red, green, and blue components, they will be removed in our feature extraction, thus well preserving the proposed histogram.

4) Cropping and Shifting: As illustrated in Fig. 4(c) and (d), cropping and shifting can be modeled by replacing the original image region with black pixels. Suppose that the ratio of the replaced region over the image is λ, where $0 \leq \lambda \leq 1$. Then, the feature after cropping or shifting operation will change as follows:

$$H_q(i) = \frac{|P_i| - |\alpha_i|}{|P| - \lambda |P|} \qquad (5)$$

where $|\alpha_i|$ is the number of replaced pixels whose red, green, and blue components satisfy case $\neq i$,$|P|=\sum_{i=1}^{6} |P_i|$ and $\sum_{i=1}^{6} |\alpha_i| = \lambda |P|$. The distance between the distorted frame $F_q$and the original frame F_t can be calculated using the following equation:

$$d(H_q, H_t) = \frac{\lambda}{1-\lambda} \qquad (6)$$

Where H_q and H_t are the features of F_q and F_t , respectively. If F_q is a near-duplication of F_t , then the above distance should be less than a threshold T . Thus, we can obtain the following relationship between the ratio λ of the replaced region and the threshold T:

$$\frac{\lambda}{1-\lambda} \leq T \Rightarrow \lambda \leq \frac{T}{T+1} \qquad (7)$$

Based on (7), we can estimate the upper bound of λ for a given threshold T.

5) Insertion of Pattern and Picture in Picture: As illustrated in Fig. 4(e) and (f), these two operations can be modeled by the replacement of an original image region with a given pattern or picture. Similar to the analysis of the cropping and shifting operations, we assume that the ratio of the inserted region over the image is λ, where $0 \leq \lambda \leq 1$, and then the feature after such operations will change as follows:

$$H_q(i) = \frac{|P_i| - |\alpha_i| + |\beta_i|}{|P|} \qquad (8)$$

where $|\alpha_i|$ and $|\beta_i|$ denote the number of elements that satisfy case $\neq i$ in the region before and after insertion, respectively, and $\sum_{i=i}^{6} |\alpha_i| = \sum_{i=1}^{6} |\beta_i| = \lambda |P|$. The distance between H_q and H_t is given in

$$d(H_q, H_t) = \lambda \qquad (9)$$

Similarly, for a given threshold T , we can obtain the upper bound of λ, knowing that it is equal to T for the two operations.

Fig. 4. Illustration of six geometric operations. (a) Letter-box. (b) Pillar-box(c) Cropping. (d) Shifting. (e) Insertion of pattern. (f) Picture in picture.

6) Contrast Enhancement: Contrast enhancement, such as histogram equalization and gamma correction, is also a commonly used operation for image manipulation. Unlike grey-scale images, color images contain color information (e.g., red, green, and blue components) for each pixel. Usually, it is unwise to conduct the same enhancement independently on their color channels, since it will introduce unnatural appearances and visually artifacts in the resulting image. A better solution is to first convert RGB color space to other color spaces, such as HSI, YCbCr, and YIQ, and then perform the contrast enhancement on the luminance component, while keeping the other color components unchanged. By doing so, the visual color of the resulting image would be preserved, allowing it to look natural.

$$\left.\begin{aligned} B &= I(1-S) \\ R &= I\left[1+\frac{S\cos H}{\cos(60^0-H)}\right] \\ G &= 3I-(R+B) \end{aligned}\right\} \qquad (10)$$

For example, the conversion formula from the HIS color space to the RGB color space is shown in Eqn (10), where$0^0 \leq H < 120^0$. Since the components H and S will not change after contrast enhancement, we can simplify in (10) as follows,

$$\left.\begin{aligned} B &= I.C_1 \\ R &= I.C_2 \\ G &= I.(3-C_1-C_2) \end{aligned}\right\} \qquad (11)$$

where C_1 and C_2 are both constants. Therefore, no matter how component I changes, the color correlation among R, G, and B will be well preserved. For the case of $120^0 \leq H < 240^0$ and $240^0 \leq H < 360^0$, the similar results could easily be obtained, which means that the proposed feature is robust against contrast enhancement. Here, we assume that there are no truncation and rounding errors during the color space conversion. In fact, such errors may affect the robustness slightly. Similar results can be obtained for other color space conversions, such as YCbCr and YIQ .

