Exploring Image Bit Planes for Video Shot Boundary Detection

Anderson Carlos Sousa e Santos(✉) and Helio Pedrini

Institute of Computing, University of Campinas, Campinas, SP 13083-852, Brazil
anderson.santos@ic.unicamp.br

Abstract. The wide availability of digital content and the advances in multimedia technology have leveraged the development of efficient mechanisms for storing, indexing, transmitting, retrieving and visualizing video data. A challenging task is to automatically construct a compact representation of video sequences to help users comprehend the most relevant information present in their content. In this work, we develop and evaluate a novel method for detecting abrupt transitions based on bit planes extracted from the video frames. Experiments are conducted on two public datasets to demonstrate the effectiveness of the proposed method. Results are compared against other approaches of the literature.

Keywords: Bit planes · Cut detection · Video shot boundary
Video analysis

1 Introduction

The availability of several mobile devices, such as digital cameras, cell phones, and tablets, has enabled people to generate a large amount of multimedia content. The growth of digital data, particularly video streaming, has contributed to the advance of many knowledge domains, for instance, entertainment, education, telemedicine, robotics, surveillance and security.

In contrast to computationally expensive and time consuming task of manual annotation, the development of automatic and scalable strategies for storing, indexing, transmitting and retrieving multimedia data [15,18] is crucial to manage such massive growth of digital content.

Shot boundary detection plays an important role in temporal video segmentation [6,9,10,12,19], whose purpose is to partition the video content into meaningful units that constitute the most representative keyframes, known as shots. The summary of a video can be constructed from a set of keyframes that represent the shots.

Two categories of video transitions between shots are commonly defined: gradual and abrupt transitions. A gradual transition represents a smooth change over several frames, whereas an abrupt transition corresponds to a cut between one frame of a shot and its adjacent frame in the next shot.

M. Mendoza and S. Velastín (Eds.): CIARP 2017, LNCS 10657, pp. 593–600, 2018.
https://doi.org/10.1007/978-3-319-75193-1_71

There are various challenges associated with the video shot boundary detection task, such as illumination variability, camera motion, diversity of video genres, as well as the inherent subjectivity of the segmentation process. Although different video shot boundary detection methods have been proposed in the literature [2,3,8,16], two common steps are generally performed: (i) a similarity or dissimilarity measure is computed for each pair of consecutive frames and (ii) a cut is detected if the measure is higher than a specified threshold.

This work investigates and evaluates a novel shot boundary detection approach based on bit planes extracted from the video frames. An adaptive thresholding scheme is employed to determine if a transition is an abrupt shot boundary. Experiments are conducted on two public video benchmarks to show the effectiveness of the proposed method. Results are compared to other approaches available in the literature.

This paper is organized as follows. The proposed shot boundary detection method is detailed in Sect. 2. Experimental results are presented and discussed in Sect. 3. Finally, some final remarks and directions for future work are included in Sect. 4.

2 Methodology

The proposed video shot boundary detection method is based on the bit planes that compose a grayscale image. Since a pixel in a grayscale frame typically requires 1 byte of storage, 8 binary images can be produced by taking each bit plane independently. Figure 1 illustrates the bit planes extracted as binary images from a grayscale image (video frame). Each of these images describes different details about the pixel intensities, so a set of these images are used to extract features of the frame.

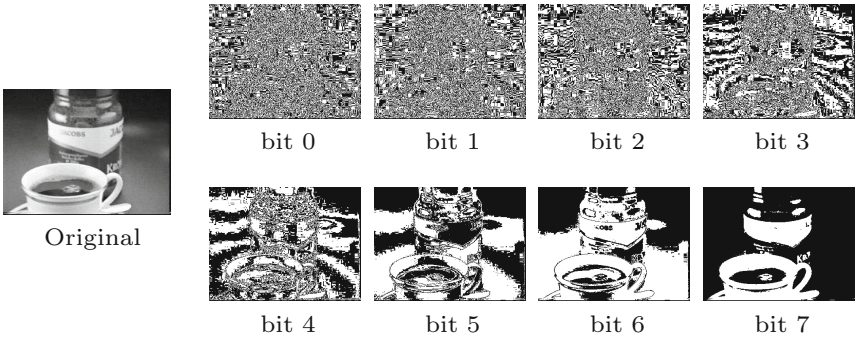

Fig. 1. Grayscale image and its decompositions in binary bit plane images.

Algorithm 1 describes how the features are extracted. For each bit plane image, a horizontal and vertical projection are calculated. These projections differ from the standard approach to summation of foreground pixels [7], whose

purpose is to describe an object present in the image, since the information here about background is also relevant.

Algorithm 1. Projection

input : Image f of size $N \times M$ pixels
Set of bit planes B
output: Feature vector W

for $b \in B$ **do**
  $f_b \leftarrow$ Extract bit plane b
  $V \leftarrow \emptyset$
  for $l \in N$ **do**
    $V_l = \sum_{j=0}^{M} -(-1)^{f(l,j)}$
  $H \leftarrow \emptyset$
  for $c \in M$ **do**
    $H_c = \sum_{i=0}^{N} -(-1)^{f_b(i,c)}$
  $W_b \leftarrow \{H, V\}$
return W

The background computation is important in the projection process since there are often cases with no foreground pixels, whose sum would result in zero. To avoid this, a transformation is applied to convert the binary image values from $[0, 1]$ to $[-1, 1]$. Thus, the projections correspond to the difference between the number of foreground and background pixels along each line and column, assigning positive values where the foreground exceeds the background, zero where they are equal, and negative values otherwise.

The concatenation of both horizontal and vertical projections constitutes the feature vector for a bit plane image. In order to compute the frame dissimilarities, a correlation-based distance (Eq. 1) between feature vectors is applied between equivalent bit planes for frames t and $t-1$. The total dissimilarity is calculated as the average distance in the set of bit planes. Algorithm 2 summarizes such procedure.

$$correlation(u, v) = 1 - \frac{(u - \overline{u}) \cdot (v - \overline{v})}{\|(u - \overline{u})\|_2 \|(v - \overline{v}\|_2} \tag{1}$$

where u and v are two feature vectors.

Finally, the vector of dissimilarities for the entire video is subject to a thresholding method [14], which is locally adaptive. It normalizes the vector into the range $[0, 1]$, takes a moving window over the vector, computes the median value

Algorithm 2. Dissimilarity

input : Video K
Set of bit planes B
output: Dissimilarity vector D

$D \leftarrow \emptyset$
for $f_i \in K$ **do**
  $w^{i-1} \leftarrow projection(f_{i-1}, B)$
  $w^{i} \leftarrow projection(f_{i}, B)$
  for $b \in B$ **do**
    $d_b \leftarrow correlation(w_b^{i-1}, w_b^i)$
  $D_i \leftarrow \frac{\sum d_b}{|B|}$
return D

within the window, and determines a shot boundary if the center of the window is the maximum value within the window and is greater than the median plus a fixed α. Figure 2 shows an example of dissimilarity vector after applying the thresholding process.

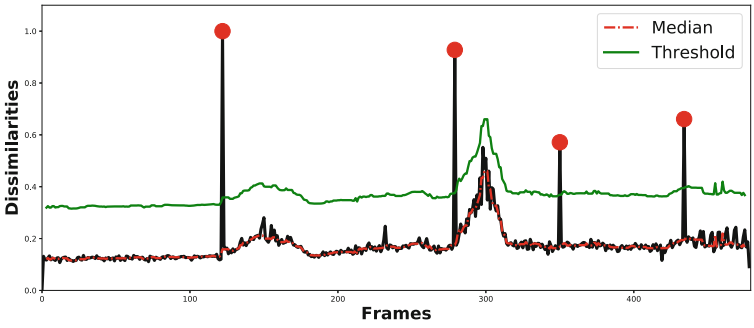

Fig. 2. Illustration of the thresholding technique.

3 Experimental Results

In this section, we present the experimental setup and analyze the results obtained on two data sets with our shot boundary detection method based on bit planes.

3.1 Data Sets

The TRECVID'2002 [17] consists of 18 video sequences with variability in quality, length, production style and noise level. The minimum number of cuts in a

video is 18 and the maximum is 163, there is an average of 83 cuts per video. The duration of the longest video is 28 min 48 s, whereas the shortest is 6 min 32 s. This data set was used as benchmark for the TREC Video Retrieval Evaluation competition in the shot boundary detection task. Unlike some recent competition data sets, this benchmark is freely available.

The VIDEOSEG'2004 [20] is composed of 10 video sequences from different genres, sizes, digitization quality and production effects. The video with maximum number of cuts is 87, minimum number of cuts is 0, where the average number of cuts is 28.