D. Video Sequence Matching

We analyzed the robustness of the proposed feature for each single video frame. In this subsection, we will extend the image feature to a video itself. To represent a video, we need to first extract key-frames from the video signal. Here, a key-frame is extracted per second. Assuming that a video V is composed of consecutive N key-frames, denoted by $V = \langle V_1, V_2, \dots, V_N \rangle$, then the feature of the video can be denoted as

$$H_V = H\langle H_{V1}, H_{V2}, \dots, H_{VN} \rangle \qquad (12)$$

where H_{Vi} denotes the color correlation feature of the video frame V_i, and i = 1, 2, ...,N.

Let V_q and V_t be two videos with N key-frames. The distance between these videos is defined as the average of the distances between the corresponding key-frames as follows:

$$d(H_{Vq}, H_{Vt}) = \frac{1}{N}\sum_{k=1}^{N} d(H_{Vq,k}, H_{Vt,k}) \qquad (13)$$

where H_{Vqk}, and $H_{Vt,k}$,k are the color correlation features of the k th key-frame in the videos V_q and V_t , respectively.

III. RESULT AND DISCUSSION

Nowadays video sequence matching is important in case of video forgery detection, sensor board, quick video search etc. Here we propose a method for video sequence matching based on invariance of color correlation. This is an effective method for video sequence matching. Also space complexity of our algorithm is satisfactory, which are very important for many real-time applications. We could completed the work with positive result. Support Vector Machines (SVM) is originally designed for binary classification. SVM is a binary classifier, that is, the class labels can only take two values: ± 1.Value can also be interpreted as a confidence value. By using SVM we can improve the accuracy in video sequence matching. We Obtained query videos are shown below,

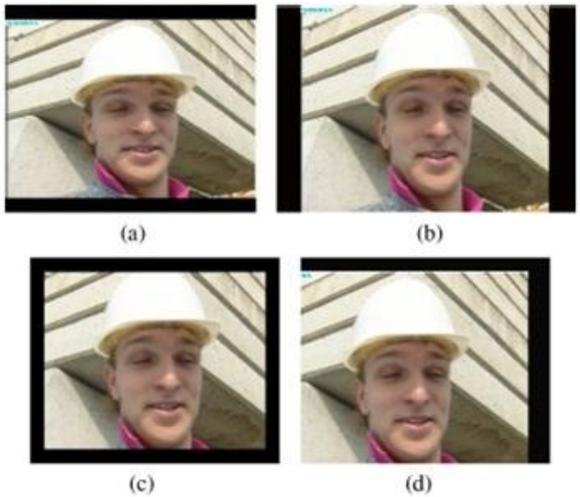
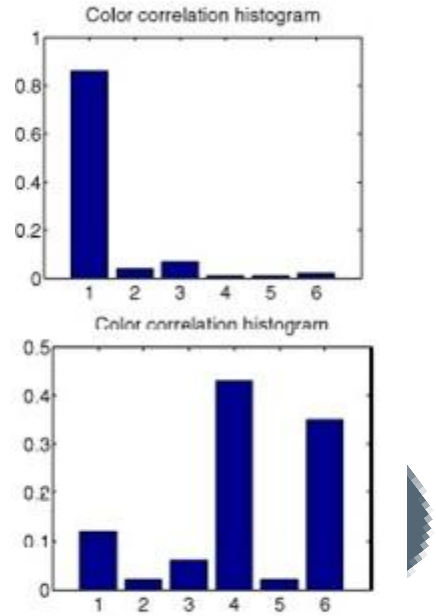


Fig.5. Four geometric operations. (a) Letter-box. (b) Pillar-box.(c) Cropping. (d) Shifting.

The colour correlations of the first frame of the two typical videos.

1. (b)

Fig.6. Two typical videos. (a) Foreman. (b) Container.



1. (b)

Fig.7. Distributions of the color correlations of the first frame of the two typical videos. (a) Foreman. (b) Container

CONCLUSION

In this paper we introduced a novel and very promising feature for video sequence matching based on the invariance of color correlation. Fingerprinting algorithm for video copyright detection system has been proposed. It can be used for copyright management and indexing applications. We first gave the robustness analysis of the proposed feature against the most commonly used operations. Furthermore, the proposed algorithm had satisfactory space complexity. Accuracy of video sequence matching is improved by multi-SVM. This work can be extended by taking large number of videos in a database with longer time duration.

REFERENCES

[1] F. Hartung and M. Kutter, “Multimedia watermarking techniques,” Proc. IEEE, vol. 87, no. 7, pp. 1079–1107, Jul. 1999.

[2] Y. Li and R. H. Deng, “Publicly verifiable ownership protection for relational databases,” in Proc. ACM Symp. Inform. Comput. Commun. Security, 2006, p. 78–89.

[3] X. Kang, J. Huang, and W. Zeng, "Efficient general print-scanning Resilient data hiding based on uniform log-polar mapping," IEEE Trans. Inf. Forensics Security, vol. 5, no. 1, pp. 1–12, Mar. 2010

[4] A. Joly, O. Buisson, and C. Frelicot, "Content-based copy retrieval using distortion-based probabilistic similarity search," IEEE Trans. Multimedia, vol. 9, no. 2, pp. 293–306, Feb. 2007.

[5] E. Chang, J. Wang, C. Li, and G. Wiederhold, "RIME: A replicated image detector for the world-wide web," Proc. SPIE Multimedia Storage Archiv. Syst., vol. 3527, pp. 58–67, Nov. 1998.

[6] R. Mohan, "Video sequence matching," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process., vol. 6. May 1998, pp. 3697–3700.

[7] C. Kim and B. Vasudev, "Spatiotemporal sequence matching for efficient video copy detection," IEEE Trans. Circuits Syst. Video Technol., vol. 15, no.1, pp. 127–132, Jan. 2005.

[8] L. Gao, Z. Li, and A. Katsaggelos, "An efficient video indexing and retrieval algorithm using the luminance field trajectory modeling," IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 10, pp. 1566–1570, Oct. 2009.

[9] P. Brasnett, S. Paschalakis, and M. Bober, "Recent developments on standardization of MPEG-7 visual signature tools," in Proc. IEEE Int. Conf. Multimedia Expo., Sep. 2010, pp. 1347–1352.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vision, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[11] M. Heikkil¨a, M. Pietik¨ainen, and C. Schmid, "Description of interest regions with local binary patterns," Pattern Recognit., vol. 42, no. 3, pp. 425–436, Mar. 2009.

[12] Y. Caspi, D. Simakov, and M. Irani, "Feature-based sequence-tosequence matching," Int. J. Comput. Vision, vol. 68, no. 1, pp. 53–64, Mar. 2006.

[13] B. Liu, Z. Li, M. Wang, and A. K. Katsaggelos, "In-sequence video duplicate detection with fast point-to-line matching," in Proc. IEEE Int. Conf. Image Process., Sep. 2010, pp. 1037–1040.

[14] S. Lee and C. D. Yoo, "Robust video fingerprinting for content-based video identification," IEEE Trans. Circuits Syst. Video Technol., vol. 18, no. 7, pp. 983–988, Jul. 2008.

[15] Vapnik, V. N.,1995, The Nature of Statistical Learning Theory. New York: Springer-Verlag.

[16] Boser, H., Guyon, I. M., & Vapnik, V. N., 1992. A training algorithm for optimal margin classifiers. In Haussler, D., Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory 144-152. Pittsburgh, PA: ACM Press.

[17] Foody, G. M., and Mathur, A., 2004, A relative evaluation of multiclass Image classification by support vector machines. IEEE Transactions on Geoscience and Remote Sensing, 42, 1335-1343.

[18] Yanqiang Lei, Weiqi Luo, Jiwu Huang "Video Sequence Matching Based on The Invariance of Color Correlation" IEEE transactions on circuits and systems for video technology, vol. 22, no. 9, september 2012.

[19] Mahesh Pal "Multiclass Approaches for Support Vector Machine Based Land Cover Classification" National Institute of Technolog Kurukshetra, 136119, Haryana (INDIA) mpce_pal@yahoo.co.uk