3.2 Evaluation Metrics

The evaluation guidelines available for the TRECVID competition are followed in this work, such that the results are reported in terms of precision, recall and harmonic mean (F_{score}), as expressed in Eqs. 2, 3 and 4, respectively. The precision indicates the capacity of the method in detecting only the real cuts, the recall indicates the capacity in finding all existing cuts, whereas the F_{score} is the harmonic mean between the other two metrics.

$$\text{Precision} = \frac{\sum_{f_i \in V} S(i) \in Cut \wedge i \in True\ Cut}{\sum_{f_i \in V} S(i) \in Cut} \tag{2}$$

$$\text{Recall} = \frac{\sum_{f_i \in V} S(i) \in Cut \wedge i \in True\ Cut}{\sum_{f_i \in V} i \in True\ Cut} \tag{3}$$

$$F_{score} = 2\,\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

where V is a Video and S a detection set.

As established in the TRECVID competition, we set a tolerance of $+5$ and -5 for the frame number of the detected transition. This is done due to possible changes in the numeration of a frame caused by different video encoders.

3.3 Parameter Settings

Similarly to other approaches [13,14], the parameters used in the thresholding stage are kept. The window size is set to 7 and $\alpha = 0.2$. The feature extraction method and the distance calculation require as parameters only the set of bit planes.

An empirical experiment is carried out to determine the most appropriate set with respect to F_{score}. A search is executed in the TRECVID'2002 data set by evaluating all combinations out of the 8 planes. Table 1 shows the best results for each number of planes used. It is important to highlight that the results reported for VIDEOSEG'2004 are based on the same set as chosen for TRECVID'2002.

As expected, the most significant bits occur in the combinations with higher accuracy. Nonetheless, the bit plane 6 presented the best results among the

Table 1. Results for different combinations of bit planes.

# Bit planes	F_{score} (%)	
	TRECVID'2002	VIDEOSEG'2004
1 {6}	83.04	82.24
2 {6, 7}	**85.75**	82.94
3 {5, 6, 7}	85.51	83.22
4 {0, 5, 6, 7}	84.97	94.62
5 {0, 2, 5, 6, 7}	83.48	94.35
6 {0, 1, 4, 5, 6, 7}	80.06	94.04
7 {0, 1, 2, 4, 5, 6, 7}	77.95	94.33
8 {0, 1, 2, 3, 4, 5, 6, 7}	74.13	**94.92**

most significant ones, such as bit plane 7. This can be explained by the fact that the more significant the bit is, the more susceptible it is with respect to small contrast variations, which also justify the favoritism of bit 0 in place of other most significant bits. When the three most significant bits are present, bit 0 works to regulate the average and avoid false transitions caused by illumination changes.

In order to have a trade-off in terms of accuracy for both data sets and have a single best set of bit planes, the results presented in the next section consider the set of 4 bit planes ({0, 5, 6, 7}).

3.4 Results

Table 2 shows the results on TRECVID'2002 data set when the dissimilarities between frames were calculated through color histograms with 32 bins and cross-correlation between pixels. Both baseline methods were implemented in our framework with an adaptive threshold, such that only the features and distances were varied.

Table 3 shows comparative results for the VIDEOSEG'2004 data set. The results for the baseline methods were extracted from the literature.

For the TRECVID'2002 data set, our method surpassed the other methods in terms of precision, while maintained a proper recall rate. The precision rate is particularly difficult in this data set since the video sequences have several different types of transitions and glitch effects, which may be confused as cuts. For the VIDEOSEG'2004 data set, our method did not obtain the best precision or recall rate individually, however, it achieved a proper trade-off between both, resulting in the highest F_{score} measure.

Table 2. Video cut detection results for TRECVID'2002.

Method	Precision (%)	Recall (%)	F_{score} (%)
Color Histogram (CH)	78.34	91.50	83.72
Normalized Cross-Correlation (NCC)	75.05	94.99	80.45
Our method	81.29	90.81	**84.97**

Table 3. Comparative results for VIDEOSEG'2004.

Method	Precision (%)	Recall (%)	F_{score} (%)
Feature-tracking [20]	87.35	96.06	90.79
Visual rhythm [5]	85.72	96.07	89.75
Discrete Cosine Transform (DCT) [1]	93.94	89.81	91.54
Minimum ratio [11]	97.10	86.30	90.95
Color Co-occurrence Matrices (CCM) [4]	83.22	87.26	83.58
Our method	94.30	95.30	**94.62**

4 Conclusions

The decomposition of a grayscale image into bit planes was explored in this work for the purpose of video shot boundary detection. The bit planes are employed in the process as binary images, where feature vectors are extracted from them by means of vertical and horizontal projections and used in the computation of dissimilarity between adjacent video frames.

Experiments were conducted on two data sets to assess the proposed methodology. It shows that features based on the bit planes image are relevant to compute similarity between images. It was capable to outperform other approaches present in the literature of cut detection, and even when using the same settings and conditions it surpass the histogram of colors.

Acknowledgments. The authors are thankful to São Paulo Research Foundation (grant FAPESP #2014/12236-1) and Brazilian Council for Scientific and Technological Development (grant CNPq #305169/2015-7 and scholarship #141647/2017-5) for their financial support.

References

1. Almeida, J., Leite, N.J., da S. Torres, R.: Rapid cut detection on compressed video. In: San Martin, C., Kim, S.-W. (eds.) CIARP 2011. LNCS, vol. 7042, pp. 71–78. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25085-9_8
2. Apostolidis, E., Mezaris, V.: Fast shot segmentation combining global and local visual descriptors. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6583–6587 (2014)

3. Birinci, M., Kiranyaz, S.: A perceptual scheme for fully automatic video shot boundary detection. Signal Process.: Image Commun. **29**(3), 410–423 (2014)
4. Cirne, M.V.M., Pedrini, H.: VISCOM: a robust video summarization approach using color co-occurrence matrices. Multimed. Tools Appl. **77**(1), 857–875 (2018). https://link.springer.com/article/10.1007%2Fs11042-016-4300-7
5. Guimarães, S., Patrocínio, Z., Paula, H., Silva, H.: A new dissimilarity measure for cut detection using bipartite graph matching. Int. J. Semant. Comput. **03**(02), 155–181 (2009)
6. Huang, T.S.: Image Sequence Analysis, vol. 5. Springer Science & Business Media, Heidelberg (1981). https://doi.org/10.1007/978-3-642-87037-8
7. Jain, R., Kasturi, R., Schunck, B.G.: Machine Vision. McGraw-Hill Inc., New York (1995)
8. Jiang, X., Sun, T., Liu, J., Chao, J., Zhang, W.: An adaptive video shot segmentation scheme based on dual-detection model. Neurocomputing **116**, 102–111 (2013)
9. Koprinska, I., Carrato, S.: Temporal video segmentation: a survey. Signal Process. Image Commun. **16**(5), 477–500 (2001)
10. Ngan, K.N., Li, H.: Video Segmentation and Its Applications. Springer Science & Business Media, New York (2011). https://doi.org/10.1007/978-1-4419-9482-0
11. Pal, G., Acharjee, S., Rudrapaul, D., Ashour, A.S., Dey, N.: Video segmentation using minimum ratio similarity measurement. Int. J. Image Min. **1**(1), 87–110 (2015)
12. Piramanayagam, S., Saber, E., Cahill, N.D., Messinger, D.: Shot boundary detection and label propagation for spatio-temporal video segmentation. In: Proceedings of SPIE, vol. 9405, pp. 94050D–94050D-7 (2015)
13. Sousa e Santos, A.C., Pedrini, H.: Adaptive video transition detection based on multiscale structural dissimilarity. In: Bebis, G., et al. (eds.) ISVC 2016. LNCS, vol. 10073, pp. 181–190. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50832-0_18
14. Sousa e Santos, A.C., Pedrini, H.: Video temporal segmentation based on color histograms and cross-correlation. In: Beltrán-Castañón, C., Nyström, I., Famili, F. (eds.) CIARP 2016. LNCS, vol. 10125, pp. 225–232. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52277-7_28
15. Tekalp, A.M.: Digital Video Processing, 2nd edn. Prentice Hall Press, Upper Saddle River (2015)
16. Tippaya, S., Sitjongsataporn, S., Tan, T., Chamnongthai, K., Khan, M.: Video shot boundary detection based on candidate segment selection and transition pattern analysis. In: IEEE International Conference on Digital Signal Processing, pp. 1025–1029. IEEE (2015)
17. TRECVID: TRECVID Data Availability (2017). http://trecvid.nist.gov/trecvid.data.html
18. Veltkamp, R., Burkhardt, H., Kriegel, H.P.: State-of-the-Art in Content-Based Image and Video Retrieval, vol. 22. Springer Science & Business Media, Heidelberg (2013). https://doi.org/10.1007/978-94-015-9664-0
19. Verma, M., Raman, B.: A hierarchical shot boundary detection algorithm using global and local features. In: Raman, B., Kumar, S., Roy, P.P., Sen, D. (eds.) Proceedings of International Conference on Computer Vision and Image Processing. AISC, vol. 460, pp. 389–397. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-2107-7_35
20. Whitehead, A., Bose, P., Laganiere, R.: Feature based cut detection with automatic threshold selection. In: Enser, P., Kompatsiaris, Y., O'Connor, N.E., Smeaton, A.F., Smeulders, A.W.M. (eds.) CIVR 2004. LNCS, vol. 3115, pp. 410–418. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27814-6_49